Oracle® Cloud

Exadata Database Service on Exascale Infrastructure

F92371-05

August 2024

Oracle Cloud Exadata Database Service on Exascale Infrastructure,

F92371-05

Primary Authors: Nirmal Kumar, Doug Williams

Contributors: Ravi Chennoju, Bryce Cracco, Yue Deng, Gayathri Dubagunta, Anil Kothuri, Natrajan Krishnamoorthi, Shuo Li, Ranjit Murali , Sanjay Narvekar, Babak Sanaee, Peter Sciarra, Lokesh Shrivastava, Nihal Shrivastava , Jason Tang, Bhuvaneswari Thiagarajan, Sujeet Vasudevan, Nayana Vishwa, Ravi Wijayaratne

Contents

1 Oracle Exadata Database Service on Exascale Infrastructure Overview

2 Preparing for Oracle Exadata Database Service on Exascale Infrastructure

5 Reference Guides for Oracle Exadata Database Service on Exascale Infrastructure

1

Oracle Exadata Database Service on Exascale Infrastructure Overview

This topic is an overview of the Oracle Exadata Database Service on Exascale Infrastructure formerly Exadata Cloud Service.

About Oracle Exadata Database Service on Exascale Infrastructure

Exadata Database Service on Exascale Infrastructure (ExaDB-XS) is Oracle’s newest deployment option for Exadata Database Service.ExaDB-XS provides a cloud service experience similar to Exadata Database Service on Dedicated Infrastructure. Customers can start with a small virtual machine (VM) cluster, and easily scale as needs grow. Oracle manages all of the physical infrastructure in a shared multitenancy infrastructure service model.

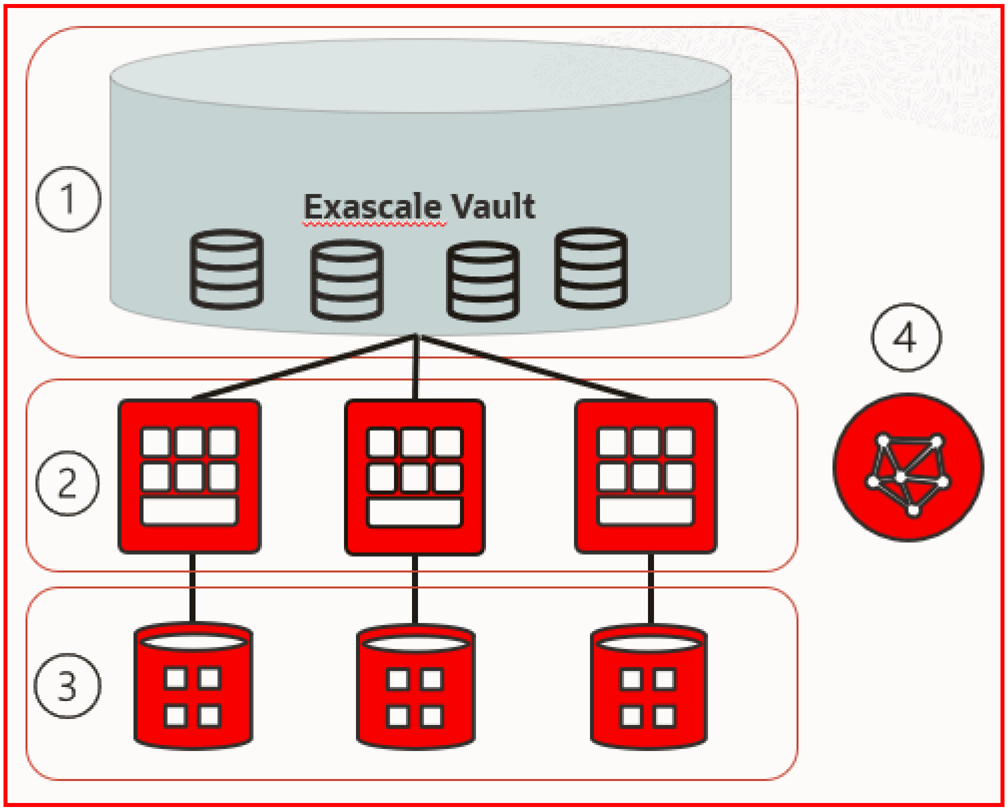
Note:

Exadata Database Service on Exascale Infrastructure is currently available in four regions: San Jose, Ashburn, Frankfurt, and Johannesburg. In regions that have multiple availability domains, only one availability domain is available at present.

Exascale is the underlying technology that serves as the foundation for this service. Exadata Database Service on Exascale Infrastructure is the next generation architecture of Oracle Exadata. It increases storage efficiency, simplifies database provisioning, and combines the extreme performance of Exadata smart software with the cost and elasticity benefits of modern clouds. Storage for database files resides in an Oracle Exadata Exascale Storage Vault. The Storage Vault provides high performance and scalable Exadata smart storage. Storage can be scaled online as needed, with a single command, and that storage becomes available for immediate use. Unlike Dedicated Infrastructure Exadata Database Service on Exascale Infrastructure does not require you to manage adding storage servers to the system, or manage storage allocations.

The following schematic overviews the overall high-level architecture of your VM Cluster and associated resources:

Figure 1-1 ExaDB-XS Architecture



The architecture consists of the following elements:

1. A single Exascale Vault, which provides storage for the databases
2. A set of VMs run on Oracle-managed multitenant physical database servers

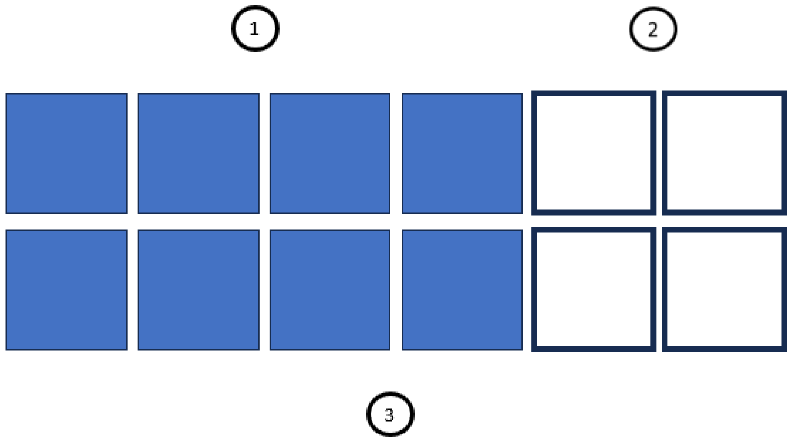
3. VM filesystems, which are centrally hosted by Oracle
4. A virtual client network (VCN), which provides client and backup network connectivity

The basic unit of consumption in ExaDB-XS is a VM cluster. To facilitate VM portability, Exascale hosts storage for VM file systems on shared storage that is fully managed by Oracle. Oracle can migrate VMs across a pool of physical servers, because the VM filesystems that host the database binaries do not reside on local physical servers. VMs are migrated automatically as required for maintenance, or in the event of a system failure. VMs can also be scaled vertically by changing the number of Elastic Compute Processing Unit (ECPU) units, and changing VM memory allocation. An ECPU is an abstracted measure of compute resources. ECPUs are based on the number of cores elastically allocated from a pool of compute servers. You need at least 8 ECPUs per VM to provision a VM Cluster. VMs can be scaled in increments of 4 ECPUs. For more information about ECPUs, see: Compute Models in Autonomous Database.

In addition to Enabled ECPUs, which are active in the VM, you can also add Additional Reserved ECPUs for your VM. These Additional Reserved ECPUs are physically allocated on a physical server for future scaling of the Enabled ECPUs on your VM, so that you can scale up to meet future workload demands without requiring a restart or relocation of your VM. This option helps to control costs for variable workloads, because database licensing is based on enabled ECPUs. Also, when you reserve additional ECPUs, memory is added to the VM Cluster based on the Total ECPU count. For this reason, reserving additional ECPUs also provides a way for you to provision additional memory without the licensing expense associated with the additional cores.

The following schematic illustrates conceptual details associated with scaling CPU and memory resources:

Figure 1-2   Core Reservation and Scaling



The illustration shows the following active and reserved cores in a VM:

1. Eight ECPUs, which are in use and active.
2. Four ECPUs in reserve, which are guaranteed to be available, and standing by, although not in use.

3. A total number of 12 ECPUs in the VM, which is the sum of Enabled cores and Reserved cores.

The total number of cores in a core reservation consists of the sum of Enabled cores and Reserved cores. To scale up your resources without restarting your systems, you can enable the reserved cores. If you want to further scale up your resources, you can add more ECPUs in units of four to your core reservation, and scale up your Enabled and Reserved cores, using a rolling restart as ECPUs are added.

Exascale also provides the benefits of redirect-on-write storage technology. With ExaDB-XS, you can provision thin clones of pluggable databases (PDBs) quickly, with space efficiency, because unchanged blocks are shared between parent and clone PDBs without being duplicated. This feature can be especially useful for development and test environments. You can create numerous thin clones of a PDB economically. For example, you can potentially give each of your developers their own PDB clone on which to work. Because Exadata Exascale has all of the performance advantages of Exadata, development environments provisioned with thin clones are representative of Exadata production environments, and not merely copies of the data.

Related Topics

- Compute Models in Autonomous Database

Accessing the Exadata Database Service on Exascale Infrastructure Using the OCI Console

Learn how to access the Oracle Exadata Database Service on Exascale Infrastructure (ExaDB-XS) service.

When the ExaDB-XS service is enabled in your OCI Tenancy you can sign in and select your tenancy region. Then, in the services menu, navigate to **Oracle Database**, and then to **Exadata Database Service on Exascale Infrastructure**. After you navigate to the main page for the service, notice that there are two main objects for this service: VM Clusters and Exascale Storage Vaults.

VM Clusters provide the compute environment where your Oracle Database instances will run. The databases themselves, which are accessed by those Oracle Database instances, are stored in the Exascale Storage Vault. Each VM Cluster has an Exascale Storage Vault assigned to it. You will create and associate the Exascale Storage Vault when creating the VM Cluster as a single, inline experience. However, if any lifecycle operations are then necessary for the Exascale Storage Vault (for example, scaling the total database storage to obtain more free space for expansion), then you complete those lifecycle operations from the Exascale Storage Vaults menu. For most other actions, including provisioning or management of databases, the correct starting point is the VM Clusters page.

Licensing Considerations for Oracle Exadata Database Service on Exascale Infrastructure

Subscription to Oracle Exadata Database Service on Exascale Infrastructure can include all of the required Oracle Database software licenses, or you can choose to bring Oracle Database software licenses that you already own to Oracle Exadata Database Service on Exascale Infrastructure.

If you choose to include Oracle Database software licenses in your Oracle Exadata Database Service on Exascale Infrastructure subscription, then the included licenses contain all of the features of Oracle Database Enterprise Edition, plus all of the database enterprise management packs, and all of the Enterprise Edition options, such as Oracle Database In-Memory and Oracle Real Application Clusters (Oracle RAC). Oracle Exadata Database Service on Exascale Infrastructure also comes with cloud-specific software tools that assist with administration tasks, such as backup, recovery, and patching.

Supported Database Edition and Versions for Oracle Exadata Database Service on Exascale Infrastructure

Oracle Exadata Database Service on Exascale Infrastructure databases require Enterprise Edition - Extreme Performance subscriptions or you can bring your own Oracle Enterprise Edition software licenses.

The Enterprise Edition - Extreme Performance provides all the features of Oracle Database Enterprise Edition, plus all the database enterprise management packs and all the Enterprise Edition options, such as Oracle Database In-Memory and Oracle Real Application Clusters (Oracle RAC).

At the time of release, Oracle Exadata Database Service on Exascale Infrastructure supports Oracle Database 23ai

For Oracle Database release and software support timelines, see Release Schedule of Current Database Releases (Doc ID 742060.1) in the My Oracle Support portal.

Related Topics

- Release Schedule of Current Database Releases (Doc ID 742060.1)

Subscription Types

Learn about the available subscription types for Oracle Exadata Database Service on Exascale Infrastructure

The available purchase models are as follows:

Pay as you Go

Pay As You Go (PAYG) pricing lets customers quickly provision services with no commitment, and they’re only charged for what they use. There’s no upfront commitment and no minimum service period. Any cloud infrastructure (IaaS) and platform (PaaS) services consumed are metered and billed based on that consumption. If, during the services period of your order, Oracle makes new IaaS and PaaS services available within your cloud services account, Oracle will notify you of any fees that would apply to their activation and use. For more details, see our complete price list.

Annual Universal Credits

Oracle Annual Universal Credits enables customers to have the flexibility to use any Oracle Cloud Infrastructure and platform services at any time, in any region, to deliver faster time to market. Customers can commit to an amount of Oracle Annual Universal Credits that can be applied towards the future usage of eligible Oracle IaaS and PaaS cloud services. This payment option offers a significant savings across cloud services, combining cost reduction and a predictable monthly spend with a ramp up period as you onboard your workloads.

Related Topics

- Universal Credit Pricing FAQ

Service Limits for Exadata Database Service on Exascale Infrastructure

Limits apply to virtual machine (VM) instance counts, total ECPU count, total local storage, and total High Capacity storage.

The limits set for Exadata Database Service on Exascale Infrastructure (ExaDB-XS) can be revised over time. The following table describes current service limits for ExaDB-XS resources:

Table 1-1 Service Limits for Exadata Database Service on Exascale Infrastructure

Limits Name	Description	Limits	Value
`exadbxs-vm-instance-base-count`	Exadata Database Service on Exascale Infrastructure - Instance Count	Number of VM Instances	4
`exadbxs-total-cpu-base-count`	Exadata Database Service on Exascale Infrastructure - Total ECPU Count	`TotalCpuCores`	64
`exadbxs-local-storage-base-gb`	Exadata Database Service on Exascale Infrastructure - Local Storage (GB)	Local Storage (in GB)	1500
`exadbxs-hc-storage-base-gb`	Exadata Database Service on Exascale Infrastructure - High Capacity Storage (GB)	High capacity storage (in GB)	2000

Metering Frequency and Per-Second Billing

See the Per-Second billing, minimums, and limitations on billing.

For each Oracle Exadata Database Service on Exascale Infrastructure virtual machine you provision, you are billed for the infrastructure for a minimum of 48 hours, and then by the second after that. Each ECPU you add to the system is billed by the second, with a minimum usage period of 1 minute.

Exadata Cloud Management Interfaces

Oracle Exadata Database Service on Exascale Infrastructure provides a variety of management interfaces to fit your use case and automation needs.

- Introduction to Exadata Cloud Management Interfaces
  The Exadata Cloud resources on Oracle Cloud Infrastructure (OCI) are created and managed through a variety of interfaces provided to fit your different management use cases.

- OCI Control Plane Interfaces for Oracle Exadata Database Service on Exascale Infrastructure
  The OCI control plane accepts input from the OCI APIs, the OCI Console, and custom interfaces built with kits, tools and plugins provided to facilitate development and simplify the management of OCI resources.
- Local VM Command-Line Interfaces
  In addition to the OCI REST-based APIs, CLI utilities located on the VM guests, provisioned as part of the VM clusters on the Exadata Cloud Infrastructure, are available to perform various lifecycle and administration operations.

Introduction to Exadata Cloud Management Interfaces

The Exadata Cloud resources on Oracle Cloud Infrastructure (OCI) are created and managed through a variety of interfaces provided to fit your different management use cases.

The various interfaces include:

- OCI Console interface and automation tools, see *Using the Console*
- Application Programming Interfaces (APIs)
- Command-Line Interfaces (CLIs)

The management interfaces are grouped into two primary categories:

- OCI Control Plane Interfaces
- Local Exadata Cloud VM CLIs

> **Note:**
>
> For more information and best practices on how these interfaces align for various Exadata Cloud database management use cases, refer to the folllowing My Oracle Support note: *Exadata Cloud API/CLI Alignment Matrix (Doc ID 2768569.1)*.

Related Topics

- Oracle Database console overview
- Using the Console
- Exadata Database Service API/CLI Alignment Matrix (Doc ID 2768569.1)

OCI Control Plane Interfaces for Oracle Exadata Database Service on Exascale Infrastructure

The OCI control plane accepts input from the OCI APIs, the OCI Console, and custom interfaces built with kits, tools and plugins provided to facilitate development and simplify the management of OCI resources.

The OCI APIs are typical REST APIs that use HTTPS requests and responses. The OCI Console, an intuitive, graphical interface for creating and managing your Exadata Cloud and other OCI resources, is one of the interfaces to the OCI APIs. When looking to develop automation utilizing the OCI APIs, a number of additional interfaces including: kits, tools and plug-ins, are provided to facilitate development and simplify the management of OCI resources. A subset of these APIs applies to Exadata Cloud resources and the containing infrastructure. Each of these various interfaces provide the same functionality, all calling the OCI APIs, and are provided to enable flexibility and choice depending on preference and use case.

- **Command Line Interface (CLI):** The OCI CLI is a small footprint tool that you can use on its own or with the Console to perform Exadata Cloud resource tasks and other OCI tasks. The CLI provides the same core functionality as the Console, plus additional commands. Some of these, such as the ability to run scripts, extend the Console's functionality.
- **Software Development Kits (SDK):** OCI provides SDKs to enable you to develop custom solutions for your Exadata Cloud and other OCI based services and applications.
- **DevOps Tools and Plug-ins:** These tools can simplify provisioning and managing infrastructure, enable automated processes and facilitate development. Tools include the OCI Terraform Provider used with Resource Manager and OCI Ansible Collection.
- **Cloud Shell:** Cloud Shell is a free-to-use, browser-based terminal, accessible from the OCI Console, that provides access to a Linux shell with pre-authenticated OCI CLI and other useful developer tools. You can use the shell to interact with Exadata Cloud and other OCI resources, follow labs and tutorials, and quickly run OCI CLI commands.
- **Documentation: Appendix and Reference:** This general reference shows how to configure the SDKs and other developer tools to integrate with Oracle Cloud Infrastructure services.
- **Documentation: REST APIs:** This complete reference provides details on the Oracle Cloud Infrastructure REST APIs, including descriptions, syntax, endpoints, errors, and signatures. Oracle Exadata Database Service on Exascale Infrastructure specific OCI REST APIs can be found throughout the documentation in the *Using the API* sections specific to each service:
  - *Using the API to Create Infrastructure Components*
  - *Using the API to Enable, Disable, or Update Database Management Service*
  - *Using the API to Manage Backup and Recovery*
  - *Using the API to manage Data Guard associations*
  - *Using the API to manage database software images*
  - *Using the API to manage Databases*
  - *Using the API to Manage Oracle Exadata Database Service on Exascale Infrastructure Instance*
  - *Using the API to Manage Oracle Database Home on Oracle Exadata Database Service on Exascale Infrastructure*
  - *Using the API to manage pluggable databases*
  - *Using the API to Patch an Oracle Exadata Database Service on Exascale Infrastructure Instance*
  - *Using the API to upgrade Databases*

Related Topics

- Command Line Interface (CLI)
- Software Development Kits
- DevOps Tools and Plug-ins
- Terraform Provider
- Resource Manager

- Ansible Collection
- Cloud Shell
- Appendix and Reference
- REST APIs

Local VM Command-Line Interfaces

In addition to the OCI REST-based APIs, CLI utilities located on the VM guests, provisioned as part of the VM clusters on the Exadata Cloud Infrastructure, are available to perform various lifecycle and administration operations.

The best practice is to use these utilities only when a corresponding Console command or OCI API is not available.

dbaascli: Use the `dbaascli` utility to perform various database lifecycle and administration operations on the Oracle Exadata Database Service on Exascale Infrastructure such as

- changing the password of a database user
- starting a database
- managing pluggable databases (PDBs)

These utilities are provided in addition to, and separate from, the OCI API-based interfaces listed above. To use the local VM command-line utilities, you must be connected to a virtual machine in an Exadata Cloud VM cluster and use the VM operating system user security, not the OCI user security, for execution. Most operations executed by these utilities sync their changes back to the OCI control plane using a process called `DB Sync`. However, there can be operations not synced with the control plane.

The cloud tooling software on the virtual machines, containing these CLI utilities, is automatically updated by Oracle on a regular basis.

2 Preparing for Oracle Exadata Database Service on Exascale Infrastructure

Review OCI as well as the site, network and storage requirements to prepare and deploy Oracle Exadata Database Service on Exascale Infrastructure in your data center.

- Oracle Cloud Infrastructure (OCI) Requirements for Oracle Exadata Database Service on Exascale Infrastructure
  Learn the basic concepts to get started using Oracle Cloud Infrastructure.
- Network Setup for Oracle Exadata Database Service on Exascale Infrastructure Instances
  This topic describes the recommended configuration for the VCN and several related requirements for the Oracle Exadata Database Service on Exascale Infrastructure instance.

Oracle Cloud Infrastructure (OCI) Requirements for Oracle Exadata Database Service on Exascale Infrastructure

Learn the basic concepts to get started using Oracle Cloud Infrastructure.

Oracle Exadata Database Service on Exascale Infrastructure is managed by the Oracle Cloud Infrastructure (OCI) control plane. The Oracle Exadata Database Service on Exascale Infrastructure resources are deployed in your OCI Tenancy.

Before you can provision Oracle Exadata Database Service on Exascale Infrastructure infrastructure, your Oracle Cloud Infrastructure tenancy must be enabled to use Oracle Exadata Database Service on Exascale Infrastructure. Review the information in this publication for further details.

The following tasks are common for all OCI deployments, refer to the links in the Related Topics to find the associated Oracle Cloud Infrastructure documentation.

- Getting Started with OCI.
  If you are new to OCI, learn the basic concepts to get started by following the *OCI Getting Started Guide* .
- Setting Up Your Tenancy.
  After Oracle creates your tenancy in OCI, an administrator at your company will need to perform some set up tasks and establish an organization plan for your cloud resources and users. The information in this topic will help you get started.
- Managing Regions
  This topic describes the basics of managing your region subscriptions.
- Managing Compartments
  This topic describes the basics of working with compartments.
- Managing Users
  This topic describes the basics of working with users.
- Managing Groups
  This topic describes the basics of working with groups.

- Required IAM Policy for Oracle Exadata Database Service on Exascale Infrastructure
  Review the identity access management (IAM) policy for provisioning Oracle Exadata Database Service on Exascale Infrastructure systems.

Related Topics

- OCI Getting Started Guide
- Setting Up Your Tenancy
- Managing Regions
- Managing Compartments
- Managing Users
- Managing Groups

Required IAM Policy for Oracle Exadata Database Service on Exascale Infrastructure

Review the identity access management (IAM) policy for provisioning Oracle Exadata Database Service on Exascale Infrastructure systems.

A **policy** is an IAM document that specifies who has what type of access to your resources. It is used in different ways:

- An individual statement written in the policy language
- A collection of statements in a single, named "policy" document, which has an Oracle Cloud ID (OCID) assigned to it
- The overall body of policies your organization uses to control access to resources

A **compartment** is a collection of related resources that can be accessed only by certain groups that have been given permission by an administrator in your organization.

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console, or the REST API with a software development kit (SDK), a command-line interface (CLI), or some other tool. If you try to perform an action, and receive a message that you don’t have permission, or are unauthorized, then confirm with your tenancy administrator the type of access you've been granted, and which compartment you should work in.

For administrators: The policy in "Let database admins manage DB systems" lets the specified group do everything with databases, and related database resources.

If you're new to policies, then see "Getting Started with Policies" and "Common Policies". If you want to dig deeper into writing policies for databases, then see "Details for the Database Service".

For more details on writing policies specific to Exadata Cloud@Customer resources see "Policy Details for Oracle Exadata Database Service on Exascale Infrastructure".

Related Topics

- Let database admins manage DB systems
- Getting Started with Policies
- Common Policies
- Policy Details for the Database Services

- Policy Details for Oracle Exadata Database Service on Exascale Infrastructure
  This topic covers details for writing policies to control access to Oracle Exadata Database Service on Exascale Infrastructure resources.

Network Setup for Oracle Exadata Database Service on Exascale Infrastructure Instances

This topic describes the recommended configuration for the VCN and several related requirements for the Oracle Exadata Database Service on Exascale Infrastructure instance.

Before you set up an Oracle Exadata Database Service on Exascale Infrastructure instance, you must set up a virtual cloud network (VCN) and other Networking service components.

- VCN and Subnets
  To launch an Oracle Exadata Database Service on Exascale Infrastructure VM cluster, you must have a Virtual Cloud Network and at least two subnets.
- Node Access to Object Storage: Static Route
- Service Gateway for the VCN
  Your VCN needs access to both Object Storage for backups and Oracle YUM repos for OS updates.
- Security Rules for the Oracle Exadata Database Service on Exascale Infrastructure
  This section lists the security rules to use with Oracle Exadata Database Service on Exascale Infrastructure.
- Ways to Implement the Security Rules
  Learn how to implement security rules within your VCN using the networking service.
- Network Requirements for Oracle Database Autonomous Recovery Service
  Oracle Database Autonomous Recovery Service requires a registered Recovery Service subnet dedicated to backup and recovery operations in your database virtual cloud network (VCN).

VCN and Subnets

To launch an Oracle Exadata Database Service on Exascale Infrastructure VM cluster, you must have a Virtual Cloud Network and at least two subnets.

To launch an Oracle Exadata Database Service on Exascale Infrastructure VM cluster, you must have a Virtual Cloud Network, at least two subnets and select the type of DNS resolver you will use:

- A VCN in the region where you want the Oracle Exadata Database Service on Exascale Infrastructure VM cluster
- At least two subnets in the VCN. The two subnets are:
  - Client subnet
  - Backup subnet
- Choose which method of DNS name resolution you will use. See *Choices for DNS in Your VCN*

In general, Oracle recommends using **regional subnets** , which span all **availability domains** in the region. For more information, see Overview of VCNs and Subnets.

You will create custom route tables for each subnet. You will also create security rules to control traffic to and from the client network and backup network of the Exadata compute nodes (for the Cloud VM cluster resource, nodes are called virtual machines). More information follows about those items.

- Option 1: Public Client Subnet with Internet Gateway
  This option can be useful when doing a proof-of-concept or development work.
- Option 2: Private Subnets
  Oracle recommends private subnets for a production system.
- Requirements for IP Address Space
  You must create a VCN with two subnets and ensure that there are enough addresses for the size of your VM cluster.
- Configuring a Static Route for Accessing the Object Store
- Setting Up DNS for an Oracle Exadata Database Service on Exascale Infrastructure Instance
  DNS lets you use host names instead of IP addresses to communicate with an Exadata Cloud Infrastructure instance.
- DNS: Short Names for the VCN, Subnets, and Oracle Exadata Database Service on Exascale Infrastructure instance
- Configure Private DNS
  Review the prerequisites needed to use Private DNS.

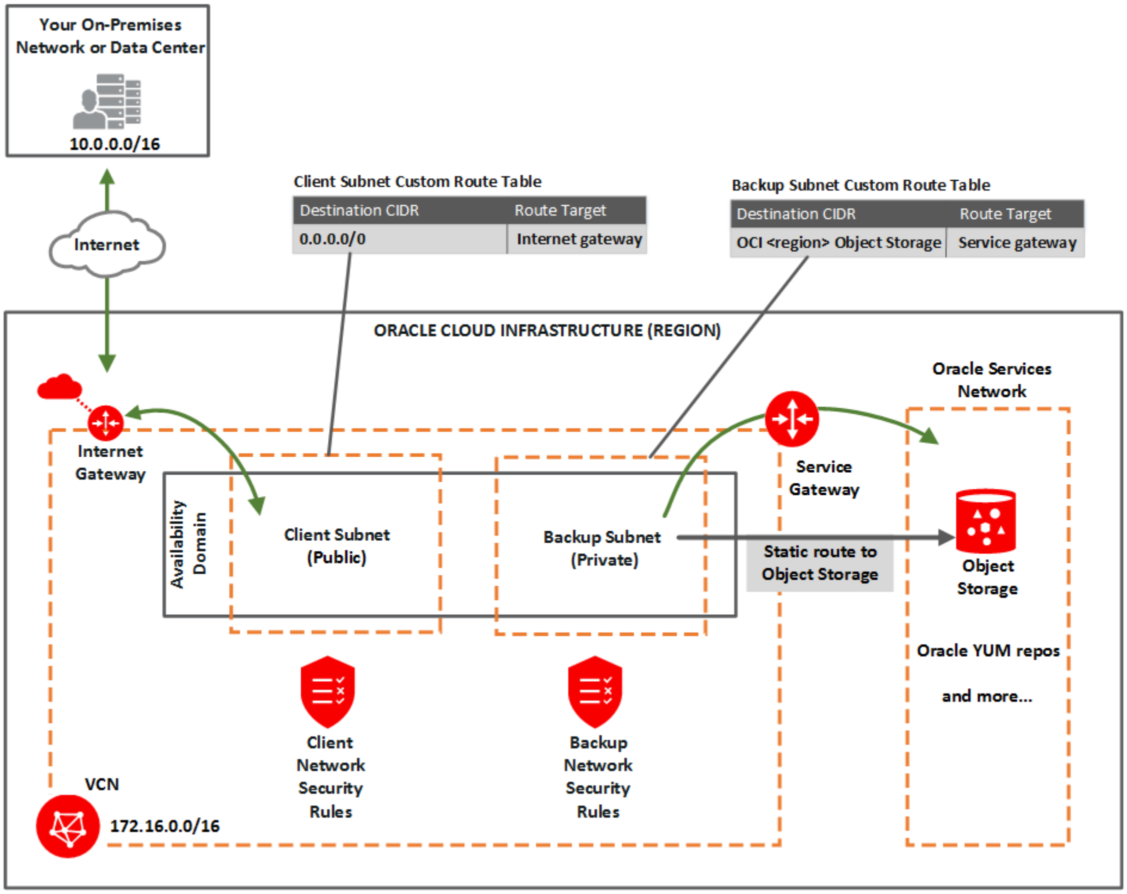
Related Topics

- Choices for DNS in Your VCN
- Overview of VCNs and Subnets
- About Regions and Availability Domains
- Availability Domains and Your VCN

Option 1: Public Client Subnet with Internet Gateway

This option can be useful when doing a proof-of-concept or development work.

You can use this setup in production if you want to use an **internet gateway** with the VCN, or if you have services that run only on a public network and need access to the database. See the following diagram and description.


Your On-Premises Network or Data Center
10.0.0.0/16
Internet
Client Subnet Custom Route Table
Backup Subnet Custom Route Table
ORACLE CLOUD INFRASTRUCTURE (REGION)
Oracle Services Network
Internet Gateway
Service Gateway
Availability Domain
Client Subnet (Public)
Backup Subnet (Private)
Static route to Object Storage
Object Storage
Oracle YUM repos
and more...
Client Network Security Rules
Backup Network Security Rules
VCN
172.16.0.0/16

Client Subnet Custom Route Table

Destination CIDR	Route Target
0.0.0.0/0	Internet gateway

Backup Subnet Custom Route Table

Destination CIDR	Route Target
OCI <region> Object Storage	Service gateway

You set up:

- Subnets:
  - *Public* client subnet (*public* means that the resources in the subnet can have public IP addresses at your discretion).
  - *Private* backup subnet (*private* means that the resources in the subnet cannot have public IP addresses and therefore cannot receive incoming connections from the internet).
- Gateways for the VCN:
  - Internet gateway (for use by the client subnet).
  - Service gateway (for use by the backup subnet) .
- Route tables:
  - Custom route table for the public client subnet, with a route for 0.0.0.0/0, and target = the internet gateway.
  - Separate custom route table for the private backup subnet, with a route rule for the service CIDR labels (see about CIDR labels under Overview of Service Gateways and Available Sevice CIDR labels, and target = the service gateway.
- Security rules to enable the desired traffic to and from the Exadata virtual machines compute nodes.
- Node Access to Object Storage: Static Route on the Exadata Cloud Service instance's compute nodes (to enable access to OCI services by way of the backup subnet).

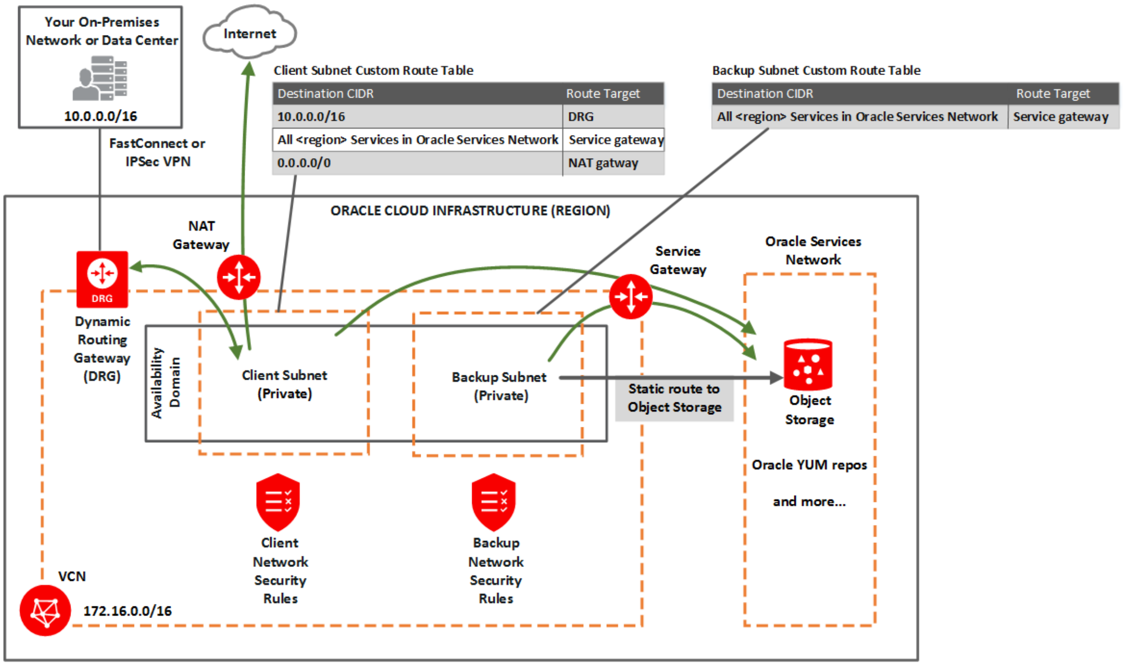
Important:

See this known issue for information about configuring route rules with **service gateway** as the target on route tables associated with public subnets.

Option 2: Private Subnets

Oracle recommends private subnets for a production system.

Both subnets are private and cannot be reached from the internet. See the following diagram and description.



Client Subnet Custom Route Table

Destination CIDR	Route Target
10.0.0.0/16	DRG
All <region> Services in Oracle Services Network	Service gateway
0.0.0.0/0	NAT gatway

Backup Subnet Custom Route Table

Destination CIDR	Route Target
All <region> Services in Oracle Services Network	Service gateway

You set up:

- Subnets:
  - *Private* client subnet.
  - *Private* backup subnet.
- Gateways for the VCN:
  - Dynamic routing gateway (DRG), with a FastConnect or IPSec VPN to your on-premises network (for use by the client subnet).
  - Service gateway For use by the backup and client subnets to reach OCI Services, such as Object Storage for backups, YUM for OS updates, IAM (Identitiy Access Management) and OCI Vault (KMS Integration) Also see Option 2: Service Gateway Access to Both Object Storage and YUM Repos.
  - NAT gateway(*optional*) For use by the client subnet to reach public endpoints not supported by the service gateway.
- Route tables:
  - Custom route table for the private client subnet, with the following rules:
    * A rule for the on-premises network's CIDR, and target = DRG.

        - A rule for the service CIDR label called **All <region> Services in Oracle Services Network**, and target = the service gateway. The *Oracle Services Network* is a conceptual network in Oracle Cloud Infrastructure that is reserved for Oracle services. The rule enables the client subnet to reach the regional Oracle YUM repository for OS updates. Also see Option 2: Service Gateway Access to Both Object Storage and YUM Repos.
        - Optionally, a rule for 0.0.0.0/0, and target = NAT gateway.
    - Separate custom route table for the private backup subnet, with one rule:
        - The same rule as for the client subnet: for the service CIDR label called **All <region> Services in Oracle Services Network**, and target = the service gateway. This rule enables the backup subnet to reach the regional Object Storage for backups.
- Security rules to enable the desired traffic to and from the Exadata nodes. See Security Rules for the Exadata Cloud Service instance.
- Optionally add a Static route on the compute nodes to other OCI services (for VM clusters, the virtual machines) to enable access, if the services are only reachable on the backup subnet and not via. the client subnet, e.g. when using a NAT Gateway.

Requirements for IP Address Space

You must create a VCN with two subnets and ensure that there are enough addresses for the size of your VM cluster.

> **Note:**
>
> IP addresses must not overlap, especially when Exadata Cloud Infrastructure instances (and thus VCNs) are in more than one region.
>
> If you're setting up VM Clusters (and thus VCNs) in more than one region, then ensure that the IP address space of the VCNs does not overlap. This is important if you want to set up disaster recovery with Oracle Data Guard.

For the client subnet, each node requires four IP addresses, and in addition, three addresses are reserved for Single Client Access Names (SCANs). For the backup subnet, each node requires three addresses. The Networking service reserves three IP addresses in each subnet.

Use the following formula to calculate the minimum number of IP addresses where the variable *n* is the number of VMs in the VM cluster:

The minimum number of client addresses = 4**n*+6

The minimum number of backup addresses = 3**n*+3

> **Note:**
>
> Allocating a larger space for the subnet than the minimum required (for example, at least /25 instead of /28) can reduce the relative impact of those reserved addresses on the subnet's available space. To plan for future growth, add addresses that you expect to require as you scale up your VM Cluster, not only the number of VMs you plan to provision for immediate need.

Configuring a Static Route for Accessing the Object Store

All the traffic in an Oracle Exadata Database Service on Exascale Infrastructure instance is, by default, routed through the data network. To route backup traffic to the backup interface (BONDETH1), you need to configure a static route on *each* of the compute nodes in the cluster.
For instructions, see Node Access to Object Storage: Static Route.

Setting Up DNS for an Oracle Exadata Database Service on Exascale Infrastructure Instance

DNS lets you use host names instead of IP addresses to communicate with an Exadata Cloud Infrastructure instance.

You can use the **Internet and VCN Resolver** (the DNS capability built into the VCN) as described in DNS in Your Virtual Cloud Network. Oracle recommends using a VCN Resolver for DNS name resolution for the client subnet. It automatically resolves the Swift endpoints required for backing up databases, patching, and updating the cloud tooling on an Exadata instance.

DNS: Short Names for the VCN, Subnets, and Oracle Exadata Database Service on Exascale Infrastructure instance

For the nodes to communicate, the VCN must use the Internet and VCN Resolver. The Internet and VCN resolver enables hostname assignment to the nodes, and DNS resolution of those hostnames by resources in the VCN.
The Internet and VCN resolver enables round robin resolution of the database's SCANs. It also enables resolution of important service endpoints required for backing up databases, patching, and updating the cloud tooling on an Oracle Exadata Database Service on Exascale Infrastructure instance. The Internet and VCN Resolver is the VCN's default choice for DNS in the VCN. For more information, see DNS in Your Virtual Cloud Network and also DHCP Options.

When you create the VCN, subnets, and Exadata, you must carefully set the following identifiers, which are related to DNS in the VCN:

- VCN domain label
- Subnet domain label
- Hostname prefix for the Oracle Exadata Database Service on Exascale Infrastructure instance's cloud VM cluster or DB system resource

These values make up the node's fully qualified domain name (FQDN):

```
<hostname_prefix>-######.<subnet_domain_label>.<vcn_domain_label>.oraclevcn.com
```

For example:

```
exacs-abcde1.clientpvtad1.acmevcniad.oraclevcn.com
```

In this example, you assign `exacs` as the hostname prefix when you create the cloud VM cluster or DB system. The Database service automatically appends a hyphen and a five-letter string with the node number at the end. For example:

- Node 1: `exacs-abcde1.clientpvtad1.acmevcniad.oraclevcn.com`
- Node 2: `exacs-abcde2.clientpvtad1.acmevcniad.oraclevcn.com`

- Node 3: `exacs-abcde3.clientpvtad1.acmevcniad.oraclevcn.com`
- And so on

Requirements for the hostname prefix:

- Recommended maximum: 12 characters. For more information, see the example under the following section, "Requirements for the VCN and subnet domain labels".
- Cannot be the string *localhost*

Requirements for the VCN and subnet domain labels:

- Recommended maximum: 14 characters each. The actual underlying requirement is a total of 28 characters *across both domain labels* (excluding the period between the labels). For example, both of these are acceptable: `subnetad1.verylongvcnphx` or `verylongsubnetad1.vcnphx`. For simplicity, the recommendation is 14 characters each.
- No hyphens or underscores.
- Recommended: include the region name in the VCN's domain label, and include the availability domain name in the subnet's domain label.
- In general, the FQDN has a maximum total limit of 63 characters. Here is a safe general rule:

  `<12_chars_max>-######.<14_chars_max>.<14_chars_max>.oraclevcn.com`

The preceding maximums are not enforced when you create the VCN and subnets. However, if the labels exceed the maximum, the Exadata deployment fails.

- DNS: Between On-Premises Network and VCN
  Oracle recommends using a private DNS resolver to enable the use of hostnames when on-premises hosts and VCN resources communicate with each other.

DNS: Between On-Premises Network and VCN

Oracle recommends using a private DNS resolver to enable the use of hostnames when on-premises hosts and VCN resources communicate with each other.

See Private DNS resolvers for information on creating and using private resolvers. For a reference architecture see Use private DNS in your VCN in the Oracle Architecture Center.

Configure Private DNS

Review the prerequisites needed to use Private DNS.

- Private view and private zone must be created before launching DB system provisioning. For details, see Private DNS resolvers.
- Forwarding to another DNS server should be set up beforehand in the DNS console. This can be done by going to the VCN's resolver, and creating the endpoint and then the rules. For details, see DNS in Your Virtual Cloud Network.
- Private zone's name cannot have more than 4 labels. For example, a.b.c.d is allowed while a.b.c.d.e is not.
- It is also required to add the private view to the resolver of the VCN. For details, see Adding a Private View to a Resolver.
- When provisioning a Exadata VM Cluster using Private DNS feature, Exadata needs to create reverse DNS zones in the compartment of Exadata VM Cluster. If the compartment has defined tags or tag defaults, additional policies related to managing tags are needed. For details, see:

- Required Permissions for Working with Defined Tags
- Required Permissions for Working with Tag Defaults

Node Access to Object Storage: Static Route

To be able to back up databases, and patch and update cloud tools on an Oracle Exadata Database Service on Exascale Infrastructure instance, you must configure access to Oracle Cloud Infrastructure Object Storage. Regardless of how you configure the VCN with that access (for example, with a service gateway), you may also need to configure a static route to Object Storage on each of the compute nodes in the cluster. This is only required if you are not using automatic backups. If you are using customized backups using the backup APIs, then you must route traffic destined for Object Storage through the backup interface (BONDETH1). This is not necessary if you are using the automatic backups created with the Console, APIs, or CLIs.

Caution:

You must configure a static route for Object Storage access on each compute node in an Oracle Exadata Database Service on Exascale Infrastructure instance if you *are not* creating automatic backups with the Console, APIs, or CLIs. Otherwise, attempts to back up databases, and patch or update tools on the system, can fail.

Note:

When you enable the first automatic backup for a database the static route configuration will be automatically done on the service.

If you want to patch the service before creating a database, the manual static route is required to be able to patch the GI or DB Home.

The static route may also be required to access other services (IAM, KMS) if these are not reachable via client subnet and only the backup subnet uses the setting to access all servcies within a region.

- Object Storage IP allocations
- To configure a static route for Object Storage access

Object Storage IP allocations

Oracle Cloud Infrastructure Object Storage uses the CIDR block IP range 134.70.0.0/16 for all regions.

As of June 1, 2018, Object Storage no longer supports the following discontinued IP ranges. Oracle recommends that you remove these older IP addresses from your access-control lists, firewall rules, and other rules after you have adopted the new IP ranges.

The **discontinued** IP ranges are:

- Germany Central (Frankfurt): 130.61.0.0/16
- UK South (London): 132.145.0.0/16
- US East (Ashburn): 129.213.0.0/16

- US West (Phoenix): 129.146.0.0/16

To configure a static route for Object Storage access

1. SSH to a compute node in the Oracle Exadata Database Service on Exascale Infrastructure instance.

   ```
   ssh -i <private_key_path> opc@<node_ip_address>
   ```

2. Log in as opc and then sudo to the root user. Use `sudo su -` with a hyphen to invoke the root user's profile.

   ```
   login as: opc

   [opc@dbsys ~]$ sudo su -
   ```

3. Identify the gateway configured for the BONDETH1 interface.

   ```
   [root@dbsys ~]# grep GATEWAY /etc/sysconfig/network-scripts/ifcfg-bondeth1
   |awk -F"=" '{print $2}'


   10.0.4.1
   ```

4. Add the following static rule for BONDETH1 to the `/etc/sysconfig/network-scripts/route-bondeth1` file:

   ```
   10.0.X.0/XX dev bondeth1 table 211
   default via <gateway> dev bondeth1 table 211
   134.70.0.0/17 via <gateway_from_previous_step> dev bondeth1
   ```

5. Restart the interface.

   ```
   [root@dbsys ~]# ifdown bondeth1; ifup bondeth1;
   ```

   The file changes from the previous step take effect immediately after the ifdown and ifup commands run.

6. Repeat the preceding steps on *each* compute node in the Oracle Exadata Database Service on Exascale Infrastructure instance.

Service Gateway for the VCN

Your VCN needs access to both Object Storage for backups and Oracle YUM repos for OS updates.

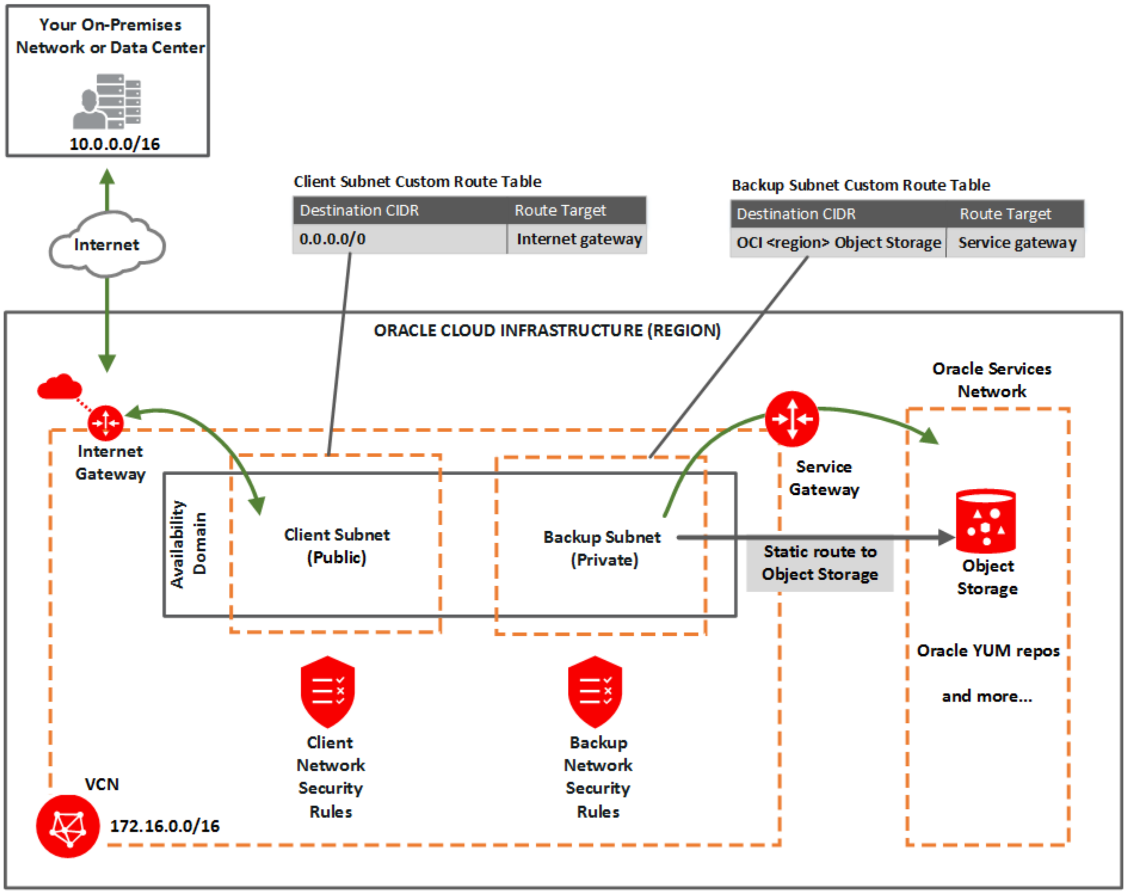
- Option 1: Service Gateway Access to OCI Services
  You configure the *backup subnet* to use the service gateway for access only to Object Storage.

- Option 2: Service Gateway Access to Both Object Storage and YUM Repos
  You configure *both the client subnet and backup subnet* to use the service gateway for access to the Oracle Services Network, which includes both Object Storage and the Oracle YUM repos.

Option 1: Service Gateway Access to OCI Services

You configure the *backup subnet* to use the service gateway for access only to Object Storage.

As a reminder, here's the diagram for option 1:



Client Subnet Custom Route Table

Destination CIDR	Route Target
0.0.0.0/0	Internet gateway

Backup Subnet Custom Route Table

Destination CIDR	Route Target
OCI <region> Object Storage	Service gateway

In general, you must:

- Perform the *tasks for setting up a service gateway on a VCN*, and specifically enable the service CIDR label called **OCI *<region>* Object Storage**.
- In the task for updating routing, add a route rule to the *backup* subnet's custom route table. For the destination service, use **OCI *<region>* Object Storage** and target = the service gateway.
- In the task for updating security rules in the subnet, perform the task on the *backup* network's network security group (NSG) or custom security list. Set up a security rule with the destination service set to **OCI *<region>* Object Storage**. See "Rule Required Specifically for the Backup Network" Rule Required Specifically for the Backup Network .

Related Topics

- Tasks for Setting Up a Service Gateway on a VCN in the Console

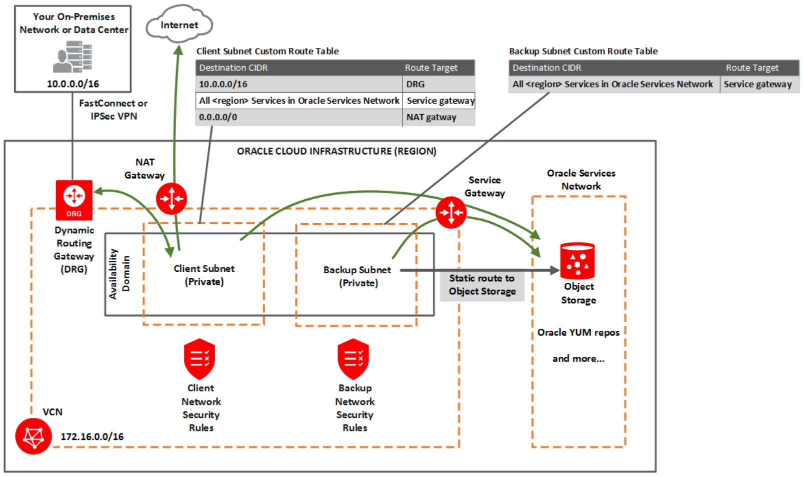
- Rule Required Specifically for the Backup Network
  The following security rule is important for the backup network because it enables the DB system to communicate with Object Storage through the service gateway (and optionally with the Oracle YUM repos if the client network doesn't have access to them).

Option 2: Service Gateway Access to Both Object Storage and YUM Repos

You configure *both the client subnet and backup subnet* to use the service gateway for access to the Oracle Services Network, which includes both Object Storage and the Oracle YUM repos.

> **Note:**
>
> See this known issues for information about accessing Oracle YUM services through the service gateway.

As a reminder, here's the diagram for option 2:



Client Subnet Custom Route Table

Destination CIDR	Route Target
10.0.0.0/16	DRG
All <region> Services in Oracle Services Network	Service gateway
0.0.0.0/0	NAT gatway

Backup Subnet Custom Route Table

Destination CIDR	Route Target
All <region> Services in Oracle Services Network	Service gateway

In general, you must:

- Perform the *tasks for setting up a service gateway on a VCN*, and specifically enable the service CIDR label called **All <*region*> Services in Oracle Services Network**.
- In the task for updating routing in each subnet, add a rule to each subnet's custom route table. For the destination service, use **All <*region*> Services in Oracle Services Network** and target = the service gateway.
- In the task for updating security rules for the subnet, perform the task on the *backup* network's network security group (NSG) or custom security list. Set up a security rule with the destination service set to **OCI <*region*> Object Storage**. See *Security Rules*. Note that the client subnet already has a broad egress rule that covers access to the YUM repos.

Here are a few additional details about using the service gateway for option 2:

- Both the client subnet and backup subnet use the service gateway, but to access different services. You cannot enable both the **OCI <*region*> Object Storage** service CIDR label and the **All <*region*> Services in Oracle Services Network** for the service gateway. To cover the needs of both subnets, you must enable **All <*region*> Services in Oracle Services Network** for the service gateway. The VCN can have only a single service gateway.

- Any route rule that targets a given service gateway must use an enabled service CIDR label and not a CIDR block as the destination for the rule. That means for option 2, the route tables for both subnets must use **All <*region*> Services in Oracle Services Network** for their service gateway rules.
- Unlike route rules, security rules can use either *any* service CIDR label (whether the VCN has a service gateway or not) or a CIDR block as the source or destination CIDR for the rule. Therefore, although the backup subnet has a route rule that uses **All <*region*> Services in Oracle Services Network**, the subnet can have a security rule that uses **OCI <*region*> Object Storage**. See *Security Rules for the Exadata Cloud Service instance*.

Related Topics

- Oracle Service Gateway
- Tasks for Setting up a Service Gateway on a VCN

Security Rules for the Oracle Exadata Database Service on Exascale Infrastructure

This section lists the security rules to use with Oracle Exadata Database Service on Exascale Infrastructure.

Security rules control the types of traffic allowed for the client network and backup network of the virtual machines. The rules are divided into three sections.

There are different ways to implement these rules. For more information, see Ways to Implement the Security Rules.

> **Note:**
>
> For X8M and X9M systems, Oracle recommends that all ports on the client subnet need to be open for ingress and egress traffic. This is a requirement for adding additional database servers to the system.

Rules Required for Both the Client Network and Backup Network

There are several general rules that enable essential connectivity for hosts in the VCN.

If you use security lists to implement your security rules, then be aware that the rules that follow are included by default in the default security list. Update or replace the list to meet your particular security needs. The two ICMP rules (general ingress rules 2 and 3) are required for proper functioning of network traffic within the Oracle Cloud Infrastructure environment. Adjust the general ingress rule 1 (the SSH rule) and the general egress rule 1 to allow traffic only to and from hosts that require communication with resources in your VCN.

General ingress rule 1: Allows SSH traffic from anywhere

- **Stateless:** No (all rules must be stateful)
- **Source Type:** CIDR
- **Source CIDR:** 0.0.0.0/0
- **IP Protocol:** SSH
- **Source Port Range:** All
- **Destination Port Range:** 22

General ingress rule 2: Allows Path MTU Discovery fragmentation messages

This rule enables hosts in the VCN to receive Path MTU Discovery fragmentation messages. Without access to these messages, hosts in the VCN can have problems communicating with hosts outside the VCN.

- **Stateless:** No (all rules must be stateful)
- **Source Type:** CIDR
- **Source CIDR:** 0.0.0.0/0
- **IP Protocol:** ICMP
- **Type:** 3
- **Code:** 4

General ingress rule 3: Allows connectivity error messages within the VCN

This rule enables the hosts in the VCN to receive connectivity error messages from each other.

- **Stateless:** No (all rules must be stateful)
- **Source Type:** CIDR
- **Source CIDR:** Your VCN's CIDR
- **IP Protocol:** ICMP
- **Type:** 3
- **Code:** All

General egress rule 1: Allows all egress traffic

- **Stateless:** No (all rules must be stateful)
- **Destination Type:** CIDR
- **Destination CIDR:** 0.0.0.0/0
- **IP Protocol:** All

Rules Required Specifically for the Client Network

The following security rules are important for the client network.

> **Important:**
>
> - Client ingress rules 1 and 2 only cover connections initiated from within the client subnet. If you have a client that resides *outside the VCN*, Oracle recommends setting up two *additional* similar rules that instead have the **Source CIDR** set to the public IP address of the client.
> - Client ingress rules 3 and 4 and client egress rules 1 and 2 allow TCP and ICMP traffic inside the client network and enable the nodes to communicate with each other. If TCP connectivity fails across the nodes, the Exadata cloud VM cluster or DB system resource fails to provision.

Client ingress rule 1: Allows ONS and FAN traffic from within the client subnet

The first rule is recommended and enables the Oracle Notification Services (ONS) to communicate about Fast Application Notification (FAN) events.

- **Stateless:** No (all rules must be stateful)
- **Source Type:** CIDR
- **Source CIDR:** Client subnet's CIDR
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range:** 6200
- **Description:** An optional description of the rule.

Client ingress rule 2: Allows SQL*NET traffic from within the client subnet

This rule is for SQL*NET traffic and is required in these cases:

- If you need to enable client connections to the database
- If you plan to use Oracle Data Guard
- **Stateless:** No (all rules must be stateful)
- **Source Type:** CIDR
- **Source CIDR:** Client subnet's CIDR
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range:** 1521
- **Description:** An optional description of the rule.

Client egress rule 1: Allows all TCP traffic inside the client subnet

- **Stateless:** No (all rules must be stateful)
- **Destination Type:** CIDR
- **Destination CIDR:** 0.0.0.0/0
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range:** 22
- **Description:** An optional description of the rule.

Client egress rule 2: Allows all egress traffic (allows connections to the Oracle YUM repos)

Client egress rule 3 is important because it allows connections to the Oracle YUM repos. It is redundant with the general egress rule in this topic (and in the default security list). It is optional but recommended in case the general egress rule (or default security list) is inadvertently changed.

- **Stateless:** No (all rules must be stateful)
- **Destination Type:** CIDR

- **Destination CIDR:** 0.0.0.0/0
- **IP Protocol:** All
- **Description:** An optional description of the rule.

Rule Required Specifically for the Backup Network

The following security rule is important for the backup network because it enables the DB system to communicate with Object Storage through the service gateway (and optionally with the Oracle YUM repos if the client network doesn't have access to them). It is redundant with the general egress rule in this topic (and in the default security list). It is optional but recommended in case the general egress rule (or default security list) is inadvertently changed.

Backup egress rule: Allows access to Object Storage

- **Stateless:** No (all rules must be stateful)
- **Destination Type:** Service
- **Destination Service:**
  - The service CIDR label called **OCI <*region*> Object Storage**
  - If the client network does not have access to the Oracle YUM repos, use the service CIDR label called **All <*region*> Services in Oracle Services Network**
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range:** 443 (HTTPS)
- **Description:** An optional description of the rule.

- Rules Required for Both the Client Network and Backup Network
  This topic has several general rules that enable essential connectivity for hosts in the VCN.
- Rules Required Specifically for the Client Network
  The following security rules are important for the client network.
- Rule Required Specifically for the Backup Network
  The following security rule is important for the backup network because it enables the DB system to communicate with Object Storage through the service gateway (and optionally with the Oracle YUM repos if the client network doesn't have access to them).
- Rules Required for Events Service
  The compute instance must have either a public IP address or a service gateway to be able to send compute instance metrics to the Events service.
- Rules Required for Monitoring Service
  The compute instance must have either a public IP address or a service gateway to be able to send compute instance metrics to the Monitoring service.

Rules Required for Both the Client Network and Backup Network

This topic has several general rules that enable essential connectivity for hosts in the VCN.

If you use security lists to implement your security rules, be aware that the rules that follow are included by default in the *default security list*. Update or replace the list to meet your particular security needs. The two ICMP rules (general ingress rules 2 and 3) are required for proper functioning of network traffic within the Oracle Cloud Infrastructure environment. Adjust the general ingress rule 1 (the SSH rule) and the general egress rule 1 to allow traffic only to and from hosts that require communication with resources in your VCN.

- General ingress rule 1: Allows SSH traffic from anywhere
- General ingress rule 2: Allows Path MTU Discovery fragmentation messages
- General ingress rule 3: Allows connectivity error messages within the VCN
  This rule enables the hosts in the VCN to receive connectivity error messages from each other.
- General egress rule 1: Allows all egress traffic

Related Topics

- default security list

General ingress rule 1: Allows SSH traffic from anywhere

- **Stateless:** No (all rules must be stateful)
- **Source Type:** CIDR
- **Source CIDR:** 0.0.0.0/0
- **IP Protocol:** SSH
- **Source Port Range:** All
- **Destination Port Range:** 22

General ingress rule 2: Allows Path MTU Discovery fragmentation messages

This rule enables hosts in the VCN to receive Path MTU Discovery fragmentation messages. Without access to these messages, hosts in the VCN can have problems communicating with hosts outside the VCN.

- **Stateless:** No (all rules must be stateful)
- **Source Type:** CIDR
- **Source CIDR:** 0.0.0.0/0
- **IP Protocol:** ICMP
- **Type:** 3
- **Code:** 4

General ingress rule 3: Allows connectivity error messages within the VCN

This rule enables the hosts in the VCN to receive connectivity error messages from each other.

- **Stateless:** No (all rules must be stateful)
- **Source Type:** CIDR
- **Source CIDR:** Your VCN's CIDR
- **IP Protocol:** ICMP
- **Type:** All
- **Code:** All

General egress rule 1: Allows all egress traffic

- **Stateless:** No (all rules must be stateful)
- **Destination Type:** CIDR

- **Destination CIDR:** 0.0.0.0/0
- **IP Protocol:** All

If the customer enables notification of Data Plane Guest VM Events, the default egress rule is sufficient.

Rules Required Specifically for the Client Network

The following security rules are important for the client network.

> **Note:**
>
> - For X8M systems, Oracle recommends that all ports on the client subnet need to be open for ingress and egress traffic. This is a requirement for adding additional database servers to the system.
> - Client ingress rules 1 and 2 only cover connections initiated from within the client subnet. If you have a client that resides *outside the VCN*, Oracle recommends setting up two *additional* similar rules that instead have the **Source CIDR** set to the public IP address of the client.
> - Client ingress rules 3 and 4 and client egress rules 1 and 2 allow TCP and ICMP traffic inside the client network and enable the nodes to communicate with each other. If TCP connectivity fails across the nodes, the Exadata cloud VM cluster or DB system resource fails to provision.

- Client ingress rule 1: Allows ONS and FAN traffic from within the client subnet
  The first rule is recommended and enables the Oracle Notification Services (ONS) to communicate about Fast Application Notification (FAN) events.
- Client ingress rule 2: Allows SQL*NET traffic from within the client subnet
  This rule is for SQL*NET traffic and is required in these cases:
- Client egress rule 1: Allows all TCP traffic inside the client subnet
  This rule is for SQL*NET traffic as noted.
- Client egress rule 2: Allows all egress traffic (allows connections to the Oracle YUM repos)
  Client egress rule 3 is important because it allows connections to the Oracle YUM repos.

Client ingress rule 1: Allows ONS and FAN traffic from within the client subnet

The first rule is recommended and enables the Oracle Notification Services (ONS) to communicate about Fast Application Notification (FAN) events.

- **Stateless:** No (all rules must be stateful)
- **Source Type:** CIDR
- **Source CIDR:** Client subnet's CIDR
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range:** 6200
- **Description:** An optional description of the rule.

Client ingress rule 2: Allows SQL*NET traffic from within the client subnet

This rule is for SQL*NET traffic and is required in these cases:

- If you need to enable client connections to the database
- If you plan to use Oracle Data Guard
- **Stateless:** No (all rules must be stateful)
- **Source Type:** CIDR
- **Source CIDR:** Client subnet's CIDR
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range:** 1521
- **Description:** An optional description of the rule.

Client egress rule 1: Allows all TCP traffic inside the client subnet

This rule is for SQL*NET traffic as noted.

- **Stateless:** No (all rules must be stateful)
- **Destination Type:** CIDR
- **Destination CIDR:** 0.0.0.0/0
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range:** 22
- **Description:** An optional description of the rule.

Client egress rule 2: Allows all egress traffic (allows connections to the Oracle YUM repos)

Client egress rule 3 is important because it allows connections to the Oracle YUM repos.

It is redundant with the general egress rule 1: Allow all egress traffic (and in the *default security list*). It is optional but recommended in case the general egress rule (or default security list) is inadvertently changed.

- **Stateless:** No (all rules must be stateful)
- **Destination Type:** CIDR
- **Destination CIDR:** 0.0.0.0/0
- **IP Protocol:** All
- **Description:** An optional description of the rule.

Related Topics

- default security list

Rule Required Specifically for the Backup Network

The following security rule is important for the backup network because it enables the DB system to communicate with Object Storage through the service gateway (and optionally with the Oracle YUM repos if the client network doesn't have access to them).

It is redundant with the *general egress rule 1: Allows all egress traffic* in (and in the). It is optional but recommended in case the general egress rule (or default security list) is inadvertently changed.

- Backup egress rule: Allows access to Object Storage

Related Topics

- General egress rule 1: Allows all egress traffic
- default security list

Backup egress rule: Allows access to Object Storage

- **Stateless:** No (all rules must be stateful)
- **Destination Type:** Service
- **Destination Service:**
  - The service CIDR label called **OCI <*region*> Object Storage**
  - If the client network does not have access to the Oracle YUM repos, use the service CIDR label called **All <*region*> Services in Oracle Services Network**
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range:** 443 (HTTPS)
- **Description:** An optional description of the rule.

Rules Required for Events Service

The compute instance must have either a public IP address or a service gateway to be able to send compute instance metrics to the Events service.

The default egress rules are sufficient to to allow the compute instance to send compute instance metrics to the Events service.

If the instance does not have a public IP address, set up a service gateway on the virtual cloud network (VCN). The service gateway lets the instance send compute instance metrics to the Events service without the traffic going over the internet. Here are special notes for setting up the service gateway to access the Events service:

- When creating the service gateway, enable the service label called **All <region> Services in Oracle Services Network**. It includes the Events service.
- When setting up routing for the subnet that contains the instance, set up a route rule with **Target Type** set to **Service Gateway**, and the **Destination Service** set to **All <region> Services in Oracle Services Network**.

  For detailed instructions, see Access to Oracle Services: Service Gateway.

Rules Required for Monitoring Service

The compute instance must have either a public IP address or a service gateway to be able to send compute instance metrics to the Monitoring service.

The default egress rules are sufficient to to allow the compute instance to send compute instance metrics to the Monitoring service.

If the instance does not have a public IP address, set up a service gateway on the virtual cloud network (VCN). The service gateway lets the instance send compute instance metrics to the Monitoring service without the traffic going over the internet. Here are special notes for setting up the service gateway to access the Monitoring service:

- When creating the service gateway, enable the service label called **All <region> Services in Oracle Services Network**. It includes the Monitoring service.
- When setting up routing for the subnet that contains the instance, set up a route rule with **Target Type** set to **Service Gateway**, and the **Destination Service** set to **All <region> Services in Oracle Services Network**.

  For detailed instructions, see Access to Oracle Services: Service Gateway.

Ways to Implement the Security Rules

Learn how to implement security rules within your VCN using the networking service.

The Networking service offers two ways to implement security rules within your VCN:

- Network security groups
- Security lists

For a comparison of the two methods, see Comaprison of Security Lists and Network Security Groups.

- If you use network security groups
- If you use security lists
  If you choose to use security lists, here is the recommended process:

If you use network security groups

If you choose to use network security groups (NSGs), then here is the recommended process:

1. Create an NSG for the client network. Add the following security rules to that NSG:
   - The rules listed in "Rules Required for Both the Client Network and Backup Network"
   - The rules listed in "Rules Required Specifically for the Client Network"
2. Create a separate NSG for the backup network. Add the following security rules to that NSG:
   - The rules listed in "Rules Required for Both the Client Network and Backup Network"
   - The rules listed in "Rules Required Specifically for the Client Network"
3. As the database administrator, when you create an Exadata instance on Exadata Database Service on Exascale Infrastructure, you must choose several networking components (for example, which VCN and subnets to use):

- When you choose the client subnet, you can also choose which NSG or NSGs to use. Choose the client network's NSG.
- When you choose the backup subnet, you can also choose which NSG or NSGs to use. Choose the backup network's NSG.

You can instead create a separate NSG for the general rules. Then when database administrators choose which NSGs to use for the client network, they choose both the general NSG and the client network NSG. Similarly for the backup network, they choose both the general NSG and the backup network NSG.

If you use security lists

If you choose to use security lists, here is the recommended process:

If you choose to use security lists, here is the recommended process:

1. Configure the client subnet to use the required security rules:
   a. Create a custom security list for the client subnet and add the rules listed in Rules Required Specifically for the Client Network.
   b. Associate the following two security lists with the client subnet:
      - VCN's *default security list* with all its default rules. This automatically comes with the VCN. By default it contains the rules in Rules Required for Both the Client Network and Backup Network.
      - The new custom security list you created for the client subnet.
2. Configure the backup subnet to use the required security rules:
   a. Create a custom security list for the backup subnet and add the rules listed in Rule Required Specifically for the Backup Network.
   b. Associate the following two security lists with the backup subnet:
      - VCN's *default security list* with all its default rules. This automatically comes with the VCN. By default it contains the rules in Rules Required for Both the Client Network and Backup Network.
      - The new custom security list you created for the backup subnet.

Later when the database administrator creates the Exadata Cloud Service instance, they must choose several networking components. When they select the client subnet and backup subnet that you've already created and configured, the security rules are automatically enforced for the nodes created in those subnets.

> **WARNING:**
>
> **Do not remove the default egress rule from the default security list**. If you do, make sure to instead include the following replacement egress rule in the client subnet's security list:
>
> - **Stateless:** No (all rules must be stateful)
> - **Destination Type:** CIDR
> - **Destination CIDR:** 0.0.0.0/0
> - **IP Protocol:** All

Network Requirements for Oracle Database Autonomous Recovery Service

Oracle Database Autonomous Recovery Service requires a registered Recovery Service subnet dedicated to backup and recovery operations in your database virtual cloud network (VCN).

To use Recovery Service for backups, follow the steps outlined in *Configuring your Tenancy for Recovery Service*.

- Create a Service Gateway to Object Storage
  In the OCI Console, create a service gateway to Object Storage. The service gateway is required for automation updates and configuration metadata.

Related Topics

- Configuring your Tenancy for Recovery Service

Create a Service Gateway to Object Storage

In the OCI Console, create a service gateway to Object Storage. The service gateway is required for automation updates and configuration metadata.

1. Open the navigation menu. Click **Networking**, and then click **Virtual Cloud Networks**.
2. Select the VCN where your database services to be backed up are located.
3. On the resulting Virtual Cloud Network Details page, under **Resources**,click **Service Gateways**.
4. Click **Create Service Gateway** and provide the following details.
   a. **Name**: A descriptive name for the service gateway. It doesn't have to be unique. Avoid entering confidential information.
   b. **Compartment**: The compartment where you want to create the service gateway, if different from the compartment you're currently working in.
   c. **Services**: Select the service CIDR Label, `All <region> Services in Oracle Services Network` from the drop-down list.
   d. **Tags:** (advanced option) If you have permissions to create a resource, then you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see *Resource Tags*. If you are not sure whether to apply tags, skip this option (you can apply tags later) or ask your administrator.
5. Click **Create Service Gateway**.

   Wait for the gateway to be created before proceeding to the next step.
6. Under **Resources**, click **Route Tables**.

   Route Table Association: You can associate a specific VCN route table with this gateway. If you associate a route table, afterward the gateway must always have a route table associated with it. You can modify the rules in the current route table or replace them with another route table.
7. Click the **Route Table** name that is being used by the subnet for Recovery Service.
8. In the resulting Route Table Details page, click **Add Route Rules** in the **Route Rules** section.

When you configure a service gateway for a particular service CIDR label, you must also create a route rule that specifies that label as the destination and the target as the service gateway. You do this for each subnet that needs to access the gateway.

9. In the resulting Add Route Rules dialog, enter the following details:
   1. **Target Type**: Service Gateway.
   2. **Destination Service**: The service CIDR label that is enabled for the gateway. `All <region> Services in Oracle Services Network`
   3. **Target Service Gateway**: Select the name that you provided in step 4.
   4. **Description**: An optional description of the rule.
10. Click **Add Route Rules**.

Related Topics

- Resource Tags

3

Getting Started with Oracle Exadata Database Service on Exascale Infrastructure Deployment

After completing the preparation tasks in Preparing for Oracle Exadata Database Service on Exascale Infrastructure, get started with deploying your Oracle Exadata Database Service on Exascale Infrastructure system following these procedures.

- Tagging Oracle Exadata Database Service on Exascale Infrastructure Resources
  Tagging is a powerful foundational service for Oracle Cloud Infrastructure (OCI) that enables users to search, control access, and do bulk actions on a set of resources based on the tag.
- Restarting a VM for Planned Maintenance
  To facilitate maintenance of Oracle Exadata Database Service on Exascale Infrastructure virtual machines (VM), Oracle notifies you of planned VM restarts.
- Connecting to an Oracle Exadata Database Service on Exascale Infrastructure VM
  Learn how to connect to an Oracle Exadata Database Service on Exascale Infrastructure virtual machine (VM) using SSH or SQL Developer.
- Capacity Limits for Exadata Database Service on Exascale Infrastructure
  To understand the resource capacity of the ExaDB-XS service, review these tables
- Best Practices for Oracle Exadata Database Service on Exascale Infrastructure VMs
  Oracle recommends that you follow these best practice guidelines to ensure the manageability of your Oracle Exadata Database Service on Exascale Infrastructure virtual machines (VMs).

Tagging Oracle Exadata Database Service on Exascale Infrastructure Resources

Tagging is a powerful foundational service for Oracle Cloud Infrastructure (OCI) that enables users to search, control access, and do bulk actions on a set of resources based on the tag.

Importance of Tagging

Using the Oracle Cloud Infrastructure (OCI) tagging system, you can tag resources in accordance with your organizational scheme, which enables you to group resources, manage costs, and give insights into usage. Tags also help you to build a governance model around security and Maximum Availability Architecture (MAA). As your organization expands its cloud footprint, it can become challenging to keep track of the deployment architectures, security best practices, MAA, application tier, and so on. Using metadata tags to identify workload attributes can help keep up with the security and availability of your tenancy without cost overruns.

To enable customers to manage OCI resources securely and cost-effectively, Oracle provides a set of predefined tags in line with best practices for tagging resources. These tags are grouped into two namespaces, the `oracleStandard` namespace, and the

`OracleApplicationName` namespace. You can think of a tag namespace as a container for your tag keys.

Consider a scenario where your organization has multiple cloud resources such as Exadata Infrastructure, VM Cluster, DB Home, Oracle Database and VM Cluster Networks across multiple compartments in your tenancy. Suppose you want to track these cloud resources for specific purposes, report on them, or take bulk actions. In that case, you will need a system that lets you group these resources based on different criteria such as environment, criticality, target users, application, and so on. You can achieve this by applying appropriate tags to these resources.

For example, you can tag all resources in your development stack with `Oracle-Standard.Environment=Dev` or for a business-critical application stack set `Oracle-Standard.Criticality=High` or `Extreme`. In the event of service disruptions due to various reasons, you would then be able to quickly identify all OCI resources associated with an application or business function, or be able to separate critical and non-critical workloads.

Tagging can also help you to deploy optimized configurations based on workload attributes identified via tags. For example, database deployments for the PeopleSoft application require a specific configuration. Setting the `ApplicationName` and `AppMajorVersion` tags while deploying an Oracle Database can ensure that the database is configured and ready for the particular application (in this case, PeopleSoft) out of the box.

Moreover, integration with the Cloud Advisor OCI service can provide you with direct, deep insight into how well your cloud services adhere to the corporate guidelines and help your management govern with a vision. See *Cloud Advisor Overview* for more details.

Adding Tags

You can tag resources using the Oracle Cloud Infrastructure (OCI) console, command-line interface, or SDK.

There are many cloud resources that can be tagged in an Oracle Exadata Database Service on Exascale Infrastructure deployment. Exadata Infrastructure, VM Cluster, DB Home, Oracle Database, Autonomous Exadata VM Cluster, Autonomous Container Database, Autonomous Database, and VM Cluster Networks are some of them. Tags can either be applied while creating the resources or modified later. For example, you can apply tags to an Autonomous Container Database (ACD) while provisioning the ACD or add them later from its Details page.

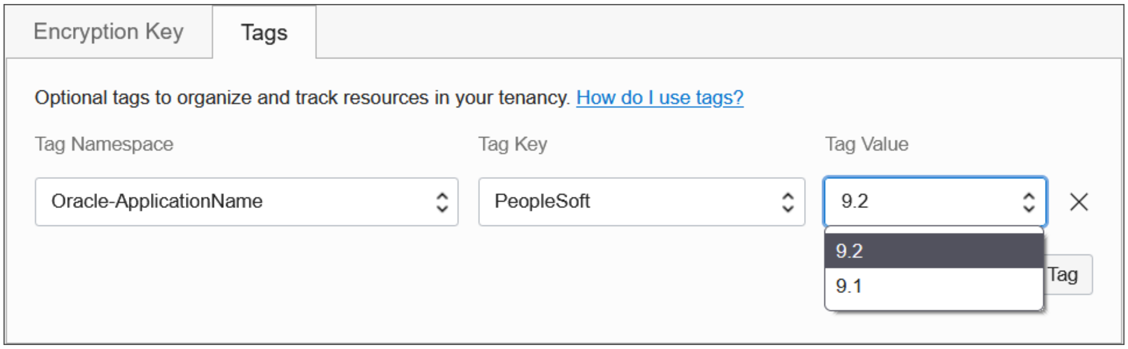
See *How Tagging Works* for more details on using tags. Tagging integrates with Oracle Cloud Infrastructure authorization system. You can use IAM policy controls to enable delegation or restriction of tag manipulation. See *Authentication and Authorization* to learn about the permissions required to work with defined and free-form tags. (Required) Enter introductory text here, including the definition and purpose of the concept.

> **Tip:**
>
> For a "try it out" tutorial that demonstrates implementing tags in Oracle Autonomous Database, refer to *Lab 14: Oracle Standard Tags* in *Oracle Autonomous Database Dedicated for Fleet Administrators Workshop* on Oracle LiveLabs.

Your tenancies come with a library of standard tags that would apply to most resources. These tags are currently available as a set of Tag Namespaces that your governance administrators can deploy. OCI best practices recommend applying these tags to all resources a standard tag can be applied to. Besides reporting and governance, OCI service automation can deliver workload-specific optimizations based on standard tag values.

For example, database deployments for the PeopleSoft application require a specific configuration. By setting the appropriate application tag key in the `Oracle-ApplicationName` tag namespace while deploying an Autonomous Database, can ensure that the database is configured ready for the particular application (in this example, PeopleSoft) out of the box.

Figure 3-1 Tagging Example



Optional tags to organize and track resources in your tenancy. How do I use tags?

Tag Namespace: Oracle-ApplicationName

Tag Key: PeopleSoft

Tag Value: 9.2

Oracle Standard Tags

Your tenancy governance administrators can deploy the standard tags at the tenancy level. Your administrators can also mark certain tags as required, thereby enforcing tags on resources in those compartments. The following are the standard tags defined in the namespace called `OracleStandard`. For more information about importing standard tags, see *To import standard tags* under the *Managing Tag Namespaces* section.

Table 3-1 Oracle Standard Tags

Tag Key	Tag Value Options	Description
`OracleStandard.Criticality`	• Extreme • High • Medium • Low	Enables tiering of resources in line with corporate application classification standards. Customer governance can use this tag for reporting and ensuring resources are configured as per the guideline for the tier they belong to. For example, a database resource with `OracleStandard.Criticality` set to Extreme or High may require the highest availability SLA and may need to be configured with Autonomous Data Guard.

Table 3-1 (Cont.) Oracle Standard Tags

Tag Key	Tag Value Options	Description
`OracleStandard.Environment`	• Dev • Test • Prod • Pre-Prod • Staging • Trial • Sandbox • User Testing	Denotes a resource lifecycle. In the case of databases, it helps determine consolidation density, database distribution across containers, set maintenance plans, and manage clones.
`OracleStandard.Sensitivity`	• Public • Internal • Sensitive • Highly Sensitive • Extremely Sensitive	An application or database classification tag. `OracleStandard.Sensitivity` set to Highly Sensitive may indicate that an access control list or certain Network Security Group (NSG) enforcement is mandatory to restrict access.
`OracleStandard.Regulation`	Refer to *List of Compliance Regulations* for values.	Denotes one or more compliance regulations that a resource must adhere to. Tag administrators may add values to the list from the OCI Governance and Administration console. Refer to *Using Predefined Values* for more details.
`OracleStandard.TargetUsers`	• Public • Customers • Partners • Company • Division • Department • Workgroup	Denotes the end users of a resource. Another form of resource classification that helps determine target users and allows governance teams to set corporate standards based on user or application type.
`OracleStandard.EndUserCount`	• 1 • 10 • 100 • 1000 • 10000 • 100000 • 1000000 • 1000000 • 10000000	An approximate count of end-users. This tag helps determine the number of users impacted or the blast radius during an availability or security event. This also helps prioritize recovery efforts in the event of major outages affecting a large number of cloud resources.
`OracleStandard.OwnerEmail`	Free form tag. For example *john.smith@example.com* or *app_support_grp@example.com*	Denotes the email address of the resource owner.

Table 3-1 (Cont.) Oracle Standard Tags

Tag Key	Tag Value Options	Description
`OracleStandard.Org`	• HR • Finance • Marketing • Sales • Legal • R&D • Customer Suppport • Internal Support • Manufacturing	Identifies the customer's line of business or department that owns or uses the resource. This may help with cost aggregation reports and determining usage across business units.Tag administrators may add relevant values to the list from the OCI Governance and Administration console. Refer to *Using Predefined Values* for more details.
`OracleStandard.CostCenter`	• 12345 • WebMarketing	Freeform field for cost center.
`OracleStandard.RecoveryTimeObjectiveMinutes`	0-10080	Time in minutes. Denotes the maximum time within which the resource is required to recover from a failure.
`OracleStandard.RecoveryPointObjectiveMinutes`	0-1440	Time in minutes. Maximum data loss tolerance for a data store resource such as a database or a storage device.

Related Topics

- To Import standard tags
- Cloud Advisor Overview
- Oracle Autonomous Database Dedicated for Fleet Administrators Workshop
- How Tagging Works
- Authentication and Authorization
- Managing Tag Namespaces
- Using Predefined Values

Restarting a VM for Planned Maintenance

To facilitate maintenance of Oracle Exadata Database Service on Exascale Infrastructure virtual machines (VM), Oracle notifies you of planned VM restarts.

The Oracle Exadata Database Service on Exascale Infrastructure VMs use underlying physical hosts that periodically must undergo maintenance. When such maintenance is required, Oracle schedules a restart of your VM, and notifies you of the upcoming restart. The restart enables your VM to be migrated to a new physical host that is not in need of maintenance. Stopping and starting the node will also result in the migration to a new physical host. The only effect to your VM is the restart itself. The planned maintenance of the original physical hardware takes place after your VM has been migrated to its new host, and has no effect on your VM. If you do not restart your VM during the notification period, then Oracle will restart the VM at the end of the notification period.

Note:

When Oracle schedules a restart of your VM, other VMs in that VM Cluster will not be affected by the planned maintenance. The other nodes in your cluster continue to stay available as part of your high availability (HA) strategy.

Connecting to an Oracle Exadata Database Service on Exascale Infrastructure VM

Learn how to connect to an Oracle Exadata Database Service on Exascale Infrastructure virtual machine (VM) using SSH or SQL Developer.

How you connect depends on how your cloud network is set up. You can find information on various networking scenarios in Networking Overview, but for specific recommendations on how you should connect to a database in the cloud, contact your network security administrator.

Note:

Oracle Exadata Database Service on Exascale Infrastructure servers cannot be joined to Active Directory domains, and the service does not support the use of Active Directory for user authentication and authorization.

- Prerequisites for Accessing Oracle Exadata Database Service on Exascale Infrastructure
  To use SSH to access a compute node in an Oracle Exadata Database Service on Exascale Infrastructure (ExaDB-XS) instance, you need this information.
- SCAN Listener Port Setting
  When creating a cloud VM cluster, you can optionally designate a different SCAN listener port number.
- Connecting to a Virtual Machine with SSH
  You can connect to the virtual machines in an Oracle Exadata Database Service on Exascale Infrastructure system by using a Secure Shell (SSH) connection.
- Using Oracle Net Services to Connect to a Database
  Oracle Database Oracle Exadata Database Service on Exascale Infrastructure supports remote database access by using Oracle Net Services.
- Connect to the Oracle Exadata Database Service on Exascale Infrastructure Service
  Learn how to connect to an Oracle Exadata Database Service on Exascale Infrastructure system using SSH, and how to connect to an Oracle Exadata Database Service on Exascale Infrastructure database using Oracle Net Services (SQL*Net).

Prerequisites for Accessing Oracle Exadata Database Service on Exascale Infrastructure

To use SSH to access a compute node in an Oracle Exadata Database Service on Exascale Infrastructure (ExaDB-XS) instance, you need this information.

Note:

Before you can access ExaDB-XS, you must have configured Exadata Database service on Exascale Infrastructure.

- The full path to the file that contains the private key associated with the public key used when the system was launched.
- The public or private IP address of the Oracle Exadata Database Service on Exascale Infrastructure instance.

  Use the private IP address to connect to the system from your on-premises network, or from within the virtual cloud network (VCN). This includes connecting from a host located on-premises connecting through a VPN or FastConnect to your VCN, or from another host in the same VCN. Use the public IP address to connect to the system from outside the cloud (with no VPN). You can find the IP addresses in the Oracle Cloud Infrastructure Console. On the **Exadata VM Cluster Details** page, click Virtual Machines in the **Resources** list.

  The values are displayed in the **Public IP Address** and **Private IP Address & DNS Name** columns of the table displaying the **Virtual Machines** or **Nodes** of the Oracle Exadata Database Service on Exascale Infrastructure instance.

SCAN Listener Port Setting

When creating a cloud VM cluster, you can optionally designate a different SCAN listener port number.

The default SCAN listener port for cloud VM clusters is 1521. With the console, you have the option to designate a different SCAN listener port number at VM Cluster provisioning. In the OCI Console, this option appears under **Advanced Options** when creating the cluster.

Note:

Manually changing the SCAN listener port of a VM cluster after provisioning using the backend software is not supported. This change can cause Data Guard provisioning to fail.

Connecting to a Virtual Machine with SSH

You can connect to the virtual machines in an Oracle Exadata Database Service on Exascale Infrastructure system by using a Secure Shell (SSH) connection.

Most Unix-style systems (including Linux, Oracle Solaris, and macOS) include an SSH client. For Microsoft Windows systems, you can download a free SSH client called PuTTY from the following site: "http://www.putty.org".

- Connecting from a Unix-Style System
  To access a virtual machine on an Oracle ExaDB-XS system from a Unix-style system using SSH, use this procedure.
- Connecting to a Virtual Machine from a Microsoft Windows System Using PuTTY
  Learn how to access a virtual machine from a Microsoft Windows system using PuTTY.

- Accessing a Database After You Connect to the Virtual Machine
  After you connect to a virtual machine, you can use the following series of commands to identify a database and connect to it.

Related Topics

- http://www.putty.org/

Connecting from a Unix-Style System

To access a virtual machine on an Oracle ExaDB-XS system from a Unix-style system using SSH, use this procedure.

- Enter the following SSH command to access the virtual machine:

  ```
  ssh -i private-key user@node
  ```

  In the preceding syntax:

  - *private-key* is the full path and name of the file that contains the SSH private key that corresponds to a public key that is registered in the system.
  - `user` is the operating system user that you want to use to connect:
    - To perform operations as the Oracle Database software owner, connect as as `opc` and `su oracle`. The `oracle` user does not have `root` user access to the virtual machine.
    - To perform operations that require `root` access to the virtual machine, such as patching, connect as `opc`. The `opc` user can use the `sudo -s` command to gain `root` access to the virtual machine.
  - *node* is the host name or IP address for the virtual machine that you want to access.

Connecting to a Virtual Machine from a Microsoft Windows System Using PuTTY

Learn how to access a virtual machine from a Microsoft Windows system using PuTTY.

Before you use the PuTTY program to connect to a virtual machine, you need the following:

- The IP address of the virtual machine
- The SSH private key file that matches the public key associated with the deployment. This private key file must be in the PuTTY `.ppk` format. If the private key file was originally created on the Linux platform, you can use the PuTTYgen program to convert it to the `.ppk` format.

Before you begin

To connect to a virtual machine using the PuTTY program on Windows:

1. Download and install PuTTY.

   To download PuTTY, go to http://www.putty.org/ and click the **You can download PuTTY here** link.
2. Run the PuTTY program (`putty.exe`).

   The PuTTY Configuration window is displayed, showing the **Session** panel.
3. In the **Host Name (or IP address)** field, enter the host name or IP address of the virtual machine that you want to access.

4. Confirm that the **Connection type** option is set to **SSH**.
5. In the **Category** tree, expand **Connection** if necessary and then click **Data**.

   The **Data** panel is displayed.
6. In the **Auto-login username** field, enter the operating system user that you want to use to connect.
   - To perform operations that require `root`, connect as the user `opc`.
   - To access to the virtual machine for user operations (for example, to run backups), connect as the user oracle. (This user can also use the the sudo command to gain `root` or `oracle` access to the VM.
7. Confirm that the **When username is not specified** option is set to **Prompt**.
8. In the **Category** tree, expand **SSH** and then click **Auth**.

   The **Auth** panel is displayed.
9. Click**Browse** next to the **Private key file for authentication** field. In the **Select private key file** window, navigate to and open the private key file that matches the public key that is associated with the deployment.
10. In the **Category** tree, click **Session**.

    The **Session** panel is displayed.
11. In the **Saved Sessions** field, enter a name for the connection configuration, and click **Save**.
12. Click **Open** to open the connection.

    The PuTTY Configuration window closes and the PuTTY terminal window displays.

    If this is the first time you are connecting to the VM, then the PuTTY Security Alert window is displayed, prompting you to confirm the public key. Click **Yes** to continue connecting.

Accessing a Database After You Connect to the Virtual Machine

After you connect to a virtual machine, you can use the following series of commands to identify a database and connect to it.

1. Access the VM using SSH as the `opc` user.
2. Log in as the Oracle user. For example: `sudo su oracle`
3. Use the `srvctl` utility located under the Oracle Grid Infrastructure home directory to list the databases on the system. For example:

```
/u01/app/12.2.0.1/grid/bin/srvctl config database -v
nc122   /u02/app/oracle/product/12.2.0/dbhome_6 12.2.0.1.0
s12c    /u02/app/oracle/product/12.2.0/dbhome_2 12.2.0.1.0
```

4. Identify the database instances for the database that you want to access. For example:

```
/u01/app/12.2.0.1/grid/bin/srvctl status database -d s12c
Instance s12c1 is running on node node01
Instance s12c2 is running on node node02
```

5. Configure the environment settings for the database that you want to access. For example:

   ```
   . oraenv
   ORACLE_SID = [oracle] ? s12c
   The Oracle base has been set to /u02/app/oracle


   export ORACLE_SID=s12c1
   ```

6. You can use the `svrctl` command to display more detailed information about the database. For example:

   ```
   srvctl config database -d s12c
   Database unique name: s12c
   Database name:
   Oracle home: /u02/app/oracle/product/12.2.0/dbhome_2
   Oracle user: oracle
   Spfile: +DATAC4/s12c/spfiles12c.ora
   Password file: +DATAC4/s12c/PASSWORD/passwd
   Domain: example.com
   Start options: open
   Stop options: immediate
   Database role: PRIMARY
   Management policy: AUTOMATIC
   Server pools:
   Disk Groups: DATAC4
   Mount point paths:
   Services:
   Type: RAC
   Start concurrency:
   Stop concurrency:
   OSDBA group: dba
   OSOPER group: racoper
   Database instances: s12c1,s12c2
   Configured nodes: node01,node02
   CSS critical: no
   CPU count: 0
   Memory target: 0
   Maximum memory: 0
   Default network number for database services:
   Database is administrator managed
   ```

7. You can access the database by using SQL*Plus. For example:

   ```
   sqlplus / as sysdba

   SQL*Plus: Release 12.2.0.1.0 Production ...

   Copyright (c) 1982, 2016, Oracle.  All rights reserved.

   Connected to:
   Oracle Database 12c EE Extreme Perf Release 12.2.0.1.0 - 64bit Production
   ```

Using Oracle Net Services to Connect to a Database

Oracle Database Oracle Exadata Database Service on Exascale Infrastructure supports remote database access by using Oracle Net Services.

Because Oracle Exadata Database Service on Exascale Infrastructure uses Oracle Grid Infrastructure, you can make Oracle Net Services connections by using **Single Client Access Name** (SCAN) connections. SCAN is a feature that provides a consistent mechanism for clients to access the Oracle Database instances running in a cluster.

By default, the SCAN is associated with three virtual IP addresses (VIPs). Each SCAN VIP is also associated with a SCAN listener that provides a connection endpoint for Oracle Database connections using Oracle Net Services. To maximize availability, Oracle Grid Infrastructure distributes the SCAN VIPs and SCAN listeners across the available cluster nodes. In addition, if there is a node shutdown or failure, then the SCAN VIPs and SCAN listeners are automatically migrated to a surviving node. By using SCAN connections, you enhance the ability of Oracle Database clients to have a reliable set of connection endpoints that can service all of the databases running in the cluster.

The SCAN listeners are in addition to the Oracle Net Listeners that run on every node in the cluster, which are also known as the node listeners. When an Oracle Net Services connection comes through a SCAN connection, the SCAN listener routes the connection to one of the node listeners, and plays no further part in the connection. A combination of factors, including listener availability, database instance placement, and workload distribution, determines which node listener receives each connection.

> **Note:**
>
> This documentation provides basic requirements for connecting to your Oracle Exadata Database Service on Exascale Infrastructure databases by using Oracle Net Services.

- Prerequisites for Connecting to a Database with Oracle Net Services
  Review the prerequisites to connect to an Oracle Database instance on Oracle ExaDB-XS using Oracle Net Services.
- Connecting to a Database with SQL Developer
  You can connect to a database with SQL Developer by using one of the following methods:
- Connecting to a Database Using SCAN
  To create an Oracle Net Services connection by using the SCAN listeners, you can choose between two approaches.
- Connecting to a Database Using a Node Listener
  To connect to an Oracle Database instance on Oracle Exadata Database Service on Exascale Infrastructure with a connect descriptor that bypasses the SCAN listeners, use this procedure to route your connection directly to a node listener.

Prerequisites for Connecting to a Database with Oracle Net Services

Review the prerequisites to connect to an Oracle Database instance on Oracle ExaDB-XS using Oracle Net Services.

To connect to an Oracle Database on Oracle Exadata Database Service on Exascale Infrastructure with Oracle Net Services, you need the following:

- The IP addresses for your SCAN VIPs, or the hostname or IP address for a virtual machine that hosts the database that you want to access.
- The database identifier: Either the database system identifier (SID), or a service name.

Connecting to a Database with SQL Developer

You can connect to a database with SQL Developer by using one of the following methods:

- Create a temporary SSH tunnel from your computer to the database. This method provides access only for the duration of the tunnel. (When you are done using the database, be sure to close the SSH tunnel by exiting the SSH session.)
- Open the port used as the Oracle SCAN listener by updating the security list used for the cloud VM cluster or DB system resource in the Exadata Cloud Service instance. The default SCAN listener port is 1521. This method provides more durable access to the database. For more information, see Updating the Security List.

After you've created an SSH tunnel or opened the SCAN listener port as described above, you can connect to an Oracle Exadata Database Service on Exascale Infrastructure instance using SCAN IP addresses or public IP addresses, depending on how your network is set up and where you are connecting from. You can find the IP addresses in the Console, in the **Database** details page.

- To connect using SCAN IP addresses
  You can connect to the database using the SCAN IP addresses if your client is on-premises and you are connecting using a FastConnect or Site-to-Site VPN connection.
- To connect using public IP addresses
  You can use the node's public IP address to connect to the database if the client and database are in different VCNs, or if the database is on a VCN that has an internet gateway.

To connect using SCAN IP addresses

You can connect to the database using the SCAN IP addresses if your client is on-premises and you are connecting using a FastConnect or Site-to-Site VPN connection.

You have the following options:

- Use the private SCAN IP addresses, as shown in the following `tnsnames.ora` example:

```
testdb=
  (DESCRIPTION =
    (ADDRESS_LIST=
      (ADDRESS = (PROTOCOL = TCP)(HOST = <scanIP1>)(PORT = 1521))
      (ADDRESS = (PROTOCOL = TCP)(HOST = <scanIP2>)(PORT = 1521)))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = <dbservice.subnetname.dbvcn.oraclevcn.com>)
    )
  )
```

- Define an external SCAN name in your on-premises DNS server. Your application can resolve this external SCAN name to the DB System's private SCAN IP addresses, and then the application can use a connection string that includes the external SCAN name. In the following tnsnames.ora example, extscanname.example.com is defined in the on-premises DNS server.

```
testdb =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = <extscanname.example.com>)(PORT =
1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = <dbservice.subnetname.dbvcn.oraclevcn.com>)
    )
  )
```

To connect using public IP addresses

You can use the node's public IP address to connect to the database if the client and database are in different VCNs, or if the database is on a VCN that has an internet gateway.

However, there are important implications to consider:

- When the client uses the public IP address, the client bypasses the SCAN listener and reaches the node listener, so server side load balancing is not available.
- When the client uses the public IP address, it cannot take advantage of the VIP failover feature. If a node becomes unavailable, new connection attempts to the node will hang until a TCP/IP timeout occurs. You can set client side sqlnet parameters to limit the TCP/IP timeout.

The following tnsnames.ora example shows a connection string that includes the CONNECT_TIMEOUT parameter to avoid TCP/IP timeouts.

```
test=
  (DESCRIPTION =
    (CONNECT_TIMEOUT=60)
    (ADDRESS_LIST=
      (ADDRESS = (PROTOCOL = TCP)(HOST = <publicIP1>)(PORT = 1521))
      (ADDRESS = (PROTOCOL = TCP)(HOST = <publicIP2>)(PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = <dbservice.subnetname.dbvcn.oraclevcn.com>)
    )
  )
```

Connecting to a Database Using SCAN

To create an Oracle Net Services connection by using the SCAN listeners, you can choose between two approaches.

- Connecting to a Database Using a Connect Descriptor that References All of the SCAN VIPs
  You can set up a connect descriptor for Oracle Exadata Database Service on Exascale Infrastructure System using multiple SCAN listeners.

- Connecting to a Database Use a Connect Descriptor that References a Custom SCAN Name
  You can set up a connect descriptor for Oracle Exadata Database Service on Exascale Infrastructure System using a custom SCAN name.

Connecting to a Database Using a Connect Descriptor that References All of the SCAN VIPs

You can set up a connect descriptor for Oracle Exadata Database Service on Exascale Infrastructure System using multiple SCAN listeners.

This approach requires you to supply all of the single client access name (SCAN) virtual IP (VIP) addresses, and enables Oracle Net Services to connect to an available SCAN listener.

- Use the following template to define a Net Services alias, which is typically used to provide a convenient name for the connect descriptor:

  ```
  alias-name = (DESCRIPTION=
     (ADDRESS_LIST=
       (ADDRESS=(PROTOCOL=tcp)(HOST=SCAN-VIP-1)(PORT=1521))
       (ADDRESS=(PROTOCOL=tcp)(HOST=SCAN-VIP-2)(PORT=1521))
       (ADDRESS=(PROTOCOL=tcp)(HOST=SCAN-VIP-3)(PORT=1521)))
     (CONNECT_DATA=(sid-or-service-entry)))
  ```

  Where:

  `alias-name` is the name you use to identify the alias.

  `SCAN-VIP-[1-3]` are the IP addresses for the SCAN VIPs.

  `sid-or-service-entry` identifies the database SID or service name using one of the following formats:

  - `SID=sid-name`. For example: `SID=S12C1`.
  - `SERVICE_NAME=service-name`. For example: `SERVICE_NAME=PDB1.example.yourcloud.com`.

  Note:

  By default, Oracle Net Services randomly selects one of the addresses in the address list to balance the load between the SCAN listeners.

Connecting to a Database Use a Connect Descriptor that References a Custom SCAN Name

You can set up a connect descriptor for Oracle Exadata Database Service on Exascale Infrastructure System using a custom SCAN name.

Using this approach, you define a custom single client access name (SCAN) name in your domain name server (DNS), which resolves to the three SCAN virtual IP addresses (VIPs).

- Use the following template to define a Net Services alias that references the custom SCAN name:

  ```
  alias-name = (DESCRIPTION=
    (ADDRESS_LIST=(ADDRESS=(PROTOCOL=tcp)(HOST=scan-name)(PORT=1521)))
    (CONNECT_DATA=(sid-or-service-entry)))
  ```

  Where:

  alias-name is the name you use to identify the alias.

  scan-name is the custom SCAN name.

  sid-or-service-entry identifies the database SID or service name using one of the following formats:

  - `SID=`*sid-name*. For example: `SID=S12C1`.
  - `SERVICE_NAME=`*service-name*. For example: `SERVICE_NAME=PDB1.example.yourcloud.com`.

  Alternatively, you can use the easy connect method to specify a connect descriptor with the following format:

  ```
  scan-name:1521/sid-or-service-entry
  ```

  For example:

  ```
  exa1scan.example.com:1521/S12C1
  ```

  Or

  ```
  exa1scan.example.com:1521/PDB1.example.yourcloud.com
  ```

Connecting to a Database Using a Node Listener

To connect to an Oracle Database instance on Oracle Exadata Database Service on Exascale Infrastructure with a connect descriptor that bypasses the SCAN listeners, use this procedure to route your connection directly to a node listener.

By using this method, you give up the high-availability and load-balancing provided by SCAN. However, this method may be desirable if you want to direct connections to a specific node or network interface. For example, you might want to ensure that connections from a program that performs bulk data loading use the backup network.

Using this approach, you direct your connection using the hostname or IP address of the node.

Example 3-1 Defining a Net Service Alias That Directly References the Node

```
alias-name = (DESCRIPTION=
  (CONNECT_TIMEOUT=timeout)
  (ADDRESS_LIST=(ADDRESS=(PROTOCOL=tcp)(HOST=node)(PORT=1521)))
  (CONNECT_DATA=(sid-or-service-entry)))
```

Where:

alias-name is the name you use to identify the alias.

timeout specifies a timeout period (in seconds), which enables you to terminate a connection attempt without having to wait for a TCP timeout. The (CONNECT_TIMEOUT=*timeout*) parameter is optional.

node is the hostname or IP address for the virtual machine that you want to use.

sid-or-service-entry identifies the database SID or service name using one of the following formats:

- SID=*sid-name*. For example, SID=S12C1.
- SERVICE_NAME=*service-name*. For example, SERVICE_NAME=PDB1.example.oraclecloudatcust.com.

Alternatively, you can use the easy connect method to specify a connect descriptor with the following format:

```
node:1521/sid-or-service-entry
```

For example:

```
exa1node01.example.com:1521/S12C1
```

Or

```
exa1node01.example.com:1521/PDB1.example.oraclecloudatcust.com
```

Connect to the Oracle Exadata Database Service on Exascale Infrastructure Service

Learn how to connect to an Oracle Exadata Database Service on Exascale Infrastructure system using SSH, and how to connect to an Oracle Exadata Database Service on Exascale Infrastructure database using Oracle Net Services (SQL*Net).

- Connecting to a Database with SQL Developer
  You can connect to a database with SQL Developer by using one of the following methods:
- Connecting to a Database with Oracle Net Services
  You can connect to the virtual machines in an Oracle Exadata Database Service on Exascale Infrastructure system using Oracle Net Services.

Connecting to a Database with SQL Developer

You can connect to a database with SQL Developer by using one of the following methods:

- Create a temporary SSH tunnel from your computer to the database. This method provides access only for the duration of the tunnel. (When you are done using the database, be sure to close the SSH tunnel by exiting the SSH session.)
- Open the port used as the Oracle SCAN listener by updating the security list used for the cloud VM cluster or DB system resource in the Exadata Cloud Service instance. The default SCAN listener port is 1521. This method provides more durable access to the database. For more information, see Updating the Security List.

After you've created an SSH tunnel or opened the SCAN listener port as described above, you can connect to an Oracle Exadata Database Service on Exascale Infrastructure instance using SCAN IP addresses or public IP addresses, depending on how your network is set up and where you are connecting from. You can find the IP addresses in the Console, in the **Database** details page.

Connecting to a Database with Oracle Net Services

You can connect to the virtual machines in an Oracle Exadata Database Service on Exascale Infrastructure system using Oracle Net Services.

- Using Oracle Net Services to Connect to a Database
  Oracle Database Oracle Exadata Database Service on Exascale Infrastructure supports remote database access by using Oracle Net Services.
- Prerequisites for Connecting to a Database with Oracle Net Services
  Review the prerequisites to connect to an Oracle Database instance on Oracle ExaDB-XS using Oracle Net Services.
- Connecting to a Database Using SCAN
  To create an Oracle Net Services connection by using the SCAN listeners, you can choose between two approaches.
- Connecting to a Database Using a Node Listener
  To connect to an Oracle Database instance on Oracle Exadata Database Service on Exascale Infrastructure with a connect descriptor that bypasses the SCAN listeners, use this procedure to route your connection directly to a node listener.

Using Oracle Net Services to Connect to a Database

Oracle Database Oracle Exadata Database Service on Exascale Infrastructure supports remote database access by using Oracle Net Services.

Because Oracle Exadata Database Service on Exascale Infrastructure uses Oracle Grid Infrastructure, you can make Oracle Net Services connections by using **Single Client Access Name** (SCAN) connections. SCAN is a feature that provides a consistent mechanism for clients to access the Oracle Database instances running in a cluster.

By default, the SCAN is associated with three virtual IP addresses (VIPs). Each SCAN VIP is also associated with a SCAN listener that provides a connection endpoint for Oracle Database connections using Oracle Net Services. To maximize availability, Oracle Grid Infrastructure distributes the SCAN VIPs and SCAN listeners across the available cluster nodes. In addition, if there is a node shutdown or failure, then the SCAN VIPs and SCAN listeners are automatically migrated to a surviving node. By using SCAN connections, you enhance the ability of Oracle Database clients to have a reliable set of connection endpoints that can service all of the databases running in the cluster.

The SCAN listeners are in addition to the Oracle Net Listeners that run on every node in the cluster, which are also known as the node listeners. When an Oracle Net Services connection comes through a SCAN connection, the SCAN listener routes the connection to one of the node listeners, and plays no further part in the connection. A combination of factors, including listener availability, database instance placement, and workload distribution, determines which node listener receives each connection.

> **Note:**
>
> This documentation provides basic requirements for connecting to your Oracle Exadata Database Service on Exascale Infrastructure databases by using Oracle Net Services.

Prerequisites for Connecting to a Database with Oracle Net Services

Review the prerequisites to connect to an Oracle Database instance on Oracle ExaDB-XS using Oracle Net Services.

To connect to an Oracle Database on Oracle Exadata Database Service on Exascale Infrastructure with Oracle Net Services, you need the following:

- The IP addresses for your SCAN VIPs, or the hostname or IP address for a virtual machine that hosts the database that you want to access.
- The database identifier: Either the database system identifier (SID), or a service name.

Connecting to a Database Using SCAN

To create an Oracle Net Services connection by using the SCAN listeners, you can choose between two approaches.

- Identifying IP Addresses Using the SDK or CLI
  You can use the SDK or the OCI CLI to identify the IP addresses of Oracle Exadata Database Service on Exascale Infrastructure compute nodes. You can then use the IP addresses to connect to your system.
- Connecting to a Database Using a Connect Descriptor that References All of the SCAN VIPs
  You can set up a connect descriptor for Oracle Exadata Database Service on Exascale Infrastructure System using multiple SCAN listeners.
- Connecting to a Database Use a Connect Descriptor that References a Custom SCAN Name
  You can set up a connect descriptor for Oracle Exadata Database Service on Exascale Infrastructure System using a custom SCAN name.

Identifying IP Addresses Using the SDK or CLI

You can use the SDK or the OCI CLI to identify the IP addresses of Oracle Exadata Database Service on Exascale Infrastructure compute nodes. You can then use the IP addresses to connect to your system.

NOT_SUPPORTED

1. Use the GetDbNode API to return the details of the Oracle Exadata Database Service on Exascale InfrastructuredbNode. Note the OCIDs returned for the `hostIpId` and `backupIpId` parameters of the dbNode.
2. With the OCIDs found in the `hostIpId` and `backupIpId` parameters, you can use the GetPrivateIp API to get the private IP addresses used by the client and backup subnets. For public subnet IP addresses, use the GetPublicIpByPrivateIpId API.

Connecting to a Database Using a Connect Descriptor that References All of the SCAN VIPs

You can set up a connect descriptor for Oracle Exadata Database Service on Exascale Infrastructure System using multiple SCAN listeners.

This approach requires you to supply all of the single client access name (SCAN) virtual IP (VIP) addresses, and enables Oracle Net Services to connect to an available SCAN listener.

- Use the following template to define a Net Services alias, which is typically used to provide a convenient name for the connect descriptor:

  ```
  alias-name = (DESCRIPTION=
    (ADDRESS_LIST=
      (ADDRESS=(PROTOCOL=tcp)(HOST=SCAN-VIP-1)(PORT=1521))
      (ADDRESS=(PROTOCOL=tcp)(HOST=SCAN-VIP-2)(PORT=1521))
      (ADDRESS=(PROTOCOL=tcp)(HOST=SCAN-VIP-3)(PORT=1521)))
    (CONNECT_DATA=(sid-or-service-entry)))
  ```

  Where:

  `alias-name` is the name you use to identify the alias.

  `SCAN-VIP-[1-3]` are the IP addresses for the SCAN VIPs.

  `sid-or-service-entry` identifies the database SID or service name using one of the following formats:

  - `SID=`*`sid-name`*. For example: `SID=S12C1`.
  - `SERVICE_NAME=`*`service-name`*. For example: `SERVICE_NAME=PDB1.example.yourcloud.com`.

  > **Note:**
  >
  > By default, Oracle Net Services randomly selects one of the addresses in the address list to balance the load between the SCAN listeners.

Connecting to a Database Use a Connect Descriptor that References a Custom SCAN Name

You can set up a connect descriptor for Oracle Exadata Database Service on Exascale Infrastructure System using a custom SCAN name.

Using this approach, you define a custom single client access name (SCAN) name in your domain name server (DNS), which resolves to the three SCAN virtual IP addresses (VIPs).

- Use the following template to define a Net Services alias that references the custom SCAN name:

  ```
  alias-name = (DESCRIPTION=
    (ADDRESS_LIST=(ADDRESS=(PROTOCOL=tcp)(HOST=scan-name)(PORT=1521)))
    (CONNECT_DATA=(sid-or-service-entry)))
  ```

  Where:

  `alias-name` is the name you use to identify the alias.

scan-name is the custom SCAN name.

sid-or-service-entry identifies the database SID or service name using one of the following formats:

- SID=*sid-name*. For example: SID=S12C1.
- SERVICE_NAME=*service-name*. For example: SERVICE_NAME=PDB1.example.yourcloud.com.

Alternatively, you can use the easy connect method to specify a connect descriptor with the following format:

```
scan-name:1521/sid-or-service-entry
```

For example:

```
exa1scan.example.com:1521/S12C1
```

Or

```
exa1scan.example.com:1521/PDB1.example.yourcloud.com
```

Connecting to a Database Using a Node Listener

To connect to an Oracle Database instance on Oracle Exadata Database Service on Exascale Infrastructure with a connect descriptor that bypasses the SCAN listeners, use this procedure to route your connection directly to a node listener.

By using this method, you give up the high-availability and load-balancing provided by SCAN. However, this method may be desirable if you want to direct connections to a specific node or network interface. For example, you might want to ensure that connections from a program that performs bulk data loading use the backup network.

Using this approach, you direct your connection using the hostname or IP address of the node.

Example 3-2 Defining a Net Service Alias That Directly References the Node

```
alias-name = (DESCRIPTION=
  (CONNECT_TIMEOUT=timeout)
  (ADDRESS_LIST=(ADDRESS=(PROTOCOL=tcp)(HOST=node)(PORT=1521)))
  (CONNECT_DATA=(sid-or-service-entry)))
```

Where:

alias-name is the name you use to identify the alias.

timeout specifies a timeout period (in seconds), which enables you to terminate a connection attempt without having to wait for a TCP timeout. The (CONNECT_TIMEOUT=*timeout*) parameter is optional.

node is the hostname or IP address for the virtual machine that you want to use.

sid-or-service-entry identifies the database SID or service name using one of the following formats:

- SID=*sid-name*. For example, SID=S12C1.

- `SERVICE_NAME=`*`service-name`*. For example, `SERVICE_NAME=PDB1.example.oraclecloudatcust.com`.

Alternatively, you can use the easy connect method to specify a connect descriptor with the following format:

```
node:1521/sid-or-service-entry
```

For example:

```
exa1node01.example.com:1521/S12C1
```

Or

```
exa1node01.example.com:1521/PDB1.example.oraclecloudatcust.com
```

Capacity Limits for Exadata Database Service on Exascale Infrastructure

To understand the resource capacity of the ExaDB-XS service, review these tables

Database Storage Vault Minimum Capacity

The total minimum capacity billed for ExaDB-XS vaults is 300 GB. Images are stored in an Oracle Advanced Cluster File System (ACFS), and the remainder of space is available for a first database, as described in the following table.

Table 3-2 ExaDB-XS Minimum Database Storage Vault Capacity for Systems and Database Use

Purpose	Minimum Capacity
System use (images stored in ACFS)	50 GB
Database use (provisioning a first database)	250 GB

VM File System Storage Minimum Capacity

The total minimum capacity billed for virtual machine (VM) storage is 280 GB. File system minimum capacities are listed in the following table.

Table 3-3 ExaDB-XS VM File System Storage Minimum Billed Capacity

File System	Minimum Total Capacity (GB)	Minimum Usable Capacity (GB)
`/boot`	0.512	0.412
`/` (mirrored)	30	15
`/tmp`	10	10
`/var` (mirrored)	10	5
`/var/log`	18	18
`/var/log/audit`	3	3

Table 3-3 (Cont.) ExaDB-XS VM File System Storage Minimum Billed Capacity

File System	Minimum Total Capacity (GB)	Minimum Usable Capacity (GB)
/home	4	4
Swap space (/swap)	16	16
/crashfiles	20	20
/u01	82	80
/u02	84	81
Overhead	2	Not applicable
All file systems (total minimum)	280	Not applicable

Best Practices for Oracle Exadata Database Service on Exascale Infrastructure VMs

Oracle recommends that you follow these best practice guidelines to ensure the manageability of your Oracle Exadata Database Service on Exascale Infrastructure virtual machines (VMs).

When followed, best practice guidelines can prevent problems that can affect the manageability and performance of yourOracle Exadata Database Service on Exascale Infrastructure VMs:

- Wherever possible, use the Oracle-supplied cloud interfaces such as the Oracle Cloud Infrastructure Console, API, or CLI, or cloud-specific tools such as `dbaascli` to perform lifecycle management and administrative operations on your Oracle Exadata Database Service on Exascale Infrastructure VM. For example, use the OCI console, API, CLI, or dbaascli to apply Oracle Database patches instead of manually running `opatch`. In addition, if an operation can be performed by using the Console as well as a command-line utility, Oracle recommends that you use the Console. For example, use the Console instead of using `dbaascli` to create databases.
- Do not change the Guest OS users or manually manipulate SSH key settings associated with your VM.
- Apply *only* patches that are available through the Database service. Do *not* apply patches from any other source unless you are directed to do so by Oracle Support.
- Apply the quarterly patches regularly, every quarter if possible.
- Do not change the ports for Oracle Net Listener.

4 How-to Guides

A collection of tasks and procedures for managing Exadata Database Service on Dedicated Infrastructure.

- Configure Oracle Database Features for Oracle Exadata Database Service on Exascale Infrastructure
  Learn how to configure Oracle Multitenant, tablespace encryption, and other options for your Oracle Exadata Database Service on Exascale Infrastructure instance.
- Migrate to Oracle Exadata Database Service on Exascale Infrastructure
  For general guidance on methods and tools to migrate databases to Oracle Cloud Infrastructure database services, including Oracle Exadata Database Service on Exascale Infrastructure see "Migrating Databases to the Cloud".
- Connect Identity and Access Management (IAM) Users to Oracle Exadata Database Service on Exascale Infrastructure
  You can configure Exadata Database Service on Exascale Infrastructure to use Oracle Cloud Infrastructure Identity and Access Management (IAM) authentication and authorization to allow IAM users to access an Oracle Database with IAM credentials.

Manage Database Security with Oracle Data Safe

Learn how to use Oracle Data Safe with Oracle Exadata Database Service on Exascale Infrastructure

- About Oracle Data Safe
- Get Started
- Using Oracle Data Safe

About Oracle Data Safe

Your corporate policy requires that you monitor your databases and retain audit records. Your developers are asking for copies of production data for that new application, and you're wondering what kinds of sensitive information it will contain. Meanwhile, you need to make sure that recent maintenance activities haven't left critical security configuration gaps on your production databases and that staff changes haven't left dormant user accounts on the databases. Oracle Data Safe assists you with these tasks and is included with your Exadata Database Service*.

Oracle Data Safe is a unified control center, that helps you to manage the day-to-day security and compliance requirements of Oracle Databases no matter if they are running in the Oracle Cloud Infrastructure, at Cloud@Customer, on-premises or in any other cloud.

Data Safe supports you to evaluate security controls, assess user security, monitor user activity, and address data security compliance requirements for your database by evaluating the sensitivity of your data as well as masking sensitive data for non-production databases.

Data Safe provides the following features:

- **Security Assessment**: Configuration errors and configuration drift are significant contributors to data breaches. Use security assessment to evaluate your database's configuration and compare it to Oracle and industry best practices. Security assessment reports on areas of risk and notifies you when configurations change.
- **User Assessment**: Many breaches start with a compromised user account. User Assessment helps you spot the riskiest database accounts - those accounts which, if compromised, could cause the most damage - and take proactive steps to secure them. User Assessment Baselines make it easy to know when new accounts are added, or an account's privileges are modified. Use OCI events to receive proactive notifications when a database deviates from its baseline.

- **Activity Auditing**: Understanding and reporting on user activity, data access, and changes to database structures supports regulatory compliance requirements and can aid in post-incident investigations. Activity auditing collects audit records from databases and helps you manage audit policies. Audit insights make it easy to identify inefficient audit policies, while alerts based on audit data proactively notify you of risky activity.
- **Sensitive Data Discovery**: Knowing what sensitive data is managed in your applications is critical for security and privacy. Data discovery scans your database for over 150 different types of sensitive data, helping you understand what types and how much sensitive data you are storing. Use these reports to formulate audit policies, develop data masking templates, and create effective access control policies.
- **Data Masking**: Minimizing the amount of sensitive data your organization maintains helps you meet compliance requirements and satisfy data privacy regulations. Data masking helps you remove risk from your non-production databases by replacing sensitive information with masked data. With reusable masking templates, over 50 included masking formats, and the ability to easily create custom formats for your organization's unique requirements, data masking can streamline your application development and testing operations.

**Includes 1 million audit records per database per month if using the audit collection for Activity Auditing*

Get Started

To get started you just need to register your database with Oracle Data Safe:

- Pre-requisite: Obtain the necessary Identity and Access Management (IAM) permissions to register your target database in Data Safe: Permissions to Register an Oracle Cloud Database with Oracle Data Safe
- Connecting your database to Data Safe
  - If your database is running in a private virtual cloud network (VCN), you can connect it to Data Safe via a **Data Safe private endpoint**.

    The private endpoint essentially represents the Oracle Data Safe service in your VCN with a private IP address in a subnet of your choice.

    You can create the private endpoint in the VCN of your database either before the registration or during the registration process. You can find more details on how to create the private endpoint under Create an Oracle Data Safe Private Endpoint.
- Register your database in Data Safe

Using Oracle Data Safe

Once your database is registered in Data Safe, you can leverage all features.

Security Assessment

Security Assessments are automatically scheduled once a week in Data Safe and provide an overall picture of your database security posture. It analyzes your database configurations, users and user entitlements, as well as security policies to uncover security risks and improve the security posture of Oracle Databases within your organization. A security assessment provides findings with recommendations for remediation activities that follow best practices to reduce or mitigate risk.

Start by reviewing the security assessment report for your database: View the latest assessment for a target database

You can find more details on Security Assessment under Security Assessment Overview.

User Assessment

User Assessments are automatically scheduled once a week in Data Safe and help you to identify highly privileged user accounts that could pose a threat if misused or compromised. User Assessment reviews information about your users in the data dictionaries on your target databases and then calculates a potential risk for each user, based on system privileges and role grants.

Start by reviewing the user assessment report for your database: View the latest user assessment for a target database

You can find more details on User Assessment under User Assessment Overview.

Data Discovery

Data Discovery searches for sensitive columns in your database. It comes with over 150 pre-defined sensitive types and you can also create your own sensitive types. You tell Data Discovery if you want to scan your entire database or just certain schemas and what type of sensitive information to look for, and it finds the sensitive columns that meet your criteria and stores them in a sensitive data model (SDM).

Start by discovering sensitive data in your database: Create Sensitive Data Models

You can find more details on Data Discovery under Data Discovery Overview.

Data Masking

Data masking, also known as static data masking helps you to replace sensitive or confidential information in your non-production databases with realistic and fully functional data with similar characteristics as the original data. Data Safe comes with pre-defined masking formats for each of the pre-defined sensitive types that can also be leveraged for your own sensitive types.

Once you know where sensitive data is stored in your database (for instance after running Data Discovery in Data Safe), you can start by creating a masking policy: Create Masking Policies

After you created a masking policy and copied your production database, you can mask your non-production copy: Mask Sensitive Data on a Target Database

You can find more details on Data Masking under Data Masking Overview.

Activity Auditing

Activity Auditing in Oracle Data Safe helps to ensure accountability and improve regulatory compliance. With Activity Auditing, you can collect and retain audit records per industry and regulatory compliance requirements and monitor user activities on Oracle databases with pre-defined reports and alerts. For example, you can audit access to sensitive data, security-relevant events, administrator and user activities, activities recommended by compliance regulations like the Center for Internet Security (CIS), and activities defined by your own organization.

If you are using the audit collection in Data Safe, up to 1 million audit records per target database per month are included for your Cloud@Customer database.

To use activity auditing, start the audit trail for your target database in Data Safe: Start an Audit Trail

Once the audit trail is started, you can monitor and analyze your audit data with pre-defined audit reports: View a Predefined or Custom Audit Report

You can find more details on Activity Auditing under Activity Auditing Overview.

Connecting to an Oracle Exadata Database Service on Exascale Infrastructure VM

Learn how to connect to an Oracle Exadata Database Service on Exascale Infrastructure virtual machine (VM) using SSH or SQL Developer.

How you connect depends on how your cloud network is set up. You can find information on various networking scenarios in Networking Overview, but for specific recommendations on how you should connect to a database in the cloud, contact your network security administrator.

> **Note:**
>
> Oracle Exadata Database Service on Exascale Infrastructure servers cannot be joined to Active Directory domains, and the service does not support the use of Active Directory for user authentication and authorization.

- Connection Prerequisites
  Review the requirements for SSH access to a virtual machine (VM) in Oracle Exadata Database Service on Exascale Infrastructure.
- About Connecting to a VM with SSH
  You can connect to the virtual machines (VMs) in an Oracle Exadata Database Service on Exascale Infrastructure system by using a Secure Shell (SSH) connection.
- Connect to the Oracle Exadata Database Service on Exascale Infrastructure Service
  Learn how to connect to an Oracle Exadata Database Service on Exascale Infrastructure system using SSH, and how to connect to an Oracle Exadata Database Service on Exascale Infrastructure database using Oracle Net Services (SQL*Net).

Connection Prerequisites

Review the requirements for SSH access to a virtual machine (VM) in Oracle Exadata Database Service on Exascale Infrastructure.

You'll need the following:

- The full path to the file that contains the private key associated with the public key used when the system was launched.
- The public or private IP address of the Oracle Exadata Database Service on Exascale Infrastructure VM.

  Use the private IP address to connect to the system from your on-premises network, or from within the virtual cloud network (VCN). This includes connecting from a host located on-premises connecting through a VPN or FastConnect to your VCN, or from another host in the same VCN. Use the public IP address to connect to the system from outside the cloud (with no VPN). You can find the IP addresses in the Oracle Cloud InfrastructureConsole as follows:

  - *Cloud VM clusters:* On the **Exadata VM Cluster Details** page, click **Virtual Machines** in the **Resources** list.
  - *DB systems:* On the **DB System Details** page, click **Nodes** in the **Resources** list.

  The values are displayed in the **Public IP Address** and **Private IP Address & DNS Name** columns of the table displaying the **Virtual Machines** or **Nodes** of the Oracle Exadata Database Service on Exascale Infrastructure VM.

About Connecting to a VM with SSH

You can connect to the virtual machines (VMs) in an Oracle Exadata Database Service on Exascale Infrastructure system by using a Secure Shell (SSH) connection.

Most Unix-style systems (including Linux, Oracle Solaris, and Apple MacOS) include an SSH client. For Microsoft Windows, you can download a free SSH client called PuTTY from the following address: http://www.putty.org

- Connecting from a Unix-Style System
  To access a virtual machine on an Oracle ExaDB-XS system from a Unix-style system using SSH, use this procedure.
- Connecting to a Virtual Machine from a Microsoft Windows System Using PuTTY
  Learn how to access a virtual machine from a Microsoft Windows system using PuTTY.
- To access a database after you connect to the VM
  To connect to the database, you set environment information for the database.

Connecting from a Unix-Style System

To access a virtual machine on an Oracle ExaDB-XS system from a Unix-style system using SSH, use this procedure.

- Enter the following SSH command to access the virtual machine:

  ```
  ssh -i private-key user@node
  ```

  In the preceding syntax:

  - `private-key` is the full path and name of the file that contains the SSH private key that corresponds to a public key that is registered in the system.
  - `user` is the operating system user that you want to use to connect:
    - To perform operations as the Oracle Database software owner, connect as as `opc` and `su oracle`. The `oracle` user does not have `root` user access to the virtual machine.
    - To perform operations that require `root` access to the virtual machine, such as patching, connect as `opc`. The `opc` user can use the `sudo -s` command to gain `root` access to the virtual machine.
  - `node` is the host name or IP address for the virtual machine that you want to access.

Connecting to a Virtual Machine from a Microsoft Windows System Using PuTTY

Learn how to access a virtual machine from a Microsoft Windows system using PuTTY.

Before you use the PuTTY program to connect to a virtual machine, you need the following:

- The IP address of the virtual machine
- The SSH private key file that matches the public key associated with the deployment. This private key file must be in the PuTTY `.ppk` format. If the private key file was originally created on the Linux platform, you can use the PuTTYgen program to convert it to the `.ppk` format.

Before you begin

To connect to a virtual machine using the PuTTY program on Windows:

1. Download and install PuTTY.

   To download PuTTY, go to http://www.putty.org/ and click the **You can download PuTTY here** link.
2. Run the PuTTY program (`putty.exe`).

   The PuTTY Configuration window is displayed, showing the **Session** panel.
3. In the **Host Name (or IP address)** field, enter the host name or IP address of the virtual machine that you want to access.
4. Confirm that the **Connection type** option is set to **SSH**.
5. In the **Category** tree, expand **Connection** if necessary and then click **Data**.

   The **Data** panel is displayed.
6. In the **Auto-login username** field, enter the operating system user that you want to use to connect.
   - To perform operations that require `root`, connect as the user `opc`.
   - To access to the virtual machine for user operations (for example, to run backups), connect as the user oracle. (This user can also use the the sudo command to gain `root` or `oracle` access to the VM.
7. Confirm that the **When username is not specified** option is set to **Prompt**.
8. In the **Category** tree, expand **SSH** and then click **Auth**.

   The **Auth** panel is displayed.
9. Click**Browse** next to the **Private key file for authentication** field. In the **Select private key file** window, navigate to and open the private key file that matches the public key that is associated with the deployment.
10. In the **Category** tree, click **Session**.

    The **Session** panel is displayed.
11. In the **Saved Sessions** field, enter a name for the connection configuration, and click **Save**.
12. Click **Open** to open the connection.

    The PuTTY Configuration window closes and the PuTTY terminal window displays.

    If this is the first time you are connecting to the VM, then the PuTTY Security Alert window is displayed, prompting you to confirm the public key. Click **Yes** to continue connecting.

To access a database after you connect to the VM

To connect to the database, you set environment information for the database.

1. Log in as `opc` and then use `sudo` to connect as the `oracle` user.

   ```
   login as: opc

   [opc@host_name ~]$ sudo su - oracle
   ```

2. Source the database's `.env` file to set the environment.

   ```
   [oracle@host_name]# . database_name.env
   ```

   In the following example, the host name is `ed1db01` and the database name is `cdb01`.

   ```
   [oracle@ed1db01]# . cdb01.env
   ORACLE_SID = [root]
   The Oracle base has been set to /u01/app/grid
   ```

Connect to the Oracle Exadata Database Service on Exascale Infrastructure Service

Learn how to connect to an Oracle Exadata Database Service on Exascale Infrastructure system using SSH, and how to connect to an Oracle Exadata Database Service on Exascale Infrastructure database using Oracle Net Services (SQL*Net).

- Connecting to a Database with SQL Developer
  You can connect to a database with SQL Developer by using one of the following methods:
- Connecting to a Database with Oracle Net Services
  You can connect to the virtual machines in an Oracle Exadata Database Service on Exascale Infrastructure system using Oracle Net Services.

Connecting to a Database with SQL Developer

You can connect to a database with SQL Developer by using one of the following methods:

- Create a temporary SSH tunnel from your computer to the database. This method provides access only for the duration of the tunnel. (When you are done using the database, be sure to close the SSH tunnel by exiting the SSH session.)
- Open the port used as the Oracle SCAN listener by updating the security list used for the cloud VM cluster or DB system resource in the Exadata Cloud Service instance. The default SCAN listener port is 1521. This method provides more durable access to the database. For more information, see Updating the Security List.

After you've created an SSH tunnel or opened the SCAN listener port as described above, you can connect to an Oracle Exadata Database Service on Exascale Infrastructure instance using SCAN IP addresses or public IP addresses, depending on how your network is set up and where you are connecting from. You can find the IP addresses in the Console, in the **Database** details page.

Connecting to a Database with Oracle Net Services

You can connect to the virtual machines in an Oracle Exadata Database Service on Exascale Infrastructure system using Oracle Net Services.

- Using Oracle Net Services to Connect to a Database
  Oracle Database Oracle Exadata Database Service on Exascale Infrastructure supports remote database access by using Oracle Net Services.
- Prerequisites for Connecting to a Database with Oracle Net Services
  Review the prerequisites to connect to an Oracle Database instance on Oracle ExaDB-XS using Oracle Net Services.
- Connecting to a Database Using SCAN
  To create an Oracle Net Services connection by using the SCAN listeners, you can choose between two approaches.
- Connecting to a Database Using a Node Listener
  To connect to an Oracle Database instance on Oracle Exadata Database Service on Exascale Infrastructure with a connect descriptor that bypasses the SCAN listeners, use this procedure to route your connection directly to a node listener.

Using Oracle Net Services to Connect to a Database

Oracle Database Oracle Exadata Database Service on Exascale Infrastructure supports remote database access by using Oracle Net Services.

Because Oracle Exadata Database Service on Exascale Infrastructure uses Oracle Grid Infrastructure, you can make Oracle Net Services connections by using **Single Client Access Name** (SCAN) connections. SCAN is a feature that provides a consistent mechanism for clients to access the Oracle Database instances running in a cluster.

By default, the SCAN is associated with three virtual IP addresses (VIPs). Each SCAN VIP is also associated with a SCAN listener that provides a connection endpoint for Oracle Database connections using Oracle Net Services. To maximize availability, Oracle Grid Infrastructure distributes the SCAN VIPs and SCAN listeners across the available cluster nodes. In addition, if there is a node shutdown or failure, then the SCAN VIPs and SCAN listeners are automatically migrated to a surviving node. By using SCAN connections, you enhance the ability of Oracle Database clients to have a reliable set of connection endpoints that can service all of the databases running in the cluster.

The SCAN listeners are in addition to the Oracle Net Listeners that run on every node in the cluster, which are also known as the node listeners. When an Oracle Net Services connection comes through a SCAN connection, the SCAN listener routes the connection to one of the node listeners, and plays no further part in the connection. A combination of factors, including listener availability, database instance placement, and workload distribution, determines which node listener receives each connection.

> **Note:**
>
> This documentation provides basic requirements for connecting to your Oracle Exadata Database Service on Exascale Infrastructure databases by using Oracle Net Services.

Prerequisites for Connecting to a Database with Oracle Net Services

Review the prerequisites to connect to an Oracle Database instance on Oracle ExaDB-XS using Oracle Net Services.

To connect to an Oracle Database on Oracle Exadata Database Service on Exascale Infrastructure with Oracle Net Services, you need the following:

- The IP addresses for your SCAN VIPs, or the hostname or IP address for a virtual machine that hosts the database that you want to access.
- The database identifier: Either the database system identifier (SID), or a service name.

Connecting to a Database Using SCAN

To create an Oracle Net Services connection by using the SCAN listeners, you can choose between two approaches.

- Identifying IP Addresses Using the SDK or CLI
  You can use the SDK or the OCI CLI to identify the IP addresses of Oracle Exadata Database Service on Exascale Infrastructure compute nodes. You can then use the IP addresses to connect to your system.
- Connecting to a Database Using a Connect Descriptor that References All of the SCAN VIPs
  You can set up a connect descriptor for Oracle Exadata Database Service on Exascale Infrastructure System using multiple SCAN listeners.
- Connecting to a Database Use a Connect Descriptor that References a Custom SCAN Name
  You can set up a connect descriptor for Oracle Exadata Database Service on Exascale Infrastructure System using a custom SCAN name.

Identifying IP Addresses Using the SDK or CLI

You can use the SDK or the OCI CLI to identify the IP addresses of Oracle Exadata Database Service on Exascale Infrastructure compute nodes. You can then use the IP addresses to connect to your system.

NOT_SUPPORTED

1. Use the GetDbNode API to return the details of the Oracle Exadata Database Service on Exascale InfrastructuredbNode. Note the OCIDs returned for the `hostIpId` and `backupIpId` parameters of the dbNode.
2. With the OCIDs found in the `hostIpId` and `backupIpId` parameters, you can use the GetPrivateIp API to get the private IP addresses used by the client and backup subnets. For public subnet IP addresses, use the GetPublicIpByPrivateIpId API.

Connecting to a Database Using a Connect Descriptor that References All of the SCAN VIPs

You can set up a connect descriptor for Oracle Exadata Database Service on Exascale Infrastructure System using multiple SCAN listeners.

This approach requires you to supply all of the single client access name (SCAN) virtual IP (VIP) addresses, and enables Oracle Net Services to connect to an available SCAN listener.

- Use the following template to define a Net Services alias, which is typically used to provide a convenient name for the connect descriptor:

```
alias-name = (DESCRIPTION=
  (ADDRESS_LIST=
    (ADDRESS=(PROTOCOL=tcp)(HOST=SCAN-VIP-1)(PORT=1521))
    (ADDRESS=(PROTOCOL=tcp)(HOST=SCAN-VIP-2)(PORT=1521))
    (ADDRESS=(PROTOCOL=tcp)(HOST=SCAN-VIP-3)(PORT=1521)))
  (CONNECT_DATA=(sid-or-service-entry)))
```

Where:

`alias-name` is the name you use to identify the alias.

`SCAN-VIP-[1-3]` are the IP addresses for the SCAN VIPs.

`sid-or-service-entry` identifies the database SID or service name using one of the following formats:

- `SID=`*`sid-name`*. For example: `SID=S12C1`.
- `SERVICE_NAME=`*`service-name`*. For example: `SERVICE_NAME=PDB1.example.yourcloud.com`.

> **Note:**
>
> By default, Oracle Net Services randomly selects one of the addresses in the address list to balance the load between the SCAN listeners.

Connecting to a Database Use a Connect Descriptor that References a Custom SCAN Name

You can set up a connect descriptor for Oracle Exadata Database Service on Exascale Infrastructure System using a custom SCAN name.

Using this approach, you define a custom single client access name (SCAN) name in your domain name server (DNS), which resolves to the three SCAN virtual IP addresses (VIPs).

- Use the following template to define a Net Services alias that references the custom SCAN name:

```
alias-name = (DESCRIPTION=
   (ADDRESS_LIST=(ADDRESS=(PROTOCOL=tcp)(HOST=scan-name)(PORT=1521)))
   (CONNECT_DATA=(sid-or-service-entry)))
```

Where:

`alias-name` is the name you use to identify the alias.

`scan-name` is the custom SCAN name.

`sid-or-service-entry` identifies the database SID or service name using one of the following formats:

- `SID=`*`sid-name`*. For example: `SID=S12C1`.
- `SERVICE_NAME=`*`service-name`*. For example: `SERVICE_NAME=PDB1.example.yourcloud.com`.

Alternatively, you can use the easy connect method to specify a connect descriptor with the following format:

```
scan-name:1521/sid-or-service-entry
```

For example:

```
exa1scan.example.com:1521/S12C1
```

Or

```
exa1scan.example.com:1521/PDB1.example.yourcloud.com
```

Connecting to a Database Using a Node Listener

To connect to an Oracle Database instance on Oracle Exadata Database Service on Exascale Infrastructure with a connect descriptor that bypasses the SCAN listeners, use this procedure to route your connection directly to a node listener.

By using this method, you give up the high-availability and load-balancing provided by SCAN. However, this method may be desirable if you want to direct connections to a specific node or network interface. For example, you might want to ensure that connections from a program that performs bulk data loading use the backup network.

Using this approach, you direct your connection using the hostname or IP address of the node.

Example 4-1 Defining a Net Service Alias That Directly References the Node

```
alias-name = (DESCRIPTION=
  (CONNECT_TIMEOUT=timeout)
  (ADDRESS_LIST=(ADDRESS=(PROTOCOL=tcp)(HOST=node)(PORT=1521)))
  (CONNECT_DATA=(sid-or-service-entry)))
```

Where:

alias-name is the name you use to identify the alias.

timeout specifies a timeout period (in seconds), which enables you to terminate a connection attempt without having to wait for a TCP timeout. The (CONNECT_TIMEOUT=*timeout*) parameter is optional.

node is the hostname or IP address for the virtual machine that you want to use.

sid-or-service-entry identifies the database SID or service name using one of the following formats:

- SID=*sid-name*. For example, SID=S12C1.
- SERVICE_NAME=*service-name*. For example, SERVICE_NAME=PDB1.example.oraclecloudatcust.com.

Alternatively, you can use the easy connect method to specify a connect descriptor with the following format:

```
node:1521/sid-or-service-entry
```

For example:

```
exa1node01.example.com:1521/S12C1
```

Or

```
exa1node01.example.com:1521/PDB1.example.oraclecloudatcust.com
```

Manage Oracle Exadata Database Service on Exascale Infrastructure

Use the provided tools to manage the Infrastructure.

- Using the Console to Provision Oracle Exadata Database Service on Exascale Infrastructure
  Learn how to provision an Oracle Exadata Database Service on Exascale Infrastructure system.
- Using the API to Create Infrastructure Components
  See how to use the API for common administrative tasks
- Using the API to Manage Oracle Exadata Database Service on Exascale Infrastructure Instance
  Use these API operations to manage Oracle Exadata Database Service on Exascale Infrastructure instance components.

Using the Console to Provision Oracle Exadata Database Service on Exascale Infrastructure

Learn how to provision an Oracle Exadata Database Service on Exascale Infrastructure system.

- Lifecycle Management Operations
- Network Management Operations
- Management Tasks for the Oracle Cloud Infrastructure Platform
- Oracle Database License Management Tasks
  Learn about licsensing for Oracle Exadata Database Service on Exascale Infrastructure

Lifecycle Management Operations

- To check the status of a cloud VM cluster
- To start, stop, or restart an Oracle Exadata Database Service on Exascale Infrastructure cloud VM cluster

To check the status of a cloud VM cluster

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Click **Exadata VM Clusters** under **Oracle Exadata Database Service on Exascale Infrastructure**.
4. In the list of cloud VM clusters, find the cluster you're interested in and check its icon. The icon text indicates the status of the system. The following lifecycle states apply to the cloud VM cluster:
   - **Provisioning:** Resources are being reserved for the Cloud Exadata infrastructure resource. Provisioning can take several minutes. The resource is not ready to use yet.

- **Available:** The Cloud Exadata infrastructure was successfully provisioned. You can create a cloud VM cluster on the resource to complete the infrastructure provisioning.
- **Updating:** The Cloud Exadata infrastructure is being updated. The resource goes into the updating state during management tasks. For example, when moving the resource to another compartment, or creating a cloud VM cluster on the resource.
- **Terminating:** The Cloud Exadata infrastructure is being deleted by the terminate action in the Console or API.
- **Terminated:**The Cloud Exadata infrastructure has been deleted and is no longer available.
- **Failed:** An error condition prevented the provisioning or continued operation of the Cloud Exadata infrastructure.

To view the status of a virtual machine (database node) in the cloud VM cluster, under Resources, click **Virtual Machines** to see the list of virtual machines. In addition to the states listed for a cloud VM cluster, a virtual machine's status can be one of the following:

- **Starting:** The database node is being powered on by the start or reboot action in the Console or API.
- **Stopping:** The database node is being powered off by the stop or reboot action in the Console or API.
- **Stopped:** The database node was powered off by the stop action in the Console or API.

To start, stop, or restart an Oracle Exadata Database Service on Exascale Infrastructure cloud VM cluster

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Navigate to the cloud VM cluster or DB system you want to start, stop, or reboot:

   Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.
4. Under **Resources**, click **Virtual Machines** to display the compute nodes of the cloud service instance. Click the Actions menu (

   ⋮

) for a node and then click one of the following actions:

   - **Start:**;Restarts a stopped node. After the node is restarted, the **Stop** action is enabled.
   - **Stop:** Shuts down the node. After the node is powered off, the **Start** action is enabled.
   - **Reboot:** Shuts down the node, and then restarts it.

> **Note:**
>
> - For billing purposes, the **Stop** state has no effect on the resources you consume. Billing continues for virtual machines or nodes that you stop, and related resources continue to apply against any relevant quotas. You must **Terminate** a cloud VM cluster to remove its resources from billing and quotas.
> - After you restart or restart a node, the floating IP address might take several minutes to be updated and display in the Console.

Network Management Operations

- To edit the network security groups (NSGs) for your client or backup network

To edit the network security groups (NSGs) for your client or backup network

Your client and backup networks can each use up to five network security groups (NSGs). Note that if you choose a subnet with a security list, the security rules for the cloud VM cluster or DB system will be a union of the rules in the security list and the NSGs. For more information, see Network Security Groups and Network Setup for Oracle Exadata Database Service on Exascale Infrastructure Instances.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Navigate to the cloud VM cluster or DB system you want to manage:
   Cloud VM clusters (new resource model): Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.

   DB systems: Under Bare Metal, VM, and Exadata, click **DB Systems**. In the list of DB systems, find the Exadata DB system you want to access, and then click its name to display details about it.
4. In the **Network** details, click the **Edit** link to the right of the **Client Network Security Groups** or **Backup Network Security Groups** field.
5. In the **Edit Network Security Groups** dialog, click **+ Another Network Security Group** to add an NSG to the network.

   To change an assigned NSG, click the drop-down menu displaying the NSG name, then select a different NSG.

   To remove an NSG from the network, click the **X**;icon to the right of the displayed NSG name.
6. Click **Save**.

Management Tasks for the Oracle Cloud Infrastructure Platform

- To view a work request for your Oracle Exadata Database Service on Exascale Infrastructure resources

- To move an Exadata Database Service on Exascale Infrastructure resource to another VM cluster
- To manage tags for your Oracle Exadata Database Service on Exascale Infrastructure resources

To view a work request for your Oracle Exadata Database Service on Exascale Infrastructure resources

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**.
2. Choose your **Compartment**.

   A list of DB systems is displayed.
3. Find the Cloud Exadata infrastructure, cloud VM cluster, DB system or database resource you're interested in, and click the name.
4. In the **Resources** section, click **Work Requests**. The status of all work requests appears on the page.
5. To see the log messages, error messages, and resources that are associated with a specific work request, click the operation name. Then, select an option in the **More information** section.

   For associated resources, you can click the Actions icon (three dots) next to a resource to copy the resource's OCID.

Related Topics

- Work Requests

To move an Exadata Database Service on Exascale Infrastructure resource to another VM cluster

> **Note:**
>
> - To move resources between compartments, VM cluster users must have sufficient access permissions on the compartment to which the VM cluster is being moved, as well as the current compartment. For more information about permissions for Database resources, see "Details for the Database Service".
> - If your Oracle Exadata Database Service on Exascale Infrastructure VM cluster is in a security zone, then the destination compartment must also be in a security zone. See "Security Zone Policies" for a full list of policies that affect Database service resources.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**.
2. Choose your **Compartment**.
3. Navigate to the Cloud Exadata infrastructure, cloud VM cluster that you want to move:

   Under **Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.
4. Click **Move Resource**.

5. Select the new compartment.
6. Click **Move Resource**.

Related Topics

- Details for the Database Service
- Security Zone Policies

To manage tags for your Oracle Exadata Database Service on Exascale Infrastructure resources

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Find the Cloud Exadata infrastructure, cloud VM cluster, DB system or database resource you're interested in, and click the name.
4. Click the **Tags** tab to view or edit the existing tags. Or click **More Actions** and then **Apply Tags** to add new ones.

Related Topics

- Resource Tags

Oracle Database License Management Tasks

Learn about licsensing for Oracle Exadata Database Service on Exascale Infrastructure

- To manage your BYOL database licenses
  If you want to control the number of database licenses that you run at any given time, you can scale up or down the number of ECPUs on the instance. These additional licenses are metered separately.
- To change the license type of a cloud VM cluster or DB system

To manage your BYOL database licenses

If you want to control the number of database licenses that you run at any given time, you can scale up or down the number of ECPUs on the instance. These additional licenses are metered separately.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Navigate to the cloud VM cluster or DB system you want to scale:

   Under **Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.
4. Click **Scale VM Cluster** (for cloud VM clusters) or **Scale CPU;Cores** (for DB systems) and then specify a new number of CPU cores. The text below the field indicates the acceptable values, based on the shape used when the DB system was launched.
5. Click **Update**.

To change the license type of a cloud VM cluster or DB system

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Navigate to the cloud VM cluster or DB system you want to manage:

   Under **Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.
4. On the resource details page, click **Update License Type**.

   The dialog displays the options with your current license type selected.
5. Select the new license type.
6. Click **Save**.

Using the API to Create Infrastructure Components

See how to use the API for common administrative tasks

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to create Oracle Exadata Database Service on Exascale Infrastructure components.

Exascale Database Storage Vault resource

- CreateExascaleDbStorageVaul
- GetExascaleDbStorageVault
- ListExascaleDbStorageVaults

Exadata VM cluster resource

- CreateExadbVmCluster
- GetExadbVmCluster
- ListExadbVmClusters

Databases

- GetDatabase
- ListDatabases

Database Versions

- ListDbVersions

Database Homes

- CreateDbHome
- GetDbHome

- ListDbHomes

Using the API to Manage Oracle Exadata Database Service on Exascale Infrastructure Instance

Use these API operations to manage Oracle Exadata Database Service on Exascale Infrastructure instance components.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Exascale Database Storage Vault resource

- ChangeExascaleDbStorageVaultCompartment
- CreateExascaleDbStorageVault
- DeleteExascaleDbStorageVault
- GetExascaleDbStorageVault
- ListExascaleDbStorageVaults
- UpdateExascaleDbStorageVault

Exadata VM cluster

- ChangeExadbVmClusterCompartment
- CreateExadbVmCluster
- DeleteExadbVmCluster
- GetExadbVmCluster
- ListExadbVmClusters
- RemoveVirtualMachineFromExadbVmCluster
- UpdateExadbVmCluster

Virtual machines nodes (all Oracle Exadata Database Service on Exascale Infrastructure instances)

- DbNodeAction
- ListDbNodes
- GetDbNode

Manage VM Clusters

Learn how to manage your VM clusters on Oracle Exadata Database Service on Exascale Infrastructure.

- Using the Console to Manage VM Clusters on Oracle Exadata Database Service on Exascale Infrastructure
  Learn how to use the console to create, edit, and manage your VM Clusters on Oracle Exadata Database Service on Exascale Infrastructure.

- Adding or Removing a VM From a VM Cluster
  You can scale VM Clusters horizontally by adding or removing VMs to or from an existing VM Cluster.
- Overview of Automatic Diagnostic Collection
  By enabling diagnostics collection and notifications, Oracle Cloud Operations and you will be able to identify, investigate, track, and resolve guest VM issues quickly and effectively. Subscribe to Events to get notified about resource state changes.
- Incident Logs and Trace Files
  This section lists all of the files that can be collected by Oracle Support if you opt-in for incident logs and trace collection.
- Health Metrics
  Review the list of database and non-database health metrics collected by Oracle Trace File Analyzer.
- Using the API to Manage Oracle Exadata Database Service on Exascale Infrastructure Instance
  Use these API operations to manage Exadata Cloud Infrastructure virtual machines (VMs) and databases on Oracle Exadata Database Service on Exascale Infrastructure (ExaDB-XS).

Related Topics

- Application Checklist for Continuous Service for MAA Solutions

Using the Console to Manage VM Clusters on Oracle Exadata Database Service on Exascale Infrastructure

Learn how to use the console to create, edit, and manage your VM Clusters on Oracle Exadata Database Service on Exascale Infrastructure.

- To create a cloud VM cluster
  Create a VM cluster in an Oracle Exadata Database Service on Exascale Infrastructure instance.
- Using the Console to Enable, Partially Enable, or Disable Diagnostics Collection
  You can enable, partially enable, or disable diagnostics collection for your Guest VMs after provisioning the VM cluster. Enabling diagnostics collection at the VM cluster level applies the configuration to all the resources such as DB home, Database, and so on under the VM cluster.
- Using the Console to Update the License Type on a VM Cluster
  To modify licensing, be prepared to provide values for the fields required for modifying the licensing information.
- To scale VM Clusters
  Increase or decrease the ECPUs, memory or storage available to a VM cluster in Oracle Exadata Database Service on Exascale Infrastructure
- To add SSH keys to a VM cluster
  The VM cluster exists, and you wish to add a another user which requires another SSH key.
- Using the Console to Add SSH Keys After Creating a VM Cluster
- Using the Console to Stop, Start, or Reboot a VM Cluster Virtual Machine
  Use the console to stop, start, or reboot a virtual machine.

- Using the Console to Check the Status of a VM Cluster Virtual Machine
  Review the health status of a VM cluster virtual machine.
- Using the Console to Move a VM Cluster to Another Compartment
  To change the compartment that contains your VM cluster on Oracle Exadata Database Service on Exascale Infrastructure, use this procedure.
- To change the VM cluster display name
- Using the Console to Terminate a VM Cluster
  Before you can terminate a VM cluster, you must first terminate the databases that it contains.
- To view details about private DNS configuration

To create a cloud VM cluster

Create a VM cluster in an Oracle Exadata Database Service on Exascale Infrastructure instance.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Click **Exadata VM Clusters**.
3. Click **Create VM Cluster**.

   The **Create VM Cluster** page is displayed. Provide the required information to configure the VM cluster.
4. **Compartment**: Select a compartment for the VM cluster resource.
5. **Display name**: Enter a user-friendly display name for the VM cluster. The name doesn't need to be unique. An Oracle Cloud Identifier (OCID) will uniquely identify the DB system. Avoid entering confidential information.
6. **Provide the cluster name**: Select the name of the VM cluster.
7. **Select an availability domain**: Select the availability domain from the displayed options available.
8. **Configure the VM cluster**: Provide the following information:
   - **Number of VMs in the cluster**: Specify the number of the VMs that you want to configure for the cluster, between 2 and 10.
   - **ECPUs enabled per VM**: Specify the number of ECPU cores that you want to enable for the VM cluster. The minimum is 8 ECPU. The maximum number of ECPUs is 200 per VM, or limited by the number of total ECPUs you have specified for the VM. The value you select must be a multiple of 4. You can open the reserve additional ECPU section to reserve additional ECPUs.
   - **Total ECPUs enabled across cluster per VM**: Provide a total number of ECPUs to allocate per VM. The total must be a number between 8 and 200.
9. (Optional) To reserve additional ECPUs, click **Show reserve additonal ECPU**. Provide the following information:
   - **ECPUs additional reserved per VM (read only)**: Indicates the additional reserved ECPUs. The number of additional ECPUs will be automatically calculated based on the total enabled ECPUs. Additional reserved ECPUs are not active for licensing purposes but are reserved for your VM, and ready and waiting for scaling the Enabled ECPUs.

- **Total ECPUs per VM**: Provide a total number of ECPUs to allocate per VM. The total must be a number between 8 and 200.
- **Memory per VM (GB)**: This is a read-only field. It displays amount of memory allocated to each VM. Memory is calculated based on 11 GB per total cores. The **Total memory across VM Cluster (GB)** field automatically updates to provide you with the total amount of memory allocated across the VM cluster, based on the memory allocation per VM that you specify.

10. **VM file system storage capacity per VM (GB)**: Specify storage capacity per VM in gigabytes (GB).

    Provide how much storage you want for all VM filesystems together. The VM Filesystems storage includes `/u02` capacity, where your Database Homes will go, along with all of the other VM filesystems (`/`, `/boot`, `/tmp`, `/var`, `/var/log`, `/var/log/audit`, `/home`, `swap`, `kdump`, `/u01`, `grid`, `/u02`). Any extra capacity selected beyond system minimums will go into `/u02`.

    > **Note:**
    >
    > For information about reserved and enabled cores, and an overview of the ExaDB-XS architecture, see "About Exadata Database Service on Exascale Infrastructure"

11. **Exascale Database Storage Vault**: Select either **Create new vault** or **Select existing vault**. If you select an existing vault, then select the vault in the compartment. Click **Change compartment** to select a vault in a different compartment.

    When you create a new vault, the Provisioning status window opens to provide you with the status of vault creation, and the name of the vault that is being created in the format `Vault-`*`YYYYMMDDHHMM`* indicating the creation date, where *YYYY* is the year, *MM* is the month, *DD* is the day, *HH* is the hour, and *MM* is the minute.

    > **Note:**
    >
    > If the vault creation failed, then the Provisioning status window provides you with the work request error message indicating the point where the vault creation operation failed, and the work request ID. Make a note of this work request ID, and open a Service Request with My Oracle Support.

12. **Configure Exascale Database Storage Vault**: Select the storage configuration to use for your database's storage. To begin, select whether you want to create a new Vault, or use an existing Vault.

    For a new Vault, specify the following:

    - **Storage Vault Name**: Name the new Exascale Vault. *Optional*: Use the link provided to change to another compartment where you want to place the Vault.
    - **Enter the Storage Capacity for Databases**: The amount of usable disk storage capacity that will be available for storing databases that is desired. Specify the size in gigabytes (GB) between 300 to 100,000.
    - (Optional) **Add smart flash as a percentage of storage capacity provisioned (%)**: Select this option to purchase and specify an additional amount of flash cache over and above the amount of default flash cache that is included in the normal Storage capacity for Databases. Additional flash cache can potentially enable increased

performance without adding additional storage capacity in some workloads. Additional flash cache also includes additional memory cache. Specify the additional flash cache as a percentage of the total storage provisioned. If you wish to provision additional flash cache, you must add at least 100 GB of additional flash cache. The amount of smart flash cache in GB that will be added is specified in the read-only field **Smart flash cache to be added (GB)**.

The minimum size configuration for an Exascale Database Storage Vault is 300 GB. 50 GB of the space that you allocate in your Vault is reserved for a 200 GB ACFS file system. This ACFS file system resides within your Exascale Database Storage Vault, but is reserved for system use. Thus, if you provisioned the minimum of 300 GB in your Exascale Database Storage Vault, then 250 GB of that 300 GB capacity will be available storage for your databases.

13. **Select the Oracle Grid Infrastructure version**: This field displays the Oracle Grid Infrastructure versions available for deployment in the VM cluster.
14. **Add SSH key**:Add the public key portion of each key pair that you want to use for SSH access to the DB system:
    - **Generate SSH key pair** (Default option) Select this option to generate an SSH keypair. Then in the dialog below click **Save private key** to download the key, and optionally click **Save public key** to download the key.

      > **Note:**
      >
      > Download the private key so that you can connect to the database system using SSH. It will not be shown again.

    - **Upload SSH key files**: Select this option to browse or drag and drop `.pub` files.
    - **Paste SSH keys**: Select this option to paste in individual public keys.
15. **Configure the network settings**: Specify the following:
    - **Virtual cloud network**: Select the virtual cloud network (VCN) for the compartment in which you want to create the VM cluster. Click **Change Compartment** to select a VCN in a different compartment.
    - **Client subnet**: Select the client subnet in the compartment. This is the subnet to which the VM cluster should attach. Click **Change Compartment** to select a subnet in a different compartment.

      > **Note:**
      >
      > You must select the VCN before you can select a client subnet.

      Do not use a subnet that overlaps with 192.168.16.16/28, which is used by the Oracle Clusterware private interconnect on the database instance. Specifying an overlapping subnet causes the private interconnect to malfunction.
    - **Backup subnet**: Select the subnet to use for the backup network, which is typically used to transport backup information to and from the **Backup Destination**, and for Data Guard replication. Click **Change Compartment** to select a subnet in a different compartment, if applicable.

      Do not use a subnet that overlaps with 192.168.128.0/20. This restriction applies to both the client subnet and backup subnet.

> **Note:**
>
> You must select the VCN before you can select a backup client subnet.

- **Use network security groups to control traffic**: Optionally, you can specify one or more network security groups (NSGs) for both the client and backup networks. NSGs function as virtual firewalls, allowing you to apply a set of ingress and egress **security rules** to your Oracle Exadata Database Service on Exascale Infrastructure VM cluster.

  Note that if you choose a subnet with a **security list**, then the security rules for the VM cluster will be a union of the rules in the security list and the NSGs.

  To use network security groups:

  – Check the **Use network security groups to control traffic** check box. This box appears under both the selector for the client subnet and the backup subnet. You can apply NSGs to either the client or the backup network, or to both networks. Note that you must have a virtual cloud network selected to be able to assign NSGs to a network.

  – Specify the NSG to use with the network. You might need to use more than one NSG. If you're not sure, contact your network administrator.

- **Hostname prefix** Provide your choice of hostname for the Exadata DB system. The host name must begin with an alphabetic character and can contain only alphanumeric characters and hyphens (-). The maximum number of characters allowed for an Exadata DB system is 12.

  > **Caution:**
  >
  > The hostname must be unique within the subnet. If it is not unique, then the VM cluster will fail to provision.

- **Host domain name**: The domain name for the VM cluster. This is a read-only field. Make a note of the host domain name for your reference.

  If you plan to store database backups in Object Storage or Autonomous Recovery service, Oracle recommends that you use a VCN Resolver for DNS name resolution for the client subnet because it automatically resolves the Swift endpoints used for backups.

- **Host and domain URL** This read-only field combines the host and domain names to display the fully qualified domain name (FQDN) for the database. The maximum length is 63 characters.

16. **Choose a license type**: The type of license that you want to use for the VM cluster. Your choice affects metering for billing.

    - **License Included** means the cost of the cloud service includes a license for the Database service.

    - **Bring Your Own License (BYOL)** means you are an Oracle Database customer with an Unlimited License Agreement or Non-Unlimited License Agreement, and you want to use your license with Oracle Cloud Infrastructure. This option removes the need for separate on-premises licenses and cloud licenses.

17. Click **Create Exadata VM Cluster**.

18. (Optional) **Provide a contact for your VM Cluster**. Exadata Database Service on Exascale Infrastructure leverages the OCI Announcements Service. Oracle recommends

that you provide your contact details here. Oracle then automatically subscribes you to announcements relevant to this service, including maintenance and outage notifications, among others. If you do not choose to provide a contact now, then you will have to subscribe to announcements manually later, leveraging the OCI Announcements Service directly. To learn more about subscribing, see Subscribing to Announcements.

19. Click **Show Advanced Options** to specify advanced options for the VM cluster:

    - **Time zone:** This option is located in the **Management** tab. The default time zone for the DB system is UTC, but you can specify a different time zone. The time zone options are those supported in both the `Java.util.TimeZone` class and the Oracle Linux operating system. For more information, see *DB System Time Zone* .

      > **Note:**
      >
      > If you want to set a time zone other than UTC or the browser-detected time zone, and if you do not see the time zone you want, try selecting the **Select another time zone**, option, then selecting "Miscellaneous" in the **Region or country** list and searching the additional **Time zone** selections.

    - **SCAN Listener Port**: This option is located in the **Network** tab. You can assign a SCAN listener port (TCP/IP) in the range between 1024 and 8999. The default is 1521

      > **Note:**
      >
      > Manually changing the SCAN listener port of a VM cluster after provisioning using the backend software is not supported. This change can cause Data Guard provisioning to fail.

      .

    - **Tags**: If you have permissions to create a resource, then you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see *Resource Tags* . If you are not sure whether to apply tags, skip this option (you can apply tags later) or ask your administrator.

Related Topics

- Network Security Groups
- Security Lists
- Resource Tags
- Overview of Database Service Events
  The Database Service Events feature implementation enables you to be notified about health issues with your Oracle Databases, or with other components on the Guest VM.

Using the Console to Enable, Partially Enable, or Disable Diagnostics Collection

You can enable, partially enable, or disable diagnostics collection for your Guest VMs after provisioning the VM cluster. Enabling diagnostics collection at the VM cluster level applies the configuration to all the resources such as DB home, Database, and so on under the VM cluster.

> **Note:**
>
> - You are opting in with the understanding that the list of events, metrics, and log files collected can change in the future. You can opt-out of this feature at any time.
> - Oracle may add more metrics in the future, but if you have already chosen to collect metrics, you need not update your opt-in value. It will remain enabled/ disabled based on your current preference.
> - If you have previously opted in for incident log and trace file collection and decide to opt out when Oracle Cloud operations run a log collection job, then the job will run its course and will not cancel. Future log collections won't happen until you opt-in again to the incident logs and trace file collection option.

1. Open the navigation menu. Under **Database**, click **Exadata Database Service on Exascale Infrastructure**.
2. Choose the **Region** that contains your Exadata infrastructure.
3. Click **VM Clusters**.
4. Click the name of the VM cluster you want to enable or disable diagnostic data collection.
5. On the VM Cluster Details page, under **General Information**, enable, partially enable, or disable **Diagnostics Collection** beside **Diagnostics Collection**.
6. In the **Edit Diagnostics Collection Settings** dialog, enable or disable any of the Diagnostics Collections. By enabling diagnostics collection and notifications, Oracle Cloud Operations and you will be able to identify, investigate, track, and resolve guest VM issues quickly and effectively. Subscribe to Events to get notified about resource state changes.
   - **Enable Diagnostics Events** Allow Oracle to collect and publish critical, warning, error, and information events to me. For more information, see *Overview of Database Service Events*
   - **Enable Health Monitoring** Allow Oracle to collect health metrics/events such as Oracle Database up/down, disk space usage, and so on, and share them with Oracle Cloud operations. You will also receive notification of some events.
   - **Enable Incident logs and trace collection**. Allow Oracle to collect incident logs and traces to enable fault diagnosis and issue resolution.
     Note: You had previously opted in for incident log and trace file collection and decide to opt-out when Oracle Cloud operations run a log collection job, the job will run its course and will not cancel. Future log collections will not run until you opt-in again to the incident logs and trace file collection option.
7. Select or clear the checkboxes and then click **Save Changes**.

Related Topics

- Overview of Database Service Events
  The Database Service Events feature implementation enables you to be notified about health issues with your Oracle Databases, or with other components on the Guest VM.

Using the Console to Update the License Type on a VM Cluster

To modify licensing, be prepared to provide values for the fields required for modifying the licensing information.

1. Open the navigation menu. Under **Oracle Database**, click **Oracle Exadata Database Service on Exascale Infrastructure**.
2. Choose the **Region** and **Compartment** that contains the VM cluster for which you want to update the license type.
3. Click **VM Clusters**.
4. Click the name of the VM cluster for which you want to update the license type.

   The VM Cluster Details page displays information about the selected VM cluster.
5. Click **Update License Type**.
6. In the dialog box, choose one of the following license types and then click **Save Changes**.
   - **Bring Your Own License (BYOL):** Select this option if your organization already owns Oracle Database software licenses that you want to use on the VM cluster.
   - **License Included:** Select this option to subscribe to Oracle Database software licenses as part of Oracle Exadata Database Service on Exascale Infrastructure.

   Updating the license type does not change the functionality or interrupt the operation of the VM cluster.

To scale VM Clusters

Increase or decrease the ECPUs, memory or storage available to a VM cluster in Oracle Exadata Database Service on Exascale Infrastructure

Scaling up or down VM cluster resources requires thorough auditing of existing usage and capacity management by the customer DB administrator. Review the existing usage to avoid failures during or after a scale down operation. While scaling up, consider how much of these resources are left for the next VM cluster you are planning to create. Oracle Exadata Database Service on Exascale Infrastructure tooling calculates the current usage of memory, local disk, and ASM storage in the VM cluster, adds headroom to it, and arrives at a minimum value below which you cannot scale down, and expects that you specify the value below this minimum value.

- You can scale ECPUs enabled per VM. Keep in mind that memory scales with the total ECPU count.

1. Navigate to the **VM Cluster Details** page
2. Click **Scale VM Cluster**.

   The **Configure the VM Cluster** window opens, and displays the current configuration of your VM cluster. .
3. Scale your VM cluster as required:
   - **ECPUs enabled per VM**: Specify the number of ECPU cores that you want to enable for the VM cluster. The minimum is 8 ECPU. The maximum number of ECPUs is 200 per VM, or limited by the number of total ECPUs you have specified for the VM. The value you select must be a multiple of 4. You can open the reserve additional ECPU section to reserve additional ECPUs.

- **ECPUs additional reserved per VM (read only)**: Indicates the additional reserved ECPUs. The number of additional ECPUs will be automatically calculated based on the total enabled ECPUs. Additional reserved ECPUs are not active for licensing purposes but are reserved for your VM, and ready and waiting for scaling the Enabled ECPUs.
- **Total ECPUs per VM**: Provide a total number of ECPUs to allocate per VM. The total must be a number between 8 and 200.
- **Memory per VM (GB)**: This is a read-only field. It displays amount of memory allocated to each VM. Memory is calculated based on 11 GB per total cores. The **Total memory across VM Cluster (GB)** field automatically updates to provide you with the total amount of memory allocated across the VM cluster, based on the memory allocation per VM that you specify.

4. **VM file system storage capacity per VM (GB)**: Specify storage capacity per VM in gigabytes (GB).

   Provide how much storage you want for all VM filesystems together. The VM Filesystems storage includes `/u02` capacity, where your Database Homes will go, along with all of the other VM filesystems (`/`, `/boot`, `/tmp`, `/var`, `/var/log`, `/var/log/audit`, `/home`, `swap`, `kdump`, `/u01`, `grid`, `/u02`). Any extra capacity selected beyond system minimums will go into `/u02`.

   > **Note:**
   >
   > For information about reserved and enabled cores, and an overview of the ExaDB-XS architecture, see "About Exadata Database Service on Exascale Infrastructure"

To add SSH keys to a VM cluster

The VM cluster exists, and you wish to add a another user which requires another SSH key.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Click **Exadata VM Clusters**.
4. In the list of VM clusters, find the cluster you want to manage and click its highlighted name.
5. Click **Add SSH Keys**.
6. Select one of the following options:
   - **Generate SSH key pair**: Use this option to create a new SSH key pair. Click both **Save Private Key** and **Save Public Key** when using this option. The private key is downloaded to your local machine, and should be stored in a safe location. You cannot download another copy of the private key generated during this operation after completing the operation.
   - **Upload SSH key files**: Select this option to browse or drag and drop .pub files.
   - **Paste SSH keys**: Select this option to paste in individual public keys. To paste multiple keys, click **+ Another SSH Key**, and supply a single key for each entry.
7. Click **Save Changes**.

Using the Console to Add SSH Keys After Creating a VM Cluster

1. Open the navigation menu. Under **Oracle Database**, click **Exadata Database Service on Exascale Infrastructure**.
2. Click **VM Clusters**.
3. Click the name of the VM cluster that you want to add SSH key(s).
4. In the VM Cluster Details page, click **Add SSH Keys**.
5. In the ADD SSH Keys dialog, choose any one of the methods:
   - **Generate SSH key pair:** Select this option if you want the Control Plane to generate public/private key pairs for you.
     Click **Save Private Key** and **Save Public Key** to download and save SSH Key pair.
   - **Upload SSH key files:** Select this option to upload the file that contains SSH Key pair.
   - **Paste SSH keys:** Select this option to paste the SSH key string.
     To provide multiple keys, click **Another SSH Key**. For pasted keys, ensure that each key is on a single, continuous line. The length of the combined keys cannot exceed 10,000 characters.
6. Click **Save Changes**.

Related Topics

- Managing Key Pairs on Linux Instances

Using the Console to Stop, Start, or Reboot a VM Cluster Virtual Machine

Use the console to stop, start, or reboot a virtual machine.

1. Open the navigation menu. Under **Oracle Database**, click **Exadata Database Service on Exascale Infrastructure**.
2. Choose the **Region** and **Compartment** that is associated with the VM cluster that contains the virtual machine that you want to stop, start, or reboot.
3. Click **VM Clusters**.
4. Click the name of the VM cluster that contains the virtual machine that you want to stop, start, or reboot.

   The VM Cluster Details page displays information about the selected VM cluster.
5. In the **Resources** list, click **Virtual Machines**.

   The list of virtual machines is displayed.
6. In the list of nodes, click the **Actions** icon (three dots) for a node, and then click one of the following actions:
   a. **Start:** Restarts a stopped node. After the node is restarted, the **Stop** action is enabled.
   b. **Stop:** Shuts down the node. After the node is stopped, the **Start** action is enabled.
   c. **Reboot:** Shuts down the node, and then restarts it.

Using the Console to Check the Status of a VM Cluster Virtual Machine

Review the health status of a VM cluster virtual machine.

1. Open the navigation menu. Under **Oracle Database**, click **Exadata Database Service on Exascale Infrastructure**.
2. Choose the **Region** and **Compartment** that is associated with the VM cluster that contains the virtual machine that you are interested in.
3. Click **VM Clusters**.
4. Click the name of the VM cluster that contains the virtual machine that you are interested in.

   The VM Cluster Details page displays information about the selected VM cluster.
5. In the **Resources** list, click **Virtual Machines**.

   The list of virtual machines displays. For each virtual machine in the VM cluster, the name, state, and client IP address are displayed.
6. In the node list, find the virtual machine that you are interested in and check its state.

   The color of the icon and the associated text it indicates its status.

   - **Available:** Green icon. The node is operational.
   - **Starting:** Yellow icon. The node is starting because of a start or reboot action in the Console or API.
   - **Stopping:** Yellow icon. The node is stopping because of a stop or reboot action in the Console or API.
   - **Stopped:** Yellow icon. The node is stopped.
   - **Failed:** Red icon. An error condition prevents the continued operation of the virtual machine.

Using the Console to Move a VM Cluster to Another Compartment

To change the compartment that contains your VM cluster on Oracle Exadata Database Service on Exascale Infrastructure, use this procedure.

When you move a VM cluster, the compartment change is also applied to the virtual machines and databases that are associated with the VM cluster. However, the compartment change does not affect any other associated resources, such as the Exadata infrastructure, which remains in its current compartment.

1. Open the navigation menu. Under **Oracle Database**, click **Exadata Database Service on Exascale Infrastructure**.
2. Choose the **Region** and **Compartment** that contains the VM cluster that you want to move.
3. Click **VM Clusters**.
4. Click the name of the VM cluster that you want to move.

   The VM Cluster Details page displays information about the selected VM cluster.
5. Click **Move Resource**.
6. In the resulting dialog, choose the new compartment for the VM cluster, and click **Move Resource**.

To change the VM cluster display name

> **Note:**
>
> This topic only applies to Oracle Exadata Database Service on Exascale Infrastructure instances using the new Oracle Exadata Database Service on Exascale Infrastructuree instance resource model.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Click **Exadata VM Clusters** under **Oracle Exadata Database Service on Exascale Infrastructure**.
4. In the list of Exadata VM Clusters resources, click the name of the VM Cluster you're interested in
5. On rthe **Infrastructure Details** page, click **More Actions** and **Update Display Name** .
6. In the **Update Display Name** dialog, Enter the **New display name**, and the **current display name** as instructed.
7. Click **Update Display Name**.

Using the Console to Terminate a VM Cluster

Before you can terminate a VM cluster, you must first terminate the databases that it contains.

Terminating a VM cluster removes it from the Cloud Control Plane. In the process, the virtual machines and their contents are destroyed.

1. Open the navigation menu. Under **Oracle Database**, click **Exadata Database Service on Exascale Infrastructure**.
2. Choose the **Region** and **Compartment** that contains the VM cluster that you want to terminate.
3. Click **VM Clusters**.
4. Click the name of the VM cluster that you want to terminate.

   The VM Cluster Details page displays information about the selected VM cluster.
5. Click **More Actions**, and then click **Terminate**.
6. In the resulting dialog:
   - Review the message about the backup retention policy
   - Enter the name of the VM cluster
   - Click **Terminate VM Cluster** to confirm the action.

> **Note:**
>
> The database stays in a terminated state with backups listed until all backups are expired.
>
> The Exascale Vault that had been associated with the VM Cluster survives the deletion of the VM Cluster. This is because Exascale Vaults can be shared among multiple VM Clusters. If the VM Cluster you've terminated was the only one using the VM Cluster, then you should also terminate the Exascale Vault to stop billing related to the Database Storage. See *Managing Exascale Database Storage Vaults* for more information.

To view details about private DNS configuration

1. Open the navigation menu. Under **Database**, click **Exadata Database Service on Exascale Infrastructure**.
2. Choose the **Region** that contains your Exadata infrastructure.
3. Choose the **Compartment** that contains your Exadata infrastructure.
4. Click **VM Clusters**.
5. Click the name of the VM cluster that is configured with a private DNS you want to view.
6. Under the Network section, Private DNS and Private Zone are displayed, if a private DNS is configured.
7. Click the **Private View** name to edit the configuration.

Related Topics

- Using the Console to manage private DNS

Adding or Removing a VM From a VM Cluster

You can scale VM Clusters horizontally by adding or removing VMs to or from an existing VM Cluster.

- Add a VM to a VM Cluster
  Add a Virtual Machine to a VM Cluster
- Terminate a VM from a VM Cluster
  To remove a virtual machine from a provisioned cluster, use this procedure.

Add a VM to a VM Cluster

Add a Virtual Machine to a VM Cluster

> **Note:**
>
> - This operation is only available with Multi-VM enabled Infrastructure.
> - To add a VM to a VM Cluster requires that all TCP ports are open for the client subnet CIDR for ingress and egress.

1. Open the navigation menu. Under **Oracle Database**, click **Exadata Database Service on Exascale Infrastructure**.
2. Choose the **Region** and **Compartment** that contains the VM cluster that you want to scale.
3. Click **VM Clusters**.
4. Click the name of the VM cluster to which you want to add a virtual machine.
5. Under Resources, select **Virtual Machines**, and click the **Add Virtual Machines** button.
6. In the Add Virual Machines window, select the DB server where you want the new VM to reside.

   > **Note:**
   >
   > The VM that is added will have the same resources as the other VMs in the cluster.

7. Click **Add Virtual Machine**.

> **Note:**
>
> Add a VM to a VM Cluster is NOT supported using Terraform.

Terminate a VM from a VM Cluster

To remove a virtual machine from a provisioned cluster, use this procedure.

1. Open the navigation menu. Under **Oracle Database**, click **Exadata Database Service on Exascale Infrastructure**.
2. Choose the **Region** and **Compartment** that contains the VM cluster that you want to scale.
3. Click **VM Clusters**.
4. Click the name of the VM cluster for which you want to remove a virtual machine.
5. On the Exadata VM Cluster Details page, in the Virtual Machines section, select the Virtual Machine that will be removed, click the more commands symbol (three dots) and click **Terminate**

> **Note:**
>
> Remove a VM from a VM Cluster is not supported using Terraform at this time.

Overview of Automatic Diagnostic Collection

By enabling diagnostics collection and notifications, Oracle Cloud Operations and you will be able to identify, investigate, track, and resolve guest VM issues quickly and effectively. Subscribe to Events to get notified about resource state changes.

- **Enable Diagnostic Events**

Allow Oracle to collect and publish critical, warning, error, and information events to you. For more information, see *Database Service Events*.

- **Enable Health Monitoring**

  Allow Oracle to collect health metrics/events such as Oracle Database up/down, disk space usage, and so on, and share them with Oracle Cloud operations. You will also receive notification of some events. For more information, see *Health Metrics*.

- **Enable Incident Logs and Trace Collection**

  Allow Oracle to collect incident logs and traces to enable fault diagnosis and issue resolution. For more information, see *Incident Logs and Trace Files*.

Diagnostics Collection is:

- **Enabled:** When you choose to collect diagnostics, health metrics, incident logs, and trace files (all three options).
- **Disabled:** When you choose not to collect diagnostics, health metrics, incident logs, and trace files (all three options).
- **Partially Enabled:** When you choose to collect diagnostics, health metrics, incident logs, and trace files (one or two options).

Disabling diagnostic events and health monitoring will only stop the collection and notification of data/events from the time you uncheck the checkboxes tied to the options. However, historical data will not be purged from Oracle Cloud Operations data repositories.

Incident Logs and Trace Files

This section lists all of the files that can be collected by Oracle Support if you opt-in for incident logs and trace collection.

> **Note:**
>
> - Oracle will create a service request (SR) against the infrastructure Customer Support Identifier (CSI) when an issue is detected and needs customer interaction to resolve.
> - The customer's Oralce Cloud Infrastructure tenancy admin email will be used as the CSI contact to create SR and attach logs to it. Ensure tenancy admin is added as a CSI contact in My Oracle Support (MOS).

Oracle Trace File Analyze (TFA) Component Driven Logs Collections

The directories are generally assigned to a component and that component can then be used to guide TFA to the files it needs to collect, for example, requesting the CRS component would tell TFA to look at directories mapped to the CRS component and find files that match the required collection time frame.

Note:

If have previously opted in for incident log and trace file collection and decide to opt out when Oracle Cloud operations run a log collection job, then the job will run its course and will not cancel. Future log collections won't happen until you opt-in again to the incident logs and trace file collection option.

TFA is shipped with scripts that run when a particular component is requested, for example, for CRS component, `crscollect.pl` will run a number of `crsctl` commands and gather the input. By default, TFA does not redact collected logs.

Table 4-1 Oracle Trace File Analyze (TFA) Component Driven Logs Collections

Component	Script	Files/Directories
`OS`: Operating system logs	`oscollect.pl`	• `/var/log/messages` • OSWatcher archive • **Exadata Only:** ExaWatcher archive `/opt/oracle.ExaWatcher/archive/`

Table 4-1 (Cont.) Oracle Trace File Analyze (TFA) Component Driven Logs Collections

Component	Script	Files/Directories
CRS: Grid Infrastructure and cluster logs	crscollect.pl	• /etc/oracle • GIHOME/crf/db/HOSTNAME1 • GIHOME/crs/log • GIHOME/css/log • GIHOME/cv/log • GIHOME/evm/admin/log • GIHOME/evm/admin/logger • GIHOME/evm/log • GIHOME/log/-/client • GIHOME/log/HOSTNAME1 • GIHOME/log/HOSTNAME1/admin • GIHOME/log/HOSTNAME1/client • GIHOME/log/HOSTNAME1/crflogd • GIHOME/log/HOSTNAME1/crfmond • GIHOME/log/HOSTNAME1/crsd • GIHOME/log/HOSTNAME1/cssd • GIHOME/log/HOSTNAME1/ctssd • GIHOME/log/HOSTNAME1/diskmon • GIHOME/log/HOSTNAME1/evmd • GIHOME/log/HOSTNAME1/gipcd • GIHOME/log/HOSTNAME1/gnsd • GIHOME/log/HOSTNAME1/gpnpd • GIHOME/log/HOSTNAME1/mdnsd • GIHOME/log/HOSTNAME1/ohasd • GIHOME/log/HOSTNAME1/racg • GIHOME/log/HOSTNAME1/srvm • GIHOME/log/HOSTNAME1/xag

Table 4-1 (Cont.) Oracle Trace File Analyze (TFA) Component Driven Logs Collections

Component	Script	Files/Directories
		• `GIHOME/log/diag/asmtool` • `GIHOME/log/diag/clients` • `GIHOME/log/procwatcher/PRW_SYS_HOSTNAME1` • `GIHOME/network/log` • `GIHOME/opmn/logs` • `GIHOME/racg/log` • `GIHOME/scheduler/log` • `GIHOME/srvm/log` • `GRIDBASE/crsdata/@global/cvu` • `GRIDBASE/crsdata/HOSTNAME1/core` • `GRIDBASE/crsdata/HOSTNAME1/crsconfig` • `GRIDBASE/crsdata/HOSTNAME1/crsdiag` • `GRIDBASE/crsdata/HOSTNAME1/cvu` • `GRIDBASE/crsdata/HOSTNAME1/evm` • `GRIDBASE/crsdata/HOSTNAME1/output` • `GRIDBASE/crsdata/HOSTNAME1/ovmmwallets` • `GRIDBASE/crsdata/HOSTNAME1/scripts` • `GRIDBASE/crsdata/HOSTNAME1/trace` • `GRIDBASE/diag/crs/-/crs/cdump` • `GRIDBASE/diag/crs/HOSTNAME1/crs/cdump` • `GRIDBASE/diag/crs/HOSTNAME1/crs/incident` • `GRIDBASE/diag/crs/HOSTNAME1/crs/trace`

Table 4-1 (Cont.) Oracle Trace File Analyze (TFA) Component Driven Logs Collections

Component	Script	Files/Directories
Database: Oracle Database logs	No DB Specific Script - runs opatch lsinventory for the ORACLE_HOME the DB runs from TFA will run ipspack based on the time range for certain DB incidents.	• ORACLE_BASE/diag/rdbms/*<dbname>*/*<instance_name>*/cdump • ORACLE_BASE/diag/rdbms/*<dbname>*/*<instance_name>*/trace • ORACLE_BASE/diag/rdbms/*<dbname>*/*<instance_name>*/incident

Cloud Tool Logs

- **Creg files:** /var/opt/oracle/creg/*.ini files with masked sensitive info
- **Cstate file:** /var/opt/oracle/cstate.xml
- **Database related tooling logs:**

  If dbName specified, /var/opt/oracle/log/*<dbName>*, else collect logs for all databases /var/opt/oracle/log/

  If dbName specified, /var/opt/oracle/dbaas_acfs/log/*<dbName>*, else collect logs for all databases /var/opt/oracle/log/*<dbName>*
- **Database env files:** If dbName specified, /home/oracle/*<dbName>*.env, else collect logs for all databases /home/oracle/*.env
- **Pilot logs:** /home/opc/.pilotBase/logs
- **List of log directories:**
  - /var/opt/oracle/log
  - /var/opt/oracle/dbaas_acfs/log
  - /var/opt/oracle/dbaas_acfs/dbsystem_details
  - /var/opt/oracle/dbaas_acfs/job_manager
  - /opt/oracle/dcs/log

DCS Agent Logs

- /opt/oracle/dcs/log/

Tooling-Related Grid Infrastructure/Database Logs

- **Grid Infrastructure:** GI_HOME/cfgtoollogs
- **Database alertlog:** /u02/app/oracle/diag/rdbms/*/*/alert*.log

Health Metrics

Review the list of database and non-database health metrics collected by Oracle Trace File Analyzer.

> **Note:**
>
> Oracle may add more metrics in the future, but if you have already chosen to collect metrics, you need not update your opt-in value. It will remain enabled/disabled based on your current preference.

Guest VM Health Metrics List - Database Metrics

Table 4-2 Guest VM Health Metrics List - Database Metrics

Metric Name	Metric Display Name	Unit	Aggregation	Interval	Collection Frequency	Description
`CpuUtilization`	CPU Utilization	Percentage	Mean	One minute	Five minutes	The CPU utilization is expressed as a percentage, which is aggregated across all consumer groups. The utilization percentage is reported with respect to the number of CPUs the database is allowed to use, which is two times the number of ECPUs.
`StorageUtilization`	Storage Utilization	Percentage	Mean	One hour	One hout	The percentage of provisioned storage capacity currently in use. Represents the total allocated space for all tablespaces.

Table 4-2 (Cont.) Guest VM Health Metrics List - Database Metrics

Metric Name	Metric Display Name	Unit	Aggregation	Interval	Collection Frequency	Description
`BlockChanges`	DB Block Changes	Changes per second	Mean	One minute	Five minutes	The Average number of blocks changed per second.
`ExecuteCount`	Execute Count	Count	Sum	One minute	Five minutes	The number of user and recursive calls that executed SQL statements during the selected interval.
`CurrentLogons`	Current Logons	Count	Sum	One minute	Five minutes	The number of successful logons during the selected interval.
`TransactionCount`	Transaction Count	Count	Sum	One minute	Five minutes	The combined number of user commits and user rollbacks during the selected interval.
`UserCalls`	User Calls	Count	Sum	One minute	Five minutes	The combined number of logons, parses, and execute calls during the selected interval.
`ParseCount`	Parse Count	Count	Sum	One minute	Five minutes	The number of hard and soft parses during the selected interval.
`StorageUsed`	Storage Space Used	GB	Max	One hour	One hour	Total amount of storage space used by the database at the collection time.

Table 4-2   (Cont.) Guest VM Health Metrics List - Database Metrics

Metric Name	Metric Display Name	Unit	Aggregation	Interval	Collection Frequency	Description
`StorageAllocated`	Storage Space Allocated	GB	Max	One hour	One hour	Total amount of storage space allocated to the database at the collection time.
`StorageUsedByTablespace`	Storage Space Used By Tablespace	GB	Max	One hour	One hour	Total amount of storage space used by tablespace at the collection time. In the case of container databases, this metric provides root container tablespaces.
`StorageAllocatedByTablespace`	Allocated Storage Space By Tablespace	GB	Max	One hour	One hour	Total amount of storage space allocated to the tablespace at the collection time. In the case of container databases, this metric provides root container tablespaces.
`StorageUtilizationByTablespace`	Storage Space Utilization By Tablespace	Percentage	Mean	One hour	One hour	This indicates the percentage of storage space utilized by the tablespace at the collection time. In the case of container databases, this metric provides root container tablespaces.

Guest VM Health Metrics List - Non-Database Metrics

Table 4-3 Guest VM Health Metrics List - Non-Database Metrics

Metric Name	Metric Display Name	Unit	Aggregation	Collection Frequency	Description
`FilesystemUtilization`	Filesystem Utilization	Percentage	Max	One minute	Percent utilization of provisioned filesystem.
`CpuUtilization`	CPU Utilization	Percentage	Mean	One minute	Percent CPU utilization.
`MemoryUtilization`	Memory Utilization	Percentage	Mean	One minute	Percentage of memory available for starting new applications, without swapping. The available memory can be obtained via the following command: `cat /proc/meminfo`.
`SwapUtilization`	Swap Utilization	Percentage	Mean	One minute	Percent utilization of total swap space.
`LoadAverage`	Load Average	Number	Mean	One minute	System load average over 5 minutes.
`NodeStatus`	Node Status	Integer	Mean	One minute	Indicates whether the host is reachable.

Using the API to Manage Oracle Exadata Database Service on Exascale Infrastructure Instance

Use these API operations to manage Exadata Cloud Infrastructure virtual machines (VMs) and databases on Oracle Exadata Database Service on Exascale Infrastructure (ExaDB-XS).

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage Oracle Exadata Database Service on Exascale Infrastructure instance components.

Exascale Database Storage Vault resource

- ChangeExascaleDbStorageVaultCompartment

- CreateExascaleDbStorageVaul
- DeleteExascaleDbStorageVault
- GetExascaleDbStorageVault
- ListExascaleDbStorageVaults
- UpdateExascaleDbStorageVault

Exadata VM cluster

- ChangeExadbVmClusterCompartment
- CreateExadbVmCluster
- DeleteExadbVmCluster
- GetExadbVmCluster
- ListExadbVmClusters
- RemoveVirtualMachineFromExadbVmCluste
- UpdateExadbVmCluster

Manage Exascale Database Vaults on Exadata Database Service on Exascale Infrastructure

You can view, scale, and delete Exascale Database Storage Vaults on Oracle Exadata Database Service on Exascale Infrastructure (ExaDB-XS).

Viewing Exascale Database Storage Vaults

1. Navigate to Exascale Database Storage Vaults
2. Select **Exascale Database Storage Vaults**.
3. Select the Vault for which you want to view information.

Deleting Exascale Database Storage Vaults

1. Navigate to Exadata Database Service on Exascale Infrastructure.
2. Select **Exascale Database Storage Vaults**
3. Select the Vault that you want to delete.
4. Select **Delete**.
5. Confirm that you want to delete the Vault.

> **Note:**
>
> You can only delete Exascale Database Storage Vaults that no longer have any associated VM Clusters. If you still have associated VM Clusters on the Vault, then you must first delete those VM Clusters, and then return to these steps to delete your Exascale Database Storage Vault.

Scaling Exascale Database Storage Vaults

1. Navigate to Exadata Database Service on Exascale Infrastructure.

2. Select **Exascale Database Storage Vaults**.
3. Select the Vault that you want to scale.
4. Select **Scale Storage Vault**.
5. On the Scale Storage Vault dialog, enter a number for the capacity for High Capacity storage. This number should be the value for the total storage that you want to have provisioned after the scaling operation completes. In addition to the default flash cache included in the base storage capacity, you can choose to configure additional smart flash cache as a percentage ot provisioned storage capacity.
6. Click **Save Changes**. Your Vault will be scaled automatically.

Manage Oracle Database Software Images

This topic provides an overview of the database software image resource type, which you can use to create databases and Oracle Database Homes, and to patch databases.

Database software images give you the ability to create a customized Oracle Database software configuration that includes your chosen updates (PSU, RU or RUR), and optionally, a list of one-off (or interim) patches or an Oracle Home inventory file. This reduces the time required to provision and configure your databases, and makes it easy for your organization to create an approved "gold image" for developers and database administrators.

- Using Database Software Images in Oracle Cloud Infrastructure
- Using a Database Software Image with an Oracle Exadata Database Service on Exascale Infrastructure Instance
  Provision database homes, patch custom images, or set up Data Guard standby database using custom database images.
- Using the Console for database software images
- Using the API to manage database software images
  Use these API operations to manage database software images:

Using Database Software Images in Oracle Cloud Infrastructure

- Creation and Storage of Database Software Images
  Database software images are resources within your tenancy that you create prior to provisioning or patching a DB system,Oracle Exadata Database Service on Exascale Infrastructure instance, Database Home, or database.
- Using the OPatch lsinventory Command to Verify the Patches Applied to an Oracle Home
  OPatch utility enables you to apply the interim patches to Oracle Database software. You can find opatch utility in the `$ORACLE_HOME/Opatch` directory.

Creation and Storage of Database Software Images

Database software images are resources within your tenancy that you create prior to provisioning or patching a DB system,Oracle Exadata Database Service on Exascale Infrastructure instance, Database Home, or database.

There is no limit on the number of database software images you can create in your tenancy, and you can create your images with any Oracle Database software version and update supported in Oracle Cloud Infrastructure.

Database software images are automatically stored in Oracle-managed Object Storage and can be viewed and managed in the Oracle Cloud Infrastructure Console. Note that database software images incur Object Storage usage costs. Database software image are regional-level resources and can be accessed from any availability domain within their region.

See To create a database software image for information on creating an image.

Using the OPatch lsinventory Command to Verify the Patches Applied to an Oracle Home

OPatch utility enables you to apply the interim patches to Oracle Database software. You can find opatch utility in the `$ORACLE_HOME/Opatch` directory.

Using the `lsinventory` command provided by OPatch, you can create a file that lists the interim patches applied to an Oracle Database Home. This file can then be uploaded to the OCI Console during the creation of a custom Database Software Image to add the exact set of patches used by the source Database Home to the list of patches included in the software image. You can find OPatch utility in the `$ORACLE_HOME/Opatch` directory. The following example shows how to use the `lsinventory` command to create the lsinventory file:

1. Run the `opatch lsinventory` command to get the list of interim patches applied.

   ```
   $ORACLE_HOME/OPatch/opatch lsinventory
   Oracle Interim Patch Installer version 12.2.0.1.21
   Copyright (c) 2021, Oracle Corporation. All rights reserved.
   Oracle Home : /u02/app/oracle/product/19.0.0.0/dbhome_2
   Central Inventory : /u01/app/oraInventory
   from : /u02/app/oracle/product/19.0.0.0/dbhome_2/oraInst.loc
   OPatch version : 12.2.0.1.21
   OUI version : 12.2.0.7.0
   Log file location : /u02/app/oracle/product/19.0.0.0/dbhome_2/cfgtoollogs/
   opatch/opatch2021-01-21_09-22-45AM_1.log
   Lsinventory Output file location : /u02/app/oracle/product/19.0.0.0/
   dbhome_2/cfgtoollogs/opatch/lsinv/lsinventory2021-01-21_09-22-45AM.txt
   ```

2. Use the `lsinventory` output file to extract the additional One Off Patches applied to a specific Oracle Home.

Using a Database Software Image with an Oracle Exadata Database Service on Exascale Infrastructure Instance

Provision database homes, patch custom images, or set up Data Guard standby database using custom database images.

Provisioning: After you create a database software image, you can use it to create an Oracle Database Home in an Oracle Exadata Database Service on Exascale Infrastructure instance. For more information, see To create a new Database Home in an existing Exadata Cloud Service instance.

Patching: To patch a database in an Oracle Exadata Database Service on Exascale Infrastructure instance using a custom database software image, create the Database Home using the image, and then move the database to that Database Home. For more information, see Patching Individual Oracle Databases in an Exadata Cloud Service Instance.

Setting up Data Guard: When creating an Oracle Data Guard association, you can use a custom database software image to create a new Database Home for the new standby database. For more information, see To enable Oracle Data Guard on an Exadata Cloud Service instance database.

Using the Console for database software images

- To create a database software image
  Follow this procedure to create a database on Oracle Exadata Database Service on Exascale Infrastructure
- To create a database software image from a Database Home
- To view the patch information of a database software image
  Use this procedure to view the Oracle Database version, update information (PSU/BP/RU level) and included one-off (interim) patches of a database software image.
- To delete a database software image
  Use the following instructions to delete a database software image.

To create a database software image

Follow this procedure to create a database on Oracle Exadata Database Service on Exascale Infrastructure

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**.
2. Under **Resources**, click **Database Software Images**.
3. Click **Create Database Software Image**.
4. In the **Display name** field, provide a display name for your image. Avoid entering confidential information.
5. Choose your **Compartment**.
6. Choose the **Database version** for your image. You can create a database software image using any supported Oracle Database release update (RU).

   > **Note:**
   >
   > At the time of initial release of Exadata Database Service on Exascale Infrastructure, only Oracle Database 23ai is supported.

7. Choose the **patch set update, proactive bundle patch, or release update**. For information on Oracle Database patching models, see Release Update Introduction and FAQ (Doc ID 2285040.1)
8. Optionally, you can enter a comma-separated list of one-off (interim) patch numbers.
9. Optionally, you can upload an Oracle Home inventory file from an existing Oracle Database. See Using the OPatch lsinventory Command to Verify the Patches Applied to an Oracle Home for instructions on creating an inventory file using OPatch.
10. Click **Show Advanced Options** to add **tags** to your database software image. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
11. Click **Create Database Software Image**.

To create a database software image from a Database Home

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure** .
2. Choose your **Compartment**.
3. Navigate to the Database Home: Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.
4. Click **Database Homes** under **Resources**.
5. Find the Database Home you want to use to create the database software image in the list of Database Homes. Click the name of the Database Home to display details about it.
6. Click **Create Image from Database Home**.
7. In the **Create Database Software Image** panel, enter a **Display name** and select a compartment for the software image.
8. Click **Create**.

To view the patch information of a database software image

Use this procedure to view the Oracle Database version, update information (PSU/BP/RU level) and included one-off (interim) patches of a database software image.

Use the following instructions:

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**.
2. Under **Resources**, click **Database Software Images**.
3. In the list of database software images, find the image you want to view and click on the display name of the image.
4. On Database Software Image Details page for your selected image, details about the image are displayed:
   - The Oracle Database version is displayed in the **General Information** section. For example: 19.0.0.0
   - The **PSU/BP/RU** field of the **Patch Information** section displays update level for the image. For example: 19.5.0.0
   - The **One-Off Patches** field displays the number of one-off patches included in the image, if any. The count includes all patches specified when creating the image (including patches listed in lsinventory). To view the included patches (if any are included), click the **Copy All** link and paste the list of included patches into a text editor. The copied list of patch numbers is comma-separated and can be used to create additional database software images.

To delete a database software image

Use the following instructions to delete a database software image.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**.
2. Under **Resources**, click **Database Software Images**.

3. In the list of database software images, find the image you want to delete and click the Actions icon (three dots) at the end of the row.
4. Click **Delete**.

Using the API to manage database software images

Use these API operations to manage database software images:

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

- CreateDatabaseSoftwareImage
- ListDatabaseSoftwareImages
- GetDatabaseSoftwareImage
- DeleteDatabaseSoftwareImage
- ChangeDatabaseSoftwareImageCompartment

Create Oracle Database Homes on an Oracle Exadata Database Service on Exascale Infrastructure System

Learn to create Oracle Database Homes on Oracle Exadata Database Service on Exascale Infrastructure.

- About Creating Oracle Database Homes on an Oracle Exadata Database Service on Exascale Infrastructure System
  You can add Oracle Database homes (referred to as **Database Homes** in Oracle Cloud Infrastructure) to an existing VM cluster by using the Oracle Cloud Infrastructure Console, the API, or the CLI.
- To create a new Database Home in an existing Oracle Exadata Database Service on Exascale Infrastructure instance
  To create an Oracle Database home in an existing VM cluster with the Console, be prepared to provide values for the fields required.
- To create a database software image from a Database Home
- Using the API to Create Oracle Database Home on Oracle Exadata Database Service on Exascale Infrastructure
  To create an Oracle Database home, review the list of API calls.

About Creating Oracle Database Homes on an Oracle Exadata Database Service on Exascale Infrastructure System

You can add Oracle Database homes (referred to as **Database Homes** in Oracle Cloud Infrastructure) to an existing VM cluster by using the Oracle Cloud Infrastructure Console, the API, or the CLI.

A Database Home is a directory location on the Exadata database virtual machines that contains Oracle Database software binary files.

> **Note:**
>
> Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

You can also add and remove Database homes, and perform other management tasks on a Database home by using the `dbaascli` utility.

Related Topics

- Using the dbaascli Utility on Oracle Exadata Database Service on Exascale Infrastructure
  Learn to use the `dbaascli` utility on Oracle Exadata Database Service on Exascale Infrastructure.

To create a new Database Home in an existing Oracle Exadata Database Service on Exascale Infrastructure instance

To create an Oracle Database home in an existing VM cluster with the Console, be prepared to provide values for the fields required.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Navigate to the cloud VM cluster on which you want to create the new Database Home.

   Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.
4. Under **Resources**, click **Database Homes**.

   A list of Database Homes is displayed.
5. Click **Create Container Database**.
6. In the **Create Container Database** dialog, enter the following:
   - **Database Home display name:** The display name for the Database Home. Avoid entering confidential information.
     Database image: Determines what Oracle Database version is used for the database. You can have databases with different minor versions the same database home. The major versions must remain the same. By default, the latest Oracle-published database software image is selected.

     Click **Change Database Image** to use an older Oracle-published image or a custom database software image that you have created in advance, then select an **Image Type**:
     - **Oracle Provided Database Software Images:** These images contain generally available versions of Oracle Database software.
     - **Custom Database Software Images:** These images are created by your organization and contain customized configurations of software updates and patches. Use the **Select a compartment** and **Select a Database version** selectors to limit the list of custom database software images to a specific compartment or Oracle Database software major release version.

> **Note:**
>
> For the Oracle Database major version releases available in Oracle Cloud Infrastructure, images are provided for the current version plus the three most recent older versions (N through N - 3). For example, if an instance is using Oracle Database 19c, and the latest version of 19c offered is 19.8.0.0.0, images available for provisioning are for versions 19.8.0.0.0, 19.7.0.0, 19.6.0.0 and 19.5.0.0.

> **Note:**
>
> The custom database software image must be based on an Oracle Database release that meets the following criteria:
>
> - The release is currently supported by Oracle Cloud Infrastructure.
> - The release is supported by the hardware model on which you are creating the Database Home.

After choosing a software image, click **Select** to return to the Create Database dialog.

- Click **Show Advanced Options** to specify advanced options for the Database Home.
  - **Tags:** If you have permissions to create a resource, then you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure whether to apply tags, skip this option (you can apply tags later) or ask your administrator.

7. Click **Create**.

When the Database home creation is complete, the status changes from Provisioning to Available.

To create a database software image from a Database Home

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure** .
2. Choose your **Compartment**.
3. Navigate to the Database Home: Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.
4. Click **Database Homes** under **Resources**.
5. Find the Database Home you want to use to create the database software image in the list of Database Homes. Click the name of the Database Home to display details about it.
6. Click **Create Image from Database Home**.
7. In the **Create Database Software Image** panel, enter a **Display name** and select a compartment for the software image.
8. Click **Create**.

Using the API to Create Oracle Database Home on Oracle Exadata Database Service on Exascale Infrastructure

To create an Oracle Database home, review the list of API calls.

For information about using the API and signing requests, see "REST APIs" and "Security Credentials". For information about SDKs, see "Software Development Kits and Command Line Interface".

To create Database Homes in Oracle Exadata Database Service on Exascale Infrastructure, use the API operation `CreateDbHome`.

For the complete list of APIs, see "Database Service API".

Related Topics

- REST APIs
- Security Credentials
- Software Development Kits and Command Line Interface
- CreateDbHome
- Database Service API

Managing Oracle Database Homes on an Oracle Exadata Database Service on Exascale Infrastructure Instance

You can delete or view information about Oracle Database Homes (referred to as "Database Homes" in Oracle Cloud Infrastructure) by using the Oracle Cloud Infrastructure Console, the API, or the CLI.

- Manage Database Home Using the Console
  Use the OCI console to manage the various operations needed on a Database Home.
- Using the API to Manage Oracle Database Home on Oracle Exadata Database Service on Exascale Infrastructure
  Review the list of API calls to manage Oracle Database home.

Manage Database Home Using the Console

Use the OCI console to manage the various operations needed on a Database Home.

- To view information about a Database Home
  To view the details of a Database home, use this procedure.
- To delete a database home
  You cannot delete a Database Home that contains databases. You must first terminate the databases to empty the Database Home. See To terminate a database to learn how to terminate a database.
- To manage tags for your Database Home
  To add and modify metadata tags to help to manage your Database, use this procedure.
- Using the Console to Move a Database to Another Database Home
  Learn to move a database to another Database Home.

To view information about a Database Home

To view the details of a Database home, use this procedure.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Navigate to the cloud VM cluster or DB system containing the Database Home.

   Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster
4. On the VM Cluster Details page, under Resources, click **Database Homes**.
5. In the list of Database Homes, find the Database Home you are interested in, and then click its name to display details about it.

(Optional) Enter the result of the procedure here.

To delete a database home

You cannot delete a Database Home that contains databases. You must first terminate the databases to empty the Database Home. See To terminate a database to learn how to terminate a database.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Navigate to the cloud VM cluster or DB system containing the Database Home you want to delete:

   Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.
4. On the VM Cluster Details page, under Resources, click **Database Homes**.
5. In the list of Database Homes, find the Database Home you want to delete, and then click its name to display details about it.
6. On the Database Home Details page, click **Delete**.

   If the Database Home contains databases, you will not be able to proceed. You must cancel the deletion, empty the Database Home as applicable, and then retry the deletion.

To manage tags for your Database Home

To add and modify metadata tags to help to manage your Database, use this procedure.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Navigate to the cloud VM cluster or DB system containing the Database Home:

Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters.** In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.

4. Under **Resources**, click **Database Homes**.
5. In the list of Database Homes, find the Database Home you want to administer.
6. Click the the Actions icon (

   ⋮

) on the row listing the Database Home, and then click **Add Tags**.

Using the Console to Move a Database to Another Database Home

Learn to move a database to another Database Home.

1. Open the navigation menu. Under **Oracle Database**, click **Exadata Database Service on Exascale Infrastructure**.
2. Choose your **Compartment** that contains the VM cluster that hosts the database that you want to move.
3. Click **Exadata VM Clusters** in the left hand navigation.
4. Click the name of the VM cluster that contains the database that you want to move.
5. In the Resources list of the VM Cluster Details page, click **Databases**.
6. Click the name of the database that you want to move.

   The Database Details page displays information about the selected database.
7. Click **More Actions** and **Move To Another Home**.
8. In the resulting dialog, select the target Database Home.

   > **Note:**
   >
   > Oracle recommends using Database Homes, which are running the latest (N) to 3 versions from the latest (N-3) RU versions when updating the software version of the database by moving them to a target DB Home. Only DB Homes provisioned with database versions, which meet this best practice criterion are available as target homes to move your database.
9. Click **Move Database**.

The database will be stopped in the current home and then restarted in the destination home. While the database is being moved, the Database Home status displays as **Moving Database**. When the operation completes, Database Home is updated with the current home. If the operation is unsuccessful, the status of the database displays as **Failed**, and the Database Home field provides information about the reason for the failure.

Using the API to Manage Oracle Database Home on Oracle Exadata Database Service on Exascale Infrastructure

Review the list of API calls to manage Oracle Database home.

For information about using the API and signing requests, see "REST APIs" and "Security Credentials". For information about SDKs, see "Software Development Kits and Command Line Interface".

Use these API operations to manage Database Homes:

- `ListDbHomes`
- `GetDbHome`
- `DeleteDbHome`

For the complete list of APIs, see "Database Service API".

Related Topics

- REST APIs
- Security Credentials
- Software Development Kits and Command Line Interface
- ListDbHomes
- GetDbHome
- DeleteDbHome
- Database Service API

Manage Databases on Oracle Exadata Database Service on Exascale Infrastructure

- Prerequisites and Limitations for Creating and Managing Oracle Databases on Oracle Exadata Database Service on Exascale Infrastructure
  Review the prerequisites for creating and managing Oracle Databases on Oracle Exadata Database Service on Exascale Infrastructure.
- Oracle Database Releases Supported by Oracle Exadata Database Service on Exascale Infrastructure
  Learn about the versions of Oracle Database that Oracle Exadata Database Service on Exascale Infrastructure supports.
- Provisioning and Managing Exadata Databases
  This topic describes creating and managing Oracle Databases on an Oracle Exadata Database Service on Exascale Infrastructure instance instance.
- Using the API to manage Databases
- Create and Manage Exadata Pluggable Databases
  You can create and manage pluggable databases (PDBs) in Oracle Exadata Database Service on Exascale Infrastructure using the Console and APIs.
- Restoring an Exadata Pluggable Database
  You can perform in-place and out of place restore of an Exadata pluggable database.
- Changing the Database Passwords
  To change the SYS password, or to change the TDE wallet password, use this procedure.

Prerequisites and Limitations for Creating and Managing Oracle Databases on Oracle Exadata Database Service on Exascale Infrastructure

Review the prerequisites for creating and managing Oracle Databases on Oracle Exadata Database Service on Exascale Infrastructure.

Before you can create and use an Oracle Database on Oracle Exadata Database Service on Exascale Infrastructure, you must:

- Configure a VM cluster
- Create any required backup destinations

You can create one or more databases on each Oracle Exadata Database Service on Exascale Infrastructure system. Other than the storage and processing limits of your Oracle Exadata system, there is no maximum for the number of databases that you can create. By default, databases on Oracle Exadata Database Service on Exascale Infrastructure use Oracle Database Enterprise Edition - Extreme Performance. This edition provides all the features of Oracle Database Enterprise Edition, plus all of the database enterprise management packs, and all of the Enterprise Edition options, such as Oracle Database In-Memory, and Oracle Real Application Clusters (Oracle RAC). If you use your own Oracle Database licenses, then your ability to use various features is limited by your license holdings. TDE Encryption is required for all cloud databases. All new tablespaces will automatically be enabled for encryption.

Oracle Database Releases Supported by Oracle Exadata Database Service on Exascale Infrastructure

Learn about the versions of Oracle Database that Oracle Exadata Database Service on Exascale Infrastructure supports.

At the time of this release, Oracle Exadata Database Service on Exascale Infrastructure supports Oracle Database 23ai only.

For Oracle Database release and software support timelines, see *Release Schedule of Current Database Releases (Doc ID 742060.1)* in the My Oracle Support portal.

Related Topics

- https://support.oracle.com/epmos/faces/DocContentDisplay?id=742060.1

Provisioning and Managing Exadata Databases

This topic describes creating and managing Oracle Databases on an Oracle Exadata Database Service on Exascale Infrastructure instance instance.

In this documentation, "database" refers to a container database (CDB). When you provision a database in an Exadata cloud VM cluster, the database includes an initial pluggable database (PDB).

You can create Database homes, databases, and pluggable databases at any time by using the Console.

When you add a database to a VM cluster on an Exadata instance, the database versions you can select from depend on the current patch level of that resource. You may have to patch your VM cluster to add later database versions.

After you provision a database, you can move it to another Database home. Consolidating databases under the same home can facilitate management of these resources. All databases in a given Database Home share the Oracle Database binaries and therefore, have the same database version. The Oracle-recommended way to patch a database to a version that is different from the current version is to move the database to a home running the target version. For information about patching, see Patching an Exadata Cloud Service Instance.

When you create an Exadata database, you can choose to encrypt the database using your own encryption keys that you manage. You can rotate encryption keys, periodically, to maintain security compliance and, in cases of personnel changes, to disable access to a database.

> **Note:**
>
> - The encryption key you use must be AES-256.
> - To ensure that your Exadata database uses the most current versions of the Vault encryption key, rotate the key from the Database Details page on the Oracle Cloud Infrastructure Console. Do not use the Vault service's Console pages to rotate your Database keys.

If you want to use your own encryption keys to encrypt a database that you create, then you must create a dynamic group and assign specific policies to the group for customer-managed encryption keys. See Managing Dynamic Groups and Let security admins manage vaults, keys, and secrets. Additionally, see To integrate customer-managed key management into Exadata Cloud Service if you need to update customer-managed encryption libraries for the Vault service.

You can also add and remove databases, and perform other management tasks on a database by using command line utilities. For information and instructions on how to use these utilities, see Creating and Managing Exadata Databases Manually.

- Database Memory Initialization Parameters
- Customer-Managed Keys in Oracle Exadata Database Service on Exascale Infrastructure
  Customer-managed keys for Oracle Exadata Database Service on Exascale Infrastructure is a feature of Oracle Cloud Infrastructure (OCI) Vault service that enables you to encrypt your data using encryption keys that you control.
- Using the Console to Manage Databases on Oracle Exadata Database Service on Exascale Infrastructure
  To create or terminate a database, complete procedures using the Oracle Exadata console.

Database Memory Initialization Parameters

- When creating a container database, the initialization parameter, `SGA_TARGET` is set by the automation. This will automatically size the SGA memory pools. The setting will vary depending on the size of the database VM total memory. If the VM has less than or equal to 60 GB of system memory, `SGA_TARGET` is set to 3800 MB. If the VM has 60 GB or more system memory, `SGA_TARGET` is set to 7600 MB.
- The database initialization parameter `USE_LARGE_PAGES` is set to ONLY upon database creation, which will require the use of large pages for SGA memory. If the VM is configured with insufficient large pages, the instance will fail to start.

Customer-Managed Keys in Oracle Exadata Database Service on Exascale Infrastructure

Customer-managed keys for Oracle Exadata Database Service on Exascale Infrastructure is a feature of Oracle Cloud Infrastructure (OCI) Vault service that enables you to encrypt your data using encryption keys that you control.

The OCI Vault service provides you with centralized key management capabilities that are highly available and durable. This key-management solution also offers secure key storage using isolated partitions (and a lower-cost shared partition option) in FIPS 140-2 Level 3-certified hardware security modules, and integration with select Oracle Cloud Infrastructure services. Use customer-managed keys when you need security governance, regulatory compliance, and homogenous encryption of data, while centrally managing, storing, and monitoring the life cycle of the keys you use to protect your data.

You can do the following:

- Enable customer-managed keys when you create databases in Oracle Exadata Database Service on Exascale Infrastructure
- Switch from Oracle-managed keys to customer-managed keys
- Rotate your keys to maintain security compliance

Requirements

To enable management of customer-managed encryption keys, you must create a policy in the tenancy that allows a particular dynamic group to do so, similar to the following: `allow dynamic-group dynamic_group_name to manage keys in tenancy`.

Another policy is needed if the Vault being used by the customer is replicated. For vaults that are replicated, this policy is needed: `allow dynamic-group dynamic_group_name to read vaults in tenancy`

Limitations

To enable Oracle Data Guard on Oracle Exadata Database Service on Exascale Infrastructure databases that use customer-managed keys, the primary and standby databases must be in the same realm.

- To integrate customer-managed key management into Oracle Exadata Database Service on Exascale Infrastructure
  If you choose to encrypt databases in an Oracle Exadata Database Service on Exascale Infrastructure instance using encryption keys that you manage, then you may update the following two packages (using Red Hat Package Manager) to enable DBAASTOOLS to interact with the APIs that customer-managed key management uses.

Related Topics

- Replicating Vaults and Keys
- Learn About Oracle Cloud Basics

To integrate customer-managed key management into Oracle Exadata Database Service on Exascale Infrastructure

If you choose to encrypt databases in an Oracle Exadata Database Service on Exascale Infrastructure instance using encryption keys that you manage, then you may update the following two packages (using Red Hat Package Manager) to enable DBAASTOOLS to interact with the APIs that customer-managed key management uses.

KMS TDE CLI

To update the KMS TDE CLI package, you must complete the following task on all nodes in the Oracle Exadata Database Service on Exascale Infrastructure instance:

1. Deinstall current KMS TDE CLI package, as follows:

   ```
   rpm -ev kmstdecli
   ```

2. Install the updated KMS TDE CLI package, as follows:

   ```
   rpm -ivh kms_tde_cli
   ```

LIBKMS

LIBKMS is a library package necessary to synchronize a database with customer-managed key management through PKCS11. When a new version of LIBKMS is installed, any databases converted to customer-managed key management continue to use the previous LIBKMS version, until the database is stopped and restarted.

To update the LIBKMS package, you must complete the following task on all nodes in the Oracle Exadata Database Service on Exascale Infrastructure instance:

1. Confirm that the LIBKMS package is already installed, as follows:

   ```
   rpm -qa --last | grep libkmstdepkcs11
   ```

2. Install a new version of LIBKMS, as follows:

   ```
   rpm -ivh libkms
   ```

3. Use SQL*Plus to stop and restart all databases converted to customer-managed key management, as follows:

   ```
   shutdown immediate;
   startup;
   ```

4. Ensure that all converted databases are using the new LIBKMS version, as follows:

   ```
   for pid in $(ps aux | grep "<dbname>" | awk '{print $2;}'); do echo $pid;
   sudo lsof -p $pid | grep kms | grep "pkcs11_[0-9A-Za-z.]*" | sort -u; done
   | grep pkcs11
   ```

5. Deinstall LIBKMS packages that are no longer being used by any database, as follows:

   ```
   rpm -ev libkms
   ```

Using the Console to Manage Databases on Oracle Exadata Database Service on Exascale Infrastructure

To create or terminate a database, complete procedures using the Oracle Exadata console.

- To create a database in an existing Oracle Exadata Database Service on Exascale Infrastructure VM Cluster
  Learn how you can create your first or subsequent databases.
- Using the Console to Manage SYS User and TDE Wallet Passwords
  Learn to manage administrator (SYS user) and TDE wallet passwords.
- To view details of a Protected Database
  To view the details of a Protected Database, use this procedure.
- To create a database from a backup
  Learn how to use a backup to create a database on Exadata Database Service on Exascale Infrastructure.
- To create a database from the latest backup
  Use this procedure to create a database from the latest backup on Oracle Exadata Database Service on Exascale Infrastructure.
- To move a database to another Database Home
  To patch a single Oracle Database in your Oracle Exadata Database Service on Exascale Infrastructure instance, you move it to another Database Home.
- To terminate a database
  Use this procedure to terminate a database on Oracle Exadata Database Service on Exascale Infrastructure.
- To administer Vault encryption keys
  Use this procedure to rotate the Vault encryption key or or change the encryption management configuration.

To create a database in an existing Oracle Exadata Database Service on Exascale Infrastructure VM Cluster

Learn how you can create your first or subsequent databases.

> **Note:**
>
> If IORM is enabled on the Oracle Exadata Database Service on Exascale Infrastructure VM Cluster, then the default directive will apply to the new database and system performance might be impacted. Oracle recommends that you review the IORM settings and make applicable adjustments to the configuration after the new database is provisioned.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Navigate to the cloud VM cluster or DB system you want to create the database in:
   Cloud VM clusters (The New Exadata Cloud Infrastructure Resource Model): Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.
4. Click **Create Database**.
5. In the **Create Database** dialog, enter the following:

> **Note:**
>
> You cannot modify the `db_name`, `db_unique_name`, and SID prefix after creating the database.

- **Database name:** The name for the database. The database name must meet the requirements:
  - Maximum of 8 characters
  - Contain only alphanumeric characters
  - Begin with an alphabetic character
  - Cannot be part of the first 8 characters of a `DB_UNIQUE_NAME` on the VM cluster
  - DO NOT use the following reserved names: `grid`, `ASM`
- **Database unique name suffix:**
  Optionally, specify a value for the `DB_UNIQUE_NAME` database parameter. The value is case insensitive.

  The unique name must meet the requirements:
  - Maximum of 30 characters
  - Contain only alphanumeric or underscore (_) characters
  - Begin with an alphabetic character
  - Unique across the VM cluster. Recommended to be unique across the tenancy.

  If not specified, the system automatically generates a unique name value, as follows:

  ```
  <db_name>_<3_chars_unique_string>_<region-name>
  ```

- **Database version:** The version of the database. You can mix database versions on the Exadata DB system.
- **Database Home:** The Oracle Database Home for the database. Choose the applicable option:
  - **Select an existing Database Home:** The Database Home display name field allows you to choose the Database Home from the existing homes for the database version you specified. If no Database Home with that version exists, you must create a new one.
  - **Create a new Database Home**: Use this option to provision a new Database Home for your Data Guard peer database.
    Click **Change Database Image** to use an older Oracle-published image or a custom *database software image* that you have created in advance, then select an **Image Type**:
    - **Oracle Provided Database Software Images:**
      then you can use the **Display all available version** switch to choose from all available PSUs and RUs. The most recent release for each major version is indicated with a **latest** label.

> **Note:**
>
> For the Oracle Database major version releases available in Oracle Cloud Infrastructure, images are provided for the current version plus the three most recent older versions (N through N - 3). For example, if an instance is using Oracle Database 19c, and the latest version of 19c offered is 19.8.0.0.0, images available for provisioning are for versions 19.8.0.0.0, 19.7.0.0, 19.6.0.0 and 19.5.0.0.

    * **Custom Database Software Images:** These images are *created by your organization* and contain customized configurations of software updates and patches. Use the **Select a compartment** and **Select a Database version** selectors to limit the list of custom database software images to a specific compartment or Oracle Database software major release version.

- **PDB name:** *(Optional)* You can specify the name of the pluggable database. The PDB name must begin with an alphabetic character, and can contain a maximum of eight alphanumeric characters. The only special character permitted is the underscore (_). To avoid potential service name collisions when using Oracle Net Services to connect to the PDB, ensure that the PDB name is unique across the entire VM cluster. If you do not provide the name of the first PDB, then a system-generated name is used.
- **Create administrator credentials:** *(Read only)* A database administrator `SYS` user will be created with the password you supply.
  - **Username:** SYS
  - **Password:** Supply the password for this user. The password must meet the following criteria:
    A strong password for SYS, SYSTEM, TDE wallet, and PDB Admin. The password must be 9 to 30 characters and contain at least two uppercase, two lowercase, two numeric, and two special characters. The special characters must be _, #, or **-**. The password must not contain the username (SYS, SYSTEM, and so on) or the word "**oracle**" either in forward or reversed order and regardless of casing.
  - **Confirm password:** Re-enter the SYS password you specified.
  - Using a **TDE wallet password** is optional. If you are using customer-managed encryption keys stored in a vault in your tenancy, the TDE wallet password is not applicable to your DB system. Use **Show Advanced Options** at the end of the Create Database dialog to configure customer-managed keys.
    If you are using customer-managed keys, or if you want to specify a different TDE wallet password, uncheck the **Use the administrator password for the TDE wallet box**. If you are using customer-managed keys, leave the TDE password fields blank. To set the TDE wallet password manually, enter a password in the **Enter TDE wallet password** field, and then confirm by entering it into the **Confirm TDE wallet password** field.
- **Configure database backups:** Specify the settings for backing up the database to Autonomous Recovery Service or Object Storage:
  - **Enable automatic backup**: Check the check box to enable automatic incremental backups for this database. If you are creating a database in a security zone compartment, you must enable automatic backups.
  - **Backup Destination**: Your choices are **Autonomous Recovery Service** or **Object Storage**.
  - **Backup Scheduling**:

  - **Object Storage (L0)**:
    - **Full backup scheduling day**: Choose a day of the week for the initial and future L0 backups to start.
    - **Full backup scheduling time (UTC)**: Specify the time window when the full backups start when the automatic backup capability is selected.
    - **Take the first backup immediately**: A full backup is an operating system backup of all datafiles and the control file that constitute an Oracle Database. A full backup should also include the parameter file(s) associated with the database. You can take a full database backup when the database is shut down or while the database is open. You should not normally take a full backup after an instance failure or other unusual circumstances.

      If you choose to defer the first full backup your database may not be recoverable in the event of a database failure.
  - **Object Storage (L1)**:
    - **Incremental backup scheduling time (UTC)**: Specify the time window when the incremental backups start when the automatic backup capability is selected.
  - **Autonomous Recovery Service (L0)**:
    - **Scheduled day for initial backup**: Choose a day of the week for the initial backup.
    - **Scheduled time for initial backup (UTC)**: Select the time window for the initial backup.
    - **Take the first backup immediately**: A full backup is an operating system backup of all datafiles and the control file that constitute an Oracle Database. A full backup should also include the parameter file(s) associated with the database. You can take a full database backup when the database is shut down or while the database is open. You should not normally take a full backup after an instance failure or other unusual circumstances.

      If you choose to defer the first full backup your database may not be recoverable in the event of a database failure.
  - **Autonomous Recovery Service (L1)**:
    - **Scheduled time for daily backup (UTC)**: Specify the time window when the incremental backups start when the automatic backup capability is selected.

- **Deletion options after database termination**: Options that you can use to retain protected database backups after the database is terminated. These options can also help restore the database from backups in case of accidental or malicious damage to the database.
  - **Retain backups for the period specified in your protection policy or backup retention period**: Select this option if you want to retain database backups for the entire period defined in the Object Storage Backup retention period or Autonomous Recovery Service protection policy after the database is terminated.
  - **Retain backups for 72 hours, then delete**: Select this option to retain backups for a period of 72 hours after you terminate the database.

  - **Backup Retention Period/Protection Policy**: If you choose to enable automatic backups, you can choose a policy with one of the following preset retention periods, or a Custom policy.

    Object Storage Backup retention period: 7, 15, 30, 45, 60. Default: 30 days. The system automatically deletes your incremental backups at the end of your chosen retention period.

    Autonomous Recovery Service protection policy:

    * **Bronze:** 14 days
    * **Silver:** 35 days
    * **Gold:** 65 days
    * **Platinum:** 95 days
    * Custom defined by you
    * **Default:** Silver - 35 days
  - **Enable Real-Time Data Protection**: Real-time protection is the continuous transfer of redo changes from a protected database to **Autonomous Recovery Service**. This reduces data loss and provides a recovery point objective (RPO) near 0. This is an extra cost option.

6. Click **Show Advanced Options** to specify advanced options for the database:
   - **Management:**

     Oracle SID prefix: The Oracle Database instance number is automatically added to the SID prefix to create the `INSTANCE_NAME` database parameter. The `INSTANCE_NAME` parameter is also known as the `SID`. The `SID` is unique across the cloud VM Cluster. If not specified, `SID` prefix defaults to the `db_name`.

     > **Note:**
     >
     > Entering an `SID` prefix is only available for Oracle 12.1 databases and above.

     The `SID` prefix must meet the requirements:

     - Maximum of 12 characters
     - Contain only alphanumeric characters. You can, however, use underscore (_), which is the only special character that is not restricted by this naming convention.
     - Begin with an alphabetic character
     - Unique in the VM cluster
     - DO NOT use the following reserved names: `grid`, `ASM`
   - **Character set:** The character set for the database. The default is AL32UTF8.
   - **National character set:** The national character set for the database. The default is AL16UTF16.
   - **Encryption:**

     If you are creating a database in an Exadata Cloud Service VM Cluster, then you can choose to use encryption based on encryption keys that you manage. By default, the database is configured using Oracle-managed encryption keys. To configure the database with encryption based on encryption keys you manage:

a. Select **Use customer-managed keys**. You must have a valid encryption key in Oracle Cloud Infrastructure Vault service. See Let security admins manage vaults, keys, and secrets.

> **Note:**
>
> You must use AES-256 encryption keys for your database.

b. Choose a **Vault**.

c. Select a **Master encryption key**.

d. To specify a key version other than the latest version of the selected key, check **Choose the key version** and enter the OCID of the key you want to use in the **Key version OCID** field.

> **Note:**
>
> The Key version will only be assigned to the container database (CDB), and not to its pluggable database (PDB). PDB will be assigned an automatically generated new key version.

- **Tags**: If you have permissions to create a resource, then you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see *Resource Tags* . If you are not sure whether to apply tags, skip this option (you can apply tags later) or ask your administrator.

7. Click **Create Database**.

After database creation is complete, the status changes from **Provisioning** to **Available**, and on the database details page for the new database, the **Encryption** section displays the encryption key name and the encryption key OCID.

> **WARNING:**
>
> Do not delete the encryption key from the vault. This causes any database protected by the key to become unavailable.

Related Topics

- security zone compartment
- Resource Tags
- Let security admins manage vaults, keys, and secrets

Using the Console to Manage SYS User and TDE Wallet Passwords

Learn to manage administrator (SYS user) and TDE wallet passwords.

1. Open the navigation menu. Click **Oracle Database**, then click **Oracle Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment** that contains the VM cluster that hosts the database that you want to change passwords.

3. Click the name of the VM cluster that contains the database that you want to change passwords.
4. In the **Resources** list of the VM Cluster Details page, click **Databases**.
5. Click the name of the database that you want to change passwords.
   The Database Details page displays information about the selected database.
6. On the Database Details page, click More actions, and then click **Manage passwords.**
7. In the resulting **Manage passwords** dialog, click **Update Administrator Password** or **Update TDE Wallet Password**.
   Depending on the option you select, the system displays the fields to edit.
   - **Update Administrator Password**: Enter the new password in both the New administrator password and Confirm administrator password fields.

     > **Note:**
     >
     > The **Update Administrator Password** option will change the sys user password only. Passwords for other administrator accounts such as system, pdbadmin, and TDE wallet will not be changed.

   - **Update TDE Wallet Password**: Enter the current wallet password in the **Enter existing TDE wallet password** field, and then enter the new password in both the **New TDE wallet password** and **Confirm TDE wallet password** fields.
8. Click **Apply** to update your chosen password.

To view details of a Protected Database

To view the details of a Protected Database, use this procedure.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Navigate to the database:
   Under **Exadata at Oracle Cloud**, click **Exadata VM Clusters**.

   In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.

   On the cloud **VM cluster** details page, in the Databases table, click the name of the database to display the **Database Details** page.The Backup section displays the state of the automatic backups. If the Autonomous Recovery Service is the destination, a link will be available which includes additional details. You can also check if Real-time Data Protection is enabled or disabled. Click the **Autonomous Recovery Service** link to be taken to the page containing the Protected Database details.For more information about Protected Databases, see *Viewing Protected Database Details*.

Related Topics

- Viewing Protected Database Details

To create a database from a backup

Learn how to use a backup to create a database on Exadata Database Service on Exascale Infrastructure.

Before you begin, note the following:

- When you create a database from a backup, the availability domain is the same as the availability domain that hosts the backup or a different one within the same region.
- The Oracle Database software version you specify must be the same or later version as that of the backed-up database.
- If you are creating a database from an automatic backup, then you can choose any level 0 weekly backup, or a level 1 incremental backup created after the most recent level 0 backup. For more information on automatic backups, see Using the Console
- If the backup being used to create a database is in a security zone compartment, the database cannot be created in a compartment that is not in a security zone. See the Security Zone Policies topic for a full list of policies that affect Database service resources.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**.
2. Choose your **Compartment**.
3. Navigate to a backup.
   - *Standalone backups:* Click **Standalone Backups** under **Oracle Exadata Database Service on Exascale Infrastructure**.
   - *Automatic backups:* Navigate to the Database Details page of the database associated with the backup:
     - Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.

     Click the name of the database associated with the backup that you will use to create the new database. Locate the backup in the list of backups on the Database Details page.
4. Click the Actions menu icon (⋮) for the backup you chose.
5. Click **Create Database**. On the **Create Database from Backup** page, configure the database as follows.
6. In the **Provide basic information for the Exadata infrastructure** section:
   - **Select an availability domain:** It could be the same as the availability domain that hosts the backup or a different one within the same region
   - **Select Exadata infrastructure:** Select an Exadata infrastructure from the chosen compartment. Click the **Change Compartment** hyperlink to choose a different compartment.
7. In the **Configure your DB system** section:
   - Choose a cloud VM cluster to run the database from the **Select a VM cluster** drop-down list.
8. In the **Configure Database Home** section:
   - **Select an existing Database Home**: If you choose this option, make a selection from the **Select a Database Home** drop-down list.

> **Note:**
>
> You can not create a database from backup in the same Database home where the source database exists.

- **Create a new Database home**: If you choose this option, then enter a name for the new Database home in the **Database Home display name** field. Click **Change Database Image** to select a database software image for the new Database home. In the **Select a Database Software Image** panel, do the following:
    a. Select the compartment containing the database software image you want to use to create the new Database home.
    b. Select the Oracle Database software version that the new Database home will use, and then choose an image from the list of available images for your selected software version.
    c. Click **Select**.

9. In the **Configure database** section:

> **Note:**
>
> You cannot modify the `db_name`, `db_unique_name` , and SID prefix after creating the database.

- In the **Database name** field, name the database or accept the default name. The database name must meet the requirements:
    - Maximum of 8 characters
    - Contain only alphanumeric characters
    - Begin with an alphabetic character
    - Cannot be part of first 8 characters of a different database's `db_unique_name` on the VM cluster
    - Must not use the following reserved names: grid, ASM
- **Database unique name**: Specify a value for the `DB_UNIQUE_NAME` database parameter. The unique name must meet the requirements:
    - Maximum of 30 characters
    - Contain only alphanumeric or underscore (_) characters
    - Begin with an alphabetic character
    - Unique across the VM cluster. Recommended to be unique across the tenancy.

  If not specified, the system automatically generates a unique name value, as follows:

  ```
  <db_name>_<3_chars_unique_string>_<region-name>
  ```

- **Administrator username**: This read-only field displays the username for the administrator, "sys".
- In the **Password** and **Confirm password** fields, enter and re-enter a password. A strong password for SYS administrator must be 9 to 30 characters and contain at least two uppercase, two lowercase, two numeric, and two special characters. The

special characters must be _, #, or -. The password must not contain the user name (SYS, SYSTEM, and so on) or the word "oracle" either in forward or reverse order and regardless of casing.

10. In the **Enter the source database's TDE wallet or RMAN password** field, enter a password that matches either the Transparent Data Encryption (TDE) wallet password or RMAN password for the source database.
11. Click **Show Advanced Options** to specify advanced options for the database:
    - **Management**
      Oracle SID prefix: This option is in the **Management** tab. The Oracle Database instance number is automatically added to the SID prefix to create the `INSTANCE_NAME` database parameter. If not provided, then the SID prefix defaults to the first twelve characters of the `db_name`.

      The SID prefix must meet the requirements:

      - Maximum of 12 characters
      - Contain only alphanumeric characters
      - Begin with an alphabetic character
      - Unique in the VM cluster
      - Must not use the following reserved names: grid, ASM
12. Click **Create Database**.

NOT_SUPPORTED

1. Click the Exadata cloud VM cluster or DB system name that contains the specific database to display the details page.
2. From the list of databases, click the database name associated with the backup you want to use to display a list of backups on the database details page. You can also access the list of backups for a database by clicking **Backups** in the **Resources** section.

NOT_SUPPORTED

1. Click **Standalone Backups** under **Exadata Database Service on Exascale Infrastructure**.
2. In the list of standalone backups, find the backup you want to use to create the database.

To create a database from the latest backup

Use this procedure to create a database from the latest backup on Oracle Exadata Database Service on Exascale Infrastructure.

Before you begin, note the following:

- When you create a database from a backup, the availability domain is the same as the availability domain that hosts the backup or a different one within the same region.
- The Oracle Database software version you specify must be the same or later version as that of the backed-up database.
- If the backup being used to create a database is in a security zone compartment, the database cannot be created in a compartment that is not in a security zone. See the Security Zone Policies topic for a full list of policies that affect Database service resources.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**

2. Choose your **Compartment**.
3. Navigate to the cloud VM cluster that contains the source database you are using to create the new database:

   Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.
4. Under **Databases**, click the name of the database you are using as the source for the new database.
5. On the Database Details page, click **Create Database from Last Backup**.
6. In the **Provide basic information for the Exadata infrastructure** section:
   - **Select an availability domain:** It could be the same as the availability domain that hosts the backup or a different one within the same region.
   - **Select Exadata infrastructure:** Select an Exadata infrastructure from the chosen compartment. Click the **Change Compartment** hyperlink to choose a different compartment.
7. On the **Create Database from Backup** page, configure the database as follows.
8. In the **Configure your DB system** section: Choose a cloud VM cluster to run the database from the **Select a VM cluster** drop-down list.
9. In the **Configure Database Home** section:
   - **Select an existing Database Home**: If you choose this option, make a selection from the **Select a Database Home** drop-down list.
   - **Create a new Database home**: If you choose this option, enter a name for the new Database Home in the **Database Home display name** field. Click **Change Database Image** to select a database software image for the new Database Home. In the **Select a Database Software Image** panel, do the following:
     a. Select the compartment containing the database software image you want to use to create the new Database Home.
     b. Select the Oracle Database software version that the new Database Home will use, then choose an image from the list of available images for your selected software version.
     c. Click **Select**.
10. In the **Configure database** section:

    > **Note:**
    >
    > You cannot modify the `db_name`, `db_unique_name`, and SID prefix after creating the database.

    - **Database name:** The name for the database. The database name must meet the requirements:
      – Maximum of 8 characters
      – Contain only alphanumeric characters
      – Begin with an alphabetic character
      – Cannot be part of first 8 characters of a DB_UNIQUE_NAME on the VM cluster

    - DO NOT use the following reserved names: grid, ASM

  - **Database unique name:** Optionally, specify a value for the `DB_UNIQUE_NAME` database parameter. The value is case insensitive.

    The unique name must meet the requirements:

    - Maximum of 30 characters
    - Contain only alphanumeric or underscore (_) characters
    - Begin with an alphabetic character
    - Unique across the VM cluster. Recommended to be unique across the tenancy.

    If not specified, the system automatically generates a unique name value, as follows:

    ```
    <db_name>_<3_chars_unique_string>_<region-name>
    ```

  - **Administrator username**: This read-only field displays the username for the administrator, `sys`.
  - In the **Password** and **Confirm password** fields, enter and re-enter a password.
    A strong password for SYS administrator must be 9 to 30 characters and contain at least two uppercase, two lowercase, two numeric, and two special characters. The special characters must be _, #, or -. The password must not contain the user name (SYS, SYSTEM, and so on) or the word "oracle" either in forward or reverse order and regardless of casing.

11. In the **Enter the source database's TDE wallet or RMAN password** field, enter a password that matches either the Transparent Data Encryption (TDE) wallet password or RMAN password for the source database.
12. Click **Show Advanced Options** to specify advanced options for the database.
  - **Management**
    Oracle SID prefix: The Oracle Database instance number is automatically added to the SID prefix to create the INSTANCE_NAME database parameter. The INSTANCE_NAME parameter is also known as the SID. The SID is unique across the cloud VM cluster. If not specified, SID prefix defaults to the first 12 characters of the `db_name`. The SID prefix must meet the requirements:

    - Maximum of 12 characters
    - Contain only alphanumeric characters
    - Begin with an alphabetic character
    - Unique in the VM cluster
    - DO NOT use the following reserved names: grid, ASM

13. Click **Create Database**.

To move a database to another Database Home

To patch a single Oracle Database in your Oracle Exadata Database Service on Exascale Infrastructure instance, you move it to another Database Home.

You can move a database to any Database Home that meets at either of the following criteria:

- The target Database Home uses the same Oracle Database software version (including patch updates) as the source Database Home

- The target Database Home is based on either the latest version of the Oracle Database software release used by the database, or one of the three prior versions of the release

Moving a database to a new Database Home brings the database up to the patch level of the target Database Home. For information on patching Database Homes, see Database Home Patching.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Navigate to the database you want to move.
   Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, click the name of the VM cluster that contains the database you wan to move.
4. Click **More Actions**, then click **Move to Another Home**.
5. Select the target Database Home.
6. Click **Move Database**.
7. Confirm the move operation.

   The database is moved in a rolling fashion. The database instance will be stopped, node by node, in the current home and then restarted in the destination home. While the database is being moved, the Database Home status displays as **Moving Databse**. When the operation completes, Database Home is updated with the current home. Datapatch is run automatically, as part of the database move, to complete post-patch SQL actions for all patches, including one-offs, on the new Database Home. If the database move operation is unsuccessful, then the status of the database displays as `Failed`, and the Database Home field provides information about the reason for the failure.

To terminate a database

Use this procedure to terminate a database on Oracle Exadata Database Service on Exascale Infrastructure.

You'll get the chance to back up the database prior to terminating it. This creates a standalone backup that can be used to create a database later. We recommend that you create this final backup for any production (non-test) database.

> **Note:**
>
> Terminating a database removes all automatic incremental backups of the database from Oracle Cloud Infrastructure Object Storage. However, all full backups that were created on demand, including your final backup, will persist as standalone backups.

You cannot terminate a database that is assuming the primary role in a Data Guard association. To terminate it, you can switch it over to the standby role.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Navigate to the database:

Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.

On the cloud VM cluster details page, in the Databases table, click the name of the database to display the Database Details page.

4. Click **More Actions**, and then click **Terminate**.
   For the database using Oracle Cloud Infrastructure Object Storage or Oracle Database Autonomous Recovery Service: In the confirmation dialog,
   - Review the message about the backup retention policy.
   - Configure automatic backups as needed.
   - Type the name of the database to confirm the termination
5. Click **Terminate Database**.
   The database's status indicates Terminating.

> **Note:**
>
> The database stays in a terminated state with backups listed until all backups are expired.

To administer Vault encryption keys

Use this procedure to rotate the Vault encryption key or or change the encryption management configuration.

After you provision a database in an Exadata DB system or cloud VM cluster, you can rotate the Vault encryption key or change the encryption management configuration for that database.

> **Note:**
>
> - To ensure that your Exadata database uses the most current version of the Vault encryption key, rotate the key from the database details page on the Oracle Cloud Infrastructure Console. Do not use the Vault service.
> - You can rotate Vault encryption keys only on databases that are configured with customer-managed keys.
> - You can change encryption key management from Oracle-managed keys to customer-managed keys but you cannot change from customer-managed keys to Oracle-managed keys.
> - Oracle supports administering encryption keys on databases after Oracle Database 11*g* release 2 (11.2.0.4).

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your compartment from the **Compartment** drop-down.
3. Navigate to the cloud VM cluster that contains the database for which you want to change encryption management or to rotate a key.

Cloud VM clusters: Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, locate the VM cluster you want to access and click its highlighted name to view the details page for the cluster.

4. In the **Databases** section, click the name of the database for which you want to change encryption management or to rotate a key to display its details page.
5. Click the **More Actions** drop-down.
6. Click **Administer Encryption Key**.
   To rotate an encryption key on a database using customer-managed keys:
   a. Click **Rotate Encryption Key** to display a confirmation dialog.
   b. Click **Rotate Key**.

   To change key management type from Oracle-managed keys to customer-managed keys:
   a. Click **Change Key Management Type**.
   b. Select **Use customer-managed keys**.
      You must have a valid encryption key in Oracle Cloud Infrastructure Vault service and provide the information in the subsequent steps. See Key and Secret Management Concepts.
   c. Choose a vault from the **Vault in *compartment*** drop-down. You can change the compartment by clicking the **Change Compartment** link.
   d. Select an encryption key from the **Master encryption key in *compartment*** drop-down. You can change the compartment containing the encryption key you want to use by clicking the **Change Compartment** link.
   e. If you want to use an encryption key that you import into your vault, then select **Change Compartment** and enter the OCID of the key you want to use in the **Key version OCID** field.
7. Click **Apply**.

> **Note:**
>
> Changing key management causes the database to become briefly unavailable.

> **Caution:**
>
> After changing key management to customer-managed keys, do not delete the encryption key from the vault as this can cause the database to become unavailable.

On the database details page for this database, the **Encryption** section displays the encryption key name and the encryption key OCID.

Using the API to manage Databases

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage databases.

- ListDatabases
- GetDatabase
- CreateDatabase
- UpdateDatabase - Use this operation to move a database to another Database Home
- DeleteDatabase

For the complete list of APIs for the Database service, see Database Service API.

Create and Manage Exadata Pluggable Databases

You can create and manage pluggable databases (PDBs) in Oracle Exadata Database Service on Exascale Infrastructure using the Console and APIs.

In this documentation, "database" refers to a container database, also called a CDB. For more information on these resource types, see Multitenant Architecture in the Oracle Database documentation.

Databases created in Oracle Exadata Database Service on Exascale Infrastructure include an initial PDB that you can access from the Database Details page in the Console. You can create and manage additional PDBs in the database using the Console or APIs.

- **Backup**
  When the CDB is configured with the auto-backup feature, you have the option to take a backup of the PDB during create, clone, or relocate operations. The PDB backup destination will always be the same as the CDB, and the backups cannot be accessed directly or created on demand. Oracle recommends that you immediately back up the PDB after you create or clone them. This is because the PDB will not be recoverable until the next daily auto-backup completes successfully, leading to a possible data loss.
- **Restore**
  - **Oracle Exadata Database Service on Exascale Infrastructure**
    - **In place restore**: You can restore a PDB within the same CDB to last known good state or to a specified timestamp.
    - **Out of place restore**: You can restore a PDB by creating a database (CDB) from the backup, and then selecting a PDB or a subset of them that you want to restore on the new database.
- **Relocate**
  You can relocate a PDB from one CDB to another CDB within the same availability domain (AD):
  - Across compartments, VM clusters, DB system, or VCNs. If two different VCNs are used, then both VCNs must be peered before relocating.
  - To the same or a higher database version.

  During relocate, the PDB will be removed from the source CDB and moved to the destination CDB that is up and running. In an Oracle Data Guard association, a PDB relocated to the primary will be synchronized with the standby as well.
- **Clone**
  A clone is an independent and complete copy of the given database as it existed at the time of the cloning operation. You can create clones of your PDB within the same CDB or a different CDB and refresh the cloned PDB.

  The following types of clones are supported:

  - **Local clone**: A copy of the PDB is created within the same CDB.

  - **Remote clone**: A copy of the PDB is created in a different CDB.

  You can perform a remote clone of a PDB from one CDB to another CDB within the same availability domain (AD):

  - Across compartments, VM clusters, DB system, or VCNs. If two different VCNs are used, then both VCNs must be peered before cloning.
  - To the same or a higher database version.
  - **Refreshable clone**: A copy of the PDB is created in a different CDB, and you will be able to refresh the cloned PDB.
    You can perform a refreshable clone of a PDB from one CDB to another CDB within the same availability domain (AD):
    - Across compartments, VM clusters, DB system, or VCNs. If two different VCNs are used, then both VCNs must be peered before cloning.
    - To the same or a higher database version.

- **Refreshable Clone**
  A refreshable clone enables you to keep your remote clone updated with the source PDB. You can only refresh while the PDB is in mount mode. The only open mode you can have is read-only and refresh cannot be done while it is in read-only mode.
  - A database link user credential is required for creating a refreshable clone.
  - Clone, relocate, and in-place restore operations are not supported in the refreshable clone. Relocate and in-place restore operations are not supported in the source, and the source can only be deleted after disconnecting or deleting the refreshable clone.
  - In an Oracle Data Guard association, a refreshable clone cannot be created on standby, but it can be created on the primary. However, the primary will not be synced to the standby.

  > **Note:**
  >
  > A PDB in standby cannot be used as the source for a refreshable PDB.

- **Convert Refreshable PDB to Regular PDB**
  You can convert a refreshable PDB to a regular PDB by disconnecting the refreshable clone (destination PDB) from the source PDB at any time. If the refresh PDB is in a Data Guard association, when it is converted to a regular PDB the PDB will be synced to the standby as part of the conversion process.
- **Open Modes**
  On the Console, you can see the open modes of a PDB, such as read-write, read-only, and mounted. If the PDB status is the same across all nodes, then the system displays the same status for all PDBs. If the PDB statuses are different across the nodes, then the system displays a message indicating on which nodes the PDBs are opened in read-write mode. You cannot change the open mode of a PDB through the API or Console. However, you can start or stop a PDB. Starting the PDB will start it in read-write mode. Stopping the PDB will close it and it will remain in mount mode.
- Limitations for Pluggable Database Management
- Creating an Exadata Pluggable Database
- Managing an Exadata Pluggable Database
  This topic includes the procedures to connect to, start, stop, and delete a pluggable database (PDB).

- Cloning an Exadata Pluggable Database
  You can create local, remote, and refreshable clones.

Limitations for Pluggable Database Management

- New PDBs created with SQL are not immediately discovered by OCI's control plane and displayed in the Console. However, OCI does perform a sync operation on a regular basis to discover manually-created PDBs, and they should be visible in the Console and with API-based tools within 45 minutes of creation. Oracle recommends using the Console or API-based tools (including the OCI CLI , SDKs, and Terraform) to create PDBs.
- Pluggable database operations are supported only for databases using Oracle Database 19c and later.
- PDBs are backed up at the CDB level when using the OCI Console or APIs, and each backup includes all the PDBs in the database. However, the dbaascli utility's dbaascli database backup command allows you to create backups of specified PDBs. See Using the dbaascli Utility on Oracle Exadata Database Service on Exascale Infrastructure for more information.
- Restore operations are performed at the CDB level when using the OCI Console or APIs. However, the dbaascli utility's dbaascli pdb recover command allows you to restore backups of specified PDBs. See Using the dbaascli Utility on Oracle Exadata Database Service on Exascale Infrastructure for more information.

Creating an Exadata Pluggable Database

You can create a pluggable database (PDB) in Exadata Cloud Service from the OCI Console, or with the APIs and API-based tools (the OCI CLI, SDKs, and Terraform). PDBs must be created one at a time. During the PDB create operation, the parent database (CDB) is in the "Updating" state. Creating a new PDB has no impact on existing PDBs in the database.

- Using the console to create pluggable database
  To create the PDB, complete this procedure for Oracle Exadata Database Service on Exascale Infrastructure
- Using the console to relocate a pluggable database
  To relocate the PDB, complete this procedure for Oracle Exadata Database Service on Exascale Infrastructure
- Using the API to create pluggable database

Using the console to create pluggable database

To create the PDB, complete this procedure for Oracle Exadata Database Service on Exascale Infrastructure

> **Note:**
>
> Creating a pluggable database (PDB) is not supported for databases using Data Guard.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**.

2. Choose your **Compartment**.
3. Navigate to the database:

   Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.

   On the cloud VM cluster details page, in the **Databases** table, click the name of the database to display the Database Details page.
4. On the Database Details page, click **Pluggable Databases** in the **Resources** section of the page.
5. Click **Create Pluggable Database**.
6. In the **Create Pluggable Database** dialog, enter the following:
   - **PDB Name**: Enter a name for the PDB. The name must begin with an alphabetic character and can contain a maximum of 30 alphanumeric characters. Note: For bare metal DB systems, you cannot have two PDBs in the same database that use the same PDB name. You can use the same name for PDBs in different databases within the same DB system.
   - **Unlock my PDB Admin account**: *Optional.* Select this option to specify a PDB Admin password and configure the PDB to be unlocked at creation.
   - **PDB Admin password**: If you clicked **Unlock my PDB Admin** account, then create and enter a PDB admin password. The password must contain the following:
     - A minimum of 9 and a maximum of 30 characters
     - At least two uppercase characters
     - At least two lowercase characters
     - At least two special characters. The valid special characters are: underscore (_), a hash sign (#), and a dash (-). You can use two of the same characters or any combination of two of the same characters.
     - At least two numeric characters (0 - 9)
   - **Confirm PDB Admin password**: Reenter the PDB admin password.
   - **TDE wallet password**: *Applicable only to databases using Oracle-managed encryption keys*. Enter the TDE wallet password for the parent CDB.
   - **Take a backup of the PDB immediately after creating it:** You must enable auto-backup on the CDB to back up a PDB immediately after creating it. This check box is checked by default if auto-backup was enabled on the CDB.

     > **Note:**
     >
     > If the check box is unchecked, the system displays a warning stating that PDB cannot be recovered until the next daily backup has been successfully completed.
7. Click **Create Pluggable Database**.

WHAT NEXT?

After creating your PDB, you can get connection strings for the administrative service using the OCI Console.

Using the console to relocate a pluggable database

To relocate the PDB, complete this procedure for Oracle Exadata Database Service on Exascale Infrastructure

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**.
2. Choose your **Compartment**.
3. Navigate to the database:

   Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.

   On the cloud VM cluster details page, in the **Databases** table, click the name of the database to display the Database Details page.
4. On the Database Details page, click **Pluggable Databases** in the **Resources** section of the page.
5. Click the name of the PDB that you want to relocate.
   From the Pluggable Database details page, click **More Actions**, and then select **Relocate**.

   (or)

   Click the Actions menu (three dots) and select **Relocate**.
6. In the resulting Relocate Pluggable Database window, enter the following:
   - **VM Cluster:** Use the menu to select the destination VM cluster.
   - **Destination database:** Use the menu to select an existing database where the PDB will be created. This database can be of the same version as the CDB the source PDB is in or of a higher version.
   - **New PDB name for the clone:** The name must begin with an alphabetic character and can contain up to 30 characters. To keep the PDB name the same, just re-enter the source PDB name.
   - **Database TDE wallet password:** Enter the TDE wallet password for the parent CDB of the source PDB.
   - **Unlock my PDB Admin Account:**
     - To enter the administrator's password, check this check box.
       - **PDB Admin Password:** Enter PDB admin password. The password must contain the following:
         - a minimum of 9 and a maximum of 30 characters
         - at least two uppercase characters
         - at least two lowercase characters
         - at least two special characters. The valid special characters are underscore (_), a pound or hash sign (#), and dash (-). You can use two of the same characters or any combination of two of the same characters.
         - at least two numeric characters (0 - 9)
       - **Confirm PDB Admin Password:** Enter the same PDB Admin password in the confirmation field.

– To skip entering the administrator's password, uncheck this check box. If you uncheck this check box, then the PDB is created but you cannot use it. To use the PDB, you must reset the administrator password.

> **Note:**
>
> When you create a new PDB, a local user in the PDB is created as the administrator and granted the `PDB_DBA` role locally to the administrator.

To reset the password:

a. Connect to the container where your PDB exists using the SQL*Plus `CONNECT` statement.

```
SQL> show con_name;
CON_NAME
------------------------
CDB$ROOT
```

For more information, see *Administering a CDB* and *Administering PDBs* in *Oracle Multitenant Administrator’s Guide*.

b. Find the administrator name of your PDB:

```
SQL> select grantee from cdb_role_privs where con_id = (select
con_id from cdb_pdbs where pdb_name = '<PDB_NAME>') and
granted_role = 'PDB_DBA';
```

c. Switch into your PDB:

```
SQL> alter session set container=<PDB_NAME>;
Session altered.
SQL> show con_name;
CON_NAME
------------------------
<PDB_NAME>
```

d. Reset the PDB administrator password:

```
SQL> alter user <PDB_Admin> identified by <PASSWORD>;
User altered.
```

- **Source database SYS password**: Enter the database admin password.
- **Database link**: Enter the user name and password for the database link. Note that the user must be precreated in the source database. The DB link will be created in the destination using that username and password.
- **Take a backup of the PDB immediately after creating it**: You must enable auto-backup on the CDB to back up a PDB immediately after creating it. This check box is checked by default if auto-backup was enabled on the CDB.

> **Note:**
>
> If the checkbox is unchecked, then the system displays a warning stating that PDB cannot be recovered until the next daily backup has been successfully completed.

- **Advanced Options**
  Tags Optionally, you can apply tags. If you have permission to create a resource, you also have permission to apply free-form tags to that resource. To apply a defined tag, you must have permission to use the tag namespace. For more information about tagging, see *Resource Tags*. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

7. Click **Relocate pluggable database**.

> **Note:**
>
> Relocate will incur downtime during the process. The time required is based on the size of the PDB.

Using the API to create pluggable database

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the CreatePluggableDatabase API to create pluggable databases on Oracle Exadata Database Service on Exascale Infrastructure.

For the complete list of APIs for the Database service, see Database Service API.

Managing an Exadata Pluggable Database

This topic includes the procedures to connect to, start, stop, and delete a pluggable database (PDB).

It also includes instructions for getting PDB connection strings for the administrative service.

- To start a pluggable database
  To start the PDB, complete this procedure for Oracle Exadata Database Service on Exascale Infrastructure
- To stop a pluggable database
  To stop the PDB, complete this procedure for Oracle Exadata Database Service on Exascale Infrastructure.
- To delete a pluggable database
- To get connection strings for a pluggable database

To start a pluggable database

To start the PDB, complete this procedure for Oracle Exadata Database Service on Exascale Infrastructure

> **Note:**
>
> The PDB must be available and stopped to use this procedure.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**.
2. Choose your Compartment.
3. Navigate to the database.
   - Under **Oracle Exadata Database Service on Dedicated Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.

   On the Cloud VM Cluster details page, in the **Databases** table, click the name of the database to display the Database Details page.
4. Click **Pluggable Databases** in the **Resources** section of the page.
5. In the list of pluggable databases, find the pluggable database (PDB) you want to start. Click the PDB name to display details about it.
6. Click **Start**.
7. In the **Start PDB** dialog, click **Start PDB** to confirm the start operation.

To stop a pluggable database

To stop the PDB, complete this procedure for Oracle Exadata Database Service on Exascale Infrastructure.

> **Note:**
>
> The PDB must be available and running (started) to use this procedure.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**.
2. Choose your Compartment.
3. Navigate to the database.
   - Under **Oracle Exadata Database Service on Dedicated Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.

   On the Cloud VM Cluster details page, in the **Databases** table, click the name of the database to display the Database Details page.
4. Click **Pluggable Databases** in the **Resources** section of the page.
5. In the list of pluggable databases, find the pluggable database (PDB) you want to stop. Click the PDB name to display details about it.
6. Click **Start**.
7. In the **Stop PDB** dialog, click **Stop PDB** to confirm the stop operation.

To delete a pluggable database

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**.
2. Choose your Compartment.
3. Navigate to the database:

   *Cloud VM clusters (*new resource model*)* Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.

   DB systems Under **Bare Metal, VM, and Exadata**, click **DB Systems**. In the list of DB systems, find the Exadata DB system you want to access, and then click its name to display details about it.

   On the cloud VM cluster or DB system details page, in the **Databases** table, click the name of the database to display the Database Details page.
4. Click **Pluggable Databases** in the **Resources** section of the page.
5. In the list of pluggable databases, find the pluggable database (PDB) you want to delete. Click the PDB name to display details about it.
6. Click **More Actions**, then choose **Delete**.
7. In the **Delete PDB** dialog box, enter the name of the PDB that you want to delete to confirm the action, then click **Delete PDB**.

To get connection strings for a pluggable database

> **Note:**
>
> This topic explains how to get connection strings for the administrative service of a PDB. Oracle recommends that you connect applications to an application service, using strings created for the application service.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**.
2. Choose your Compartment.
3. Navigate to the database:

   *Cloud VM clusters (*new resource model*)* Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.

   DB systems Under **Bare Metal, VM, and Exadata**, click **DB Systems**. In the list of DB systems, find the Exadata DB system you want to access, and then click its name to display details about it.

   On the cloud VM cluster or DB system details page, in the **Databases** table, click the name of the database to display the Database Details page.
4. Click **Pluggable Databases** in the **Resources** section of the page.

5. In the list of pluggable databases, find the PDB, and then click its name to display details about it.
6. Click **PDB Connection**.
7. In the **Pluggable Database Connection** dialog, use the **Show** and **Copy** links to display and copy connection strings, as needed.
8. Click **Close** to exit the dialog.

Cloning an Exadata Pluggable Database

You can create local, remote, and refreshable clones.

A clone is an independent and complete copy of the given database as it existed at the time of the cloning operation. You can create clones of your PDB within the same CDB or a different CDB and also refresh the cloned PDB.

The following types of clones are supported:

- **Local clone:** A clone of the PDB is created within the same CDB.
- **Remote clone:** A clone of the PDB is created in a different CDB.
- **Refreshable clone:** A clone of the PDB is created in a different CDB, and you will be able to refresh the cloned PDB.

- Using the Console to Create a Local Clone of a Pluggable Database (PDB)
  Complete this procedure on Oracle Exadata Database Service on Exascale Infrastructure.
- Using the Console to Create a Remote Clone of a Pluggable Database (PDB)
  Complete this procedure on Oracle Exadata Database Service on Exascale Infrastructure.
- Using the Console to Create a Refreshable Clone of a Pluggable Database (PDB)
  Complete this procedure on Oracle Exadata Database Service on Exascale Infrastructure.
- Using the Console to Refresh a Cloned Pluggable Database (PDB)
  Complete this procedure on Oracle Exadata Database Service on Exascale Infrastructure.
- Using the Console to Convert a Refreshable Clone to a Regular Pluggable Database (PDB)
  Complete this procedure on Oracle Exadata Database Service on Exascale Infrastructure.
- Using the API to clone a pluggable database
  Learn how to manage pluggable databases (PDBs) using the pluggable database API endpoints.

Using the Console to Create a Local Clone of a Pluggable Database (PDB)

Complete this procedure on Oracle Exadata Database Service on Exascale Infrastructure.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**.
2. Choose your Compartment.
3. Navigate to the database:

   Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.

   On the cloud VM cluster details page, in the **Databases** table, click the name of the database to display the Database Details page.

4. Click **Pluggable Databases** in the **Resources** section of the page.
5. In the list of pluggable databases, find the pluggable database (PDB) you want to clone, and then click its name to display details about it.
6. Click **Clone**.
7. In the **Clone PDB** dialog box, enter the following:
   - **Select clone type**: Select **Local clone** to create a copy of the source PDB to the same CDB.
   - **Exadata VM Cluster**: Use the menu to select the cloud VM cluster of the target database.

     > **Note:**
     >
     > The target VM Cluster may be on a different Exadata infrastructure.

   - **Destination database**: This field is disabled.
   - **PDB name**: Provide a name for the new cloned PDB. The name must begin with an alphabetic character and can contain up to 30 characters.
   - **Database TDE wallet password**: *Not applicable for databases using customer-managed keys from the Vault service.* Enter the TDE wallet password for the parent database (CDB) of the source PDB.
   - **Unlock my PDB Admin account**: *Optional.* Select this option to specify a PDB Admin password and configure the PDB to be unlocked at creation.
   - **PDB Admin password**: Create and enter a new PDB Admin password. The password must contain the following:
     - 9–30 characters
     - At least two uppercase characters
     - At least two lowercase characters
     - At least two special characters. The valid special characters are: underscore (_), a hash sign (#), and a dash (-). You can use two of the same characters or any combination of two of these characters.
     - At least two numeric characters (0-9)
   - **Confirm PDB Admin password**: Enter the PDB Admin password again to confirm.
   - **Take a backup of the PDB immediately after creating it**: You must enable auto-backup on the CDB to back up a PDB immediately after creating it. This check box is checked by default if auto-backup was enabled on the CDB.

     > **Note:**
     >
     > If the checkbox is unchecked, the system displays a warning stating that PDB cannot be recovered until the next daily backup has been successfully completed.

   - *Optional.* **Enable thin clone**: Select this option to leverage Exascale redirect-on-write technology to create a thin clone of the PDB. This option results in the reuse of duplicate blocks with the parent PDB, shared with the clone. Deselecting this option results in a traditional, full clone with all blocks copied, and fully independent from the parent.

- **Advanced Options**
  Tags: Optionally, you can apply tags. If you have permission to create a resource, you also have permission to apply free-form tags to that resource. To apply a defined tag, you must have permission to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

8. Click **Clone pluggable database**.

Using the Console to Create a Remote Clone of a Pluggable Database (PDB)

Complete this procedure on Oracle Exadata Database Service on Exascale Infrastructure.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**.
2. Choose your Compartment.
3. Navigate to the database:

   Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.

   On the cloud VM cluster details page, in the **Databases** table, click the name of the database to display the Database Details page.
4. Click **Pluggable Databases** in the **Resources** section of the page.
5. In the list of pluggable databases, find the pluggable database (PDB) you want to clone, and then click its name to display details about it.
6. Click **Clone**.
7. In the **Clone PDB** dialog box, enter the following:
   - **Select clone type:** Select **Remote clone** to create a copy of the source PDB to the same CDB.
   - **Exadata VM Cluster**: Use the menu to select the cloud VM cluster of the target database.

     > **Note:**
     >
     > The target VM Cluster may be on a different Exadata infrastructure.

   - **Destination database**: Use the menu to select an existing database where the PDB will be created. This database can be of the same version as the CDB the source PDB is in or of a higher version.
   - **PDB name**: Provide a name for the new cloned PDB. The name must begin with an alphabetic character and can contain up to 30 characters.
   - **Database TDE wallet password**: *Not applicable for databases using customer-managed keys from the Vault service.* Enter the TDE wallet password for the parent database (CDB) of the source PDB.
   - **Unlock my PDB Admin account**: *Optional.* Select this option to specify a PDB Admin password and configure the PDB to be unlocked at creation.

- **PDB Admin password**: Create and enter a new PDB Admin password. The password must contain the following:
  - 9–30 characters
  - At least two uppercase characters
  - At least two lowercase characters
  - At least two special characters. The valid special characters are: underscore (_), a hash sign (#), and a dash (-). You can use two of the same characters or any combination of two of these characters.
  - At least two numeric characters (0-9)
- **Confirm PDB Admin password**: Enter the PDB Admin password again to confirm.
- **Database link**: Enter the user name and password for the database link. Note that the user must be precreated in the source database. The DB link will be created in the destination using that username and password.
- **Take a backup of the PDB immediately after creating it**: You must enable auto-backup on the CDB to back up a PDB immediately after creating it. This check box is checked by default if auto-backup was enabled on the CDB.

  > **Note:**
  >
  > If the checkbox is unchecked, the system displays a warning stating that PDB cannot be recovered until the next daily backup has been successfully completed.

- *Optional.* **Enable thin clone**: Select this option to leverage Exascale redirect-on-write technology to create a thin clone of the PDB. This option results in the reuse of duplicate blocks with the parent PDB, shared with the clone. Deselecting this option results in a traditional, full clone with all blocks copied, and fully independent from the parent.
- **Advanced Options:**
  - **Tags:** Optionally, you can apply tags. If you have permission to create a resource, you also have permission to apply free-form tags to that resource. To apply a defined tag, you must have permission to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

8. Click **Clone pluggable database**.

Using the Console to Create a Refreshable Clone of a Pluggable Database (PDB)

Complete this procedure on Oracle Exadata Database Service on Exascale Infrastructure.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**.
2. Choose your Compartment.
3. Navigate to the database:

   Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.

DB systems Under **Bare Metal, VM, and Exadata**, click **DB Systems**. In the list of DB systems, find the Exadata DB system you want to access, and then click its name to display details about it.

On the cloud VM cluster or DB system details page, in the **Databases** table, click the name of the database to display the Database Details page.

4. Click **Pluggable Databases** in the **Resources** section of the page.
5. In the list of pluggable databases, find the pluggable database (PDB) you want to clone, and then click its name to display details about it.
6. Click **Clone**.
7. In the **Clone PDB** dialog box, enter the following:
   - **Select clone type:** Select Refreshable clone to create a copy of the source PDB to the same CDB.
     For more information about refreshable clones, see About Refreshable Clone PDBs.
   - **Exadata VM Cluster**: Use the menu to select the cloud VM cluster of the target database.

     > **Note:**
     >
     > The target VM Cluster may be on a different Exadata infrastructure.

   - **Destination database**: Use the menu to select an existing database where the PDB will be created. This database can be of the same version as the CDB the source PDB is in or of a higher version.
   - **PDB name**: Provide a name for the new cloned PDB. The name must begin with an alphabetic character and can contain up to 30 characters.
   - **Database TDE wallet password**: *Not applicable for databases using customer-managed keys from the Vault service.* Enter the TDE wallet password for the parent database (CDB) of the source PDB.
   - **Unlock my PDB Admin account**: *Optional.* Select this option to specify a PDB Admin password and configure the PDB to be unlocked at creation.
   - **PDB Admin password**: Create and enter a new PDB Admin password. The password must contain:
     - 9–30 characters
     - At least two uppercase characters
     - At least two lowercase characters
     - At least two special characters. The valid special characters are: underscore (_), a hash sign (#), and a dash (-). You can use two of the same characters or any combination of two of these characters.
     - At least two numeric characters (0-9)
   - **Confirm PDB Admin password**: Enter the PDB Admin password again to confirm.
   - **Database link**: Enter the user name and password for the database link. Note that the user must be precreated in the source database. The DB link will be created in the destination using that username and password.

- **Take a backup of the PDB immediately after creating it**: You must enable auto-backup on the CDB to back up a PDB immediately after creating it. This check box is checked by default if auto-backup was enabled on the CDB.

  > **Note:**
  >
  > If the checkbox is unchecked, the system displays a warning stating that PDB cannot be recovered until the next daily backup has been successfully completed.

- *Optional.* **Enable thin clone**: Select this option to leverage Exascale redirect-on-write technology to create a thin clone of the PDB. This option results in the reuse of duplicate blocks with the parent PDB, shared with the clone. Deselecting this option results in a traditional, full clone with all blocks copied, and fully independent from the parent.
- **Advanced Options:**
  - **Tags:** Optionally, you can apply tags. If you have permission to create a resource, you also have permission to apply free-form tags to that resource. To apply a defined tag, you must have permission to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

8. Click **Clone pluggable database**.

Using the Console to Refresh a Cloned Pluggable Database (PDB)

Complete this procedure on Oracle Exadata Database Service on Exascale Infrastructure.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**.
2. Choose your Compartment.
3. Navigate to the database:

   Under **Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.

   On the cloud VM cluster details page, in the **Databases** table, click the name of the database to display the Database Details page.
4. Click **Pluggable Databases** in the **Resources** section of the page.
5. In the list of pluggable databases, find the pluggable database (PDB) you want to refresh, and then click its name to display details about it.
6. Click **More Actions** and select **Refresh**.
7. In the resulting **Refresh** dialog box, click **Refresh** to confirm.

Using the Console to Convert a Refreshable Clone to a Regular Pluggable Database (PDB)

Complete this procedure on Oracle Exadata Database Service on Exascale Infrastructure.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**.
2. Choose your Compartment.

3. Navigate to the database:

   Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.

   On the cloud VM cluster or DB system details page, in the **Databases** table, click the name of the database to display the Database Details page.

4. Click **Pluggable Databases** in the **Resources** section of the page.
5. In the list of pluggable databases, find the pluggable database (PDB) you want to convert to a regular PDB, and then click its name to display details about it.
6. In the resulting **Convert to regular PDB** dialog box, enter the following:
   - **Database TDE wallet password**: *Not applicable for databases using customer-managed keys from the Vault service.* Enter the TDE wallet password for the parent database (CDB) of the source PDB.
   - **Take a backup of the PDB immediately after creating it**: You must enable auto-backup on the CDB to back up a PDB immediately after creating it. This check box is checked by default if auto-backup was enabled on the CDB.

     > **Note:**
     >
     > If the checkbox is unchecked, then the system displays a warning stating that PDB cannot be recovered until the next daily backup has been successfully completed.

7. Click **Convert**.

Using the API to clone a pluggable database

Learn how to manage pluggable databases (PDBs) using the pluggable database API endpoints.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these APIs to clone pluggable databases:

- LocalclonePluggableDatabase
- RemoteclonePluggabledatabase

For the complete list of APIs for the Database service, see Database Service API.

Restoring an Exadata Pluggable Database

You can perform in-place and out of place restore of an Exadata pluggable database.

The following types of clones are supported:

- **In place restore**: You can restore a PDB within the same CDB to the last known good state, or to a specified timestamp.
- **Out of place restore**: You can restore a PDB by creating a database (CDB) from the backup, and then selecting a PDB or a subset of them you want to restore on the new database.

- Using the Console to Perform an In-Place Restore of a Pluggable Database (PDB)
  Complete this procedure for an in-place PDB restore using an RMAN backup on Exadata Database Service on Exascale Infrastructure
- Using the Console to Perform an Out-of-Place Restore of a Pluggable Database (PDB)
  Complete this procedure for an out-of-place PDB restore on Exadata Database Service on Exascale Infrastructure

Using the Console to Perform an In-Place Restore of a Pluggable Database (PDB)

Complete this procedure for an in-place PDB restore using an RMAN backup on Exadata Database Service on Exascale Infrastructure

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**.
2. Choose your Compartment.
3. Navigate to the database:

   Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.

   On the VM Cluster Details page, in the **Databases** table, click the name of the database to display the Database Details page.
4. Click **Pluggable Databases** in the **Resources** section of the page.
5. In the list of pluggable databases, find the pluggable database (PDB) that you want to restore, and then click its name to display details about it.
6. In the resulting Restore PDB dialog, enter the following:
   - **Restore to latest:** Select this option to restore and recover the database with zero, or least possible, data loss.
   - **Restore to a timestamp:** Select this option to restore and recover the database to the specified timestamp.
7. Click **Restore**.

Using the Console to Perform an Out-of-Place Restore of a Pluggable Database (PDB)

Complete this procedure for an out-of-place PDB restore on Exadata Database Service on Exascale Infrastructure

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**.
2. Choose your Compartment.
3. Navigate to the database:

   Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.

   On the VM Cluster Details page, in the **Databases** table, click the name of the database to display the Database Details page.
4. Click **Pluggable Databases** in the **Resources** section of the page.

5. In the list of pluggable databases, find the pluggable database (PDB) you want to restore, and then click its name to display details about it.
6. Under **Resources**, click **Backups**.
7. From the list of backups, choose a backup, click the Actions menu (three dots), and then select **Create Database**.
8. In the resulting Create database from backup dialog box, select either of these options, **Select all PDBs** or **Specify the PDBs to restore**.

Changing the Database Passwords

To change the SYS password, or to change the TDE wallet password, use this procedure.

The password that you specify in the **Database Admin Password** field when you create a new Oracle Exadata Database Service on Exascale Infrastructure instance or database is set as the password for the SYS, SYSTEM, TDE wallet, and PDB administrator credentials. Use the following procedures if you need to change passwords for an existing database.

> **Note:**
>
> if you are enabling Data Guard for a database, then the SYS password and the TDE wallet password of the primary and standby databases must all be the same.

> **Note:**
>
> Using the `dbaascli` to change the SYS password will ensure the backup/restore automation can parallelize channels across all nodes in the cluster.

To Change the SYS Password for an Oracle Exadata Database Service on Exascale Infrastructure Database

1. Log onto the Oracle Exadata Database Service on Exascale Infrastructure virtual machine as `opc`.
2. Run the following command:

   ```
   sudo dbaascli database changepassword --dbname database_name --user SYS
   ```

To Change Database Passwords in a Data Guard Environment

1. Run the following command on the primary database:

   ```
   dbaascli database changePassword —dbName <dbname> --user SYS --
   prepareStandbyBlob true --blobLocation <location to create the blob file>
   ```

2. Copy the blob file created to all the standby databases and update the file ownership to `oracle` user.

3. Run the following command on all the standby databases:

```
dbaascli database changePassword —dbName <dbname> --user SYS --
standbyBlobFromPrimary <location of copies the blob file>
```

To Change the TDE Wallet Password for an Oracle Exadata Database Service on Exascale Infrastructure Database

1. Log onto the Oracle Exadata Database Service on Exascale Infrastructure virtual machine as `opc`.
2. Run the following command:

```
sudo dbaascli tde changepassword --dbname database_name
```

Manage Database Backup and Recovery on Oracle Exadata Database Service on Exascale Infrastructure

Learn how to work with the backup and recovery facilities provided by Oracle Exadata Database Service on Exascale Infrastructure.

- Oracle Recommended Options to Perform Backup and Recovery Operations
  Oracle offers the following options for Oracle Database Backup and Recovery operations. These options are mutually exclusive.
- Managing Exadata Database Backups
  Automatic Exadata database backups are managed by Oracle Cloud Infrastructure. You configure this by using the Console or the API.
- Managed Backup Types and Usage Information
  There are two types of automatic Exadata database backups: Autonomous Recovery Service, and Oracle Object Storage.
- Default Backup Channel Allocation
  These are the default settings for database backup channels when using "Oracle Managed Backup" or "User Configured Backup".
- Prerequisites for Backups on Oracle Exadata Database Service on Exascale Infrastructure
- Using the Console to Manage Backups
- To designate Autonomous Recovery Service as a Backup Destination for an Existing Database
  To designate Autonomous Recovery Service as a Backup Destination for an existing database, use this procedure.
- Recovering an Exadata Database from Backup Destination
  This topic explains how to recover an Exadata database from a backup stored in either Object Storage or Autonomous Recovery Service by using the Console or the API.
- Managing Exadata Database Backups by Using bkup_api
- Using the API to Manage Backup and Recovery
- Alternative Backup Methods
  Learn about alternative backup methods that are available in addition to the OCI Console.

- Recovering a Database Using Oracle Recovery Manager (RMAN)
  If you backed up your database using `bkup_api`, then you can manually restore that database backup by using the Oracle Recovery Manager (RMAN) utility.

Oracle Recommended Options to Perform Backup and Recovery Operations

Oracle offers the following options for Oracle Database Backup and Recovery operations. These options are mutually exclusive.

> **Note:**
>
> A hybrid configuration, that is, mixing the options is not supported. Mixing the options will break automation.

Option 1: Oracle Managed Backups

Oracle managed backups are entirely managed by Exadata Cloud Infrastructure (ExaDB-D) or Exadata Cloud@Customer (ExaDB-C@C) based on a one-time configuration. Besides being fully integrated intoExaDB-D or ExaDB-C@C cloud services Control Plane, these backups can also be accessed through OCI APIs. Oracle recommends this approach.

- The `dbaascli database backup` and `dbaascli database recover` commands can be used in conjunction with the automated backups for certain operations. For more information, see `dbaascli database backup` and `dbaascli database recover`.
- Customers are allowed to query RMAN views or issue RMAN restore and recovery commands, for exampe, table, datafile, or tablespace recovery commands.

> **Note:**
>
> Do not use RMAN configuration to change any of the pre-tuned cloud RMAN settings.

Option 2: User Configured Backups

Customers can also configure backups from the host using the `dbaascli database backup` and `dbaascli database recover` commands. These backups, however, are not synchronized with the Control Plane nor are they integrated with the OCI APIs. Also, neither management nor lifecycle operations on these backups are supported from the service Control Plane console. Hence, this is not a recommended approach.

This approach is useful when direct access to Backup destinations is required to perform certain tasks. Accessing the OSS bucket, for example, to replicate backups across regions or monitor Backup Destinations.

For more information, see *User Configured Backup*.

Option 3: Backups using RMAN

Backups can be directly taken using RMAN with customer-owned customized scripts. Oracle, however, does not recommend this approach.

It is not recommended to use RMAN backups in conjunction with Oracle Managed Backups or User Configured Backups.

Who can use this option:

- Customers who want to maintain their existing RMAN backup/restore scripts.
- Customers who want to configure backups from Standby database in Data Guard environments to offload the backup workload to Standby.

ExaDB-D:

If you plan to backup using RMAN, then you must unregister the database from backup automation. For more information, see *Disabling Automatic Backups to Facilitate Manual Backup and Recovery Management*.

Related Topics

- dbaascli database backup
- dbaascli database recover
- Disabling Automatic Backups to Facilitate Manual Backup and Recovery Management
  Backups, configured in the Exadata Cloud Service console, API or `bkup_api` work for a variety of backup and recovery use cases.

Managing Exadata Database Backups

Automatic Exadata database backups are managed by Oracle Cloud Infrastructure. You configure this by using the Console or the API.

For unmanaged backups, see *Managing Exadata Database Backups by Using bkup_api*.

There are two destinations possible for automatic Exadata database backups: Autonomous Recovery Service, or Oracle Object Storage.

> **Note:**
>
> If you previously used `bkup_api` to configure backups and then you switch to using the Console or the API for backups:
>
> - A new backup configuration is created and associated with your database. This means that you can no longer rely on your previously configured unmanaged backups to protect your database.
> - `bkup_api` uses cron jobs to schedule backups. These jobs are not automatically removed when you switch to using managed backups.

Related Topics

- Managing Exadata Database Backups by Using bkup_api

Managed Backup Types and Usage Information

There are two types of automatic Exadata database backups: Autonomous Recovery Service, and Oracle Object Storage.

The database and infrastructure (the VM cluster or DB system) must be in an “Available” state for a backup operation to run successfully. Oracle recommends that you avoid performing actions that could interfere with availability (such as patching operations) while a backup operation is in progress. If an automatic backup operation fails, then the Database service retries the operation during the next day's backup window. If an on-demand full backup fails, then you can try the operation again when the Oracle Exadata Database Service on Exascale Infrastructure instance and database availability are restored.

When you enable the Automatic Backup feature, either service creates daily incremental backups of the database to the selected Backup Destination.

If you choose to enable automatic backups, then you can control the retention period. The system automatically deletes backups when the assigned retention period is expired.

Object Storage Backup retention period

The retention periods (in days) are 7, 15, 30, 45, 60. Default: 30 days.

The automatic backup process starts at any time during your daily backup window. You can optionally specify a 2-hour scheduling window for your database during which the automatic backup process will begin. There are 12 scheduling windows to choose from, each starting on an even-numbered hour (for example, one window runs from 4:00-6:00 AM, and the next from 6:00-8:00 AM). Backups jobs do not necessarily complete within the scheduling window.

The default backup window of 00:00 to 06:00 in the time zone of the Exadata Cloud Infrastructure instance's region is assigned to your database if you do not specify a window. Note that the default backup scheduling window is six hours long, while the backup windows you specify are two hours long.

Autonomous Recovery Service protection policy

- **Bronze** :14 days
- **Silver**: 35 days
- **Gold**: 65 days
- **Platinum**: 95 days
- Custom defined by you
- **Default**: Silver - 35 days

The automatic backup process starts at any time or within the assigned window.

> **Note:**
>
> - **Data Guard:** You can enable the Automatic Backup feature on a database with the standby role in a Data Guard association. However, automatic backups for that database will not be created until it assumes the primary role.
> - **Backup Retention Changes:** If you shorten your database's backup retention period or your protection policy in the future, existing backups falling outside the updated retention period are deleted by the system.
> - **Backup Storage Costs:** Automatic backups incur storage usage costs for either Autonomous Recovery Service or Object Storage depending on the backup destination selected.

You can create a full backup of your database at any time using either service.

When you terminate an Exadata Cloud Service instance database, all of its resources are deleted. Managed backups using the Object Storage destination will be deleted, and Managed backups using the Autonomous Recovery Service will be deleted according to the deletion option selected. Standalone backups created in Object Storage will remain after the database is terminated and must be manually deleted. You can use a standalone backup to create a new database.

To align with the Oracle recommended practice of using SYSBACKUP administrative privilege for Backup and Recovery operations, cloud automation creates a common administrative user C##DBLCMUSER with SYSBACKUP role at the CDB$ROOT container level. Backup and Recovery operations are therefore performed with the user having the least required privileges. Credentials for this user are randomly generated and securely managed by cloud automation. If the user is not found or is LOCKED and EXPIRED, then cloud automation will recreate or unlock this user during the backup or recovery operation. This change in the cloud automation was made starting with *dbaastools version 21.4.1.1.0*.

Default Backup Channel Allocation

These are the default settings for database backup channels when using "Oracle Managed Backup" or "User Configured Backup".

When a database is configured for backup using "Oracle Managed Backup" or "User Configured Backup", the tooling uses "default" for the backup channels. When default is used, dbaas will determine the number of channels to allocate at the time the backup or restore command is executed. The number of channels allocated is determined by the core count of the node. The following table provides the values used and the core range, both the core and the channel values are per node. Restore operations are prioritized. The cluster-wide total channel count is the per node value multiplied by the number of nodes. The automation uses the SCAN to distribute RMAN channels across all nodes in the cluster.

Cores Per Node	Formula	Backup Channels Allocation Per Node	Restore Channels Allocation Per Node
Less than or equal to 12	Cores <= 12	2	4
Greater than 12 and less than or equal to 24	Cores > 12 and Cores <= 24	4	8
Greater than 24	Cores > 24	8	16

If needed, a static per node value can be set by using the DBAASCLI `getConfig/configure` to generate a `bckup cfg` file, and setting the parameter `bkup_channels_node` to the number of channels per node desired.

Valid values are 1 - 32: The total channel count will be the value times the number of nodes. This value cannot exceed the limit of 255 channels. A value of `default` for `bkup_channels_node` sets core channel based allocation.

Prerequisites for Backups on Oracle Exadata Database Service on Exascale Infrastructure

Recovery Service

Ensure that your tenancy is configured to use Recovery Service.

Table 4-4 Review the prerequisite tasks before you use Recovery Service as the automatic backup destination

Task	More Information	Required or Optional
Create IAM policies	Policies to Enable Access to Recovery Service and Related Resources	Required
Configure network resources and register a Recovery Service subnet	Creating a Recovery Service Subnet in the Database VCN	Required
Create protection policies	Review Protection Policies for Database Backup Retention	Optional

For more information about Recovery Service, see Overview of Oracle Database Autonomous Recovery Service.

Object Storage

- Exadata Cloud Service requires access to the Oracle Cloud Infrastructure Object Storage. Oracle recommends using a service gateway with the VCN to enable this access. For more information, see Network Setup for Oracle Exadata Database Service on Exascale Infrastructure Instances. In that topic, pay particular attention to:
  - Service Gateway for the VCN
  - Node Access to Object Storage: Static Route
  - *Backup egress rule: Allows access to Object Storage*
  - Subnet Size Requirements and Security Rules for Recovery Service Subnet
- An existing Object Storage bucket to use as the backup destination. You can use the Console or the Object Storage API to create the bucket. For more information, see Managing Buckets.
- An auth token generated by Oracle Cloud Infrastructure. You can use the Console or the IAM API to generate the password. For more information, see Working with Auth Tokens.
- The user name specified in the backup configuration file must have tenancy-level access to Object Storage. An easy way to do this is to add the user name to the Administrators group. However, that allows access to all of the cloud services. Instead, an administrator should create a policy like the following that limits access to only the required resources in Object Storage for backing up and restoring the database:

  ```
  Allow group <group_name> to manage objects in compartment
  <compartment_name> where target.bucket.name = '<bucket_name>'
  Allow group <group_name> to read buckets in compartment <compartment_name>
  ```

  For more information about adding a user to a group, see Managing Groups. For more information about policies, see Getting Started with Policies.

Related Topics

- Auth Token

Using the Console to Manage Backups

You can use the Console to enable automatic incremental backups, create full backups on demand, and view the list of managed backups for a database. You can also use the Console to delete manual (on-demand) backups.

> **Note:**
>
> - The list of backups you see in the Console does not include any unmanaged backups (backups created directly by using `bkup_api`).
> - All backups are encrypted with the same master key used for Transparent Data Encryption (TDE) wallet encryption.
> - Backups for a particular database are listed on the details page for that database. The **Encryption Key** column displays either **Oracle-Managed Key** or a key name if you are using your own encryption keys to protect the database. See Backing Up Vaults and Keys for more information.

> **Note:**
>
> Do not delete any necessary encryption keys from the vault because this causes databases and backups protected by the key to become unavailable.

- To configure automatic backups for a database
- To create an on-demand backup of a database
- To view backup status
- To cancel a backup
- To delete full backups from Object Storage
- To delete standalone backups from Object Storage

To configure automatic backups for a database

When you create an Oracle Exadata Database Service on Exascale Infrastructure instance, you can optionally enable automatic backups for the initial database. Use this procedure to enable or disable automatic backups after the database is created.

> **Note:**
>
> Databases in a *security zone compartment* must have automatic backups enabled. See the *Security Zone Policies* topic for a full list of policies that affect Database service resources.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata on Oracle Public Cloud**.
2. Choose your **Compartment**.

3. Navigate to the cloud VM cluster or DB system containing the database you want to configure:
   Under **Oracle Exadata Database Service on Dedicated Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.
4. In the list of databases, find the database for which you want to enable or disable automatic backups, and click its name to display database details. The details indicate whether automatic backups are enabled.
5. Click **Configure Automatic Backups**.
6. In the Configure Automatic Backups dialog, enter the following details:
   - **Backup Destination**: Your choices are **Autonomous Recovery Service** (default) or **Object Storage**.
     - **Scenario 1**: The customer enables automatic backups AND has available limits AND there is available capacity in the region for Autonomous Recovery Service.

       Backup Destination: Your choices are Autonomous Recovery Service (default) or Object Storage. You can switch the backup destination from Autonomous Recovery Service to Object Storage.
     - **Scenario 2**: Customer enables automatic backups AND has exhausted the default limits for the Recovery Service AND there is available capacity in the region for Autonomous Recovery Service.

       Backup Destination: You can only use Object Storage. However, you can make an additional limits request and then use Autonomous Recovery Service.

       The system displays the following message with a link to request an increase to the limits.

       ```
       Tenancy has reached the limit for Autonomous Recovery Service. View
       your service limits and request an update.
       ```
     - **Scenario 3**: Customer enables automatic backups, and there is no available capacity in the region for Autonomous Recovery Service.

       Backup Destination: You can only use Object Storage. You can transition to Autonomous Recovery Service when there is sufficient capacity.

       The system displays the following message

       ```
       Autonomous Recovery Service has no available capacity in this region.
       Select Object Storage as your backup destination. You can transition
       from Object Storage to Autonomous Recovery Service when there is
       sufficient capacity.
       ```

       Proactively check if Autonomous Recovery Service capacity is available. If the required capacity becomes available and if you had chosen Object Storage, then you can transition to Autonomous Recovery Service.
   - **Backup Scheduling**:
     - **Object Storage (L0)**:
       - * **Full backup scheduling day**: Choose a day of the week for the initial and future L0 backups to start.
       - * **Full backup scheduling time (UTC)**: Specify the time window when the full backups start when the automatic backup capability is selected.
       - * **Take the first backup immediately**: A full backup is an operating system backup of all datafiles and the control file that constitute an Oracle Database.

A full backup should also include the parameter file(s) associated with the database. You can take a full database backup when the database is shut down or while the database is open. You should not normally take a full backup after an instance failure or other unusual circumstances.

If you choose to defer the first full backup your database may not be recoverable in the event of a database failure.

- **Object Storage (L1)**:
  - **Incremental backup scheduling time (UTC)**: Specify the time window when the incremental backups start when the automatic backup capability is selected.
- **Autonomous Recovery Service (L0)**:
  - **Scheduled day for initial backup**: Choose a day of the week for the initial backup.
  - **Scheduled time for initial backup (UTC)**: Select the time window for the initial backup.
  - **Take the first backup immediately**: A full backup is an operating system backup of all datafiles and the control file that constitute an Oracle Database. A full backup should also include the parameter file(s) associated with the database. You can take a full database backup when the database is shut down or while the database is open. You should not normally take a full backup after an instance failure or other unusual circumstances.
    If you choose to defer the first full backup your database may not be recoverable in the event of a database failure.
- **Autonomous Recovery Service (L1)**:
  - **Scheduled time for daily backup (UTC)**: Specify the time window when the incremental backups start when the automatic backup capability is selected.
- **Deletion options after database termination**: Options that you can use to retain protected database backups after the database is terminated. These options can also help restore the database from backups in case of accidental or malicious damage to the database.
  - **Retain backups for the period specified in your protection policy or backup retention period**: Select this option if you want to retain database backups for the entire period defined in the Object Storage Backup retention period or Autonomous Recovery Service protection policy after the database is terminated.
  - **Retain backups for 72 hours, then delete**: Select this option to retain backups for a period of 72 hours after you terminate the database.

- **Enable Real-Time Data Protection**: Real-time protection is the continuous transfer of redo changes from a protected database to **Autonomous Recovery Service**. This reduces data loss and provides a recovery point objective (RPO) near 0. This is an extra cost option.

7. Click **Save Changes**.
   The Database Details page displays the configuration details, **Health**, **Real-Time Data Protection**, and **Policy information** in the **Backup** section.

Related Topics

- security zone compartment
- Security Zone Policies

To create an on-demand backup of a database

> **Note:**
>
> Object Storage creates a full backup of the database while Recovery Service creates an incremental backup.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Navigate to the cloud VM cluster containing the database you want to back up:

   Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters.** In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.
4. In the list of databases, find the database for which you want to create an on-demand full backup and click its name to display database details.
5. Under **Resources**, click **Backups**.

   A list of backups is displayed.
6. Click **Create Backup**.

To view backup status

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**.
2. Choose your **Compartment**.
3. Navigate to the cloud VM cluster containing the database backup you want to view.
4. Click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.
5. In the list of databases, find the database you are interested in and click its name to display database details.
6. Under **Resources**, click **Backups**.
   A list of backups is displayed. The state column displays the status of the backup: **Active**, **Creating**, **Canceled**, **Canceling**, or **Failed**.

To cancel a backup

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata on Oracle Public Cloud**.
2. Choose your **Compartment**.
3. Navigate to the cloud VM cluster containing the database backup you want to view:
4. Click **Exadata VM Clusters**.
   In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.

5. In the list of databases, find the database you are interested in and click its name to display database details.
6. Under **Resources**, click **Backups**.
   A list of backups is displayed. The state column displays the status of the backup: **Active**, **Creating**, **Canceled**, **Canceling**, or **Failed**.
7. A backup in the Creating state may be canceled by clicking the Actions icon (three dots) on the right of the backup row and clicking **Cancel Backup**.
   A Cancel Backup confirmation dialog will appear.
8. Enter the name of the backup, and click **Cancel Backup**.
   The state changes to **Canceling**.

   The Cancel backup Work request can be viewed, by clicking **Work requests** under **Resources**.

If the Cancel backup fails:

- In the Work requests pane under Resources, you will see a line item called "**Cancel Database Backup**" with a state of "**Failed**". There will also be a work request for the backup "**Create Database Backup**" that will reflect the state of the Backup operation.

To delete full backups from Object Storage

> **Note:**
>
> You cannot explicitly delete automatic backups. Unless you terminate the database, automatic backups remain in Recovery Service and Object Storage for the number of days specified by the user, after which time they are automatically deleted.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**.
2. Choose your **Compartment**.
3. Navigate to the cloud VM cluster containing the database backup that you want to delete:

   Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.
4. In the list of databases, find the database you are interested in and click its name to display database details.
5. Under **Resources**, click **Backups**.

   A list of backups is displayed.
6. Click the Actions icon (

   ⋮

) for the backup in which you are interested, and then click **Delete**.
7. Confirm when prompted.

To delete standalone backups from Object Storage

1. Open the navigation menu. Click **Oracle Database**, then click **Standalone Backups** under **Resources**.
2. In the list of standalone backups, find the backup you want to use to delete.
3. Click the Actions menu for the backup you are interested in, and then click **Delete**.
4. In the **Delete** dialog, click **Delete** to confirm the backup deletion.

To designate Autonomous Recovery Service as a Backup Destination for an Existing Database

To designate Autonomous Recovery Service as a Backup Destination for an existing database, use this procedure.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata on Oracle Public Cloud**.
2. Choose your **Compartment**.
3. Navigate to the database:
   Cloud VM clusters (The New Exadata Cloud Infrastructure Resource Model): Under **Exadata on Oracle Public Cloud**, click **Exadata VM Clusters**.

   In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.

   DB systems: Under Oracle Base Database, click **DB Systems**.

   In the list of DB systems, find the Exadata DB system you want to access, and then click its name to display details about it.

   On the cloud **VM cluster** or **DB system** details page, in the Databases table, click the name of the database to display the **Database Details** page.
4. Click **Configure automatic backups**.
5. In the resulting window, provide the following details:
   - **Enable automatic backup**: Check the check box to enable automatic incremental backups for this database. If you are creating a database in a security zone compartment, you must enable automatic backups.
   - **Backup Destination**: Select **Autonomous Recovery Service**.
   - **Backup Scheduling**: If you enable automatic backups, you can choose a two-hour scheduling window to control when backup operations begin. If you do not specify a window, then a six-hour default window of 00:00 to 06:00 (in the time zone of the DB system's region) is used for your database.
   - **Protection Policy**: If you choose to enable automatic backups, you can choose a policy with one of the following preset retention periods, or a Custom policy.

     Object Storage Backup retention period: 7, 15, 30, 45, 60. Default: 30. The system automatically deletes your incremental backups at the end of your chosen retention period.

     Autonomous Recovery Service protection policy:
     - **Bronze:** 14 days

  - **Silver:** 35 days
  - **Gold:** 65 days
  - **Platinum:** 95 days
  - Custom defined by you
  - **Default:** Silver - 35 days
- **Enable Real-Time Data Protection**: Real-time protection is the continuous transfer of redo changes from a protected database to **Autonomous Recovery Service**. This reduces data loss and provides a recovery point objective (RPO) near 0. This is an extra cost option.

6. Click **Save Changes**.

Recovering an Exadata Database from Backup Destination

This topic explains how to recover an Exadata database from a backup stored in either Object Storage or Autonomous Recovery Service by using the Console or the API.

- Object Storage service is a secure, scalable, on-demand storage solution in Exadata Cloud Infrastructure.
- OracleDatabase Autonomous Recovery Service is a centralized, fully managed, and standalone backup solution for Oracle Cloud Infrastructure (OCI) databases.

For more information about backing up your databases to Object Storage, see *Managing Exadata Database Backups*.

- Using the Console to restore a database
  You can use the Console to restore the database from a backup in a backup destination that was created by using the Console.

Related Topics

- Managing Exadata Database Backups
  Automatic Exadata database backups are managed by Oracle Cloud Infrastructure. You configure this by using the Console or the API.

Using the Console to restore a database

You can use the Console to restore the database from a backup in a backup destination that was created by using the Console.

You can restore to:

- Restore to latest
- Restore to a timestamp
- Restore to SCN

> **Note:**
>
> The list of backups you see in the Console does not include any unmanaged backups (backups created directly by using `bkup_api`).

- To restore a database

To restore a database

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Navigate to the cloud VM cluster or DB system containing the database you want to restore:
   Cloud VM clusters (The New Exadata Cloud Infrastructure Resource Model): Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.

   DB systems: Under **Oracle Base Database**, click **DB Systems**. In the list of DB systems, find the Exadata DB system you want to access, and then click its name to display details about it.
4. In the list of databases, find the database you want to restore, and click its name to display details about it.
5. Click **Restore**.
6. Select one of the following options, and click **Restore Database**:
   - **Restore to the latest**: Restores the database to the last known good state with the least possible data loss.
   - **Restore to the timestamp**: Restores the database to the timestamp specified.
   - **Restore to System Change Number (SCN)**: Restores the database using the SCN specified. This SCN must be valid.

   > **Note:**
   >
   > You can determine the SCN number to use either by accessing and querying your database host, or by accessing any online or archived logs.
7. Confirm when prompted.
   If the restore operation fails, the database will be in a "**Restore Failed**" state. You can try restoring again using a different restore option. However, Oracle recommends that you review the `RMAN` logs on the host and fix any issues before reattempting to restore the database. These log files can be found in subdirectories of the `/var/opt/oracle/log` directory.

Managing Exadata Database Backups by Using bkup_api

You can use Exadata's backup utility, `bkup_api`, to back up databases on an Oracle Exadata Database Service on Exascale Infrastructure instance to an existing bucket in the Oracle Object Storage service and to the local disk Fast Recovery Area.

> **Note:**
>
> `bkup_api` is deprecated equivalent `dbaascli` commands are listed for each `bkup_api` command

For backups managed by Oracle Cloud Infrastructure, see Managing Exadata Database Backups.

This topic explains how to:

- Create a backup configuration file that indicates the backup destination, when the backup should run, and how long backups are retained. If the backup destination is Object Storage, the file also contains the credentials to access the service.
- Associate the backup configuration file with a database. The database will be backed up as scheduled, or you can create an on-demand backup.

> **Note:**
>
> You must update the cloud-specific tooling on all the compute nodes in your Oracle Exadata Database Service on Exascale Infrastructure instance before performing the following procedures. For more information, see Updating an Exadata Cloud Service Instance.

- Default Backup Configuration
  Description of the default backup configuration that follows Oracle best practice guidelines.
- To create a backup configuration file
- To create an on-demand backup
- To remove the backup configuration
- To delete a local backup
- To delete a backup in Object Storage

Default Backup Configuration

Description of the default backup configuration that follows Oracle best practice guidelines.

The backup configuration follows a set of Oracle best-practice guidelines:

- Full (level 0) backup of the database followed by rolling incremental (level 1) backups on a seven-day cycle (a 30-day cycle for the Object Storage destination).
- Full backup of selected system files.
- Automatic backups daily at a specific time set during the database deployment creation process.

Retention period:

- Both Object Storage and local storage: 30 days, with the 7 most recent days' backups available on local storage.
- Object Storage only: 30 days.
- Local storage only: Seven days.

Encryption:

- Both Object Storage and local storage: All backups to cloud storage are encrypted.
- Object Storage only: All backups to cloud storage are encrypted.

To create a backup configuration file

> **Note:**
>
> `bkup_api` is deprecated, use `dbaascli database backup` and its options instead

> **Note:**
>
> The following procedure must be performed on the first compute node in the Oracle Exadata Database Service on Exascale Infrastructure VM cluster or DB system resource. To determine the first compute node, connect to any compute node as the `grid` user and execute the following command:
>
> ```
> $ $ORACLE_HOME/bin/olsnodes -n
> ```
>
> The first node has the number 1 listed beside the node name.

1. SSH to the first compute node in the VM cluster or DB system resource.

   ```
   ssh -i <private_key_path> opc@<node_1_ip_address>
   ```

2. Log in as opc and then sudo to the root user.

   ```
   login as: opc

   [opc@dbsys ~]$ sudo su -
   ```

3. Use the `bkup_api get config` command to generate a file containing the current backup settings for the database deployment:

   ```
   # /var/opt/oracle/bkup_api/bkup_api get config [--file=<file_name>] --
   dbname=<db_name>
   ```

4. Use the following command to install the backup configuration, configure the credentials, schedule the backup, and associate the configuration with a database name.

   ```
   [root@dbsys bkup]# /var/opt/oracle/ocde/assistants/bkup/bkup -cfg bkup.cfg
   --dbname=<database_name>
   ```

   The backup is scheduled via cron and can be viewed at `/etc/crontab`.

When the scheduled backup runs, you can check its progress with the following command.

```
[root@dbsys bkup]# /var/opt/oracle/bkup_api/bkup_api bkup_statusThe backup
configuration file parameters are described in the following table:
```

Parameter	Description
bkup_disk=[yes|no]	Whether to back up locally to disk (Fast Recovery Area).
bkup_oss=[yes|no]	Whether to back up to Object Storage. If yes, you must also provide the parameters bkup_oss_url, bkup_oss_user, bkup_oss_passwd, and bkup_oss_recovery_window.
bkup_oss_url=<swift_url>	Required if bkup_oss=yes. The Object Storage URL including the tenant and bucket you want to use. The URL is: `https://swiftobjectstorage.<region_name>.oraclecloud.com/v1/<tenant>/<bucket>` where <tenant> is the *lowercase* tenant name (even if it contains uppercase characters) that you specify when signing in to the Console and <bucket> is the name of the existing bucket you want to use for backups.
bkup_oss_user=<oci_user_name>	Required if bkup_oss=yes. The user name for the Oracle Cloud Infrastructure user account. This is the user name you use to sign in to the Oracle Cloud Infrastructure Console. For example, `jsmith@example.com` for a local user or `<identity_provider>/jsmith@example.com` for a federated user. To determine which type of user you have, see the following topics: • Managing Users (for information on local users) • Federating with Identity Providers (for information on federated users) Note that the user must be a member of the Administrators group, as described in Prerequisites.
bkup_oss_passwd=<auth_token>	Required if bkup_oss=yes. The auth token generated by using the Console or IAM API, as described in Prerequisites. This is **not** the password for the Oracle Cloud Infrastructure user.
bkup_oss_recovery_window=n	Required if bkup_oss=yes. The number of days for which backups and archived redo logs are maintained in the Object Storage bucket. Specify 7 to 90 days.
bkup_daily_time=hh:mm	The time at which the daily backup is scheduled, specified in hours and minutes (hh:mm), in 24-hour format.
bkup_archlog_cron_entry=[yes|no]	When no backups are configured using dbaastools, setting `bkup_archlog_cron_entry=no` will remove the archive log cleanup job from crontab. The default value is “yes”.

Related Topics

- dbaascli database backup
  To configure Oracle Database with a backup storage destination, take database backups, query backups, and delete a backup, use the `dbaascli database backup` command.

To create an on-demand backup

> **Note:**
>
> `bkup_api` is deprecated. Use `dbaascli database backup` and its options instead.

You can use the `bkup_api` utility to create an on-demand backup of a database.

1. SSH to the first compute node in the Exadata VM cluster or DB system resource.

   ```
   ssh -i <private_key_path> opc@<node_1_ip_address>
   ```

   To determine the first compute node, connect to any compute node as the `grid` user and execute the following command:

   ```
   $ $ORACLE_HOME/bin/olsnodes -n
   ```

   The first node has the number 1 listed beside the node name.

2. Log in as opc and then sudo to the root user.

   ```
   login as: opc

   [opc@dbsys ~]$ sudo su -
   ```

3. You can let the backup follow the current retention policy, or you can create a long-term backup that persists until you delete it:

   - To create a backup that follows the current retention policy, enter the following command:

     ```
     # /var/opt/oracle/bkup_api/bkup_api bkup_start --dbname=<database_name>
     ```

   - To create a long-term backup, enter the following command:

     ```
     # /var/opt/oracle/bkup_api/bkup_api bkup_start --keep --
     dbname=<database_name>
     ```

4. Exit the root-user command shell and disconnect from the compute node:

   ```
   # exit
   $ exit
   ```

By default, the backup is given a timestamp-based tag. To specify a custom backup tag, add the `--tag` option to the `bkup_api` command; for example, to create a long-term backup with the tag "monthly", enter the following command:

```
# /var/opt/oracle/bkup_api/bkup_api bkup_start --keep --tag=monthly
```

After you enter a `bkup_api bkup_start` command, the `bkup_api` utility starts the backup process, which runs in the background. To check the progress of the backup process, enter the following command:

```
# /var/opt/oracle/bkup_api/bkup_api bkup_status --dbname=<database_name>
```

Related Topics

- dbaascli database backup
  To configure Oracle Database with a backup storage destination, take database backups, query backups, and delete a backup, use the `dbaascli database backup` command.

To remove the backup configuration

> **Note:**
>
> `bkup_api` is deprecated. Use `dbaascli database backup` with its option instead.

A backup configuration can contain the credentials to access the Object Storage bucket. For this reason, you might want to remove the file after successfully configuring the backup.

```
[root@dbsys bkup]# rm bkup.cfg
```

Related Topics

- dbaascli database backup
  To configure Oracle Database with a backup storage destination, take database backups, query backups, and delete a backup, use the `dbaascli database backup` command.

To delete a local backup

> **Note:**
>
> `bkup_api` is deprecated. Use `dbaascli database backup` and its option instead.

To delete a backup of a database deployment on the Oracle Exadata Database Service on Exascale Infrastructure instance, use the `bkup_api` utility.

1. Connect to the first compute node in your Exadata VM cluster or DB system resource as the `opc` user.

To determine the first compute node, connect to any compute node as the `grid` user and execute the following command:

```
$ $ORACLE_HOME/bin/olsnodes -n
```

The first node has the number 1 listed beside the node name.

2. Start a root-user command shell:

   ```
   $ sudo -s#
   ```

3. List the available backups:

   ```
   # >/var/opt/oracle/bkup_api/bkup_api recover_list --dbname=<database_name>
   ```

   where `dbname` is the database name for the database that you want to act on.

   A list of available backups is displayed.

4. Delete the backup you want:

   ```
   # /var/opt/oracle/bkup_api/bkup_api bkup_delete --bkup=<backup-tag> --
   dbname=<database_name>
   ```

   where `backup-tag` is the tag of the backup you want to delete.

5. Exit the root-user command shell:

   ```
   # exit
   $
   ```

Related Topics

- dbaascli database backup
  To configure Oracle Database with a backup storage destination, take database backups, query backups, and delete a backup, use the `dbaascli database backup` command.

To delete a backup in Object Storage

Use the RMAN `delete backup` command to delete a backup from the Object Store.

Using the API to Manage Backup and Recovery

- Using the API to manage backups
  Learn how to use the API for database backups on Oracle Exadata Database Service on Exascale Infrastructure.

Using the API to manage backups

Learn how to use the API for database backups on Oracle Exadata Database Service on Exascale Infrastructure.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage database backups:

- ListBackups
- GetBackup
- CreateBackup
- DeleteBackup
- UpdateDatabase - To enable and disable automatic backups.
- RestoreDatabase

For the complete list of APIs for the Database service, see Database Service API.

Alternative Backup Methods

Learn about alternative backup methods that are available in addition to the OCI Console.

Backup for databases on Oracle Exadata Database Service on Exascale Infrastructure can be accomplished through several methods in addition to the automatic backups configured in the console. Generally, the console (or the OCI API / CLI that correspond to it) is the preferred method as it provides the simplest and most automated method. In general, it is preferable to leverage the OCI Console, OCI API, or OCI Command-Line over alternative management methods. However, if required actions cannot be completed through the preferred methods, two other options are available to manually configure backups: `bkup_api` and Oracle Recovery Manager (RMAN).

> **Note:**
>
> `bkup_api` will be deprecated in a future release. Use the `dbaascli database backup`, `dbaascli pdb backup`, `dbaascli database recover`, and `dbaascli pdb recover` commands to backup and recover container databases and pluggable databases. For more information, see *User Configured Backup*.

RMAN is the backup tool included with the Oracle Database. For information about using RMAN, see the *Oracle Database Backup and Recovery User's Guide for Release 19*. Using RMAN to back up databases on Oracle Exadata Database Service on Exascale Infrastructure provides the most flexibility in terms of backup options, but also the most complexity.

> **Note:**
>
> While using RMAN for restoring databases backed up through any method described herein is considered safe, RMAN should NEVER be used to set up backups in conjunction with either console (and OCI API / CLI), nor in conjunction with `bkup_api`. If you choose to orchestrate backups manually leveraging RMAN, you should not use either console automated backups, nor should you use `bkup_api`. You must first completely disable console based automated backups. For more information, see *Disabling Automatic Backups to Facilitate Manual Backup and Recovery Management*.

The bkup_api method offers a middle ground between RMAN and console automated backups in terms of flexibility and simplicity. Use bkup_api if needed functionality is not supported with console automated backups, but when you wish to avoid complexity of using RMAN directly. In certain cases, bkup_api can be used to modify the console automated backup configuration, but this is not generally the case. Generally, bkup_api must be used instead of enabling backups in the console.

- Disabling Automatic Backups to Facilitate Manual Backup and Recovery Management
  Backups, configured in the Exadata Cloud Service console, API or bkup_api work for a variety of backup and recovery use cases.

Related Topics

- Managing Exadata Database Backups by Using bkup_api
- Disabling Automatic Backups to Facilitate Manual Backup and Recovery Management
  Backups, configured in the Exadata Cloud Service console, API or bkup_api work for a variety of backup and recovery use cases.

Disabling Automatic Backups to Facilitate Manual Backup and Recovery Management

Backups, configured in the Exadata Cloud Service console, API or bkup_api work for a variety of backup and recovery use cases.

Backups, configured in the Oracle Exadata Database Service on Exascale Infrastructure console, API or bkup_api work for a variety of backup and recovery use cases. If you require use cases not supported by the cloud-managed backups, then you can manage database backup and recovery manually, using the Oracle Recovery Manager (RMAN) utility. For information about using RMAN, see *Oracle Database Backup and Recovery User's Guide*.

Managing backup and recovery, using RMAN, on Oracle Exadata Database Service on Exascale Infrastructure requires taking full ownership of both database and archive log backups, and the cloud-managed backups should no longer be used. Before manual backups are started, the cloud-managed backup functionality should be disabled. This is needed so the cloud backup jobs do not purge archive logs before they are manually backed up and do not conflict with the manual backups.

You can use the bkup_api utility to disable cloud-managed backups, including disabling the automatic archive log purge job, by following this procedure:

> **Note:**
>
> If you execute these steps, then the automation will no longer purge/backup the archive logs in the FRA for the database.

1. Connect as the opc user to the first compute node.

   For detailed instructions, see *Connecting to a Compute Node with SSH*.
2. Start a root-user command shell:

   ```
   sudo -s
   ```

3. Use the `bkup_api` get config command to generate a file containing the current backup settings for the database deployment:

   ```
   /var/opt/oracle/bkup_api/bkup_api get config [--file=filename] --
   dbname=dbname
   ```

   Where:

   - *`filename`* is an optional parameter used to specify a name for the file that is generated
   - *`dbname`* is the database name for the database that you want to act on

4. Edit the parameter values in the generated file to change the following parameters.

   This will remove the backup crontab entries and disable all automatic backups. If the values are set to `yes`, then set to `no`.

   ```
   bkup_cron_entry=no
   bkup_archlog_cron_entry=no
   bkup_nfs=no
   bkup_oss=no
   bkup_local=no
   ```

5. Use the `bkup_api set config` command to update the backup settings using the file containing your updated backup settings:

   ```
   /var/opt/oracle/bkup_api/bkup_api set config --file=filename --
   dbname=dbname
   ```

   Where:

   - *`filename`* is an optional parameter used to specify a name for the file that is generated
   - *`dbname`* is the database name for the database that you want to act on

   The job to set the configuration will take several minutes to complete.

6. You can use the `bkup_api configure_status` command to check the status of the configuration update:

   ```
   /var/opt/oracle/bkup_api/bkup_api configure_status --dbname=dbname
   ```

   Where:

   - *`dbname`* is the database name for the database that you want to act on

   The **Configure backup status** starts as **running** and then moves to **finished** when complete.

7. Run the `bkup_api get config` command again and verify the settings listed above are set to `no`.

   ```
   /var/opt/oracle/bkup_api/bkup_api get config [--file=filename] --
   dbname=dbname
   ```

   Where:

   - *`filename`* is an optional parameter used to specify a name for the file that is generated

- *`dbname`* is the database name for the database that you want to act on

> **Note:**
>
> After making these changes, no backups, including archive log backups, are made by the cloud automation. Ensure that manual RMAN backups are in place to avoid filling the archive log location.

> **Note:**
>
> Changes made using the `bkup_api` command are not reflected in the Oracle Exadata Database Service on Exascale Infrastructure console.

8. Exit the root-user command shell:

```
exit
```

Related Topics

- Connecting to a Virtual Machine with SSH
  You can connect to the virtual machines in an Oracle Exadata Database Service on Exascale Infrastructure system by using a Secure Shell (SSH) connection.
- Oracle Database Backup and Recovery User's Guide for Release 19

Recovering a Database Using Oracle Recovery Manager (RMAN)

If you backed up your database using `bkup_api`, then you can manually restore that database backup by using the Oracle Recovery Manager (RMAN) utility.

If you backed up your database using `bkup_api`, then you can manually restore that database backup by using the Oracle Recovery Manager (RMAN) utility. For information about using RMAN, see the *Oracle Database Backup and Recovery User's Guide*.

> **Note:**
>
> While recovering using RMAN is safe, you must not use RMAN to initiate backups or edit backup setting in conjunction with either `backup_api` usage or in conjunction with automated console backups. Doing so could result in conflicting conditions or overwrites of settings, and backups may not run successfully.

Related Topics

- Oracle Database Backup and Recovery User's Guide for Release 19

Patch and Update an Oracle Exadata Database Service on Exascale Infrastructure System

- User-Managed Maintenance Updates
  Maintaining a secure Oracle Exadata Database Service on Exascale Infrastructure instance in the best working order requires you to perform regular maintanance.
- Patching and Updating an Oracle Exadata Database Service on Exascale Infrastructure System
  Learn how to perform patching operations on Exadata database virtual machines and Database Homes.

User-Managed Maintenance Updates

Maintaining a secure Oracle Exadata Database Service on Exascale Infrastructure instance in the best working order requires you to perform regular maintanance.

The following tasks are required

- Patching the Oracle Grid Infrastructure and Oracle Database software on the VM Cluster virtual machines. For information and instructions, see *Patching and Updating VM Cluster’s GI and Database Homes*.
- Updating the operating system on the VM Cluster virtual machines. See *Updating an Exadata Cloud VM Cluster Operating System* for information and instructions.

Patching and Updating an Oracle Exadata Database Service on Exascale Infrastructure System

Learn how to perform patching operations on Exadata database virtual machines and Database Homes.

For more guidance on achieving continuous service during patching operations, see the *Application Checklist for Continuous Service for MAA Solutions* white paper.

- Patching and Updating VM Cluster’s GI and Database Homes
  Learn how to perform patching operations on Oracle Exadata Database Service on Exascale Infrastructure resources by using the Console or API.
- Updating an Exadata Cloud VM Cluster Operating System
  Exadata VM cluster image updates allow you to update the OS image on your Exadata cloud VM cluster nodes in an automated manner from the OCI console and APIs.
- Upgrading Exadata Databases
  Oracle Database releases on Oracle Exadata Database Service on Exascale Infrastructure can be upgraded using the Console and the API.

Related Topics

- Application Checklist for Continuous Service for MAA Solutions

Patching and Updating VM Cluster’s GI and Database Homes

Learn how to perform patching operations on Oracle Exadata Database Service on Exascale Infrastructure resources by using the Console or API.

> **Note:**
>
> Oracle recommends patching databases by moving them to a Database Home that uses the target patching level. See To patch a database by moving it to another Database Home for instructions on this method of database patching.

- About Patching and Updating VM Cluster's GI and Database Homes
  Learn about types of patching performed on an Oracle Exadata Database Service on Exascale Infrastructure instances and how to complete the patching operations.
- Prerequisites for Patching and Updating an VM Cluster
  The Oracle Exadata Database Service on Exascale Infrastructure instance requires access to the Oracle Cloud Infrastructure Object Storage service, including connectivity to the applicable Swift endpoint for Object Storage
- Using the Console to Patch and Update Exadata Database Service on Exascale Infrastructure VM Clusters
  You can use the Console to view the history of patch operations on Oracle Exadata Database Service on Exascale InfrastructureOracle Exadata Database Service on Exascale Infrastructure VM clusters apply patches, and monitor the status of patch operations.
- Using the API to Patch an Oracle Exadata Database Service on Exascale Infrastructure Instance
  Use these API operations to manage patching the following Exadata resources: cloud VM clusters, databases, and Database Homes.

About Patching and Updating VM Cluster's GI and Database Homes

Learn about types of patching performed on an Oracle Exadata Database Service on Exascale Infrastructure instances and how to complete the patching operations.

- Oracle Grid Infrastructure (GI) Patching
  Patching an Oracle Exadata Database Service on Exascale Infrastructure instance updates the components on all the compute nodes in the instance. A VM cluster or DB system patch updates the Oracle Grid Infrastructure (GI) on the resource.
- Database Home Patching
  A Database Home patch updates the Oracle Database software shared by the databases in that home.
- To Upgrade the Oracle Grid Infrastructure of a Cloud VM Cluster
  Procedure for upgrading the Oracle Grid Infrastructure of a Cloud VM Cluster.
- Best Practices for Patching Oracle Exadata Database Service on Exascale Infrastructure Components

Oracle Grid Infrastructure (GI) Patching

Patching an Oracle Exadata Database Service on Exascale Infrastructure instance updates the components on all the compute nodes in the instance. A VM cluster or DB system patch updates the Oracle Grid Infrastructure (GI) on the resource.

> **Note:**
>
> You patch the Grid Infrastructure on the cloud VM cluster resource. VM clusters are used by the databases, which can be easily migrated to the new Grid Infrastructure resource with no system downtime.

Database Home Patching

A Database Home patch updates the Oracle Database software shared by the databases in that home.

Thus, you patch a database by either of the following methods:

- Move the database to a Database Home that has the correct patch version. This affects only the database being moved.
- Patching the Database Home the database is currently in. This affects all databases located in the Database Home being patched.

When patching a Database Home, you can use an Oracle-provided database software image to apply a generally-available Oracle Database software update, or you can use a custom database software image created by your organization to apply a specific set of patches required by your database. See Oracle Database Software Images for more information on creating and using custom images.

For instructions on performing patching operations, see To patch the Oracle Database software in a Database Home (cloud VM cluster). For Oracle Exadata Database Service on Exascale Infrastructure instances using the older DB system resource model, see To patch the Oracle Database software in a Database Home (DB system).

To Upgrade the Oracle Grid Infrastructure of a Cloud VM Cluster

Procedure for upgrading the Oracle Grid Infrastructure of a Cloud VM Cluster.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Click **Exadata VM Clusters**.
4. In the list of cloud VM clusters, click the name of the cluster you want to patch to display the cluster details.
5. Under **Version**, click the **View Patches** link beside the **Updates Available** field.
6. Click **Updates** to view the list of available patches and upgrades.
7. Click the Actions icon (three dots) at the end of the row listing the Oracle Grid Infrastructure (GI) upgrade, then click **Upgrade Grid Infrastructure**.
8. In the **Upgrade Grid Infrastructure** dialog, confirm you want to upgrade the GI by clicking **Upgrade Grid Infrastructure**. If you haven't run a precheck, you have the option of clicking **Run Precheck** in this dialog to precheck your cloud VM cluster prior to the upgrade.

Best Practices for Patching Oracle Exadata Database Service on Exascale Infrastructure Components

Consider the following best practices:

- Back up your databases before you apply any patches. For information about backing up the databases, see Managing Exadata Database Backups .
- Patch a VM cluster or an Exadata DB system before you patch the Databases Homes and databases on that resource.
- Before you apply any patch, run the precheck operation to ensure your VM cluster, Exadata DB system, or Database Home meets the requirements for that patch.
- To patch a database to a version other than the database version of the current home, move the database to a Database Home running the target version. This technique requires less downtime, and enables you to easily roll back the database to the previous version by moving it back to the old Database Home.
- For the Oracle Database and Oracle Grid Infrastructure major version releases available in Oracle Cloud Infrastructure, patches are provided for the current version, and the three most recent older versions (*N* through *N - 3*).

- dbaascli database runDatapatch
  To patch an Oracle Database, use the `dbaascli database runDatapatch` command.
- Customer-Managed Keys in Oracle Exadata Database Service on Exascale Infrastructure
  Customer-managed keys for Oracle Exadata Database Service on Exascale Infrastructure is a feature of Oracle Cloud Infrastructure (OCI) Vault service that enables you to encrypt your data using encryption keys that you control.
- dbaascli database addInstance
  To add the database instance on the specified node, use the `dbaascli database addInstance` command.
- dbaascli database convertToPDB
  To convert the specified non-CDB database to PDB, use the `dbaascli database convertToPDB` command.
- dbaascli database getDetails
  This command shows the detailed information of a given database e.g. dbname, node information, pluggable databases information etc.
- dbaascli database modifyParameters
  To modify or reset initialization parameters for an Oracle Database, use the `dbaascli database modifyParameters` command.
- dbaascli database upgrade
  To upgrade an Oracle Database, use the `dbaascli database upgrade` command.

dbaascli database runDatapatch

To patch an Oracle Database, use the `dbaascli database runDatapatch` command.

Prerequisites

- Before performing a `runDatapatch` operation, ensure that all of the database instances associated with the database are up and running.
- Run the command as the `root` user.

Syntax

```
dbaascli database runDatapatch --dbname
[--resume]
    [--sessionID]
[--skipPdbs | --pdbs]
[--executePrereqs]
```

```
[--patchList]
[--skipClosedPdbs]
[--rollback]
```

Where:

- `--dbname` specifies the name of the database
- `--resume` resumes the previous run
  - `--sessionID` specifies to resume a specific session ID
- `--skipPdbs` skips running the datapatch on a specified comma-delimited list of PDBs. For example: *pdb1*,*pdb2*...
- `--pdbs` runs the datapatch only on a specified comma-delimited list of PDBs. For example: *pdb1*,*pdb2*...
- `--executePrereqs` runs prerequisite checks
- `--patchList` applies or rolls back the specified comma-delimited list of patches. For example: *patch1*,*patch2*...
- `--skipClosedPdbs` skips running the datapatch on closed PDBs
- `--rollback` rolls back the patches applied

```
dbaascli database runDatapatch --dbname db19
```

Customer-Managed Keys in Oracle Exadata Database Service on Exascale Infrastructure

Customer-managed keys for Oracle Exadata Database Service on Exascale Infrastructure is a feature of Oracle Cloud Infrastructure (OCI) Vault service that enables you to encrypt your data using encryption keys that you control.

The OCI Vault service provides you with centralized key management capabilities that are highly available and durable. This key-management solution also offers secure key storage using isolated partitions (and a lower-cost shared partition option) in FIPS 140-2 Level 3-certified hardware security modules, and integration with select Oracle Cloud Infrastructure services. Use customer-managed keys when you need security governance, regulatory compliance, and homogenous encryption of data, while centrally managing, storing, and monitoring the life cycle of the keys you use to protect your data.

You can do the following:

- Enable customer-managed keys when you create databases in Oracle Exadata Database Service on Exascale Infrastructure
- Switch from Oracle-managed keys to customer-managed keys
- Rotate your keys to maintain security compliance

Requirements

To enable management of customer-managed encryption keys, you must create a policy in the tenancy that allows a particular dynamic group to do so, similar to the following: `allow dynamic-group dynamic_group_name to manage keys in tenancy`.

Another policy is needed if the Vault being used by the customer is replicated (https://docs.oracle.com/en-us/iaas/Content/KeyManagement/Tasks/replicatingvaults.htm). For vaults that are replicated, this policy is needed: `allow dynamic-group dynamic_group_name to read vaults in tenancy`

Limitations

To enable Oracle Data Guard on Oracle Exadata Database Service on Exascale Infrastructure databases that use customer-managed keys, the primary and standby databases must be in the same realm .

Related Topics

- To create a database in an existing Exadata Cloud Service instance
- To administer Vault encryption keys

dbaascli database addInstance

To add the database instance on the specified node, use the `dbaascli database addInstance` command.

Prerequisite

- Run the command as the `root` user.

Syntax

```
dbaascli database addInstance --dbname <value> --node <value> [--newNodeSID
<value>]
```

Where:

- `--dbname` specifies Oracle Database name
- `--node` specifies the node name for the database instance
  - `--newNodeSID` specifies SID for the instance to add in the new node

dbaascli database convertToPDB

To convert the specified non-CDB database to PDB, use the `dbaascli database convertToPDB` command.

Syntax

```
dbaascli database convertToPDB --dbname <value> [--cdbName <value>] [--
executePrereqs]
        {
            [--copyDatafiles [--keepSourceDB]]|[backupPrepared]
        }
        [--targetPDBName <value>] [--waitForCompletion <value>] [--resume [--
sessionID <value>]]
```

Where:

- `--dbname` specifies the name of Oracle Database
- `--cdbName` specifies the name of the target CDB in which the PDB will be created. If the CDB does not exist, then it will be created in the same Oracle home as the source non-CDB
- `--executePrereqs` specifies to run only the pre-conversion checks
- `--copyDatafiles` specifies to create a new copy of the data files instead of using the ones from the source database

`--keepSourceDB` - to preserve the source database after completing the operation.

- `--backupPrepared` - flag to acknowledge that a proper database backup is in place for the non CDB prior to performing the conversion to PDB.
- `--backupPrepared` flag to acknowledge that a proper database backup is in place for the non-CDB prior to performing the conversion to PDB
- `--targetPDBName` specifies the name of the PDB that will be created as part of the operation
- `--waitForCompletion` specifies `false` to run the operation in the background. Valid values: `true|false`
- `--resume` specifies to resume the previous execution
  - `--sessionID` specifies to resume a specific session ID

Example 4-2 dbaascli database convertToPDB

To run pre-conversion prechecks:

```
dbaascli database convertToPDB --dbname ndb19 --cdbname cdb19 --
backupPrepared --executePrereqs
```

To run a full conversion with a copy of the data files from the non-CDB:

```
dbaascli database convertToPDB --dbname tst19 --cdbname cdb19 --copyDatafiles
```

dbaascli database getDetails

This command shows the detailed information of a given database e.g. dbname, node information, pluggable databases information etc.

Prerequisites

Run the command as the `root` user or the `oracle` user

Syntax

```
dbaascli database getDetails --dbname <value>
```

Where :

- `--dbname` - Oracle database name.

dbaascli database modifyParameters

To modify or reset initialization parameters for an Oracle Database, use the `dbaascli database modifyParameters` command.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli database modifyParameters --dbname <value>
{
--setParameters <values>[--instance <value>] [--backupPrepared] [--
allowBounce]|
--resetParameters <values> [--instance <value>] [--backupPrepared] [--
allowBounce]
}
--responseFile
[--backupPrepared]
[--instance]
[--allowBounce]
[--waitForCompletion]
```

Where:

- `--dbname` specifies the name of the database.
- `--setParameters` specifies a comma-delimited list of parameters to modify with new values. For example: `parameter1=`*`valueA`*`,parameter2=`*`valueB`*, and so on. For blank values use parameter1=valueA,parameter2='',etc.
- `--resetParameters` specifies a comma-delimited list of parameters to be reset to their corresponding default values. For example, `parameter1`,`parameter2`, and so on.
- `--instance` specifies the name of the instance on which the parameters will be processed. If not specified, then the operation will be performed at the database level.
- `--backupPrepared` acknowledges that a proper database backup is in place prior to modifying critical or sensitive parameters.
- `--allowBounce` grants permission to bounce the database in order to reflect the changes on applicable static parameters.
- `--waitForCompletion` specify false to run the operation in background. Valid values : true|false.]

Example 4-3 dbaascli database modifyParameters

```
dbaascli database modifyParameters --dbname dbname --setParameters
"log_archive_dest_state_17=ENABLE"
```

dbaascli database upgrade

To upgrade an Oracle Database, use the `dbaascli database upgrade` command.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli database upgrade --dbname <value>
{--targetHome <value> | --targetHomeName <value>}
{ [--executePrereqs | --postUpgrade | --rollback]}
{[--standBy | --allStandbyPrepared]}
{[--upgradeOptions <value>]  | [--standBy]}
[--removeGRP]
[--increaseCompatibleParameter]
[--resume [--sessionID <value>]]
[--waitForCompletion <value>]
```

Where:

- `--dbname` (mandatory) specifies the name of the database.
- `--targetHome` specifies the target Oracle home location
- `--targetHomeName` specifies the name of the target Oracle Database home
- `--standBy` use this option to upgrade standby databases in Data Guard configurations
- `--allStandbyPrepared` required for Data Guard configured primary databases. Flags to acknowledge that all the required operations are performed on the standby databases prior to upgrading primary database
- `--removeGRP` automatically removes the Guaranteed Restore Point (GRP) backup only if the database upgrade was successful
- `--increaseCompatibleParameter` automatically increases the compatible parameter as part of the database upgrade. The parameter will get increased only if the database upgrade was successful
- `--executePrereqs` runs only the preupgrade checks
- `--postUpgrade` use this option if postupgrade fails and needs to rerun the postupgrade steps
- `--rollback` reverts an Oracle Database to its original Oracle home
- `--upgradeOptions` use this option to pass DBUA-specific arguments to perform the Oracle Database upgrade. Refer to the corresponding Oracle documentation for the supported arguments and options.
  `--standby`
- `--resume` to resume the previous execution
- `--sessionID` to resume a specific session id.
- `--waitForCompletion` specify false to run the operation in background. Valid values : true|false.

Example 4-4 dbaascli database upgrade pre-upgrade requisite checks

```
dbaascli database upgrade --dbbname dbname --targetHome Target Oracle home
location --executePrereqs
```

Prerequisites for Patching and Updating an VM Cluster

The Oracle Exadata Database Service on Exascale Infrastructure instance requires access to the Oracle Cloud Infrastructure Object Storage service, including connectivity to the applicable Swift endpoint for Object Storage

Oracle recommends using a service gateway with the VCN to enable this access. For more information, see these topics:

- Network Setup for Oracle Exadata Database Service on Exascale Infrastructure Instances: For information about setting up your VCN for the Exadata Cloud Service instance, including the service gateway.
- Object Storage FAQ

> **Note:**
>
> Ensure that the following conditions are met to avoid patching failures:
>
> - The `/u01` directory on the database host file system has at least 15 GB of free space for the execution of patching processes.
> - The Oracle Clusterware is up and running on the VM cluster.
> - All nodes of the VM cluster are up and running.

Using the Console to Patch and Update Exadata Database Service on Exascale Infrastructure VM Clusters

You can use the Console to view the history of patch operations on Oracle Exadata Database Service on Exascale InfrastructureOracle Exadata Database Service on Exascale Infrastructure VM clusters apply patches, and monitor the status of patch operations.

- To patch the Oracle Grid Infrastructure on an Exadata cloud VM cluster
  How to apply patches and monitor the status of patch operations on cloud VM clusters.
- To patch the Oracle Database software in a Database Home
- To patch individual Oracle Databases in Oracle Exadata Database Service on Exascale Infrastructure
  You can patch a single Oracle Database in your Oracle Exadata Database Service on Exascale Infrastructure by moving it to another Database Home.
- Viewing Patch History of Exadata Database Service on Exascale Infrastructure
  Each patch history entry represents an attempted patch operation and indicates whether the operation was successful or failed. You can retry a failed patch operation. Repeating an operation results in a new patch history entry.

To patch the Oracle Grid Infrastructure on an Exadata cloud VM cluster

How to apply patches and monitor the status of patch operations on cloud VM clusters.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Click **Exadata VM Clusters**.
4. In the list of cloud VM clusters, click the name of the cluster you want to patch to display the cluster details.
5. Under **Version**, click the **View Patches** link beside the **Updates Available**field.
6. Review the list of available patches for the cloud VM cluster.
7. Click the Actions menu for the patch you are interested in, and then click one of the following actions:
   - **Run Precheck:** Check for any prerequisites to make sure that the patch can be successfully applied.
   - **Update Grid Infrastructure:** Applies the selected patch. Oracle highly recommends that you run the precheck operation for a patch before you apply it.
8. Confirm when prompted.

The patch list displays the status of the operation. While a patch is being applied, the patch's status displays as **Patching** and the cloud VM cluster's status displays as **Updating**. Lifecycle operations on the cluster and its resources might be temporarily unavailable. If patching completes successfully, the patch's status changes to **Applied** and the status of the cluster changes to **Available**. You can view more details about an individual patch operation by clicking **Update History**.

To patch the Oracle Database software in a Database Home

> **Note:**
>
> This patching procedure updates the Oracle Database software for all databases located in the Database Home. To patch an individual database, you can move a database to another Database home using the Oracle Database software configuration that you want to have.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Click **Exadata VM Clusters**.
4. In the list of cloud VM clusters, click the name of the cluster you want to patch to display the cluster details.
5. Under **Resources**, click **Database Homes**.
6. Click the name of the Database Home you want to patch to display the Database Home details.
7. Under **Database Software Version**, locate the **Latest Patch Available** field and click **View**.
8. Review the available patches for the Database Home. You can choose to patch using an Oracle-provided software image or a custom software image. Oracle-provide images are generally available release updates. Custom software images are created by your organization with a specified set of patches. See Oracle Database Software Images for information on creating custom software images. The image you use to patch must be based on either the latest version of the Oracle Database software release or one of the three prior versions of the release.
9. Click the Actions menu at the end of the table row that lists the patch you are interested in, and then click one of the following actions:
   - **Precheck**: Check for any prerequisites to make sure that the patch can be successfully applied.
   - **Apply**: Applies the selected patch. Oracle highly recommends that you run the precheck operation for a patch before you apply it.
10. Confirm when prompted.

    The patch list displays the status of the operation. While a patch is being applied, the status of the patch displays as **Patching** and the status of the Database Home and the databases in it display as **Updating**. During the operation, each database in the home is stopped and then restarted. If patching completes successfully, the patch's status changes to **Applied** and the Database Home's status changes to **Available**. You can view more details about an individual patch operation by clicking **Update History**.

To patch individual Oracle Databases in Oracle Exadata Database Service on Exascale Infrastructure

You can patch a single Oracle Database in your Oracle Exadata Database Service on Exascale Infrastructure by moving it to another Database Home.

You can move a database to any Database Home that meets at either of the following criteria:

- The target Database Home uses the same Oracle Database software version (including patch updates) as the source Database Home
- The target Database Home is based on either the latest version of the Oracle Database software release used by the database, or one of the three prior versions of the release

Moving a database to a new Database Home brings the database up to the patch level of the target Database Home.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Navigate to the database you want to move.:
   Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, click the name of the VM cluster that contains the database you wan to move.
4. Click **More Actions**, and then click **Move to Another Home**.
5. Select the target Database Home.
6. Click **Move Database**.
7. Confirm the move operation.

   The database is moved in a rolling fashion. The database instance will be stopped, node by node, in the current home and then restarted in the destination home. While the database is being moved, the Database Home status displays as **Moving Database**. When the operation completes, the Database Home is updated with the current home. Datapatch is run automatically, as part of the database move, to complete post-patch SQL actions for all patches, including one-offs, on the new Database Home. If the database move operation is unsuccessful, then the status of the database displays as `Failed`, and the Database Home field provides information about the reason for the failure.

Viewing Patch History of Exadata Database Service on Exascale Infrastructure

Each patch history entry represents an attempted patch operation and indicates whether the operation was successful or failed. You can retry a failed patch operation. Repeating an operation results in a new patch history entry.

You can view patch history by navigating to the VM Cluster Details page.

Patch history views in the Console do not show patches that were applied by using command line tools such as `dbaascli`.

- To view the patch history of a cloud VM cluster
  Each patch history entry represents an attempted patch operation and indicates whether the operation was successful or failed.
- To view the patch history of a Database Home

To view the patch history of a cloud VM cluster

Each patch history entry represents an attempted patch operation and indicates whether the operation was successful or failed.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Click **Exadata VM Clusters**.
4. In the list of cloud VM clusters, click the name of the cluster you want to patch to display the cluster details.
5. Under **Version**, click the **View Patches** link beside the **Updates Available** field.
6. Click **Update History**.

   The Update History page displays the history of patch operations for that cloud VM cluster and for the Database Homes on that cloud VM cluster.

To view the patch history of a Database Home

Each patch history entry represents an attempted patch operation and indicates whether the operation was successful or failed. You can retry a failed patch operation. Repeating an operation results in a new patch history entry. When your service instance uses the new resource model, the patch history available by navigating to the VM Cluster Details page.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Navigate to the cloud VM cluster that contains the Database Home.

   Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.
4. Under **Resources**, click **Database Homes**.
5. Click the name of the Database Home you want to view to display the Database Home details.
6. Under **Database Software Version**, click **View** by the **Latest Patch Available** field.
7. Click **Update History**.

   The history page displays the history of patch operations for that Database Home and for the cloud VM cluster to which it belongs.

Using the API to Patch an Oracle Exadata Database Service on Exascale Infrastructure Instance

Use these API operations to manage patching the following Exadata resources: cloud VM clusters, databases, and Database Homes.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Cloud VM clusters:

- GetExadbVmClusterUpdate'
- ListExadbVmClusterUpdates

Databases:

- UpdateDatabase - Use this operation to patch a database by moving it to another Database Home

Database Homes:

- ListDbHomePatches
- ListDbHomePatchHistoryEntries
- GetDbHomePatch
- GetDbHomePatchHistoryEntry
- UpdateDbHome

For the complete list of APIs for the Database service, see Database Service API.

Updating an Exadata Cloud VM Cluster Operating System

Exadata VM cluster image updates allow you to update the OS image on your Exadata cloud VM cluster nodes in an automated manner from the OCI console and APIs.

This automated feature simplifies and speeds up VM cluster patching, makes patching less error-prone, and eliminates the need to use Patch Manager.

When you apply a patch, the system runs a precheck operation to ensure your cloud VM cluster, Exadata DB system, or Database Home meets the requirements for that patch. If the precheck is not successful, the patch is not applied, and the system displays a message that the patch cannot be applied because the precheck failed. A separate precheck operation that you can run in advance of the planned update is also available.

- Updating the Operating System using the Console

Updating the Operating System using the Console

> **Note:**
>
> After the VM cluster is upgraded to Exadata Database Service Guest VM OS 23.1, you will be able to add a new VM or a new database server to this VM cluster if Exadata Cloud Infrastructure is running an Exadata System Software version 22.1.16 and later.
> Upgrade to Exadata System Software 23.1 for Exadata Cloud Infrastructure will be available with February 2024 update cycle.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**.
3. In the list of cloud VM clusters, click the name of the cluster that you want to patch to display the details page.
4. In the **Version** section, to the right of the **Updates Available**, click **View Updates** to display the **Updates** page.
5. Review the list of available software updates and locate the OS patch you are applying.

6. Click the Actions icon (three dots) at the end of the row listing the patch you are interested in, and then click one of the following actions:
   - **Run Precheck.** Precheck checks the prerequisites to ensure that the patch can be successfully applied. Oracle highly recommends that you run the precheck operation before you apply a patch. The reason is that things can change in a database at any time, and the precheck you run just before running a patch may find errors that the previous precheck did not find

     > **Note:**
     >
     > If the precheck fails, the system displays a message in the **Apply Exadata OS Image Update** dialog that the last precheck has failed. Oracle recommends that you run the precheck again. Click the Actions icon (three dots) at the end of the row listing the OS patch to view the dialog.

   - **Apply Exadata OS Image Update**. This link displays the Apply Exadata Image Update dialog that you use to apply the patch. The dialog shows the name of the database system you are patching, the current version of the database, and the new version of the database after the patch is applied. To start the process, click **Apply Exadata OS Image Update**.
   - **Copy OCID.** This copies the Oracle Cloud ID. This can be used when troubleshooting a patch or to give to Support when contacting them.

     > **Note:**
     >
     > While the patch is running:
     >
     > – Run Precheck and Apply OS Image Update are not available. When the patch has completed, these actions are available again.
     >
     > – If the Exadata infrastructure containing this VM cluster is scheduled for maintenance that conflicts with the patching operation, the patch fails and the system displays a message explaining why. After the infrastructure maintenance is complete, run the patch operation again.

7. Confirm when prompted.

The patch list displays the status of the operation in the Version section of the database details page. Click **View Updates** to view more details about an individual patch status and to display any updates that are available to run. If no new updates are available, the system displays a message that says **No Updates Available**.

Upgrading Exadata Databases

Oracle Database releases on Oracle Exadata Database Service on Exascale Infrastructure can be upgraded using the Console and the API.

The upgrade is accomplished by moving the Exadata database to a Database Home that uses the target software version.

- Prerequisites to Upgrade Oracle Databases
  Review the list of prerequsites to upgrade an Oracle Exadata Database Service on Exascale Infrastructure Oracle Database instance.

- About Upgrading a Database
- Using the Console to Upgrade a Database
  Procedures to precheck and upgrade a database, rollback a failed upgrade, and view the upgrade history.
- Using the API to upgrade Databases
  Use the following APIs to manage database upgrades:

Related Topics

- Release Schedule of Current Database Releases (Doc ID 742060.1)

Prerequisites to Upgrade Oracle Databases

Review the list of prerequsites to upgrade an Oracle Exadata Database Service on Exascale Infrastructure Oracle Database instance.

- You must have an available Oracle Database Home that uses the four most recent versions of Oracle Database available. See *To Create a new Oracle Database Home in an existing Oracle Exadata Database Service on Exascale Infrastructure Instance* for information on creating a Database Home. You can use Oracle-published software images or a *custom database software image* based on your patching requirements to create Database Homes.
- You must ensure that all pluggable databases in the container database that is being upgraded can be opened. Pluggable databases that cannot be opened by the system during the upgrade can cause an upgrade failure.
- If you are upgrading databases in a manually-created Data Guard association (an association not created using the Console or APIs), the following apply:
  - The databases must be registered with the Cloud tooling.
  - Redo apply needs to be disabled during the upgrade of both the primary and standby.
  - If you have configured an observer, then the observer needs to be disabled prior to upgrade.

Before you start the upgrade, your Oracle Database configuration must be configured with the following settings:

- The database must be in archive log mode.
- The database must have flashback enabled.

To learn more about these settings, see the Oracle Database documentation for your database release.

Related Topics

- Oracle Database Software Images
- Oracle Database Documentation

About Upgrading a Database

For database software version upgrades, note the following:

- Database upgrades involve database downtime. Keep this in mind when scheduling your upgrade.
- Oracle recommends that you back up your database and test the new software version on a test system or a cloned version of your database before you upgrade a production

database. See *to create an on-demand full backup of a database* for information on creating an on-demand manual backup.

- Oracle recommends running an upgrade precheck operation for your database prior to attempting an upgrade so that you can discover any issues that need mitigation prior to the time you plan to perform the upgrade. The precheck operation does not affect database availability and can be performed at any time that is convenient for you.
- If your databases uses Data Guard, you can upgrade either the primary or the standby first. To upgrade a primary, follow the steps in To upgrade or precheck an Exadata database. To upgrade a standby, follow the steps in To move a database to another Database Home
- If your databases uses Data Guard, upgrading a primary or standby will disable redo apply during the upgrade operation. After you upgrade both the primary and standby, redo apply and open mode are re-enabled. Oracle recommends checking the redo apply and open mode configuration after upgrading.
- An upgrade operation cannot take place while an automatic backup operation is underway. Before upgrading, Oracle recommends disabling automatic backups and performing a manual backup. See *to configure automatic backups for a database* and *To create an on-demand full backup of a database* for more information.
- After upgrading, you cannot use automatic backups taken prior to the upgrade to restore the database to an earlier point in time.

- How the Upgrade Operation Is Performed by the Database Service
During the upgrade process, the Database service does the following:
- Rolling Back an Oracle Database Unsuccessful Upgrade
If your upgrade does not complete successfully, then you have the option of performing a rollback.
- After Upgrading an Oracle Database
After a successful upgrade, note the following:

How the Upgrade Operation Is Performed by the Database Service

During the upgrade process, the Database service does the following:

- Executes an automatic precheck. This allows the system to identify issues needing mitigation and to stop the upgrade operation.
- Sets a guaranteed restore point, enabling it to perform a flashback in the event of an upgrade failure.
- Moves the database to a user-specified Oracle Database Home that uses the desired target software version.
- Runs the Database Upgrade Assistant (DBUA) software to perform the upgrade.
- For databases in Data Guard associations, redo apply is disabled until both the primary and standby databases are successfully upgraded, at which point redo apply is re-enabled by the system. The system then enables Open Mode after redo apply is enabled.

Rolling Back an Oracle Database Unsuccessful Upgrade

If your upgrade does not complete successfully, then you have the option of performing a rollback.

Details about the failure are displayed on the **Database Details** page in the Console, allowing you to analyze and resolve the issues causing the failure.

A rollback resets your database to the state prior to the upgrade. All changes to the database made during and after the upgrade will be lost. The rollback option is provided in a banner message displayed on the database details page of a database following an unsuccessful upgrade operation. See *Using the Console to Roll Back a Failed Database Upgrade* for more information.

For standby databases in Oracle Data Guard associations, rollback is accomplished by moving the standby back to the original Database Home. See To move a database to another Database Home for instructions.

Related Topics

- To roll back a failed database upgrade

After Upgrading an Oracle Database

After a successful upgrade, note the following:

- Check that automatic backups are enabled for the database if you disabled them prior to upgrading. See *Customizing the Automatic Backup Configuration* for more information.
- Edit the Oracle Database `COMPATIBLE` parameter to reflect the new Oracle Database software version. See *What Is Oracle Database Compatibility?* for more information.
- If your database uses a `*database_name*.env` file, ensure that the variables in the file have been updated to point to the new Database home. These variables should be automatically updated during the upgrade process.
- If you are upgrading a non-container database, you can convert the database to a pluggable database after converting. See *How to Convert Non-CDB to PDB (Doc ID 2288024.1)* for instructions on converting your database to a pluggable database.
- If your old Database Home is empty and will not be reused, then you can remove it. See *Using the Console to Delete an Oracle Database Home* for more information.
- For databases in Data Guard associations, check the open mode and redo apply status after the upgrade is complete.

Related Topics

- Managing Exadata Database Backups by Using bkup_api
- What Is Oracle Database Compatibility?
- How to Convert Non-CDB to PDB - Step by Step Example (Doc ID 2288024.1)

Using the Console to Upgrade a Database

Procedures to precheck and upgrade a database, rollback a failed upgrade, and view the upgrade history.

- To upgrade or precheck an Exadata database
  Procedure to upgrade or precheck an Exadata database.
- To roll back a failed database upgrade
- To view the the upgrade history of a database
  To view upgrade history for databases on Exadata Database Service on Exascale Infrastructure, use this procedure.

To upgrade or precheck an Exadata database

Procedure to upgrade or precheck an Exadata database.

The following steps apply to databases for which either of the following apply:

- The database is the primary database in a Data Guard association
- The database is not part of a Data Guard association

To upgrade a standby database in a Data Guard configuration, move the standby to a Database Home using the Oracle Database version you are upgrading to.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Under**Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, click the name of the VM cluster that contains the database you want to upgrade.
4. In the list of databases on the details page of the VM cluster, click the name of the database you want to upgrade to view the Database Details page.
5. Click **More Actions**, then **Upgrade**.
6. In the **Upgrade Database** dialogue, select the following:
   - **Oracle Database version:** The drop-down selector lists only Oracle Database versions that are compatible with an upgrade from the current software version the database is using. The target software version must be higher than the database's current version.
   - **Target Database Home:** Select a Database Home for your database. The list of Database Homes is limited to those homes using the most recent versions of Oracle Database 19c software. Moving the database to the new Database Home results in the database being upgraded to the major release version and patching level of the new Database Home.
7. Click one of the following:
   - **Run Precheck:** This option starts an upgrade precheck to identify any issues with your database that need mitigation before you perform an upgrade.
   - **Upgrade Database:** This option starts upgrade operation. Oracle recommends performing an upgrade only after you have performed a successful precheck on the database.

To roll back a failed database upgrade

1. Open the navigation menu. Under **Oracle Database**, click **Exadata Database Service on Exascale Infrastructure** .
2. Choose your **Compartment**.

   A list of VM Clusters is displayed for the chosen Compartment.
3. In the list of VM clusters, click the name of the VM cluster that contains the database with the failed upgrade.
4. Find the database that was unsuccessfully upgraded, and click its name to display details about it.
5. The database must display a banner at the top of the details page that includes a **Rollback** button and details about what issues caused the upgrade failure.
6. Click **Rollback**.

7. In the **Confirm rollback** dialog, confirm that you want to initiate a rollback to the previous Oracle Database version.

To view the the upgrade history of a database

To view upgrade history for databases on Exadata Database Service on Exascale Infrastructure, use this procedure.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose your **Compartment**.
3. Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, click the name of the VM cluster that contains the database you want to upgrade.
4. In the list of databases on the details page of the VM cluster, click the name of the database for which you want to view the upgrade history.
5. On the Database Details page, under **Database Version**, click the **View** link that is displayed for databases that have been upgraded. This link does not appear for databases that have not been updated.
   The **Updates History** page is displayed. The table displayed on this page shows precheck and upgrade operations performed on the database.

Using the API to upgrade Databases

Use the following APIs to manage database upgrades:

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage database upgrades:

- ListDatabaseUpgradeHistoryEntries
- UpgradeDatabase

For the complete list of APIs for the Database service, see Database Service API.

> **Note:**
>
> When using the `UpgradeDatabase` API to upgrade an Oracle Exadata Database Service on Exascale Infrastructure database, you must specify `DB_HOME` as the upgrade source.

Interim Software Updates

For authorized environments, learn how to download interim software updates.

This feature enables cloud-only customers to download one-off patches from the OCI console and API. There is no option to apply the downloaded patch via console and API. To apply these patches, customers must log in to their VM and run the patch apply utility.

> **Note:**
>
> To be able to download interim software update, you should at least have an ExaDB-D infrastructure provisioned.

Downloading one-off patches does not replace Database Software Image (DSI) creation. Customers must continue to use Database Software Images (DSI) to build and deploy their customized images.

- Create Software Update
- Download an Interim Software Update
- Delete an Interim Software Update
- Move an Interim Software Update Resource to Another Compartment
- Using the API to Manage Interim Software Updates

Create Software Update

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata on Oracle Public Cloud**.
2. Under **Resources**, click **Manual software updates**.

   Manual software update page is displayed.
3. Click **Create software update**.

   Create software update panel is displayed.
4. Enter the following details in the panel:
   a. **Name**: Descriptive name for the patch download path.
   b. **Compartment**: Select a compartment where you want to create the patch resource.
   c. **Database version**: Choose the Database version for your image.
   d. **Release Update**: Choose any supported Oracle Database release update (RU).
   e. **One-off patch number**: Optionally, enter a one-off (interim) patch number.
   f. **Tag**: Apply a tag.
5. Click **Create**.

Download an Interim Software Update

The patch download path is valid for four days. Download the patch within the specified timeframe.

1. On the Update details page, click **Download**.

   The system starts downloading the patch.
2. You can also download a patch from the Interim Software Updates page.

   - Click the Actions button (three dots) for the patch you're interested in, and select **Download**.

> **Note:**
>
> You can only download the patches that are in **Available** state.

Interim Software Updates Lifecycle States:

- **Available**: Patch has been created successfully and the time-to-live (TTL) has not expired.
- **Creating**: The patch creation process is in progress.
- **Expired**: The lifetime of the patch download link has expired, which means you cannot download it.
- **Failed**: The patch create failed due to some error.
- **Terminating**: The patch deletion process is in progress.
- **Terminated**: The patch has been deleted.

Delete an Interim Software Update

Be discrete in deleting interim software updates. However, you can delete the interim software updates that have expired to free up space in the Object Store.

1. On the Update details page, click **Delete**.
2. In the resulting dialog, enter the name of the patch to confirm and then click **Delete**.
3. You can also delete a patch from the Interim Software Updates page.

   - Click the Actions button (three dots) for the patch you're interested in, and select **Delete**.

Move an Interim Software Update Resource to Another Compartment

1. On the Update details page, click **Move Resource**.
2. In the resulting dialog, choose a new compartment, and click **Move Resource**.
3. You can also move a patch resource from the Interim Software Updates page.

   - Click the Actions button (three dots) for the patch you're interested in, and select **Move Resource**.

Using the API to Manage Interim Software Updates

ExaDB-C@C and ExaDB-D use the same API to manage interim software updates.

For information about using the API and signing requests, see *REST APIs* and *Security Credentials*. For information about SDKs, see *Software Development Kits and Command Line Interface*.

Use these API operations to manage interim software updates:

- `CreateOneoffPatch`
- `DeleteOneoffPatch`

- `DownloadOneoffPatch`
- `UpdateOneoffPatch`
- `ListOneoffPatches`
- `GetOneoffPatch`
- `ChangeOneoffPatchCompartment`

Related Topics

- REST APIs
- Security Credentials
- Software Development Kits and Command Line Interface
- OneoffPatch Reference

Use Oracle Data Guard with Oracle Exadata Database Service on Exascale Infrastructure

Learn to configure and manage Data Guard associations in your VM cluster.

- About Using Oracle Data Guard with Oracle Exadata Database Service on Exascale Infrastructure
  This topic explains how to use the Console or the API to manage Data Guard associations in your VM cluster.
- Prerequisites for Using Oracle Data Guard with Oracle Exadata Database Service on Exascale Infrastructure
  An Oracle Data Guard implementation requires two existing Exadata VM Clusters: one containing an existing database that is to be duplicated by Data Guard, and one that will house the new Data Guard standby database.
- Working with Oracle Data Guard
  Oracle Data Guard ensures high availability, data protection, and disaster recovery for enterprise data.
- Using the Console to Manage Oracle Data Guard Associations
  Learn how to enable a Data Guard association between databases, change the role of a database in a Data Guard association using either a switchover or a failover operation, and reinstate a failed database.
- Using the API to manage Data Guard associations
  Use these API operations to manage Data Guard associations on an Oracle Exadata Database Service on Exascale Infrastructure instance:

About Using Oracle Data Guard with Oracle Exadata Database Service on Exascale Infrastructure

This topic explains how to use the Console or the API to manage Data Guard associations in your VM cluster.

When you use the Console or the API to enable Data Guard for an Exadata database compute node database:

- The standby database is a physical standby.
- The versions of peer databases (primary and standby) are identical.

- You are limited to one standby database for each primary database.
- The standby database is deployed as an open, read-only database (Active Data Guard).

To configure a Data Guard system between on-premises and Exadata database compute nodes, or to configure your database with multiple standbys, you must access the database host directly and set up Data Guard manually.

For complete information on Oracle Data Guard, see the *Data Guard Concepts and Administration* documentation on the *Oracle Document Portal*.

Related Topics

- Data Guard Concepts and Administration
- Oracle Document Portal

Prerequisites for Using Oracle Data Guard with Oracle Exadata Database Service on Exascale Infrastructure

An Oracle Data Guard implementation requires two existing Exadata VM Clusters: one containing an existing database that is to be duplicated by Data Guard, and one that will house the new Data Guard standby database.

When enabling Oracle Data Guard, you can create a new Database Home on the standby Exadata instance to house the new standby database during the enable Data Guard operation. Alternately, you can choose to provision the standby database in an existing Database Home on the standby instance.

You can use a custom database software image to that contains the necessary patches for your databases when creating a Database Home on either the primary or the standby Exadata instance.

If you choose to provision a standby database in an existing Database Home, ensure that the target Database Home on the standby instance has all required patches that are in use for the primary database before you provision the standby database. :

If you are creating an Oracle Data Guard Association and you are using customer managed keys to encrypt the database, you must have configured the Vault Service and created a master key. See *To administer Vault encryption keys* and *Key and Secret Management Concepts*.

- Network Requirements for Data Guard
  Ensure that you meet the requirements for using Oracle Exadata Database Service on Exascale Infrastructure with Oracle Data Guard.
- Password Requirements
  For Data Guard operations to work, the `SYS` password and the TDE wallet password of the primary and standby databases must all be the same.
- Known Issues for Exadata Cloud Infrastructure and Data Guard
  Possible TDE key replication issue, and MRP and DG LCM operation failures.
- Adding a Node to a VM Cluster
  If node addition is done either on the standby database or the primary database, the metadata must be updated manually on the database other than the one where the node was added.

- Removing a Node from a VM Cluster
  If node removal is done either on the standby database or the primary database, the metadata must be updated manually on the database other than the one where the node was removed.

Network Requirements for Data Guard

Ensure that you meet the requirements for using Oracle Exadata Database Service on Exascale Infrastructure with Oracle Data Guard.

Ensure that your environment meets the following network requirements:

- The primary and standby databases can be part of VM clusters in different compartments.
- The primary and standby databases must, however, be part of the same VCN within the same region.
- If you want to configure Oracle Data Guard across regions, then you must configure remote virtual cloud network (VCN) peering between the primary and standby databases. Networking is configured on the cloud VM cluster resource.

  For Exadata Data Guard configurations, OCI supports the use of hub-and-spoke network topology for the VCNs within each region. This means that the primary and standby databases can each utilize a "spoke" VCN that passes network traffic to the "hub" VCN that has a remote peering connection. See *Transit Routing inside a hub VCN* for information on setting up this network topology.
- To set up Oracle Data Guard within a single region, both Oracle Exadata Database Service on Exascale Infrastructure instances must use the same VCN. When setting up Data Guard within the same region, Oracle recommends that the instance containing the standby database be in a different **availability domain** from the instance containing the primary database to improve availability and disaster recovery.
- Configure the ingress and egress security rules for the subnets of both Oracle Exadata Database Service on Exascale Infrastructure instances in the Oracle Data Guard association to enable TCP traffic to move between the applicable ports. Ensure that the rules you create are stateful (the default).

  For example, if the subnet of the primary Oracle Exadata Database Service on Exascale Infrastructure instance uses the source CIDR 10.0.0.0/24 and the subnet of the standby instance uses the source CIDR 10.0.1.0/24, then create rules as shown in the subsequent example.

> **Note:**
>
> The egress rules in the example show how to enable TCP traffic only for port 1521, which is a minimum requirement for Oracle Data Guard to work. If TCP traffic is already enabled for all destinations (0.0.0.0/0) on all of your outgoing ports, then you need not explicitly add these specific egress rules.

Security Rules for Subnet of Primary Oracle Exadata Database Service on Exascale Infrastructure instance

```
Ingress Rules:

                        Stateless: No
                        Source: 10.0.1.0/24
```

```
IP Protocol: TCP
Source Port Range: All
Destination Port Range: 1521
Allows: TCP traffic for ports: 1521
```

Egress Rules:

```
Stateless: No
Destination: 10.0.1.0/24
IP Protocol: TCP
Source Port Range: All
Destination Port Range: 1521
Allows: TCP traffic for ports: 1521
```

Security Rules for Subnet of Standby Oracle Exadata Database Service on Exascale Infrastructure instance

Ingress Rules:

```
Stateless: No
Source: 10.0.0.0/24
IP Protocol: TCP
Source Port Range: All
Destination Port Range: 1521
Allows: TCP traffic for ports: 1521
```

Egress Rules:

```
Stateless: No
Destination: 10.0.0.0/24
IP Protocol: TCP
Source Port Range: All
Destination Port Range: 1521
Allows: TCP traffic for ports: 1521
```

For information about creating and editing rules, see *Security Lists* .

Related Topics

- Remote VCN Peering using an RPC
- Transit Routing inside a hub VCN
- Security Lists

Password Requirements

For Data Guard operations to work, the `SYS` password and the TDE wallet password of the primary and standby databases must all be the same.

If you change any one of these passwords, you must update the rest of the passwords to match. See *Changing the Database Passwords* to learn how to change the SYS password or the TDE wallet password.

If you make any change to the TDE wallet (such as adding a master key for a new PDB or changing the wallet password), you must copy the wallet from the primary to the standby so that Data Guard can continue to operate. For Oracle Database versions earlier than 12.2, if you change the `SYS` password on one of the peers, you need to manually sync the password file between the DB systems.

Related Topics

- Changing the Database Passwords
  To change the SYS password, or to change the TDE wallet password, use this procedure.

Known Issues for Exadata Cloud Infrastructure and Data Guard

Possible TDE key replication issue, and MRP and DG LCM operation failures.

KMS RPM `libkmstdepkcs11_1.286-1.286-1-Linux.rpm` is the latest available which supports active replication of key between cross-region KMS vaults (source and target), and it is recommended to upgrade the RPM on clusters participating in Data Guard. OCI Vault cross-region Data Guard works with a lower version of RPM, but the older version does not guarantee active replication of keys. If the TDE keys have any replication issue between vaults, Data Guard replication might have an impact (MRP fails on standby cluster due to missing key on target vault) and MRP could resume only after the keys are replicated to the target vault. To avoid MRP and DG LCM operation failures, upgrade the `libkms` RPM on both the clusters, and restart the databases (only databases using customer-managed keys).

Adding a Node to a VM Cluster

If node addition is done either on the standby database or the primary database, the metadata must be updated manually on the database other than the one where the node was added.

When adding a node to a VM cluster, an instance of the Data Guard database is automatically created on the new node. However, metadata updation on the remote database, that is, the primary database if addition is done on the standby database and vice versa, must be done manually.

This can be done by copying over the `addinstance` JSON file, `/var/opt/oracle/dbaas_acfs/<dbname>/addInstance.json` created at the end of instance addition and running the `/var/opt/oracle/ocde/rops update_instance <dbname> <path to addInstance JSON>` command on any node of the remote cluster.

Removing a Node from a VM Cluster

If node removal is done either on the standby database or the primary database, the metadata must be updated manually on the database other than the one where the node was removed.

When removing a node from a VM cluster, the instance and it's metadata on the removing node is deleted automatically. However, deletion of the corresponding metadata on the remote database, that is, the primary database if removal is done on the standby database and vice versa, must be done manually.

This can be done by running the `/var/opt/oracle/ocde/rops remove_instance <dbname> <Instance Name>` command on any node of the remote cluster.

Working with Oracle Data Guard

Oracle Data Guard ensures high availability, data protection, and disaster recovery for enterprise data.

The Data Guard implementation requires two databases, one in a primary role and one in a standby role. The two databases compose a Data Guard association. Most of your applications access the primary database. The standby database is a transactionally consistent copy of the primary database.

Data Guard maintains the standby database by transmitting and applying redo data from the primary database. If the primary database becomes unavailable, you can use Data Guard to switch or fail over the standby database to the primary role.

- Switchover
  A switchover reverses the primary and standby database roles.
- Failover
  With Oracle Data Guard, a failover transitions the standby database into the primary role after the existing primary database fails or becomes unreachable.
- Reinstate
  The reinstate command reinstates a database into the standby role in an Oracle Data Guard association.

Switchover

A switchover reverses the primary and standby database roles.

Each database continues to participate in the Data Guard association in its new role. A switchover ensures no data loss. You can use a switchover before you perform planned maintenance on the primary database. Performing planned maintenance on a Exadata database virtual machine with a Data Guard association is typically done by switching the primary to the standby role, performing maintenance on the standby, and then switching it back to the primary role.

Failover

With Oracle Data Guard, a failover transitions the standby database into the primary role after the existing primary database fails or becomes unreachable.

A failover might result in some data loss when you use **Maximum Performance** protection mode.

Reinstate

The reinstate command reinstates a database into the standby role in an Oracle Data Guard association.

You can use the reinstate command to return a failed database into service after correcting the cause of failure.

> **Note:**
>
> You can't terminate a primary database that has a Data Guard association with a peer (standby) database. Delete the standby database first. Alternatively, you can switch over the primary database to the standby role, and then terminate the former primary.
>
> You can't terminate a VM cluster that includes Data Guard-enabled databases. You must first remove the Data Guard association by terminating the standby database.

Using the Console to Manage Oracle Data Guard Associations

Learn how to enable a Data Guard association between databases, change the role of a database in a Data Guard association using either a switchover or a failover operation, and reinstate a failed database.

When you enable Data Guard, a separate Data Guard association is created for the primary and the standby database.

- To enable Data Guard on Exadata Database Service on Exascale Infrastructure
  Learn how to enable Data Guard association between databases.
- To view Data Guard associations of databases in a Cloud VM Cluster
  To view the role of each database in a Data Guard association in an Cloud VM Cluster, follow this procedure.
- To enable automatic backups on a standby database
  Learn to enable automatic backups on a standby database.
- To perform a database switchover
  You initiate a switchover operation by using the Data Guard association of the primary database.
- To edit the Oracle Data Guard association
  You edit the Oracle Data Guard association to configure the Data Guard protection for the primary database.
- To perform a database failover
  You initiate a failover operation by using the Data Guard association of the standby database.
- To reinstate a database
- To terminate a Data Guard association on an Oracle Exadata Database Service on Exascale Infrastructure instance
  On an Oracle Exadata Database Service on Exascale Infrastructure instance, you remove a Data Guard association by terminating the standby database.

To enable Data Guard on Exadata Database Service on Exascale Infrastructure

Learn how to enable Data Guard association between databases.

> **Note:**
>
> When you enable Data Guard, replication of data happens only over the client network.

1. Open the navigation menu. Under **Oracle Database**, click **Exadata Database Service on Exascale Infrastructure**.
2. Choose your **Compartment** that contains the Oracle Exadata Database Service on Exascale Infrastructure instance with the database for which you want to enable Oracle Data Guard..
3. Navigate to the cloud VM cluster that contains a database you want to assume the primary role:

4. Under **Exadata Database Service on Exascale Infrastructure**, click **Exadata VM clusters**. In the list of VM clusters, find the VM cluster that you want to access and click its highlighted name to view the details page for the cluster.
5. On the VM cluster details page, in the **Databases** section, click the name of the database that you want to make primary.
6. On the Database Details page, under **Resources**, click **Data Guard Associations**.
7. In the **Data Guard Associations** section, click **Enable Data Guard**.
8. On the Enable Data Guard page, configure your Data Guard association.
    - In the **Select VM Cluster** section, provide the following information for the standby database to obtain a list of available Exadata systems in which to locate the standby database:
        - **Region**: Select a region where you want to locate the standby database. The region where the primary database is located is selected, by default. You can choose to locate the standby database in a different region. The hint text associated with this field tells you in which region the primary database is located.
        - **Availability domain**: Select an availability domain for the standby database. The hint text associated with this field tells you in which availability domain the primary database is located.
        - **Data Guard peer resource type**: Select **VM Cluster**.

          Select a VM cluster from the drop-down list.
    - **Data Guard association details**
        - **Data Guard Type**: Select Active Data Guard or Data Guard. Active Data Guard provides additional features including: Real-Time Query and DML Offload, Automatic Block Repair, Standby Block Change Tracking, Far Sync, Global Data Services, and Application Continuity. Note that Active Data Guard requires an Oracle Active Data Guard license. For more information on Active Data Guard, see Active Data Guard. For a complete overview of both Data Guard types, see Introduction to Oracle Data Guard.
        - **Protection mode**: The protection mode can be **Maximum Performance** or **Maximum Availability**. See Oracle Data Duard Protection Modes for information on these options.
        - **Transport type:** The redo transport type used for this Data Guard association.

          See Redo Transport Services for information on these options.
    - In the **Choose Database Home** section, choose one of the following:
        - **Select an existing Database Home:** If you use this option, select a home from the Database Home display name drop-down list.
        - **Create a new Database Home:** If you choose this option, enter a name for the new Database Home in the **Database Home display name** field. Click **Change Database Image** to select a database software image for the new Database Home. In the **Select a Database Software Image** panel, do the following:

          a. Select the compartment containing the database software image you want to use to create the new Database Home.

          b. Select the Oracle Database software version that the new Database Home will use, then choose an image from the list of available images for your selected software version.

          c. Click **Select**.

> **Note:**
>
> Oracle recommends applying the same list of patches to the Database Homes of the primary and standby databases.

- In the **Configure standby database:** section, provide standby database details.

  > **Note:**
  >
  > You cannot modify the `db_unique_name` and SID prefix after creating the database.

  - **Database unique name:** Optionally, specify a value for the `DB_UNIQUE_NAME` database parameter. This value must be unique across the primary and standby cloud VM clusters. The unique name must meet the requirements:
    * Maximum of 30 characters
    * Contain only alphanumeric or underscore (_) characters
    * Begin with an alphabetic character
    * Unique across the VM cluster. Recommended to be unique across the tenancy.

    If not specified, the system automatically generates a unique name value, as follows:

    ```
    <db_name>_<3_chars_unique_string>_<region-name>
    ```

  - **Database password**: Enter the database administrator password of the primary database. Use this same database administrator password for the standby database.

    > **Note:**
    >
    > The administrator password and the TDE wallet password must be identical. If the passwords are not identical, then follow the instructions in Changing the Database Passwords to ensure that they are.

9. *Optional.* **Enable thin clone**: Select this option to leverage Exascale redirect-on-write technology to create a thin clone of the PDB. This option results in the reuse of duplicate blocks with the parent PDB, shared with the clone. Deselecting this option results in a traditional, full clone with all blocks copied, and fully independent from the parent.
10. Click **Show Advanced Options** to specify advanced options for the standby database:
    - **Management:**

      Oracle SID prefix: The Oracle Database instance number is automatically added to the SID prefix to create the `INSTANCE_NAME` database parameter. The `INSTANCE_NAME` parameter is also known as the SID. If not provided, then the SID prefix defaults to the first 12 characters of the `db_unique_name`.

      The SID prefix must meet the requirements:

  - Maximum of 12 characters
  - Contain only alphanumeric characters
  - Begin with an alphabetic character
  - Unique in the VM cluster and across primary and standby databases

11. Click **Enable Data Guard**. When you create the association, the details for a database and its peer display their respective roles as **Primary** or **Standby**.

A work request is issued to configure the Data Guard association. The progress of the request and the stages of provisoning an be viewed on the **Work Requests** page.

When the association is created, the details for a database and its peer display their respective roles as **Primary** or **Standby**.

- View Data Guard Provisioning Progress
  View the progress of Data Guard Provisioning tasks using the Work Requests page.

Related Topics

- Network Setup for Oracle Exadata Database Service on Exascale Infrastructure Instances
  This topic describes the recommended configuration for the VCN and several related requirements for the Oracle Exadata Database Service on Exascale Infrastructure instance.
- Changing the Database Passwords
  To change the SYS password, or to change the TDE wallet password, use this procedure.

View Data Guard Provisioning Progress

View the progress of Data Guard Provisioning tasks using the Work Requests page.

After you have completed the task To Enable Data Guard, multiple work requests are issued to complete the provisioning of the Data Guard association. To veiw the progress of these work requests:

1. Navigate to the **Work Requests Details** page. On the **Work Requests Details** page there is a bar in the Work Request Information tab that shows the overall progress of the Data Guard Provisioning
2. Under Resources, select **Log Messages**. The table shows a messsage for each task that is completed or in progress.

To view Data Guard associations of databases in a Cloud VM Cluster

To view the role of each database in a Data Guard association in an Cloud VM Cluster, follow this procedure.

1. Open the navigation menu. Under **Oracle Database**, click **Exadata Database Service on Exascale Infrastructure**.
2. Choose your Compartment.
3. Navigate to the cloud VM cluster that contains the databases you wish to view their roles in Data Guard associations.
4. In the **Databases** section under **Resources**, the role of each database in this VM Cluster is indicated in the **Data Guard role** column.

To enable automatic backups on a standby database

Learn to enable automatic backups on a standby database.

1. Open the navigation menu. Under **Oracle Database**, click **Exadata Database Service on Exascale Infrastructure**.
2. Choose your Compartment that contains the Exadata Cloud Infrastructure instance with the database for which you want to enable automatic database.
3. Navigate to the cloud VM cluster or DB system that contains the primary database. Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.
4. On the VM cluster page, in the **Databases** section, click the name of the primary database.
5. On the Database Details page, under **Resources**, click **Data Guard Associations**.
6. Click the name of the standby database for which you want to enable automatic backups.

   The system displays a banner if automatic backups are not enabled for this database.
7. Click **Enable automatic backups** on the banner.
8. On the resulting Configure Automatic Backups window, enter the following details:
   - **Enable automatic backup:** Check the check box to enable or disable automatic incremental backups for this database. If your database is in a security zone compartment, you must enable automatic backups.
   - **Backup Scheduling:**
     - **Full backup scheduling day:** Choose a day of the week for the initial and future L0 backups to start.
     - **Full backup scheduling time (UTC):** Specify the time window when the full backups start when the automatic backup capability is selected.
     - **Take the first backup immediately:** A full database backup includes all datafiles, control file, and parameter files associated with the target database. Archive backups are separate and decoupled and executed every 30 minutes. You can choose to execute the first full backup immediately or defer to the assigned full backup scheduling time. If you defer to the latter, the database will not be recoverable until the first backup completes.
   - **Backup Destination:** Object Storage is selected by default and you cannot change it.

> **Note:**
>
> – If automatic backup is enabled on the primary database and the backup destination is Autonomous Recovery Service, then you cannot enable backup on the standby database.
>
> – If automatic backup is enabled on the primary database and the backup destination is Object Storage, then you can enable backup on the standby database. Note that you can only select Object Storage as the backup destination.
>
> – If automatic backup is disabled on the primary database, you can still enable backup on the standby database by selecting Object Storage as the backup destination.

9. Click **Save Changes**.

To perform a database switchover

You initiate a switchover operation by using the Data Guard association of the primary database.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose the **Compartment** that contains the Oracle Exadata Database Service on Exascale Infrastructure instance with the database for which you want to enable Oracle Data Guard.
3. Navigate to the cloud VM cluster or DB system that contains the Data Guard association:

   Oracle Exadata Database Service on Exascale Infrastructure, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.
4. Under **Resources**, click **Data Guard Associations**.
5. For the Data Guard association on which you want to perform a switchover, click the Actions icon (three dots), and then click **Switchover**.
6. In the **Switchover Database** dialog box, enter the database admin password, and then click **OK**.

   This database should now assume the role of the standby, and the standby should assume the role of the primary in the Data Guard association.

To edit the Oracle Data Guard association

You edit the Oracle Data Guard association to configure the Data Guard protection for the primary database.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose the **Compartment** that contains the Exadata Cloud Service instance with the database for which you want to enable Oracle Data Guard.
3. Navigate to the cloud VM cluster or DB system that contains the Data Guard association:

Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.

4. Under **Resources**, click **Data Guard Associations**.
5. For the Data Guard association you want to manage, click the Actions menu (

   ⋮

   ), and then click **Edit Protection Mode**.
6. In the **Edit Data Guard Association** panel, configure the Data Guard association:
   - **Data Guard Type**: Select Active Data Guard or Data Guard. Active Data Guard provides additional features including: Real-Time Query and DML Offload, Automatic Block Repair, Standby Block Change Tracking, Far Sync, Global Data Services, and Application Continuity. Note that Active Data Guard requires an Oracle Active Data Guard license. For more information on Active Data Guard, see Active Data Guard. For a complete overview of both Data Guard types, see Introduction to Oracle Data Guard
   - **Protection mode**: The protection mode can be **Maximum Performance** or **Maximum Availability**. See *Oracle Data Guard Protection Modes* for information on these options.
   - **Transport type**: The redo transport type used for this Oracle Data Guard association.
   - **Database admin password**: Enter the ADMIN password for the database.
7. Click **Save**.

Related Topics

- Oracle Data Guard Protection Modes

To perform a database failover

You initiate a failover operation by using the Data Guard association of the standby database.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose the **Compartment** that contains the Oracle Exadata Database Service on Exascale Infrastructure instance with the database for which you want to enable Oracle Data Guard.
3. Navigate to the cloud VM cluster that contains the Data Guard association:

   Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.
4. Under **Resources**, click **Data Guard Associations**.
5. For the Data Guard association on which you want to perform a failover, click **Failover**.
6. In the **Failover Database** dialog box, enter the database admin password, and then click **OK**.

   This database should now assume the role of the primary, and the old primary's role should display as **Disabled Standby**.

To reinstate a database

After you fail over a primary database to its standby, the standby assumes the primary role and the old primary is identified as a disabled standby. After you correct the cause of failure, you can reinstate the failed database as a functioning standby for the current primary by using its Data Guard association.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**
2. Choose the **Compartment** that contains the Oracle Exadata Database Service on Exascale Infrastructure with the database for which you want to enable Oracle Data Guard.
3. Navigate to the cloud VM cluster or DB system that contains the Data Guard association:

   Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster you want to access and click its highlighted name to view the details page for the cluster.
4. Under **Resources**, click **Data Guard Associations**.
5. For the Data Guard association on which you want to reinstate this database, click the Actions icon (three dots), and then click **Reinstate**.
6. In the **Reinstate Database** dialog box, enter the database admin password, and then click **OK**.

   This database should now be reinstated as the standby in the Data Guard association.

To terminate a Data Guard association on an Oracle Exadata Database Service on Exascale Infrastructure instance

On an Oracle Exadata Database Service on Exascale Infrastructure instance, you remove a Data Guard association by terminating the standby database.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**.
2. Choose the **Compartment** that contains the Oracle Exadata Database Service on Exascale Infrastructure VM cluster with the database for which you want to enable Oracle Data Guard.
3. Navigate to the cloud VM cluster that contains the standby database:

   Under **Oracle Exadata Database Service on Exascale Infrastructure**, click **Exadata VM Clusters**. In the list of VM clusters, find the VM cluster that you want to access, and click its highlighted name to view the details page for the cluster.
4. For the standby database you want to terminate, click the Actions icon (⋮), and then click **Terminate**.
5. In the **Terminate Database** dialog box, enter the name of the database, and then click **OK**.

Using the API to manage Data Guard associations

Use these API operations to manage Data Guard associations on an Oracle Exadata Database Service on Exascale Infrastructure instance:

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

- CreateDataGuardAssociation
- ListDataGuardAssociations
- GetDataGuardAssociation
- UpdateDataGuardAssociation
- SwitchoverDataGuardAssociation
- FailoverDataGuardAssociation
- ReinstateDataGuardAssociation
- DeleteDatabase - To terminate an Oracle Exadata Database Service on Exascale Infrastructure instance Data Guard association, you delete the standby database.

For the complete list of APIs for the Database service, see Database Service API.

Configure Oracle Database Features for Oracle Exadata Database Service on Exascale Infrastructure

Learn how to configure Oracle Multitenant, tablespace encryption, and other options for your Oracle Exadata Database Service on Exascale Infrastructure instance.

- Using Oracle Multitenant on an Oracle Exadata Database Service on Exascale Infrastructure Instance
  Learn about requirements for different features when using Multitenant environments in Oracle Exadata Database Service on Exascale Infrastructure.
- Managing Tablespace Encryption
  Learn about how tablespace encryption is implemented in Oracle Exadata Database Service on Exascale Infrastructure

Using Oracle Multitenant on an Oracle Exadata Database Service on Exascale Infrastructure Instance

Learn about requirements for different features when using Multitenant environments in Oracle Exadata Database Service on Exascale Infrastructure.

When you create an Oracle Exadata Database Service on Exascale Infrastructure Instance, an Oracle Multitenant environment is created.

The multitenant architecture enables Oracle Database to function as a multitenant container database (CDB) that includes zero, one, or many pluggable databases (PDBs). A PDB is a portable collection of schemas, schema objects, and non-schema objects that appears to an Oracle Net Services client as a non-CDB.

To use Oracle Transparent Data Encryption (TDE) in a pluggable database (PDB), you must create and activate a master encryption key for the PDB.

In a multitenant environment, each PDB has its own master encryption key which is stored in a single keystore used by all containers.

You must export and import the master encryption key for any encrypted PDBs you plug into your Oracle Exadata Database Service on Exascale Infrastructure Instance CDB.

If your source PDB is encrypted, you must export the master encryption key and then import it.

You can export and import all of the TDE master encryption keys that belong to the PDB by exporting and importing the TDE master encryption keys from within a PDB. Export and import of TDE master encryption keys support the PDB unplug and plug operations. During a PDB unplug and plug, all of the TDE master encryption keys that belong to a PDB, as well as the metadata, are involved.

See "Using Transparent Data Encryption with Other Oracle Features" in *Oracle Database Advanced Security Guide*.

See "ADMINISTER KEY MANAGEMENT" in *Oracle Database SQL Language Reference*..

- To determine if you need to create and activate an encryption key for the PDB
- To create and activate the master encryption key in a PDB
- To export and import a master encryption key

Related Topics

- Using Transparent Data Encryption with Other Oracle Features in *Oracle Database Advanced Security Guide*
- ADMINISTER KEY MANAGEMENT in *Oracle Database SQL Language Reference*

To determine if you need to create and activate an encryption key for the PDB

1. Invoke SQL*Plus and log in to the database as the `SYS` user with `SYSDBA` privileges.
2. Set the container to the PDB:

   ```
   SQL> ALTER SESSION SET CONTAINER = pdb;
   ```

3. Query `V$ENCRYPTION_WALLET` as follows:

   ```
   SQL> SELECT wrl_parameter, status, wallet_type FROM v$encryption_wallet;
   ```

   If the `STATUS` column contains a value of `OPEN_NO_MASTER_KEY`, you need to create and activate the master encryption key.

To create and activate the master encryption key in a PDB

1. Set the container to the PDB:

   ```
   SQL> ALTER SESSION SET CONTAINER = pdb;
   ```

2. Create and activate a master encryption key in the PDB by executing the following command:

   ```
   SQL> ADMINISTER KEY MANAGEMENT SET KEY USING TAG 'tag' FORCE KEYSTORE
   IDENTIFIED BY keystore-password WITH BACKUP USING 'backup_identifier';
   ```

   In the previous command:

   - `keystore-password` is the keystore password. By default, the keystore password is set to the value of the administration password that is specified when the database is created.
   - The optional `USING TAG 'tag'` clause can be used to associate a tag with the new master encryption key.
   - The `WITH BACKUP` clause, and the optional `USING 'backup_identifier'` clause, can be used to create a backup of the keystore before the new master encryption key is created.

   See also `ADMINISTER KEY MANAGEMENT` in *Oracle Database SQL Language Reference for Release*19, 18 or 12.2.

   > **Note:**
   >
   > To enable key management operations while the keystore is in use, Oracle Database 12c Release 2, and later, includes the `FORCE KEYSTORE` option to the `ADMINISTER KEY MANAGEMENT` command. This option is also available for Oracle Database 12c Release 1 with the October 2017, or later, bundle patch.
   >
   > If your Oracle Database 12c Release 1 database does not have the October 2017, or later, bundle patch installed, you can perform the following alternative steps:
   >
   > a. Close the keystore.
   >
   > b. Open the password-based keystore.
   >
   > c. Create and activate a master encryption key in the PDB by using `ADMINISTER KEY MANAGEMENT` without the `FORCE KEYSTORE` option.
   >
   > d. Update the auto-login keystore by using `ADMINISTER KEY MANAGEMENT` with the `CREATE AUTO_LOGIN KEYSTORE FROM KEYSTORE` option.

3. Query `V$ENCRYPTION_WALLET` again to verify that the `STATUS` column is set to `OPEN`:

   ```
   SQL> SELECT wrl_parameter, status, wallet_type FROM v$encryption_wallet;
   ```

4. Query `V$INSTANCE` and take note of the value in the `HOST_NAME` column, which identifies the database server that contains the newly updated keystore files:

   ```
   SQL> SELECT host_name FROM v$instance;
   ```

5. Copy the updated keystore files to all of the other database servers.

   To distribute the updated keystore, you must perform the following actions on each database server that does not contain the updated keystore files:

a. Connect to the root container and query V$ENCRYPTION_WALLET. Take note of the keystore location contained in the WRL_PARAMETER column:

```
SQL> SELECT wrl_parameter, status FROM v$encryption_wallet;
```

b. Copy the updated keystore files.

   You must copy all of the updated keystore files from a database server that is already updated. Use the keystore location observed in the WRL_PARAMETER column of V$ENCRYPTION_WALLET.

Open the updated keystore:

```
SQL> ADMINISTER KEY MANAGEMENT SET KEYSTORE open FORCE KEYSTORE IDENTIFIED
BY keystore-password CONTAINER=all;
```

> **Note:**
>
> To enable key management operations while the keystore is in use, Oracle Database 12c Release 2, and later, includes the FORCE KEYSTORE option to the ADMINISTER KEY MANAGEMENT command. This option is also available for Oracle Database 12c Release 1 with the October 2017, or later, bundle patch.
>
> If your Oracle Database 12c Release 1 database does not have the October 2017, or later, bundle patch installed, you can perform the following alternative steps:
>
> a. Close the keystore before copying the updated keystore files.
>
> b. Copy the updated keystore files.
>
> c. Open the updated keystore by using ADMINISTER KEY MANAGEMENT without the FORCE KEYSTORE option.

6. Query GV$ENCRYPTION_WALLET to verify that the STATUS column is set to OPEN across all of the database instances:

```
SQL> SELECT wrl_parameter, status, wallet_type FROM gv$encryption_wallet;
```

To export and import a master encryption key

1. Export the master encryption key.

   a. Invoke SQL*Plus and log in to the PDB.

   b. Execute the following command:

```
SQL> ADMINISTER KEY MANAGEMENT EXPORT ENCRYPTION KEYS WITH SECRET
"secret" TO 'filename' IDENTIFIED BY keystore-password;
```

2. Import the master encryption key.

   a. Invoke SQL*Plus and log in to the PDB.

b. Execute the following command:

```
SQL> ADMINISTER KEY MANAGEMENT IMPORT ENCRYPTION KEYS WITH SECRET
"secret" FROM 'filename' IDENTIFIED BY keystore-password;
```

Managing Tablespace Encryption

Learn about how tablespace encryption is implemented in Oracle Exadata Database Service on Exascale Infrastructure

By default, all new tablespaces that you create in an Exadata database are encrypted.

However, the tablespaces that are initially created when the database is created may not be encrypted by default.

- For databases that use Oracle Database 12c Release 2 or later, only the `USERS` tablespaces initially created when the database was created are encrypted. No other tablespaces are encrypted including the non-`USERS` tablespaces in:
  - The root container (`CDB$ROOT`).
  - The seed pluggable database (`PDB$SEED`).
  - The first PDB, which is created when the database is created.
- For databases that use Oracle Database 12c Release 1 or Oracle Database 11g, none of the tablespaces initially created when the database was created are encrypted.

For further information about the implementation of tablespace encryption in Exadata, along with how it impacts various deployment scenarios, see:

Oracle Database Tablespace Encryption Behavior in Oracle Cloud (Doc ID 2359020.1).

Creating Encrypted Tablespaces

User-created tablespaces are encrypted by default.

By default, any new tablespaces created by using the `SQL CREATE TABLESPACE` command are encrypted with the AES128 encryption algorithm. You do not need to include the `USING 'encrypt_algorithm'` clause to use the default encryption.

You can specify another supported algorithm by including the USING 'encrypt_algorithm' clause in the CREATE TABLESPACE command. Supported algorithms are AES256, AES192, AES128, and 3DES168.

Managing Tablespace Encryption

You can manage the software keystore (known as an Oracle wallet in Oracle Database 11g), the master encryption key, and control whether encryption is enabled by default.

Managing the Master Encryption Key

Tablespace encryption uses a two-tiered, key-based architecture to transparently encrypt (and decrypt) tablespaces. The master encryption key is stored in an external security module (software keystore). This master encryption key is used to encrypt the tablespace encryption key, which in turn is used to encrypt and decrypt data in the tablespace.

When a database is created on an Exadata Cloud Service instance, a local software keystore is created. The keystore is local to the compute nodes and is protected by the administration password specified during the database creation process. The auto-login software keystore is automatically opened when the database is started.

You can change (rotate) the master encryption key by using the `ADMINISTER KEY MANAGEMENT SQL` statement. For example:

```
SQL> ADMINISTER KEY MANAGEMENT SET ENCRYPTION KEY USING TAG 'tag'
IDENTIFIED BY password WITH BACKUP USING 'backup';


keystore altered.
```

See "Managing the TDE Master Encryption Key" in *Oracle Database Advanced Security Guide*.

Controlling Default Tablespace Encryption

The `ENCRYPT_NEW_TABLESPACES` initialization parameter controls the default encryption of new tablespaces. In Exadata databases, this parameter is set to `CLOUD_ONLY` by default.

Values of this parameter are as follows.

Value	Description
`ALWAYS`	During creation, tablespaces are transparently encrypted with the AES128 algorithm unless a different algorithm is specified in the `ENCRYPTION` clause.
`CLOUD_ONLY`	Tablespaces created in an Exadata database are transparently encrypted with the AES128 algorithm unless a different algorithm is specified in the `ENCRYPTION` clause. For non-cloud databases, tablespaces are only encrypted if the `ENCRYPTION` clause is specified. `ENCRYPTION` is the default value.
`DDL`	During creation, tablespaces are not transparently encrypted by default, and are only encrypted if the `ENCRYPTION` clause is specified.

> **Note:**
>
> With Oracle Database 12c Release 2 (12.2), or later, you can no longer create an unencrypted tablespace in an Exadata database. An error message is returned if you set `ENCRYPT_NEW_TABLESPACES` to `DDL` and issue a `CREATE TABLESPACE` command without specifying an `ENCRYPTION` clause.

Related Topics

- *Oracle Database Advanced Security Guide* Release 19c
- *Oracle Database Advanced Security Guide* Release 18c
- *Oracle Database Advanced Security Guide* Release 12c (12.2)

Migrate to Oracle Exadata Database Service on Exascale Infrastructure

For general guidance on methods and tools to migrate databases to Oracle Cloud Infrastructure database services, including Oracle Exadata Database Service on Exascale Infrastructure see "Migrating Databases to the Cloud".

A recommended approach for migrating to Oracle Exadata Database Service on Exascale Infrastructure is using Zero Downtime Migration

Related Topics

- Migrating Databases to the Cloud

Connect Identity and Access Management (IAM) Users to Oracle Exadata Database Service on Exascale Infrastructure

You can configure Exadata Database Service on Exascale Infrastructure to use Oracle Cloud Infrastructure Identity and Access Management (IAM) authentication and authorization to allow IAM users to access an Oracle Database with IAM credentials.

- Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Authentication with Oracle Database
  Learn to enable an Oracle Database instance on Oracle Exadata Database Service on Exascale Infrastructure to allow user access with an Oracle Cloud Infrastructure IAM database password (using a password verifier), or SSO tokens.
- Prerequisites for Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Authentication on Oracle Database
  Review the prerequisites for Identity and Access Management (IAM) authentication on an Oracle Database.
- Enable, Disable, and Re-enable Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Authentication on Oracle Database
  Learn to enable, disable, and re-enable Identity and Access Management (IAM) Authentication on Oracle Database.
- Manage Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Groups and Policies, Users, Roles, and Database Passwords
  Your Oracle Exadata Database Service on Exascale Infrastructure system provides several different methods of service management.
- Configuring Client Connection
  Configure various clients to use IAM authentication.

Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Authentication with Oracle Database

Learn to enable an Oracle Database instance on Oracle Exadata Database Service on Exascale Infrastructure to allow user access with an Oracle Cloud Infrastructure IAM database password (using a password verifier), or SSO tokens.

- About Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Authentication with Oracle Database
  IAM users can connect to the database instance by using either an IAM database password verifier or an IAM token.
- Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Database Password Verifier Authentication
  You can enable an Oracle Database instance to allow user access with an Oracle Cloud Infrastructure IAM database password (using a password verifier).
- Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) SSO Token Based Authentication
  For IAM token access to the database, the client application or tool requests a database token from IAM for the IAM user.

About Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Authentication with Oracle Database

IAM users can connect to the database instance by using either an IAM database password verifier or an IAM token.

Using the IAM database password verifier is similar to the database password authentication process. However, instead of the password verifier (encrypted hash of the password) being stored in the database, the verifier is instead stored as part of the OCI IAM user profile.

The second connection method, the use of an IAM token for the database, is more modern. The use of token-based access is a better fit for Cloud resources such as Oracle Databases in the Exadata Cloud Infrastructure. The token is based on the strength that the IAM endpoint can enforce. This can be multi-factor authentication, which is stronger than the use of passwords alone. Another benefit of using tokens is that the password verifier (which is considered sensitive) is never stored or available in memory.

> **Note:**
>
> Oracle Exadata Database Service on Exascale Infrastructure integration with Oracle Cloud Infrastructure IAM is supported in commercial tenancies with identity domains as well as the legacy Oracle Cloud Infrastructure IAM, which does not include identity domains. Oracle Cloud Infrastructure IAM with identity domains was introduced with new OCI tenancies created after November 8, 2021. Only the default domain OCI IAM users are supported with the new identity domains.
> This feature is only available with the DBRU 19.17 update (Oct 2022 patch) and higher. This is not available on Oracle Database release 21c.

Oracle Cloud Infrastructure IAM integration with Oracle Exadata Database Service on Dedicated Infrastructure supports the following:

- *Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Database Password Verifier Authentication*
- *Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) SSO Token Based Authentication*

See *Authenticating and Authorizing IAM Users for Oracle DBaaS Databases* for complete details about the architecture for using IAM users on Oracle Exadata Database Service on Dedicated Infrastructure.

Related Topics

- Authenticating and Authorizing IAM Users for Oracle DBaaS Databases
- Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Database Password Verifier Authentication
  You can enable an Oracle Database instance to allow user access with an Oracle Cloud Infrastructure IAM database password (using a password verifier).
- Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) SSO Token Based Authentication
  For IAM token access to the database, the client application or tool requests a database token from IAM for the IAM user.

Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Database Password Verifier Authentication

You can enable an Oracle Database instance to allow user access with an Oracle Cloud Infrastructure IAM database password (using a password verifier).

> **Note:**
>
> The user can log in with the existing supported database clients as long as the client is at least Oracle Database release 12c.

The IAM user enters the IAM user name and IAM database password (not the OCI Console password) using any currently supported database client. The only constraint is that the database client version be either Oracle Database release 12.1.0.2 or later to use Oracle Database 12c passwords. The database client must be able to use the 12c password verifier. Using the 11g verifier encryption is not supported with IAM. No special client or tool configuration is needed for the IAM user to connect to the database.

For more information about managing IAM database password, see *Managing User Credentials*.

Related Topics

- Managing User Credentials

Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) SSO Token Based Authentication

For IAM token access to the database, the client application or tool requests a database token from IAM for the IAM user.

The client application will pass the database token directly to the database client through the database client API.

If the application or tool has not been updated to request an IAM token, then the IAM user can use OCI CLI to request and store the database token. You can request a database access token (`db-token`) using the following credentials:

- Security tokens (with IAM authentication), delegation tokens (in the OCI cloud shell) and `API-keys`, which are credentials that represent the IAM user to enable the authentication

- Instance principal tokens, which enable instances to be authorized actors (or principals) to perform actions on OCI resources after authentication
- Resource principal token, which is a credential that enables the application to authenticate itself to other OCI services
- Using an IAM user name and IAM database password (can only be requested by database client)

When the IAM users logs into the client with a slash `/` login and the `OCI_IAM` parameter is configured (`sqlnet.ora`, `tnsnames.ora`, or as part of a connect string), then the database client retrieves the database token from a file. If the IAM user submits a user name and password, the connection will use the IAM database verifier access described for client connections that use IAM database password verifiers. If the parameter `PASSWORD_AUTH=OCI_TOKEN`, then the database driver will instead use the username and password to connect directly to IAM and request a database token. The instructions in this guide show how to use the OCI CLI as a helper for the database token. If the application or tool has been updated to work with IAM, then follow the instructions for the application or tool. Some common use cases include the following: SQL*Plus on-premises, SQLcl on-premises, SQL*Plus in Cloud Shell, or applications that use SEP wallets.

There are several ways a database client can obtain an IAM database token:

- A client application or tool can request the database token from IAM for the user and can pass the database token through the client API. Using the API to send the token overrides other settings in the database client. Using IAM tokens requires the latest Oracle Database client 19c (at least 19.16). Some earlier clients (19c and 21c) provide a limited set of capabilities for token access. Oracle Database client 21c does not fully support the IAM token access feature:
  - JDBC-thin on all platforms
    - See *Support for IAM Token-Based Authentication* and *JDBC and UCP Downloads* for more information.
  - SQL*Plus and Oracle Instant Client OCI-C on Linux:

    See *Identity and Access Management (IAM) Token -Based Authentication* for more information
  - Oracle Data Provider for .NET (ODP.NET) Core: .NET clients (latest version of Linux or Windows). .NET software components are available as a free download from the following sites:
    - *Oracle Data access Components - .NET Downloads*
    - *NuGet Gallery*
    - *Visual Studio Code Market Place*
- If the application or tool does not support requesting an IAM database token through the client API, the IAM user can first use the Oracle Cloud Infrastructure command line interface (CLI) to retrieve the IAM database token and save it in a file location. For example, to use SQL*Plus and other applications and tools using this connection method, you first obtain the database token using the Oracle Cloud Infrastructure (OCI) Command Line Interface (CLI). For more information, see db-token get. If the database client is configured for IAM database tokens, when a user logs in with the slash login form, the database driver uses the IAM database token that has been saved in default or specified file location.
- A client application or tool can use an Oracle Cloud Infrastructure IAM instance principal or resource principal to get an IAM database token and use the IAM database token to

authenticate itself to an Oracle Database instance. For more information, see *Mapping Instance and Resource Principals*.

- IAM users and OCI applications can request a database token from IAM with several methods, including using an API key. See *Configuring a Client Connection for SQL*Plus That Uses an IAM Token* for an example. See *Authenticating and Authorizing IAM Users for Oracle DBaaS Databases* for a description of other methods such as using a delegation token within an OCI cloud shell.

> **Note:**
>
> If your database is in Restricted Mode, only DBAs with the `RESTRICTED SESSION` privilege can connect to the database.

If a user enters a username/password to login, then the database driver uses the password verifier method to access the database. If the parameter `PASSWORD_AUTH=OCI_TOKEN`, then the database driver will instead user the username and password to connect directly to IAM and request a database token.

Related Topics

- Support for IAM Token-Based Authentication
- JDBC and UCP Downloads
- Identity and Access Management (IAM) Token-Based Authentication
- db-token get
- Oracle Data access Components - .NET Downloads
- NuGet Gallery
- Visual Studio Code Marketplace
- Mapping Instance and Resource Principals
- Configuring a Client Connection for SQL*Plus That Uses an IAM Token
- Authenticating and Authorizing IAM Users for Oracle DBaaS Databases

Prerequisites for Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Authentication on Oracle Database

Review the prerequisites for Identity and Access Management (IAM) authentication on an Oracle Database.

- Prerequisites for IAM Authentication on Oracle Database
  Before using IAM authentication on databases in the Exadata Cloud Infrastructure, you must use the Networking service to add a service gateway, a route rule, and an egress security rule to the Virtual Cloud Network (VCN) and subnets where your database resources reside.
- Disable External Authentication Scheme
  Review the prerequisites for enabling IAM user access to Oracle Database.
- Configure TLS to Use IAM Tokens
  When sending IAM tokens from the database client to the database server, a TLS connection must be established. The TLS wallet with the database certificate for the

ExaDB-D service instance must be stored under the `WALLET_ROOT` location. Create a tls directory so it looks like: `WALLET_ROOT/<PDB GUID>/tls`.

Prerequisites for IAM Authentication on Oracle Database

Before using IAM authentication on databases in the Exadata Cloud Infrastructure, you must use the Networking service to add a service gateway, a route rule, and an egress security rule to the Virtual Cloud Network (VCN) and subnets where your database resources reside.

1. Create a service gateway in the VCN where your database resources reside by following the instructions in *Task 1: Create the service gateway* in OCI documentation.
2. After creating the service gateway, add a route rule and an egress security rule to each subnet (in the VCN) where the database resources reside so that these resources can use the gateway to use IAM authentication:
   a. Go to the **Subnet Details** page for the subnet.
   b. In the **Subnet Information** tab, click the name of the subnet's Route Table to display its **Route Table Details** page.
   c. In the table of existing Route Rules, check whether there is already a rule with the following characteristics:
      - **Destination**: All IAD Services In Oracle Services Network
      - **Target Type**: Service Gateway
      - **Target**: The name of the service gateway you just created in the VCN

      If such a rule does not exist, click **Add Route Rules** and add a route rule with these characteristics.
   d. Return to the Subnet Details page for the subnet.
   e. In the subnet's Security Lists table, click the name of the subnet's security list to display its Security List Details page.
   f. In the side menu, under **Resources**, click **Egress Rules**.
   g. In the table of existing Egress Rules, check whether there is already a rule with the following characteristics:
      - **Stateless**: No
      - **Destination**: All IAD Services In Oracle Services Network
      - **IP Protocol**: TCP
      - **Source Port Range**: All
      - **Destination Port Range**: 443
   h. If such a rule does not exist, click **Add Egress Rules** and add an egress rule with these characteristics.

Related Topics

- Task 1: Create the service gateway

Disable External Authentication Scheme

Review the prerequisites for enabling IAM user access to Oracle Database.

If the database is enabled for another external authentication scheme, verify that you want to use IAM on the Oracle Database instance. There can only be one external authentication scheme enabled at any given time.

If you want to use IAM and another external authentication scheme is enabled, you must first disable the other external authentication scheme.

Configure TLS to Use IAM Tokens

When sending IAM tokens from the database client to the database server, a TLS connection must be established. The TLS wallet with the database certificate for the ExaDB-D service instance must be stored under the `WALLET_ROOT` location. Create a tls directory so it looks like: `WALLET_ROOT/<PDB GUID>/tls`.

When configuring TLS between the database client and server there are several options to consider.

- Using a self-signed database server certificate vs a database server certificate signed by a commonly known certificate authority
- One-way TLS (TLS) vs Mutual or two-way TLS (mTLS)
- Client with or without a wallet

Self-Signed Certificate

Using a self-signed certificate is a common practice for internally facing IT resources since you can create these yourself and it's free. The resource (in our case, the database server) will have a self-signed certificate to authenticate itself to the database client. The self-signed certificate and root certificate will be stored in the database server wallet. For the database client to be able to recognize the database server certificate, a copy of the root certificate will also be needed on the client. This self-created root certificate can be stored in a client-side wallet or installed in the client system default certificate store (Windows and Linux only). When the session is established, the database client will check to see that the certificate sent over by the database server has been signed by the same root certificate.

A Well-Known Certificate Authority

Using a commonly known root certificate authority has some advantages in that the root certificate is most likely already stored in the client system default certificate store. There is no extra step for the client to store the root certificate if it is a common root certificate. The disadvantage is that this normally has a cost associated with it.

One-Way TLS

In the standard TLS session, only the server provides a certificate to the client to authenticate itself. The client doesn't need to have a separate client certificate to authenticate itself to the server (similar to how HTTPS sessions are established). While the database requires a wallet to store the server certificate, the only thing the client needs to have is the root certificate used to sign the server certificate.

Two-Way TLS (also called Mutual TLS, mTLS)

In mTLS, both the client and server have identity certificates that are presented to each other. In most cases, the same root certificate will have signed both of these certificates so the same root certificate can be used with the database server and client to authenticate the other certificate. mTLS is sometimes used to authenticate the user since the user identity is authenticated by the database server through the certificate. This is not necessary for passing IAM tokens but can be used when passing IAM tokens.

Client with a Wallet

A client wallet is mandatory when using mTLS to store the client certificate. However, the root certificate can be stored either in the same wallet or in the system default certificate store.

A Client without a Wallet

Clients can be configured without a wallet when using TLS under these conditions: 1) One-way TLS is being configured where the client does not have its own certificate and 2) the root certificate that signed the database server certificate is stored in the system default certificate store. The root certificate would most likely already be there if the server certificate is signed by a common certificate authority. If it's a self-signed certificate, then the root certificate would need to be installed in the system default certificate store to avoid using a client wallet.

For details on how to configure TLS between the database client and database server including the options described above, see *Configuring Transport Layer Security Authentication* in the *Oracle Database Security Guide*.

If you choose to use self-signed certificates and for additional wallet related tasks, see *Managing Public Key Infrastructure (PKI) Elements* in the *Oracle Database Security Guide*.

Related Topics

- Configuring Transport Layer Security Authentication
- Managing Public Key Infrastructure (PKI) Elements

Enable, Disable, and Re-enable Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Authentication on Oracle Database

Learn to enable, disable, and re-enable Identity and Access Management (IAM) Authentication on Oracle Database.

- Enable Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Authentication on Oracle Database
  Review the steps to enable or re-enable IAM user access to Oracle Database.
- Disable Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Authentication on Oracle Database
  Describes the steps to disable IAM external authentication user access for Oracle Database.
- Using Oracle Database Tools with Identity and Access Management (IAM) Authentication
  Review the notes for using Oracle Database tools with IAM authentication enabled.

Enable Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Authentication on Oracle Database

Review the steps to enable or re-enable IAM user access to Oracle Database.

> **Note:**
>
> Oracle Exadata Database Service on Dedicated Infrastructure integration with Oracle Cloud Infrastructure IAM is supported in commercial tenancies with identity domains as well as the legacy Oracle Cloud Infrastructure IAM, which does not include identity domains. Oracle Cloud Infrastructure IAM with identity domains was introduced with new OCI tenancies created after November 8, 2021. Only default domain OCI IAM users are supported with the new identity domains.
> This feature is only available with the DBRU 19.17 update (Oct 2022 patch) and higher. This is not available on Oracle Database release 21c.

1. Perform the prerequisites for IAM authorization and authentication on Oracle Database.See *Prerequisites for Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Authentication on Oracle Database* for more information.
2. Enable Oracle Cloud Infrastructure (IAM) Authentication and Authorization using the `ALTER SYSTEM` command.

   ```
   ALTER SYSTEM SET IDENTITY_PROVIDER_TYPE=OCI_IAM SCOPE=BOTH;
   ```

3. Verify the value of `IDENTITY_PROVIDER_TYPE` system parameter.

   ```
   SELECT NAME, VALUE FROM V$PARAMETER WHERE NAME='identity_provider_type';



   NAME                      VALUE
   ----------------------    -------
   identity_provider_type    OCI_IAM
   ```

Related Topics

- Prerequisites for Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Authentication on Oracle Database
  Review the prerequisites for Identity and Access Management (IAM) authentication on an Oracle Database.
- Disable External Authentication Scheme
  Review the prerequisites for enabling IAM user access to Oracle Database.

Disable Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Authentication on Oracle Database

Describes the steps to disable IAM external authentication user access for Oracle Database.

To disable IAM user access on your Oracle Database instance:

1. Disable IAM integration using the `ALTER SYSTEM` command.

   ```
   ALTER SYSTEM RESET IDENTITY_PROVIDER_TYPE SCOPE=BOTH;
   ```

2. If you also want to remove the IAM policy to allow database access, you may need to review and either modify or remove the IAM groups and the policies you set up to allow access to the database by IAM users.

Using Oracle Database Tools with Identity and Access Management (IAM) Authentication

Review the notes for using Oracle Database tools with IAM authentication enabled.

- Oracle APEX is not supported for IAM users with Oracle Database.
- Database Actions is not supported for IAM users with Oracle Database. See *Provide Database Actions Access to Database Users* for information on using regular database users with Oracle Database.
- Oracle Machine Learning Notebooks and other components are not supported for IAM Authorized users with Oracle Database. See *Add Existing Database User Account to Oracle Machine Learning Components* for information on using regular database users with Oracle Database.

Related Topics

- Provide Database Actions Access to Database Users
- Add Existing Database User Account to Oracle Machine Learning Components

Manage Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Groups and Policies, Users, Roles, and Database Passwords

Your Oracle Exadata Database Service on Exascale Infrastructure system provides several different methods of service management.

- Create Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Groups and Policies for IAM Users
  Review the steps to write policy statements for an IAM group to enable IAM user access to Oracle Cloud Infrastructure resources, specifically Oracle Database instances using IAM database tokens.
- Authorize Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Users on Oracle Database
  Review the steps to authorize IAM users on an Oracle Database instance.
- To Exclusively Map a Local IAM User to an Oracle Database Global User
  You can map a local IAM user exclusively to an Oracle Database global user.
- Add Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Roles on Oracle Database
  Optionally, create global roles to provide additional database roles and privileges to IAM users when multiple IAM users are mapped to the same shared global user.
- Create Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Database Password for IAM Users
  To add an IAM user and allow the IAM user to login to Oracle Database by supplying a username and password, you must create an IAM database password.

Create Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Groups and Policies for IAM Users

Review the steps to write policy statements for an IAM group to enable IAM user access to Oracle Cloud Infrastructure resources, specifically Oracle Database instances using IAM database tokens.

A policy is a group of statements that specifies who can access particular resources, and how. Access can be granted for the entire tenancy, databases in a compartment, or individual databases. This means you write a policy statement that gives a specific group a specific type of access to a specific type of resource within a specific compartment.

Note: Defining a policy is required to use IAM tokens to access Oracle Database. A policy is not required when using IAM database password verifiers to access Oracle Database.

1. Create an IAM group for IAM users that will access the database. Review OCI IAM documentation for creating groups and adding IAM users to a group.
   For example, create the group *DBUsers*. For more information, see *Managing Groups*.
2. Write policy statements to enable access to Oracle Cloud Infrastructure resources.
   a. In the Oracle Cloud Infrastructure console, click **Identity and Security**, and then click **Policies**.
   b. To write a policy, click **Create Policy**, and then enter a **Name** and a **Description.**
   c. Use the **Policy Builder** to create a policy. For example, to create a policy to allow users in IAM group DBUsers to access any Oracle Database in their tenancy:

      ```
      Allow group DBUsers to use database-connections in tenancy
      ```

      Where, `database-connections` is the OCI resource name to connect to the database. `Use` is the minimum verb to allow access to the database. Both `use` and `manage` can be used.

      For example to create a policy that limits members of *DBUsers* group to access Oracle Databases in the compartment *testing_compartment* only:

      ```
      allow group DBUsers to use database-connections in compartment
      testing_compartment
      ```

      For example, to create a policy that limits group access to a single database in a compartment:

      ```
      allow group DBUsers to use database-connections in compartment
      testing_compartment where target.database.id =
      'ocid1.database.oc1.iad.aaaabbbbcccc'
      ```

   d. Click **Create**.
      For more information about policies, see Managing Policies.

Notes for creating policies for use with IAM users on Oracle Database:

- Policies can allow IAM users to access Oracle Database instances across the entire tenancy, in a compartment, or can limit access to a single Oracle Database instance.
- You must use dynamic groups for Instance Principals and Resource Principals. You can create Dynamic Groups and reference dynamic groups in the policies you create to access Oracle Cloud Infrastructure. See *Accessing Cloud Resources by Configuring Policies and Roles and Managing Dynamic Groups* for details.

Related Topics

- Managing Groups
- Accessing Cloud Resources by Configuring Policies and Roles
- Managing Dynamic Groups

Authorize Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Users on Oracle Database

Review the steps to authorize IAM users on an Oracle Database instance.

To authorize IAM users to allow access to Oracle Database, map database global users to IAM groups or directly to IAM users with `CREATE USER` or `ALTER USER` statements with `IDENTIFIED GLOBALLY AS` clause.

The authorization of IAM users to an Oracle Database instance works by mapping IAM global users (schemas) to IAM users (exclusive mapping) or IAM groups (shared schema mapping).

To authorize IAM users on a database instance:

1. Log in as a user with DBA privileges to the database that is enabled to use IAM. A user with the DBA role will need the required `CREATE USER` and `ALTER USER` system privileges for these steps.
2. Create a mapping between the Oracle Database user (schema) with `CREATE USER` or `ALTER USER` statements and include the `IDENTIFIED GLOBALLY AS` clause, specifying the IAM group name. Use the following syntax to map a global user to an IAM group:

   ```
   CREATE USER global_user IDENTIFIED GLOBALLY AS
   'IAM_GROUP_NAME=IAM_GROUP_NAME';
   ```

   For example, to map an IAM group named db_sales_group to a shared database global user named sales_group:

   ```
   CREATE USER sales_group IDENTIFIED GLOBALLY AS
   'IAM_GROUP_NAME=db_sales_group';
   ```

   This creates a shared global user mapping. The mapping, with the global user `sales_group` is effective for all users in the IAM group. Thus, anyone in the `db_sales_group` can log in to the database using their IAM credentials through the shared mapping of the `sales_group` global user.

   If you want to create additional global user mappings for other IAM groups or users, follow these steps for each IAM group or user.

   > **Note:**
   >
   > Database users that are not `IDENTIFIED GLOBALLY` can continue to login as before, even when the Oracle Database is enabled for IAM authentication.

To Exclusively Map a Local IAM User to an Oracle Database Global User

You can map a local IAM user exclusively to an Oracle Database global user.

1. Log in as an user with DBA privileges to the database that is enabled to use IAM. A user with the DBA role has will need the required `CREATE USER` and `ALTER USER` system privileges that you need for these steps.

2. Create a mapping between the Oracle Database user (schema) with `CREATE USER` or `ALTER USER` statements and include the `IDENTIFIED GLOBALLY AS` clause, specifying the IAM local IAM user name. For example, to create a new database global user named `peter_fitch` and map this user to an existing local IAM user named `peterfitch`:

   ```
   CREATE USER peter_fitch IDENTIFIED GLOBALLY AS
   'IAM_PRINCIPAL_NAME=peterfitch'
   ```

You can use either instance principal or resource principal to retrieve database tokens to establish a connection from your application to an Oracle Database instance.

If you are using an instance principal or resource principal, you must map a dynamic group. Thus, you cannot exclusively map instance and resource principals. You only can map them through a shared mapping and putting the instance or resource instance in an IAM dynamic group

Add Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Roles on Oracle Database

Optionally, create global roles to provide additional database roles and privileges to IAM users when multiple IAM users are mapped to the same shared global user.

Creating global roles is optional, but useful when assigning users to a shared schema.

Use a global role to optionally differentiate users who use the same shared schema. For example, a set of users can all have the same shared schema and the shared schema could have the `CREATE SESSION` privilege. Then global roles can be used to provide differentiated privileges and roles assigned to different groups of users who all use the same shared schema.

Granting additional roles to IAM users in Oracle Database works by mapping Oracle Database global roles to IAM groups.

1. Log in as a user with DBA privileges to the database that is enabled to use IAM. A user with the DBA privileges `CREATE ROLE` and `ALTER ROLE` system privileges is needed for these steps.
2. Set database authorization for Oracle Database roles with `CREATE ROLE` or `ALTER ROLE` statements and include the `IDENTIFIED GLOBALLY AS` clause, specifying the IAM group name. Use the following syntax to map a global role to an IAM group:

   ```
   CREATE ROLE global_role IDENTIFIED GLOBALLY AS
   'IAM_GROUP_NAME=IAM_GROUP_of_WHICH_the_IAM_USER_IS_a_MEMBER';
   ```

   For example, to map an IAM group named ExporterGroup to a shared database global role named export_role:

   ```
   CREATE ROLE export_role IDENTIFIED GLOBALLY AS
   'IAM_GROUP_NAME=ExporterGroup';
   ```

3. Use the `GRANT` statements to grant the required privileges or other roles to the global role.

   ```
   GRANT CREATE SESSION TO export_role;
   GRANT DWROLE TO export_role;
   ```

4. If you want an existing database role to be associated with an IAM group, then use the `ALTER ROLE` statement to alter the existing database role to map the role to an IAM group. Use the following syntax to alter an existing database role to map it to an IAM group:

```
ALTER ROLE existing_database_role IDENTIFIED GLOBALLY AS
'IAM_GROUP_NAME=IAM_Group_Name';
```

Follow these steps for each IAM group to add additional global role mappings for other IAM groups.

Create Oracle Cloud Infrastructure (OCI) Identity and Access Management (IAM) Database Password for IAM Users

To add an IAM user and allow the IAM user to login to Oracle Database by supplying a username and password, you must create an IAM database password.

For more information, see *Working with IAM Database Passwords*.

Related Topics

- Working with IAM Database Passwords

Configuring Client Connection

Configure various clients to use IAM authentication.

- Configure a Client Connection for SQL*Plus that Uses an IAM Database Password Verifier
  You can configure SQL*Plus to use an IAM database password verifier.
- Configure Client Connection for SQL*Plus that Uses an IAM Token
  You can configure a client connection for SQL*Plus that uses an IAM token.
- Client Connections That Use a Token Requested by an IAM User Name and Database Password
  You can create a client connection that uses a token requested by an IAM user name and database password.
- Use Instance Principal to Access Database with IAM Authentication
  After the ADMIN user enables OCI IAM on the database, an application can access the database through an OCI IAM database token using an instance principal.
- Configure Proxy Authentication
  Proxy authentication allows an IAM user to proxy to a database schema for tasks such as application maintenance.
- Use Database Link with IAM Authenticated Users
  You can use a database link to connect from one database instance to another as an OCI IAM user.

Configure a Client Connection for SQL*Plus that Uses an IAM Database Password Verifier

You can configure SQL*Plus to use an IAM database password verifier.

As the IAM user, log in to the database by using the following syntax:

```
CONNECT user_name@db_connect_string
Enter password: password
```

In this specification, `user_name` is the IAM user name. There is a limit of 128 bytes for the combined `domain_name/user_name`.

The following example shows how IAM user `peter_fitch` can log in to a database instance.

```
sqlplus /nolog
connect peter_fitch@db_connect_string
Enter password: password
```

Some special characters will require double quotation marks around `user_name` and . For example:

```
"peter_fitch@example.com"@db_connect_string

"IAM database password"
```

Configure Client Connection for SQL*Plus that Uses an IAM Token

You can configure a client connection for SQL*Plus that uses an IAM token.

1. Ensure you have an IAM user account.
2. Check with an IAM administrator and the database administrator to ensure you have a policy allowing you to access the database in the compartment or your tenancy and that you are mapped to a global schema in the database.
3. If your application or tool does not support direct IAM integration, then download, install, and configure the OCI CLI. (See OCI Command Line Interface Quickstart.) Set up an API key as part of the OCI CLI configuration and select default values.
   a. Set up the API key access for the IAM user.
   b. Retrieve the `db-token`. For example:
      - Retrieve a `db-token` with an `API-key` using the OCI CLI:

        ```
        oci iam db-token get
        ```

      - Retrieve `db-token` with a security (or session) token:

        ```
        oci iam db-token get --auth security_token
        ```

      - Retrieve `db-token` with a delegation token: When you log in to the cloud shell, the delegation token is automatically generated and placed in the `/etc` directory. To get this token, execute the following command in the OCI CLI:

        ```
        oci iam db-token get
        ```

- Using an instance principal to retrieve a db-token using OCI CLI:

  ```
  oci iam db-token get --auth instance_principal
  ```

  If the security token has expired, a window will appear so the user can log in to OCI again. This generates the security token for the user. OCI CLI will use this refreshed token to get the db-token.

  See Required Keys and OCIDs for more information.

4. Ensure that you are using the latest release updates for the Oracle Database client releases 19c.
   This configuration only works with the Oracle Database client release 19c.

5. Follow the existing process to download the wallet from the database and then follow the directions for configuring it for use with SQL*Plus.

   a. Confirm that DN matching is enabled by looking for SSL_SERVER_DN_MATCH=ON in sqlnet.ora.

   b. Configure the database client to use the IAM token by adding TOKEN_AUTH=OCI_TOKEN to the sqlnet.ora file. Because you will be using the default locations for the database token file, you do not need to include the token location.

   The TOKEN_AUTH and TOKEN_LOCATION values in the tnsnames.ora connect strings take precedence over the sqlnet.ora settings for that connection. For example, for the connect string, assuming that the token is in the default location (~/.oci/db-token for Linux):

   ```
   (description=
     (retry_count=20)(retry_delay=3)
     (address=(protocol=tcps)(port=1522)
     (host=example.us-phoenix-1.oraclecloud.com))

   (connect_data=(service_name=aaabbbccc_exampledb_high.example.oraclecloud.co
   m))
     (security=(ssl_server_cert_dn="CN=example.uscom-east-1.oraclecloud.com,
        OU=Oracle BMCS US, O=Example Corporation,
        L=Redwood City, ST=California, C=US")
     (TOKEN_AUTH=OCI_TOKEN)))
   ```

After the connect string is updated with the TOKEN_AUTH parameter, the IAM user can log in to the database instance by running the following command to start SQL*Plus. You can include the connect descriptor itself or use the name of the descriptor from the tnsnames.ora file.

```
connect /@exampledb_high
```

Or:

```
connect /@(description=
  (retry_count=20)(retry_delay=3)
  (address=(protocol=tcps)(port=1522)
  (host=example.us-phoenix-1.oraclecloud.com))

(connect_data=(service_name=aaabbbccc_exampledb_high.example.oraclecloud.com))
  (security=(ssl_server_cert_dn="CN=example.uscom-east-1.oraclecloud.com,
     OU=Oracle BMCS US, O=Example Corporation,
```

```
     L=Redwood City, ST=California, C=US")
  (TOKEN_AUTH=OCI_TOKEN)))
```

The database client is already configured to get a `db-token` because `TOKEN_AUTH` has already been set, either through the `sqlnet.ora` file or in a connect string. The database client gets the `db-token` and signs it using the private key and then sends the token to the database. If an IAM user name and IAM database password are specified instead of slash `/`, then the database client will connect using the password instead of using the `db-token`.

Client Connections That Use a Token Requested by an IAM User Name and Database Password

You can create a client connection that uses a token requested by an IAM user name and database password.

- IAM users can connect to the Oracle DBaaS instance by using an IAM token that was retrieved using an IAM user name and IAM database password.
  For more information, see *About Client Connections That Use a Token Requested by an IAM User Name and Database Password*
- To set these parameters, you modify either the `sqlnet.ora` file or the `tnsnames.ora` file.
  For more information, see *Parameters to Set for Client Connections That Use a Token Requested by an IAM User Name and Database Password*
- You can configure the database client to retrieve the IAM database token using the provided IAM user name and IAM database password.
  For more information, see *Configuring the Database Client to Retrieve a Token Using an IAM User Name and Database Password*
- You can enable an IAM user name and a secure external password store (SEPS) to request the IAM database token.
  For more information, see *Configuring a Secure External Password Store Wallet to Retrieve an IAM Token*

Related Topics

- Client Connections That Use a Token Requested by an IAM User Name and Database Password
- About Client Connections That Use a Token Requested by an IAM User Name and Database Password
- Parameters to Set for Client Connections That Use a Token Requested by an IAM User Name and Database Password
- Configuring the Database Client to Retrieve a Token Using an IAM User Name and Database Password
- Configuring a Secure External Password Store Wallet to Retrieve an IAM Token

Use Instance Principal to Access Database with IAM Authentication

After the ADMIN user enables OCI IAM on the database, an application can access the database through an OCI IAM database token using an instance principal.

For more information, see *Accessing the Oracle Cloud Infrastructure API Using Instance Principals*.

For more Information, see *Accessing the Database Using an Instance Principal or a Resource Principal*.

Related Topics

- Accessing the Oracle Cloud Infrastructure API Using Instance Principals
- Accessing the Database Using an Instance Principal or a Resource Principal

Configure Proxy Authentication

Proxy authentication allows an IAM user to proxy to a database schema for tasks such as application maintenance.

Proxy authentication is typically used to authenticate the real user and then authorize them to use a database schema with the schema privileges and roles in order to manage an application. Alternatives such as sharing the application schema password are considered insecure and unable to audit which actual user performed an action.

A use case can be in an environment in which a named IAM user who is an application database administrator can authenticate by using their credentials and then proxy to a database schema user (for example, `hrapp`). This authentication enables the IAM administrator to use the `hrapp` privileges and roles as user `hrapp` in order to perform application maintenance, yet still use their IAM credentials for authentication. An application database administrator can sign in to the database and then proxy to an application schema to manage this schema.

You can configure proxy authentication for both the password authentication and token authentication methods.

Configuring Proxy Authentication for the IAM User

To configure proxy authentication for an IAM user, the IAM user must already have a mapping to a global schema (exclusive or shared mapping). A separate database schema for the IAM user to proxy to must also be available.

After you ensure that you have this type of user, alter the database user to allow the IAM user to proxy to it.

1. Log in to the database instance as a user who has the `ALTER USER` system privileges.
2. Grant permission for the IAM user to proxy to the local database user account. An IAM user cannot be referenced in the command so the proxy must be created between the database global user (mapped to the IAM user) and the target database user.In the following example, `hrapp` is the database schema to proxy to, and `peterfitch_schema` is the database global user exclusively mapped to user `peterfitch`.

   ```
   ALTER USER hrapp GRANT CONNECT THROUGH peterfitch_schema;
   ```

At this stage, the IAM user can log in to the database instance using the proxy. For example:

- To connect using a password verifier:

  ```
  CONNECT peterfitch[hrapp]@connect_string
  Enter password: password
  ```

- To connect using a token:

  ```
  CONNECT [hrapp]/@connect_string
  ```

Validating the IAM User Proxy Authentication

You can validate the IAM user proxy configuration for both password and token authentication methods.

1. Connect as the IAM user and proxied to the database user. Run the `SHOW USER` and `SELECT SYS_CONTEXT` commands.
   For example, suppose you want to check the proxy authentication of the IAM user *peterfitch* when they proxy to database user *hrapp*. You will need to connect to the database using the different types of authentication methods shown here, but the output of the commands that you execute will be the same for all types.

   - For password authentication:

     ```
     CONNECT peterfitch[hrapp]/password\!@connect_string SHOW USER;


     --The output should be USER is "HRAPP"
     SELECT SYS_CONTEXT('USERENV','AUTHENTICATION_METHOD') FROM DUAL;
     --The output should be "PASSWORD_GLOBAL"
     SELECT SYS_CONTEXT('USERENV','PROXY_USER') FROM DUAL;
     --The output should be "PETERFITCH_SCHEMA"
     SELECT SYS_CONTEXT('USERENV','CURRENT_USER') FROM DUAL;
     --The output should be "HRAPP"
     ```

   - For token authentication:

     ```
     CONNECT [hrapp]/@connect_string
     SHOW USER;


     --The output should be USER is "HRAPP "
     SELECT SYS_CONTEXT('USERENV','AUTHENTICATION_METHOD') FROM DUAL;
     --The output should be "TOKEN_GLOBAL"
     SELECT SYS_CONTEXT('USERENV','PROXY_USER') FROM DUAL;
     --The output should be "PETERFITCH_SCHEMA"
     SELECT SYS_CONTEXT('USERENV','CURRENT_USER') FROM DUAL;
     --The output should be "HRAPP"
     ```

Use Database Link with IAM Authenticated Users

You can use a database link to connect from one database instance to another as an OCI IAM user.

You can use either connected user or fixed user database link to connect to a database as an OCI IAM user.

> **Note:**
>
> Current user database link is not supported for connecting to a database in Exadata Cloud Infrastructure as an OCI IAM user.

- **Connected User Database Link**: For a connected user database link, an IAM user must be mapped to a schema in both the source and target databases connected by a database link. You can use a database password verifier or an IAM database token to use a connected user database link.

- **Fixed User Database Link**: A fixed user database link can be created using a database user or an IAM user. When using an IAM user as a fixed user database link, the IAM user must have a schema mapping in the target database. The IAM user for a database link can be configured with a password verifier only.

5

Reference Guides for Oracle Exadata Database Service on Exascale Infrastructure

Using the dbaascli Utility on Oracle Exadata Database Service on Exascale Infrastructure

Learn to use the `dbaascli` utility on Oracle Exadata Database Service on Exascale Infrastructure.

- About Using the dbaascli Utility on Oracle Exadata Database Service on Exascale Infrastructure
  You can use the `dbaascli` utility to perform various database lifecycle and administration operations on Oracle Exadata Database Service on Exascale Infrastructure
- Creating Databases Using dbaascli
  Using `dbaascli`, you can create an Oracle Database by first creating an Oracle Database home of desired version, followed by creating a database in that Oracle Database home
- Changing the Database Passwords
  To change the SYS password, or to change the TDE wallet password, use this procedure.
- Managing Oracle Exadata Database Service on Exascale Infrastructure Software Images Using the Dbaascli Utility
  You can list and download the Oracle database software images on an Oracle Exadata Database Service on Exascale Infrastructure instance, which can then be used for provisioning a database home.
- Collect Cloud Tooling Logs and Perform a Cloud Tooling Health Check Using dbaascli
  Using the dbaascli `diag` command allows you to collect Guest VM `dbaas` tooling logs for Exadata Database Service on Dedicated Infrastructure and Exadata Database Service on Cloud@Customer systems. You can use these logs to troubleshoot issues related to `dbaas` tooling.
- Updating Cloud Tooling Using dbaascli
  To update the cloud tooling release for Oracle Exadata Database Service on Exascale Infrastructure, complete this procedure.
- Creating a Duplicate Database
- dbaascli Command Reference
  You use `dbaascli` to create databases and integrate them with the cloud automation framework.

About Using the dbaascli Utility on Oracle Exadata Database Service on Exascale Infrastructure

You can use the `dbaascli` utility to perform various database lifecycle and administration operations on Oracle Exadata Database Service on Exascale Infrastructure

For example, with dbaascli, you can change the password of a database user, start a database, or manage pluggable databases (PDBs), and more.

You must use the Oracle Cloud Infrastructure console or command-line interface to scale resources. The capabilities of the `dbaascli` utility are in addition to, and separate from, the Console, API, or command-line interface (CLI). Unless specified differently, you need `root` access to `dbaascli` to run all administration commands.

To use the utility, you must be connected to an Oracle Exadata Database Service on Exascale Infrastructure virtual machine.

To get possible commands available with `dbaascli`, run `dbaascli --help`.

To get command-specific help, run `dbaascli` *`command`* `--help`. For example, `dbaascli database create --help`.

See *dbasscli Command Reference* in the document for commands and command specific information.

Creating Databases Using dbaascli

Using dbaascli, you can create an Oracle Database by first creating an Oracle Database home of desired version, followed by creating a database in that Oracle Database home

- Listing Available Software Images and Versions for Database and Grid Infrastructure
  To produce a list of available supported versions for patching, use the dbaascli cswlib showImages command.
- Creating Oracle Database Home
  To create an Oracle Database home of desired version, use the dbaascli dbhome create command.
- Creating Oracle Database In the Specified Oracle Database Home
  To create an Oracle Database in the specified Oracle Database home of desired version, use the dbaascli database create command.

Listing Available Software Images and Versions for Database and Grid Infrastructure

To produce a list of available supported versions for patching, use the dbaascli cswlib showImages command.

1. Connect to the virtual machine as the opc user.
   For detailed instructions, see *Connecting to a Virtual Machine with SSH*.
2. Start a root user command shell:

   ```
   sudo -s
   ```

3. Run the following command:

   ```
   dbaascli cswlib showImages --product database
   ```

   The command output lists the available database software images.

   ```
   dbaascli cswlib showImages --product grid
   ```

   The command output lists the available grid software images.
4. Exit the root user command shell:

   ```
   exit
   ```

   For more details on advanced supported options, see dbaascli cswlib showImages.

Example 5-1 dbaascli cswlib showImages

```
[root@dg11lrg1 dbhome_1]# dbaascli cswlib showImages
DBAAS CLI version <version>
Executing command cswlib
     showImagesJob id: 00e89b1a-1607-422c-a920-22f44bec1953Log file location:
     /var/opt/oracle/log/cswLib/showImages/dbaastools_2022-05-11_08-49-12-
AM_46941.log

############
```

```
List of Available Database Images
#############

17.IMAGE_TAG=18.17.0.0.0
   VERSION=18.17.0.0.0
   DESCRIPTION=18c JAN 2022 DB Image

18.IMAGE_TAG=19.10.0.0.0
   VERSION=19.10.0.0.0
   DESCRIPTION=19c JAN 2021 DB Image

19.IMAGE_TAG=19.11.0.0.0
   VERSION=19.11.0.0.0
   DESCRIPTION=19c APR 2021 DB Image

20.IMAGE_TAG=19.12.0.0.0
  VERSION=19.12.0.0.0
  DESCRIPTION=19c JUL 2021 DB Image

21.IMAGE_TAG=19.13.0.0.0
  VERSION=19.13.0.0.0
  DESCRIPTION=19c OCT 2021 DB Image

Images can be downloaded using their image tags. For details, see help using
'dbaascli cswlib download --help'.
dbaascli execution completed
```

Related Topics

- Connecting to a Virtual Machine with SSH
  You can connect to the virtual machines in an Oracle Exadata Database Service on Exascale Infrastructure system by using a Secure Shell (SSH) connection.
- dbaascli cswlib showImages
  To view the list of available Database and Grid Infrastructure images, use the `dbaascli cswlib showImages` command.

Creating Oracle Database Home

To create an Oracle Database home of desired version, use the `dbaascli dbhome create` command.

> **Note:**
>
> You can create an Oracle Database home with a specified Oracle home name. If you do not specify, then this is computed automatically (recommended).

1. Connect to the virtual machine as the `opc` user.
   For detailed instructions, see *Connecting to a Virtual Machine with SSH*.
2. Start a `root` user command shell:

   ```
   sudo -s
   ```

3. Run the following command:

   ```
   dbaascli dbhome create --version Oracle Home Version --imageTag image Tag
   Value
   ```

   Where:

   - `--version` specifies the Oracle Database version
   - `--imageTag` specifies the Image Tag of the image to be used

   For example:

   ```
   dbaascli dbhome create --version 19.9.0.0.0
   ```

   > **Note:**
   >
   > Specifying `imageTag` is optional. To view the Image Tags, refer to command `dbaascli cswlib showImages`. Image Tags are typically same as the version of the database. However, it is kept as a provision for cases where multiple images may need to be released for the same version - each catering to a specific customer requirement.

4. Exit the `root` user command shell:

   ```
   exit
   ```

   For more details on advanced supported options, see `dbaascli dbhome create`.

Related Topics

- Connecting to a Virtual Machine with SSH
  You can connect to the virtual machines in an Oracle Exadata Database Service on Exascale Infrastructure system by using a Secure Shell (SSH) connection.
- dbaascli dbhome create
  To create an Oracle Database home of desired version, use the `dbaascli dbhome create` command.

Creating Oracle Database In the Specified Oracle Database Home

To create an Oracle Database in the specified Oracle Database home of desired version, use the `dbaascli database create` command.

You can use the `dbaascli database create` command to:

- Create a Container Database (CDB) or non-Container Database
- Create a CDB with pluggable databases (PDBs)
- Create an Oracle Database with the specified Character Set
- Create Oracle Databases on a subset of cluster nodes

> **Note:**
>
> Databases created on a subset of nodes will not be displayed in the OCI console.

- Create Oracle Database version 12.1.0.2 or higher with the release update JAN 2021 or higher. For databases with lower versions, it is recommended to use the OCI Console based API.

1. Connect to the virtual machine as the `opc` user.
   For detailed instructions, see *Connecting to a Virtual Machine with SSH*.
2. Start a `root` user command shell:

   ```
   sudo -s
   ```

3. Run the following command:

   ```
   dbaascli database create --dbName database name --oracleHome Oracle Home
   Path
   ```

   Where:

   - `--dbName` specifies the name of the database
   - `--oracleHome` specifies Oracle home location

   To create a CDB, run the following command:

   ```
   dbaascli database create --dbName database name --oracleHome Oracle Home
   Path
   ```

   To create a non-CDB, run the following command:

   ```
   dbaascli database create --dbName database name --oracleHome Oracle Home
   Path --createAsCDB false
   ```

   When prompted, enter the `sys` and `tde` passwords.
4. Exit the `root` user command shell:

   ```
   exit
   ```

   For more details on advanced supported options, see `dbaascli database create`.

- Running Prerequisite Checks Prior to Creating Oracle Database
  To run prerequisites checks, use the `--executePrereqs` command option. This will perform only the prerequisite checks without performing the actual Oracle Database creation.
- Resuming or Reverting Oracle Database Creation Operation
  To resume or revert a failed database creation operation, use the `--resume` or `--revert` command option.

Related Topics

- Connecting to a Virtual Machine with SSH
  You can connect to the virtual machines in an Oracle Exadata Database Service on Exascale Infrastructure system by using a Secure Shell (SSH) connection.
- dbaascli database create
  To create Oracle Database, use the `dbaascli database create` command. When prompted, enter the `sys` and `tde` passwords.

Running Prerequisite Checks Prior to Creating Oracle Database

To run prerequisites checks, use the `--executePrereqs` command option. This will perform only the prerequisite checks without performing the actual Oracle Database creation.

1. Connect to the virtual machine as the `opc` user.
   For detailed instructions, see *Connecting to a Virtual Machine with SSH*.
2. Start a `root` user command shell:

   ```
   sudo -s
   ```

3. Run the following command:

   ```
   dbaascli database create --dbName database name --oracleHome Oracle Home
   Path --executePrereqs
   ```

   Where:

   - `--dbName` specifies the name of the database
   - `--oracleHome` specifies the Oracle home location
4. Exit the `root` user command shell:

   ```
   exit
   ```

   For more details on advanced supported options, see `dbaascli database create`.

Related Topics

- Connecting to a Virtual Machine with SSH
  You can connect to the virtual machines in an Oracle Exadata Database Service on Exascale Infrastructure system by using a Secure Shell (SSH) connection.
- dbaascli database create
  To create Oracle Database, use the `dbaascli database create` command. When prompted, enter the `sys` and `tde` passwords.

Resuming or Reverting Oracle Database Creation Operation

To resume or revert a failed database creation operation, use the `--resume` or `--revert` command option.

For example:

```
dbaascli database create --dbName database name --oracleHome Oracle Home Path
--resume
```

Note:

- While using the `--resume` or `--revert` command options, ensure that you use the same command from the same node that was used for actual create operation flow.
- You can resume database creation only if there is a failure in the post database creation step.

Related Topics

- Connecting to a Virtual Machine with SSH
  You can connect to the virtual machines in an Oracle Exadata Database Service on Exascale Infrastructure system by using a Secure Shell (SSH) connection.
- dbaascli database create
  To create Oracle Database, use the `dbaascli database create` command. When prompted, enter the `sys` and `tde` passwords.

Changing the Database Passwords

To change the SYS password, or to change the TDE wallet password, use this procedure.

The password that you specify in the **Database Admin Password** field when you create a new Oracle Exadata Database Service on Exascale Infrastructure instance or database is set as the password for the SYS, SYSTEM, TDE wallet, and PDB administrator credentials. Use the following procedures if you need to change passwords for an existing database.

Note:

if you are enabling Data Guard for a database, then the SYS password and the TDE wallet password of the primary and standby databases must all be the same.

Note:

Using the `dbaascli` to change the SYS password will ensure the backup/restore automation can parallelize channels across all nodes in the cluster.

To Change the SYS Password for an Oracle Exadata Database Service on Exascale Infrastructure Database

1. Log onto the Oracle Exadata Database Service on Exascale Infrastructure virtual machine as `opc`.
2. Run the following command:

```
sudo dbaascli database changepassword --dbname database_name --user SYS
```

To Change Database Passwords in a Data Guard Environment

1. Run the following command on the primary database:

   ```
   dbaascli database changePassword —dbName <dbname> --user SYS --
   prepareStandbyBlob true --blobLocation <location to create the blob file>
   ```

2. Copy the blob file created to all the standby databases and update the file ownership to `oracle` user.
3. Run the following command on all the standby databases:

   ```
   dbaascli database changePassword —dbName <dbname> --user SYS --
   standbyBlobFromPrimary <location of copies the blob file>
   ```

To Change the TDE Wallet Password for an Oracle Exadata Database Service on Exascale Infrastructure Database

1. Log onto the Oracle Exadata Database Service on Exascale Infrastructure virtual machine as `opc`.
2. Run the following command:

   ```
   sudo dbaascli tde changepassword --dbname database_name
   ```

Managing Oracle Exadata Database Service on Exascale Infrastructure Software Images Using the Dbaascli Utility

You can list and download the Oracle database software images on an Oracle Exadata Database Service on Exascale Infrastructure instance, which can then be used for provisioning a database home.

> **Note:**
>
> You can create custom database software images for your Oracle Exadata Database Service on Exascale Infrastructure instances using the Console or API. These images are stored in Object Storage, and can be used to provision a Database Home in your Exadata instance. See Oracle Database Software Images more information.

You can control the version of Oracle binaries that is installed when you provision a new database on an Oracle Exadata Database Service on Exascale Infrastructure instance by maintaining the software images on the system. Oracle provides a library of cloud software images that you can view and download onto your instance by using the `dbaascli` utility.

- Listing Available Software Images and Versions for Database and Grid Infrastructure
  To produce a list of available supported versions for patching, use the `dbaascli cswlib showImages` command.

- To download a software image
  You can download available software images onto your Oracle Exadata Database Service on Exascale Infrastructure instance by using the `cswlib download` subcommand of the `dbaascli` utility.

Listing Available Software Images and Versions for Database and Grid Infrastructure

To produce a list of available supported versions for patching, use the `dbaascli cswlib showImages` command.

1. Connect to the virtual machine as the `opc` user.
   For detailed instructions, see *Connecting to a Virtual Machine with SSH*.
2. Start a `root` user command shell:

   ```
   sudo -s
   ```

3. Run the following command:

   ```
   dbaascli cswlib showImages --product database
   ```

   The command output lists the available database software images.

   ```
   dbaascli cswlib showImages --product grid
   ```

   The command output lists the available grid software images.
4. Exit the `root` user command shell:

   ```
   exit
   ```

   For more details on advanced supported options, see `dbaascli cswlib showImages`.

Example 5-2 dbaascli cswlib showImages

```
[root@dg11lrg1 dbhome_1]# dbaascli cswlib showImages
DBAAS CLI version <version>
Executing command cswlib
      showImagesJob id: 00e89b1a-1607-422c-a920-22f44bec1953Log file location:
      /var/opt/oracle/log/cswLib/showImages/dbaastools_2022-05-11_08-49-12-
AM_46941.log

############
List of Available Database Images
#############

17.IMAGE_TAG=18.17.0.0.0
   VERSION=18.17.0.0.0
   DESCRIPTION=18c JAN 2022 DB Image

18.IMAGE_TAG=19.10.0.0.0
   VERSION=19.10.0.0.0
   DESCRIPTION=19c JAN 2021 DB Image

19.IMAGE_TAG=19.11.0.0.0
```

```
   VERSION=19.11.0.0.0
   DESCRIPTION=19c APR 2021 DB Image

20.IMAGE_TAG=19.12.0.0.0
  VERSION=19.12.0.0.0
  DESCRIPTION=19c JUL 2021 DB Image

21.IMAGE_TAG=19.13.0.0.0
  VERSION=19.13.0.0.0
  DESCRIPTION=19c OCT 2021 DB Image

Images can be downloaded using their image tags. For details, see help using 
'dbaascli cswlib download --help'.
dbaascli execution completed
```

Related Topics

- Connecting to a Virtual Machine with SSH
  You can connect to the virtual machines in an Oracle Exadata Database Service on Exascale Infrastructure system by using a Secure Shell (SSH) connection.
- dbaascli cswlib showImages
  To view the list of available Database and Grid Infrastructure images, use the `dbaascli cswlib showImages` command.

To download a software image

You can download available software images onto your Oracle Exadata Database Service on Exascale Infrastructure instance by using the `cswlib download` subcommand of the `dbaascli` utility.

1. Connect to a compute node as the `opc` user.For detailed instructions, see *Connecting to a Virtual Machine with SSH*.
2. Start a root-user command shell:

   ```
   $ sudo -s
   #
   ```

3. Execute the `dbaascli` command with the `cswlib download` subcommand:

   ```
   # dbaascli cswlib download [--version <software_version>] [--imageTag 
   <image tag
       value>]
   ```

   The command displays the location of software images that are downloaded to your Oracle Exadata Database Service on Exascale Infrastructure environment.
   The optional parameters are:

   - **version:** specifies an Oracle Database software version. For example, 19.14.0.0.0.
   - **imageTag:** specifies the image tag of the image.

4. Exit the root-user command shell:

   ```
   # exit
   $
   ```

Related Topics

- Connecting to a Virtual Machine with SSH
  You can connect to the virtual machines in an Oracle Exadata Database Service on Exascale Infrastructure system by using a Secure Shell (SSH) connection.

Collect Cloud Tooling Logs and Perform a Cloud Tooling Health Check Using dbaascli

Using the dbaascli `diag` command allows you to collect Guest VM `dbaas` tooling logs for Exadata Database Service on Dedicated Infrastructure and Exadata Database Service on Cloud@Customer systems. You can use these logs to troubleshoot issues related to `dbaas` tooling.

You can use the `diag` command to collect dbaastools logs and perform a health check on all nodes in an Exadata cluster. Note that the `--waitForCompletion` options is supported starting in version 22.4.1

> **Note:**
>
> - dbaascli `diag` commands must be run as the `root` user
> - Running the `dbaascli diag collect` command on a single node will collect log data for all nodes
> - We recommend running the commands documented in this topic using the `--waitForCompletion` option for long-running commands. Refer to the examples for sample usage.

For information on updating Exadata Cloud Tooling, see *dbaascli admin updateStack*.

- Collecting Tooling Log Data Examples
  The dbaascli dbaascli diag collect command uses the syntax shown below to collect tooling log data:
- Performing a Health Check Examples
  Use dbaascli `dbaascli diag healthcheck` command to perform a health check on all system nodes.

Related Topics

- dbaascli diag collect
  To collect diagnostics, use the `dbaascli diag collect` command.
- dbaascli admin updateStack
  To install or update a dbaastools RPM, use the `dbaascli admin updateStack` command.

Collecting Tooling Log Data Examples

The dbaascli dbaascli diag collect command uses the syntax shown below to collect tooling log data:

See `dbaascli diag collect` In the *dbaascli Command Reference* for syntax details

NOT_SUPPORTED

```
# dbaascli diag collect
DBAAS CLI version 22.4.1.0.1
Executing command diag collectJob id: 4c1c7908-541d-4ebc-8e44-042c913f6779
Loading PILOT...Session ID of the current execution is: 4
Log file location: /var/opt/oracle/log/diag/collect/pilot_2022-09-16_05-43-30-
PM_99944
-----------------
...
---------- DIAG COLLECT PLUGIN RESULT ----------
{"collectedArchive":"/var/opt/oracle/dbaas_acfs/diag_collect/
diag_cloudlogs_1663376003.tar.gz","SHA256CheckSum":"07136fc102dabf19be82143048
2412f5e43e9d32d7aa661e190e7774bcbc5e85"}
dbaascli execution completed
```

NOT_SUPPORTED

```
# dbaascli diag collect --waitForCompletion false
DBAAS CLI version 22.4.1.0.1
Executing command diag collect --waitForCompletion false
Job id: 1b8f9fec-0736-46ec-ac76-cb5a37181c14
Job accepted. Use "dbaascli job getStatus --jobID 1b8f9fec-0736-46ec-ac76-
cb5a37181c14" to check the job status.
```

Note:

Use the job status command to monitor progress.

NOT_SUPPORTED

```
# dbaascli diag collect --dbNames myOracleDatabase19cName
DBAAS CLI version 22.4.1.0.1
Executing command diag collect --dbNames myOracleDatabase19cName
Job id: 3e12abff-ee89-467d-9afd-a30f9cab5967
Loading PILOT...
Session ID of the current execution is: 26
Log file location: /var/opt/oracle/log/diag/collect/pilot_2022-12-06_05-47-54-
PM_60919
-----------------
...
---------- DIAG COLLECT PLUGIN RESULT ----------
{"collectedArchive":"/var/opt/oracle/dbaas_acfs/diag_collect/
diag_cloudlogs_20221206-0547.tar.gz","SHA256
CheckSum":"3ce17c6cbc7a4039ec2bacfa085a7e801743e6423e063b45687501c833bc8282"}
dbaascli execution completed
```

NOT_SUPPORTED

```
# dbaascli diag collect --destLocation /tmp/test/
DBAAS CLI version 22.4.1.0.1
```

```
Executing command diag collect --destLocation /tmp/test/
Job id: c73ebadc-ebac-4180-b465-3250ec695c01
Loading PILOT...
Session ID of the current execution is: 20
Log file location: /var/opt/oracle/log/diag/collect/pilot_2022-12-06_05-16-15-
PM_54647
-----------------
...
---------- DIAG COLLECT PLUGIN RESULT ----------
{"collectedArchive":"/tmp/test/diag_cloudlogs_20221206-0605.tar.gz","SHA256
CheckSum":"3670cae9a99784d1f7e647dc814e1cb519bb16f8562112d6d492babd272cd82a"}
dbaascli execution completed
```

NOT_SUPPORTED

```
# dbaascli diag collect --startTime 2021-03-19T10:00:00 --endTime
2021-03-20T10:00:00
DBAAS CLI version 22.4.1.0.1
Executing command diag collect --startTime 2021-03-19T10:00:00 --endTime
2021-03-20T10:00:00
Job id: 11b7a54a-7257-4ebd-b12c-9dc37e76ee86
Loading PILOT...
Session ID of the current execution is: 23
Log file location: /var/opt/oracle/log/diag/collect/pilot_2022-12-06_05-25-33-
PM_51327
-----------------
...
---------- DIAG COLLECT PLUGIN RESULT ----------
{"collectedArchive":"/var/opt/oracle/dbaas_acfs/diag_collect/
diag_cloudlogs_20221206-0525.tar.gz","SHA256
CheckSum":"7dd3d32e91688657e951b3ffd8d06cb335a408210bf213ad5a83c2a451ca0171"}
dbaascli execution completed
```

NOT_SUPPORTED

```
# dbaascli diag collect --nodes dbnode1,dbnode2
DBAAS CLI version 22.4.1.0.1
Executing command diag collect --nodes dbnode1,dbnode2
Job id: 26b41f03-b100-450a-9c3a-ae17120442a3
Loading PILOT...
Session ID of the current execution is: 24
Log file location: /var/opt/oracle/log/diag/collect/pilot_2022-12-06_05-27-35-
PM_82872
-----------------
...
---------- DIAG COLLECT PLUGIN RESULT ----------
{"collectedArchive":"/var/opt/oracle/dbaas_acfs/diag_collect/
diag_cloudlogs_20221206-0527.tar.gz","SHA256
CheckSum":"2a7fc0c12e0be3e8ad32f6297ed2f4532b6f883b79771dd8420a9e46102faef4"}
dbaascli execution completed
```

NOT_SUPPORTED

```
# dbaascli diag collect --components dbaastools
DBAAS CLI version 22.4.1.0.1
Executing command diag collect --components dbaastools
Job id: 7c89876d-b25c-4843-8d12-8885271891f5
Loading PILOT...
Session ID of the current execution is: 25
Log file location: /var/opt/oracle/log/diag/collect/pilot_2022-12-06_05-46-09-
PM_28513
-----------------
...
---------- DIAG COLLECT PLUGIN RESULT ----------
{"collectedArchive":"/var/opt/oracle/dbaas_acfs/diag_collect/
diag_cloudlogs_20221206-0546.tar.gz","SHA256
CheckSum":"c940bedbb7cfd63d51faa26ba2d04cc41483e76efaf038d6fe3644b64125db51"}
dbaascli execution completed
```

NOT_SUPPORTED

```
# dbaascli diag collect --objectStoreBucketUri https://objectstorage.us-
phoenix-1.oraclecloud.com/p/t0Z-kRV5pSmFzqnf-
y5XhaAbM4LS82epeBnulKnCr31IeHVjxI9tOkntLF2kq7fP/n/MyNamespace/b/MyParBucket/o/
DBAAS CLI version 22.4.1.0.1
Executing command diag collect --objectStoreBucketUri https://
objectstorage.us-phoenix-1.oraclecloud.com/p/t0Z-kRV5pSmFzqnf-
y5XhaAbM4LS82epeBnulKnCr31IeHVjxI9tOkntLF2kq7fP/n/MyNamespace/b/MyParBucket/o/
Job id: b7c5d682-5527-4911-9e60-aab7fdb03014
Loading PILOT...
Session ID of the current execution is: 27
Log file location: /var/opt/oracle/log/diag/collect/pilot_2022-12-06_05-50-11-
PM_92873
-----------------
...
---------- DIAG COLLECT PLUGIN RESULT ----------
{"collectedArchive":"https://objectstorage.us-phoenix-1.oraclecloud.com/p/t0Z-
kRV5pSmFzqnf-y5XhaAbM4LS82epeBnulKnCr31IeHVjxI9tOkntLF2kq7fP/n/MyNamespace/b/
MyParBucket/o/diag_cloudlogs_20221206-0550.tar.gz","SHA256
CheckSum":"a6f0695473f763d547826c29bda73adcc7e50962652a189c22b0a3bb4f514e48"}
dbaascli execution completed
```

Related Topics

- dbaascli diag collect
  To collect diagnostics, use the `dbaascli diag collect` command.

Performing a Health Check Examples

Use dbaascli `dbaascli diag healthcheck` command to perform a health check on all system nodes.

See *dbaascli diag healthcheck* for the syntax details in the dbaascli Command Reference.

NOT_SUPPORTED

```
# dbaascli diag healthcheck
DBAAS CLI version MAIN
Executing command diag healthcheck
INFO: Starting diag healthcheck
INFO: Collected diag logs at: /var/opt/oracle/dbaas_acfs/
diag_cloudlogs_20210322-2246.tar.gz
```

NOT_SUPPORTED

```
# dbaascli diag healthcheck --destLocation /tmp/test
DBAAS CLI version MAIN
Executing command diag healthcheck --destLocation /tmp/test
INFO: Starting diag healthcheck
INFO: Collected diag logs at: /tmp/test/diag_cloudlogs_20210322-2250.tar.gz
```

NOT_SUPPORTED

```
# dbaascli diag healthcheck --nodes rbcl1,rbcl2
DBAAS CLI version MAIN
Executing command diag healthcheck --nodes rbcl1,rbcl2
INFO: Starting diag healthcheck
INFO: Collected diag logs at: /var/opt/oracle/dbaas_acfs/
diag_cloudlogs_20210421-1915.tar.gz
```

NOT_SUPPORTED

```
# dbaascli diag healthcheck --objectStoreBucketUri https://objectstorage.us-
phoenix-1.oraclecloud.com/p/t0Z-kRV5pSmFzqnf-
y5XhaAbM4LS82epeBnulKnCr31IeHVjxI9tOkntLF2kq7fP/n/MyNamespace/b/MyParBucket/o/
DBAAS CLI version MAIN
Executing command diag healthcheck --objectStoreBucketUri https://
objectstorage.us-phoenix-1.oraclecloud.com/p/t0Z-kRV5pSmFzqnf-
y5XhaAbM4LS82epeBnulKnCr31IeHVjxI9tOkntLF2kq7fP/n/MyNamespace/b/MyParBucket/o/
INFO: Collected diag logs at: https://objectstorage.us-
phoenix-1.oraclecloud.com/p/t0Z-kRV5pSmFzqnf-
y5XhaAbM4LS82epeBnulKnCr31IeHVjxI9tOkntLF2kq7fP/n/MyNamespace/b/MyParBucket/o/
diag_cloudlogs_20210421-1839.tar.gz
```

Related Topics

- dbaascli diag collect
  To collect diagnostics, use the `dbaascli diag collect` command.
- dbaascli diag healthCheck
  To run diagnostic health checks, use the `dbaascli diag healthCheck` command.

Updating Cloud Tooling Using dbaascli

To update the cloud tooling release for Oracle Exadata Database Service on Exascale Infrastructure, complete this procedure.

Cloud-specific tooling is used on the Oracle Exadata Database Service on Exascale Infrastructure Guest VMs for local operations, including `dbaascli` commands.

The cloud tooling is automatically updated by Oracle when new releases are made available. If needed, you can follow the steps below to ensure you have the latest version of the cloud-specific tooling on all of the virtual machines in the VM cluster.

> **Note:**
>
> You can update the cloud-specific tooling by downloading and applying a software package containing the updated tools.

1. Connect to a virtual machine as the `opc` user.
   For detailed instructions, see *Connecting to a Virtual Machine with SSH*.
2. Start a `root` user command shell:

   ```
   sudo -s
   ```

3. To update to the latest available cloud tooling release, run the following command:

   ```
   dbaascli admin updateStack
   ```

   The command takes care of updating the cloud tooling release on all the nodes of the cluster.

   For more details and other available options, refer to `dbaascli admin updateStack --help`.

Related Topics

- Connecting to a Virtual Machine with SSH
  You can connect to the virtual machines in an Oracle Exadata Database Service on Exascale Infrastructure system by using a Secure Shell (SSH) connection.
- dbaascli admin updateStack
  To install or update a dbaastools RPM, use the `dbaascli admin updateStack` command.

Creating a Duplicate Database

- Using dbaascli to Duplicate a Cloud Database
- Considerations When Using OCI Vault for the Key Management

Using dbaascli to Duplicate a Cloud Database

You can create a duplicate database using `dbaascli`. This new database can be in the same cloud region as the source region or across the regions. The following steps describe how to create a duplicate database on cloud.

> **Note:**
>
> If a database is configured with OCI Vault for TDE encryption and you want to duplicate a database, then refer to the following sections.

Prepare for duplication

Ensure that the following prerequisites are ment:

- Make sure that there is a network path setup to access the source database through the `EZConnect` string.
- Copy the TDE wallet file (`ewallet.p12`) to the target database node. The node where you decide to run the `dbaascli` command.
- Create an Oracle home on the target node if required. Oracle home version must be the same version as the source or of higher RU version.

Run prerequisite checks

To run prerequisites checks, use the `--executePrereqs` command option. This will perform only the prerequisite checks without performing the actual Oracle Database duplication.

```
dbaascli database duplicate --dbName <database name> --oracleHome <Oracle
Home Path> --sourceDBConnectionString <source database EZConnect string> --
sourceDBTDEWalletLocation <location of copied wallet> --
sourceDBTdeConfigMethod FILE --tdeConfigMethod FILE --executePrereqs
```

Duplicate the database

```
dbaascli database duplicate --dbName <database name> --oracleHome <Oracle
Home Path> --sourceDBConnectionString <source database EZConnect string> --
sourceDBTDEWalletLocation <location of copied wallet> --
sourceDBTdeConfigMethod FILE --tdeConfigMethod FILE
```

> **Note:**
>
> If source database is using OKV for TDE keystore management, current duplicate database operation does not support this configuration.

Considerations When Using OCI Vault for the Key Management

This section is applicable only in the case of database is configured with OCI Vault for TDE encryption and you want to duplicate a database.

Duplicating a database within the same region

- Additional prerequisite steps
  Make sure to setup OCI Vault access policies for target database nodes. Target database nodes should be able to access both source database's OCI key vault along with its new key vault (if it is decided to use separate key vault).

- Run prerequisite checks

  ```
  dbaascli database duplicate --dbName <database name> --oracleHome <Oracle
  Home Path> --sourceDBConnectionString <source database EZConnect string> --
  sourceDBTDEWalletLocation <location of copied wallet> --
  sourceDBTdeConfigMethod KMS --sourceDBKMSKeyOCID <Source Database OCI
  Vault key OCID> --tdeConfigMethod KMS --kmsKeyOCID <OCI Vault key OCID> --
  executePrereqs
  ```

- Duplicate the database

  ```
  dbaascli database duplicate --dbName <database name> --oracleHome <Oracle
  Home Path> --sourceDBConnectionString <source database EZConnect string> --
  sourceDBTDEWalletLocation <location of copied wallet> --
  sourceDBTdeConfigMethod KMS --sourceDBKMSKeyOCID <Source Database OCI
  Vault key OCID> --tdeConfigMethod KMS --kmsKeyOCID <OCI Vault key OCID>
  ```

  Upon successful completion of this command, the database is duplicated.

Duplicating a database across regions

- Additional prerequisite steps
  - Setup a new OCI Vault for target database on the corresponding region by following the steps outlined in Prepare to Use Customer-Managed Keys in the Vault Service. Complete Tasks 1 through 3.
  - Setup OCI Vault replication from source region to target region. For more information, see Replicating Vaults and Keys.
  - Update Dynamic group policy, which is created in step 2 to allow access to replicated OCI Vault key.
- Run prerequisite checks

  ```
  dbaascli database duplicate --dbName <database name> --oracleHome <Oracle
  Home Path> --sourceDBConnectionString <source database EZConnect string> --
  sourceDBTDEWalletLocation <location of copied wallet> --
  sourceDBTdeConfigMethod KMS --sourceDBKMSKeyOCID <Source Database OCI
  Vault key OCID> --tdeConfigMethod KMS --kmsKeyOCID <OCI Vault key OCID> --
  executePrereqs
  ```

- Duplicate the database

  ```
  dbaascli database duplicate --dbName <database name> --oracleHome <Oracle
  Home Path> --sourceDBConnectionString <source database EZConnect string> --
  sourceDBTDEWalletLocation <location of copied wallet> --
  sourceDBTdeConfigMethod KMS --sourceDBKMSKeyOCID <Source Database OCI
  Vault key OCID> --tdeConfigMethod KMS --kmsKeyOCID <OCI Vault key OCID>
  ```

  Upon successful completion of this command, the database is duplicated.

dbaascli Command Reference

You use `dbaascli` to create databases and integrate them with the cloud automation framework.

`dbaascli` is a cloud native interface that can take DBCA templates as inputs, calls the functionality of DBCA to create databases, and then calls OCI APIs to integrate the database into the cloud automation framework. Customers using DBCA in scripts today can update their existing scripts to call `dbaascli` instead of DBCA. If `dbaascli` cannot be used due to a particular feature of DBCA being unavailable in dbaascl, then customers should open a My Oracle Support (MOS) request to add that functionality to `dbaascli`.

To use the `dbaascli` utility, you must be connected to an Oracle Exadata Database Service on Exascale Infrastructure compute node.

Some `dbaascli` commands can be run as the `oracle` or the `opc` user, but many commands require `root` administrator privileges. Refer to each command for specific requirements.

- dbaascli admin updateStack
  To install or update a dbaastools RPM, use the `dbaascli admin updateStack` command.
- dbaascli cswlib deleteLocal
  To delete the local image, use the `dbaascli cswlib deleteLocal` command.
- dbaascli cswlib download
  To download available software images and make them available in your Oracle Exadata Database Service on Exascale Infrastructure environment, use the `dbaascli cswlib download` command.
- dbaascli cswlib listLocal
  To view the list of locally available Database and Grid Infrastructure images, use the `dbaascli cswlib listLocal` command.
- dbaascli cswlib showImages
  To view the list of available Database and Grid Infrastructure images, use the `dbaascli cswlib showImages` command.
- dbaascli database addInstance
  To add the database instance on the specified node, use the `dbaascli database addInstance` command.
- dbaascli database backup
  To configure Oracle Database with a backup storage destination, take database backups, query backups, and delete a backup, use the `dbaascli database backup` command.
- dbaascli database bounce
  To shut down and restart a specified Oracle Exadata Database Service on Exascale Infrastructure database, use the `dbaascli database bounce` command.
- dbaascli database changepassword
  To change the password of a specified Oracle Database user, use the `dbaascli database changePassword` command. When prompted enter the user name for which you want to change the password and then enter the password.
- dbaascli database convertToPDB
  To convert the specified non-CDB database to PDB, use the `dbaascli database convertToPDB` command.
- dbaascli database create
  To create Oracle Database, use the `dbaascli database create` command. When prompted, enter the `sys` and `tde` passwords.
- dbaascli database delete
  To delete an Oracle Database, use the `dbaascli database delete` command.

- dbaascli database deleteInstance
  To delete the database instance on the specified node, use the `dbaascli database deleteInstance` command.
- dbaascli database duplicate
  To create a database from an active database, use the `dbaascli database duplicate` command.
- dbaascli database getDetails
  This command shows the detailed information of a given database e.g. dbname, node information, pluggable databases information etc.
- dbaascli database getPDBs
  To view the list of all pluggable databases in a container database, use the `dbaascli database getPDBs` command.
- dbaascli database modifyParameters
  To modify or reset initialization parameters for an Oracle Database, use the `dbaascli database modifyParameters` command.
- dbaascli database move
  To move the database from one home to another, use the `dbaascli database move` command.
- dbaascli database recover
  To recover a database, use the `dbaascli database recover` command.
- dbaascli database runDatapatch
  To patch an Oracle Database, use the `dbaascli database runDatapatch` command.
- dbaascli database createTemplate
  To create database templates (DBCA templates) that can subsequently be used to create databases, use the `dbaascli database createTemplate` command.
- dbaascli database start
  To start an Oracle Database, use the `dbaascli database start` command.
- dbaascli database status
  To check the status of an Oracle Database, use the `dbaascli database status` command.
- dbaascli database stop
  To stop an Oracle Database, use the `dbaascli database stop` command.
- dbaascli database upgrade
  To upgrade an Oracle Database, use the `dbaascli database upgrade` command.
- dbaascli dataguard prepareStandbyBlob
  To generate a blob file containing various files that are required on the standby site in case of a dataguard environment, use the `dbaascli dataguard prepareStandbyBlob` command.
- dbaascli dataguard updateDGConfigAttributes
  To update Data Guard automation attributes across all the cluster nodes, use the `dbaascli dataguard updateDGConfigAttributes` command.
- dbaascli dbhome create
  To create an Oracle Database home of desired version, use the `dbaascli dbhome create` command.
- dbaascli dbHome delete
  To delete a given Oracle Database home, use the `dbaascli dbHome delete` command.

- dbaascli dbhome getDatabases
  To view information about all Oracle Databases running from a given database Oracle home, use the `dbaascli dbHome getDatabases` command. Specify either the Oracle home location or Oracle home name.
- dbaascli dbHome getDetails
  To view information about a specific Oracle home, use the `dbaascli dbHome getDetails` command. Specify either the Oracle home location or Oracle home name.
- dbaascli dbHome patch
  To patch Oracle home from one patch level to another, use the `dbaascli dbHome patch` command.
- dbaascli dbimage purge
  The `dbimage purge` command removes the specified software image from your Oracle Exadata Database Service on Exascale Infrastructure environment.
- dbaascli diag collect
  To collect diagnostics, use the `dbaascli diag collect` command.
- dbaascli diag healthCheck
  To run diagnostic health checks, use the `dbaascli diag healthCheck` command.
- dbaascli grid configureTCPS
  To configure TCPS for the existing cluster, use the `dbaascli grid configureTCPS` command.
- dbaascli grid patch
  To patch Oracle Grid Infrastructure to the specified minor version, use the `dbaascli grid patch` command.
- dbaascli grid removeTCPSCert
  To remove existing TCPS certificates from Grid Infrastructure wallet, use the `dbaascli grid removeTCPSCert` command.
- dbaascli grid rotateTCPSCert
  To rotate TCPS certificates, use the dbaascli grid rotateTCPSCert command.
- dbaascli grid upgrade
  To upgrade Oracle Grid Infrastrucure from one major version to another, use the `dbaascli grid upgrade` command.
- dbaascli job getStatus
  To view the status of a specified job, use the `dbaascli job getStatus` command.
- dbaascli patch db apply
- dbaascli patch db prereq
- dbaascli pdb backup
  To backup a pluggable database (PDB), query PDB backups, and delete a PDB backup, use the `dbaascli pdb backup` command.
- dbaascli pdb bounce
  To bounce a pluggable database (PDB), use the `dbaascli pdb bounce` command.
- dbaascli pdb close
  To close a pluggable database (PDB), use the `dbaascli pdb close` command.
- dbaascli pdb getConnectString
  To display Oracle Net connect string information for a pluggable database (PDB) run the `dbaascli pdb getConnectString` command.

- dbaascli pdb create
  To create a new pluggable database (PDB), use the `dbaascli pdb create` command.
- dbaascli pdb createSnapshot
  To create a snapshot of a given pluggable database (PDB), use the `dbaascli pdb createSnapshot` command.
- dbaascli pdb configureSnapshot
  To configure automatic snapshots for a given pluggable database (PDB), use the `dbaascli pdb configureSnapshot` command.
- dbaascli pdb delete
  To delete a pluggable database (PDB) run the `dbaascli pdb delete` command.
- dbaascli pdb deleteSnapshot
  To delete a snapshot of a given pluggable database (PDB), use the `dbaascli pdb deleteSnapshot` command.
- dbaascli pdb getDetails
  To view details of a pluggable database (PDB), use the `dbaascli pdb getDetails` command.
- dbaascli pdb getSnapshot
  To obtain details of a given pluggable database (PDB) snapshot, use the `dbaascli pdb getSnapshot` command.
- dbaascli pdb list
  To view the list of pluggable databases (PDB) in a container database, use the `dbaascli pdb list` command.
- dbaascli pdb listSnapshots
  To list the snapshots of a given pluggable database (PDB), use the `dbaascli pdb listSnapshots` command..
- dbaascli pdb localClone
  To create a new pluggable database (PDB) as a clone of an existing PDB in the same container database (CDB), use the `dbaascli pdb localClone` command.
- dbaascli pdb open
  To open a pluggable database (PDB), use the `dbaascli pdb open` command.
- dbaascli pdb recover
  To recover a pluggable database (PDB), use the `dbaascli pdb recover` command.
- dbaascli pdb refresh
  To refresh a specified pluggable database (PDB), use the `dbaascli pdb refresh` command.
- dbaascli pdb relocate
  To relocate the specified PDB from the remote database into local database, use the `dbaascli pdb relocate` command.
- dbaascli pdb remoteClone
  To create a new pluggable database (PDB) as a clone of an existing PDB in another container database (CDB), use the `dbaascli pdb remoteClone` command.
- dbaascli system getDBHomes
  To view information about all the Oracle homes, use the `dbaascli system getDBHomes` command.
- dbaascli tde changePassword
  To change TDE keystore password as well as DB wallet password for the alias `tde_ks_passwd`, use the `dbaascli tde changePassword` command.

- dbaascli tde addSecondaryHsmKey
  To add a secondary HSM (KMS) key to the existing HSM (KMS) configuration, use the `dbaascli tde addSecondaryHsmKey` command.
- dbaascli tde enableWalletRoot
  To enable `wallet_root` spfile parameter for the existing database, use the `dbaascli tde enableWalletRoot` command.
- dbaascli tde encryptTablespacesInPDB
  To encrypt all the tablespaces in the specified PDB, use the `dbaascli tde encryptTablespacesInPDB` command.
- dbaascli tde fileToHsm
  To convert FILE based TDE to HSM (KMS/OKV) based TDE, use the `dbaascli tde fileToHsm` command.
- dbaascli tde getHsmKeys
  To get TDE active key details, use the `dbaascli tde getHsmKeys` command.
- dbaascli tde getMkidForKeyVersionOCID
  To get Master Key ID associated with the KMS key version OCID, use the `dbaascli tde getMkidForKeyVersionOCID` command.
- dbaascli tde getPrimaryHsmKey
  To get primary HSM (KMS) key from the existing HSM (KMS) configuration, use the `dbaascli tde getPrimaryHsmKey` command.
- dbaascli tde hsmToFile
  To convert HSM (KMS/OKV) based TDE to FILE based TDE, use the `dbaascli tde hsmToFile` command.
- dbaascli tde listKeys
  To list TDE master keys, use the `dbaascli tde listKeys` command.
- dbaascli tde removeSecondaryHsmKey
  To remove secondary HSM (KMS) key from the existing HSM (KMS) configuration, use the `dbaascli tde removeSecondaryHsmKey` command.
- dbaascli tde rotateMasterKey
  To rotate the master key for database encryption, use the `dbaascli tde rotateMasterKey` command.
- dbaascli tde setKeyVersion
  To set the version of the primary key to be used in DB/CDB or PDB, use the `dbaascli tde setKeyVersion` command.
- dbaascli tde setPrimaryHsmKey
  To change the primary HSM (KMS) key for the existing HSM (KMS) configuration, use the `dbaascli tde setPrimaryHsmKey` command.
- dbaascli tde status
  To display information about the keystore for the specified database, use the `dbaascli tde status` command.

dbaascli admin updateStack

To install or update a dbaastools RPM, use the `dbaascli admin updateStack` command.

Prerequisites

Run the command as the `root` user.

To use the utility, you must connect to an Oracle Exadata Database Service on Exascale Infrastructure virtual machine.

See, *Connecting to a Virtual Machine with SSH*.

Syntax

```
dbaascli admin updateStack
[--resume]
[--prechecksOnly]
[--nodes]
```

Where:

- `--resume` resumes the previous execution
- `--prechecksOnly` runs only the prechecks for this operation
- `--nodes` specifies a comma-delimited list of nodes to install the RPM on. If you do not pass this argument, then the RPM will be installed on all of the cluster nodes

Related Topics

- Connecting to a Virtual Machine with SSH
  You can connect to the virtual machines in an Oracle Exadata Database Service on Exascale Infrastructure system by using a Secure Shell (SSH) connection.

dbaascli cswlib deleteLocal

To delete the local image, use the `dbaascli cswlib deleteLocal` command.

Run the command as the `root` user.

Syntax

```
dbaascli cswLib deleteLocal --imageTag <value>
```

Where:

- `--imageTag` specifies Oracle home image tag

Example 5-3 dbaascli cswlib deletelocal

```
dbaascli cswlib deletelocal --imagetag 19.15.0.0.0
DBAAS CLI version MAIN
Executing command cswlib deletelocal --imagetag 19.15.0.0.0
Job id: 8b3e71de-4b81-4832-b49c-7f892179bb4f
Log file location: /var/opt/oracle/log/cswLib/deleteLocal/
dbaastools_2022-07-18_10-00-02-AM_73658.log
dbaascli execution completed
```

Related Topics

- Connecting to a Virtual Machine with SSH
  You can connect to the virtual machines in an Oracle Exadata Database Service on Exascale Infrastructure system by using a Secure Shell (SSH) connection.

dbaascli cswlib download

To download available software images and make them available in your Oracle Exadata Database Service on Exascale Infrastructure environment, use the `dbaascli cswlib download` command.

Prerequisites

Run the command as the `root` user.

To use the utility, you must connect to an Oracle Exadata Database Service on Exascale Infrastructure virtual machine.

See, *Connecting to a Virtual Machine with SSH*.

Syntax

```
dbaascli cswlib download --version | --imageTag
[--product]
```

Where:

- `--version` specifies an Oracle home image version
- `--imageTag` specifies the image tag of the image
- `--product` specifies the image type. Valid values: `database` or `grid`

Example 5-4   dbaascli cswlib download --product --imageTag

```
dbaascli cswlib download --product database --imageTag 19.14.0.0.0
```

Example 5-5   dbaascli cswlib download --version 19.9.0.0.0

```
dbaascli cswlib download --product database --imageTag 19.14.0.0.0
```

Related Topics

- Connecting to a Virtual Machine with SSH
  You can connect to the virtual machines in an Oracle Exadata Database Service on Exascale Infrastructure system by using a Secure Shell (SSH) connection.

dbaascli cswlib listLocal

To view the list of locally available Database and Grid Infrastructure images, use the `dbaascli cswlib listLocal` command.

Run the command as the `root` user.

Syntax

```
dbaascli cswLib listLocal [--product <value>]
```

Where:

- `--product` identifies Oracle home product type. Valid values: `database` or `grid`.

Example 5-6 dbaascli cswlib listlocal

```
dbaascli cswlib listlocal
DBAAS CLI version MAIN
Executing command cswlib listlocal
Job id: bc4f047c-0a34-4d4d-a1ea-21ddc2a9c627
Log file location: /var/opt/oracle/log/cswLib/listLocal/
dbaastools_2022-07-18_10-29-53-AM_16077.log
############ List of Available Database Images  #############
1.IMAGE_TAG=12.2.0.1.220419
  IMAGE_SIZE=5GB
  VERSION=12.2.0.1.220419
  DESCRIPTION=12.2 APR 2022 DB Image
2.IMAGE_TAG=18.16.0.0.0
  IMAGE_SIZE=6GB
  VERSION=18.16.0.0.0
  DESCRIPTION=18c OCT 2021 DB Image
3.IMAGE_TAG=19.14.0.0.0
  IMAGE_SIZE=5GB
  VERSION=19.14.0.0.0
  DESCRIPTION=19c JAN 2022 DB Image
dbaascli execution completed
```

Related Topics

- Connecting to a Virtual Machine with SSH
  You can connect to the virtual machines in an Oracle Exadata Database Service on Exascale Infrastructure system by using a Secure Shell (SSH) connection.

dbaascli cswlib showImages

To view the list of available Database and Grid Infrastructure images, use the `dbaascli cswlib showImages` command.

Run the command as the `root` user.

Syntax

```
dbaascli cswlib showImages
[--product]
```

Where:

- `--product` identifies Oracle home product type. Valid values: `database` or `grid`.

Example 5-7 dbaascli cswlib showImages

```
dbaascli cswlib showImages
```

Related Topics

- Connecting to a Virtual Machine with SSH
  You can connect to the virtual machines in an Oracle Exadata Database Service on Exascale Infrastructure system by using a Secure Shell (SSH) connection.

dbaascli database addInstance

To add the database instance on the specified node, use the `dbaascli database addInstance` command.

Prerequisite

- Run the command as the `root` user.

Syntax

```
dbaascli database addInstance --dbname <value> --node <value> [--newNodeSID
<value>]
```

Where:

- `--dbname` specifies Oracle Database name
- `--node` specifies the node name for the database instance
  - `--newNodeSID` specifies SID for the instance to add in the new node

dbaascli database backup

To configure Oracle Database with a backup storage destination, take database backups, query backups, and delete a backup, use the `dbaascli database backup` command.

Prerequisite

- Run the command as the `root` user.

Syntax

```
dbaascli database backup --dbname <value>
        {
            --list
                {
                    [--backupType <value>]
                    | [--json <value>]
                }
            | --start [--level0] [--level1]
                {
                    [--archival --tag <value>]
                    | [--archivelog]
                }
            | --delete --backupTag <value>
            | --status --uuid <value>
            | --getBackupReport
                {
                    --tag <value>
                    | --latest
                }
                --json <value>
            | --configure
                {
                    --configFile <value>
                    | --enableRTRT
                    | --disableRTRT
                }
            | --getConfig [--configFile <value>]
            | --validate [--untilTime <value>]
            | --showHistory [--all]
        }
```

Where:

```
--dbname: Oracle Database name.
--list | --start | --delete | --status | --getBackupReport | --configure | --
getConfig
--list: Returns database backup information.
    [--json: Specify the file name for JSON output.]
--start: Begins database backup.
       [--level0 | --level1 | --archival]
       [--level0: Creates a Level-0 (full) backup. ]
       [--level1: Creates a Level-1 (incremental) backup. ]
       [--archival: Creates an Archival full backup. ]
            --tag: Specify backup tag.
--delete: Deletes Archival backup.
           --backupTag <value>
--status
           --uuid <value>
--getBackupReport: Returns backup report.
           --tag: Specify backup tag.
           --latest: Returns latest backup report (all types of database backup).
           --json: Specify the file name for JSON output.
--configure: Configures database for backup.
           --configFile | --enableRTRT | --disableRTRT
           --configFile: Specify database backup configuration file.
           --enableRTRT: Enables Real Time Redo Transport.
           --disableRTRT: Disables Real Time Redo Transport.
--getConfig: Returns database backup configuration.
           [--configFile: Specify the database backup configuration file.]
--validate: Validates that backups are complete and corruption-free.
           [--untilTime: Validates from closest Level-0 (full) backup until time
provided. Input format: DD-MON-YYYY HH24:MI:SS.]
--showHistory: Displays the history of backup operations.
           [--all: Displays all backup operations.]
```

> **Note:**
>
> `enableRTRT` and `disableRTRT` are applicable only for ZDLRA backup destination on Exadata Database Service on Cloud@Customer.

Example 5-8 Examples

- To get backup configuration for a database *myTestDB*:

  ```
  dbaascli database backup --dbName myTestDB --getConfig --configFile /tmp/
  configfile_1.txt
  ```

- To set backup configuration for a database *myTestDB* by modifying the config file with configuration details:

  ```
  dbaascli database backup --dbName myTestDB --configure --configFile /tmp/
  configfile_1_modified.txt
  ```

- To take backup of the database *myTestDB*:

  ```
  dbaascli database backup --dbName myTestDB --start
  ```

- To query the status of backup request submitted with `uuid` *58fdcae0bd1c11eb92bc020017075151*:

  ```
  dbaascli database backup --dbName myTestDB --status --uuid
  58fdcae0bd1c11eb92bc020017075151
  ```

- To enable RTRT for the database *myTestDB*:

  ```
  dbaascli database backup --dbName myTestDB --configure —enableRTRT
  ```

dbaascli database bounce

To shut down and restart a specified Oracle Exadata Database Service on Exascale Infrastructure database, use the `dbaascli database bounce` command.

Prerequisites

Run the command as the `oracle` user.

Syntax

```
dbaascli database bounce
[--dbname][--rolling <value>]
```

Where:

- `--dbname` specifies the name of the database
- `--rolling` specifies `true` or `false` to bounce the database in a rolling manner. Default value is `false`.

The command performs a database shutdown in immediate mode. The database is then restarted and opened. In Oracle Database 12c or later, all of the PDBs are also opened.

Example 5-9 dbaascli database bounce

```
dbaascli database bounce --dbname dbname
```

dbaascli database changepassword

To change the password of a specified Oracle Database user, use the `dbaascli database changePassword` command. When prompted enter the user name for which you want to change the password and then enter the password.

Prerequisites

Run the command as the `root` or `oracle` user.

Syntax

```
dbaascli database changePassword [--dbname <value>] [--user <value>]
{
  [--prepareStandbyBlob <value> [--blobLocation <value>]] | [--
standbyBlobFromPrimary <value>]
}
[--resume [--sessionID <value>]]
```

Where:

- `--dbname` specifies the name of the Oracle Database that you want to act on
- `--user` specifies the user name whose password change is required
- `--prepareStandbyBlob` specifies `true` to generate a blob file containing the artifacts needed to change the password in a Data Guard environment. Valid values: `true`|`false`
- `--blobLocation` specifies the custom path where blob file will be generated
- `--standbyBlobFromPrimary` specifies the standby blob file, which is prepared from the primary database
- `--resume` specifies to resume the previous execution
  - `--sessionID` specifies to resume a specific session ID

Example 5-10 dbaascli database changePassword

```
dbaascli database changepassword --dbname db19
```

dbaascli database convertToPDB

To convert the specified non-CDB database to PDB, use the `dbaascli database convertToPDB` command.

Syntax

```
dbaascli database convertToPDB --dbname <value> [--cdbName <value>] [--
executePrereqs]
         {
              [--copyDatafiles [--keepSourceDB]]|[backupPrepared]
         }
         [--targetPDBName <value>] [--waitForCompletion <value>] [--resume [--
sessionID <value>]]
```

Where:

- `--dbname` specifies the name of Oracle Database
- `--cdbName` specifies the name of the target CDB in which the PDB will be created. If the CDB does not exist, then it will be created in the same Oracle home as the source non-CDB
- `--executePrereqs` specifies to run only the pre-conversion checks
- `--copyDatafiles` specifies to create a new copy of the data files instead of using the ones from the source database
  `--keepSourceDB` - to preserve the source database after completing the operation.
- `--backupPrepared` - flag to acknowledge that a proper database backup is in place for the non CDB prior to performing the conversion to PDB.
- `--backupPrepared` flag to acknowledge that a proper database backup is in place for the non-CDB prior to performing the conversion to PDB
- `--targetPDBName` specifies the name of the PDB that will be created as part of the operation
- `--waitForCompletion` specifies `false` to run the operation in the background. Valid values: `true|false`
- `--resume` specifies to resume the previous execution
  - `--sessionID` specifies to resume a specific session ID

Example 5-11 dbaascli database convertToPDB

To run pre-conversion prechecks:

```
dbaascli database convertToPDB --dbname ndb19 --cdbname cdb19 --
backupPrepared --executePrereqs
```

To run a full conversion with a copy of the data files from the non-CDB:

```
dbaascli database convertToPDB --dbname tst19 --cdbname cdb19 --copyDatafiles
```

dbaascli database create

To create Oracle Database, use the `dbaascli database create` command. When prompted, enter the `sys` and `tde` passwords.

Use this command to create Oracle Database version 12.1.0.2 or higher with the release update JAN 2021 or higher. For databases with lower versions, it is recommended to use the OCI Console based API.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli database create --dbName {--oracleHome | --oracleHomeName}
[--dbUniqueName <value>]
[--dbSID <value>]
[--createAsCDB <value>]
```

```
[--pdbName <value>]
[--pdbAdminUserName <value>]
[--dbCharset <value>]
[--dbNCharset <value>]
[--dbLanguage <value>]
[--dbTerritory <value>]
[--sgaSizeInMB <value>]
[--pgaSizeInMB <value>]
[--datafileDestination <value>]
[--fraDestination <value>]
[--fraSizeInMB <value>]
[--nodeList <value>]
[--tdeConfigMethod <value>]
[--kmsKeyOCID <value>]
{
            [--resume [--sessionID <value>]]
            | [--revert [--sessionID <value>]]
        }
[--executePrereqs]
[--honorNodeNumberForInstance <value>]
[--lockPDBAdminAccount <value>]
[--dbcaTemplateFilePath <value>]
[--waitForCompletion]
```

Where:

- `--dbname` specifies the name of the database
- `--oracleHome` specifies the location of the Oracle home
- `--oracleHomeName` specifies the name of the Oracle home
- `--dbUniqueName` specifies database unique name
- `--dbSID` specifies the SID of the database
- `--createAsCDB` specifies `true` or `false` to create database as CDB or Non-CDB
- `--pdbName` specifies the name of the PDB
- `--pdbAdminUserName` specify PDB admin user name
- `--dbCharset` specifies database character set
- `--dbNCharset` specifies database national character set
- `--dbLanguage` specifies the database language
- `--dbTerritory` specifies the database territory
- `--sgaSizeInMB` specifies the `sga_target` value in megabyte unit
- `--pgaSizeInMB` specifies the `pga_aggregate_target` value in megabyte unit
- `--datafileDestination` specifies the ASM disk group name to use for database datafiles
- `--fraDestination` specifies ASM disk group name to use for database Fast Recovery Area
- `--fraSizeInMB` specifies the Fast Recovery Area size value in megabyte unit
- `--nodeList` specifies a comma-delimited list of nodes for the database
- `--tdeConfigMethod` specifies TDE configuration method. Valid values: `FILE`, `KMS`

- `--kmsKeyOCID` specifies KMS key OCID to use for TDE. This is applicable only if KMS is selected for TDE
- `--resume` resumes the previous execution
- `--revert` rolls back the previous run
- `--sessionID` resumes or reverts to a specific session ID.
- `--executePrereqs` specifies `yes` to run only the prereqs for this operation. Valid values: `yes` or `no`
- `--honorNodeNumberForInstance` specifies `true` or `false` to indicate instance name to be suffixed with the cluster node numbers. Default value: `true`
- `--lockPDBAdminAccount` specifies `true` or `false` to lock the PDB admin user account. Default value is `true`
- `--dbcaTemplateFilePath` specifies the absolute path of the dbca template name to create the database.
- `--waitForCompletion` specifies `false` to run the operation in the background. Valid values: `true` or `false`

Example 5-12 dbaascli database create

```
dbaascli database create --dbName db19 --oracleHomeName myhome19 --dbSid
db19sid --nodeList node1,node2 --createAsCDB true
```

dbaascli database delete

To delete an Oracle Database, use the `dbaascli database delete` command.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli database delete --dbname <value>
[--deleteArchiveLogs <value>]
[--deleteBackups <value>]
[--precheckOnly <value>]
[--waitForCompletion <value>]
[--force]
[--dbSID <value>]
[--resume [--sessionID <value>]]
```

Where:

- `--dbname` specifies the name of the database.
- `--deleteArchiveLogs` specifies `true` or `false` to indicate deletion of database archive logs.
- `--deleteBackups` specifies `true` or `false` to indicate deletion of database backups.
- `--precheckOnly` specifies `yes` to run only the prechecks for this operation. Valid values: `yes` or `no`.

- `--waitForCompletion` specifies `false` to run the operation in the background. Valid values: `true` or `false`.
- `--force` flag to force delete database.
- `--dbSID` specify database SID.
- `--resume` to resume the previous execution.
- `--sessionID` to resume a specific session id.

Example 5-13 dbaascli database delete

```
dbaascli database delete --dbname db19
```

dbaascli database deleteInstance

To delete the database instance on the specified node, use the `dbaascli database deleteInstance` command.

Prerequisite

- Run the command as the `root` user.

Syntax

```
dbaascli database deleteInstance --dbname <value> --node <value> [--
continueOnUnreachableNode]
```

Where:

- `--dbname` specifies Oracle Database name
- `--node` specifies the node name for database instance
- `--continueOnUnreachableNode` specifies to perform the operation even if the node is unreachable

Example 5-14 database deleteinstance

```
database deleteinstance --node test-node
```

dbaascli database duplicate

To create a database from an active database, use the `dbaascli database duplicate` command.

Prerequisite

- Run the command as the `root` user.

Syntax

```
dbaascli database duplicate --dbName <value> --sourceDBConnectionString
<value>
        {
            --oracleHome <value>
            | --oracleHomeName <value>
        }
[--dbSID <value>]
[--dbUniqueName <value>]
[--sgaSizeInMB <value>]
[--pgaSizeInMB <value>]
[--datafileDestination <value>]
[--fraDestination <value>]
[--fraSizeInMB <value>]
[--sourceDBWalletLocation <value>]
[--nodeList <value>]
        {
            [--resume [--sessionID <value>]]
            | [--revert [--sessionID <value>]]
        }
[--rmanParallelism <value>]
[--rmanSectionSizeInGB <value>]
[--tdeConfigMethod <value>]
[--kmsKeyOCID <value>]
[--sourceDBTdeConfigMethod <value>]
[--sourceDBKmsKeyOCID <value>]
[--executePrereqs <value>]
[--waitForCompletion <value>]
[--skipPDBs <value>]
```

Where:

- `--dbName` specifies Oracle Database name
- `--sourceDBConnectionString` specifies source database connection string in the format of `<scan_name>:<scan_port>/<database_service_name>`
- `--oracleHome` specifies Oracle home location
- `--oracleHomeName` specifies Oracle home name
- `--dbSID` specifies database SID
- `--dbUniqueName` specifies database unique name
- `--sgaSizeInMB` specifies `sga_target` value in mega byte unit
- `--pgaSizeInMB` specifies `pga_aggregate_target` value in mega byte unit
- `--datafileDestination` specifies ASM disk group name to use for database datafiles
- `--fraDestination` specifies ASM disk group name to use for database fast recovery area
- `--fraSizeInMB` specifies fast recovery area size value in mega byte unit
- `--sourceDBWalletLocation` specifies source database TDE wallet file location. This is required to duplicate database from active database
- `--nodeList` specifies a comma-delimited list of nodes for the database
- `--resume` specifies to resume the previous execution
  - `--sessionID` specifies to resume a specific session ID
- `--revert` specifies to rollback the previous execution
  - `--sessionID` specifies to rollback a specific session ID

- `--rmanParallelism` specifies parallelsim value
- `--rmanSectionSizeInGB` specifies RMAN section size in GB
- `--tdeConfigMethod` specifies TDE configuration method. Allowed values are `FILE` and `KMS`.
- `--kmsKeyOCID` specifies KMS key OCID to use for TDE. This is applicable only if KMS is selected for TDE.
- `--sourceDBTdeConfigMethod` specifies source database TDE configuration method. Allowed values are `FILE` and `KMS`.
- `--sourceDBKmsKeyOCID` specifies source database KMS key OCID to use for TDE. This is applicable only if KMS is selected for TDE.
- `--executePrereqs` specifies `yes` to run only the prereqs for this operation. Valid values: `yes|no`
- `--waitForCompletion` specifies `false` to run the operation in background. Valid values: `true|false`
- `--skipPDBs` specifies a comma-delimited list of source database PDB names, which needs to be excluded for the duplicate database operation. Example: pdb1,pdb2...

Example 5-15 dbaascli database duplicate

```
dbaascli database duplicate --sourceDBConnectionString test-user-
scan.dbaastoolslrgsu.dbaastoolslrgvc.oraclevcn.com:1521/
mynew.dbaastoolslrgsu.dbaastoolslrgvc.oraclevcn.com --oracleHome /u02/app/
oracle/product/19.0.0.0/dbhome_2 --dbName newdup --
sourceDBWalletLocation /var/opt/oracle/dbaas_acfs/tmp/prim_wallet
```

dbaascli database getDetails

This command shows the detailed information of a given database e.g. dbname, node information, pluggable databases information etc.

Prerequisites

Run the command as the `root` user or the `oracle` user

Syntax

```
dbaascli database getDetails --dbname <value>
```

Where :

- `--dbname` - Oracle database name.

dbaascli database getPDBs

To view the list of all pluggable databases in a container database, use the `dbaascli database getPDBs` command.

Run the command as the `root` or `oracle`user.

Syntax

```
dbaascli database getPDBs --dbname <value>
```

Where:

- `--dbname` specifies the name of the container database

Example 5-16 dbaascli database getPDBs --dbname

```
dbaascli database getPDBs --dbname apr_db1
```

dbaascli database modifyParameters

To modify or reset initialization parameters for an Oracle Database, use the `dbaascli database modifyParameters` command.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli database modifyParameters --dbname <value>
{
--setParameters <values>[--instance <value>] [--backupPrepared] [--
allowBounce]|
--resetParameters <values> [--instance <value>] [--backupPrepared] [--
allowBounce]
}
--responseFile
[--backupPrepared]
[--instance]
[--allowBounce]
[--waitForCompletion]
```

Where:

- `--dbname` specifies the name of the database.
- `--setParameters` specifies a comma-delimited list of parameters to modify with new values. For example: `parameter1=`*`valueA`*,`parameter2=`*`valueB`*, and so on. For blank values use parameter1=valueA,parameter2='',etc.
- `--resetParameters` specifies a comma-delimited list of parameters to be reset to their corresponding default values. For example, `parameter1`,`parameter2`, and so on.
- `--instance` specifies the name of the instance on which the parameters will be processed. If not specified, then the operation will be performed at the database level.
- `--backupPrepared` acknowledges that a proper database backup is in place prior to modifying critical or sensitive parameters.
- `--allowBounce` grants permission to bounce the database in order to reflect the changes on applicable static parameters.

- `--waitForCompletion` specify false to run the operation in background. Valid values : true|false.]

Example 5-17 dbaascli database modifyParameters

```
dbaascli database modifyParameters --dbname dbname --setParameters
"log_archive_dest_state_17=ENABLE"
```

dbaascli database move

To move the database from one home to another, use the `dbaascli database move` command.

Prerequisites

- Before performing a move operation, ensure that all of the database instances associated with the database are up and running.
- Run the command as the `root` user.

Syntax

```
dbaascli database move
{
  --oracleHome <value> | --oracleHomeName <value>
}
--dbname <value>
[--executePrereqs]
[--resume [--sessionID <value>]]
[--rollback [--sessionID <value>]]
[--skipDatapatch]
[--skipPDBs <value>]
[--skipClosedPDBs]
[--continueWithDbDowntime]
[--allowParallelDBMove]
[--waitForCompletion <value>]
[--nodeList <value>]
```

Where:

- `--oracleHome` specifies Oracle home path
- `--oracleHomeName` specifies the name of Oracle home
- `--dbname` specifies the name of the database
- `--executePrereqs` runs the prerequisite checks and report the results
- `--resume` resumes the previous run
  - `--sessionID` specifies to resume a specific session ID
- `--rollback` rolls the database back to previous home
  - `--sessionID` specifies to resume a specific session ID
- `--skipDatapatch` skips running the datapatch on the databases
- `--skipPdbs` skips running the datapatch on a specified comma-delimited list of PDBs. For example: *pdb1*,*pdb2*...

- `--skipClosedPDBs` skips patching closed PDBs
- `--continueWithDbDowntime` continues patching with database downtime. This option can be used in environments wherein there is only one active instance up and the patching operation can be continued even with a downtime.
- `--allowParallelDBMove` allows database move in parallel.
- `--waitForCompletion` specifies `false` to run the operation in the background. **Valid values:** `true|false`
- `--nodeList` specifies a comma-delimited list of nodes if operation has to be performed on a subset of nodes

Example 5-18 dbaascli database move

```
dbaascli database move --dbname testdb1 --oracleHome /u02/app/oracle/product/
12.1.0/dbhome_2
```

dbaascli database recover

To recover a database, use the `dbaascli database recover` command.

Prerequisite

- Run the command as the `root` user.
- Database must have been configured with backup storage destination details where backups are stored.

Syntax

```
dbaascli database recover --dbname <value>
        {
            --start
                {
                    --untilTime <value>
                    | --untilSCN <value>
                    | --latest
                    | --tag <value>
                }
            | --status --uuid <value>
        }
```

Where:

```
--dbname: Oracle Database name.
     --start | --status
--start: Begins database recovery.
     --untilTime | --untilSCN | --latest | --tag
     --untilTime: Recovers database until time. Input format: DD-MON-YYYY
HH24:MI:SS.
     --untilSCN: Recovers database until SCN.
     --latest: Recovers database to last known state.
     --tag: Recovers database to archival tag.
--status
     --uuid <value>
```

Example 5-19 Examples

- To recover the database *myTestDb* to latest:

  ```
  dbaascli database recover --dbname myTestDb --start --latest
  ```

- To query the status of recovery request submitted with `uuid` *2508ea18be2911eb82d0020017075151*:

  ```
  dbaascli database recover --dbname myTestDb --status --uuid
  2508ea18be2911eb82d0020017075151
  ```

dbaascli database runDatapatch

To patch an Oracle Database, use the `dbaascli database runDatapatch` command.

Prerequisites

- Before performing a `runDatapatch` operation, ensure that all of the database instances associated with the database are up and running.
- Run the command as the `root` user.

Syntax

```
dbaascli database runDatapatch --dbname
[--resume]
    [--sessionID]
[--skipPdbs | --pdbs]
[--executePrereqs]
[--patchList]
[--skipClosedPdbs]
[--rollback]
```

Where:

- `--dbname` specifies the name of the database
- `--resume` resumes the previous run
  - `--sessionID` specifies to resume a specific session ID
- `--skipPdbs` skips running the datapatch on a specified comma-delimited list of PDBs. For example: *pdb1*,*pdb2*...
- `--pdbs` runs the datapatch only on a specified comma-delimited list of PDBs. For example: *pdb1*,*pdb2*...
- `--executePrereqs` runs prerequisite checks
- `--patchList` applies or rolls back the specified comma-delimited list of patches. For example: *patch1*,*patch2*...
- `--skipClosedPdbs` skips running the datapatch on closed PDBs
- `--rollback` rolls back the patches applied

```
dbaascli database runDatapatch --dbname db19
```

dbaascli database createTemplate

To create database templates (DBCA templates) that can subsequently be used to create databases, use the `dbaascli database createTemplate` command.

Prerequistes:

Run the command as the `root` user.

Syntax

Create a new DBCA template from the specified database.

```
dbaascli database createTemplate --dbname <value> --templateLocation <value>
[--templateName <value>]
[--rmanParallelism <value>]
```

Where:

- `--dbname` specifies the name of the database.
- `--templateLocation` specifies the template name.
- `--rmanParallelism` specifies the parallelsim value.

dbaascli database start

To start an Oracle Database, use the `dbaascli database start` command.

Prerequisites

Run the command as the `root` user.

Syntax

```
dbaascli database start
[--dbname]
[--mode]
```

Where:

- `--dbname` specifies the name of the database
- `--mode` specifies mount or nomount to start database in the corresponding mode

The command starts and opens the database. In Oracle Database 12c or later, all of the PDBs are also opened.

Example 5-20 dbaascli database start

```
dbaascli database start --dbname dbname --mode mount
```

dbaascli database status

To check the status of an Oracle Database, use the `dbaascli database status` command.

Prerequisites

Run the command as the `root` user.

Syntax

```
dbaascli database status
[--service][--dbname]
[--user]
[--password]
```

Where:

- `--service` specifies the name of the service
- `--dbname` specifies the name of the database
- `--user` specifies the user name of the service
- `--password` specifies the password of the user

Output from the command includes the open mode of the database, the software release and edition of the database, and release version of other software components.

Example 5-21 dbaascli database status

```
dbaascli database status --dbname db19
```

dbaascli database stop

To stop an Oracle Database, use the `dbaascli database stop` command.

Prerequisites

Run the command as the `root` user.

Syntax

```
dbaascli database stop
[--dbname <value>]
[--mode <value>]
```

Where:

- `--dbname` specifies the name of the database that you want to stop
- `--mode` specifies the mode of the database. Valid values: `abort`, `immediate`, `normal`, `transactional`

The command performs a database shutdown in immediate mode. No new connections or new transactions are permitted. Active transactions are rolled back, and all connected users are disconnected.

Example 5-22 dbaascli database stop

```
dbaascli database stop --dbname db19
```

dbaascli database upgrade

To upgrade an Oracle Database, use the `dbaascli database upgrade` command.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli database upgrade --dbname <value>
{--targetHome <value> | --targetHomeName <value>}
{ [--executePrereqs | --postUpgrade | --rollback]}
{[--standBy | --allStandbyPrepared]}
{[--upgradeOptions <value>]  | [--standBy]}
[--removeGRP]
[--increaseCompatibleParameter]
[--resume [--sessionID <value>]]
[--waitForCompletion <value>]
```

Where:

- `--dbname` (mandatory) specifies the name of the database.
- `--targetHome` specifies the target Oracle home location
- `--targetHomeName` specifies the name of the target Oracle Database home
- `--standBy` use this option to upgrade standby databases in Data Guard configurations
- `--allStandbyPrepared` required for Data Guard configured primary databases. Flags to acknowledge that all the required operations are performed on the standby databases prior to upgrading primary database
- `--removeGRP` automatically removes the Guaranteed Restore Point (GRP) backup only if the database upgrade was successful
- `--increaseCompatibleParameter` automatically increases the compatible parameter as part of the database upgrade. The parameter will get increased only if the database upgrade was successful
- `--executePrereqs` runs only the preupgrade checks
- `--postUpgrade` use this option if postupgrade fails and needs to rerun the postupgrade steps
- `--rollback` reverts an Oracle Database to its original Oracle home
- `--upgradeOptions` use this option to pass DBUA-specific arguments to perform the Oracle Database upgrade. Refer to the corresponding Oracle documentation for the supported arguments and options.
  `--standby`
- `--resume` to resume the previous execution
- `--sessionID` to resume a specific session id.

- `--waitForCompletion` specify false to run the operation in background. Valid values : true|false.

Example 5-23 dbaascli database upgrade pre-upgrade requisite checks

```
dbaascli database upgrade --dbbname dbname --targetHome Target Oracle home
location --executePrereqs
```

dbaascli dataguard prepareStandbyBlob

To generate a blob file containing various files that are required on the standby site in case of a dataguard environment, use the `dbaascli dataguard prepareStandbyBlob` command.

Run the command as the `root` or `oracle` user.

Syntax

```
dbaascli dataguard prepareStandbyBlob --dbname <value> --blobLocation <value>
```

Where:

- `--dbname` specifies the Oracle Database name
- `--blobLocation` specifies the custom directory location where the standby blob file will be generated in a Data Guard environment

dbaascli dataguard updateDGConfigAttributes

To update Data Guard automation attributes across all the cluster nodes, use the `dbaascli dataguard updateDGConfigAttributes` command.

Run the command as the `root` or `oracle`user.

Syntax

```
dbaascli dataguard updateDGConfigAttributes --attributes <value>
```

Where:

- `--attributes` contains the Data Guard automation attributes that are to be modified. Accepts comma-delimited values in the format <*attribute*=*value*>. Attributes must be predefined in the Data Guard configuration file.

dbaascli dbhome create

To create an Oracle Database home of desired version, use the `dbaascli dbhome create` command.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli dbhome create --version <value>
[--oracleHome <value>]
[--oracleHomeName <value>]
[--enableUnifiedAuditing <value>]
[--imageTag <value>]
[--ImageLocation <value>
```

Where:

- `--version` specifies the version of Oracle Home specified as five numeric segments separated by periods, for example, 19.12.0.0.0
- `--oracleHome` specifies the location of Oracle home
- `--oracleHomeName` specifies user-defined Oracle home name. If not provided, then the default name will be used
- `--enableUnifiedAuditing` specifies `true` or `false` to enable or disable unified auditing link option in Oracle home
- `--imageTag` specifies Oracle home image tag
- `--imageLocation` - path of the image to be used.
- `--waitForCompletion` specifies `false` to run the operation in background. Valid values: `true` or `false`.

Example 5-24 dbaascli dbhome create

```
dbaascli dbhome create --version 19.11.0.0.0
```

Alternatively, `dbaascli dbhome create --version 19.8.0.0.0.0 --imageTag 19.8.0.0.0` for cases where image tags are different from version.

dbaascli dbHome delete

To delete a given Oracle Database home, use the `dbaascli dbHome delete` command.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli dbHome delete
{ --oracleHome <value>
| --oracleHomeName <value> } [--resume [--sessionID <value>]]
```

Where:

- `--oracleHome` specifies the location of the Oracle home
- `--oracleHomeName` specifies the name of the Oracle home
- `--resume` resumes the previous execution

  - `--sessionID` specifies to resume a specific session ID

dbaascli dbhome getDatabases

To view information about all Oracle Databases running from a given database Oracle home, use the `dbaascli dbHome getDatabases` command. Specify either the Oracle home location or Oracle home name.

Run the command as the `root` user.

Syntax

```
dbaascli dbHome getDatabases
{ --oracleHomeName value | --oracleHome value }
```

Where:

- `--oracleHomeName` specifies user-defined Oracle home name
- `--oracleHome` specifies the location (path) of Oracle home

Example 5-25   dbaascli dbHome getDatabases --oracleHome

```
dbaascli dbHome getDatabases --oracleHome /u02/app/mar_home/
```

dbaascli dbHome getDetails

To view information about a specific Oracle home, use the `dbaascli dbHome getDetails` command. Specify either the Oracle home location or Oracle home name.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli dbHome getDetails
{ --oracleHomeName value | --oracleHome value }
```

Where:

- `--oracleHomeName` specifies user-defined Oracle home name
- `--oracleHome` specifies the location of Oracle home

Example 5-26   dbaascli dbHome getDetails - using Oracle home location

```
dbaascli dbHome getDetails --oracleHome /u02/app/home_db19c/
```

Example 5-27   dbaascli dbHome getDetails - using Oracle home name

```
dbaascli dbHome getDetails --oracleHomeName home_db19c
```

dbaascli dbHome patch

To patch Oracle home from one patch level to another, use the `dbaascli dbHome patch` command.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli dbHome patch --oracleHome | --oracleHomeName
--targetVersion
[--resume]
    [--sessionID]
[--continueWithDbDowntime]
[--skipUnreachableNodes]
[--nodes]
[--executePrereqs]
[--skipDatapatch]
[--imageFilePath]
[--skipPDBs]
[--skipClosedPDBs]
[--rollback]
```

Where:

- `--oracleHome` specifies the path of Oracle home
- `--oracleHomeName` specifies the name of Oracle home
- `--targetVersion` specifies the target version of Oracle Home specified as five numeric segments separated by periods, for example, 19.12.0.0.0.
- `--resume` resumes the previous run
  - `--sessionID` specifies to resume a specific session ID
- `--continueWithDbDowntime` continues patching with database downtime. This option can be used in environments wherein there is only one active instance up and the patching operation can be continued even with a downtime.
- `--skipUnreachableNodes` skips operation on unreachable nodes
- `--nodes` specifies a comma-delimited list of nodes if patching has to be performed on a subset of nodes
- `--executePrereqs` runs prereqs
- `--skipDatapatch` skips running `datapatch` on the databases
- `--imageFilePath` specifies the absolute path of the image file to be used
- `--skipPDBs` skips running the datapatch on a specified comma-delimited list of PDBs. For example: *cdb1:pdb1*,*cdb2:pdb2*, and so on
- `--skipClosedPdbs` skips running `datapatch` on closed PDBs
- `--rollback` rolls back patched Oracle home.

Example 5-28 dbaascli dbhome patch

```
dbaascli dbhome patch --targetVersion 19.10.0.0.0 --oracleHome /u02/app/
oracle/product/19.0.0.0/dbhome_2
```

dbaascli dbimage purge

The `dbimage purge` command removes the specified software image from your Oracle Exadata Database Service on Exascale Infrastructure environment.

Connect to the compute node as the `opc` user and execute this command as the `root` user.

```
# dbaascli dbimage purge --version software_version --bp software_bp [--cdb ( yes | no )]
```

In the preceding command:

- *`software_version`* — specifies the Oracle Database software version. For example, `11204`, `12102`, `12201`, `18000`, `19000`.
- *`software_bp`* — identifies the bundle patch release. For example, `APR2018`, `JAN2019`, `OCT2019`, and so on.
- `--cdb` — optionally specifies whether to remove the software image that supports the Oracle multitenant architecture. Default is `yes`. If you specify `--cdb no`, then the software image that contains binaries to support non-container databases (non-CDB) is removed.

If the command will remove a software image that is not currently available in the software image library, and therefore cannot be downloaded again, then the command pauses and prompts for confirmation.

You cannot remove the current default software image for any software version. To avoid this restriction, you must make another software image the current default.

dbaascli diag collect

To collect diagnostics, use the `dbaascli diag collect` command.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli diag collect [--components <value>] [--startTime <value>] [--endTime
<value>] [--nodes <value>] [--dbNames <value>]
        {
            [--objectStoreBucketUri <value>]
            | [--destLocation <value>]
        }
        [--waitForCompletion <value>]
```

Where:

- `--components` specifies a list of components for log collection.
  Valid values:
  - db

  - gi
  - os
  - dbaastools
  - all
- `--startTime` specifies the start time for log collection. Valid date and time format: `YYYY-MM-DDTHH24:MM:SS`
- `--endTime` specifies the end time for log collection. Valid date and time format: `YYYY-MM-DDTHH24:MM:SS`
- `--nodes` specifies a comma-delimited list of nodes to collect logs
- `--dbNames` specifies the database name for which to collect logs. You can specify only one database name.
- `--objectStoreBucketURI` specifies an Object Storage service pre-authenticated request (PAR) URL used to upload collected logs. Logs are collected from Guest VM. For more information, see *Using Pre-Authenticated Requests*.
- `--destLocation` specifies the location on Guest VM to collect logs. Default: `/var/opt/oracle/dbaas_acfs`
- `--waitForCompletion` Values: `true|false`. Default `true`. Specify `false` to run in the background.

Related Topics

- Using Pre-Authenticated Requests
- Collecting Tooling Log Data Examples
  The dbaascli dbaascli diag collect command uses the syntax shown below to collect tooling log data:

dbaascli diag healthCheck

To run diagnostic health checks, use the `dbaascli diag healthCheck` command.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli diag healthCheck
[--destLocation]
[--nodes]
[--objectStoreBucketURI]
```

Where:

- `--destLocation` specifies the location on Guest VM to collect logs. Default: `/var/opt/oracle/dbaas_acfs`
- `--nodes` specifies a comma-delimited list of nodes to collect logs
- `--objectStoreBucketURI` specifies an Object Storage service pre-authenticated request (PAR) URL used to upload collected logs. Logs are collected from Guest VM. For more information, see *Using Pre-Authenticated Requests*.

Related Topics

- Using Pre-Authenticated Requests

dbaascli grid configureTCPS

To configure TCPS for the existing cluster, use the `dbaascli grid configureTCPS` command.

Prerequisite

Run the command as the `root` user.

Syntax

> **Note:**
>
> By default, TCPS is enabled for databases on Oracle Exadata Database Service on Dedicated Infrastructure systems.

> **Note:**
>
> TCPS is not enabled for databases on Exadata Database Service on Cloud@Customer systems. To enable TCPS for a given database, update the database specific `sqlnet.ora` file with `WALLET_LOCATION = (SOURCE=(METHOD=FILE)(METHOD_DATA=(DIRECTORY=/var/opt/oracle/dbaas_acfs/grid/tcps_wallets)))` on all database nodes and then bounce the database. This will enable TCPS usage for the database. However, enabling TCPS will cause ZDLRA connection to fail. On Exadata Database Service on Cloud@Customer systems, you can enable either ZDLRA or TCPS configuration. Enabling both ZDLRA and TCPS simultaneously will not work.

```
dbaascli grid configureTCPS
[--pkcs12WalletFilePath]
[--caCertChain]
[--precheckOnly]
[--serverCert]
[--privateKey]
[--certType]
[--privateKeyPasswordProtected]
```

Where:

- `--pkcs12WalletFilePath` specifies the absolute path of the certificate file, which is in the `pkcs12` wallet format
- `--caCertChain` concatenated list of certs, containing intermediate CA's and root CA certs
- `--precheckOnly` specifies `yes` to run only the prechecks for this operation. Valid values: `yes` or `no`.
- `--serverCert` specifies the path of PEM certificate to use or rotate for TCPS configuration.
- `--privateKey` specifies the path of the private key file of the certificate.

- --certType type of the cert to be added to the Grid Infrastructure wallet. Accepted values are: SELF_SIGNED_CERT, CA_SIGNED_CERT, or PKCS12_CERT. Default: SELF_SIGNED_CERT
- --privateKeyPasswordProtected specifies if the private key is password protected or not. Valid values: true or false. Default: true.

Example 5-29 dbaascli grid configureTCPS

To configure grid using self-signed certificate:

```
dbaascli grid configureTCPS
```

To configure grid using CA-signed certificate:

```
dbaascli grid configureTCPS --cert_type CA_SIGNED_CERT --server_cert /tmp/
certs/server_cert.pem --ca_cert_chain /tmp/certs/ca.pem --private_key /tmp/
certs/encrypted_private.key --private_key_password_protected false
```

dbaascli grid patch

To patch Oracle Grid Infrastructure to the specified minor version, use the dbaascli grid patch command.

Prerequisites

Run the command as the root user.

Syntax

```
dbaascli grid patch --targetVersion <value>
[--containerURL <value>]
[--executePrereqs]
[--nodeList <value>]
[--rollback]
[--resume [--sessionID <value>]]
[--continueWithDbDowntime]
        {
            [--createImage [--createImageDir <value>]]
            | [--imageFile <value>]
        }
[--waitForCompletion <value>]
```

Where:

- --targetVersion specifies the target version of Oracle Home specified as five numeric segments separated by periods (e.g. 19.12.0.0.0)
- --containerURL specifies custom URL for fetching Grid Infrastructure image
- --executePrereqs option to run prereqs
- --nodeList specifies a comma-delimited list of nodes if patching has to be performed on a subset of nodes
- --rollback specifies to roll back patched Oracle home
- --resume resumes the previous run

  - `--sessionID` specifies to resume a specific session ID
- `--continueWithDbDowntime` continues patching with database downtime. This option can be used in environments wherein there is only 1 active instance up and the patching operation can be continued even with a downtime.
- `--createImage` creates an image from a copy of the active Grid home, patched to the specified target version
  - `--createImageDir` specifies fully qualified path of the directory where the image is to be created
- `--imageFile` specifies fully qualified path of the image to be used
- `--waitForCompletion` specifies `false` to run the operation in background. Valid values: `true|false`

Example 5-30 dbaascli grid patch

```
dbaascli grid patch --targetVersion 19.12.0.0.0
```

dbaascli grid removeTCPSCert

To remove existing TCPS certificates from Grid Infrastructure wallet, use the `dbaascli grid removeTCPSCert` command.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli grid removeTCPSCert --subject <value>
 {
   --userCert | --trustedCert | --requestedCert
 }
 [--serialNumber <value>] [--executePrereqs] [--resume [--sessionID <value>]]
[--bounceListeners]
```

Where:

- `--subject` specifies subject of the certificate
- `--userCert` flag to indicate user certificate
- `--trustedCert` flag to indicate trusted certificate
- `--requestedCert` flag to indicate requested certificate
- `--serialNumber` specifies the serial number of the certificate
- `--executePrereqs` runs the prerequisite checks and reports the results
- `--resume` resumes the previous run
  - `--sessionID` specifies to resume a specific session ID
- `--bounceListeners` flag to bounce the Grid Infrastructure listener and scan listener

dbaascli grid rotateTCPSCert

To rotate TCPS certificates, use the dbaascli grid rotateTCPSCert command.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli grid rotateTCPSCert
[--pkcs12WalletFilePath]
[--caCertChain]
[--precheckOnly]
[--serverCert]
[--privateKey]
[--certType]
[--privateKeyPasswordProtected]
```

Where:

- `--pkcs12WalletFilePath` specifies the absolute path of the certificate file, which is in the `pkcs12` wallet format
- `--caCertChain` concatenated list of certs, containing intermediate CA's and root CA certs
- `--precheckOnly` specifies `yes` to run only the prechecks for this operation. Valid values: `yes` or `no`.
- `--serverCert` specifies the path of PEM certificate to use or rotate for TCPS configuration.
- `--privateKey` specifies the path of the private key file of the certificate.
- `--certType` type of the cert to be added to the Grid Infrastructure wallet. Accepted values are: `SELF_SIGNED_CERT`, `CA_SIGNED_CERT`, or `PKCS12_CERT`. Default: `SELF_SIGNED_CERT`
- `--privateKeyPasswordProtected` specifies if the private key is password protected or not. Valid values: `true` or `false`. Default: `true`.

Example 5-31 dbaascli grid rotateTCPSCert

To rotate cert using self-signed certificate (default option):

```
dbaascli grid rotateTCPSCert
```

To rotate cert using CA-signed certificate:

```
dbaascli grid rotateTCPSCert --cert_type CA_SIGNED_CERT --server_cert /tmp/
certs/server_cert.pem --ca_cert_chain /tmp/certs/ca.pem --private_key /tmp/
certs/encrypted_private.key --privateKeyPasswordProtected true
```

dbaascli grid upgrade

To upgrade Oracle Grid Infrastrucure from one major version to another, use the `dbaascli grid upgrade` command.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli grid upgrade --version
[--resume]
[--executePrereqs]
[--containerURL]
[--softwareOnly]
[--targetHome]
[--revert]
```

Where:

- `--version` specifies the target version
- `--resume` resumes the previous run
- `--executePrereqs` runs prereqs for Grid Infrastrucure upgrade
- `--containerUrl` specifies the custom URL for fetching Grid Infrastrucure image
- `--softwareOnly` installs only the Grid Infrastructure software
- `--targetHome` specifies the path of existing target Grid home
- `--revert` reverts failed run

Example 5-32 dbaascli grid upgrade

```
daascli grid upgrade --version 19.11.0.0.0 --executePrereqs
DBAAS CLI version MAIN
Executing command grid upgrade --version 19.11.0.0.0 --executePrereqs
```

dbaascli job getStatus

To view the status of a specified job, use the `dbaascli job getStatus` command.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli job getStatus --jobID
```

Where:

- `--jodID` specifies the job ID

Example 5-33 dbaascli job getStatus

```
dbaascli job getStatus --jobID 13c82031-f202-41b7-9aef-f4a71df0f551
DBAAS CLI version MAIN
Executing command job getStatus --jobID 13c82031-f202-41b7-9aef-f4a71df0f551
{
  "jobId" : "13c82031-f202-41b7-9aef-f4a71df0f551",
  "status" : "Success",
  "message" : "database create job: Success",
  "createTimestamp" : 1628095442431,
  "updatedTime" : 1628095633660,
  "description" : "Service job report for operation database create",
  "appMessages" : {
    "schema" : [ ],
    "errorAction" : "SUCCEED_AND_SHOW"
  },
  "resourceList" : [ ],
  "pct_complete" : "100"
}
```

dbaascli patch db apply

Note:

`dbaascli patch db prereq` and `dbaascli patch db apply` commands have been deprecated in `dbaascli` release 21.2.1.2.0, and replaced with `dbaascli grid patch`, `dbaascli dbhome patch`, and `dbaascli database move` commands.
For more information, see:

- `dbaascli grid patch`
- `dbaascli dbhome patch`
- `dbaascli database move`
- *Patching Oracle Grid Infrastructure and Oracle Databases Using dbaascli*

dbaascli patch db prereq

> **Note:**
>
> `dbaascli patch db prereq` and `dbaascli patch db apply` commands have been deprecated in `dbaascli` release 21.2.1.2.0, and replaced with `dbaascli grid patch`, `dbaascli dbhome patch`, and `dbaascli database move` commands.
> For more information, see:
>
> - `dbaascli grid patch`
> - `dbaascli dbhome patch`
> - `dbaascli database move`
> - *Patching Oracle Grid Infrastructure and Oracle Databases Using dbaascli*

dbaascli pdb backup

To backup a pluggable database (PDB), query PDB backups, and delete a PDB backup, use the `dbaascli pdb backup` command.

Prerequisite

- Run the command as the `root` user.

Syntax

```
dbaascli pdb backup --pdbName <value> --dbname <value>
        {
            --start
                {
                    [--level1]
                    | [--archival --tag <value>]
                }
            | --delete --backupTag <value>
            | --status --uuid <value>
            | --getBackupReport --json <value> --tag <value>
            | --list [--json <value>]
        }
```

Where:

```
--pdbName: PDB name.
--dbname: Oracle Database name.
--start | --delete | --status | --getBackupReport | --list
--start: Begins PDB backup.
      [--level1 | --archival]
      [--level1: Creates a Level-1 (incremental) backup.]
      [--archival: Creates an archival full backup.]
         --tag: Specify backup tag.
--delete: Deletes archival backup.
         --backupTag: Specify backup tag to delete.
```

```
--status
         --uuid <value>
--getBackupReport: Returns backup report.
        --json: Specify the file name for JSON output.
        --tag: Specify backup tag.
--list: Returns PDB backup information.
        [--json: Specify the file name for JSON output.]
```

Example 5-34   Examples

- To take level1 backup for a PDB *pdb1* in a CDB *myTestDb*:

  ```
  dbaascli pdb backup --dbname myTestDb --pdbName pdb1 --start --level1
  ```

- To query the status of PDB backup request submitted with `uuid` *eef16b26361411ecb13800163e8e4fac*:

  ```
  dbaascli pdb backup --dbname myTestDb --pdbName pdb1 --status --uuid 
  eef16b26361411ecb13800163e8e4fac
  ```

Related Topics

- Connecting to a Virtual Machine with SSH
  You can connect to the virtual machines in an Oracle Exadata Database Service on Exascale Infrastructure system by using a Secure Shell (SSH) connection.

dbaascli pdb bounce

To bounce a pluggable database (PDB), use the `dbaascli pdb bounce` command.

Prerequisite

Run the command as the `oracle` user.

Syntax

```
dbaascli pdb bounce --dbname --pdbName | --pdbUID
[-openMode]
```

Where:

- `--dbname` specifies the name of the container database that hosts the PDB
- `--pdbName` specifies the name of the PDB
- `--pdbUID` specifies the identifier of the PDB
- `--openMode` specifies the target `OPEN MODE` of PDB

Example 5-35   dbaascli pdb bounce

```
dbaascli pdb bounce --dbname cdb_name --pdbName pdb name associated with the 
CDB
```

```
dbaascli pdb bounce --dbname cdb_name --pdbUID con_uid of that pdb
```

Optional:

- `--openMode READ_WRITE`
- `--openMode READ_ONLY`

dbaascli pdb close

To close a pluggable database (PDB), use the `dbaascli pdb close` command.

Prerequisite

Run the command as the `oracle` user.

Syntax

```
dbaascli pdb close --dbname --pdbName | --pdbUID
```

Where:

- `--dbname` specifies the name of the container database that hosts the PDB.
- `--pdbname` specifies the name of the PDB that you want to close.
- `--pdbUID` specifies the identifier of the PDB

Upon successful completion of running this command, the PDB is closed on all of the container database instances.

Example 5-36 dbaascli pdb close

```
dbaascli pdb close --dbname cdb name --pdbName pdb name associated with the
CDB
```

```
dbaascli pdb close --dbname cdb name --pdbUID con_uid of that pdb
```

dbaascli pdb getConnectString

To display Oracle Net connect string information for a pluggable database (PDB) run the `dbaascli pdb getConnectString` command.

Prerequisite

Run the command as the `oracle` user.

Syntax

```
dbaascli pdb getConnectString --dbname --pdbName | --pdbUID
```

Where:

- `--dbname` specifies the name of the container database that hosts the PDB
- `--pdbname` specifies the name of the PDB for which you want to display connect string information
- `--pdbUID` specifies the identifier of the PDB

Example 5-37 dbaascli pdb getConnectString

```
dbaascli pdb getConnectString --dbname dbname --pdbName pdbName
```

dbaascli pdb create

To create a new pluggable database (PDB), use the `dbaascli pdb create` command.

Prerequisite

Run the command as the `oracle` user.

Syntax

```
dbaascli pdb create --pdbName <value> --dbName <value>
[--maxCPU <value>]
[--maxSize <value>]
[--pdbAdminUserName <value>]
[--lockPDBAdminAccount <value>]
[--resume [--sessionID <value>]]
[--executePrereqs <value>]
[--waitForCompletion <value>]
[--blobLocation |--standbyBlobFromPrimary <value>]
```

Where:

- `--pdbName` specifies the name of the new PDB that you want to create
- `--dbName` specifies the name of the container database that hosts the new PDB
- `--maxCPU` optionally specifies the maximum number of CPUs that are available to the PDB. Setting this option is effectively the same as setting the `CPU_COUNT` parameter in the PDB
- `--maxSize` optionally specifies the maximum total size of data files and temporary files for tablespaces belonging to the PDB. Setting this option is effectively the same as setting the `MAXSIZE PDB` storage clause in the `CREATE PLUGGABLE DATABASE` SQL command. You can impose a limit by specifying an integer followed by a size unit (`K`, `M`, `G`, or `T`), or you can specify `UNLIMITED` to explicitly enforce no limit
- `--pdbAdminUserName` specifies the new PDB admin user name
- `--lockPDBAdminAccount` specifies `true` or `false` to lock the PDB admin user account. Default value is `true`.
- `--resume` resumes the previous run
  - `--sessionID` specifies to resume a specific session ID
- `--executePrereqs` specifies `yes` to run only the prereqs for this operation. Valid values: `yes` or `no`
- `--waitForCompletion` specifies `false` to run the operation in the background. Valid values: `true` or `false`
- `--blobLocation` custom directory location where the standby blob file will be generated in a DG environment.
- `--standbyBlobFromPrimary` specifies the location of the standby blob file, which is prepared from the primary database. This is required only for standby database PDB operations.

> **Note:**
>
> the parameters`blobLocation` and `standbyBlobFromPrimary` are mutually exclusive.

During the PDB creation process, you are prompted to specify the administration password for the new PDB.

Example 5-38 dbaascli pdb create

To create a PDB from seed in a standard database in a non-Data Guard environment:

```
dbaascli pdb create --dbName db721 --pdbName new_pdb1 --maxsize 5G --maxcpu 2
```

To create PDB in Data Guard environment:

```
dbaascli pdb create --dbName db721 --pdbName new_pdb1
```

```
dbaascli pdb create --dbName db721 --pdbName new_pdb1 --
standbyBlobFromPrimary /tmp/send_db721.tar
```

dbaascli pdb createSnapshot

To create a snapshot of a given pluggable database (PDB), use the `dbaascli pdb createSnapshot` command.

Prerequisite

Run the command as the `oracle` user.

Syntax

```
dbaascli pdb createSnapshot
{
    --pdbName <value> | --pdbUID <value>
}
--dbName <value>
--snapshotName <value>
[--pdbAdminUserName <value>]
[--executePrereqs] [--resume [--sessionID <value>]]
[--waitForCompletion true|false]
```

Where:

- `--pdbName` specifies the PDB name from which to create the snapshot.
- `--pdbUID` specifies the user ID (UID) of the PDB from which to create the snapshot.
- `--dbName` Oracle Database name.
- `--snapshotName` specifies the PDB snapshot name.
- `--pdbAdminUserName` specifies the PDB administrator user name.
- `--executePrereqs` runs the prerequisite checks and reports the results.

- `--resume [--sessionID <value>]` resumes the previous operation. It can take the flag `--sessionID <value>` to specify to resume from a specific session ID (`<value>`).
- `--waitForCompletion true|false` specifies whether to run the operation in foreground (`true`) or background (`false`). Valid values: `true`, `false`.

Example 5-39 dbaascli pdb createSnapshot

In the following example, a snapshot is created from the database named `db721`, in the PDB name `pdb1`. The snapshot name that is given is `snap1`.

```
dbaascli pdb createSnapshot --dbName db721 --pdbName pdb1 --snapshotName snap1
```

dbaascli pdb configureSnapshot

To configure automatic snapshots for a given pluggable database (PDB), use the `dbaascli pdb configureSnapshot` command.

Prerequisite

Run the command as the `oracle` user.

Syntax

```
dbaascli pdb configureSnapshot
{
    --pdbName <value> | --pdbUID <value>
}
--dbName <value>
--snapshotIntervalInMins <value>
[--executePrereqs]
[--maxPDBSnapshots <value>]
[--waitForCompletion <value>]
```

Where:

- `--pdbName <value>` specifies the name of the PDB for which automatic snapshot configuration will be set.
- `--pdbUID <value>` specifies the user ID (UID) of the PDB for which automatic snapshot configuration will be set.
- `--dbName` Oracle Database name.
- `--snapshotIntervalInMins <value>` specifies the interval, in minutes, for when automatic PDB snapshots will be taken.
- `--executePrereqs` runs the prerequisite checks and reports the results.
- `--maxPDBSnapshots <value>]` specifies the maximum number of snapshots to create for the given PDB. .
- `--waitForCompletion true|false` specifies whether to run the operation in foreground (`true`) or background (`false`). Valid values: `true`, `false`.

Example 5-40 dbaascli pdb configureSnapshot

In the following example, an automatic snapshot plan is configured on the database named `db721`, in the PDB name `pdb1`. The snapshot interval is set to run automatic snapshot creation every 60 minutes.

```
dbaascli pdb configureSnapshot --dbName db721 --pdbName pdb1 --
snapshotIntervalInMins 60
```

dbaascli pdb delete

To delete a pluggable database (PDB) run the `dbaascli pdb delete` command.

Prerequisite

Run the command as the `oracle` user.

Syntax

```
dbaascli pdb delete --dbName value
{ --pdbName value | --pdbUID value }
[--executePrereqs value]
[--waitForCompletion value]
[--resume [--sessionID value]]
[--allStandbyPrepared]
[--cleanupRelocatedPDB]
```

Where:

- `--dbName` specifies the name of the container database that hosts the PDB
- `--pdbName` specifies the name of the PDB that you want to delete
- `--pdbUID` specifies the UID of the PDB that you want to delete
- `--executePrereqs` specifies `yes` to run only the prereqs for this operation. Valid values: `yes` or `no`
- `--waitForCompletion` specifies `false` to run the operation in the background. Valid values: `true` or `false`
- `--resume` specifies to resume the previous execution
  - `--sessionID` specifies to resume a specific session ID
- `--allStandbyPrepared` specifies to confirm that the operation has been successfully run on all the standby databases
- `--cleanupRelocatedPDB` - option to cleanup source database after a PDB has been relocated.

Example: dbaascli pdb delete
To delete a PDB from a standard database in a non-Data Guard environment or from Standby database in Data Guard environment.

```
dbaascli pdb delete --dbName db721 --pdbName pdb1
```

To create PDB from Primary database in Data Guard environment:

```
dbaascli pdb create --dbName db721 --pdbName pdb1 --allStandbyPrepared
```

dbaascli pdb deleteSnapshot

To delete a snapshot of a given pluggable database (PDB), use the `dbaascli pdb deleteSnapshot` command.

Prerequisite

Run the command as the `oracle` user.

Syntax

```
dbaascli pdb deleteSnapshot
{
    --pdbName <value> | --pdbUID <value>
}
{
    --snapshotName <value> | --snapshotUID <value>
}
--dbName <value>
[--executePrereqs]
[--waitForCompletion <value>
[--resume [--sessionID <value>]
]
]
```

Where:

- `--pdbName <value>` specifies the name of the PDB for which automatic snapshot configuration will be set.
- `--pdbUID <value>` specifies the user ID (UID) of the PDB for which automatic snapshot configuration will be set.
- `--snapshotName <value>` specifies the name of the PDB snapshot that you want to delete.
- `--snapshotUID <value>` specifies the user ID (UID) of the PDB snapshot that you want to delete.
- `--dbName` specifies the Oracle Database name.
- `--executePrereqs` runs the prerequisite checks and reports the results.
- `--waitForCompletion true|false` specifies whether to run the operation in foreground (`true`) or background (`false`). Valid values: `true`, `false`.
- `--resume [sessionID <value>]` specifies to resume the previous operation. To specify resuming from a particular session ID, add the flag sessionID, and provide the session ID number.

Example 5-41 dbaascli pdb configureSnapshot

In the following example, the PDB snapshot `snap1` is specified for deletion in the PDB named `pdb1`, for the database named `db721`:

```
dbaascli pdb deleteSnapshot --dbName db721 --pdbName pdb1 --snapshotName snap1
```

dbaascli pdb getDetails

To view details of a pluggable database (PDB), use the `dbaascli pdb getDetails` command.

Prerequisite

Run the command as the `oracle` user.

Syntax

```
dbaascli pdb getDetails --dbname --pdbName | --pdbUID
```

Where:

- `--dbname` specifies the name of the container database that hosts the PDB
- `--pdbname` specifies the name of the PDB that you want to delete
- `--pdbUID` specifies the identifier of the PDB

Example 5-42 dbaascli pdb getDetails

```
dbaascli pdb getDetails--dbname cdb name --pdbName pdb name associated with
the CDB
```

```
dbaascli pdb getDetails--dbname cdb name --pdbUID con_uid of that pdb
```

dbaascli pdb getSnapshot

To obtain details of a given pluggable database (PDB) snapshot, use the `dbaascli pdb getSnapshot` command.

Prerequisite

Run the command as the `oracle` user.

Syntax

```
dbaascli pdb getSnapshot
{
    --pdbName <value>| --pdbUID <value>
}
{
    --snapshotName <value> | --snapshotUID <value>
}
--dbName <value>
```

Where:

- `--pdbName <value>` specifies the name of the PDB for which you want to obtain details.
- `--pdbUID <value>` specifies the user ID (UID) of the PDB for the snapshot for which you want to obtain details.

- `--snapshotName <value>` specifies the name of the snapshot for which you want to obtain details
- `--snapshotUID <value>` specifies the user ID (UID) of the snapshot for which you want to obtain details.
- `--dbName` specifies the Oracle Database name.

Example 5-43 dbaascli pdb configureSnapshot

In the following example, the details are obtained for the snapshot named `snap1` in the database named `db721`, in the PDB name `pdb1`:

```
dbaascli pdb getSnapshot --dbName db721 --pdbName pdb1 --snapshotName snap1
```

dbaascli pdb list

To view the list of pluggable databases (PDB) in a container database, use the `dbaascli pdb list` command.

Prerequisite

Run the command as the `oracle` user.

Syntax

```
dbaascli pdb list --dbname
```

Where:

- `--dbname` specifies the name of the container database that hosts the PDB

Example 5-44 dbaascli pdb list

```
dbaascli pdb list --dbname cdb name
```

dbaascli pdb listSnapshots

To list the snapshots of a given pluggable database (PDB), use the `dbaascli pdb listSnapshots` command..

Prerequisite

Run the command as the `oracle` user.

Syntax

```
dbaascli pdb listSnapshots
{
    --pdbName <value> | --pdbUID <value>
}
--dbName <value>
```

Where:

- `--pdbName <value>` specifies the PDB name for which the snapshots will be listed.

- `--pdbUID <value>` specifies the UID of the PDB for which the snapshots will be listed.
- `--dbName <value>` specifies the Oracle Database name.

Example 5-45 dbaascli pdb listSnapshots

In the following example, the command lists the snapshots for database `db721`, and the pdb name `pdb1`:

```
dbaascli pdb listSnapshots --dbName db721 --pdbName pdb1
```

dbaascli pdb localClone

To create a new pluggable database (PDB) as a clone of an existing PDB in the same container database (CDB), use the `dbaascli pdb localClone` command.

Prerequisite

Run the command as the `oracle` user.

Syntax

```
dbaascli pdb localClone --pdbName <value> --dbName <value>
[--targetPDBName <value>]
[--powerLimit <value>]
[--maxCPU <value]
[--maxSize <value>]
[--resume [--sessionID <value>]]
[--executePrereqs]
[--waitForCompletion <value>]
{[--blobLocation <value>]|[--standbyBlobFromPrimary <value>]}
[--excludeUserTablespaces <value>]
[--excludePDBData <value>]
[--pdbAdminUserName <value>]
[--lockPDBAdminAccount <value>]
[--sourcePDBServiceConvertList <value>]
```

Where:

- `--pdbName` specifies the name of the new PDB that you want to clone
- `--dbName` specifies the name of the database
- `--targetPDBName` specifies the name for the target PDB (new cloned PDB)
- `--powerLimit` specifies the degree of parallelism to be used for the clone operation. Valid value is between 1 and 128
- `--maxCPU` specifies the maximum number of CPUs to be allocated for the PDB
- `--maxSize` specifies the maximum storage size in GB for the new PDB
- `--resume` resumes the previous run
  - `--sessionID` specifies to resume a specific session ID
- `--executePrereqs` specifies `yes` to run only the prereqs for this operation. Valid values: `yes` or `no`

- `--waitForCompletion` specifies `false` to run the operation in the background. Valid values: `true` or `false`
- `--blobLocation` custom directory location where the standby blob file will be generated in a DG environment.
- `--standbyBlobFromPrimary` specifies the location of the standby blob file which is prepared from the primary database. This is required only for standby database PDB operations.

> **Note:**
>
> The parameters `--blobLocation` and `--standbyBlobFromPrimary` are mutually exclusive.

- `--excludeUserTablespaces` option to skip user table spaces, example t1,t2,t3.
- `--excludePDBData` specify true/yes to skip user data from source pdb.
- `--pdbAdminUserName` specify new PDB admin user name.
- `--lockPDBAdminAccount` specify true or false to lock the PDB admin user account. Default value is true.
- `--sourcePDBServiceConvertList` specify comma separated list of source to target service names which need to be converted. Syntax is source_srv1:new_srv1,source_srv2:new_srv2.

The newly cloned PDB inherits administration passwords from the source PDB.

Example 5-46 dbaascli pdb localClone

```
dbaascli pdb localClone --dbName db35 --pdbName PDB35 --targetPDBName
local_clone1 --maxCPU 2 --maxSize 15
```

dbaascli pdb open

To open a pluggable database (PDB), use the `dbaascli pdb open` command.

Run the command as the `root` or `oracle` user.

Syntax

```
dbaascli pdb open
 {
   --pdbName <value> | --pdbUID <value>
 }
--dbname <value> [--openMode <value>] [--startServices <value>] [--
waitForCompletion <value>] [--setPDBRefreshModeNone [--skipPDBRefresh] [--
pdbAdminUserName <value>]]
```

Where:

- `--pdbName` specifies the name of the PDB that you want to open
- `--pdbUID` specifies the identifier of the PDB
- `--dbname` specifies the name of the container database that hosts the PDB.

- `--openMode` specifies the target OPEN MODE of PDB
- `--startServices`: specifies to start all or list all services corresponding to a PDB. Accepted values are `all` or a comma-delimited list of PDB services.
- `--waitForCompletion`: specify `false` to run the operation in the background. Valid values: `true|false`
- `--setPDBRefreshModeNone`: specifies to convert a refreshable PDB to non-refreshable PDB
  - `--skipPDBRefresh`: specifies to skip refreshable PDB refresh
  - `--pdbAdminUserName`: specifies new PDB admin user name

Upon successful completion, the PDB is opened on all of the container database instances.

Example 5-47   dbaascli pdb open

```
dbaascli pdb open --dbname cdb name --pdbName pdb name associated with the CDB
```

```
dbaascli pdb open --dbname cdb name --pdbUID con_uid of that pdb
```

Optional: `--openMode READ_WRITE`/`READ_ONLY`

dbaascli pdb recover

To recover a pluggable database (PDB), use the `dbaascli pdb recover` command.

Prerequisite

- Run the command as the `root` user.
- Database must be configured with backup storage destination details where backups are stored.

Syntax

```
dbaascli pdb recover --pdbName <value> --dbname <value>
        {
            --start
                {
                    --untilTime <value>
                    | --untilSCN <value>
                    | --latest
                    | --tag <value>
                }
            | --status --uuid <value>
        }
```

Where:

```
--pdbName: PDB name.
--dbname: Oracle Database name.
--start | --status
--start
      --untilTime | --untilSCN | --latest | --tag
      --untilTime: Recovers PDB until time. Input format: DD-MON-YYYY HH24:MI:SS.
```

```
        --untilSCN: Recovers PDB until SCN.
        --latest: Recovers PDB to last known state.
        --tag: Recovers PDB to archival tag.
--status
        --uuid <value>
```

Example 5-48 Examples

- To recover a PDB *pdb1* in a CDB *myTestDb* to latest:

  ```
  dbaascli pdb recover --dbname myTestDb --pdbName pdb1 --start --latest
  ```

- To query the status of PDB recovery request submitted with `uuid` *81a17352362011ecbc3000163e8e4fac*:

  ```
  dbaascli pdb recover --dbname myTestDb --pdbName pdb1 --status --uuid
  81a17352362011ecbc3000163e8e4fac
  ```

Related Topics

- Connecting to a Virtual Machine with SSH
  You can connect to the virtual machines in an Oracle Exadata Database Service on Exascale Infrastructure system by using a Secure Shell (SSH) connection.

dbaascli pdb refresh

To refresh a specified pluggable database (PDB), use the `dbaascli pdb refresh` command.

Run the command as the `root` or `oracle` user.

Syntax

```
dbaascli pdb refresh --dbname <value>
   {
      --pdbName <value> | --pdbUID <value>
    }
    [--waitForCompletion <value>]
```

Where:

- `--dbname`: specifies the name of the Oracle Database
- `--pdbName`: specifies the name of the pluggable database
- `--pdbUID`: specifies the identifier of the pluggable database
- `--waitForCompletion`: specify `false` to run the operation in the background. Valid values: `true|false`

Related Topics

- Connecting to a Virtual Machine with SSH
  You can connect to the virtual machines in an Oracle Exadata Database Service on Exascale Infrastructure system by using a Secure Shell (SSH) connection.

dbaascli pdb relocate

To relocate the specified PDB from the remote database into local database, use the `dbaascli pdb relocate` command.

Prerequisite

Run the command as the `oracle` user. When prompted, you must supply the SYS user password for the source database.

Syntax

```
dbaascli pdb relocate --pdbName <value> --dbName <value> --
sourceDBConnectionString <value>
[--targetPDBName <value>]
[--powerLimit <value>]
[--maxCpu <value>]
[--maxSize <value>]
[--resume [--sessionID <value>]]
[--executePrereqs <value>]
[--sourcePDBServices <value>]
[--sourcePDBReadOnlyServices <value>]
[--waitForCompletion <value>]
{
    [--blobLocation <value>] | [--standbyBlobFromPrimary <value>]
}
[--upgradePDB <value>]
[--updateDBBlockCacheSize]
{
    [skipOpenPDB] | [--completePDBRelocate]
}
```

Where:

- `--pdbName` specifies the source PDB name to relocate
- `--dbName` specifies the target database name
- `--sourceDBConnectionString` specifies the source database connection string in the format `<scan_name>:<scan_port>/<database_service_name>`
- `--targetPDBName` specifies a name for the target PDB (new relocated PDB)
- `--powerLimit` specifies the degree of parallelism to be used for the relocate operation
- `--maxCpu` specifies the maximum number of CPUs to be allocated for the PDB
- `--maxSize` specifies the maximum storage size in GB for the new PDB
- `--resume` specifies to resume the previous execution
  - `--sessionID` specifies to resume a specific session ID
- `--executePrereqs` specifies `yes` to run only the prereqs for this operation. Valid values: `yes|no`
- `--sourcePDBServices` specifies a list of comma-delimited source PDB services

- `--sourcePDBReadOnlyServices` specifies a comma-delimited list of source PDB read-only services
- `--waitForCompletion` specifies `false` to run the operation in the background. Valid values: `true|false`
- `--blobLocation` specifies the location of a custom directory where the standby BLOB file will be generated in a Data Guard environment.
- `--standbyBlobFromPrimary` specifies the location of the standby BLOB file, which is prepared from the primary database. This is required only for standby operations.

> **Note:**
>
> The parameters `--blobLocation` and mutually exclusive.

- `--upgradePDB` specifies `true` to upgrade the PDB as part of this operation. Valid values : `true | false`.
- `--updateDBBlockCachesize` option to enable application to set `db block cache size` initialization parameters to support data copy with different block size.
- `--skipOpenPDB` - indicates that the PDB should not be opened at the end of the current operation.
- `--completePDBRelocate` - completes the PDB relocation if done as a two-step operation.

Example 5-49 dbaascli pdb relocate

```
dbaascli pdb relocate --sourceDBConnectionString test-
scan.dbaastoolslrgsu.dbaastoolslrgvc.oraclevcn.com:1521/
source_cdb_service_name --pdbName source_pdb --dbName target_db
```

dbaascli pdb remoteClone

To create a new pluggable database (PDB) as a clone of an existing PDB in another container database (CDB), use the `dbaascli pdb remoteClone` command.

Run the command as the `root` or `oracle` user.

Syntax

```
dbaascli pdb remoteClone --pdbName <value> --dbName <value> --
sourceDBConnectionString <value> [--targetPDBName <value>] [--powerLimit
<value>] [--maxCPU <value>] [--maxSize <value>] [--resume [--sessionID
<value>]] [--executePrereqs] [--waitForCompletion <value>] [--
sourcePDBExportedTDEKeyFile <value>]
        {
            [--blobLocation <value>]
            | [--standbyBlobFromPrimary <value>]
        }
[--excludeUserTablespaces <value>]
[--excludePDBData <value>]
[--pdbAdminUserName <value>]
[--lockPDBAdminAccount <value>]
[--sourcePDBServiceConvertList <value>]
[--refreshablePDB --refreshMode <value> [--refreshIntervalInMinutes <value>]
--dblinkUsername <value> [--honorCaseSensitiveUserName]]
[--updateDBBlockCacheSize]
```

Where:

- `--pdbName` specifies the name of the source PDB that you want to clone
- `--dbname` specifies the name (`DB_NAME`) of the CDB that hosts the newly cloned PDB
- `--sourceDBConnectionString` specifies the source database connection string in the format *`scan_name`*`:`*`scan_port`*`/`*`database_service_name`*
- `--targetPDBName` specifies the name for the target PDB (new cloned PDB)
- `--powerLimit` specifies the degree of parallelism to be used for the clone operation. Valid value is between 1 and 128
- `--maxCPU` specifies the maximum number of CPUs to be allocated for the PDB
- `--maxSize` specifies the maximum storage size in GB for the new PDB
- `--resume` resumes the previous run
  - `--sessionID` specifies to resume a specific session ID
- `--executePrereqs` specifies `yes` to run only the prereqs for this operation. Valid values: `yes` or `no`
- `--waitForCompletion` specifies `false` to run the operation in the background. Valid values: `true` or `false`
- `--sourcePDBExportedTDEKeyFile` specifies the source PDB exported key file. This variable is applicable to only 12.1 database.
- `--blobLocation` specifies the custom path where the standby blob file will be generated in a Data Guard environment
- `--standbyBlobFromPrimary` specify the location of the standby blob file, which is prepared from the primary database. This is required only for standby database PDB operations

> **Note:**
>
> The parameters `--blobLocation` and `--standbyBlobFromPrimary` are mutually exclusive.

- `--excludeUserTablespaces` option to skip user table spaces, example *t1*,*t2*,*t3*.
- `--excludePDBData` specify `true`/`yes` to skip user data from source PDB.
- `--pdbAdminUserName` specifies new PDB admin user name
- `--lockPDBAdminAccount` specify `true` or `false` to lock the PDB admin user account. Default value is `true`.
- `--sourcePDBServiceConvertList` specify a comma-delimited list of source to target service names, which need to be converted. Syntax is `source_srv1:new_srv1, source_srv2:new_srv2`.
- `--refreshablePDB` specifies to create refreshable PDB
  - `--refreshMode` specifies refresh mode for refreshable PDB. Valid values: `AUTO`|`MANUAL`

        - `--refreshIntervalInMinutes` specifies refresh interval for `refreshablePDB` in minutes
    - `--dblinkUsername` specifies common user of a remote database used for database link to connect to the remote database
        - `--honorCaseSensitiveUserName` indicates specified username is case sensitive
- `--updateDBBlockCacheSize`: specifies to enable application to set `db block cache size` initialization parameters to support data copy with a different block size

When promoted, you must supply the SYS user password for the source PDB. The newly cloned PDB inherits administration passwords from the source PDB. The cloned PDB is named using the following format: `dbname_sourcepdbname`. This command is supported only for databases that are not in a Data Guard configuration and use Oracle Database version 12.2.0.1, or later.

Example 5-50 dbaascli pdb remoteClone

```
dbaascli pdb remoteClone --sourceDBConnectionString test-
can.dbaastoolslrgsu.dbaastoolslrgvc.oraclevcn.com:1521 --pdbName source_pdb1
--dbName db9944 --targetPDBName new_pdb1 --maxsize 5 --maxcpu 2


dbaascli pdb remoteClone --sourceDBConnectionString
orcla.dbaastoolslrgsu.dbaastoolslrgvc.oraclevcn.com --pdbName source_pdb1 --
dbName db9944 --targetPDBName new_pdb1 --maxsize 5 --maxcpu 2
```

dbaascli system getDBHomes

To view information about all the Oracle homes, use the `dbaascli system getDBHomes` command.

Prerequisite

Run the command as the `root` or `oracle` user.

Syntax

```
dbaascli system getDBHomes
```

Example 5-51 dbaascli system getDBHomes

```
dbaascli system getDBHomes
```

dbaascli tde changePassword

To change TDE keystore password as well as DB wallet password for the alias `tde_ks_passwd`, use the `dbaascli tde changePassword` command.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli tde changePassword [--dbname <value>]
  {            [--prepareStandbyBlob <value> [--blobLocation <value>]]
               | [--standbyBlobFromPrimary <value>]
  }
  [--resume [--sessionID <value>]]
```

Where:

- `--dbname` specifies the name of the database
- `--prepareStandbyBlob` - specify true to generate a blob file containing the artifacts needed to perform the operation in a DG environment.
- `--blobLocation` - custom path where the standby blob file will be generated in a DG environment.
- `--standbyBlobFromPrimary` - specify the location of the standby blob file which is prepared from the primary database. This is required only for standby operations.
- `--resume` - to resume the previous execution
- `--sessionID` - to resume a specific session id.

```
dbaascli tde changepassword --dbname
      <dbname>
```

1. Change the TDE password in primary database.

   ```
   dbaascli tde changepassword --dbname
         <dbname> --prepareStandbyBlob true --blobLocation
         <Location where blob file has to be generated>
   ```

2. Copy the created standby blob to standby database environment.
3. Change the TDE password in standby database

   ```
   dbaascli tde changepassword --dbname
        <dbname> --standbyBlobFromPrimary <Location of blob generated from
       primary>
   ```

dbaascli tde addSecondaryHsmKey

To add a secondary HSM (KMS) key to the existing HSM (KMS) configuration, use the `dbaascli tde addSecondaryHsmKey` command.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli tde addSecondaryHsmKey --dbname <value> --secondaryKmsKeyOCID <value>
[--executePrereqs]
```

Where:

- `--secondaryKmsKeyOCID` specifies the secondary KMS key to add to the existing HSM (KMS) configuration
- `--dbname` specifies the name of the database
- `--executePrereqs` sexecute the prerequisites checks and report the results.

Example 5-52 dbaascli tde addSecondaryHsmKey

```
dbaascli tde addSecondaryHsmKey --dbname dbname --secondaryKmsKeyOCID
ocid1.key.oc1.eu-
frankfurt-1.bjqnwclvaafak.abtheljsgfxa2xe5prvlzdxtygoiqpm2pu2afgta54krxwllk5ux
ainvvxza
```

```
dbaascli tde addSecondaryHsmKey --dbname dbname --secondaryKmsKeyOCID
ocid1.key.oc1.eu-
frankfurt-1.bjqnwclvaafak.abtheljsgfxa2xe5prvlzdxtygoiqpm2pu2afgta54krxwllk5ux
ainvvxza --precheckOnly yes
```

dbaascli tde enableWalletRoot

To enable `wallet_root` spfile parameter for the existing database, use the `dbaascli tde enableWalletRoot` command.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli tde enableWalletRoot --dbname <value>
[--dbRestart <value>]
[--executePrereqs]
[--resume [--sessionID <value>]]
```

Where:

- `--dbname` specifies the name of the Oracle Database.
- `--dbrestart` specifies the database restart option. Valid values are: `rolling` or `full`.
  Default value: `rolling`
  If you do not pass the `dbrestart` argument, then the database restarts in a `rolling` manner.
- `--precheckOnly` runs only the precheck for this operation. Valid values are: `yes` or `no`
- `--resume` to resume the previous execution
- `--sessionID` to resume a specific session id.

Example 5-53 dbaascli tde enableWalletRoot

```
dbaascli tde enableWalletRoot --dbname db name --dbrestart rolling|full
```

```
dbaascli tde enableWalletRoot --dbname orcl
```

```
dbaascli tde enableWalletRoot --dbname orcl--dbrestart full
```

dbaascli tde encryptTablespacesInPDB

To encrypt all the tablespaces in the specified PDB, use the `dbaascli tde encryptTablespacesInPDB` command.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli tde encryptTablespacesInPDB --dbname <value> --pdbName <value>
[--executePrereqs]
```

Where:

- `--pdbName` specifies the name of the PDB to encrypt all the tablespaces.
- `--dbname` specifies the name of the Oracle Database.
- `--executePrereqs` execute the prerequisites checks and report the results.

Example 5-54 dbaascli tde encryptTablespacesInPDB

```
dbaascli tde encryptTablespacesInPDB --dbname dbname --pdbName pdb
```

```
dbaascli tde encryptTablespacesInPDB --dbname dbname --pdbName pdb --
executePrereqs
```

dbaascli tde fileToHsm

To convert FILE based TDE to HSM (KMS/OKV) based TDE, use the `dbaascli tde fileToHsm` command.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli tde fileToHsm --kmsKeyOCID <value> --dbname <value>
[--skipPatchCheck <value>]
[--executePrereqs ]
[--primarySuc <value>]
```

```
{
    [--resume [--sessionID <value>]] | [--revert [--sessionID <value>]]
}
[--waitForCompletion <value>]
```

Where:

- `--kmsKeyOCID` specifies the KMS key OCID to use for TDE. This is applicable only if KMS is selected for TDE
- `--dbname` specifies the name of the database
- `--skipPatchCheck` skips validation check for required patches if the value passed for this argument is `true`. Valid values: `true` or `false`
- `--executePrereqs` sexecute the prerequisites checks and report the results.
- `--primarySuc` specify this property in the standby database of the Data Guard environment once the command is successfully run on the primary database
- `--resume` specifies to resume the previous run
  - `--sessionID` specifies to resume a specific session ID
- `--revert` specifies to rollback the previous run
  - `--sessionID` specifies to rollback a specific session ID
- `--waitForCompletion` specify false to run the operation in background. Valid values : true| false.

Example 5-55 dbaascli tde fileToHsm --kmsKeyOCID

```
dbaascli tde fileToHSM --dbname dbname --kmsKeyOCID ocid1.key.oc1.eu-
frankfurt-.bjqnwclvaafak.abtheljsgfxa2xe5prvlzdxtygoiqpm2pu2afgta54krxwllk5uxa
invvxza
```

```
dbaascli tde fileToHSM --dbname dbname --kmsKeyOCID ocid1.key.oc1.eu-
frankfurt-.bjqnwclvaafak.abtheljsgfxa2xe5prvlzdxtygoiqpm2pu2afgta54krxwllk5uxa
invvxza --executePrereqs
```

```
dbaascli tde fileToHSM --dbname dbname --kmsKeyOCID ocid1.key.oc1.eu-
frankfurt-.bjqnwclvaafak.abtheljsgfxa2xe5prvlzdxtygoiqpm2pu2afgta54krxwllk5uxa
invvxza --resume
```

dbaascli tde getHsmKeys

To get TDE active key details, use the `dbaascli tde getHsmKeys` command.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli tde getHsmKeys
[--dbname]
[--infoFile]
```

Where:

- `--dbname` specifies the name of the database
- `--infoFile` specifies the file path where the list of OCIDs will be saved. The output is in JSON format

Example 5-56 dbaascli tde getHsmKeys

```
dbaascli tde getHsmkeys --dbname dbname
```

```
dbaascli tde getHsmkeys --dbname dbname --infoFile infoFilePath
```

dbaascli tde getMkidForKeyVersionOCID

To get Master Key ID associated with the KMS key version OCID, use the `dbaascli tde getMkidForKeyVersionOCID` command.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli tde getMkidForKeyVersionOCID --kmsKeyVersionOCID <value>
[--dbname <value>]
[--waitForCompletion <value>]
```

Where:

- `--kmsKeyVersionOCID` specifies the KMS key version OCID to set
- `--dbname` specifies the name of the database
- `--waitForCompletion` specify `false` to run the operation in background. Valid values : `true|false`.

Example 5-57 dbaascli tde getMkidForKeyVersionOCID

```
dbaascli tde getMkidForKeyVersionOCID --dbname dbname --kmsKeyVersionOCID
ocid1.keyversion.oc1.eu-
frankfurt-1.bjqnwclvaafak.bc4hmd3olgaaa.abtheljsyxtgn4vzi2bbpcej6a7abcwvylkd2l
x56lu2s6iwnxwgigu23nha
```

dbaascli tde getPrimaryHsmKey

To get primary HSM (KMS) key from the existing HSM (KMS) configuration, use the `dbaascli tde getPrimaryHsmKey` command.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli tde getPrimaryHsmKey
[--dbname]
```

Where:

- `--dbname` specifies the name of the database

Example 5-58 dbaascli tde getPrimaryHsmKey

```
dbaascli tde getPrimaryHsmKey --dbname dbname
```

dbaascli tde hsmToFile

To convert HSM (KMS/OKV) based TDE to FILE based TDE, use the `dbaascli tde hsmToFile` command.

Run the command as the `root` user.

Syntax

```
dbaascli tde hsmToFile
[--dbname <value>]
{
    [--prepareStandbyBlob <value> [--blobLocation <value>]
   | [--standbyBlobFromPrimary <value>]
}
]
[--skipPatchCheck <value>]
[--executePrereqs ]
[--primarySuc <value>]
{
     [--resume [--sessionID <value>]] |
     [--revert [--sessionID <value>]]
}
[--waitForCompletion <value>]
```

Where:

- `--dbname` specifies the name of the database
- `--prepareStandbyBlob` specify `true` to generate a blob file containing the artifacts needed to perform the operation in a DG environment.

- `--blobLocation` custom directory location where the standby blob file will be generated in a DG environment.
- `--standbyBlobFromPrimary` specify the location of the standby blob file which is prepared from the primary database. This is required only for standby operations.]
- `--skipPatchCheck` skips validation check for required patches if the value passed for this argument is `true`. Valid values: `true` or `false`
- `--executePrereqs` execute the prerequisites checks and report the results.
- `--primarySuc` specify this property in the standby database of the Data Guard environment once the command is successfully run on the primary database
- `--resume` resumes the previous run
  - `--sessionID` specifies to resume a specific session ID
- `--revert` specifies to roll back the previous run
  - `--sessionID` specifies to rollback a specific session ID
- `--waitForCompletion` specifies `false` to run the operation in background. Valid values: `true|false`

Example 5-59 dbaascli tde hsmToFile

```
dbaascli tde hsmToFile --dbname dbname
```

```
dbaascli tde hsmToFile --dbname dbname --executePrereqs
```

```
dbaascli tde hsmToFile --dbname dbname --resume
```

dbaascli tde listKeys

To list TDE master keys, use the `dbaascli tde listKeys` command.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli tde listKeys
[--dbname <value>]
[--infoFilePath <value>]
```

Where:

- `--dbname` specifies the name of the database
- `--infoFilePath` specify the absolute path of the file where the results will be saved.

Example 5-60 dbaascli tde listKeys

```
dbaascli tde listKeys --dbname dbname
```

```
dbaascli tde listKeys --dbname dbname --infoFilePath infoFilePath
```

dbaascli tde removeSecondaryHsmKey

To remove secondary HSM (KMS) key from the existing HSM (KMS) configuration, use the `dbaascli tde removeSecondaryHsmKey` command.

Prerequisite

Run the command as the `root` user.

Syntax

```
dbaascli tde removeSecondaryHsmKey --dbname <value>
[--confirmDeletion]
[--secondaryKmsKeyOCID]
[--executePrereqs]
```

Where:

- `--dbname` specifies the name of the database
- `--confirmDeletion` if not specified the user will be prompted while deleting all existing HSM(KMS) keys.
- `--secondaryKmsKeyOCID` secondary KMS key to be removed from existing HSM(KMS) configuration. If not specified all secondary KMS keys will be removed.
- `--executePrereqs` execute the prerequisites checks and report the results.

Example 5-61 dbaascli tde removeSecondaryHsmKey

```
dbaascli tde removeSecondaryHsmKey --dbname dbname
```

```
dbaascli tde removeSecondaryHsmKey --dbname dbname --secondaryKmsKeyOCID
ocid1.key.oc1.eu-
frankfurt-1.bjqnwclvaafak.abtheljsgfxa2xe5prvlzdxtygoiqpm2pu2afgta54krxwllk5ux
ainvvxza
```

```
dbaascli tde removeSecondaryHsmKey --dbname dbname --secondaryKmsKeyOCID
ocid1.key.oc1.eu-
frankfurt-1.bjqnwclvaafak.abtheljsgfxa2xe5prvlzdxtygoiqpm2pu2afgta54krxwllk5ux
ainvvxza --executePrereqs
```

dbaascli tde rotateMasterKey

To rotate the master key for database encryption, use the `dbaascli tde rotateMasterKey` command.

Prerequisites:

Run the command as the `root` user.

Syntax

```
dbaascli tde rotateMasterKey --dbname <value>
[--rotateMasterKeyOnAllPDBs]
[--pdbName <value>]
[--executePrereqs]
[--resume [--sessionID <value>]]
{
     [--prepareStandbyBlob <value> [--blobLocation <value>]]
     | [--standbyBlobFromPrimary <value>]
 }
```

Where:

- `--dbname` specifies the name of the Oracle Database
- `--rotateMasterKeyOnAllPDBs` specifies `true` to rotate master key of all PDBs in CDB. Valid values: `true|false`
- `--pdbName` specifies the name of the PDB
- `--executePrereqs` runs the prerequisites checks and report the results
- `--resume` specifes to resume the previous execution
- `--sessionID` specifies to resume a specific session ID
- `--prepareStandbyBlob` specifies `true` to generate a BLOB file containing the artifacts needed to perform the operation in a Data Guard environment
- `--blobLocation` specifies the location of the custom directory where the standby BLOB file will be generated in a Data Guard environment
- `--standbyBlobFromPrimary` specifies the location of the standby BLOB file, which is prepared from the primary database. This is required only for standby operations.

dbaascli tde setKeyVersion

To set the version of the primary key to be used in DB/CDB or PDB, use the `dbaascli tde setKeyVersion` command.

Run the command as the `root` user.

Syntax

```
dbaascli tde setKeyVersion --kmsKeyVersionOCID <value> --dbname <value>
[--pdbName <value>]
[--masterKeyID <value>]
[--standbySuc]
```

```
[--executePrereqs]
[--waitForCompletion <value>]
```

Where:

- `--kmsKeyVersionOCID` specifies the KMS key version OCID to set.
- `--dbname` specifies the name of the database.
- `--pdbName` name of the PDB to use the key version OCID.
- `--masterKeyID` specifies the master key ID of the given key version OCID. This is applicable to the Data Guard environment.
- `--standbySuc` specify this property in the primary database of the Data Guard environment once the command is successfully run on the standby database
- `--executePrereqs` execute the prerequisites checks and report the results.
- `--waitForCompletion` specify `false` to run the operation in background. Valid values: `true|false`

Example 5-62 dbaascli tde setKeyVersion

```
dbaascli tde setKeyVersion --dbname dbname --kmsKeyVersionOCID
ocid1.keyversion.oc1.eu-
frankfurt-1.bjqnwclvaafak.bc4hmd3olgaaa.abtheljsyxtgn4vzi2bbpcej6a7abcwvylkd2l
x56lu2s6iwnxwgigu23nha
```

```
dbaascli tde setKeyVersion --dbname dbname --kmsKeyVersionOCID
ocid1.keyversion.oc1.eu-
frankfurt-1.bjqnwclvaafak.bc4hmd3olgaaa.abtheljsyxtgn4vzi2bbpcej6a7abcwvylkd2l
x56lu2s6iwnxwgigu23nha --executePrereqs
```

```
dbaascli tde setKeyVersion --dbname dbname --pdbName pdb --kmsKeyVersionOCID
ocid1.keyversion.oc1.eu-
frankfurt-1.bjqnwclvaafak.bc4hmd3olgaaa.abtheljsyxtgn4vzi2bbpcej6a7abcwvylkd2l
x56lu2s6iwnxwgigu23nha
```

dbaascli tde setPrimaryHsmKey

To change the primary HSM (KMS) key for the existing HSM (KMS) configuration, use the `dbaascli tde setPrimaryHsmKey` command.

Run the command as the `root` user.

Syntax

```
dbaascli tde setPrimaryHsmKey --primaryKmsKeyOCID <value> --dbname <value>
[--allStandbyPrepared]
[--bounceDatabase]
[--executePrereqs]
[--resume [--sessionID <value>]]
```

Where:

- `--primaryKmsKeyOCID` specifies the primary KMS key to set
- `--dbname` specifies the name of the database
- `--allStandbyPrepared` specify to confirm that the operation has been successfully run on all the standby databases.
- `--bounceDatabase` specify this flag to do rolling database bounce for this operation
- `--executePrereqs` execute the prerequisites checks and report the results.
- `--resume` to resume the previous execution
- `--sessionID` to resume a specific session id.

Example 5-63 dbaascli tde setPrimaryHsmKey

```
dbaascli tde setPrimaryHsmKey --dbname dbname --primaryKmsKeyOCID
ocid1.key.oc1.eu-
frankfurt-1.bjqnwclvaafak.abtheljsgfxa2xe5prvlzdxtygoiqpm2pu2afgta54krxwllk5ux
ainvvxza


dbaascli tde setPrimaryHsmKey --dbname dbname --primaryKmsKeyOCID
ocid1.key.oc1.eu-
frankfurt-1.bjqnwclvaafak.abtheljsgfxa2xe5prvlzdxtygoiqpm2pu2afgta54krxwllk5ux
ainvvxza --executePrereqs
```

dbaascli tde status

To display information about the keystore for the specified database, use the `dbaascli tde status` command.

Prerequisite

Run the command as the `oracle` user.

Syntax

```
dbaascli tde status --dbname dbname
```

Where:

- `--dbname` specifies the name of the database that you want to check.

Output from the command includes the type of keystore, and the status of the keystore.

Example 5-64 dbaascli tde status

```
dbaascli tde status --dbname dbname
```

Database Service Events

The Database Service emits events, which are structured messages that indicate changes in resources.

Overview of Database Service Events

The Database Service Events feature implementation enables you to be notified about health issues with your Oracle Databases, or with other components on the Guest VM.

It is possible that Oracle Database or Clusterware may not be healthy or various system components may be running out of space in the Guest VM. You are not notified of this situation, unless you opt-in.

> **Note:**
>
> You are opting in with the understanding that the list of events can change in the future. You can opt-out of this feature at any time

Database Service Events feature implementation generates events for Guest VM operations and conditions, as well as Notifications for customers by leveraging the existing OCI Events service and Notification mechanisms in their tenancy. Customers can then create topics and subscribe to these topics through email, functions, or streams.

> **Note:**
>
> Events flow on Oracle Exadata Database Service on Exascale Infrastructure depends on the following components: Oracle Trace File Analyzer (TFA), sysLens, and Oracle Database Cloud Service (DBCS) agent. Ensure that these components are up and running.

Manage Oracle Trace File Analyzer

- To check the run status of Oracle Trace File Analyzer, run the `tfactl status` command as `root` or a non-root user:

  ```
  # tfactl status
  .---------------------------------------------------------------------------
  -----------------------.
  | Host    | Status of TFA | PID    | Port | Version     | Build ID
  | Inventory Status|
  +----------------+---------------+--------+------+------------
  +----------------------+------------+
  | node1       | RUNNING     | 41312  | 5000 | 22.1.0.0.0 |
  22100020220310214615 | COMPLETE         |
  | node2       | RUNNING     | 272300 | 5000 | 22.1.0.0.0 |
  22100020220310214615 | COMPLETE         |
  '----------------+---------------+--------+------+------------
  +----------------------+------------'
  ```

- To start the Oracle Trace File Analyzer daemon on the local node, run the `tfactl start` command as `root`:

  ```
  # tfactl start
  Starting TFA..
  ```

```
Waiting up to 100 seconds for TFA to be started..
. . . . .
. . . . .
. . . . .
. . . . .
. . . . .
. . . . .
. . . . .
. . . . .
Successfully started TFA Process..
. . . . .
TFA Started and listening for commands
```

- To stop the Oracle Trace File Analyzer daemon on the local node, run the `tfactl stop` command as `root`:

```
# tfactl stop
Stopping TFA from the Command Line
Nothing to do !
Please wait while TFA stops
Please wait while TFA stops
TFA-00002 Oracle Trace File Analyzer (TFA) is not running
TFA Stopped Successfully
Successfully stopped TFA..
```

Manage sysLens

- If sysLens is running, then once every 15 minutes data is collected in the local domU to discover the events to be reported. To check if sysLens is running, run the `systemctl status syslens` command as `root` in the domU:

```
# systemctl status syslens
\u25cf syslens.service
Loaded: loaded (/etc/systemd/system/syslens.service; disabled; vendor
preset: disabled)
Active: active (running) since Wed 2022-03-16 18:08:59 UTC; 34s ago
Main PID: 358039 (python3)
Memory: 31.6M
CGroup: /system.slice/syslens.service
\u2514\u2500358039 /usr/bin/python3 /var/opt/oracle/syslens/bin/
syslens_main.py --archive /var/opt/oracle/log/...

Mar 16 18:08:59 node1 systemd[1]: Started syslens.service.
Mar 16 18:09:09 node1 su[360495]: (to oracle) root on none
Mar 16 18:09:09 node1 su[360539]: (to grid) root on none
Mar 16 18:09:10 node1 su[360611]: (to grid) root on none
Mar 16 18:09:11 node1 su[360653]: (to oracle) root on none
```

- If the sysLens is enabled, when there is a reboot of the domU, then sysLens starts automatically. To validate if sysLens is enabled to collect telemetry, run the `systemctl is-enabled syslens` command as `root` in the domU:

```
# systemctl is-enabled syslens
enabled
```

- To validate if sysLens is configured to notify events, run the `/usr/bin/syslens --config /var/opt/oracle/syslens/data/exacc.syslens.config --get-key enable_telemetry` command as `root` in the domU:

```
# /usr/bin/syslens --config /var/opt/oracle/syslens/data/
exacc.syslens.config --get-key enable_telemetry
syslens Collection 2.3.3
on
```

Manage Database Service Agent

View the `/opt/oracle/dcs/log/dcs-agent.log` file to identify issues with the agent.

- To check the status of the Database Service Agent, run the `systemctl status` command:

```
# systemctl status dbcsagent.service
dbcsagent.service
Loaded: loaded (/usr/lib/systemd/system/dbcsagent.service; enabled; vendor
preset: disabled)
Active: active (running) since Fri 2022-04-01 13:40:19 UTC; 6min ago
Process: 9603 ExecStopPost=/bin/bash -c kill `ps -fu opc |grep "java.*dbcs-
agent.*jar" |awk '{print $2}' ` (code=exited, status=0/SUCCESS)
Main PID: 10055 (sudo)
CGroup: /system.slice/dbcsagent.service
□ 10055 sudo -u opc /bin/bash -c umask 077; /bin/java -
Doracle.security.jps.config=/opt/oracle/...
```

- To start the agent if it is not running, run the `systemctl start` command as the `root` user:

```
systemctl start dbcsagent.service
```

Related Topics

- Using the Console to Enable, Partially Enable, or Disable Diagnostics Collection
  You can enable, partially enable, or disable diagnostics collection for your Guest VMs after provisioning the VM cluster. Enabling diagnostics collection at the VM cluster level applies the configuration to all the resources such as DB home, Database, and so on under the VM cluster.
- Overview of Events
- Notifications Overview

Monitor Metrics for VM Cluster Resources

You can monitor the health, capacity, and performance of your VM clusters and databases with metrics, alarms, and notifications. You can use Oracle Cloud Infrastructure Console, Monitoring APIs, or Database Management APIs to view metrics.

Note: To view metrics you must have the required access as specified in an Oracle Cloud Infrastructure policy (whether you're using the Console, the REST API, or another tool). See Getting Started with Policies for information on policies.

WARNING:

Metrics, events, and audit events will not be sent if Cluster Ready Services (CRS) is not running before Autonomous Health Framework (AHF) starts.

- View Metrics for VM Cluster
- View Metrics for a Database
- View Metrics for VM Clusters in a Compartment
- View Metrics for Databases in a Compartment
- Manage Oracle Trace File Analyzer
- Manage Database Service Agent

View Metrics for VM Cluster

Perform the following steps to view the metrics for Guest VMs using the console.

Note:

When there is a network problem and Oracle Trace File Analyzer (TFA) is unable to post metrics, TFA will wait for one hour before attempting to retry posting the metrics. This is required to avoid creating a backlog of metrics processing on TFA.

Potentially one hour of metrics will be lost between network restore and the first metric posted.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata Database Service on Exascale Infrastructure**.
2. Choose your **Compartment**. A list of VM clusters is displayed.
3. In the list of VM clusters, click the VM cluster for which you want to view the metrics. Details of the VM cluster you selected are displayed.
4. In the **Resources** section, click **Metrics**.
   A chart for each metrics is displayed. By default, the metrics for the last one hour are displayed.

   You can only select the `oci_database_cluster` namespace from the **Metric namespace** drop-down.
5. If you want to change the interval, select the required start time and end time. Alternatively, you can select the interval from the Quick Selects drop down menu. The metrics are refreshed immediately for the selected interval.
6. For each metric, you can choose the interval and statistic independently.
   - Interval - The time period for which the metric is calculated.
   - Statistic - The mathematical method by which the metric is calculated.
7. For each metric, you can choose the following options from the 'Options' drop down menu.
   - View Query in Metrics Explorer

- Copy Chart URL
- Copy Query (MQL)
- Create an Alarm on this Query
- Table View

For Detailed information on various options for viewing the metrics chart, see Viewing Default Metric Charts.

View Metrics for a Database

Perform the following steps to view the metrics for a database using the console.

> **Note:**
>
> When there is a network problem and Oracle Trace File Analyzer (TFA) is unable to post metrics, TFA will wait for one hour before attempting to retry posting the metrics. This is required to avoid creating a backlog of metrics processing on TFA.
>
> Potentially one hour of metrics will be lost between network restore and the first metric posted.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata on Oracle Public Cloud**.
2. Choose your **Compartment**. A list of VM clusters is displayed.
3. In the list of VM clusters, click the VM cluster that contains the database for which you want to view the metrics. Details of the VM cluster you selected are displayed.
4. In the list of databases, click the database for which you want to view the metrics.
5. In the **Resources** section, click **Metrics**.
   A chart for each metrics is displayed. By default, the metrics for the last one hour are displayed.
6. Select a namespace from the **Metric namespace** from where you wish to view metrics.

   > **Note:**
   >
   > - When Database Management is enabled, you will have an option to choose from `oci_database` or `oracle_oci_database` namespace.
   > - When Database Management is disabled, then you can view metrics only from the `oci_database` namespace.

7. If you want to change the interval, select the required start time and end time. Alternatively, you can select the interval from the Quick Selects drop down menu. The metrics are refreshed immediately for the selected interval.
8. For each metric, you can choose the interval and statistic independently.
   - Interval - The time period for which the metric is calculated.
   - Statistic - The mathematical method by which the metric is calculated.

9. For each metric, you can choose the following options from the 'Options' drop down menu.
   - View Query in Metrics Explorer
   - Copy Chart URL
   - Copy Query (MQL)
   - Create an Alarm on this Query
   - Table View

For Detailed information on various options for viewing the metrics chart, see Viewing Default Metric Charts.

View Metrics for a PDB

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata on Oracle Public Cloud**.
2. Choose your **Compartment**. A list of VM clusters is displayed.
3. In the list of VM clusters, click the VM cluster that contains the database for which you want to view the metrics. Details of the VM cluster you selected are displayed.
4. In the list of databases, click the database that contains the PBD for which you want to view the metrics.
5. Under **Resources**, click **Pluggable Databases**.
6. In the list of VM clusters, click the PDB that you wish to view metrics.
7. Select a namespace from the **Metric namespace** from where you wish to view metrics.

> **Note:**
>
> - When Database Management is enabled, you will have an option to choose from `oracle_oci_database` namespace.
> - When Database Management is disabled, then the system will display a banner asking you to enable Database Management to provide metrics.

View Metrics for VM Clusters in a Compartment

Perform the following steps to view the metrics for databases in a compartment using the console.

> **Note:**
>
> When there is a network problem and Oracle Trace File Analyzer (TFA) is unable to post metrics, TFA will wait for one hour before attempting to retry posting the metrics. This is required to avoid creating a backlog of metrics processing on TFA.
>
> Potentially one hour of metrics will be lost between network restore and the first metric posted.

1. Open the Oracle Cloud Infrastructure **Console** by clicking the menu icon next to **Oracle Cloud**.
2. From the left navigation list click **Observability & Management**.
3. Under **Monitoring**, click **Service Metrics**.
4. On the Service Metrics page, under **Compartment** select your compartment.
5. On the Service Metrics page, under **Metric Namespace** select `oci_database_cluster`.
6. If there are multiple VM clusters in the compartment you can show metrics aggregated across the clusters by selecting **Aggregate Metric Streams**.
7. If you want to limit the metrics you see, next to **Dimensions** click **Add** (click **Edit** if you have already added dimensions).
8. In the **Dimension Name** field select a dimension.
9. In the **Dimension Value** field select a value.
10. Click **Done**.
11. In the **Edit dimensions** dialog click **+Additional Dimension** to add an additional dimension. Click **X** to remove a dimension.
12. To create an alarm on a specific metric, click **Options** and select **Create an Alarm on this Query**. See *Managing Alarms* for information on setting and using alarms.

> **Note:**
>
> If you don't see any metrics, check the network settings and AHF version listed in the prerequisites section.

Related Topics

- Managing Alarms

View Metrics for Databases in a Compartment

Perform the following steps to view the metrics for databases in a compartment using the console.

> **Note:**
>
> When there is a network problem and Oracle Trace File Analyzer (TFA) is unable to post metrics, TFA will wait for one hour before attempting to retry posting the metrics. This is required to avoid creating a backlog of metrics processing on TFA.
>
> Potentially one hour of metrics will be lost between network restore and the first metric posted.

1. Open the Oracle Cloud Infrastructure **Console** by clicking the menu icon next to **Oracle Cloud**.
2. From the left navigation list click **Observability & Management**.
3. Under **Monitoring**, click **Service Metrics**.

4. On the Service Metrics page, under **Compartment** select your compartment.
5. On the Service Metrics page, under **Metric Namespace** select **`oci_database`**.
6. If there are multiple databases in the compartment you can show metrics aggregated across the databases by selecting **Aggregate Metric Streams**.
7. If you want to limit the metrics you see, next to **Dimensions** click **Add** (click **Edit** if you have already added dimensions).
8. In the **Dimension Name** field select a dimension.
9. In the **Dimension Value** field select a value.
10. Click **Done**.
11. In the **Edit dimensions** dialog click **+Additional Dimension** to add an additional dimension. Click **X** to remove a dimension.
12. To create an alarm on a specific metric, click **Options** and select **Create an Alarm on this Query**. See Managing Alarms for information on setting and using alarms.

Manage Oracle Trace File Analyzer

The deployment of the cloud-certified Autonomous Health Framework (AHF), which includes Oracle Trace File Analyzer, is managed by Oracle. You shouldn’t install this manually on the guest VMs.

- To check the run status of Oracle Trace File Analyzer, run the `tfactl status` command as `root` or a non-root user:

```
# tfactl status
.---------------------------------------------------------------------------
-----------------------.
| Host            | Status of TFA | PID    | Port | Version     | Build
ID              | Inventory Status|
+----------------+---------------+--------+------+------------
+----------------------+------------+
| node1           | RUNNING       |  41312 | 5000 | 22.1.0.0.0 |
22100020220310214615| COMPLETE     |
| node2           | RUNNING       | 272300 | 5000 | 22.1.0.0.0 |
22100020220310214615| COMPLETE     |
'----------------+---------------+--------+------+------------
+----------------------+------------'
```

- To start the Oracle Trace File Analyzer daemon on the local node, run the `tfactl start` command as `root`:

```
# tfactl start
Starting TFA..
Waiting up to 100 seconds for TFA to be started..
. . . . .
. . . . .
. . . . .
. . . . .
. . . . .
. . . . .
. . . . .
. . . . .
```

```
Successfully started TFA Process..
. . . . .
TFA Started and listening for commands
```

- To stop the Oracle Trace File Analyzer daemon on the local node, run the `tfactl stop` command as `root`:

```
# tfactl stop
Stopping TFA from the Command Line
Nothing to do !
Please wait while TFA stops
Please wait while TFA stops
TFA-00002 Oracle Trace File Analyzer (TFA) is not running
TFA Stopped Successfully
Successfully stopped TFA..
```

Manage Database Service Agent

View the `/opt/oracle/dcs/log/dcs-agent.log` file to identify issues with the agent.

- To check the status of the Database Service Agent, run the `systemctl status` command:

```
# systemctl status dbcsagent.service
dbcsagent.service
Loaded: loaded (/usr/lib/systemd/system/dbcsagent.service; enabled; vendor
preset: disabled)
Active: active (running) since Fri 2022-04-0113:40:19UTC; 6min ago
Process: 9603ExecStopPost=/bin/bash -c kill `ps -fu opc |grep "java.*dbcs-
agent.*jar"|awk '{print $2}'` (code=exited, status=0/SUCCESS)
Main PID: 10055(sudo)
CGroup: /system.slice/dbcsagent.service
□ 10055sudo -u opc /bin/bash -c umask 077; /bin/java
```

- To start the agent if it is not running, run the `systemctl start` command as the `root` user:

```
systemctl start dbcsagent.service
```

Metrics for Oracle Exadata Database Service on Exascale Infrastructure in the Monitoring Service

learn about the metrics emitted by the Exadata Cloud Infrastructure Database service in the oci_database_cluster and oci_database namespaces for Oracle Databases.

Dimensions

All the metrics discussed in this topic include the following dimensions.

- RESOURCEID - The OCID of the VM Cluster.
- RESOURCENAME - The name of the VM Cluster.

The metrics listed in the following table are automatically available for the VM cluster.

Metric Name	Metric Display Name	Unit	Description and Metric Chart Defaults	Collection Frequency	Dimensions
`CpuUtilization`	**CPU Utilization**	percentage	Percent CPU utilization	1 minute	hostName deploymentType
`FilesystemUtilization`	**Filesystem Utilization**	percentage	Percent utilization of provisioned filesystem	1 minute	hostName deploymentType filesystemName
`LoadAverage`	**Load Average**	integer	System load average over 5 minutes	1 minute	hostName deploymentType
`MemoryUtilization`	**Memory Utilization**	percentage	Percentage of memory available for starting new applications, without swapping. The available memory can be obtained via the following command: `cat /proc/meminfo`	1 minute	hostName deploymentType
`NodeStatus`	**Node Status**	integer	Indicates whether the host is reachable.	1 minute	hostName deploymentType
`SwapUtilization`	**Swap Utilization**	percentage	Percent utilization of total swap space	1 minute	hostName deploymentType

The metrics listed in the following table are automatically available for the database.

Metric Name	Metric Display Name	Unit	Dsicription and Metric Chart Defaults	Collection Frequency	Dimensions
`CpuUtilization`	**CPU Utilization**	percentage	The CPU utilization expressed as a percentage, aggregated across all consumer groups. The utilization percentage is reported with respect to the number of CPUs the database is allowed to use.	5 minutes	instanceNumber instanceName hostName deploymentType resourceId_{database|pdb} resourceName_{database|pdb}

Metric Name	Metric Display Name	Unit	Dsicription and Metric Chart Defaults	Collection Frequency	Dimensions

`StorageUtilization`	**Storage Utilization**	percentage	The percentage of provisioned storage capacity currently in use. Represents the total allocated space for all tablespaces.	1 hour	deploymentType
					resourceId_{database|pdb}
					resourceName_{database|pdb}

`BlockChanges`	**DB Block Changes**	Changes per second	The Average number of blocks changed per second.	5 minutes	instanceNumber instanceName hostName deploymentType resourceId_{database|pdb} resourceName_{database|pdb}
`ExecuteCount`	**Execute Count**	Count	The number of user and recursive calls that executed SQL statements during the selected interval.	5 minutes	instanceNumber instanceName hostName deploymentType
`ExecuteCount`	**Execute Count**	Count	The number of user and recursive calls that executed SQL statements during the selected interval.	5 minutes	instanceNumber instanceName hostName deploymentType
`CurrentLogons`	**Current Logons**	Count	The number of successful logons during the selected interval.	5 minutes	instanceNumber instanceName hostName deploymentType resourceId_{database|pdb} resourceName_{database|pdb}

Metric Name	Metric Display Name	Unit	Dsicription and Metric Chart Defaults	Collection Frequency	Dimensions
`TransactionCount`	**Transaction Count**	Count	The combined number of user commits and user rollbacks during the selected interval.	5 minutes	instanceNumber instanceName hostName deploymentType resourceId_{database|pdb} resourceName_{database|pdb}
`UserCalls`	**User Calls**	Count	The combined number of logons, parses, and execute calls during the selected interval.	5 minutes	instanceNumber instanceName hostName deploymentType resourceId_{database|pdb} resourceName_{database|pdb}
`ParseCount`	**Parse Count**	Count	The number of hard and soft parses during the selected interval.	5 minutes	instanceNumber instanceName hostName deploymentType resourceId_{database|pdb} resourceName_{database|pdb}

`StorageUsed`	**Storage Space Used**	GB	Total amount of storage space used by the database at the collection time.	1 hour	deploymentType
					resourceId_{database|pdb}
					resourceName_{database|pdb}

`StorageAllocated`	**Storage Space Allocated**	GB	Total amount of storage space allocated to the database at the collection time	1 hour	deploymentType
					resourceId_{database|pdb}
					resourceName_{database|pdb}

Metric Name	Metric Display Name	Unit	Dsicription and Metric Chart Defaults	Collection Frequency	Dimensions
`StorageUsedByTablespace`	**Storage Space Used By Tablespace**	GB	Total amount of storage space used by tablespace at the collection time. In case of container database, this metric provides root container tablespaces.	1 hour	tablespaceName, tablespaceType deploymentType resourceId_{database|pdb} resourceName_{database|pdb}
StorageAllocatedByTablespace	**Allocated Storage Space By Tablespace**	GB	Total amount of storage space allocated to the tablespace at the collection time. In case of container database, this metric provides root container tablespaces.	1 hour	TablespaceName, tablespaceType, deploymentType, resourceId_{database|pdb} resourceName_{database|pdb}
`StorageUtilizationByTablespace`	**Storage Space Utilization By Tablespace**	percentage	This indicates the percentage of storage space utilized by the tablespace at the collection time. In case of container database, this metric provides root container tablespaces..	1 hour	tablespaceName, tablespaceType deploymentType

Metrics for Exadata Cloud Infrastructure in the Database Management Service

Database Management provides comprehensive database performance diagnostics and management capabilities for Oracle Databases.

This article describes the metrics emitted by the Exadata Cloud Infrastructure Database service in the `oracle_oci_database` namespace for Oracle Databases.

To use database metrics for these Oracle Databases in Exadata Cloud Infrastructure Database Service, you must enable Database Management for the database you want to monitor. You can enable either Basic Management or Full Management for your database. See Enable Database Management for instructions.

Dimensions

All the metrics discussed in this topic include the following dimensions.

- RESOURCEID - The OCID of the database.
- RESOURCENAME - The name of the database.
- DEPLOYMENTTYPE - The deployment type of the database.

The database metrics can be provided for the basic and full Database Management options.

> **Note:**
>
> Valid alarm intervals are 5 minutes or greater due to the frequency at which these metrics are emitted. See To create an alarm for details on creating alarms.

Basic Database Management Metrics in the oracle_oci_database Namespace

The metrics listed in the following table are automatically available for Oracle Databases when the **Basic Database Management** option is enabled.

Metric Name	Metric Display Name	Unit	Description and Metric Chart Defaults	Collection Frequency	Dimensions
`BlockChanges`	**DB Block Changes**	changes per second	The average number of blocks changed per second. Statistic: Mean Interval: 1 minute	5 minutes	instanceNumber instanceName hostName
`CpuUtilization`	**CPU Utilization**	percent	The CPU utilization expressed as a percentage, aggregated across all consumer groups. The utilization percentage is reported with respect to the number of CPUs the database is allowed to use. Statistic: Mean Interval: 1 minute	5 minutes	instanceNumber instanceName hostName
`CurrentLogons`	**Current Logons**	count	The number of successful logons during the selected interval. Statistics: Sum Interval: 1 minute	5 minutes	instanceNumber instanceName hostName

Metric Name	Metric Display Name	Unit	Description and Metric Chart Defaults	Collection Frequency	Dimensions
`ExecuteCount`	**Execute Count**	count	The number of user and recursive calls that executed SQL statements during the selected interval. Statistic: Sum Interval: 1 minute	5 minutes	instanceNumber instanceName hostName
`ParseCount`	**Parse Count (Total)**	count	The number of hard and soft parses during the selected interval. Statistic: Sum Interval: 1 minute	5 minutes	instanceNumber instanceName hostName
`StorageAllocated`	**Allocated Storage Space**	GB	The maximum amount of space allocated by tablespace during the interval. For container databases, this metric provides data for root container tablespaces. Statistic: Max Interval: 30 minutes	30 minutes	N/A
`StorageAllocatedByTablespace`	**Allocated Storage Space By Tablespace**	GB	The maximum amount of space allocated by tablespace during the interval. For container databases, this metric provides data for root container tablespaces. Statistic: Max Interval: 30 minutes	30 minutes	tablespaceName tablespaceType

Metric Name	Metric Display Name	Unit	Description and Metric Chart Defaults	Collection Frequency	Dimensions
`StorageUsed`	**Storage Space Used**	GB	The maximum amount of space used during the interval. Statistic: Max Interval: 30 minutes	30 minutes	N/A
`StorageUsedByTablespace`	**Storage Space Used By Tablespace**	GB	The maximum amount of space used by tablespace during the interval. For container databases, this metric provides data for root container tablespaces. Statistic: Max Interval: 30 minutes	30 minutes	tablespaceName tablespaceType
`StorageUtilization`	**Storage Utilization**	percent	The percentage of provisioned storage capacity currently in use. Represents the total allocated space for all tablespaces. Statistic: Mean Interval: 30 minutes	30 minutes	N/A
`StorageUtilizationByTablespace`	**Storage Space Utilization By Tablespace**	percent	The percentage of the space utilized, by tablespace. For container databases, this metric provides data for root container tablespaces. Statistic: mean Interval: 30 minutes	30 minutes	tablespaceName tablespaceType

Metric Name	Metric Display Name	Unit	Description and Metric Chart Defaults	Collection Frequency	Dimensions
`TransactionCount`	**Transaction Count**	count	The combined number of user commits and user rollbacks during the selected interval. Statistic: Sum Interval: 1 minute	5 minutes	instanceNumber instanceName hostName
`UserCalls`	**User Calls**	count	The combined number of logons, parses, and execute calls during the selected interval. Statistic: Sum Interval: 1 minute	5 minutes	instanceNumber instanceName hostName

NOT_SUPPORTED

The metrics listed in the following table are automatically available for Oracle Databases when the **Full Database Management** option is enabled.

Metric Name	Metric Display Name	Unit	Description and Metric Chart Defaults	Collection Frequency	Dimensions
`AllocatedStorageUtilizationByTablespace`	**Allocated Space Utilization By Tablespace**	percent	The percentage of space used by tablespace, out of all allocated. For container databases, this metric provides data for root container tablespaces. Statistic: Mean Interval: 30 minutes	30 minutes	tablespaceName tablespaceType

Metric Name	Metric Display Name	Unit	Description and Metric Chart Defaults	Collection Frequency	Dimensions
`AvgGCCRBlockReceiveTime`	**Average GC CR Block Receive Time**	milliseconds	The average global cache CR (consistent-read) block receive time. Statistic: Mean Interval: 5 minutes *For RAC / cluster databases only.*	5 minutes	instanceNumber instanceName hostName
`BlockingSessions`	**Blocking Sessions**	count	Current blocking sessions. Statistic: Max Interval: 15 minutes *Not applicable for container databases.*	15 minutes	N/A
`CPUTime`	**CPU Time**	seconds per second	The average rate of accumulation of CPU time by foreground sessions in the database instance over the time interval. The CPU time component of Average Active Sessions. Statistic: Mean Interval: 1 minute	5 minutes	instanceNumber instanceName hostName

Metric Name	Metric Display Name	Unit	Description and Metric Chart Defaults	Collection Frequency	Dimensions
`DbmgmtJobExecutionsCount`	??	??	The number of SQL job executions on a single managed database or a database group, and their status. Status dimensions can be the following values: "Succeeded," "Failed," "InProgress." Statistic: Sum Interval: 1 minute	??	managedDatabaseId managedDatabaseGroupId jobId status
`DBTime`	**DB Time**	seconds per second	The average rate of accumulation of database time (CPU + Wait) by foreground sessions in the database instance over the time interval. Also known as Average Active Sessions. Statistic: Mean Interval: 1 minute	5 minutes	instanceNumber instanceName hostName
`FRASpaceLimit`	**Flash Recovery Area Limit**	GB	The flash recovery area space limit. Statistic: Max Interval: 15 minutes *Not applicable for pluggable databases.*	15 minutes	N/A
`FRAUtilization`	**Flash Recovery Area Utilization**	percent	The flash recovery area utilization. Statistic: Mean Interval: 15 minutes *Not applicable for pluggable databases.*	15 minutes	N/A

Metric Name	Metric Display Name	Unit	Description and Metric Chart Defaults	Collection Frequency	Dimensions
`GCCRBlocksReceived`	**GC CR Blocks Received**	blocks per second	The global cache CR (consistent-read) blocks received per second. Statistic: Mean Interval: 5 minutes *For RAC / cluster databases only.*	5 minutes	instanceNumber instanceName hostName
`GCCurrentBlocksReceived`	**GC Current Blocks Received**	blocks per second	Represents global cache current blocks received per second. Statistic reports the mean value. Statistic: Mean Interval: 5 minutes *For Real Application Cluster (RAC) databases only.*	5 minutes	instanceNumber instanceName hostName
`InterconnectTraffic`	**Average Interconnect Traffic**	MB per second	The average internode data transfer rate. Statistic: Mean Interval: 5 minutes *For RAC / cluster databases only.*	5 minutes	instanceNumber instanceName hostName
`InvalidObjects`	**Invalid Objects**	count	Invalid database objects count. Statistic: Max Interval: 24 hours *Not applicable for container databases.*	24 hours	N/A

Metric Name	Metric Display Name	Unit	Description and Metric Chart Defaults	Collection Frequency	Dimensions
`IOPS`	**IOPS**	operations per second	The average number of input-output operations per second. Statistic: Mean Interval: 1 minute	5 minutes	instanceNumber instanceName hostName ioType (Read, Write)
`IOThroughput`	**IO Throughput**	MB per second	The average throughput in MB per second. Statistic: Mean Interval: 1 minute	5 minutes	instanceNumber instanceName hostName ioType (Read, Write)
`LogicalBlocksRead`	**Logical Reads**	reads per second	The average number of blocks read from SGA/ Memory (buffer cache) per second. Statistic: Mean Interval: 1 minute	5 minutes	instanceNumber instanceName hostName
`MaxTablespaceSize`	**Max Tablespace Size**	GB	The maximum possible tablespace size. For container databases, this metric provides data for root container tablespaces. Statistic: Max Interval: 30 minutes	30 minutes	tablespaceName tablespaceType

Metric Name	Metric Display Name	Unit	Description and Metric Chart Defaults	Collection Frequency	Dimensions
`MemoryUsage`	**Memory Usage**	MB	Memory pool total size in MB. Statistic: Mean Interval: 15 minutes	15 minutes	instanceNumber instanceName hostName memoryType (SGA, PGA) memoryPool (AllocatedPGA, Buffercachel, FixedSGA, JavaPool, LargePool, LogBuffer, OtherPools, SharedPool, StreamsPool)
`MonitoringStatus`	**Monitoring Status**	not applicable	The monitoring status of the resource. If a metric collection fails, error information is captured in this metric. Statistic: Mean Interval: 5 minutes	5 minutes	collectionName errorSeverity errorCode
`NonReclaimableFRA`	**Non Reclaimable Fast Recovery Area**	percent	The Non-reclaimable fast recovery area. Statistic: Mean Interval: 15 minutes *Not applicable for pluggable databases.*	15 minutes	N/A
`ParsesByType`	**Parses By Type**	parses per second	The number of hard or soft parses per second. Statistic: Mean Interval: 1 minute	5 minutes	instanceNumber instanceName hostName parseType (HardParse, SoftParse)

Metric Name	Metric Display Name	Unit	Description and Metric Chart Defaults	Collection Frequency	Dimensions
`ProblematicScheduledDBMSJobs`	**Problematic Scheduled DBMS Jobs**	count	The problematic scheduled database jobs count. Statistic: Max Interval: 15 minutes *Not applicable for container databases.*	15 minutes	type (Broken, Failed)
`Processes`	**Process Count**	count	The database processes count. Statistic: Max Interval: 1 minute *Not applicable for pluggable databases.*	5 minutes	instanceNumber instanceName hostName
`ProcessLimitUtilization`	**Process Limit Utilization**	percent	The process limit utilization. Statistic: Mean Interval: 1 minute *Not applicable for pluggable databases.*	5 minutes	instanceNumber instanceName hostName
`ReclaimableFRA`	**Reclaimable Fast Recovery Area**	percent	The reclaimable fast recovery area. Statistic: Mean Interval: 15 minutes *Not applicable for pluggable databases.*	15 minutes	N/A
`ReclaimableFRASpace`	**Flash Recovery Area Reclaimable Space**	GB	The flash recovery area reclaimable space. Statistic: Mean Interval: 15 minutes *Not applicable for pluggable databases.*	15 minutes	N/A

Metric Name	Metric Display Name	Unit	Description and Metric Chart Defaults	Collection Frequency	Dimensions
`RedoSize`	**Redo Generated**	MB per second	The average amount of redo generated, in MB per second. Statistic: Mean Interval: 1 minute	5 minutes	instanceNumber instanceName hostName
`SessionLimitUtilization`	**Session Limit Utilization**	percent	The session limit utilization. Statistic: Mean Interval: 1 minute *Not applicable for pluggable databases.*	5 minutes	instanceNumber instanceName hostName
`Sessions`	**Sessions**	count	The number of sessions in the database. Statistic: Mean Interval: 1 minute	5 minutes	instanceNumber instanceName hostName
`TransactionsByStatus`	**Transactions By Status**	transactions per second	The number of committed or rolled back transactions per second. Statistic: Mean Interval: 1 minute	5 minutes	instanceNumber instanceName hostName transactionStatus (Committed, RolledBack)
`UnusableIndexes`	**Unusable Indexes**	count	Unusable indexes count in database schema. Statistic: Max Interval: 24 hours *Not applicable for container databases.*	24 hours	schemaName
`UsableFRA`	**Usable Fast Recovery Area**	percent	The useable fast recovery area. Statistic: Mean Interval: 15 minutes *Not applicable for pluggable databases.*	15 minutes	N/A

Metric Name	Metric Display Name	Unit	Description and Metric Chart Defaults	Collection Frequency	Dimensions
`UsedFRASpace`	**Flash Recovery Area Usage**	GB	The flash recovery area space usage. Statistic: Max Interval: 15 minutes *Not applicable for pluggable databases.*	15 minutes	N/A
`WaitTime`	**Wait Time**	seconds per second	The average rate of accumulation of non-idle wait time by foreground sessions in the database instance over the time interval. The wait time component of Average Active Sessions. Statistic: Mean Interval: 5 minutes	5 minutes	instanceNumber instanceName hostName waitClass

Oracle Exadata Database Service on Exascale Infrastructure Events

Oracle Exadata Database Service on Exascale Infrastructure resources emit events, which are structured messages that indicate changes in resources.

- About Event Types on Oracle Exadata Database Service on Exascale Infrastructure
  Learn about the event types available for Oracle Exadata Database Service on Exascale Infrastructure resources.
- Prerequisites for Event Service
  The following prerequisites are required for the Events to flow out of the VM Cluster.
- Oracle Exadata Database Service on Exascale Infrastructure Event Types
  Learn about the event types available for Exadata Database Service on Exascale Infrastructure resources.
- Oracle Exadata Database Service on Exascale Infrastructure Maintenance Event Types
  The events in this section are emitted by the cloud Exadata infrastructure resource for Maintenance Events

- Exadata Cloud Infrastructure Critical and Information Event Types
  Exadata Cloud Infrastructure infrastructure resources emit "critical" and "information" data plane events that allow you to receive notifications when your infrastructure resource needs attention.
- Exadata Cloud Infrastructure VM Cluster Event Types
  Review the list of events that can be emitted by VM Cluster
- VM Node Subsetting Event Types
  Review the list of event types that VM Node Subsetting emits.
- Data Guard Association Event Types
  Review the list of event types that Data Guard associations emit.
- Oracle Database Home Event Types
  Review the list of events emitted by Oracle Database Homes.
- Database Event Types
  These are the event types that Oracle Databases in Exadata Cloud Service instances emit.
- Pluggable Database Event Types
  These are the event types that Oracle pluggable databases in Oracle Cloud Infrastructure emit.
- Database Service Events
  The Database Service emits events, which are structured messages that indicate changes in resources.
- Application VIP Event Types
  These are the event types that Application VIPs in Oracle Cloud Infrastructure emit.
- Interim Software Updates Event Types
  These are the event types that Interim Software Updates in Oracle Cloud Infrastructure emit.
- Serial Console Connection Event Types
  Review the list of event types that serial console connection emits.

About Event Types on Oracle Exadata Database Service on Exascale Infrastructure

Learn about the event types available for Oracle Exadata Database Service on Exascale Infrastructure resources.

Oracle Exadata Database Service on Exascale Infrastructure resources emit events, which are structured messages that indicate changes in resources. For more information about Oracle Cloud Infrastructure Events, see *Overview of Events*. You may subscribe to events and be notified when they occur using the Oracle Notification service, see *Notifications Overview*.

Related Topics

- Overview of Events
- Notifications Overview

Prerequisites for Event Service

The following prerequisites are required for the Events to flow out of the VM Cluster.

The Event Service requires the following:

1. Events on the VM Cluster depends on Oracle Trace File Analyzer (TFA) agent. Ensure that these components are up and running. AHF version **22.2.2** or higher is required for capturing events from the VM Cluster. To enable AHF Telemetry for the VM Cluster using the dbcli ulitilty, see AHF Telemetry Commands
2. The following network configurations are required.
   a. **Egress rules for outgoing traffic**: The default egress rules are sufficient to enable the required network path : For more information, see Default Security List .If you have blocked the outgoing traffic by modifying the default egress rules on your Virtual Cloud Network(VCN), you will need to revert the settings to allow outgoing traffic. The default egress rule allowing outgoing traffic is as follows:
      - Stateless: No (all rules must be stateful)
      - Destination Type: CIDR
      - Destination CIDR: **All <region> Services in Oracle Services Network**
      - IP Protocol: TCP
      - Destination Port: 443 (HTTPS)

   b. **Public IP or Service Gateway**: The database server host must have either a public IP address or a service gateway to be able to send database server host metrics to the Monitoring service.
      If the instance does not have a public IP address, set up a service gateway on the virtual cloud network (VCN). The service gateway lets the instance send database server host metrics to the Monitoring service without the traffic going over the internet.
      Here are special notes for setting up the service gateway to access the Monitoring service:
      i. When creating the service gateway, enable the service label called **All <region> Services in Oracle Services Network**. It includes the Monitoring service.
      ii. When setting up routing for the subnet that contains the instance, set up a route rule with **Target Type** set to **Service Gateway**, and the **Destination Service** set to **All <region> Services in Oracle Services Network**.

Oracle Exadata Database Service on Exascale Infrastructure Event Types

Learn about the event types available for Exadata Database Service on Exascale Infrastructure resources.

Oracle Exadata Database Service on Exascale Infrastructure resources emit events, which are structured messages that indicate changes in resources. For more information about Oracle Cloud Infrastructure Events, see *Overview of Events*. You may subscribe to events and be notified when they occur using the Oracle Notification service, see *Notifications Overview*.

Resource Events and Operations for ExaDB-XS

Table 5-1 Resource Events and Operations for ExaDB-XS

Friendly Name	Begin Event Sample	End event Sample
Create Storage Vault	`com.oraclecloud.DatabaseService.CreateExascaleDbStorageVault.begin`	`com.oraclecloud.DatabaseService.CreateExascaleDbStorageVault.end`

Table 5-1 (Cont.) Resource Events and Operations for ExaDB-XS

Friendly Name	Begin Event Sample	End event Sample
Create VM Cluster	`com.oraclecloud.DatabaseService.CreateExadbVmCluster.begin`	`com.oraclecloud.DatabaseService.CreateExadbVmCluster.end`
Get ExaDB VM Cluster	`com.oraclecloud.DatabaseService.GetExadbVmCluster`	This is synchronous operation, so there is no end event.
List ExaDB VM Cluster	`com.oraclecloud.databaseservice.ListExadbVmClusters`	This is synchronous operation, so there is no end event.
Update ExaDB VM Cluster	`com.oraclecloud.DatabaseService.UpdateExadbVmCluster.begin`	`com.oraclecloud.DatabaseService.UpdateExadbVmCluster.end`
Delete ExaDB VM Cluster	`com.oraclecloud.DatabaseService.DeleteExadbVmCluster.begin`	`com.oraclecloud.DatabaseService.DeleteExadbVmCluster.end`
Change Compartment ExaDB VM Cluster	`com.oraclecloud.DatabaseService.ChangeExadbVmClusterCompartment.begin`	`com.oraclecloud.DatabaseService.ChangeExadbVmClusterCompartment.end`
Remove Virtual Machine ExaDB VM Cluster	`com.oraclecloud.DatabaseService.ExadbVmClusterTerminateVirtualMachine.begin`	`com.oraclecloud.DatabaseService.ExadbVmClusterTerminateVirtualMachine.end`
Get Exascale DB Storage Vault	`com.oraclecloud.DatabaseService.GetExascaleDbStorageVault`	This is synchronous operation, so there is no end event.
List Exascale DB Storage Vaults	`com.oraclecloud.databaseservice.ListExascaleDbStorageVaults`	This is synchronous operation, so there is no end event.
Update Exascale DB Storage Vault	`com.oraclecloud.DatabaseService.UpdateExascaleDbStorageVault.begin`	`com.oraclecloud.DatabaseService.UpdateExascaleDbStorageVault.end`
Delete Exascale DB Storage Vault	`com.oraclecloud.DatabaseService.DeleteExascaleDbStorageVault.begin`	`com.oraclecloud.DatabaseService.DeleteExascaleDbStorageVault.end`
ChangeCompartment Exascale DB Storage Vault	`com.oraclecloud.DatabaseService.ChangeExascaleDbStorageVaultCompartment.begin`	`com.oraclecloud.DatabaseService.ChangeExascaleDbStorageVaultCompartment.end`

This is a reference event for an Oracle Exadata Database Service on Exascale Infrastructure\ resource:

```
{
  "datetime": <date>,
  "logContent": {
    "data": {
      "additionalDetails": {
        "cpuCoreCount": 4,
        "dbNodeIds": "<DBNodeID>, <DBNodeID",
        "exascaleDatabaseStorageVaultId": "<StorageVaultID>",
        "giVersion": "23.4.0.23.00",
        "licenseType": "LICENSE_INCLUDED",
        "lifecycleState": "TERMINATING",
        "localStorageInGbs": 586,
        "reservedCpuCoreCount": 4,
        "timeCreated": "2024-06-13T00:52:43Z",
        "timeUpdated": "2024-06-13T18:19:55Z",
        "timeZone": "UTC"
      },
      "availabilityDomain": "",
      "compartmentId": "ocid1.compartment.oc1<unique_ID>",
      "compartmentName": "<UniqueID>",
      "definedTags": {},
      "eventGroupingId": "/<ID>",
      "eventName": "GetExadbVmCluster",
      "freeformTags": {},
      "identity": {
        "authType": "natv",
        "callerId": null,
        "callerName": null,
        "consoleSessionId": null,
        "credentials": null,
        "ipAddress": "192.0.2.4",
        "principalId": "splat/<ID>",
        "principalName": "splat",
        "tenantId": "ocid1.tenancy.oc1<UniqueID>",
        "userAgent": "Jersey/2.38 (HttpUrlConnection 17.0.6)"
      },
      "message": "GetExadbVmCluster succeeded",
      "request": {
        "action": "GET",
        "headers": {},
        "id": "/<uniqueID>",
        "parameters": {},
        "path": "/20160918/exadbVmClusters/ocid1.<uniqueID>"
      },
      "resourceId": "ocid1.exadbvmcluster.oc1.<UniqueID>",
      "response": {
        "headers": {},
        "message": null,
        "payload": null,
        "responseTime": "2024-06-13T18:21:00.379Z",
        "status": "200"
      },
      "stateChange": {
        "current": {
          "cpuCoreCount": 4,
          "definedTags": {},
          "displayName": "audittest",
          "freeTags": {},
          "licenseType": "LICENSE_INCLUDED",
          "lifecycleState": "TERMINATING",
          "localStorageInGbs": 586,
          "reservedCpuCoreCount": 4,
          "sshPublicKeys": "..."
        },
        "previous": null
```

```
      }
    },
    "dataschema": "2.0",
    "id": "<uniqueID>",
    "oracle": {
      "compartmentid": "ocid1.compartment.oc1<UniqueID>",
      "ingestedtime": "2024-06-13T18:21:06.462Z",
      "loggroupid": "_Audit",
      "tenantid": "ocid1.tenancy.oc1<UniqueID>"
    },
    "source": "audittest",
    "specversion": "1.0",
    "time": "2024-06-13T18:21:00.277Z",
    "type": "com.oraclecloud.DatabaseService.GetExadbVmCluster"
  }
}
```

Related Topics

- Overview of Events
- Notifications Overview
- ExadbVmClusterUpdate Reference
- ExascaleDbStorageVault Reference
- ExadbVmCluster Reference

Oracle Exadata Database Service on Exascale Infrastructure Maintenance Event Types

The events in this section are emitted by the cloud Exadata infrastructure resource for Maintenance Events

> **Note:**
>
> Exadata systems that use the old DB system resource model are deprecated and will be desupported in a future release. The DB system event are not described.

Friendly Name	Event Type	Event Messages
Cloud Exadata Infrastructure – Maintenance Scheduled	`com.oraclecloud.databaseservice.cloudexadatainfrastructuremaintenancescheduled`	• **Rolling:** Oracle Cloud Operations is announcing the availability of a new quarterly maintenance update for Cloud Exadata Infrastructure. Oracle has scheduled the installation of this new update on your service instance <*infra-name*>, `ocid` <*infra-ocid*> on <*time-scheduled*>. The maintenance method for this maintenance is <*maintenance-method*> as selected per the maintenance preferences. • **Non Rolling:** Oracle Cloud Operations is announcing the availability of a new quarterly maintenance update for Cloud Exadata Infrastructure. Oracle has scheduled the installation of this new update on your service instance <*infra-name*>, `ocid` <*infra-ocid*> on <*time-scheduled*>. The maintenance method for this maintenance is <*maintenance-method*> as selected per the maintenance preferences. Non-rolling maintenance minimizes maintenance time but will result in full system downtime.

Friendly Name	Event Type	Event Messages
Cloud Exadata Infrastructure – Maintenance Reminder	`com.oraclecloud.databaseservice.cloudexadatainfrastructuremaintenancereminder`	- **Rolling:** This is an Oracle Cloud Operations reminder notice. Oracle has scheduled a quarterly maintenance update installation for Cloud Exadata Infrastructure <*infra-name*>, ocid <*ocid*> in approximately <*no-of-days*> days on <*time-scheduled*>. The maintenance method for this maintenance is <*maintenance-method*> as selected per the maintenance preferences. - **Non Rolling:** This is an Oracle Cloud Operations reminder notice. Oracle has scheduled a quarterly maintenance update installation for Cloud Exadata Infrastructure <*infra-name*>, ocid <*ocid*> in approximately <*no-of-days*> days on <*time-scheduled*>. The maintenance method for this maintenance is <*maintenance-method*> as selected per the maintenance preferences. Non-rolling maintenance minimizes maintenance time but will result in full system downtime.
Cloud Exadata Infrastructure - Maintenance Begin	`com.oraclecloud.databaseservice.cloudexadatainfrastructuremaintenance.begin`	This is an Oracle Cloud Operations notice regarding the quarterly maintenance update installation for your Cloud Exadata Infrastructure instance <*infra-name*>, ocid <*infra-ocid*>. The update installation for the service started at <*time scheduled*>. A follow-up notice will be sent when the maintenance update operation has completed.
Cloud Exadata Infrastructure - Maintenance End Success	`com.oraclecloud.databaseservice.cloudexadatainfrastructuremaintenance.end.success`	This is an Oracle Cloud Operations notice that your Cloud Exadata Infrastructure quarterly maintenance update installation for service instance <*infra-name*>, ocid <*infra-ocid*> which started at <*maintenance-start-time*> is now successfully complete.

Friendly Name	Event Type	Event Messages
Cloud Exadata Infrastructure - Maintenance End Failed	`com.oraclecloud.databaseservice.cloudexadatainfrastructuremaintenance.end.failed.`	This is an Oracle Cloud Operations notice that your Cloud Exadata Infrastructure quarterly maintenance update installation for service instance <*infra-name*>, ocid <*infra-ocid*> which started at <*maintenance-start-time*> has failed to complete due to technical reasons and operations team are currently looking into the issue. You will receive regular notifications to track progress of this maintenance.
Cloud Exadata Infrastructure - Maintenance VM Begin	`com.oraclecloud.databaseservice.cloudexadatainfrastructuremaintenancevm.begin.`	This is an Oracle Cloud Operations notice regarding the quarterly maintenance update of Virtual Machines component of your Cloud Exadata Infrastructure instance <*infra-name*>, ocid <*infra-ocid*>, Database Server <*dbserver name*>, `ocid` <*dbserver ocid*> has started. A follow-up notice will be sent when Virtual Machines maintenance operation has completed.
Cloud Exadata Infrastructure - MaintenanceVM End	`com.oraclecloud.databaseservice.cloudexadatainfrastructuremaintenancevm.end`	This is an Oracle Cloud Operations notice that quarterly maintenance update of the Database Server component of your Cloud Exadata Infrastructure instance <*infra-name*>, `ocid` <*infra-ocid*>; Database Server <*dbserver name*> `ocid` <*dbserver ocid*> has completed.
Cloud Exadata Infrastructure - Maintenance Storage Servers Start	`com.oraclecloud.databaseservice.cloudexadatainfrastructuremaintenancestorageservers.start`	This is an Oracle Cloud Operations notice regarding the quarterly maintenance update of Storage servers component of your Cloud Exadata Infrastructure instance <*infra-name*>, `ocid` <*infra-ocid*> has started. A follow-up notice will be sent when storage servers maintenance operation has completed.
Cloud Exadata Infrastructure - Maintenance Storage Servers End	`com.oraclecloud.databaseservice.cloudexadatainfrastructuremaintenancestorageservers.end`	This is an Oracle Cloud Operations notice that quarterly maintenance update of Storage servers component of your Cloud Exadata Infrastructure instance <*infra-name*>, `ocid` <*infra-ocid*> has completed.

Friendly Name	Event Type	Event Messages
Cloud Exadata Infrastructure - Maintenance Network Switches Begin	`com.oraclecloud.databaseservice.cloudexadatainfrastructuremaintenancenetworkswitches.begin`	This is an Oracle Cloud Operations notice regarding the quarterly maintenance update of the network fabric switches component of your Cloud Exadata Infrastructure instance <*infra-name*>, ocid <*infra-ocid*> has started. A follow-up notice will be sent when the network fabric switches maintenance operation has completed.
Cloud Exadata Infrastructure - Maintenance Network Switches End	`com.oraclecloud.databaseservice.cloudexadatainfrastructuremaintenancenetworkswitches.end`	This is an Oracle Cloud Operations notice that quarterly maintenance update of the network fabric switches component of your Cloud Exadata Infrastructure instance <*infra-name*>, ocid <*infra-ocid*> has completed.
Cloud Exadata Infrastructure - Maintenance Custom Action Time Begin	`com.oraclecloud.databaseservice.cloudexadatainfrastructuremaintenancecustomactiontime.begin`	This is an Oracle Cloud Operations notice that the custom action timeout for your Cloud Exadata Infrastructure instance <*infra-name*>, ocid <*infra-ocid*>; Database Server <*dbserver name*>, ocid <*dbserver ocid*> has started. A follow-up notice will be sent when the custom action timeout has ended.
Cloud Exadata Infrastructure - Maintenance Custom Action Time End	`com.oraclecloud.databaseservice.cloudexadatainfrastructuremaintenancecustomactiontime.end`	This is an Oracle Cloud Operations notice that the custom action timeout for your Cloud Exadata Infrastructure instance <*infra-name*>, ocid <*infra-ocid*>; Database Server <*dbserver name*>, ocid <*dbserver ocid*> has ended.
Cloud Exadata Infrastructure - Maintenance Rescheduled	`com.oraclecloud.databaseservice.cloudexadatainfrastructuremaintenancerescheduled`	Oracle Cloud Operations is announcing reschedule of a quarterly maintenance update for Cloud Exadata Infrastructure. A maintenance run has been rescheduled on your service instance <*infra-name*>, ocid <*infra-ocid*> to <*new-schedule-time*>.

Friendly Name	Event Type	Event Messages
Cloud Exadata Infrastructure - Maintenance Method Change	`com.oraclecloud.databaseservice.cloudexadatainfrastructuremaintenancemethodchange`	Oracle Cloud Operations is announcing a change related to quarterly maintenance update for Cloud Exadata Infrastructure. There's a change in maintenance method on your service instance <*infra-name*>, ocid <*infra-ocid*> to <*new-maintenance-method*>.

This is a reference event for a Cloud Exadata Infrastructure resource:

```
{
  "cloudEventsVersion": "0.1",
  "eventId": "<unique_ID>",
  "eventType":
"com.oraclecloud.databaseservice.cloudexadatainfrastructuremaintenance.end",
  "source": "DatabaseService",
  "eventTypeVersion": "1.0",
  "eventTime": "2019-06-27T21:16:04.000Z",
  "contentType": "application/json",
  "extensions": {
    "compartmentId": "ocid1.compartment.oc1.<unique_ID>"
  },
  "data": {
    "compartmentId": "ocid1.compartment.oc1.<unique_ID>",
    "compartmentName": "example_name",
    "resourceName": "my_exadata_infrastructure",
    "resourceId": "ocid1.dbsystem.oc1.eu-frankfurt-1.<unique_ID>", ,
    "availabilityDomain": "tXPJ:EU-FRANKFURT-1-AD-3",
    "freeFormTags": {
      "Department": "Finance"
    },
    "definedTags": {
      "Operations": {
        "CostCenter": "42"
      }
    },
    "additionalDetails" : {
"subnetId" : "ocid1.subnet.oc1.eu-frankfurt-1.<unique_ID>",
"lifecycleState" : "MAINTENANCE_IN_PROGRESS",
"sshPublicKeys" : "...",
"cpuCoreCount" : 32,
"version" : "19.2.8.0.0.191119",
"nsgIds" : "null",
"backupSubnetId" : "ocid1.subnet.oc1.eu-frankfurt-1.<unique_ID>",
"licenseType" : "BRING_YOUR_OWN_LICENSE",
"dataStoragePercentage" : 80,
"patchHistoryEntries" : "null",
"lifecycleMessage" : "The underlying infrastructure of this system (cell
storage) is being updated and this will not impact database
                      availability.",
"exadataIormConfig" : "ExadataIormConfigCache(lifecycleState=DISABLED,
lifeCycleDetails=null, objective=Auto,
                       dbPlans=[DbIormConfigCache(dbName=default, share=null,
flashCacheLimit=null), DbIormConfigCache(dbName=<my_database1>,
                       share=null, flashCacheLimit=null),
DbIormConfigCache(dbName=<my_database2>, share=null, flashCacheLimit=null),
                       DbIormConfigCache(dbName=<my_database3>, share=null,
flashCacheLimit=null), DbIormConfigCache(dbName=<my_database4>,
                       share=null, flashCacheLimit=null),
DbIormConfigCache(dbName=<my_database5>, share=null, flashCacheLimit=null),
                       DbIormConfigCache(dbName=<my_database6>, share=null,
flashCacheLimit=null), DbIormConfigCache(dbName=<my_database7>,
                       share=null, flashCacheLimit=null),
DbIormConfigCache(dbName=<my_database8>, share=null, flashCacheLimit=null),
                       DbIormConfigCache(dbName=<my_database9>, share=null,
flashCacheLimit=null), DbIormConfigCache(dbName=<my_database10>,
                       share=null, flashCacheLimit=null),
DbIormConfigCache(dbName=<my_database11>, share=null, flashCacheLimit=null)],
                       undoData=null)"
}
},
"eventID" : "<unique_ID>",
"extensions" : {
"compartmentId" : "ocid1.compartment.oc1.<unique_ID>"
}
}
```

Exadata Cloud Infrastructure Critical and Information Event Types

Exadata Cloud Infrastructure infrastructure resources emit "critical" and "information" data plane events that allow you to receive notifications when your infrastructure resource needs attention.

Exadata Cloud Service infrastructure resources emit "critical" and "information" data plane events that allow you to receive notifications when your infrastructure resource needs urgent attention ("critical" events), or notifications for events that are not critical, but which you may want to monitor ("information" events). The eventType values for these events are the following:

- `com.oraclecloud.databaseservice.exadatainfrastructure.critical`
- `com.oraclecloud.databaseservice.exadatainfrastructure.information`

These events use the `additionalDetails` section of the event message to provide specific details about what is happening within the infrastructure resource emitting the event. In the `additionalDetails` section, the `eventName` field provides the name of the critical or information event. *(Note that some fields in the example that follows have been omitted for brevity.)*

```
 {
  "eventType" :
"com.oraclecloud.databaseservice.exadatainfrastructure.critical",
  ....
  "data" : {
   ....
     "additionalDetails" : {
      ....
      "description" : "SQL statement terminated by Oracle Database Resource 
Manager due to excessive consumption of CPU and/or I/O.
                     The execution plan associated with the terminated SQL 
stmt is quarantined. Please find the sql identifier in
                     sqlId field of this JSON payload. This feature protects 
an Oracle database from performance degradation.
                     Please review the SQL statement. You can see the 
statement using the following commands: \"set serveroutput off\",
                     \"select sql_id, sql_text from v$sqltext where sql_id 
=<sqlId>\", \"set serveroutput on\"",
      "component" : "storage",
      "infrastructureType" : "exadata",
      "eventName" : "HEALTH.INFRASTRUCTURE.CELL.SQL_QUARANTINE",
      "quarantineMode" : "\"FULL Quarantine\""
       ....
    }
  },
  "eventID" : "<unique_ID>",
  ....
  }
}
```

In the tables below, you can read about the conditions and operations that trigger "critical" and "information" events. Each condition or operation is identified by a unique `eventName` value.

Critical events for Exadata Cloud Service infrastructure:

Critical Event - EventName	Description
`HEALTH.INFRASTRUCTURE.CELL.SQL_QUARANTINE`	SQL statement terminated by Oracle Database Resource Manager due to excessive consumption of CPU and/or I/O. The execution plan associated with the terminated SQL stmt is quarantined. Please find the sql identifier in sqlId field of this JSON payload. This feature protects an Oracle database from performance degradation. Please review the SQL statement. You can see the statement using the following commands: • `\"set serveroutput off\"` • `\"select sql_id, sql_text from v$sqltext where sql_id =<sqlId>\"` • `\"set serveroutput on\"`

Informational events for Exadata Cloud Service infrastructure:

Information Event - EventName	Description
`HEALTH.INFRASTRUCTURE.CELL.FLASH_DISK_FAILURE`	Flash Disk Failure has been detected. This is being investigated by Oracle Exadata team and the disk will be replaced if needed. No action needed from the customer.

NOT_SUPPORTED

In the following example of a "critical" event, you can see within the `additionalDetails` section of the event message that this particular message concerns an SQL statement that was terminated by Oracle Database Resource Manager because it was consuming excessive

CPU or I/O resources. The `eventName` and `description` fields within the `additionalDetails` section provide information regarding the critical situation:

```
 {
  "eventType" :
"com.oraclecloud.databaseservice.exadatainfrastructure.critical",
  "cloudEventsVersion" : "0.1",
  "eventTypeVersion" : "2.0",
  "source" : "Exadata Storage",
  "eventTime" : "2021-07-30T04:53:18Z",
  "contentType" : "application/json",
  "data" : {
    "compartmentId" : "ocid1.tenancy.oc1.<unique_ID>",
    "compartmentName" : "example_name",
    "resourceName" : "my_exadata_resource",
    "resourceId" : "ocid1.dbsystem.oc1.phx.<unique_ID>",
    "availabilityDomain" : "phx-ad-2",
     "additionalDetails" : {
      "serviceType" : "exacs",
      "sqlID" : "gnwfm1jgqcfuu",
      "systemId" : "ocid1.dbsystem.oc1.eu-frankfurt-1.<unique_ID>",
      "creationTime" : "2021-05-14T13:29:28+00:00",
      "dbUniqueID" : "1558836122",
      "quarantineType" : "SQLID",
      "dbUniqueName" : "AB0503_FRA1S6",
      "description" : "SQL statement terminated by Oracle Database Resource
Manager due to excessive consumption of CPU and/or I/O.
                      The execution plan associated with the terminated SQL
stmt is quarantined. Please find the sql identifier in sqlId
                      field of this JSON payload. This feature protects an
Oracle database from performance degradation.
                      Please review the SQL statement. You can see the
statement using the following commands: \"set serveroutput off\",
                      \"select sql_id, sql_text from v$sqltext where sql_id
=<sqlId>\", \"set serveroutput on\"",
      "quarantineReason" : "Manual",
      "asmClusterName" : "None",
      "component" : "storage",
      "infrastructureType" : "exadata",
      "name" : "143",
      "eventName" : "HEALTH.INFRASTRUCTURE.CELL.SQL_QUARANTINE",
      "comment" : "None",
      "quarantineMode" : "\"FULL Quarantine\"",
      "rpmVersion" : "OSS_20.1.8.0.0_LINUX.X64_210317",
      "cellsrvChecksum" : "14f73eb107dc1be0bde757267e931991",
      "quarantinePlan" : "SYSTEM"
    }
  },
  "eventID" : "<unique_ID>",
  "extensions" : {
    "compartmentId" : "ocid1.tenancy.oc1.<unique_ID>"
  }
}
```

NOT_SUPPORTED

In the following example of an "information" event, you can see within the `additionalDetails` section of the event message that this particular message concerns a flash disk failure that is being investigated by the Oracle Exadata operations team. The `eventName` and `description` fields within the `additionalDetails` section provide information regarding the event:

```
{
  "eventType" :
"com.oraclecloud.databaseservice.exadatainfrastructure.information",
  "cloudEventsVersion" : "0.1",
  "eventTypeVersion" : "2.0",
  "source" : "Exadata Storage",
  "eventTime" : "2021-12-17T19:14:42Z",
  "contentType" : "application/json",
  "data" : {
    "compartmentId" :
"ocid1.tenancy.oc1..aaaaaaaao3lj36x6lwxyvc4wausjouca7pwyjfwb5ebsq5emrpqlql2gj5
iq",
    "compartmentName" : "intexadatateam",
    "resourceId" :
"ocid1.dbsystem.oc1.phx.abyhqljt5y3taezn7ug445fzwlngjfszbedxlcbctw45ykkaxyzc5i
sxoula",
    "availabilityDomain" : "phx-ad-2",
    "additionalDetails" : {
      "serviceType" : "exacs",
      "component" : "storage",
      "systemId" :
"ocid1.dbsystem.oc1.phx.abyhqljt5y3taezn7ug445fzwlngjfszbedxlcbctw45ykkaxyzc5i
sxoula",
      "infrastructureType" : "exadata",
      "description" : "Flash Disk Failure has been detected. This is being
investigated by Oracle Exadata team and the disk will be
                       replaced if needed. No action needed from the
customer.",
      "eventName" : "HEALTH.INFRASTRUCTURE.CELL.FLASH_DISK_FAILURE",
      "FLASH_1_1" : "S2T7NA0HC01251  failed",
      "otto-ingestion-time" : "2021-12-17T19:14:43.205Z",
      "otto-send-EventService-time" : "2021-12-17T19:14:44.198Z"
    }
  },
  "eventID" : "30130ab4-42fa-4285-93a7-47e49522c698",
  "extensions" : {
    "compartmentId" :
"ocid1.tenancy.oc1..aaaaaaaao3lj36x6lwxyvc4wausjouca7pwyjfwb5ebsq5emrpqlql2gj5
iq"
  }
}
```

Exadata Cloud Infrastructure VM Cluster Event Types

Review the list of events that can be emitted by VM Cluster

Friendly Name	Event Type
Cloud VM Cluster - Change Compartment Begin	`com.oraclecloud.databaseservice.changecloudvmclustercompartment.begin`
Cloud VM Cluster - Change Compartment End	`com.oraclecloud.databaseservice.changecloudvmclustercompartment.end`
Cloud VM Cluster - Create Begin	`com.oraclecloud.databaseservice.createcloudvmcluster.begin`
Cloud VM Cluster - Create End	`com.oraclecloud.databaseservice.createcloudvmcluster.end`
Cloud VM Cluster - Delete Begin	`com.oraclecloud.databaseservice.deletecloudvmcluster.begin`
Cloud VM Cluster - Delete End	`com.oraclecloud.databaseservice.deletecloudvmcluster.end`
Cloud VM Cluster - Update Begin	`com.oraclecloud.databaseservice.updatecloudvmcluster.begin`
Cloud VM Cluster - Update End	`com.oraclecloud.databaseservice.updatecloudvmcluster.end`
Cloud VM Cluster - Update IORM Configuration Begin	`com.oraclecloud.databaseservice.updatecloudvmclusteriormconfig.begin`
Cloud VM Cluster - Update IORM Configuration End	`com.oraclecloud.databaseservice.updatecloudvmclusteriormconfig.end`
Cloud VM Cluster - Add Virtual Machine Begin	`com.oraclecloud.databaseservice.cloudvmclusteraddvirtualmachine.begin`
Cloud VM Cluster - Add Virtual Machine End	`com.oraclecloud.databaseservice.cloudvmclusteraddvirtualmachine.end`

NOT_SUPPORTED

This is a reference event for a cloud VM cluster resource:

```
{
    "cloudEventsVersion": "0.1",
    "eventID": "<unique_ID>",
    "eventType": 
"com.oraclecloud.databaseservice.updatecloudvmclusteriormconfig.begin",
    "source": "databaseservice",
    "eventTypeVersion": "2.0",
    "eventTime": "2022-06-27T21:16:04.000Z",
    "contentType": "application/json",
    "data": {
      "eventGroupingId": "<unique_ID>",
      "eventName": "UpdateCloudVmClusterIormConfig",
      "compartmentName": "example_compartment",
      "resourceName": "my_container_database",
      "resourceId": "ocid1.cloudvmcluster.oc1.<unique_ID>",
      "resourceVersion": null,
      "additionalDetails": {
        "cloudExadataInfrastructureId":
```

```
"ocid1.cloudexadatainfrastructure.oc1.<unique_ID>",
        "freeFormTags": {},
        "definedTags": {},
        "licenseType": "BRING_YOUR_OWN_LICENSE",
        "lifecycleState": "AVAILABLE",
        "giVersion": "19.0.0.0.0",
        "cpuCoreCount": 16
      }
    }
  },
  "timeCreated": "2022-06-15T16:31:31.979Z"
}
```

This is a reference event for Add Virtual Machine Begin:

```
{
  "id":
"ocid1.eventschema.oc1.phx.n2p4ijm0jyuia5p6lzhps0axtqft2d2ueywaq4oxcr3ywlzt9jd
689kvxazo",
  "serviceName": "Database",
  "displayName": "Cloud VM Cluster - Add Virtual Machine Begin",
  "eventType":
"com.oraclecloud.databaseservice.cloudvmclusteraddvirtualmachine.begin",
  "source": "databaseservice",
  "eventTypeVersion": "2.0",
  "eventTime": "2023-01-06T21:16:04.000Z",
  "contentType": "application/json",
  "additionalDetails": [
    {
      "name": "timeCreated",
      "type": "string"
    },
    {
      "name": "timeUpdated",
      "type": "string"
    },
    {
      "name": "lifecycleState",
      "type": "string"
    },
    {
      "name": "lifecycleDetails",
      "type": [
        "null",
        "string"
      ]
    },
    {
      "name": "cloudExadataInfrastructureId",
      "type": [
        "null",
        "string"
      ]
    },
    {
        "name": "cpuCoreCount",
        "type": [
          "null",
          "Integer"
        ]
      },
      {
        "name": "ocpuCountFractional",
        "type": [
          "null",
          "Float"
        ]
      },
      {
        "name": "dataStorageSizeInTBs",
        "type": [
          "null",
          "Integer"
        ]
      },
      {
        "name": "dataStorageSizeInGBs",
        "type": [
          "null",
          "Integer"
        ]
      },
      {
        "name": "licenseType",
        "type": [
          "null",
          "string"
        ]
      },
      {
        "name": "giVersion",
        "type": [
          "null",
          "string"
        ]
      },
      {
        "name": "dbNodeIds",
        "type": [
          "null",
          "string"
        ]
      },
      {
        "name": "timeZone",
        "type": [
          "null",
          "string"
        ]
      }
    ],
    "exampleEvent": {
      "eventType": 
  "com.oraclecloud.databaseservice.cloudvmclusteraddvirtualmachine.begin",
      "cloudEventsVersion": "0.1",
      "eventTypeVersion": "2.0",
      "source": "databaseservice",
      "eventID": "bc78609a-783a-9034-ccd1-12ab908df913",
      "eventTime": "2023-01-06T23:18:04.000Z",
      "contentType": "application/json",
      "data": {
        "eventGroupingId": "csid201fe4f3443a853d76e9cec3ef4a/
  3200918f142a44adb715d8aaf4f5ba99/DC62865A826A6E98699590E7F33C5064",
        "eventName": "CloudVmClusterAddVirtualMachine",
        "compartmentId": "ocid1.compartment.oc1.....unique_id",
        "compartmentName": null,
        "resourceName": "my_cloud_vm_cluster",
        "resourceId": "ocid1.cloudvmcluster.oc1.....unique_id",
        "resourceVersion": null,
        "availabilityDomain": "",
        "tagSlug": "tag_slug",
        "identity": {
          "principalName": null,
          "principalId": null,
          "authType": null,
          "callerName": null,
          "callerId": null,
          "tenantId": null,
          "ipAddress": null,
          "credentials": null,
          "authZPolicies": null,
          "userGroups": null,
          "userAgent": null,
          "consoleSessionId": null
        },
        "request": {
          "id": "01858321-0045-4bc5-b0d9-a917a6a40901",
          "path": null,
          "action": null,
          "parameters": null,
          "headers": null
        },
        "response": {
          "status": null,
          "responseTime": null,
          "headers": null,
          "payload": null,
          "message": null
        },
        "stateChange": {
          "previous": null,
          "current": {
            "licenseType": "BRING_YOUR_OWN_LICENSE",
            "dataStorageSizeGb": 60,
            "lifecycleState": "AVAILABLE",
            "sshPublicKeys": "...",
            "displayName": "my_cloud_vm_cluster",
            "cpuCoreCount": 16,
            "freeTags": {},
            "definedTags": {},
            "ocpuCountFractional": 16.0
          }
        },
        "additionalDetails": {
          "timeCreated": "2023-01-06T22:18:04.000Z",
          "timeUpdated": "2023-01-06T22:20:04.000Z",
          "lifecycleState": "AVAILABLE",
          "lifecycleDetails": null,
          "cloudExadataInfrastructureId":
  "ocid1.cloudexadatainfrastructure.oc1.....unique_id",
          "cpuCoreCount": 16,
          "ocpuCountFractional": 16.0,
          "dataStorageSizeInTBs": 4,
          "dataStorageSizeInGBs": 60,
          "licenseType": "BRING_YOUR_OWN_LICENSE",
          "giVersion": "19.0.0.0.0",
          "dbNodeIds": "[ocid1.dbnode.oc1.....unique_id,...]",
          "timeZone": "UTC"
        },
        "internalDetails": {
          "attributes": null
        }
      }
    },
    "timeCreated": "2023-01-06T23:18:04.000Z"
  }
```

This is a reference event for Add Virtual Machine End:

```
{
  "id":
"ocid1.eventschema.oc1.phx.v87pke1z9k9u6xaqo51taf6bunf0gc2wyhrbmjzbh3h1pjwakav
mf2borxgb",
  "serviceName": "Database",
  "displayName": "Cloud VM Cluster - Add Virtual Machine End",
  "eventType":
"com.oraclecloud.databaseservice.cloudvmclusteraddvirtualmachine.end",
  "source": "databaseservice",
  "eventTypeVersion": "2.0",
  "eventTime": "2023-01-06T21:16:04.000Z",
  "contentType": "application/json",
  "additionalDetails": [
    {
      "name": "timeCreated",
      "type": "string"
    },
    {
      "name": "timeUpdated",
      "type": "string"
    },
    {
      "name": "lifecycleState",
      "type": "string"
    },
    {
      "name": "lifecycleDetails",
      "type": [
        "null",
        "string"
      ]
    },
    {
      "name": "cloudExadataInfrastructureId",
      "type": [
        "null",
        "string"
      ]
    },
    {
      "name": "cpuCoreCount",
      "type": [
        "null",
        "Integer"
      ]
    },
    {
      "name": "ocpuCountFractional",
      "type": [
        "null",
        "Float"
      ]
    },
    {
      "name": "dataStorageSizeInTBs",
      "type": [
        "null",
        "Integer"
      ]
    },
    {
      "name": "dataStorageSizeInGBs",
      "type": [
        "null",
        "Integer"
      ]
    },
    {
      "name": "licenseType",
      "type": [
        "null",
        "string"
      ]
    },
    {
      "name": "giVersion",
      "type": [
        "null",
        "string"
```

```
      ]
    },
    {
      "name": "dbNodeIds",
      "type": [
        "null",
        "string"
      ]
    },
    {
      "name": "timeZone",
      "type": [
        "null",
        "string"
      ]
    }
  ],
  "exampleEvent": {
    "eventType": 
"com.oraclecloud.databaseservice.cloudvmclusteraddvirtualmachine.end",
    "cloudEventsVersion": "0.1",
    "eventTypeVersion": "2.0",
    "source": "databaseservice",
    "eventID": "ced78bb7-3903-acd8-ff78-5567aa01a912",
    "eventTime": "2023-01-06T23:18:04.000Z",
    "contentType": "application/json",
    "data": {
      "eventGroupingId": "csid89a04ef74ccb8b48340f56e656cf/
729c99d3e5a34d548ddc31c054810454/634F086E8618E0A660946A6862C82A68",
      "eventName": "CloudVmClusterAddVirtualMachine",
      "compartmentId": "ocid1.compartment.oc1.....unique_id",
      "compartmentName": null,
      "resourceName": "my_cloud_vm_cluster",
      "resourceId": "ocid1.cloudvmcluster.oc1.....unique_id",
      "resourceVersion": null,
      "availabilityDomain": "",
      "tagSlug": "tag_slug",
      "identity": {
        "principalName": null,
        "principalId": null,
        "authType": null,
        "callerName": null,
        "callerId": null,
        "tenantId": null,
        "ipAddress": null,
        "credentials": null,
        "authZPolicies": null,
        "userGroups": null,
        "userAgent": null,
        "consoleSessionId": null
      },
      "request": {
        "id": "07197e12-b680-475e-851e-bb89fcd8376d",
        "path": null,
        "action": null,
        "parameters": null,
        "headers": null
      },
      "response": {
        "status": null,
        "responseTime": null,
        "headers": null,
        "payload": null,
        "message": null
      },
      "stateChange": {
        "previous": null,
        "current": {
          "licenseType": "BRING_YOUR_OWN_LICENSE",
          "dataStorageSizeGb": 60,
          "lifecycleState": "AVAILABLE",
          "sshPublicKeys": "...",
          "displayName": "my_cloud_vm_cluster",
          "cpuCoreCount": 16,
          "freeTags": {},
          "definedTags": {},
          "ocpuCountFractional": 16.0
        }
      },
      "additionalDetails": {
        "timeCreated": "2023-01-06T22:18:04.000Z",
        "timeUpdated": "2023-01-06T22:20:04.000Z",
        "lifecycleState": "AVAILABLE",
        "lifecycleDetails": null,
        "cloudExadataInfrastructureId":
"ocid1.cloudexadatainfrastructure.oc1.....unique_id",
        "cpuCoreCount": 16,
        "ocpuCountFractional": 16.0,
        "dataStorageSizeInTBs": 4,
        "dataStorageSizeInGBs": 60,
        "licenseType": "BRING_YOUR_OWN_LICENSE",
        "giVersion": "19.0.0.0.0",
        "dbNodeIds": "[ocid1.dbnode.oc1.....unique_id,...]",
        "timeZone": "UTC"
      },
      "internalDetails": {
        "attributes": null
      }
    }
  },
  "timeCreated": "2023-01-06T23:18:04.000Z"
}
```

This is a reference event for Cloud VM Cluster - Update Begin:

```
{
  "id":
"ocid1.eventschema.oc1.phx.ekmz1phzp4bl1k7m7tbygulbnakmjnrsi99eqjops3zvpt337pn
nfmj6r79j",
  "serviceName": "Database",
  "displayName": "Cloud VM Cluster - Update Begin",
  "eventType": "com.oraclecloud.databaseservice.updatecloudvmcluster.begin",
  "source": "databaseservice",
  "eventTypeVersion": "2.0",
  "eventTime": "2019-06-27T21:16:04.000Z",
  "contentType": "application/json",
  "additionalDetails": [
    {
      "name": "id",
      "type": "string"
    },
    {
      "name": "defineTags",
      "type": [
        "null",
        "Map<String, Map<String, Object>>"
      ]
    },
    {
      "name": "freeFormTags",
      "type": [
        "null",
        "Map<String, String>"
      ]
    },
    {
      "name": "timeCreated",
      "type": "string"
    },
    {
      "name": "timeUpdated",
      "type": "string"
    },
    {
      "name": "lifecycleState",
      "type": "string"
    },
    {
      "name": "lifecycleDetails",
      "type": [
        "null",
        "string"
      ]
    },
    {
      "name": "cloudExadataInfrastructureId",
      "type": "string"
    },
    {
      "name": "cpuCoreCount",
      "type": [
        "null",
        "Integer"
      ]
    },
    {
      "name": "dataStorageSizeInGBs",
      "type": [
        "null",
        "Integer"
      ]
    },
    {
      "name": "licenseType",
      "type": [
        "null",
        "string"
      ]
    },
    {
      "name": "giVersion",
      "type": [
        "null",
        "string"
      ]
    },
    {
      "name": "dbNodeIds",
      "type": [
        "null",
        "string"
      ]
    },
    {
      "name": "timeZone",
      "type": [
        "null",
        "string"
      ]
    }
  ],
  "exampleEvent": {
    "cloudEventsVersion": "0.1",
    "eventID": "b28fcda6-3d7b-4044-aa8e-7c21cde84b44",
    "eventType": "com.oraclecloud.databaseservice.updatecloudvmcluster.begin",
    "source": "databaseservice",
    "eventTypeVersion": "2.0",
    "eventTime": "2019-06-27T21:16:04.000Z",
    "contentType": "application/json",
    "data": {
      "eventGroupingId": "4976b940-2c2d-4380-a669-1d70d071b187",
      "eventName": "UpdateCloudVmCluster",
      "compartmentName": "example_compartment",
      "resourceName": "my_container_database",
      "resourceId": "ocid1.cloudvmcluster.oc1.....unique_id",
      "resourceVersion": null,
      "additionalDetails": {
        "cloudExadataInfrastructureId":
"ocid1.cloudexadatainfrastructure.oc1.....unique_id",
        "freeFormTags": {},
        "definedTags": {},
        "licenseType": "BRING_YOUR_OWN_LICENSE",
        "lifecycleState": "AVAILABLE",
        "giVersion": "19.0.0.0.0",
        "cpuCoreCount": 16
      }
    }
  },
  "timeCreated": "2020-06-15T16:31:31.979Z"
}
```

This is a reference event for Cloud VM Cluster - Update End:

```
{
  "id":
"ocid1.eventschema.oc1.phx.svwkildsx63clp1q6phba7d6lns1rl92yc3uyc2ea5utjprqcwu
hbgvht4we",
  "serviceName": "Database",
  "displayName": "Cloud VM Cluster - Update End",
  "eventType": "com.oraclecloud.databaseservice.updatecloudvmcluster.end",
  "source": "databaseservice",
  "eventTypeVersion": "2.0",
  "eventTime": "2019-06-27T21:16:04.000Z",
  "contentType": "application/json",
  "additionalDetails": [
    {
      "name": "id",
      "type": "string"
    },
    {
      "name": "defineTags",
      "type": [
        "null",
        "Map<String, Map<String, Object>>"
      ]
    },
    {
      "name": "freeFormTags",
      "type": [
        "null",
        "Map<String, String>"
      ]
    },
    {
      "name": "timeCreated",
      "type": "string"
    },
    {
      "name": "timeUpdated",
      "type": "string"
    },
    {
      "name": "lifecycleState",
      "type": "string"
    },
    {
      "name": "lifecycleDetails",
      "type": [
      "null",
      "string"
    ]
  },
  {
    "name": "cloudExadataInfrastructureId",
    "type": "string"
  },
  {
    "name": "cpuCoreCount",
    "type": [
      "null",
      "Integer"
    ]
  },
  {
    "name": "dataStorageSizeInGBs",
    "type": [
      "null",
      "Integer"
    ]
  },
  {
    "name": "licenseType",
    "type": [
      "null",
      "string"
    ]
  },
  {
    "name": "giVersion",
    "type": [
      "null",
      "string"
    ]
  },
  {
    "name": "dbNodeIds",
    "type": [
      "null",
      "string"
    ]
  },
  {
    "name": "timeZone",
    "type": [
      "null",
      "string"
    ]
  }
],
"exampleEvent": {
  "cloudEventsVersion": "0.1",
  "eventID": "b28fcda6-3d7b-4044-aa8e-7c21cde84b44",
  "eventType": "com.oraclecloud.databaseservice.updatecloudvmcluster.end",
  "source": "databaseservice",
    "eventTypeVersion": "2.0",
    "eventTime": "2019-06-27T21:16:04.000Z",
    "contentType": "application/json",
    "data": {
      "eventGroupingId": "4976b940-2c2d-4380-a669-1d70d071b187",
      "eventName": "UpdateCloudVmCluster",
      "compartmentName": "example_compartment",
      "resourceName": "my_container_database",
      "resourceId": "ocid1.cloudvmcluster.oc1.....unique_id",
      "resourceVersion": null,
      "additionalDetails": {
        "cloudExadataInfrastructureId": 
"ocid1.cloudexadatainfrastructure.oc1.....unique_id",
        "freeFormTags": {},
        "definedTags": {},
        "licenseType": "BRING_YOUR_OWN_LICENSE",
        "lifecycleState": "AVAILABLE",
        "giVersion": "19.0.0.0.0",
        "cpuCoreCount": 16
      }
    }
  },
  "timeCreated": "2020-06-15T16:31:31.979Z"
}
```

VM Node Subsetting Event Types

Review the list of event types that VM Node Subsetting emits.

Table 5-2 VM Node Subsetting Events

Friendly Name	Event Type
VM Cluster - Add Virtual Machine Begin	com.oraclecloud.databaseservice.vmclusteraddvirtualmachine.begin
VM Cluster - Add Virtual Machine End	com.oraclecloud.databaseservice.vmclusteraddvirtualmachine.end
VM Cluster - Terminate Virtual Machine Begin	com.oraclecloud.databaseservice.vmclusterterminatevirtualmachine.begin
VM Cluster - Terminate Virtual Machine End	com.oraclecloud.databaseservice.vmclusterterminatevirtualmachine.end

Example 5-65 VM Node Subsetting Examples

This is a reference event for VM Cluster - Add Virtual Machine Begin:

```
"exampleEvent": {
"cloudEventsVersion": "0.1",
  "eventID": "60600c06-d6a7-4e85-b56a-1de3e6042f57",
  "eventType": 
"com.oraclecloud.databaseservice.vmclusteraddvirtualmachine.begin",
  "source": "databaseservice",
  "eventTypeVersion": "1.0",
  "eventTime": "2019-06-27T21:16:04.000Z",
  "contentType": "application/json",
  "extensions": {
"compartmentId": "ocid1.compartment.oc1..unique_ID"
  },
  "data": {
"compartmentId": "ocid1.compartment.oc1..unique_ID",
    "compartmentName": "example_name",
    "resourceName": "my_database",
    "resourceId": "Vmcluster-unique_ID",
    "availabilityDomain": "all",
    "freeFormTags": {},
    "definedTags": {},
    "additionalDetails": {
"id": "ocid1.id..oc1...unique_ID",
      "lifecycleState": "AVAILABLE",
      "timeCreated": "2019-09-03T12:00:00.000Z",
      "timeUpdated": "2019-09-03T12:30:00.000Z",
      "displayName": "testDisplayName",
      "lifecycleDetails": "detail message",
      "exadataInfrastructureId": "ExatraInfra-unique_ID",
      "vmClusterNetworkId": "VmCluster-unique_ID",
      "cpuCoreCount": 2,
      "dataStorageSizeInTBs": 4,
      "memorySizeInGBs": 30,
      "dbNodeStorageSizeInGBs": 60,
      "dbVersion": "19.0.0.0",
      "licenseType": "BRING_YOUR_OWN_LICENSE",
      "giVersion": "19.0.0.0",
      "dbNodeIds": "[ocid1.dbnode.1, ocid1.dbnode.2,...]",
      "dbServerIds": "[ocid1.dbserver.1, ocid1.dbserver.2,...]",
      "timeZone": "US/Pacific"
    }
  }
}
```

This is a reference event for VM Cluster - Add Virtual Machine End:

```
"exampleEvent": {
"cloudEventsVersion": "0.1",
  "eventID": "60600c06-d6a7-4e85-b56a-1de3e6042f57",
  "eventType":
"com.oraclecloud.databaseservice.vmclusteraddvirtualmachine.end",
  "source": "databaseservice",
  "eventTypeVersion": "1.0",
  "eventTime": "2019-06-27T21:16:04.000Z",
  "contentType": "application/json",
  "extensions": {
"compartmentId": "ocid1.compartment.oc1..unique_ID"
  },
  "data": {
"compartmentId": "ocid1.compartment.oc1..unique_ID",
    "compartmentName": "example_name",
    "resourceName": "my_database",
    "resourceId": "Vmcluster-unique_ID",
    "availabilityDomain": "all",
    "freeFormTags": {},
    "definedTags": {},
    "additionalDetails": {
"id": "ocid1.id..oc1...unique_ID",
      "lifecycleState": "AVAILABLE",
      "timeCreated": "2019-09-03T12:00:00.000Z",
      "timeUpdated": "2019-09-03T12:30:00.000Z",
      "displayName": "testDisplayName",
      "lifecycleDetails": "detail message",
      "exadataInfrastructureId": "ExatraInfra-unique_ID",
      "vmClusterNetworkId": "VmCluster-unique_ID",
      "cpuCoreCount": 2,
      "dataStorageSizeInTBs": 4,
      "memorySizeInGBs": 30,
      "dbNodeStorageSizeInGBs": 60,
      "dbVersion": "19.0.0.0",
      "licenseType": "BRING_YOUR_OWN_LICENSE",
      "giVersion": "19.0.0.0",
      "dbNodeIds": "[ocid1.dbnode.1, ocid1.dbnode.2,...]",
      "dbServerIds": "[ocid1.dbserver.1, ocid1.dbserver.2,...]",
      "timeZone": "US/Pacific"
    }
  }
}
```

This is a reference event for VM Cluster - Terminate Virtual Machine Begin:

```
"exampleEvent": {
"cloudEventsVersion": "0.1",
  "eventID": "60600c06-d6a7-4e85-b56a-1de3e6042f57",
  "eventType":
"com.oraclecloud.databaseservice.vmclusterterminatevirtualmachine.begin",
  "source": "databaseservice",
  "eventTypeVersion": "1.0",
  "eventTime": "2019-06-27T21:16:04.000Z",
  "contentType": "application/json",
  "extensions": {
"compartmentId": "ocid1.compartment.oc1..unique_ID"
  },
  "data": {
"compartmentId": "ocid1.compartment.oc1..unique_ID",
    "compartmentName": "example_name",
    "resourceName": "my_database",
    "resourceId": "Vmcluster-unique_ID",
    "availabilityDomain": "all",
    "freeFormTags": {},
    "definedTags": {},
    "additionalDetails": {
"id": "ocid1.id..oc1...unique_ID",
      "lifecycleState": "AVAILABLE",
      "timeCreated": "2019-09-03T12:00:00.000Z",
      "timeUpdated": "2019-09-03T12:30:00.000Z",
      "displayName": "testDisplayName",
      "lifecycleDetails": "detail message",
      "exadataInfrastructureId": "ExatraInfra-unique_ID",
   "vmClusterNetworkId": "VmCluster-unique_ID",
   "cpuCoreCount": 2,
   "dataStorageSizeInTBs": 4,
   "memorySizeInGBs": 30,
   "dbNodeStorageSizeInGBs": 60,
   "dbVersion": "19.0.0.0",
   "licenseType": "BRING_YOUR_OWN_LICENSE",
   "giVersion": "19.0.0.0",
   "dbNodeIds": "[ocid1.dbnode.1, ocid1.dbnode.2,...]",
   "dbServerIds": "[ocid1.dbserver.1, ocid1.dbserver.2,...]",
   "timeZone": "US/Pacific"
  }
 }
}
```

This is a reference event for VM Cluster - Terminate Virtual Machine End:

```
"exampleEvent": {
"cloudEventsVersion": "0.1",
 "eventID": "60600c06-d6a7-4e85-b56a-1de3e6042f57",
 "eventType":
"com.oraclecloud.databaseservice.vmclusterterminatevirtualmachine.end",
 "source": "databaseservice",
 "eventTypeVersion": "1.0",
 "eventTime": "2019-06-27T21:16:04.000Z",
 "contentType": "application/json",
 "extensions": {
"compartmentId": "ocid1.compartment.oc1..unique_ID"
 },
 "data": {
"compartmentId": "ocid1.compartment.oc1..unique_ID",
  "compartmentName": "example_name",
  "resourceName": "my_database",
  "resourceId": "Vmcluster-unique_ID",
  "availabilityDomain": "all",
  "freeFormTags": {},
  "definedTags": {},
  "additionalDetails": {
"id": "ocid1.id..oc1...unique_ID",
   "lifecycleState": "AVAILABLE",
   "timeCreated": "2019-09-03T12:00:00.000Z",
   "timeUpdated": "2019-09-03T12:30:00.000Z",
   "displayName": "testDisplayName",
   "lifecycleDetails": "detail message",
   "exadataInfrastructureId": "ExatraInfra-unique_ID",
   "vmClusterNetworkId": "VmCluster-unique_ID",
   "cpuCoreCount": 2,
   "dataStorageSizeInTBs": 4,
   "memorySizeInGBs": 30,
   "dbNodeStorageSizeInGBs": 60,
   "dbVersion": "19.0.0.0",
   "licenseType": "BRING_YOUR_OWN_LICENSE",
   "giVersion": "19.0.0.0",
   "dbNodeIds": "[ocid1.dbnode.1, ocid1.dbnode.2,...]",
   "dbServerIds": "[ocid1.dbserver.1, ocid1.dbserver.2,...]",
      "timeZone": "US/Pacific"
    }
  }
}
```

Data Guard Association Event Types

Review the list of event types that Data Guard associations emit.

Friendly Name	Event Type
Change Protection Mode Begin	com.oraclecloud.databaseservice.changeprotectionmode.begin
Change Protection Mode End	com.oraclecloud.databaseservice.changeprotectionmode.end
Data Guard Association - Create Begin	com.oraclecloud.databaseservice.createdataguardassociation.begin
Data Guard Association - Create End	com.oraclecloud.databaseservice.createdataguardassociation.end
Data Guard Association - Failover Begin	com.oraclecloud.databaseservice.failoverdataguardassociation.begin
Data Guard Association - Failover End	com.oraclecloud.databaseservice.failoverdataguardassociation.end
Data Guard Association - Reinstate Begin	com.oraclecloud.databaseservice.reinstatedataguardassociation.begin
Data Guard Association - Reinstate End	com.oraclecloud.databaseservice.reinstatedataguardassociation.end
Data Guard Association - Switchover Begin	com.oraclecloud.databaseservice.switchoverdataguardassociation.begin
Data Guard Association - Switchover End	com.oraclecloud.databaseservice.switchoverdataguardassociation.end

NOT_SUPPORTED

This is a reference event for Data Guard associations:

```
{
    "cloudEventsVersion": "0.1",
    "contentType": "application/json",
    "data": {
        "additionalDetails": {
            "ApplyLag": null,
            "DGConfigId": "7e8eff2b-a4cd-474a-abd5-940b05c0b1fd",
            "DGConfigState": "null",
            "DatabaseId": "ocid1.database.oc1.iad.<unique_ID>",
            "DbHomeId": "ocid1.dbhome.oc1.iad.<unique_ID>",
            "DbSystemId": "ocid1.dbsystem.oc1.iad.<unique_ID>",
            "LastSyncedTime": null,
            "SyncState": "null",
            "dcsDgUpdateTimestamp": null,
            "lastUpdatedIdentifier": null,
            "lifeCycleMessage": null,
            "lifecycleState": "PROVISIONING",
            "timeCreated": "2019-10-25T21:42:19.041Z",
            "timeUpdated": "2019-10-25T21:42:19.041Z"
        },
        "availabilityDomain": "XXIT:US-ASHBURN-AD-1",
        "compartmentId": "ocid1.compartment.oc1.<unique_ID>",
        "compartmentName": "example_compartment",
        "resourceId": "ocid1.dgassociation.oc1.iad.<unique_ID>"
    },
    "eventID": "5b8b7fbf-2e9a-4730-9761-e52715b7bc79",
    "eventTime": "2019-10-25T21:42:16.579Z",
    "eventType": 
"com.oraclecloud.databaseservice.createdataguardassociation.begin",
    "eventTypeVersion": "2.0",
    "extensions": {
        "compartmentId": "ocid1.compartment.oc1.<unique_ID>"
    },
    "source": "DatabaseService"
}
```

Oracle Database Home Event Types

Review the list of events emitted by Oracle Database Homes.

Friendly Name	Event Type
DB Home - Create Begin	com.oraclecloud.databaseservice.createdbhome.begin
DB Home - Create End	com.oraclecloud.databaseservice.createdbhome.end
DB Home - Patch Begin	com.oraclecloud.databaseservice.patchdbhome.begin
DB Home - Patch End	com.oraclecloud.databaseservice.patchdbhome.end
DB Home - Terminate Begin	com.oraclecloud.databaseservice.deletedbhome.begin
DB Home - Terminate End	com.oraclecloud.databaseservice.deletedbhome.end
DB Home - Update Begin	com.oraclecloud.databaseservice.updatedbhome.begin
DB Home - Update End	com.oraclecloud.databaseservice.updatedbhome.end

NOT_SUPPORTED

This is a reference event for Database Homes:

```
{
    "cloudEventsVersion": "0.1",
    "eventID": "60600c06-d6a7-4e85-b56a-1de3e6042f57",
    "eventType": "com.oraclecloud.databaseservice.createdbhome.begin",
    "source": "databaseservice",
    "eventTypeVersion": "2.0",
    "eventTime": "2019-08-29T21:16:04Z",
    "contentType": "application/json",
  "extensions": {
    "compartmentId": "ocid1.compartment.oc1.<unique_ID>"
  },
  "data": {
    "compartmentId": "ocid1.compartment.oc1.<unique_ID>",
    "compartmentName": "example_compartment",
    "resourceName": "my_dbhome",
    "resourceId": "DbHome-unique_ID",
    "availabilityDomain": "all",
    "freeFormTags": {},
    "definedTags": {},
    "additionalDetails": {
      "id": "ocid1.id.oc1.<unique_ID>",
      "lifecycleState": "PROVISIONING",
      "timeCreated": "2019-08-29T12:00:00.000Z",
      "timeUpdated": "2019-08-29T12:30:00.000Z",
      "lifecycleDetails": "detail message",
      "dbSystemId": "DbSystem-unique_ID",
      "dbVersion": "19.0.0.0",
      "recordVersion": 4,
      "displayName": "example_display_name"
    }
  }
}
```

Database Event Types

These are the event types that Oracle Databases in Exadata Cloud Service instances emit.

Friendly Name	Event Type
Database - Automatic Backup Begin	`com.oraclecloud.databaseservice.automaticbackupdatabase.begin`
Database - Automatic Backup End	`com.oraclecloud.databaseservice.automaticbackupdatabase.end`
Database - Create Backup Begin	`com.oraclecloud.databaseservice.backupdatabase.begin`
Database - Create Backup End	`com.oraclecloud.databaseservice.backupdatabase.end`
Database - Critical	`com.oraclecloud.databaseservice.database.critical`
Database - Information	`com.oraclecloud.databaseservice.database.information`
Database - Delete Backup Begin	`com.oraclecloud.databaseservice.deletebackup.begin`
Database - Delete Backup End	`com.oraclecloud.databaseservice.deletebackup.end`
Database - Migrate to KMS Key Begin	`com.oraclecloud.databaseservice.migratedatabasekmskey.begin`
Database - Migrate to KMS Key End	`com.oraclecloud.databaseservice.migratedatabasekmskey.end`
Database - Move Begin	`com.oraclecloud.databaseservice.movedatabase.begin`

Friendly Name	Event Type
Database - Move End	com.oraclecloud.databaseservice.movedatabase.end
Database - Restore Begin	com.oraclecloud.databaseservice.restoredatabase.begin
Database - Restore End	com.oraclecloud.databaseservice.restoredatabase.end
Database - Rotate KMS Key Begin	com.oraclecloud.databaseservice.rotatedatabasekmskey.begin
Database - Rotate KMS Key End	com.oraclecloud.databaseservice.rotatedatabasekmskey.end
Database - Terminate Begin	com.oraclecloud.databaseservice.database.terminate.begin
Database - Terminate End	com.oraclecloud.databaseservice.database.terminate.end
Database - Update Begin	com.oraclecloud.databaseservice.updatedatabase.begin
Database - Update End	com.oraclecloud.databaseservice.updatedatabase.end
Database - Upgrade Begin	com.oraclecloud.databaseservice.upgradedatabase.begin
Database - Upgrade End	com.oraclecloud.databaseservice.upgradedatabase.end

NOT_SUPPORTED

This is a reference event for databases:

```
{
"eventType" : "com.oraclecloud.databaseservice.backupdatabase.begin",
udEventsVersion" : "0.1",
"eventTypeVersion" : "2.0",
"source" : "DatabaseService",
"eventTime" : "2020-01-08T17:31:43.666Z",
"contentType" : "application/json",
"data" : {
"compartmentId" : "ocid1.compartment.oc1.<unique_ID>",
"compartmentName": "example_compartment_name",
"resourceName": "my_backup",
"resourceId": "ocid1.dbbckup.oc1.<unique_ID>",
"availabilityDomain": "<availability_domain>",
"additionalDetails" : {
"timeCreated" : "2020-01-08T17:31:44Z",
"lifecycleState" : "CREATING",
"dbSystemId" : "ocid1.dbsystem.oc1.<unique_ID>",
"dbHomeId" : ocid1.dbhome.oc1.<unique_ID>",
"dbUniqueName" : DB1115_iad1dv",
"dbVersion" : "11.2.0.4.190716",
"databaseEdition" : "ENTERPRISE_EDITION_HIGH_PERFORMANCE",
"autoBackupsEnabled" : "false",
"backupType" : "FULL",
"databaseId" : "ocid1.database.oc1.<unique_ID>",
},
"definedTags" : {
      "My_example_tag_name" :
        { "Example_key" : "Example_value" }
      },
    "eventID": "<unique_ID>",
    "extensions" : {
      "compartmentId": "ocid1.compartment.oc1.<unique_ID>"
    }
}
```

Pluggable Database Event Types

These are the event types that Oracle pluggable databases in Oracle Cloud Infrastructure emit.

Friendly Name	Event Type
Pluggable Database - Create Begin	`com.oraclecloud.databaseservice.createpluggabledatabase.begin`
Pluggable Database - Create End	`com.oraclecloud.databaseservice.createpluggabledatabase.end`
Pluggable Database - Delete Begin	`com.oraclecloud.databaseservice.deletepluggabledatabase.begin`
Pluggable Database - Delete End	`com.oraclecloud.databaseservice.deletepluggabledatabase.end`
Pluggable Database - Local Clone Begin	`com.oraclecloud.databaseservice.localclonepluggabledatabase.begin`
Pluggable Database - Local Clone End	`com.oraclecloud.databaseservice.localclonepluggabledatabase.end`
Pluggable Database - Remote Clone Begin	`com.oraclecloud.databaseservice.remoteclonepluggabledatabase.begin`
Pluggable Database - Remote Clone End	`com.oraclecloud.databaseservice.remoteclonepluggabledatabase.end`
Start Pluggable Database - Begin	`com.oraclecloud.databaseservice.startpluggabledatabase.begin`
Start Pluggable Database - End	`com.oraclecloud.databaseservice.startpluggabledatabase.end`
Stop Pluggable Database - Begin	`com.oraclecloud.databaseservice.stoppluggabledatabase.begin`
Stop Pluggable Database - End	`com.oraclecloud.databaseservice.stoppluggabledatabase.end`
Pluggable Database - Convert to Regular Begin	`com.oraclecloud.databaseservice.pluggabledatabase.converttoregular.begin`
Pluggable Database - Convert to Regular End	`com.oraclecloud.databaseservice.pluggabledatabase.converttoregular.end`
Pluggable Database - Inplace Restore Begin	`com.oraclecloud.databaseservice.pluggabledatabase.inplacerestore.begin`
Pluggable Database - Inplace Restore End	`com.oraclecloud.databaseservice.pluggabledatabase.inplacerestore.end`
Pluggable Database - Refresh Begin	`com.oraclecloud.databaseservice.pluggabledatabase.refresh.begin`

Friendly Name	Event Type
Pluggable Database - Refresh End	com.oraclecloud.databaseservice.pluggabledatabase.refresh.end
Pluggable Database - Relocate Begin	com.oraclecloud.databaseservice.pluggabledatabase.relocate.begin
Pluggable Database - Relocate End	com.oraclecloud.databaseservice.pluggabledatabase.relocate.end

NOT_SUPPORTED

This is a reference event for pluggable databases (PDBs):

```
{
  "eventID": "unique_id",
  "eventTime": "2021-03-23T00:49:14.123Z",
  "extensions": {
    "compartmentId": "ocid1.compartment.oc1.<unique_ID>"
  },
  "eventType":
"com.oraclecloud.databaseservice.remoteclonepluggabledatabase.begin",
  "eventTypeVersion": "2.0",
  "cloudEventsVersion": "0.1",
  "source": "databaseservice",
  "contentType": "application/json",
  "definedTags": {},
  "data": {
    "compartmentId": "ocid1.compartment.oc1.<unique_ID>",
    "compartmentName": "MyCompartment",
    "resourceName": "11092020_PKS_PDB1",
    "resourceId": "ocid1.pluggabledatabases.oc1.phx.<unique_ID>",
    "availabilityDomain": "XXIT:PHX-AD-1",
    "freeFormTags": {},
    "definedTags": {},
    "additionalDetails": {
      "id": "ocid1.pluggabledatabases.oc1.phx.<unique_ID>",
      "timeCreated": "2021-03-13T21:15:59.000Z",
      "timeUpdated": "2021-03-13T21:15:59.000Z",
      "databaseId": "ocid1.database.oc1.<unique_ID>",
      "lifecycleState": "AVAILABLE",
      "lifecycleDetails": "Pluggable Database is available",
      "displayName": "Pluggable Database - Remote Clone Begin"
    }
  }
}
```

This is a reference event for Pluggable Database - Convert to Regular Begin:

```
"exampleEvent": {
    "eventID": "unique_id",
    "eventTime": "2021-03-23T00:49:14.123Z",
    "extensions": {
      "compartmentId": "ocid1.compartment.oc1..unique_id"
    },
    "eventType":
"com.oraclecloud.databaseservice.pluggabledatabase.converttoregular.begin",
    "eventTypeVersion": "2.0",
    "cloudEventsVersion": "0.1",
    "source": "databaseservice",
    "contentType": "application/json",
    "definedTags": {},
    "data": {
      "compartmentId": "ocid1.compartment.oc1.......unique_id",
      "compartmentName": "MyCompartment",
      "resourceName": "11092020_PKS_PDB1",
      "resourceId": "ocid1.pluggabledatabases.oc1.phx.unique_id",
      "availabilityDomain": "XXIT:PHX-AD-1",
      "freeFormTags": {},
      "definedTags": {},
      "additionalDetails": {
        "id": "ocid1.pluggabledatabases.oc1.phx.unique_id",
        "isRefreshableClone": true,
        "timeCreated": "2021-03-13T21:15:59.000Z",
        "timeUpdated": "2021-03-13T21:15:59.000Z",
        "databaseId": "ocid1.database.oc1.....unique_id",
        "lifecycleState": "UPDATING",
        "displayName": "Pluggable Database - Convert to Regular Begin"
      }
    }
  },
  "activationTime": "2021-03-23T15:00:00.000Z",
  "eventTypeVersion": "2.0"
}
```

This is a reference event for Pluggable Database - Convert to Regular End:

```
"exampleEvent": {
    "eventID": "unique_id",
    "eventTime": "2021-03-23T00:49:14.123Z",
    "extensions": {
      "compartmentId": "ocid1.compartment.oc1..unique_id"
    },
    "eventType":
"com.oraclecloud.databaseservice.pluggabledatabase.converttoregular.end",
    "eventTypeVersion": "2.0",
    "cloudEventsVersion": "0.1",
    "source": "databaseservice",
    "contentType": "application/json",
    "definedTags": {},
    "data": {
      "compartmentId": "ocid1.compartment.oc1.......unique_id",
      "compartmentName": "MyCompartment",
      "resourceName": "11092020_PKS_PDB1",
      "resourceId": "ocid1.pluggabledatabases.oc1.phx.unique_id",
      "availabilityDomain": "XXIT:PHX-AD-1",
      "freeFormTags": {},
      "definedTags": {},
      "additionalDetails": {
        "id": "ocid1.pluggabledatabases.oc1.phx.unique_id",
        "isRefreshableClone": false,
        "timeCreated": "2021-03-13T21:15:59.000Z",
        "timeUpdated": "2021-03-13T21:15:59.000Z",
        "databaseId": "ocid1.database.oc1.....unique_id",
        "lifecycleState": "AVAILABLE",
        "displayName": "Pluggable Database - Convert to Regular End"
      }
    }
  },
  "activationTime": "2021-03-23T15:00:00.000Z",
  "eventTypeVersion": "2.0"
}
```

This is a reference event for Pluggable Database - Inplace Restore Begin:

```
"exampleEvent": {
    "eventID": "unique_id",
    "eventTime": "2021-03-23T00:49:14.123Z",
    "extensions": {
      "compartmentId": "ocid1.compartment.oc1..unique_id"
    },
    "eventType":
"com.oraclecloud.databaseservice.pluggabledatabase.inplacerestore.begin",
    "eventTypeVersion": "2.0",
    "cloudEventsVersion": "0.1",
    "source": "databaseservice",
    "contentType": "application/json",
    "definedTags": {},
    "data": {
      "compartmentId": "ocid1.compartment.oc1.......unique_id",
      "compartmentName": "MyCompartment",
      "resourceName": "11092020_PKS_PDB1",
      "resourceId": "ocid1.pluggabledatabases.oc1.phx.unique_id",
      "availabilityDomain": "XXIT:PHX-AD-1",
      "freeFormTags": {},
      "definedTags": {},
      "additionalDetails": {
        "id": "ocid1.pluggabledatabases.oc1.phx.unique_id",
        "timeCreated": "2021-03-13T21:15:59.000Z",
        "timeUpdated": "2021-03-13T21:15:59.000Z",
        "databaseId": "ocid1.database.oc1.....unique_id",
        "lifecycleState": "RESTORE_IN_PROGRESS",
        "isRefreshableClone": false,
        "displayName": "Pluggable Database - Inplace Restore Begin"
      }
    }
  },
  "activationTime": "2021-03-23T15:00:00.000Z",
  "eventTypeVersion": "2.0"
}
```

This is a reference event for Pluggable Database - Inplace Restore End:

```
"exampleEvent": {
    "eventID": "unique_id",
    "eventTime": "2021-03-23T00:49:14.123Z",
    "extensions": {
      "compartmentId": "ocid1.compartment.oc1..unique_id"
    },
    "eventType": 
"com.oraclecloud.databaseservice.pluggabledatabase.inplacerestore.end",
    "eventTypeVersion": "2.0",
    "cloudEventsVersion": "0.1",
    "source": "databaseservice",
    "contentType": "application/json",
    "definedTags": {},
    "data": {
      "compartmentId": "ocid1.compartment.oc1.......unique_id",
      "compartmentName": "MyCompartment",
      "resourceName": "11092020_PKS_PDB1",
      "resourceId": "ocid1.pluggabledatabases.oc1.phx.unique_id",
      "availabilityDomain": "XXIT:PHX-AD-1",
      "freeFormTags": {},
      "definedTags": {},
      "additionalDetails": {
        "id": "ocid1.pluggabledatabases.oc1.phx.unique_id",
        "timeCreated": "2021-03-13T21:15:59.000Z",
        "timeUpdated": "2021-03-13T21:15:59.000Z",
        "databaseId": "ocid1.database.oc1.....unique_id",
        "lifecycleState": "AVAILABLE",
        "isRefreshableClone": false,
        "lifecycleDetails": "Pluggable Database is available",
        "displayName": "Pluggable Database - Inplace Restore End"
      }
    }
  },
  "activationTime": "2021-03-23T15:00:00.000Z",
  "eventTypeVersion": "2.0"
}
```

This is a reference event for Pluggable Database - Refresh Begin:

```
"exampleEvent": {
    "eventID": "unique_id",
    "eventTime": "2021-03-23T00:49:14.123Z",
    "extensions": {
      "compartmentId": "ocid1.compartment.oc1..unique_id"
    },
    "eventType": 
"com.oraclecloud.databaseservice.pluggabledatabase.refresh.begin",
    "eventTypeVersion": "2.0",
    "cloudEventsVersion": "0.1",
    "source": "databaseservice",
    "contentType": "application/json",
    "definedTags": {},
    "data": {
      "compartmentId": "ocid1.compartment.oc1.......unique_id",
      "compartmentName": "MyCompartment",
      "resourceName": "11092020_PKS_PDB1",
      "resourceId": "ocid1.pluggabledatabases.oc1.phx.unique_id",
      "availabilityDomain": "XXIT:PHX-AD-1",
      "freeFormTags": {},
      "definedTags": {},
      "additionalDetails": {
        "id": "ocid1.pluggabledatabases.oc1.phx.unique_id",
        "timeCreated": "2021-03-13T21:15:59.000Z",
        "timeUpdated": "2021-03-13T21:15:59.000Z",
        "isRefreshableClone": true,
        "databaseId": "ocid1.database.oc1.....unique_id",
        "lifecycleState": "AVAILABLE",
        "lifecycleDetails": "Pluggable Database is available",
        "displayName": "Pluggable Database - Refresh Begin"
      }
    }
  },
  "activationTime": "2021-03-23T15:00:00.000Z",
  "eventTypeVersion": "2.0"
}
```

This is a reference event for Pluggable Database - Refresh End:

```
"exampleEvent": {
    "eventID": "unique_id",
    "eventTime": "2021-03-23T00:49:14.123Z",
    "extensions": {
      "compartmentId": "ocid1.compartment.oc1..unique_id"
    },
    "eventType":
"com.oraclecloud.databaseservice.pluggabledatabase.refresh.end",
    "eventTypeVersion": "2.0",
    "cloudEventsVersion": "0.1",
    "source": "databaseservice",
    "contentType": "application/json",
    "definedTags": {},
    "data": {
      "compartmentId": "ocid1.compartment.oc1.......unique_id",
      "compartmentName": "MyCompartment",
      "resourceName": "11092020_PKS_PDB1",
      "resourceId": "ocid1.pluggabledatabases.oc1.phx.unique_id",
      "availabilityDomain": "XXIT:PHX-AD-1",
      "freeFormTags": {},
      "definedTags": {},
      "additionalDetails": {
        "id": "ocid1.pluggabledatabases.oc1.phx.unique_id",
        "timeCreated": "2021-03-13T21:15:59.000Z",
        "timeUpdated": "2021-03-13T21:15:59.000Z",
        "databaseId": "ocid1.database.oc1.....unique_id",
        "lifecycleState": "AVAILABLE",
        "isRefreshableClone": true,
        "lifecycleDetails": "Pluggable Database is available",
        "displayName": "Pluggable Database - Refresh End"
      }
    }
  },
  "activationTime": "2021-03-23T15:00:00.000Z",
  "eventTypeVersion": "2.0"
}
```

This is a reference event for Pluggable Database - Relocate Begin:

```
"exampleEvent": {
    "eventID": "unique_id",
    "eventTime": "2021-03-23T00:49:14.123Z",
    "extensions": {
      "compartmentId": "ocid1.compartment.oc1..unique_id"
    },
    "eventType":
"com.oraclecloud.databaseservice.pluggabledatabase.relocate.begin",
    "eventTypeVersion": "2.0",
    "cloudEventsVersion": "0.1",
    "source": "databaseservice",
    "contentType": "application/json",
    "definedTags": {},
    "data": {
      "compartmentId": "ocid1.compartment.oc1.......unique_id",
      "compartmentName": "MyCompartment",
      "resourceName": "11092020_PKS_PDB1",
      "resourceId": "ocid1.pluggabledatabases.oc1.phx.unique_id",
      "availabilityDomain": "XXIT:PHX-AD-1",
      "freeFormTags": {},
      "definedTags": {},
      "additionalDetails": {
        "id": "ocid1.pluggabledatabases.oc1.phx.unique_id",
        "timeCreated": "2021-03-13T21:15:59.000Z",
        "timeUpdated": "2021-03-13T21:15:59.000Z",
        "databaseId": "ocid1.database.oc1.....unique_id",
        "lifecycleState": "AVAILABLE",
        "isRefreshableClone": false,
        "lifecycleDetails": "Pluggable Database is available",
        "displayName": "Pluggable Database - Relocate Begin"
      }
    }
  },
  "activationTime": "2021-03-23T15:00:00.000Z",
  "eventTypeVersion": "2.0"
}
```

This is a reference event for Pluggable Database - Relocate End:

```
"exampleEvent": {
    "eventID": "unique_id",
    "eventTime": "2021-03-23T00:49:14.123Z",
    "extensions": {
      "compartmentId": "ocid1.compartment.oc1..unique_id"
    },
    "eventType":
"com.oraclecloud.databaseservice.pluggabledatabase.relocate.end",
    "eventTypeVersion": "2.0",
    "cloudEventsVersion": "0.1",
    "source": "databaseservice",
    "contentType": "application/json",
    "definedTags": {},
    "data": {
      "compartmentId": "ocid1.compartment.oc1.......unique_id",
      "compartmentName": "MyCompartment",
      "resourceName": "11092020_PKS_PDB1",
      "resourceId": "ocid1.pluggabledatabases.oc1.phx.unique_id",
      "availabilityDomain": "XXIT:PHX-AD-1",
      "freeFormTags": {},
      "definedTags": {},
      "additionalDetails": {
        "id": "ocid1.pluggabledatabases.oc1.phx.unique_id",
        "timeCreated": "2021-03-13T21:15:59.000Z",
        "timeUpdated": "2021-03-13T21:15:59.000Z",
        "databaseId": "ocid1.database.oc1.....unique_id",
        "lifecycleState": "AVAILABLE",
        "lifecycleDetails": "Pluggable Database is available",
        "displayName": "Pluggable Database - Relocate End"
      }
    }
  },
  "activationTime": "2021-03-23T15:00:00.000Z",
  "eventTypeVersion": "2.0"
}
```

Database Service Events

The Database Service emits events, which are structured messages that indicate changes in resources.

- Overview of Database Service Events
  The Database Service Events feature implementation enables you to be notified about health issues with your Oracle Databases, or with other components on the Guest VM.
- Receive Notifications about Database Service Events
  Subscribe to the Database Service Events and get notified.
- Database Service Event Types
  Review the list of event types that the Database Service emits.
- Temporarily Restrict Automatic Diagnostic Collections for Specific Events
  Use the `tfactl blackout` command to temporarily suppress automatic diagnostic collections.

Overview of Database Service Events

The Database Service Events feature implementation enables you to be notified about health issues with your Oracle Databases, or with other components on the Guest VM.

It is possible that Oracle Database or Clusterware may not be healthy or various system components may be running out of space in the Guest VM. You are not notified of this situation, unless you opt-in.

> **Note:**
>
> You are opting in with the understanding that the list of events can change in the future. You can opt-out of this feature at any time

Database Service Events feature implementation generates events for Guest VM operations and conditions, as well as Notifications for customers by leveraging the existing OCI Events service and Notification mechanisms in their tenancy. Customers can then create topics and subscribe to these topics through email, functions, or streams.

> **Note:**
>
> Events flow on Oracle Exadata Database Service on Exascale Infrastructure depends on the following components: Oracle Trace File Analyzer (TFA), sysLens, and Oracle Database Cloud Service (DBCS) agent. Ensure that these components are up and running.

Manage Oracle Trace File Analyzer

- To check the run status of Oracle Trace File Analyzer, run the `tfactl status` command as `root` or a non-root user:

  ```
  # tfactl status
  .---------------------------------------------------------------------------
  -----------------------.
  | Host     | Status of TFA | PID    | Port | Version    | Build ID
  | Inventory Status|
  +----------------+---------------+--------+------+------------
  +----------------------+------------+
  | node1      | RUNNING     | 41312  | 5000 | 22.1.0.0.0 |
  2210002022031021 4615 | COMPLETE        |
  | node2      | RUNNING     | 272300 | 5000 | 22.1.0.0.0 |
  22100020220310214615 | COMPLETE        |
  '----------------+---------------+--------+------+------------
  +----------------------+------------'
  ```

- To start the Oracle Trace File Analyzer daemon on the local node, run the `tfactl start` command as `root`:

  ```
  # tfactl start
  Starting TFA..
  Waiting up to 100 seconds for TFA to be started..
  . . . . .
  . . . . .
  . . . . .
  . . . . .
  . . . . .
  . . . . .
  . . . . .
  . . . . .
  Successfully started TFA Process..
  ```

```
. . . . .
TFA Started and listening for commands
```

- To stop the Oracle Trace File Analyzer daemon on the local node, run the `tfactl stop` command as `root`:

  ```
  # tfactl stop
  Stopping TFA from the Command Line
  Nothing to do !
  Please wait while TFA stops
  Please wait while TFA stops
  TFA-00002 Oracle Trace File Analyzer (TFA) is not running
  TFA Stopped Successfully
  Successfully stopped TFA..
  ```

Manage sysLens

- If sysLens is running, then once every 15 minutes data is collected in the local domU to discover the events to be reported. To check if sysLens is running, run the `systemctl status syslens` command as `root` in the domU:

  ```
  # systemctl status syslens
  \u25cf syslens.service
  Loaded: loaded (/etc/systemd/system/syslens.service; disabled; vendor
  preset: disabled)
  Active: active (running) since Wed 2022-03-16 18:08:59 UTC; 34s ago
  Main PID: 358039 (python3)
  Memory: 31.6M
  CGroup: /system.slice/syslens.service
  \u2514\u2500358039 /usr/bin/python3 /var/opt/oracle/syslens/bin/
  syslens_main.py --archive /var/opt/oracle/log/...

  Mar 16 18:08:59 node1 systemd[1]: Started syslens.service.
  Mar 16 18:09:09 node1 su[360495]: (to oracle) root on none
  Mar 16 18:09:09 node1 su[360539]: (to grid) root on none
  Mar 16 18:09:10 node1 su[360611]: (to grid) root on none
  Mar 16 18:09:11 node1 su[360653]: (to oracle) root on none
  ```

- If the sysLens is enabled, when there is a reboot of the domU, then sysLens starts automatically. To validate if sysLens is enabled to collect telemetry, run the `systemctl is-enabled syslens` command as `root` in the domU:

  ```
  # systemctl is-enabled syslens
  enabled
  ```

- To validate if sysLens is configured to notify events, run the `/usr/bin/syslens --config /var/opt/oracle/syslens/data/exacc.syslens.config --get-key enable_telemetry` command as `root` in the domU:

  ```
  # /usr/bin/syslens --config /var/opt/oracle/syslens/data/
  exacc.syslens.config --get-key enable_telemetry
  syslens Collection 2.3.3
  on
  ```

Manage Database Service Agent

View the `/opt/oracle/dcs/log/dcs-agent.log` file to identify issues with the agent.

- To check the status of the Database Service Agent, run the `systemctl status` command:

  ```
  # systemctl status dbcsagent.service
  dbcsagent.service
  Loaded: loaded (/usr/lib/systemd/system/dbcsagent.service; enabled; vendor
  preset: disabled)
  Active: active (running) since Fri 2022-04-01 13:40:19 UTC; 6min ago
  Process: 9603 ExecStopPost=/bin/bash -c kill `ps -fu opc |grep "java.*dbcs-
  agent.*jar" |awk '{print $2}' ` (code=exited, status=0/SUCCESS)
  Main PID: 10055 (sudo)
  CGroup: /system.slice/dbcsagent.service
  □ 10055 sudo -u opc /bin/bash -c umask 077; /bin/java -
  Doracle.security.jps.config=/opt/oracle/...
  ```

- To start the agent if it is not running, run the `systemctl start` command as the `root` user:

  ```
  systemctl start dbcsagent.service
  ```

Related Topics

- Using the Console to Enable, Partially Enable, or Disable Diagnostics Collection
  You can enable, partially enable, or disable diagnostics collection for your Guest VMs after provisioning the VM cluster. Enabling diagnostics collection at the VM cluster level applies the configuration to all the resources such as DB home, Database, and so on under the VM cluster.
- Overview of Events
- Notifications Overview

Receive Notifications about Database Service Events

Subscribe to the Database Service Events and get notified.

To receive notifications, subscribe to Database Service Events and get notified using the Oracle Notification service, see *Notifications Overview*. For more information about Oracle Cloud Infrastructure Events, see *Overview of Events*.

Events Service - Event Types:

- Database - Critical
- DB Node - Critical
- DB Node - Error
- DB Node - Warning
- DB Node - Information
- DB System - Critical

Related Topics

- Overview of Events
- Notifications Overview

Database Service Event Types

Review the list of event types that the Database Service emits.

> **Note:**
>
> - Critical events are triggered due to several types of critical conditions and errors that cause disruption to the database and other critical components. For example, database hang errors, and availability errors for databases, database nodes, and database systems to let you know if a resource becomes unavailable.
> - Information events are triggered when the database and other critical components work as expected. For example, a clean shutdown of CRS, CDB, client, or scan listener, or a startup of these components will create an event with the severity of INFORMATION.
> - Threshold limits reduce the number of notifications customers will receive for similar incident events whilst at the same time ensuring they receive the incident events and are reminded in a timely fashion.

Table 5-3 Database Service Events

Friendly Name	Event Name	Event Type	Threshold
Resource Utilization - Disk Usage	`HEALTH.DB_GUEST.FILESYSTEM.FREE_SPACE` This event is reported when VM guest file system free space falls below 10% free, as determined by the operating system `df(1)` command, for the following file systems: • `/` • `/u01` • `/u02` • `/var` (X8M and later only) • `/tmp` (X8M and later only)	`com.oraclecloud.databaseservice.dbnode.critical`	Critical Threshold: 90%
CRS status Up/Down	`AVAILABILITY.DB_GUEST.CRS_INSTANCE.DOWN.` An event of type CRITICAL is created when the Cluster Ready Service (CRS) is detected to be down.	`com.oraclecloud.databaseservice.dbnode.critical` (if .DOWN and NOT "user_action")	N/A

Table 5-3 (Cont.) Database Service Events

Friendly Name	Event Name	Event Type	Threshold
	`AVAILABILITY.DB_GUEST.CRS_INSTANCE.DOWN_CLEARED` An event of type INFORMATION is created once it is determined that the event for CRS down has cleared.	`com.oraclecloud.databaseservice.dbnode.information` (if .DOWN_CLEARED)	N/A
SCAN Listener Up/Down	`AVAILABILITY.DB_CLUSTER.SCAN_LISTENER.DOWN` A DOWN event is created when a SCAN listener goes down. The event is of type INFORMATION when a SCAN listener is shutdown due to user action, such as with the Server Control Utility (`srvctl`) or Listener Control (`lsnrctl`) commands, or any Oracle Cloud maintenance action that uses those commands, such as performing a grid infrastructure software update. The event is of type CRITICAL when a SCAN listener goes down unexpectedly. A corresponding DOWN_CLEARED event is created when a SCAN listener is started. There are three SCAN listeners per cluster called LISTENER_SCAN[1,2,3].	`com.oraclecloud.databaseservice.dbnode.critical` (if .DOWN and NOT "user_action")	N/A
	`AVAILABILITY.DB_CLUSTER.SCAN_LISTENER.DOWN_CLEARED` An event of type INFORMATION is created once it is determined that the event for SCAN Listener down has cleared.	`com.oraclecloud.databaseservice.dbnode.information` (if .DOWN_CLEARED)	N/A

Table 5-3 (Cont.) Database Service Events

Friendly Name	Event Name	Event Type	Threshold
Net Listener Up/Down	`AVAILABILITY.DB_GUEST.CLIENT_LISTENER.DOWN` A DOWN event is created when a client listener goes down. The event is of type INFORMATION when a client listener is shutdown due to user action, such as with the Server Control Utility (`srvctl`) or Listener Control (`lsnrctl`) commands, or any Oracle Cloud maintenance action that uses those commands, such as performing a grid infrastructure software update. The event is of type CRITICAL when a client listener goes down unexpectedly. A corresponding DOWN_CLEARED event is created when a client listener is started. There is one client listener per node, each called LISTENER.	`com.oraclecloud.databaseservice.database.critical` (if .DOWN and NOT "user_action")	N/A
	`AVAILABILITY.DB_GUEST.CLIENT_LISTENER.DOWN_CLEARED` An event of type INFORMATION is created once it is determined that the event for Client Listener down has cleared.	`com.oraclecloud.databaseservice.database.information` (if .DOWN_CLEARED)	N/A

Table 5-3 (Cont.) Database Service Events

Friendly Name	Event Name	Event Type	Threshold
CDB Up/Down	`AVAILABILITY.DB_GUEST.CDB_INSTANCE.DOWN` A DOWN event is created when a database instance goes down. The event is of type INFORMATION when a database instance is shutdown due to user action, such as with the SQL*Plus (`sqlplus`) or Server Control Utility (`srvctl`) commands, or any Oracle Cloud maintenance action that uses those commands, such as performing a database home software update. The event is of type CRITICAL when a database instance goes down unexpectedly. A corresponding DOWN_CLEARED event is created when a database instance is started.	`com.oraclecloud.databaseservice.database.critical` (if .DOWN and NOT "user_action")	N/A
	`AVAILABILITY.DB_GUEST.CDB_INSTANCE.DOWN_CLEARED` An event of type INFORMATION is created once it is determined that the event for the CDB down has cleared.	`com.oraclecloud.databaseservice.database.information` (if .DOWN_CLEARED)	N/A
CRS Eviction	`AVAILABILITY.DB_GUEST.CRS_INSTANCE.EVICTION` An event of type CRITICAL is created when the Cluster Ready Service (CRS) evicts a node from the cluster. The CRS alert.log is parsed for the CRS-1632 error indicating that a node is being removed from the cluster.	An event of type CRITICAL is created when the Cluster Ready Service (CRS) evicts a node from the cluster. The CRS alert.log is parsed for the CRS-1632 error indicating that a node is being removed from the cluster.	N/A

Table 5-3 (Cont.) Database Service Events

Friendly Name	Event Name	Event Type	Threshold
Critical DB Errors	`HEALTH.DB_CLUSTER.CDB.CORRUPTION` Database corruption has been detected on your primary or standby database. The database alert.log is parsed for any specific errors that are indicative of physical block corruptions, logical block corruptions, or logical block corruptions caused by lost writes.	`com.oraclecloud.databaseservice.database.critical`	N/A
Other DB Errors	`HEALTH.DB_CLUSTER.CDB.ARCHIVER_HANG` An event of type CRITICAL is created if a CDB is either unable to archive the active online redo log or unable to archive the active online redo log fast enough to the log archive destinations.	`com.oraclecloud.databaseservice.database.critical`	N/A
	`HEALTH.DB_CLUSTER.CDB.DATABASE_HANG` An event of type CRITICAL is created when a process/session hang is detected in the CDB.	N/A	N/A
Backup Failures	`HEALTH.DB_CLUSTER.CDB.BACKUP_FAILURE` An event of type CRITICAL is created if there is a CDB backup with a FAILED status reported in the `v$rman_status` view.	`com.oraclecloud.databaseservice.database.critical`	N/A
Disk Group Usage	`HEALTH.DB_CLUSTER.DISK_GROUP.FREE_SPACE` An event of type CRITICAL is created when an ASM disk group reaches space usage of 90% or higher. An event of type INFORMATION is created when the ASM disk group space usage drops below 90%.	`com.oraclecloud.databaseservice.dbsystem.critical` `com.oraclecloud.databaseservice.dbsystem.information` (if < 90%)	Critical threshold: 90%

Table 5-3 (Cont.) Database Service Events

Friendly Name	Event Name	Event Type	Threshold
Memory Usage	`CONFIGURATION.DB_GUEST.MEMORY.HUGEPAGES_TOO_LARGE` An event of type CRITICAL is created when the amount of memory in the VM configured for HugePages is 90% or more of the total VM memory.	`com.oraclecloud.databaseservice.dbnode.critical`	90%
sshd Configuration	`CONFIGURATION.DB_GUEST.SSHD.INVALID` An event of type CRITICAL is created if unexpected values are set in the `/etc/ssh/sshd_config` file.	`com.oraclecloud.databaseservice.dbnode.critical`	N/A
Disk Issues	`HEALTH.DB_GUEST.FILESYSTEM.CORRUPTION` A Write-then-Read operation with a dummy file has failed for a file system, typically indicating the operating system had detected an I/O error or inconsistency (i.e. corruption) with the file system and changed the file system mount mode from read-write to read-only. The following file systems are tested: • `/` • `/u01` • `/u02`	`com.oraclecloud.databaseservice.dbnode.critical`	N/A

Table 5-3 (Cont.) Database Service Events

Friendly Name	Event Name	Event Type	Threshold
Oracle EXAchk Reported Issues	`HEALTH.DB_CLUSTER.EXACHK.CRITICAL_ALERT` Oracle EXAchk is Exadata database platform's holistic health check that includes software, infrastructure and database configuration checks. CRITICAL check alerts should be addressed in 24 hours to maintain the maximum stability and availability of your system. This database service event alerts every 24 hours whenever there are any CRITICAL checks that are flagged in the most recent Oracle EXAchk report. The event will point to the latest Oracle EXAchk zip report.	`com.oraclecloud.databaseservice.dbnode.critical`	N/A

Example 5-66 Database Service DB Node Critical Events Examples

DB node critical reference events:

```
{
  "eventType" : "com.oraclecloud.databaseservice.dbnode.critical",
  "cloudEventsVersion" : "0.1",
  "eventTypeVersion" : "2.0",
  "source" : "SYSLENS/host_Name/DomU",
  "eventTime" : "2022-03-04T18:19:42Z",
  "contentType" : "application/json",
  "data" : {
    "compartmentId" : "compartment_ID",
    "compartmentName" : "compartment_Name",
    "resourceName" : "resource_Name",
    "resourceId" : "resource_ID",
    "additionalDetails" : {
      "serviceType" : "EXACS",
      "hostName" : "host_Name",
      "description" : "The '/' filesystem is over 90% used.",
      "eventName" : "HEALTH.DB_GUEST.FILESYSTEM.FREE_SPACE",
      "status" : "online"
    }
  },
  "eventID" : "a9752630-9be7-11ec-a203-00163eb980bb",
  "extensions" : {
    "compartmentId" : "compartment_ID"
```

```
 }
}
```

Temporarily Restrict Automatic Diagnostic Collections for Specific Events

Use the `tfactl blackout` command to temporarily suppress automatic diagnostic collections.

If you set blackout for a target, then Oracle Trace File Analyzer stops automatic diagnostic collections if it finds events in the alert logs for that target while scanning. By default, blackout will be in effect for 24 hours.

You can also restrict automatic diagnostic collection at a granular level, for example, only for **ORA-00600** or even only **ORA-00600** with specific arguments.

Syntax

```
tfactl blackout add|remove|print
-targettype host|crs|asm|asmdg|database|dbbackup|db_dataguard|db_tablespace|
pdb_tablespace|pdb|listener|service|os
-target all|name
[-container name]
[-pdb pdb_name]
-event all|"event_str1,event_str2"|availability
[-timeout nm|nh|nd|none]
[-c|-local|-nodes "node1,node2"]
[-reason "reason for blackout"]
[-docollection]
```

Parameters

Table 5-4 tfactl blackout Command Parameters

Parameter	Description
`add|remove|print|`	Adds, removes, or prints blackout conditions.

Table 5-4 (Cont.) tfactl blackout Command Parameters

Parameter	Description
`targettype` *`type`* **Target type:** `host`|`crs`|`asm`|`asmdg`|`database`|`dbbackup`|`db_dataguard`|`db_tablespace` |`pdb_tablespace`|`pdb`|`listener`|`service`|`os`	Limits blackout only to the specified target type. `host`: The whole node is under blackout. If there is host blackout, then every blackout element that's shown true in the Telemetry JSON will have the reason for the blackout. `crs`: Blackout the availability of the Oracle Clusterware resource or events in the Oracle Clusterware logs. `asm`: Blackout the availability of Oracle Automatic Storage Management (Oracle ASM) on this machine or events in the Oracle ASM alert logs. `asmdg`: Blackout an Oracle ASM disk group. `database`: Blackout the availability of an Oracle Database, Oracle Database backup, tablespace, and so on, or events in the Oracle Database alert logs. `dbbackup`: Blackout Oracle Database backup events (such as CDB or archive backups). `db_dataguard`: Blackout Oracle Data Guard events. `db_tablespace`: Blackout Oracle Database tablespace events (container database). `pdb_tablespace`: Blackout Oracle Pluggable Database tablespace events (Pluggable database). `pdb`: Blackout Oracle Pluggable Database events. `listener`: Blackout the availability of a listener. `service`: Blackout the availability of a service. `os`: Blackout one or more operating system records.
`target all|`*`name`*	Specify the target for blackout. You can specify a comma-delimited list of targets. By default, the target is set to `all`.
`container` *`name`*	Specify the database container name (`db_unique_name`) where the blackout will take effect (for PDB, `DB_TABLESPACE`, and `PDB_TABLESPACE`).
`pdb` *`pdb_name`*	Specify the PDB where the blackout will take effect (for `PDB_TABLESPACE` only).
`events all|"`*`str1`*`,`*`str2`*`"`	Limits blackout only to the availability events, or event strings, which should not trigger auto collections, or be marked as blacked out in telemetry JSON. `all`: Blackout everything for the target specified. *string*: Blackout for incidents where any part of the line contains the strings specified. Specify a comma-delimited list of strings.
`timeout` *`nh`*`|`*`nd`*`|none`	Specify the duration for blackout in number of hours or days before timing out. By default, the timeout is set to 24 hours (24h).
`c|local`	Specify if blackout should be set to cluster-wide or local. By default, blackout is set to `local`.
`reason` *`comment`*	Specify a descriptive reason for the blackout.
`docollection`	Use this option to do an automatic diagnostic collection even if a blackout is set for this target.

Example 5-67 tfactl blackout

- To blackout **event:** ORA-00600 on **target type:** database, **target:** mydb

  ```
  tfactl blackout add -targettype database -target mydb -event "ORA-00600"
  ```

- To blackout **event:** ORA-04031 on **target type:** database, **target:** all

  ```
  tfactl blackout add -targettype database -target all -event "ORA-04031" -
  timeout 1h
  ```

- To blackout **db backup events** on **target type:** dbbackup, **target:** mydb

  ```
  tfactl blackout add -targettype dbbackup -target mydb
  ```

- To blackout **db dataguard events** on **target type:** db_dataguard, **target:** mydb

  ```
  tfactl blackout add -targettype db_dataguard -target mydb -timeout 30m
  ```

- To blackout **db tablespace events** on **target type:** db_tablespace, **target:** system, **container:** mydb

  ```
  tfactl blackout add -targettype db_tablespace -target system -container
  mydb -timeout 30m
  ```

- To blackout **ALL events** on **target type:** host, **target:** all

  ```
  tfactl blackout add -targettype host -event all -target all -timeout 1h -
  reason "Disabling all events during patching"
  ```

- To print blackout details

  ```
  tfactl blackout print



  .--------------------------------------------------------------------------
  ---------------------------------------------------------------------------
  --------------------------.
  |

  myhostname
                      |
  +---------------+---------------------+-----------
  +------------------------------+------------------------------+--------
  +---------------+---------------------------------------+
  | Target Type   | Target              | Events    | Start
  Time                   | End Time                     | Status | Do
  Collection | Reason                               |
  +---------------+---------------------+-----------
  +------------------------------+------------------------------+--------
  +---------------+---------------------------------------+
  | HOST          | ALL                 | ALL       | Thu Mar 24 16:48:39
  UTC 2022 | Thu Mar 24 17:48:39 UTC 2022 | ACTIVE | false         |
  Disabling all events during patching |
  ```

```
| DATABASE      | MYDB                 | ORA-00600 | Thu Mar 24 16:39:03
UTC 2022 | Fri Mar 25 16:39:03 UTC 2022 | ACTIVE | false         |
NA                                   |
| DATABASE      | ALL                  | ORA-04031 | Thu Mar 24 16:39:54
UTC 2022 | Thu Mar 24 17:39:54 UTC 2022 | ACTIVE | false         |
NA                                   |
| DB_DATAGUARD  | MYDB                 | ALL       | Thu Mar 24 16:41:38
UTC 2022 | Thu Mar 24 17:11:38 UTC 2022 | ACTIVE | false         |
NA                                   |
| DBBACKUP      | MYDB                 | ALL       | Thu Mar 24 16:40:47
UTC 2022 | Fri Mar 25 16:40:47 UTC 2022 | ACTIVE | false         |
NA                                   |
| DB_TABLESPACE | SYSTEM_CDBNAME_MYDB | ALL       | Thu Mar 24 16:45:56
UTC 2022 | Thu Mar 24 17:15:56 UTC 2022 | ACTIVE | false         |
NA                                   |
'---------------+---------------------+-----------
+------------------------------+------------------------------+--------
+---------------+--------------------------------------'
```

- To remove blackout for **event:** ORA-00600 on **target type:** database, **target:** mydb

  ```
  tfactl blackout remove -targettype database -event "ORA-00600" -target mydb
  ```

- To remove blackout for **db backup events** on **target type:** dbbackup, **target:** mydb

  ```
  tfactl blackout remove -targettype dbbackup -target mydb
  ```

- To remove blackout for **db tablespace events** on **target type:** db_tablespace, **target:** system, **container:** mydb

  ```
  tfactl blackout remove -targettype db_tablespace -target system -container
  mydb
  ```

- To remove blackout for **host events** on **target type:** host, **target:** all

  ```
  tfactl blackout remove -targettype host -event all -target all
  ```

Application VIP Event Types

These are the event types that Application VIPs in Oracle Cloud Infrastructure emit.

Friendly Name	Event Type
Application Virtual IP (VIP) - Create Begin	`com.oraclecloud.databaseservice.createapplicationvip.begin`
Application Virtual IP (VIP) - Create End	`com.oraclecloud.databaseservice.createapplicationvip.end`
Application Virtual IP (VIP) - Delete Begin	`com.oraclecloud.databaseservice.deleteapplicationvip.begin`
Application Virtual IP (VIP) - Delete End	`com.oraclecloud.databaseservice.deleteapplicationvip.end`

Application VIP Event Types Examples:

This is a reference event for Application Virtual IP (VIP) - Create Begin:

```
{
  "id":
"ocid1.eventschema.oc1.phx.5ur5er8bddumnu9r84rtt2c3282s5no31vsthibyqvvsisotnwp
csg9idv6q",
  "serviceName": "Database",
  "displayName": "Application Virtual IP (VIP) - Create Begin",
  "eventType": "com.oraclecloud.databaseservice.createapplicationvip.begin",
  "source": "databaseservice",
  "eventTypeVersion": "2.0",
  "eventTime": "2022-12-15T21:16:04.000Z",
  "contentType": "application/json",
  "additionalDetails": [
    {
      "name": "id",
      "type": "string"
    },
    {
      "name": "definedTags",
      "type": [
        "null",
        "Map<String, Map<String, Object>>"
      ]
    },
    {
      "name": "freeFormTags",
      "type": [
        "null",
        "Map<String, String>"
      ]
    },
    {
      "name": "timeCreated",
      "type": "string"
    },
    {
      "name": "timeUpdated",
      "type": "string"
    },
    {
      "name": "lifecycleState",
      "type": "string"
    },
    {
      "name": "lifecycleDetails",
      "type": [
        "null",
        "string"
      ]
    },
    {
      "name": "hostnameLabel",
      "type": [
        "null",
        "string"
    ]
  },
  {
    "name": "cloudVmClusterId",
    "type": [
      "null",
      "string"
    ]
  },
  {
    "name": "compartmentId",
    "type": [
      "null",
      "string"
    ]
  },
  {
    "name": "vcnIpId",
    "type": [
      "null",
      "string"
    ]
  },
  {
    "name": "ipAddress",
    "type": [
      "null",
      "string"
    ]
  },
  {
    "name": "subnetId",
    "type": [
      "null",
      "string"
    ]
  },
  {
    "name": "networkType",
    "type": [
      "null",
      "string"
    ]
  }
],
"exampleEvent": {
  "eventType": "com.oraclecloud.databaseservice.createapplicationvip.begin",
  "cloudEventsVersion": "0.1",
  "eventTypeVersion": "2.0",
  "source": "databaseservice",
  "contentType": "application/json",
  "eventID": "ab2ac219-b435-1045-aaf3-13cd909ec106",
  "eventTime": "2022-12-16T21:16:04.000Z",
  "data": {
    "resourceId": "ocid1.applicationvip.oc1.....unique_id",
    "resourceName": "my_application_vip",
      "tagSlug": null,
      "compartmentId": "ocid1.compartment.oc1.....unique_id",
      "request": {
        "id": "4260c9fd-d36b-4bc8-866e-c2dd53f34b2f",
        "path": null,
        "action": null,
        "parameters": null,
        "headers": null
      },
      "response": {
        "status": null,
        "responseTime": null,
        "headers": null,
        "payload": null,
        "message": ""
      },
      "stateChange": {
        "previous": null,
        "current": {
          "lifecycleState": "PROVISIONING",
          "hostnameLabel": "my_application_vip",
          "freeTags": {},
          "definedTags": {}
        }
      },
      "eventGroupingId": "csid74237ee84398b60cf1b834c81602/
f43a881dc99542318d46fa9285bdf2c5/6AC9F7641E1A5AD5C27D1650CB17E822",
      "eventName": "CreateApplicationVip",
      "availabilityDomain": "",
      "resourceVersion": null,
      "additionalDetails": {
        "id": "ocid1.applicationvip.oc1.....unique_id",
        "freeformTags": {},
        "definedTags": {},
        "timeCreated": "2022-12-15T21:17:59.000Z",
        "timeUpdated": "2022-12-15T21:18:04.389Z",
        "lifecycleState": "PROVISIONING",
        "lifecycleDetails": "",
        "hostnameLabel": "my_application_vip",
        "cloudVmClusterId": "ocid1.cloudvmcluster.oc1.....unique_id",
        "compartmentId": "ocid1.compartment.oc1.....unique_id",
        "vcnIpId": "ocid1.privateip.oc1.....unique_id",
        "ipAddress": "10.0.0.0",
        "subnetId": "ocid1.subnet.oc1.....unique_id",
        "networkType": "CLIENT"
      }
    }
  },
  "timeCreated": "2022-12-15T16:31:31.979Z"
}
```

This is a reference event for Application Virtual IP (VIP) - Create End:

```
{
  "id":
```

```
"ocid1.eventschema.oc1.phx.c1ok1948lwge4il6m85ta4jdlbnh1yjrjltrabujyv52calb0el
p263oyqrm",
  "serviceName": "Database",
  "displayName": "Application Virtual IP (VIP) - Create End",
  "eventType": "com.oraclecloud.databaseservice.createapplicationvip.end",
  "source": "databaseservice",
  "eventTypeVersion": "2.0",
  "eventTime": "2022-12-15T21:16:04.000Z",
  "contentType": "application/json",
  "additionalDetails": [
    {
      "name": "id",
      "type": "string"
    },
    {
      "name": "definedTags",
      "type": [
        "null",
        "Map<String, Map<String, Object>>"
      ]
    },
    {
      "name": "freeFormTags",
      "type": [
        "null",
        "Map<String, String>"
      ]
    },
    {
      "name": "timeCreated",
      "type": "string"
    },
    {
      "name": "timeUpdated",
      "type": "string"
    },
    {
      "name": "lifecycleState",
      "type": "string"
    },
    {
      "name": "lifecycleDetails",
      "type": [
        "null",
        "string"
      ]
    },
    {
      "name": "hostnameLabel",
      "type": [
        "null",
        "string"
      ]
    },
    {
      "name": "cloudVmClusterId",
      "type": [
        "null",
        "string"
      ]
    },
    {
      "name": "compartmentId",
      "type": [
        "null",
        "string"
      ]
    },
    {
      "name": "vcnIpId",
      "type": [
        "null",
        "string"
      ]
    },
    {
      "name": "ipAddress",
      "type": [
        "null",
        "string"
      ]
    },
    {
      "name": "subnetId",
      "type": [
        "null",
        "string"
      ]
    },
    {
      "name": "networkType",
      "type": [
        "null",
        "string"
      ]
    }
  ],
  "exampleEvent": {
    "eventType": "com.oraclecloud.databaseservice.createapplicationvip.end",
    "cloudEventsVersion": "0.1",
    "eventTypeVersion": "2.0",
    "source": "databaseservice",
    "contentType": "application/json",
    "eventID": "bc122d87-ac42-8731-ccd1-09ab320eef11",
    "eventTime": "2022-12-16T21:16:04.000Z",
    "data": {
      "resourceId": "ocid1.applicationvip.oc1.....unique_id",
      "resourceName": "my_application_vip",
      "tagSlug": null,
      "compartmentId": "ocid1.compartment.oc1.....unique_id",
      "request": {
        "id": "195eb9b5-b5a0-474d-a1c3-86189d8eeb2c",
        "path": null,
        "action": null,
        "parameters": null,
        "headers": null
      },
      "response": {
        "status": null,
        "responseTime": null,
        "headers": null,
        "payload": null,
        "message": ""
      },
      "stateChange": {
        "previous": null,
        "current": {
          "lifecycleState": "AVAILABLE",
          "hostnameLabel": "my_application_vip",
          "freeTags": {},
          "definedTags": {}
        }
      },
      "eventGroupingId": 
"6CEB05B6C81E4B19855AD716E90F5BC3/070ECF4976BDD89B16849A92B95564A6/1418EDD7590
B8D5DDFF947FC3161F358",
      "eventName": "CreateApplicationVip",
      "availabilityDomain": "",
      "resourceVersion": null,
      "additionalDetails": {
        "id": "ocid1.applicationvip.oc1.....unique_id",
        "freeformTags": {},
        "definedTags": {},
        "timeCreated": "2022-12-15T21:17:59.000Z",
        "timeUpdated": "2022-12-15T21:18:04.389Z",
        "lifecycleState": "AVAILABLE",
        "lifecycleDetails": "",
        "hostnameLabel": "my_application_vip",
        "cloudVmClusterId": "ocid1.cloudvmcluster.oc1.....unique_id",
        "compartmentId": "ocid1.compartment.oc1.....unique_id",
        "vcnIpId": "ocid1.privateip.oc1.....unique_id",
        "ipAddress": "10.0.0.0",
        "subnetId": "ocid1.subnet.oc1.....unique_id",
        "networkType": "CLIENT"
      }
    }
  },
  "timeCreated": "2022-12-15T16:31:31.979Z"
}
```

This is a reference event for Application Virtual IP (VIP) - Delete Begin:

```
{
  "id": 
"ocid1.eventschema.oc1.phx.m2gheil6f1nfzb9ggpkkv17wdomdks8zin9nntqlghui6bckh17
yu0m5jcqt",
  "serviceName": "Database",
"displayName": "Application Virtual IP (VIP) - Delete Begin",
"eventType": "com.oraclecloud.databaseservice.deleteapplicationvip.begin",
"source": "databaseservice",
"eventTypeVersion": "2.0",
"eventTime": "2022-12-15T21:16:04.000Z",
"contentType": "application/json",
"additionalDetails": [
  {
    "name": "id",
    "type": "string"
  },
  {
    "name": "definedTags",
    "type": [
      "null",
      "Map<String, Map<String, Object>>"
    ]
  },
  {
    "name": "freeFormTags",
    "type": [
      "null",
      "Map<String, String>"
    ]
  },
  {
    "name": "timeCreated",
    "type": "string"
  },
  {
    "name": "timeUpdated",
    "type": "string"
  },
  {
    "name": "lifecycleState",
    "type": "string"
  },
  {
    "name": "lifecycleDetails",
    "type": [
      "null",
      "string"
    ]
  },
  {
    "name": "hostnameLabel",
    "type": [
      "null",
      "string"
    ]
  },
  {
    "name": "cloudVmClusterId",
    "type": [
      "null",
      "string"
    ]
  },
  {
    "name": "compartmentId",
    "type": [
      "null",
      "string"
    ]
  },
  {
    "name": "vcnIpId",
    "type": [
      "null",
      "string"
    ]
  },
  {
    "name": "ipAddress",
    "type": [
      "null",
      "string"
    ]
  },
  {
    "name": "subnetId",
    "type": [
      "null",
      "string"
    ]
  },
  {
    "name": "networkType",
    "type": [
      "null",
      "string"
    ]
  }
],
"exampleEvent": {
  "eventType": "com.oraclecloud.databaseservice.deleteapplicationvip.begin",
  "cloudEventsVersion": "0.1",
  "eventTypeVersion": "2.0",
  "source": "databaseservice",
  "contentType": "application/json",
  "eventID": "e32cb1fe-123d-8341-de13-2be5f18ab31e",
  "eventTime": "2022-12-16T21:16:04.000Z",
  "data": {
    "resourceId": "ocid1.applicationvip.oc1.....unique_id",
    "resourceName": "my_application_vip",
    "tagSlug": null,
    "compartmentId": "ocid1.compartment.oc1.....unique_id",
    "request": {
      "id": "23a08e08-6b1e-40f0-a027-f2601dfd44ea",
      "path": null,
      "action": null,
      "parameters": null,
        "headers": null
      },
      "response": {
        "status": null,
        "responseTime": null,
        "headers": null,
        "payload": null,
        "message": ""
      },
      "stateChange": {
        "previous": null,
        "current": {
          "lifecycleState": "TERMINATING",
          "hostnameLabel": "my_application_vip",
          "freeTags": {},
          "definedTags": {}
        }
      },
      "eventGroupingId": "csidb3f42d234534bc8bc8849b892e84/
fbd51970d2a2486f94671614b5ea0571/9DFE1BEB5433FF69BABCCB7E34F2EAF4",
      "eventName": "DeleteApplicationVip",
      "availabilityDomain": "",
      "resourceVersion": null,
      "additionalDetails": {
        "id": "ocid1.applicationvip.oc1.....unique_id",
        "freeformTags": {},
        "definedTags": {},
        "timeCreated": "2022-12-15T21:17:59.000Z",
        "timeUpdated": "2022-12-15T21:18:04.389Z",
        "lifecycleState": "TERMINATING",
        "lifecycleDetails": "",
        "hostnameLabel": "my_application_vip",
        "cloudVmClusterId": "ocid1.cloudvmcluster.oc1.....unique_id",
        "compartmentId": "ocid1.compartment.oc1.....unique_id",
        "vcnIpId": "ocid1.privateip.oc1.....unique_id",
        "ipAddress": "10.0.0.0",
        "subnetId": "ocid1.subnet.oc1.....unique_id",
        "networkType": "CLIENT"
      }
    }
  },
  "timeCreated": "2022-12-15T16:31:31.979Z"
}
```

This is a reference event for Application Virtual IP (VIP) - Delete End:

```
{
  "id":
"ocid1.eventschema.oc1.phx.9d1tjgkavhn0rq4qdlmofrjro9npvugu73dp07uht0igxs9732x
6yar1m5l5",
  "serviceName": "Database",
  "displayName": "Application Virtual IP (VIP) - Delete End",
  "eventType": "com.oraclecloud.databaseservice.deleteapplicationvip.end",
  "source": "databaseservice",
  "eventTypeVersion": "2.0",
    "eventTime": "2022-12-15T21:16:04.000Z",
    "contentType": "application/json",
    "additionalDetails": [
      {
        "name": "id",
        "type": "string"
      },
      {
        "name": "definedTags",
        "type": [
          "null",
          "Map<String, Map<String, Object>>"
        ]
      },
      {
        "name": "freeFormTags",
        "type": [
          "null",
          "Map<String, String>"
        ]
      },
      {
        "name": "timeCreated",
        "type": "string"
      },
      {
        "name": "timeUpdated",
        "type": "string"
      },
      {
        "name": "lifecycleState",
        "type": "string"
      },
      {
        "name": "lifecycleDetails",
        "type": [
          "null",
          "string"
        ]
      },
      {
        "name": "hostnameLabel",
        "type": [
          "null",
          "string"
        ]
      },
      {
        "name": "cloudVmClusterId",
        "type": [
          "null",
          "string"
        ]
      },
      {
        "name": "compartmentId",
    "type": [
      "null",
      "string"
    ]
  },
  {
    "name": "vcnIpId",
    "type": [
      "null",
      "string"
    ]
  },
  {
    "name": "ipAddress",
    "type": [
      "null",
      "string"
    ]
  },
  {
    "name": "subnetId",
    "type": [
      "null",
      "string"
    ]
  },
  {
    "name": "networkType",
    "type": [
      "null",
      "string"
    ]
  }
],
"exampleEvent": {
  "eventType": "com.oraclecloud.databaseservice.deleteapplicationvip.end",
  "cloudEventsVersion": "0.1",
  "eventTypeVersion": "2.0",
  "source": "databaseservice",
  "contentType": "application/json",
  "eventID": "17619ca1-07ae-4e2d-a818-5b5f1fcd4f70",
  "eventTime": "2022-12-16T21:16:04.000Z",
  "data": {
    "resourceId": "ocid1.applicationvip.oc1.....unique_id",
    "resourceName": "my_application_vip",
    "tagSlug": null,
    "compartmentId": "ocid1.compartment.oc1.....unique_id",
    "request": {
      "id": "1b0d242b-b3cd-4d61-9779-2de23e0e6742",
      "path": null,
      "action": null,
      "parameters": null,
      "headers": null
    },
    "response": {
      "status": null,
        "responseTime": null,
        "headers": null,
        "payload": null,
        "message": ""
      },
      "stateChange": {
        "previous": null,
        "current": {
          "lifecycleState": "TERMINATED",
          "hostnameLabel": "my_application_vip",
          "freeTags": {},
          "definedTags": {}
        }
      },
      "eventGroupingId": "csid80b16d4d459eaaa60ad25a9829d8/
b3e19f76a81549e6b7bf1d8619f7c191/C683214FCB0BF3CEC1C8B23C2FEE983E",
      "eventName": "DeleteApplicationVip",
      "availabilityDomain": "",
      "resourceVersion": null,
      "additionalDetails": {
        "id": "ocid1.applicationvip.oc1.....unique_id",
        "freeformTags": {},
        "definedTags": {},
        "timeCreated": "2022-12-15T21:17:59.000Z",
        "timeUpdated": "2022-12-15T21:18:04.389Z",
        "lifecycleState": "TERMINATED",
        "lifecycleDetails": "",
        "hostnameLabel": "my_application_vip",
        "cloudVmClusterId": "ocid1.cloudvmcluster.oc1.....unique_id",
        "compartmentId": "ocid1.compartment.oc1.....unique_id",
        "vcnIpId": "ocid1.privateip.oc1.....unique_id",
        "ipAddress": "10.0.0.0",
        "subnetId": "ocid1.subnet.oc1.....unique_id",
        "networkType": "CLIENT"
      }
    }
  },
  "timeCreated": "2022-12-15T16:31:31.979Z"
}
```

Interim Software Updates Event Types

These are the event types that Interim Software Updates in Oracle Cloud Infrastructure emit.

Friendly Name	Event Type
Oneoff Patch - Create Begin	`com.oraclecloud.databaseservice.createoneoffpatch.begin`
Oneoff Patch - Create End	`com.oraclecloud.databaseservice.createoneoffpatch.end`
Oneoff Patch - Delete Begin	`com.oraclecloud.databaseservice.deleteoneoffpatch.begin`
Oneoff Patch - Delete End	`com.oraclecloud.databaseservice.deleteoneoffpatch.end`

Friendly Name	Event Type
Oneoff Patch - Download Begin	`com.oraclecloud.databaseservice.downloadoneoffpatch.begin`
Oneoff Patch - Download End	`com.oraclecloud.databaseservice.downloadoneoffpatch.end`

Interim Software Updates Event Types Examples:

This is a reference event for This is a reference event for Oneoff Patch - Create Begin:

```
{
  "id":
"ocid1.eventschema.oc1.phx.abyhqljrsllp7rfneajgq2knxbqopwux24za7qzoe3mfj2bzfxt
nwqcxpbcq",
  "exampleEvent": {
    "cloudEventsVersion": "0.1",
    "eventID": "60600c06-d6a7-4e85-b59a-1de3e6042f57",
    "eventType": "com.oraclecloud.databaseservice.createoneoffpatch.begin",
    "source": "databaseservice",
    "eventTypeVersion": "1.0",
    "eventTime": "2020-06-27T21:16:04.000Z",
    "contentType": "application/json",
    "extensions": {
      "compartmentId": "ocid1.compartment.oc1..unique_ID"
    },
    "data": {
      "compartmentId": "ocid1.compartment.oc1..unique_ID",
      "compartmentName": "example_name",
      "resourceName": "my_oneoffpatch",
      "resourceId": "OneOffPatch-unique_ID",
      "availabilityDomain": "all",
      "freeFormTags": {},
      "definedTags": {},
      "additionalDetails": {
        "id": "ocid1.id..oc1...unique_ID",
        "lifecycleState": "AVAILABLE",
        "timeCreated": "2020-08-26T12:00:00.000Z",
        "displayName": "testDisplayName",
        "databaseVersion": "19.6.0.0",
        "patchSet": "test_patch_set"
      }
    }
  },
  "serviceName": "Database",
  "displayName": "Oneoff Patch - Create Begin",
  "eventType": "com.oraclecloud.databaseservice.createoneoffpatch.begin",
  "additionalDetails": [
    { "name": "id", "type": "string" },
    { "name": "lifecycleState", "type": "string" },
    { "name": "timeCreated", "type": "string" },
    { "name": "displayName", "type": "string" },
    { "name": "dbVersion", "type": "string" },
    { "name": "patchType", "type": "string" },
    { "name": "patchShapeFamily", "type": "string" },
    { "name": "releaseUpdate", "type": "string" }
  ],
  "timeCreated": "2020-06-26T13:31:31.979Z"
}
```

This is a reference event for Oneoff Patch - Create End:

```
{
  "id":
"ocid1.eventschema.oc1.phx.abyhqljrj4vvuph4qvj5eateeel6axblhkq3caqndgmjvwl3sld
pgb255j2q",
  "exampleEvent": {
    "cloudEventsVersion": "0.1",
    "eventID": "60600c06-d6a7-4e85-b59a-1de3e6042f57",
    "eventType": "com.oraclecloud.databaseservice.createoneoffpatch.end",
    "source": "databaseservice",
    "eventTypeVersion": "1.0",
    "eventTime": "2020-06-27T21:16:04.000Z",
    "contentType": "application/json",
    "extensions": {
      "compartmentId": "ocid1.compartment.oc1..unique_ID"
    },
    "data": {
      "compartmentId": "ocid1.compartment.oc1..unique_ID",
      "compartmentName": "example_name",
      "resourceName": "my_oneoffpatch",
      "resourceId": "OneOffPatch-unique_ID",
      "availabilityDomain": "all",
      "freeFormTags": {},
      "definedTags": {},
      "additionalDetails": {
        "id": "ocid1.id..oc1...unique_ID",
        "lifecycleState": "AVAILABLE",
        "timeCreated": "2020-08-26T12:00:00.000Z",
        "displayName": "testDisplayName",
        "databaseVersion": "19.6.0.0",
        "patchSet": "test_patch_set"
      }
    }
  },
  "serviceName": "Database",
  "displayName": "Oneoff Patch - Create End",
  "eventType": "com.oraclecloud.databaseservice.createoneoffpatch.end",
  "additionalDetails": [
    { "name": "id", "type": "string" },
    { "name": "lifecycleState", "type": "string" },
    { "name": "timeCreated", "type": "string" },
    { "name": "displayName", "type": "string" },
    { "name": "dbVersion", "type": "string" },
    { "name": "patchType", "type": "string" },
    { "name": "patchShapeFamily", "type": "string" },
    { "name": "releaseUpdate", "type": "string" }
  ],
  "timeCreated": "2020-06-26T13:31:31.979Z"
}
```

This is a reference event for Oneoff Patch - Delete Begin:

```
{
  "id":
"ocid1.eventschema.oc1.phx.abyhqljrdripga5rryplwmv4ws6hqzr3pjyl7wfvoaqutvg2ey2
vtycn5onq",
  "exampleEvent": {
    "cloudEventsVersion": "0.1",
    "eventID": "60600c06-d6a7-4e85-b59a-1de3e6042f57",
    "eventType": "com.oraclecloud.databaseservice.deleteoneoffpatch.begin",
    "source": "databaseservice",
    "eventTypeVersion": "1.0",
    "eventTime": "2020-06-27T21:16:04.000Z",
    "contentType": "application/json",
    "extensions": {
      "compartmentId": "ocid1.compartment.oc1..unique_ID"
    },
    "data": {
      "compartmentId": "ocid1.compartment.oc1..unique_ID",
      "compartmentName": "example_name",
      "resourceName": "my_oneoffpatch",
      "resourceId": "OneOffPatch-unique_ID",
      "availabilityDomain": "all",
      "freeFormTags": {},
      "definedTags": {},
      "additionalDetails": {
        "id": "ocid1.id..oc1...unique_ID",
        "lifecycleState": "AVAILABLE",
        "timeCreated": "2020-08-26T12:00:00.000Z",
        "displayName": "testDisplayName",
        "databaseVersion": "19.6.0.0",
        "patchSet": "test_patch_set"
      }
    }
  },
  "serviceName": "Database",
  "displayName": "Oneoff Patch - Delete Begin",
  "eventType": "com.oraclecloud.databaseservice.deleteoneoffpatch.begin",
  "additionalDetails": [
    { "name": "id", "type": "string" },
    { "name": "lifecycleState", "type": "string" },
    { "name": "timeCreated", "type": "string" },
    { "name": "displayName", "type": "string" },
    { "name": "dbVersion", "type": "string" },
    { "name": "patchType", "type": "string" },
    { "name": "patchShapeFamily", "type": "string" },
    { "name": "releaseUpdate", "type": "string" }
  ],
  "timeCreated": "2020-06-26T13:31:31.979Z"
}
```

This is a reference event for Oneoff Patch - Delete End:

```
{
  "id":
```

```
"ocid1.eventschema.oc1.phx.abyhqljrgwk2gvx5lmx6fiwotgdy32mdmrnkyzznz37dpb4mmeh
gzt37vl7a",
  "exampleEvent": {
    "cloudEventsVersion": "0.1",
    "eventID": "60600c06-d6a7-4e85-b59a-1de3e6042f57",
    "eventType": "com.oraclecloud.databaseservice.deleteoneoffpatch.end",
    "source": "databaseservice",
    "eventTypeVersion": "1.0",
    "eventTime": "2020-06-27T21:16:04.000Z",
    "contentType": "application/json",
    "extensions": {
      "compartmentId": "ocid1.compartment.oc1..unique_ID"
    },
    "data": {
      "compartmentId": "ocid1.compartment.oc1..unique_ID",
      "compartmentName": "example_name",
      "resourceName": "my_oneoffpatch",
      "resourceId": "OneOffPatch-unique_ID",
      "availabilityDomain": "all",
      "freeFormTags": {},
      "definedTags": {},
      "additionalDetails": {
        "id": "ocid1.id..oc1...unique_ID",
        "lifecycleState": "AVAILABLE",
        "timeCreated": "2020-08-26T12:00:00.000Z",
        "displayName": "testDisplayName",
        "databaseVersion": "19.6.0.0",
        "patchSet": "test_patch_set"
      }
    }
  },
  "serviceName": "Database",
  "displayName": "Oneoff Patch - Delete End",
  "eventType": "com.oraclecloud.databaseservice.deleteoneoffpatch.end",
  "additionalDetails": [
    { "name": "id", "type": "string" },
    { "name": "lifecycleState", "type": "string" },
    { "name": "timeCreated", "type": "string" },
    { "name": "displayName", "type": "string" },
    { "name": "dbVersion", "type": "string" },
    { "name": "patchType", "type": "string" },
    { "name": "patchShapeFamily", "type": "string" },
    { "name": "releaseUpdate", "type": "string" }
  ],
  "timeCreated": "2020-06-26T13:31:31.979Z"
}
```

This is a reference event for Oneoff Patch - Download Begin:

```
{
  "id":
"ocid1.eventschema.oc1.phx.abyhqljr3vkb7klt5hkbsnqzjaxmszsqomanlbqmr2tsrcq7xaf
cv2c74l2q",
  "exampleEvent": {
    "cloudEventsVersion": "0.1",
    "eventID": "60600c06-d6a7-4e85-b59a-1de3e6042f57",
    "eventType": "com.oraclecloud.databaseservice.downloadoneoffpatch.begin",
    "source": "databaseservice",
    "eventTypeVersion": "1.0",
    "eventTime": "2020-06-27T21:16:04.000Z",
    "contentType": "application/json",
    "extensions": {
      "compartmentId": "ocid1.compartment.oc1..unique_ID"
    },
    "data": {
      "compartmentId": "ocid1.compartment.oc1..unique_ID",
      "compartmentName": "example_name",
      "resourceName": "my_oneoffpatch",
      "resourceId": "OneOffPatch-unique_ID",
      "availabilityDomain": "all",
      "freeFormTags": {},
      "definedTags": {},
      "additionalDetails": {
        "id": "ocid1.id..oc1...unique_ID",
        "lifecycleState": "AVAILABLE",
        "timeCreated": "2020-08-26T12:00:00.000Z",
        "displayName": "testDisplayName",
        "databaseVersion": "19.6.0.0",
        "patchSet": "test_patch_set"
      }
    }
  },
  "serviceName": "Database",
  "displayName": "Oneoff Patch - Download Begin",
  "eventType": "com.oraclecloud.databaseservice.downloadoneoffpatch.begin",
  "additionalDetails": [
    { "name": "id", "type": "string" },
    { "name": "lifecycleState", "type": "string" },
    { "name": "timeCreated", "type": "string" },
    { "name": "displayName", "type": "string" },
    { "name": "dbVersion", "type": "string" },
    { "name": "patchType", "type": "string" },
    { "name": "patchShapeFamily", "type": "string" },
    { "name": "releaseUpdate", "type": "string" }
  ],
  "timeCreated": "2020-06-26T13:31:31.979Z"
}
```

This is a reference event for Oneoff Patch - Download End:

```
{
  "id":
"ocid1.eventschema.oc1.phx.abyhqljrn2lruez55ah56kqksi5qfg6m7igvven7o2qkahlk5tk
wrj5ll3oa",
  "exampleEvent": {
    "cloudEventsVersion": "0.1",
    "eventID": "60600c06-d6a7-4e85-b59a-1de3e6042f57",
    "eventType": "com.oraclecloud.databaseservice.downloadoneoffpatch.end",
    "source": "databaseservice",
    "eventTypeVersion": "1.0",
  "eventTime": "2020-06-27T21:16:04.000Z",
  "contentType": "application/json",
  "extensions": {
   "compartmentId": "ocid1.compartment.oc1..unique_ID"
  },
  "data": {
   "compartmentId": "ocid1.compartment.oc1..unique_ID",
   "compartmentName": "example_name",
   "resourceName": "my_oneoffpatch",
   "resourceId": "OneOffPatch-unique_ID",
   "availabilityDomain": "all",
   "freeFormTags": {},
   "definedTags": {},
   "additionalDetails": {
    "id": "ocid1.id..oc1...unique_ID",
    "lifecycleState": "AVAILABLE",
    "timeCreated": "2020-08-26T12:00:00.000Z",
    "displayName": "testDisplayName",
    "databaseVersion": "19.6.0.0",
    "patchSet": "test_patch_set"
   }
  }
 },
 "serviceName": "Database",
 "displayName": "Oneoff Patch - Download End",
 "eventType": "com.oraclecloud.databaseservice.downloadoneoffpatch.end",
 "additionalDetails": [
  { "name": "id", "type": "string" },
  { "name": "lifecycleState", "type": "string" },
  { "name": "timeCreated", "type": "string" },
  { "name": "displayName", "type": "string" },
  { "name": "dbVersion", "type": "string" },
  { "name": "patchType", "type": "string" },
  { "name": "patchShapeFamily", "type": "string" },
  { "name": "releaseUpdate", "type": "string" }
 ],
 "timeCreated": "2020-06-26T13:31:31.979Z"
}
```

Serial Console Connection Event Types

Review the list of event types that serial console connection emits.

Table 5-5 Serial Console Connection Events

Friendly Name	Event Type
DB Node Console Connection - Create Begin	com.oraclecloud.databaseservice.createdbnodeconsoleconnection.begin
DB Node Console Connection - Create End	com.oraclecloud.databaseservice.createdbnodeconsoleconnection.end
DB Node Console Connection - Delete Begin	com.oraclecloud.databaseservice.deletedbnodeconsoleconnection.begin

Table 5-5 (Cont.) Serial Console Connection Events

Friendly Name	Event Type
DB Node Console Connection - Delete End	com.oraclecloud.databaseservice.deletedbnodeconsoleconnection.end
DB Node Console Connection - Update	com.oraclecloud.databaseservice.updatedbnodeconsoleconnection
DB Node - Update	com.oraclecloud.databaseservice.updatedbnode

Example 5-68 Serial Console Connection Event Types Examples

This is a reference event for DB Node Console Connection - Create Begin:

```
"exampleEvent": {
  "cloudEventsVersion": "0.1",
  "eventID": "60600c06-d6a7-4e85-b56a-1de3e6042f57",
  "eventType":
"com.oraclecloud.databaseservice.createdbnodeconsoleconnection.begin",
  "source": "databaseservice",
  "eventTypeVersion": "1.0",
  "eventTime": "2019-08-29T21:16:04.000Z",
  "contentType": "application/json",
  "extensions": {
    "compartmentId": "ocid1.compartment.oc1..unique_ID"
  },
  "data": {
    "compartmentId": "ocid1.compartment.oc1..unique_ID",
    "resourceId": "ocid1.dbnodeconsoleconnection.oc1..unique_ID",
    "freeFormTags": {},
    "definedTags": {},
    "additionalDetails": {
      "id": "ocid1.dbnodeconsoleconnection.oc1..unique_ID",
      "lifecycleState": "CREATING",
      "timeCreated": "2019-08-29T12:00:00.000Z",
      "timeUpdated": "2019-08-29T12:30:00.000Z",
      "lifecycleDetails": "detail message",
      "dbnodeId": "ocid1.dbnode.oc1..unique_ID",
      "tenantId": "ocid1.tenant.oc1..unique_ID",
      "compartmentId": "ocid1.compartment.oc1..unique_ID"
    }
  }
}
```

This is a reference event for DB Node Console Connection - Create End:

```
"exampleEvent": {
  "cloudEventsVersion": "0.1",
  "eventID": "60600c06-d6a7-4e85-b56a-1de3e6042f57",
  "eventType":
"com.oraclecloud.databaseservice.createdbnodeconsoleconnection.end",
  "source": "databaseservice",
  "eventTypeVersion": "1.0",
  "eventTime": "2019-08-29T21:16:04.000Z",
  "contentType": "application/json",
  "extensions": {
    "compartmentId": "ocid1.compartment.oc1..unique_ID"
  },
  "data": {
    "compartmentId": "ocid1.compartment.oc1..unique_ID",
    "resourceId": "ocid1.dbnodeconsoleconnection.oc1..unique_ID",
    "freeFormTags": {},
    "definedTags": {},
    "additionalDetails": {
      "id": "ocid1.dbnodeconsoleconnection.oc1..unique_ID",
      "lifecycleState": "ACTIVE",
      "timeCreated": "2019-08-29T12:00:00.000Z",
      "timeUpdated": "2019-08-29T12:30:00.000Z",
      "lifecycleDetails": "detail message",
      "dbnodeId": "ocid1.dbnode.oc1..unique_ID",
      "tenantId": "ocid1.tenant.oc1..unique_ID",
      "compartmentId": "ocid1.compartment.oc1..unique_ID"
    }
  }
}
```

This is a reference event for DB Node Console Connection - Delete Begin:

```
"exampleEvent": {
  "cloudEventsVersion": "0.1",
  "eventID": "60600c06-d6a7-4e85-b56a-1de3e6042f57",
  "eventType":
"com.oraclecloud.databaseservice.deletedbnodeconsoleconnection.begin",
  "source": "databaseservice",
  "eventTypeVersion": "1.0",
  "eventTime": "2019-08-29T21:16:04.000Z",
  "contentType": "application/json",
  "extensions": {
    "compartmentId": "ocid1.compartment.oc1..unique_ID"
  },
  "data": {
    "compartmentId": "ocid1.compartment.oc1..unique_ID",
    "resourceId": "ocid1.dbnodeconsoleconnection.oc1..unique_ID",
    "freeFormTags": {},
    "definedTags": {},
    "additionalDetails": {
      "id": "ocid1.dbnodeconsoleconnection.oc1..unique_ID",
      "lifecycleState": "DELETING",
      "timeCreated": "2019-08-29T12:00:00.000Z",
      "timeUpdated": "2019-08-29T12:30:00.000Z",
      "lifecycleDetails": "detail message",
      "dbnodeId": "ocid1.dbnode.oc1..unique_ID",
      "tenantId": "ocid1.tenant.oc1..unique_ID",
      "compartmentId": "ocid1.compartment.oc1..unique_ID"
    }
  }
}
```

This is a reference event for DB Node Console Connection - Delete End:

```
"exampleEvent": {
  "cloudEventsVersion": "0.1",
  "eventID": "60600c06-d6a7-4e85-b56a-1de3e6042f57",
  "eventType":
"com.oraclecloud.databaseservice.deletedbnodeconsoleconnection.end",
  "source": "databaseservice",
  "eventTypeVersion": "1.0",
  "eventTime": "2019-08-29T21:16:04.000Z",
  "contentType": "application/json",
  "extensions": {
    "compartmentId": "ocid1.compartment.oc1..unique_ID"
  },
  "data": {
    "compartmentId": "ocid1.compartment.oc1..unique_ID",
    "resourceId": "ocid1.dbnodeconsoleconnection.oc1..unique_ID",
    "freeFormTags": {},
    "definedTags": {},
    "additionalDetails": {
      "id": "ocid1.dbnodeconsoleconnection.oc1..unique_ID",
      "lifecycleState": "DELETED",
      "timeCreated": "2019-08-29T12:00:00.000Z",
      "timeUpdated": "2019-08-29T12:30:00.000Z",
      "lifecycleDetails": "detail message",
      "dbnodeId": "ocid1.dbnode.oc1..unique_ID",
      "tenantId": "ocid1.tenant.oc1..unique_ID",
      "compartmentId": "ocid1.compartment.oc1..unique_ID"
    }
  }
}
```

This is a reference event for DB Node Console Connection - Update:

```
"exampleEvent": {
  "cloudEventsVersion": "0.1",
  "eventID": "60600c06-d6a7-4e85-b56a-1de3e6042f57",
  "eventType":
"com.oraclecloud.databaseservice.updatedbnodeconsoleconnection",
  "source": "databaseservice",
  "eventTypeVersion": "1.0",
  "eventTime": "2019-08-29T21:16:04.000Z",
  "contentType": "application/json",
  "extensions": {
    "compartmentId": "ocid1.compartment.oc1..unique_ID"
  },
  "data": {
    "compartmentId": "ocid1.compartment.oc1..unique_ID",
    "resourceId": "ocid1.dbnodeconsoleconnection.oc1..unique_ID",
    "freeFormTags": {},
    "definedTags": {},
    "additionalDetails": {
      "id": "ocid1.dbnodeconsoleconnection.oc1..unique_ID",
      "lifecycleState": "ACTIVE",
      "timeCreated": "2019-08-29T12:00:00.000Z",
      "timeUpdated": "2019-08-29T12:30:00.000Z",
      "lifecycleDetails": "detail message",
      "dbnodeId": "ocid1.dbnode.oc1..unique_ID",
      "tenantId": "ocid1.tenant.oc1..unique_ID",
      "compartmentId": "ocid1.compartment.oc1..unique_ID"
    }
  }
}
```

This is a reference event for DB Node - Update:

```
"exampleEvent": {
  "cloudEventsVersion": "0.1",
  "eventID": "60600c06-d6a7-4e85-b56a-1de3e6042f57",
  "eventType": "com.oraclecloud.databaseservice.updatedbnode",
  "source": "databaseservice",
  "eventTypeVersion": "1.0",
  "eventTime": "2019-06-27T21:16:04.000Z",
  "contentType": "application/json",
  "extensions": {
    "compartmentId": "ocid1.compartment.oc1..unique_ID"
  },
  "data": {
    "compartmentId": "ocid1.compartment.oc1..unique_ID",
    "compartmentName": "example_name",
    "resourceName": "my_dbnode",
    "resourceId": "DbNode-unique_ID",
    "availabilityDomain": "all",
    "freeFormTags": {},
    "definedTags": {},
    "additionalDetails": {
      "id": "ocid1.id..oc1...unique_ID",
      "lifecycleState": "AVAILABLE",
      "timeCreated": "2019-08-26T12:00:00.000Z",
      "timeUpdated": "2019-08-26T12:30:00.000Z",
      "dbSystemId": "ocid1.dbsystem.oc1.phx.unique_ID",
      "lifecycleDetails": "detail message",
      "vmClusterId": "VmCluster-unique_ID",
      "dbHostId": "dbHost-unique_ID",
      "nodeNumber": 2,
      "powerAction": "HardReset",
      "hostName": "testHostName"
    }
  }
}
```

- Viewing Audit Log Events
  Oracle Cloud Infrastructure Audit service provides records of API operations performed against supported services as a list of log events.

Viewing Audit Log Events

Oracle Cloud Infrastructure Audit service provides records of API operations performed against supported services as a list of log events.

An audit event is generated when you connect to the serial console using a Secure Shell (SSH) connection. Navigate to Audit in the Console and search for `VmConsoleConnected`. When you navigate to Audit in the Console, a list of results is generated for the current compartment. Audit logs are organized by compartment, so if you are looking for a particular event, you must know which compartment the event occurred in. You can filter the list in the following ways:

- Date and time
- Request Action Types (operations)
- Keywords

For more information, see *Viewing Audit Log Events*.

Example 5-69   Serial Console Connection Audit Event Example

This is a reference event for Serial Console Connection:

```
{

  "eventType": "VmConsoleConnected",
  "cloudEventsVersion": "0.1",
  "eventTypeVersion": "2.0",
  "source": "VmConsoleConnectionAPI",
  "eventId": "2367d627-cff8-11ed-bfd3-02001714f979",
  "eventTime": "2023-03-31T19:13:37.120Z",
  "contentType": "application/json",

  "data": {
    "eventGroupingId": "2367d62d-cff8-11ed-bfd3-02001714f979",
    "eventName": "VmConsoleConnected",
    "compartmentId": "ocid1.compartment.oc1..<TRUNCATED>aaaaxxxxx",
    "compartmentName": "exacc-dev",
    "resourceName": "",
    "resourceId":
"ocid1.dbnodeconsoleconnection.oc1.iad.<TRUNCATED>aaaaaaxxxxx",
    "availabilityDomain": null,
    "freeformTags": null,
    "definedTags": null,

    "identity": {
      "principalName": "dsaes",
      "principalId": "ocid1.user.oc1..<TRUNCATED>aaaaaaaaxxxxxxxxxx",
      "authType": "Native",
      "callerName": null,
      "callerId": null,
      "tenantId": "ocid1.tenancy.oc1..<TRUNCATED>aaaaaaxxxxx",
      "ipAddress": null,
      "credentials": null,
      "userAgent": null,
      "consoleSessionId": null
  },

  "request": {
   "id": "",
   "path": "",
   "action": "",
   "parameters": null,
   "headers": null
 },

  "response": {
   "status": "",
   "responseTime": "0001-01-01T00:00:00.000Z",
   "headers": null,
   "payload": null,
   "message": ""
  },

  "stateChange": null,

  "additionalDetails": {
   "DBNodeId": "ocid1.dbnode.oc1.iad.<TRUNCATED>aaaaaxxxxxxx"
  }
 }
}
```

Related Topics

- Overview of Audit
- Viewing Audit Log Events
- Setting Audit Log Retention Period

Monitor Metrics to Diagnose and Troubleshoot Problems with Pluggable Databases

Enable Database Management service to view metrics to diagnose and troubleshoot problems with pluggable databases.

- About Database Management
- Using the Console to Enable Database Management for a Container Database (CDB)
  To enable Database Management for a container database (CDB), use this procedure.
- Using the Console to Enable Database Management for a Pluggable Database (PDB)
  To enable Database Management for a pluggable database (PDB), use this procedure.
- Using the Console to Edit Database Management for a Pluggable Database (PDB)
  To edit the Database Management configuration for a pluggable database (PDB), use this procedure.
- Using the Console to Disable Database Management for a Pluggable Database (PDB)
  To disable Database Management for a pluggable database (PDB), use this procedure.
- Using the Console to View Performance Hub for a Container Database (CDB)
  To view Performance Hub for a container database (CDB), use this procedure. You must first enable Database Management to view the performance report.

- Using the Console to View Performance Hub for a Pluggable Database (PDB)
  To view Performance Hub for a pluggable database (PDB), use this procedure. You must first enable Database Management to view the performance report.
- Using the API to Enable, Disable, or Update Database Management Service
- Oracle Cloud Database Metrics
  Use the metrics to diagnose and troubleshoot issues.

About Database Management

As a Database Administrator, you can use the Oracle Cloud Infrastructure Database Management service to monitor and manage Oracle Databases. For more information, see *About Database Management*.

Performance Hub provides a visual representation of diagnostic data that you can leverage to fix performance issues or tune the database to improve performance. For more information about Performance Hub, see *Performance Hub*.

Related Topics

- About Database Management
- Performance Hub

Using the Console to Enable Database Management for a Container Database (CDB)

To enable Database Management for a container database (CDB), use this procedure.

> **Note:**
>
> You can also enable Database Management for a database from the Database Management Administration page. For more information, see *Enable Database Management for Oracle Cloud Databases*.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata on Oracle Public Cloud**.
2. Choose your **Compartment**.

   A list of Exadata VM Clusters is displayed.
3. In the list of Exadata VM Clusters, click the Exadata VM Cluster that contains the database for which you want to enable Database Management.

   Exadata VM Cluster Details page is displayed.

   Under **Resources**, **Databases** is selected by default.
4. In the list of databases, click the database for which you want to enable Database Management.

   Database Details page is displayed.
5. In the **Database Information** section, under the **Associated Services**, check the status of Database Management.

   If the Database Management is displayed as **Not Enabled**, perform the following steps:

Enable Database Management

1. Click **Enable**.

   Enable Database Management window is displayed.

2. In the **Database information** section, provide the following details:

   - **Database type**: Read-only. Type of the database.
   - **Exadata VM Cluster**: Read-only. Compartment in which the database is located.
   - **Database home**: Read-only. Database home of the database.
   - **Database name**: Read-only. Name of the database.
   - **Service name**: The unique service name of the database. A default unique name is displayed, which can be changed if required.
   - **Protocol**: Select either TCP or TCPS to connect to the Oracle Cloud Database. TCP is selected by default.

     > **Note:**
     >
     > – If Oracle Data Guard is enabled after Database Management was enabled for an Exadata VM Cluster using the TCPS protocol, then TCPS will have to be reconfigured. Enabling Oracle Data Guard is causing TCPS configuration to be overwritten, and it's recommended that TCPS is configured on an Exadata VM Cluster after enabling Oracle Data Guard.
     >
     > – Database Management currently does not support Oracle Data Guard configuration and Database Management features are not available for standby databases.

   - **Port**: Specify the port number.
     If TCP is selected in the **Protocol** field, then the port number `1521` is displayed by default. You can change it if required. You can select the port number from a range of 1 to 65535.
   - **Database wallet secret**: This field is only displayed if TCPS is selected in the **Protocol** field.

     a. Select the secret that contains the database wallet from the drop-down list. If an existing database wallet secret is not available, then select **Create new secret...** from the drop-down list.
     The Create database wallet secret panel is displayed and you can create a new secret.

     For information on database wallets and creating a secret in the Vault service, see *Oracle Cloud Database-related Prerequisite Tasks*.

     b. If the Database Management (`dpd`) service policy that grants Database Management permission to read the secret that contains the database wallet is not created, then the `System policies are required...` message is displayed. You can click **Add policy** to view and automatically create the service policy.
     For information on Vault service permissions required to use existing secrets or create new secrets, see *Permissions Required to Enable Database Management for Oracle Cloud Databases*.

3. In the **Specify credentials for the connection** section, provide the following details:
   - **Database user name**: Enter the database user name.
   - **Database user password secret**:
     a. Select the secret that contains the database user password from the drop-down list. If the compartment in which the secret resides is different from the compartment displayed, then click **Change compartment** and select another compartment. If an existing secret with the database user password is not available, then select **Create new secret...** from the drop-down list.
        The Create password secret panel is displayed and you can create a new secret.

        For information on database monitoring user credentials and saving the database user password as a secret in the Vault service, see *Oracle Cloud Database-related Prerequisite Tasks*.
     b. If the Database Management (`dpd`) service policy that grants Database Management permission to read the secret that contains the database wallet is not created, then the `System policies are required...` message is displayed. You can click **Add policy** to view and automatically create the service policy.
        For information on Vault service permissions required to use existing secrets or create new secrets, see *Permissions Required to Enable Database Management for Oracle Cloud Databases*.
4. In the **Private endpoint information** section, select the private endpoint that will act as a representation of Database Management in the VCN in which the Oracle Cloud Database can be accessed.

   You can choose the private endpoint from a different compartment as well. You must ensure that the appropriate Database Management private endpoint is available.

   Here are the two types of Database Management private endpoints:
   - Private endpoint for single instance Databases in the bare metal and virtual machine DB systems.
   - Private endpoint for Oracle RAC Databases in the virtual machine DB system.

   If a Database Management private endpoint is not available, then you must create one.

   For information on how to create a private endpoint, see *Create a Database Management Private Endpoint*.
5. In the **Management options** section, choose between the following options:
   - **Full management**: This includes fleet management, advanced Performance Hub, and other SKU features along with basic management capabilities.
   - **Basic management**: This includes basic monitoring metrics and the ASH Analytics and SQL Monitoring features in Performance Hub for container databases.
     For more information on management options, see *About Management Options*.
6. Click **Enable Database Management**.
7. A confirmation message with a link to the Oracle Cloud Database's **Work requests** section on the **Database information** page is displayed. Click the link to monitor the progress of the work request.
8. In the **Database Information** section, under the **Associated Services**, verify if the status of **Database Management** is **Enabled**.

   The **Disable** option is also displayed, which you can click to disable Database Management.

If you encounter issues when enabling Database Management, see *Issues Encountered When Enabling Database Management for Oracle Cloud Databases* for likely causes and solutions.

Related Topics

- Permissions Required to Enable Database Management for Oracle Cloud Databases
- Oracle Cloud Database-related Prerequisite Tasks
- Enable Database Management for Oracle Cloud Databases
- Issues Encountered When Enabling Database Management for Oracle Cloud Databases

Using the Console to Enable Database Management for a Pluggable Database (PDB)

To enable Database Management for a pluggable database (PDB), use this procedure.

> **Note:**
>
> You can also enable Database Management for a database from the Database Management Administration page. For more information, see *Enable Database Management for Oracle Cloud Databases*.

Prerequisite

To enable the Database Management for a pluggable database, enable Database Management for the associated database with the **Full Management** option.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata on Oracle Public Cloud**.
2. Choose your **Compartment**.

   A list of Exadata VM Clusters is displayed.
3. In the list of Exadata VM Clusters, click the Exadata VM Cluster that contains the pluggable database for which you want to enable Database Management.

   Exadata VM Cluster Details page is displayed.

   Under **Resources**, **Databases** is selected by default.
4. In the list of databases, click the database that contains the pluggable database for which you want to enable Database Management.

   Database Details page is displayed.
5. Under **Resources**, click **Pluggable Databases**.
6. In the list of pluggable databases, click the pluggable database for which you want to enable Database Management.

   Pluggable Database Details page is displayed.
7. In the **Database Information** section, under the **Associated Services**, check the status of Database Management.

   If the Database Management is displayed as **Not Enabled**, perform the following steps:

Enable Database Management

1. Click **Enable**.

   Enable Database Management window is displayed.

2. In the **Database information** section, provide the following details:

   - **Database type**: Read-only. Type of the database.
   - **Exadata VM Cluster**: Read-only. Compartment in which the database is located.
   - **Database home**: Read-only. Database home of the database.
   - **Pluggable Database name**: Read-only. Name of the database.
   - **Service name**: The unique service name of the database. A default unique name is displayed, which can be changed if required.
   - **Protocol**: Select either TCP or TCPS to connect to the Oracle Cloud Database. TCP is selected by default.

     > **Note:**
     >
     > – If Oracle Data Guard is enabled after Database Management was enabled for an Exadata VM Cluster using the TCPS protocol, then TCPS will have to be reconfigured. Enabling Oracle Data Guard is causing TCPS configuration to be overwritten, and it's recommended that TCPS is configured on an Exadata VM Cluster after enabling Oracle Data Guard.
     >
     > – Database Management currently does not support Oracle Data Guard configuration and Database Management features are not available for standby databases.

   - **Port**: Specify the port number.
     If TCP is selected in the **Protocol** field, then the port number `1521` is displayed by default. You can change it if required. You can select the port number from a range of 1 to 65535.
   - **Database wallet secret**: This field is only displayed if TCPS is selected in the **Protocol** field.

     a. Select the secret that contains the database wallet from the drop-down list. If an existing database wallet secret is not available, then select **Create new secret...** from the drop-down list.
     The Create database wallet secret panel is displayed and you can create a new secret.

     For information on database wallets and creating a secret in the Vault service, see *Oracle Cloud Database-related Prerequisite Tasks*.

     b. If the Database Management (`dpd`) service policy that grants Database Management permission to read the secret that contains the database wallet is not created, then the `System policies are required...` message is displayed. You can click **Add policy** to view and automatically create the service policy.
     For information on Vault service permissions required to use existing secrets or create new secrets, see *Permissions Required to Enable Database Management for Oracle Cloud Databases*.

3. In the **Specify credentials for the connection** section, provide the following details:
   - **Database user name**: Enter the database user name.
   - **Database user password secret**:
     a. Select the secret that contains the database user password from the drop-down list. If the compartment in which the secret resides is different from the compartment displayed, then click **Change compartment** and select another compartment. If an existing secret with the database user password is not available, then select **Create new secret...** from the drop-down list.
     The Create password secret panel is displayed and you can create a new secret.

     For information on database monitoring user credentials and saving the database user password as a secret in the Vault service, see *Oracle Cloud Database-related Prerequisite Tasks*.

     b. If the Database Management (`dpd`) service policy that grants Database Management permission to read the secret that contains the database wallet is not created, then the `System policies are required...` message is displayed. You can click **Add policy** to view and automatically create the service policy.
     For information on Vault service permissions required to use existing secrets or create new secrets, see *Permissions Required to Enable Database Management for Oracle Cloud Databases*.
4. In the **Private endpoint information** section, select the private endpoint that will act as a representation of Database Management in the VCN in which the Oracle Cloud Database can be accessed.

   You can choose the private endpoint from a different compartment as well. You must ensure that the appropriate Database Management private endpoint is available.

   Here are the two types of Database Management private endpoints:
   - Private endpoint for single instance Databases in the bare metal and virtual machine DB systems.
   - Private endpoint for Oracle RAC Databases in the virtual machine DB system.

   If a Database Management private endpoint is not available, then you must create one.

   For information on how to create a private endpoint, see *Create a Database Management Private Endpoint*.
5. In the **Management options** section, choose between the following options:
   - **Full management**: This includes fleet management, advanced Performance Hub, and other SKU features along with basic management capabilities.
   - **Basic management**: This includes basic monitoring metrics and the ASH Analytics and SQL Monitoring features in Performance Hub for container databases.
     For more information on management options, see *About Management Options*.
6. Click **Enable Database Management**.
7. A confirmation message with a link to the Oracle Cloud Database's **Work requests** section on the **Database information** page is displayed. Click the link to monitor the progress of the work request.
8. In the **Database Information** section, under the **Associated Services**, verify if the status of **Database Management** is **Enabled**.

   The **Disable** option is also displayed, which you can click to disable Database Management.

If you encounter issues when enabling Database Management, see *Issues Encountered When Enabling Database Management for Oracle Cloud Databases* for likely causes and solutions.

Related Topics

- Permissions Required to Enable Database Management for Oracle Cloud Databases
- Oracle Cloud Database-related Prerequisite Tasks
- Enable Database Management for Oracle Cloud Databases
- Issues Encountered When Enabling Database Management for Oracle Cloud Databases

Using the Console to Edit Database Management for a Pluggable Database (PDB)

To edit the Database Management configuration for a pluggable database (PDB), use this procedure.

> **Note:**
>
> You can also enable Database Management for a database from the Database Management Administration page. For more information, see *Enable Database Management for Oracle Cloud Databases*.

Prerequisite

To enable the Database Management for a pluggable database, enable Database Management for the associated database with the **Full Management** option.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata on Oracle Public Cloud**.
2. Choose your **Compartment**.

   A list of Exadata VM Clusters is displayed.
3. In the list of Exadata VM Clusters, click the Exadata VM Cluster that contains the pluggable database for which you want to edit Database Management.

   Exadata VM Cluster Details page is displayed.

   Under **Resources**, **Databases** is selected by default.
4. In the list of databases, click the database that contains the pluggable database for which you want to edit Database Management.

   Database Details page is displayed.
5. Under **Resources**, click **Pluggable Databases**.
6. In the list of pluggable databases, click the pluggable database for which you want to edit Database Management.

   Pluggable Database Details page is displayed.
7. In the **Database Information** section, under the **Associated Services**, check the status of Database Management.

   If the Database Management is displayed as **Enabled**, perform the following steps to edit Database Management:

Edit Database Management

1. Click **Enable**.

   Edit Database Management window is displayed.

2. In the **Database information** section, provide the following details:

   - **Database type**: Read-only. Type of the database.
   - **Exadata VM Cluster**: Read-only. Compartment in which the database is located.
   - **Database home**: Read-only. Database home of the database.
   - **Pluggable Database name**: Read-only. Name of the database.
   - **Service name**: The unique service name of the database. A default unique name is displayed, which can be changed if required.
   - **Protocol**: Select either TCP or TCPS to connect to the Oracle Cloud Database. TCP is selected by default.

     > **Note:**
     >
     > – If Oracle Data Guard is enabled after Database Management was enabled for an Exadata VM Cluster using the TCPS protocol, then TCPS will have to be reconfigured. Enabling Oracle Data Guard is causing TCPS configuration to be overwritten, and it's recommended that TCPS is configured on an Exadata VM Cluster after enabling Oracle Data Guard.
     >
     > – Database Management currently does not support Oracle Data Guard configuration and Database Management features are not available for standby databases.

   - **Port**: Specify the port number.
     If TCP is selected in the **Protocol** field, then the port number `1521` is displayed by default. You can change it if required. You can select the port number from a range of 1 to 65535.
   - **Database wallet secret**: This field is only displayed if TCPS is selected in the **Protocol** field.

     a. Select the secret that contains the database wallet from the drop-down list. If an existing database wallet secret is not available, then select **Create new secret...** from the drop-down list.
        The Create database wallet secret panel is displayed and you can create a new secret.

        For information on database wallets and creating a secret in the Vault service, see *Oracle Cloud Database-related Prerequisite Tasks*.

     b. If the Database Management (`dpd`) service policy that grants Database Management permission to read the secret that contains the database wallet is not created, then the `System policies are required...` message is displayed. You can click **Add policy** to view and automatically create the service policy.
        For information on Vault service permissions required to use existing secrets or create new secrets, see *Permissions Required to Enable Database Management for Oracle Cloud Databases*.

3. In the **Specify credentials for the connection** section, provide the following details:
   - **Database user name**: Enter the database user name.
   - **Database user password secret**:
     a. Select the secret that contains the database user password from the drop-down list. If the compartment in which the secret resides is different from the compartment displayed, then click **Change compartment** and select another compartment. If an existing secret with the database user password is not available, then select **Create new secret...** from the drop-down list.
        The Create password secret panel is displayed and you can create a new secret.

        For information on database monitoring user credentials and saving the database user password as a secret in the Vault service, see *Oracle Cloud Database-related Prerequisite Tasks*.
     b. If the Database Management (`dpd`) service policy that grants Database Management permission to read the secret that contains the database wallet is not created, then the `System policies are required...` message is displayed. You can click **Add policy** to view and automatically create the service policy.
        For information on Vault service permissions required to use existing secrets or create new secrets, see *Permissions Required to Enable Database Management for Oracle Cloud Databases*.
4. In the **Private endpoint information** section, select the private endpoint that will act as a representation of Database Management in the VCN in which the Oracle Cloud Database can be accessed.

   You can choose the private endpoint from a different compartment as well. You must ensure that the appropriate Database Management private endpoint is available.

   Here are the two types of Database Management private endpoints:
   - Private endpoint for single instance Databases in the bare metal and virtual machine DB systems.
   - Private endpoint for Oracle RAC Databases in the virtual machine DB system.

   If a Database Management private endpoint is not available, then you must create one.

   For information on how to create a private endpoint, see *Create a Database Management Private Endpoint*.
5. In the **Management options** section, choose between the following options:
   - **Full management**: This includes fleet management, advanced Performance Hub, and other SKU features along with basic management capabilities.
   - **Basic management**: This includes basic monitoring metrics and the ASH Analytics and SQL Monitoring features in Performance Hub for container databases.
     For more information on management options, see *About Management Options*.
6. Click **Enable Database Management**.
7. A confirmation message with a link to the Oracle Cloud Database's **Work requests** section on the **Database information** page is displayed. Click the link to monitor the progress of the work request.
8. In the **Database Information** section, under the **Associated Services**, verify if the status of **Database Management** is **Enabled**.

   The **Disable** option is also displayed, which you can click to disable Database Management.

If you encounter issues when enabling Database Management, see *Issues Encountered When Enabling Database Management for Oracle Cloud Databases* for likely causes and solutions.

Related Topics

- Permissions Required to Enable Database Management for Oracle Cloud Databases
- Oracle Cloud Database-related Prerequisite Tasks
- Enable Database Management for Oracle Cloud Databases
- Issues Encountered When Enabling Database Management for Oracle Cloud Databases

Using the Console to Disable Database Management for a Pluggable Database (PDB)

To disable Database Management for a pluggable database (PDB), use this procedure.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata on Oracle Public Cloud**.
2. Choose your **Compartment**.

   A list of Exadata VM Clusters is displayed.
3. In the list of Exadata VM Clusters, click the Exadata VM Cluster that contains the pluggable database for which you want to disable Database Management.

   Exadata VM Cluster Details page is displayed.

   Under **Resources**, **Databases** is selected by default.
4. In the list of databases, click the database that contains the pluggable database for which you want to disable Database Management.

   Database Details page is displayed.
5. Under **Resources**, click **Pluggable Databases**.
6. In the list of pluggable databases, click the pluggable database for which you want to disable Database Management.

   Pluggable Database Details page is displayed.
7. In the **Database Information** section, under the **Associated Services**, check the status of Database Management.
8. If the Database Management is displayed as **Enabled**, perform the following steps to disable Database Management:
   a. Click **Disable**.
   b. A confirmation message with a link to the **Work requests** section on the **Database information** page is displayed. Click the link to monitor the progress of the work request.
   c. In the **Database Information** section, under the **Associated Services**, verify if the status of Database Management is **Disabled**.

Using the Console to View Performance Hub for a Container Database (CDB)

To view Performance Hub for a container database (CDB), use this procedure. You must first enable Database Management to view the performance report.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata on Oracle Public Cloud**.
2. Choose your **Compartment**.

   A list of Exadata VM Clusters is displayed.
3. In the list of Exadata VM Clusters, click the Exadata VM Cluster that contains the database for which you want to view Performance Hub.

   Exadata VM Cluster Details page is displayed.

   Under **Resources**, **Databases** is selected by default.
4. In the list of databases, click the database for which you want to view Performance Hub.

   Database Details page is displayed.
5. Click **Performance Hub**.

With Basic Management, Performance Hub provides **ASH Analytics** and **SQL Monitoring**. Advanced Management will additionally provide **ADDM**, **Workload**, and **Blocking Sessions**.

Performance Hub allows you to download reports for your managed databases. For more information about downloading reports, see *Automatic Workload Repository (AWR) Report*, *Active Sessions History (ASH) Report*, and *Performance Hub Report*.

Related Topics

- Automatic Workload Repository (AWR) Report
- Active Sessions History (ASH) Report
- Performance Hub Report

Using the Console to View Performance Hub for a Pluggable Database (PDB)

To view Performance Hub for a pluggable database (PDB), use this procedure. You must first enable Database Management to view the performance report.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata on Oracle Public Cloud**.
2. Choose your **Compartment**.

   A list of Exadata VM Clusters is displayed.
3. In the list of Exadata VM Clusters, click the Exadata VM Cluster that contains the pluggable database for which you want to view Performance Hub.

   Exadata VM Cluster Details page is displayed.

   Under **Resources**, **Databases** is selected by default.
4. In the list of databases, click the database that contains the pluggable database.

   Database Details page is displayed.
5. Under **Resources**, click **Pluggable Databases**.
6. In the list of pluggable databases, click the pluggable database that you're interested in.

   Pluggable Database Details page is displayed.
7. Click **Performance Hub**.

With Basic Management, Performance Hub provides **ASH Analytics** and **SQL Monitoring**. Advanced Management will additionally provide **ADDM**, **Workload**, and **Blocking Sessions**.

Performance Hub allows you to download reports for your managed databases. For more information about downloading reports, see *Automatic Workload Repository (AWR) Report*, *Active Sessions History (ASH) Report*, and *Performance Hub Report*.

Related Topics

- Automatic Workload Repository (AWR) Report
- Active Sessions History (ASH) Report
- Performance Hub Report

Using the API to Enable, Disable, or Update Database Management Service

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to configure the Database Management service.

- Enable Database Management service for an Oracle Database located in Oracle Cloud Infrastructure to access tools including Metrics and Performance hub: `enableDatabaseManagement`
- Disable Database Management service: `disableDatabaseManagement`
- Update Database Management configuration: `updateDatabaseManagement`

Oracle Cloud Database Metrics

Use the metrics to diagnose and troubleshoot issues.

The metrics for Oracle Cloud Databases help measure useful quantitative data, such as CPU and storage utilization, the number of successful and failed database logon and connection attempts, database operations, SQL queries, transactions, and so on.

For more information, see *Oracle Cloud Database Metrics*.

- Using the Console View Metrics for a Container Database (CDB)
  To view metrics for a container database (CDB), you must first enable Database Management with the **Full Management** option.
- Using the Console to View Metrics for a Pluggable Database (PDB)
  To view metrics for a Pluggable Database (PDB), the following prerequisites must be met:

Related Topics

- Oracle Cloud Database Metrics

Using the Console View Metrics for a Container Database (CDB)

To view metrics for a container database (CDB), you must first enable Database Management with the **Full Management** option.

To enable Database Management for databases, see *Using the Console to Enable Database Management for a Database*.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata on Oracle Public Cloud**.

2. Choose your **Compartment**.

   A list of Exadata VM Clusters is displayed.

3. In the list of Exadata VM Clusters, click the Exadata VM Cluster that contains the database for which you want to view the metrics.

   Exadata VM Cluster Details page is displayed.

   Under **Resources**, **Databases** is selected by default.

4. In the list of databases, click the database for which you want to view the metrics.

   Database Details page is displayed.

5. Under **Resources**, click **Metrics**.

Related Topics

- Using the Console to Enable Database Management for a Container Database (CDB)
  To enable Database Management for a container database (CDB), use this procedure.

Using the Console to View Metrics for a Pluggable Database (PDB)

To view metrics for a Pluggable Database (PDB), the following prerequisites must be met:

- Enable Database Management for databases with the **Full Management** option.
- Enable Database Management for pluggable databases.

1. Open the navigation menu. Click **Oracle Database**, then click **Exadata on Oracle Public Cloud**.

2. Choose your **Compartment**.

   A list of Exadata VM Clusters is displayed.

3. In the list of Exadata VM Clusters, click the Exadata VM Cluster that contains the pluggable database for which you want to view the metrics.

   Exadata VM Cluster Details page is displayed.

   Under **Resources**, **Databases** is selected by default.

4. In the list of databases, click the database that contains the pluggable database.

   Database Details page is displayed.

5. Under **Resources**, click **Pluggable Databases**.

6. In the list of pluggable databases, click the pluggable database for which you want to view the metrics.

   Pluggable Database Details page is displayed.

7. Under **Resources**, click **Metrics**.

8. Select a namespace from the **Metric namespace** from where you wish to view metrics.

> **Note:**
>
> - When Database Management is enabled, then you can view metrics only from the `oracle_oci_database` namespace.
> - When Database Management is disabled, then a banner, "`Database management must be enabled to provide data for metrics.`" is displayed.

With Basic Management, Performance Hub provides **ASH Analytics** and **SQL Monitoring**. Advanced Management will additionally provide **ADDM**, **Workload**, and **Blocking Sessions**.

Related Topics

- Using the Console to Enable Database Management for a Container Database (CDB)
  To enable Database Management for a container database (CDB), use this procedure.
- Using the Console to Enable Database Management for a Pluggable Database (PDB)
  To enable Database Management for a pluggable database (PDB), use this procedure.

Policy Details for Oracle Exadata Database Service on Exascale Infrastructure

This topic covers details for writing policies to control access to Oracle Exadata Database Service on Exascale Infrastructure resources.

> **Note:**
>
> For more information on Policies, see "How Policies Work".
>
> For a sample policy, see "Let database admins manage Oracle Exadata Database Service on Exascale Infrastructure instances".

- About Resource-Types
  Learn about resource-types you can use in your policies.
- Resource-Types for Exadata Cloud Service Instances
  Instance resource types include aggregate resource types and individual resource types.
- Supported Variables
  Use variables when adding conditions to a policy.
- Details for Verb + Resource-Type Combinations
  Review the list of permissions and API operations covered by each verb.

Related Topics

- How Policies Work
- Let database admins manage Oracle Exadata Database Service on Exascale Infrastructure instances

About Resource-Types

Learn about resource-types you can use in your policies.

An aggregate resource-type covers the list of individual resource-types that directly follow. For example, writing one policy to allow a group to have access to the `database-family` is equivalent to writing separate policies for the group that would grant access to the `cloud-exadata-infrastructures`, `cloud-vmclusters`, `db-nodes`, `db-homes`, `databases`, `database-software-image`, and `backups` resource-types. For more information, see Resource-Types.

Resource-Types for Exadata Cloud Service Instances

Instance resource types include aggregate resource types and individual resource types.

Aggregate Resource-Type

`database-family`

Individual Resource-Types

`db-nodes`

`db-homes`

`databases`

`pluggable-databases`

`db-backups`

`dbnode-console-connection`

Supported Variables

Use variables when adding conditions to a policy.

Oracle Exadata Database Service on Exascale Infrastructure supports only the general variables. For more information, see "General Variables for All Requests".

Related Topics

- General Variables for All Requests

Details for Verb + Resource-Type Combinations

Review the list of permissions and API operations covered by each verb.

For more information, see "Permissions", "Verbs", and "Resource-Types".

- Database-Family Resource Types
  Understand the level of access of each verb.
- db-backups
  Review the list of permissions and API operations for `db-backups` resource-type.
- databases (CDBs)
  Review the list of permissions and API operations for `databases` resource-type.

- data-guard-association
  Review the list of permissions and API operations for `data-guard-association` resource-type.
- db-nodes
  Review the list of permissions and API operations for `db-nodes` resource-type.
- db-homes
  Review the list of permissions and API operations for `db-homes` resource-type.
- database-software-images
  Review the list of permissions and API operations for `database-software-images` resource-type.
- exadb-vm-clusters
  Review the list of permissions and API operations for the `exadb-vm-clusters` resource-type.
- exascale-db-storage-vaults
  Review the list of permissions and API operations for the `exascale-db-storage-vaults` resource-type.
- key-stores
  Review the list of permissions and API operations for `key-store` resource-type.
- Permissions Required for Each API Operation
  The following tables list the API operations for Oracle Exadata Database Service on Exascale Infrastructure instances in a logical order, grouped by resource type.
- pluggable-databases (PDBs)
  Review the list of permissions and API operations for `pluggable-databases` resource-type.

Related Topics

- Permissions
- Verbs
- Resource-Types

Database-Family Resource Types

Understand the level of access of each verb.

The level of access is cumulative as you go from `inspect` > `read` > `use` > `manage`. A plus sign (+) in a table cell indicates incremental access compared to the cell directly above it, whereas "no extra" indicates no incremental access.

For example, the `read` verb for the `vmclusters` resource-type covers no extra permissions or API operations compared to the `inspect` verb. However, the `use` verb includes one more permission, fully covers one more operation, and partially covers another additional operation.

db-backups

Review the list of permissions and API operations for `db-backups` resource-type.

Verbs	Permissions	APIs Fully Covered	APIs Partially Covered
inspect	DB_BACKUP_INSPECT	GetBackup ListBackups	ChangeCloudVmClusterCompartment (also needs use cloud-vmclusters, use db-homes, and use databases)
read	*INSPECT +* DB_BACKUP_CONTENT_READ	*none*	RestoreDatabase (also needs use databases)
use	*no extra*	*no extra*	*none*
manage	*USE +* DB_BACKUP_CREATE DB_BACKUP_DELETE	DeleteBackup	CreateBackup (also needs read databases)

databases (CDBs)

Review the list of permissions and API operations for databases resource-type.

Verbs	Permissions	APIs Fully Covered	APIs Partially Covered
inspect	DATABASE_INSPECT	ListDatabases GetDatabase ListDataGuardAssociations GetDataGuardAssociation	enableDatabaseManagement disableDatabaseManagement updateDatabaseManagement
read	*INSPECT+* DATABASE_CONTENT_READ	*no extra*	*no extra*
use	*READ +* DATABASE_CONTENT_WRITE DATABASE_UPDATE	UpdateDatabase SwitchoverDataGuardAssociation FailoverDataGuardAssociation ReinstateDataGuardAssociation	CreateDataGuardAssociation ChangeCloudVmClusterCompartment (also needs use cloud-vmclusters, use db-homes, and inspect db-backups) enableDatabaseManagement disableDatabaseManagement updateDatabaseManagement

Verbs	Permissions	APIs Fully Covered	APIs Partially Covered
manage	*USE* + DATABASE_CREATE DATABASE_DELETE	*no extra*	CreateDatabase (also needs use cloud-vmclusters, use db-homes, and if automatic backups to be enabled, also needs manage backups) DeleteDatabase (also needs use cloud-vmclusters, use db-homes, and if automatic backups to be enabled, also needs manage backups) CreateCloudVmCluster, DeleteCloudVmCluster (both also need manage cloud-vmclusters, manage db-homes, use vnics, and use subnets)

data-guard-association

Review the list of permissions and API operations for data-guard-association resource-type.

Table 5-6 INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
DATABASE_INSPECT	ListDataGuardAssociations, GetDataGuardAssociation	CreateDataGuardAssociation

Table 5-7 READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*no extra*

Table 5-8 USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + VM_CLUSTER_UPDATE + DB_HOME_UPDATE DATABASE_UPDATE	DeleteDatabase SwitchoverDataGuardAssociation,FailoverDataGuardAssociation, ReinstateDataGuardAssociation	CreateDataGuardAssociation

Table 5-9 MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + DATABASE_DELETE	DeleteDatabase	*none*

db-nodes

Review the list of permissions and API operations for db-nodes resource-type.

> **Note:**
>
> For Oracle Exadata Database Service on Exascale Infrastructure VM clusters, the database node is sometimes referred to as a virtual machine.

Verbs	Permissions	APIs Fully Covered	APIs Partially Covered
inspect	DB_NODE_INSPECT DB_NODE_QUERY	GetDbNode	*none*
read	*no extra*	*no extra*	*none*
use	DB_NODE_UPDATE	UpdateDbNode	*none*
manage	*USE* + DB_NODE_POWER_ACTIONS	DbNodeAction	*none*

db-homes

Review the list of permissions and API operations for db-homes resource-type.

Verbs	Permissions	APIs Fully Covered	APIs Partially Covered
inspect	DB_HOME_INSPECT	ListDBHome GetDBHome ListDbHomePatches ListDbHomePatchHistoryEntries GetDbHomePatch GetDbHomePatchHistoryEntry	*none*
read	*no extra*	*no extra*	*none*
use	DB_HOME_UPDATE	UpdateDBHome	ChangeCloudVmClusterCompartment (also needs use cloud-vmclusters, use databases, and inspect backups)

Verbs	Permissions	APIs Fully Covered	APIs Partially Covered
manage	USE + DB_HOME_CREATE DB_HOME_DELETE	no extra	CreateCloudVmCluster, DeleteCloudVmCluster (both also need manage cloud-vmclusters, manage databases, use vnics, and use subnets). If automatic backups are enabled on the default database, also needs manage backups CreateDbHome, (also needs use cloud-vmclusters and manage databases). If creating the Database Home by restoring from a backup, also needs read backups DeleteDbHome, (also needs use cloud-vmclusters and manage databases). If automatic backups are enabled on the default database, also needs manage backups. If the performFinalBackup option is selected, also needs manage backups and read databases.

database-software-images

Review the list of permissions and API operations for database-software-images resource-type.

Verbs	Permissions	APIs Fully Covered	APIs Partially Covered
inspect	DB_SOFTWARE_IMG_INSPECT	ListDatabaseSoftwareImages GetDatabaseSoftwareImage	*none*
read	no extra	none	*none*
use	*READ* + DB_SOFTWARE_IMG_UPDATE	UpdateDatabaseSoftwareImage ChangeDatabaseSoftwareImageCompartment	*none*

Verbs	Permissions	APIs Fully Covered	APIs Partially Covered
manage	*USE* + DB_SOFTWARE_IMG_CREATE DB_SOFTWARE_IMG_DELETE	CreateDatabaseSoftwareImage DeleteDatabaseSoftwareImage	*none*

exadb-vm-clusters

Review the list of permissions and API operations for the exadb-vm-clusters resource-type.

Table 5-10 INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
EXADB_VM_CLUSTER_INSPECT	ListExadbVmClusters GetExadbVmCluster ListExadbVmClusterUpdates GetExadbVmClusterUpdate ListExadbVmClusterUpdateHistoryEntries GetExadbVmClusterUpdateHistoryEntry	None

Table 5-11 READ

Permissions	APIs Fully Covered	APIs Partially Covered
No extra	*No extra*	None

Table 5-12 USE

Permissions	APIs Fully Covered	APIs Partially Covered
inspect + EXADB_VM_CLUSTER_UPDATE	RemoveVirtualMachineFromExadbVmClusterDetails	ChangeExadbVmClusterCompartment (also needs use db-homes, use databases, and inspect db-backups)

Table 5-13 MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered

`use + EXADB_VM_CLUSTER_CREATE, EXADB_VM_CLUSTER_DELETE`	*No extra*	`CreateExadbVmCluster`
		(also needs `manage db-homes`, `manage databases`, `use exascale-db-storage-vaults`, `use vnics`, and `use subnets`)
		`DeleteExadbVmCluster`
		(also needs `manage db-homes`, `manage databases`, `use exascale-db-storage-vaults`, `use vnics`, and `use subnets`)
		`UpdateExadbVmCluster`
		(also needs `use subnets`, `use vnics`, and `use private-ip`)

exascale-db-storage-vaults

Review the list of permissions and API operations for the `exascale-db-storage-vaults` resource-type.

Table 5-14 INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
`EXASCALE_DB_STORAGE_VAULT_INSPECT`	`ListExascaleDbStorageVaults` `GetExascaleDbStorageVault`	None

Table 5-15 READ

Permissions	APIs Fully Covered	APIs Partially Covered
No extra	*No extra*	None

Table 5-16 USE

Permissions	APIs Fully Covered	APIs Partially Covered
`inspect + EXASCALE_DB_STORAGE_VAULT_UPDATE`	`ChangeExascaleDbStorageVaultCompartment` `UpdateExascaleDbStorageVault`	None

Table 5-17 MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
use + EXASCALE_DB_STORAGE_VAULT_CREATE EXASCALE_DB_STORAGE_VAULT_DELETE	CreateExascaleDbStorageVault DeleteExascaleDbStorageVault	None

key-stores

Review the list of permissions and API operations for `key-store` resource-type.

Table 5-18 INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
KEY_STORE_INPSECT AUTONOMOUS_CONTAINER_DATABASE_INSPECT AUTONOMOUS_DATABASE_INSPECT AUTONOMOUS_DB_BACKUP_INSPECT	GetKeyStore GetAutonomousContainerDatabase GetAutonomousDatabase GetAutonomousDatabaseBackup	ChangeKeyStoreCompartment RotateAutonomousContainerDatabaseKey

Table 5-19 READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*no extra*

Table 5-20 USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + KEY_STORE_UPDATE + AUTONOMOUS_VM_CLUSTER_UPDATE + AUTONOMOUS_CONTAINER_DATABASE_UPDATE AUTONOMOUS_DATABASE_UPDATE	UpdateKeyStore *none* *none* *none* RotateAutonomousDatabaseKey	ChangeKeyStoreCompartment CreateAutonomousContainerDatabase RotateAutonomousContainerDatabaseKey *none*

Table 5-21 MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + KEY_STORE_CREATE + KEY_STORE_DELETE + AUTONOMOUS_CONTAINER_DATABASE_CREATE	CreateKeyStore DeleteKeyStore CreateAutonomousContainerDatabase	*none* *none* *none*

Permissions Required for Each API Operation

The following tables list the API operations for Oracle Exadata Database Service on Exascale Infrastructure instances in a logical order, grouped by resource type.

Database API Operations

For information about permissions, see:

Permissions.

The following tables list of API operations and permissions by API peration.

Table 5-22 Cloud Exadata Infrastructure Resource

API Operation	Permissions Required to Use the Operation
`ListCloudExadataInfrastructures`	`CLOUD_EXADATA_INFRASTRUCTURE_INSPECT`
`GetCloudExadataInfrastructure`	`CLOUD_EXADATA_INFRASTRUCTURE_INSPECT`
`CreateCloudExadataInfrastructure`	`CLOUD_EXADATA_INFRASTRUCTURE_CREATE`
`UpdateCloudExadataInfrastructure`	`CLOUD_EXADATA_INFRASTRUCTURE_UPDATE`
`ChangeCloudExadataInfrastructureCompartment`	`CLOUD_EXADATA_INFRASTRUCTURE_UPDATE`
`DeleteCloudExadataInfrastructure`	`CLOUD_EXADATA_INFRASTRUCTURE_DELETE`
`AddStorageCapacityCloudExadataInfrastructure`	`CLOUD_EXADATA_INFRASTRUCTURE_UPDATE`

Table 5-23 Cloud VM Cluster

API Operation	Permissions Required to Use the Operation
`ListCloudVmClusters`	`CLOUD_VM_CLUSTER_INSPECT`
`GetCloudVmCluster`	`CLOUD_VM_CLUSTER_INSPECT`
`CreateCloudVmCluster`	`CLOUD_VM_CLUSTER_CREATE` and `CLOUD_EXADATA_INFRASTRUCTURE_UPDATE` and `VNIC_CREATE and VNIC_ATTACH` and `SUBNET_ATTACH` and (needed if Private DNS is used: `DNS_ZONE_READ`, `DNS_RECORD_UPDATE`, `DNS_ZONE_CREATE` `DNS_VIEW_INSPECT`)
`ChangeCloudVmClusterCompartment`	`CLOUD_VM_CLUSTER_UPDATE`
`UpdateCloudVmCluster`	`CLOUD_VM_CLUSTER_UPDATE` and `CLOUD_EXADATA_INFRASTRUCTURE_UPDATE`
`GetCloudVmClusterIormConfig`	`CLOUD_VM_CLUSTER_INSPECT`
`UpdateCloudVmClusterIormConfig`	`CLOUD_VM_CLUSTER_UPDATE`
`DeleteCloudVmCluster`	`CLOUD_VM_CLUSTER_DELETE` and `CLOUD_EXADATA_INFRASTRUCTURE_UPDATE` and `DB_HOME_DELETE` and `VNIC_DELETE` and `SUBNET_DETACH` and `VNIC_DETACH` and (needed if Private DNS is used: `DNS_ZONE_READ`, `DNS_RECORD_UPDATE`, `DNS_ZONE_DELETE`)

Table 5-23 (Cont.) Cloud VM Cluster

API Operation	Permissions Required to Use the Operation
AddVmToCloudVmCluster	CLOUD_VM_CLUSTER_UPDATE and CLOUD_EXADATA_INFRASTRUCTURE_UPDATE and (needed if Private DNS is used: DNS_ZONE_READ, DNS_RECORD_UPDATE, DNS_ZONE_CREATE, DNS_VIEW_INSPECT)
RemoveVmFromCloudVmCluster	CLOUD_VM_CLUSTER_UPDATE and CLOUD_EXADATA_INFRASTRUCTURE_UPDATE and (needed if Private DNS is used: DNS_ZONE_READ, DNS_RECORD_UPDATE, DNS_ZONE_DELETE)

Table 5-24 Cloud VM Cluster Maintenance Updates and Update History

API Operation	Permissions Required to Use the Operation
ListCloudVmClusterUpdates	CLOUD_VM_CLUSTER_INSPECT
GetCloudVmClusterUpdate	CLOUD_VM_CLUSTER_INSPECT
ListCloudVmClusterUpdateHistoryEntries	CLOUD_VM_CLUSTER_INSPECT
GetCloudVmClusterUpdateHistoryEntry	CLOUD_VM_CLUSTER_INSPECT

Table 5-25 Virtual Machines / Nodes

API Operation	Permissions Required to Use the Operation
ListDbNodes	DB_NODE_INSPECT
GetDbNode	DB_NODE_INSPECT
DbNodeAction	DB_NODE_POWER_ACTIONS

Table 5-26 Database Homes

API Operation	Permissions Required to Use the Operation
ListDbHomes	DB_HOME_INSPECT
GetDbHome	DB_HOME_INSPECT
ListDbHomePatches	DB_HOME_INSPECT
ListDbHomePatchHistoryEntries	DB_HOME_INSPECT
GetDbHomePatch	DB_HOME_INSPECT
GetDbHomePatchHistoryEntry	DB_HOME_INSPECT
CreateDbHome	DB_SYSTEM_INSPECT and DB_SYSTEM_UPDATE and DB_HOME_CREATE and DATABASE_CREATE To enable automatic backups for the database, also need DB_BACKUP_CREATE and DATABASE_CONTENT_READ
UpdateDbHome	DB_HOME_UPDATE

Table 5-26 (Cont.) Database Homes

API Operation	Permissions Required to Use the Operation
DeleteDbHome	DB_SYSTEM_UPDATE and DB_HOME_DELETE and DATABASE_DELETE If automatic backups are enabled, also need DELETE_BACKUP If performing a final backup on termination, also need DB_BACKUP_CREATE and DATABASE_CONTENT_READ

Table 5-27 Databases (CDB)

API Operation	Permissions Required to Use the Operation
ListDatabases	DATABASE_INSPECT
GetDatabase	DATABASE_INSPECT
CreateDatabase	DATABASE_UPDATE To enable automatic backups, also need DB_BACKUP_CREATE and DATABASE_CONTENT_READ
UpdateDatabase	DATABASE_UPDATE To enable automatic backups, also need DB_BACKUP_CREATE and DATABASE_CONTENT_READ
DeleteDatabase	For new resource model using VM cluster resource: CLOUD_VM_CLUSTER_INSPECT and DB_HOME_UPDATE and DATABASE_DELETE
enableDatabaseManagement	DATABASE_INSPECT and DATABASE_UPDATE
disableDatabaseManagement	DATABASE_INSPECT and DATABASE_UPDATE
disableDatabaseManagement	DATABASE_INSPECT and DATABASE_UPDATE

Table 5-28 Pluggable Databases (PBDs)

API Operation	Permissions Required to Use the Operation
ListPluggableDatabase	PLUGGABLE_DATABASE_INSPECT
GetPluggableDatabase	PLUGGABLE_DATABASE_INSPECT
CreatePluggableDatabase	PLUGGABLE_DATABASE_CREATE and DATABASE_INSPECT and DATABASE_UPDATE
UpdatePluggableDatabase	PLUGGABLE_DATABASE_INSPECT and PLUGGABLE_DATABASE_UPDATE
StartPluggableDatabase	PLUGGABLE_DATABASE_INSPECT and PLUGGABLE_DATABASE_UPDATE
StopPluggableDatabase	PLUGGABLE_DATABASE_INSPECT and PLUGGABLE_DATABASE_UPDATE
DeletePluggableDatabase	PLUGGABLE_DATABASE_DELETE and DATABASE_INSPECT and DATABASE_UPDATE

Table 5-28 (Cont.) Pluggable Databases (PBDs)

API Operation	Permissions Required to Use the Operation
LocalClonePluggableDatabase	PLUGGABLE_DATABASE_INSPECT and PLUGGABLE_DATABASE_UPDATE and PLUGGABLE_DATABASE_CONTENT_READ and PLUGGABLE_DATABASE_CONTENT_WRITE and PLUGGABLE_DATABASE_CREATE and DATABASE_INSPECT and DATABASE_UPDATE
RemoteClonePluggableDatabase	PLUGGABLE_DATABASE_INSPECT and PLUGGABLE_DATABASE_UPDATE and PLUGGABLE_DATABASE_CONTENT_READ and PLUGGABLE_DATABASE_CONTENT_WRITE and PLUGGABLE_DATABASE_CREATE and DATABASE_INSPECT and DATABASE_UPDATE
enableDatabaseManagement	DATABASE_INSPECT and DATABASE_UPDATE
disableDatabaseManagement	DATABASE_INSPECT and DATABASE_UPDATE
disableDatabaseManagement	DATABASE_INSPECT and DATABASE_UPDATE

Table 5-29 System Shapes and Database Versions

API Operation	Permissions Required to Use the Operation
ListDbSystemShapes	(no permissions required; available to anyone)
ListDbVersions	(no permissions required; available to anyone)

Table 5-30 Oracle Data Guard Associations

API Operation	Permissions Required to Use the Operation
GetDataGuardAssociation	DATABASE_INSPECT
ListDataGuardAssociations	DATABASE_INSPECT
CreateDataGuardAssociation	DB_SYSTEM_UPDATE and DB_HOME_CREATE and DB_HOME_UPDATE and DATABASE_CREATE and DATABASE_UPDATE
SwitchoverDataGuardAssociation	DATABASE_UPDATE
FailoverDataGuardAssociation	DATABASE_UPDATE
ReinstateDataGuardAssociation	DATABASE_UPDATE

Table 5-31 Backups and Database Restore

API Operation	Permissions Required to Use the Operation
GetBackup	DB_BACKUP_INSPECT
ListBackups	DB_BACKUP_INSPECT
CreateBackup	DB_BACKUP_CREATE and DATABASE_CONTENT_READ
DeleteBackup	DB_BACKUP_DELETE and DB_BACKUP_INSPECT

Table 5-31 (Cont.) Backups and Database Restore

API Operation	Permissions Required to Use the Operation
RestoreDatabase	DB_BACKUP_INSPECT and DB_BACKUP_CONTENT_READ and DATABASE_CONTENT_WRITE

Table 5-32 Application VIP

API Operation	Permissions Required to Use the Operation
CreateApplicationVip	APPLICATION_VIP_CREATE and CLOUD_VM_CLUSTER_UPDATE and PRIVATE_IP_CREATE and PRIVATE_IP_ASSIGN and VNIC_ASSIGN and SUBNET_ATTACH
DeleteApplicationVip	APPLICATION_VIP_DELETE and CLOUD_VM_CLUSTER_UPDATE and PRIVATE_IP_DELETE and PRIVATE_IP_UNASSIGN and VNIC_UNASSIGN and SUBNET_DETACH
ListApplicationVips	APPLICATION_VIP_INSPECT
ListApplicationVips	APPLICATION_VIP_INSPECT

Table 5-33 Serial Console Access to VM

API Operation	Permissions Required to Use the Operation
AddVirtualMachineToVmCluster	VM_CLUSTER_UPDATE and EXADATA_INFRASTRUCTURE_UPDATE
RemoveVirtualMachineFromVmCluster	VM_CLUSTER_UPDATE and EXADATA_INFRASTRUCTURE_UPDATE
CreateDbNodeConsoleConnection	DBNODE_CONSOLE_CONNECTION_CREATE and DBNODE_CONSOLE_CONNECTION_INSPECT
GetDbNodeConsoleConnection	DBNODE_CONSOLE_CONNECTION_INSPECT
ListDbNodeConsoleConnections	DBNODE_CONSOLE_CONNECTION_INSPECT
DeleteDbNodeConsoleConnection	DBNODE_CONSOLE_CONNECTION_DELETE
UpdateDbNodeConsoleConnection	DBNODE_CONSOLE_CONNECTION_UPDATE
UpdateDbNode	DB_NODE_UPDATE

pluggable-databases (PDBs)

Review the list of permissions and API operations for pluggable-databases resource-type.

Verbs	Permissions	APIs Fully Covered	APIs Partially Covered
inspect	PLUGGABLE_DATABASE_INSPECT	ListPluggableDatabases GetPluggableDatabase	UpdatePluggableDatabase StartPluggableDatabase StopPluggableDatabase LocalClonePluggableDatabase RemoteClonePluggableDatabase RefreshPluggableDatabase ConvertRefreshablePluggableDatabase
	DATABASE_INSPECT	*no extra*	CreatePluggableDatabase DeletePluggableDatabase LocalClonePluggableDatabase RemoteClonePluggableDatabase
read	*INSPECT* + PLUGGABLE_DATABASE_CONTENT_READ	*no extra*	CreatePluggableDatabase (Additional permissions are required if auto-backups are enabled on the CDB and includes this PDB.) UpdatePluggableDatabase (Additional permissions are required if auto-backups are enabled on the CDB and includes this PDB.) LocalClonePluggableDatabase RemoteClonePluggableDatabase
use	*READ* + PLUGGABLE_DATABASE_CONTENT_WRITE	*no extra*	LocalClonePluggableDatabase RemoteClonePluggableDatabase

Verbs	Permissions	APIs Fully Covered	APIs Partially Covered
	PLUGGABLE_DATABASE_UPDATE	*no extra*	UpdatePluggableDatabase StartPluggableDatabase StopPluggableDatabase LocalClonePluggableDatabase RemoteClonePluggableDatabase RefreshPluggableDatabase ConvertRefreshablePluggableDatabase
	DATABASE_UPDATE	*no extra*	CreatePluggableDatabase DeletePluggableDatabase LocalClonePluggableDatabase RemoteClonePluggableDatabase
manage	*USE* + PLUGGABLE_DATABASE_CREATE	*no extra*	CreatePluggableDatabase LocalClonePluggableDatabase RemoteClonePluggableDatabase
	PLUGGABLE_DATABASE_DELETE	*no extra*	DeletePluggableDatabase

Managing Exadata Resources with Oracle Enterprise Manager Cloud Control

To manage and monitor Exadata Cloud Infrastructure and Exadata Database Service on Cloud@Customer resources, use Oracle Enterprise Manager Cloud Control.

For complete documentation and Oracle By Example tutorials, see the following documentation resources: *Oracle Enterprise Manager Cloud Control for Oracle Exadata Cloud* and *Setting Up Oracle Enterprise Manager 13.4 on Oracle Cloud Infrastructure*.

- Overview of Oracle Enterprise Manager Cloud Control
  Oracle Enterprise Manager Cloud Control provides a complete lifecycle management solution for Oracle Cloud Infrastructure's Exadata Cloud Infrastructure (ExaDB-D) and Exadata Database Service on Cloud@Customer (ExaDB-C@C) services.
- Features of Enterprise Manager Cloud Control
  Familiarize yourself with the features of Enterprise Manager Cloud Control to manage and monitor Exadata Cloud and Exadata Cloud@Customer resources.

- Analyzing Exadata Cloud Service Database Performance
  This topic describes how to use Database Metrics and Performance Hub to monitor, analyze, and tune the performance of OCI user-managed databases, including Oracle Exadata Database Service on Exascale Infrastructure databases and databases running on virtual machine and bare metal systems.

Related Topics

- Oracle Enterprise Manager Cloud Control for Oracle Exadata Cloud
- Setting Up Oracle Enterprise Manager 13.4 on Oracle Cloud Infrastructure

Overview of Oracle Enterprise Manager Cloud Control

Oracle Enterprise Manager Cloud Control provides a complete lifecycle management solution for Oracle Cloud Infrastructure's Exadata Cloud Infrastructure (ExaDB-D) and Exadata Database Service on Cloud@Customer (ExaDB-C@C) services.

Enterprise Manager Cloud Control discovers ExaDB-D and ExaDB-C@C services as a single target and automatically identifies and organizes all dependent components. Using Enterprise Manager Cloud Control you can then:

- Monitor and manage all Exadata, ExaDB-D and ExaDB-C@C systems, along with any other targets, from a single interface
- Visualize storage and compute data
- View performance metrics of your Exadata components

Features of Enterprise Manager Cloud Control

Familiarize yourself with the features of Enterprise Manager Cloud Control to manage and monitor Exadata Cloud and Exadata Cloud@Customer resources.

Enterprise Manager Target for Exadata Cloud

The target for Oracle Cloud Infrastructure Exadata resources, which covers both Exadata Cloud and Exadata Cloud@Customer does the following:

- Automatically identifies and organizes related targets.
- Provides a high-level integration point for Enterprise Manager framework features such as incident rules, groups, notifications, and monitoring templates.

Improved Performance Monitoring

Enterprise Manager Cloud Control enhances performance monitoring in the following ways:

- Adds Exadata Storage Server and Exadata Storage Grid targets.
- Offers visualization of storage and compute performance for your Exadata Cloud and Exadata Cloud@Customer resources.
- Enables use of the same Maximum Availability Architecture (MAA) key performance indicators (KPI) developed for Oracle Exadata Database Machine.

Scripted CLI-based Discovery

Enterprise Manager Cloud Control uses scripts to discover Oracle Cloud Infrastructure Exadata resources. Scripts search the existing hosts, clusters, ASM, databases and related targets, and add the storage server targets.

"Single Pane of Glass" View of On-Premises and Oracle Cloud Infrastructure Exadata Resources

Enterprise Manager Cloud Control 's use of a single Exadata target type provides a consistent Enterprise Manager experience across on-premises, Exadata Cloud, and Exadata Cloud@Customer resources. The common Exadata target menu allows you to easily navigate to, monitor and manage all of your Exadata systems.

Visualization

Enterprise Manager Cloud Control allows you to visualize the database and related targets associated with each Exadata Cloud and Exadata Cloud@Customer system.

Analyzing Exadata Cloud Service Database Performance

This topic describes how to use Database Metrics and Performance Hub to monitor, analyze, and tune the performance of OCI user-managed databases, including Oracle Exadata Database Service on Exascale Infrastructure databases and databases running on virtual machine and bare metal systems.

With this tool, you can view real-time and historical performance data. For information about using Performance Hub, see Using Performance Hub to Analyze Database Performance.

To use Database Metrics and Performance Hub for Oracle Exadata Database Service on Exascale Infrastructure, Virtual Machine, and Bare Metal databases, Database Management must be enabled for the database. When enabling a database, the database administrator can choose from two database management options: Basic Management and Full Management. For information about using Database Metrics and Performance Hub with Virtual Machine, Bare Metal, Oracle Exadata Database Service on Exascale Infrastructure and external databases, see Enable Database Management.

> **Note:**
>
> Using Identity and Access Management (IAM), you can create a policy that grants users access to Performance Hub while limiting actions they can take on Autonomous Databases, databases running on virtual machine and bare metal systems, Oracle Database Cloud Service, Oracle Exadata Database Service on Exascale Infrastructure, and external databases. For information about IAM policies and ExaDB-XS databases, see *Required IAM Policy* . For information about policies and how to use them, see How Policies Work.

Related Topics

- Required IAM Policy for Oracle Exadata Database Service on Exascale Infrastructure
  Review the identity access management (IAM) policy for provisioning Oracle Exadata Database Service on Exascale Infrastructure systems.

Security Guide for Oracle Exadata Database Service on Exascale Infrastructure

This guide describes security for an Oracle Exadata Database Service on Exascale Infrastructure. It includes information about the best practices for securing the Oracle Exadata Database Service on Exascale Infrastructure.

- Part 1: Security Configurations and Default Enabled Features
- Enabling Additional Security Capabilities

Part 1: Security Configurations and Default Enabled Features

- Responsibilities
  Oracle Exadata Database Service on Exascale Infrastructure is jointly managed by the customer and Oracle.
- Infrastructure Security
  Secutrity features offered by Oracle Exadata Database Service on Exascale Infrastructure.
- Guiding Principles Followed for Security Configuration Defaults
- Security Features
- Guest VM Default Fixed Users
  Several user accounts regularly manage the components of Oracle Exadata Database Service on Exascale Infrastructure. These users are required and may not be modified.
- Default Security Settings: Customer VM
  Learn about the default security settings used with Oracle Exadata Database Service on Exascale Infrastructure instances.
- Default Processes on Customer VM
  A list of the processes that run by default on the customer VM, also called DOMU, or Guest VM and Guest OS
- Default Backup Security Configuration
  Learn about the backup security policy for
- Operator Access to Customer System and Customer Data
  Only automated tooling is permitted to access VM for lifecycle automation.
- Compliance Requirements
- Break Glass Procedure for Accessing Customer's Guest VM
  There are situations where some problems can only be resolved by Oracle logging into the customer guest VM.

Responsibilities

Oracle Exadata Database Service on Exascale Infrastructure is jointly managed by the customer and Oracle.

The Oracle Exadata Database Service on Exascale Infrastructure deployment is divided into two areas of responsibility:

Customer accessible services: components that the customer can access as part of their subscription to Oracle Exadata Database Service on Exascale Infrastructure

- Customer accessible virtual machines (VM)
- Customer accessible database services

Oracle Managed Infrastructure: components that are owned and operated by Oracle to run customer accessible services

- Power Distribution Units (PDUs)
- Out of band (OOB) management switches » InfiniBand switches
- Exadata Storage Servers

- Physical Exadata database servers
- Datacenter security which hosts Exadata Servers with customer information

Customers control and monitor access to customer services, including network access to their VMs (through OCI Virtual Cloud Networks and OCI Security Lists), authentication to access the VM, and authentication to access databases running in the VMs. Oracle controls and monitors access to Oracle Managed Infrastructure components and physical server security. Oracle staff are not authorized to access customer services, including customer VMs and databases except where customers are unable to access the customer VM. See the Exadata Cloud Service Security Controls document, https://www.oracle.com/a/ocom/docs/engineered-systems/exadata/exadata-cloud-service-security.pdf, Exception Workflows .

Customers access Oracle databases (DB) running on Oracle Exadata Database Service on Exascale Infrastructure via client and backup VCNs to the databases running in the customer VM using standard Oracle database connection methods, such as Oracle Net on port 1521. Customer's access the VM running the Oracle databases via standard Oracle Linux methods, such as token based ssh on port 22.

Infrastructure Security

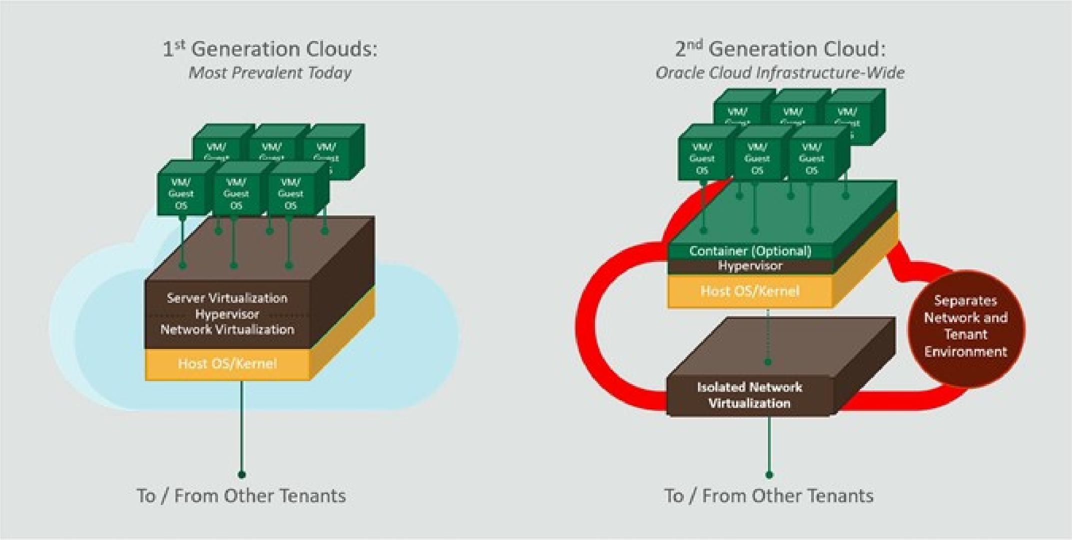
Secutrity features offered by Oracle Exadata Database Service on Exascale Infrastructure.

- **Oracle Cloud Physical Security**
  Oracle Cloud data centers align with Uptime Institute and Telecommunications Industry Association (TIA) ANSI/TIA-942-A Tier 3 (99.982% Availability) or Tier 4 (99.995% Availability) standards and follow a N2 ('N' stands for Need) redundancy methodology for critical equipment operation. Data centers housing Oracle Cloud Infrastructure services use redundant power sources and maintain generator backups in case of widespread electrical outage. Server rooms are closely monitored for air temperature and humidity, and fire-suppression systems are in place. Data center staff are trained in incident response and escalation procedures to address security and availability events that may arise. For more information see *Oracle Cloud Infrastructure Security Guide*. For further details on Oracle Cloud Infrastructure Data Center compliance, and see *Oracle Cloud Compliance*.
- **Operator access to customer systems**
  Oracle access protocols include:
  - Physical access to facilities is limited to certain Oracle employees, contractors, and authorized visitors.
  - Oracle employees, subcontractors, and authorized visitors are issued identification cards that must be worn while on Oracle premises.
  - Visitors are required to sign a visitor's register, be escorted and/or observed when they are on Oracle premises, and/or be bound by the terms of a confidentiality agreement with Oracle.
  - Security monitors the possession of keys/access cards and the ability to access facilities. Staff leaving Oracle's employment must return keys/cards and key/cards are deactivated upon termination.
  - Security authorizes all repairs and modifications to the physical security barriers or entry controls at service locations.
  - Oracle use a mixture of 24/7 onsite security officers or patrol officers, depending on the risk/protection level of the facility. In all cases officers are responsible for patrols, alarm response, and recording of security incidents.
  - Oracle has implemented centrally managed electronic access control systems with integrated intruder alarm capability. The access logs are kept for a minimum of six

months. Furthermore, the retention period for CCTV monitoring and recording ranges from 30-90 days minimum, depending on the facility's functions and risk level.

- **Hypervisor Customer Isolation**
  The hypervisor is the software that manages virtual devices in a cloud environment, handling server and network virtualization. In traditional virtualization environments, the hypervisor manages network traffic, enabling it to flow between VM instances and between VM instances and physical hosts. This adds considerable complexity and computational overhead in the hypervisor. Proof-of concept computer security attacks, such as virtual machine escape attacks, have highlighted the substantial risk that can come with this design. These attacks exploit hypervisor complexity by enabling an attacker to "breakout" of a VMinstance, access the underlying operating system, and gain control of the hypervisor. The attacker can then access other hosts, sometimes undetected. Oracle Cloud Infrastructure reduces this risk by decoupling network virtualization from the hypervisor. We've implemented network virtualization as a highly customized hardware and software layer that moves cloud control away from the hypervisor and host, and puts it on its own network. This hardened and monitored layer of control is what enables isolated network virtualization. Isolated network virtualization reduces risk by limiting the attack surface. Even if a malicious actor succeeds with a VM escape attack on a single host, it's designed so they can't reach other hosts in the cloud infrastructure. The attack is effectively contained to the one host. Isolated network virtualization is implemented in every data center in every region, which means that all Oracle Cloud Infrastructure tenants benefit from this reduced risk.

Figure 5-1 Isolated Network Virualization Reduces Risk in Oracle Generation 2 Cloud

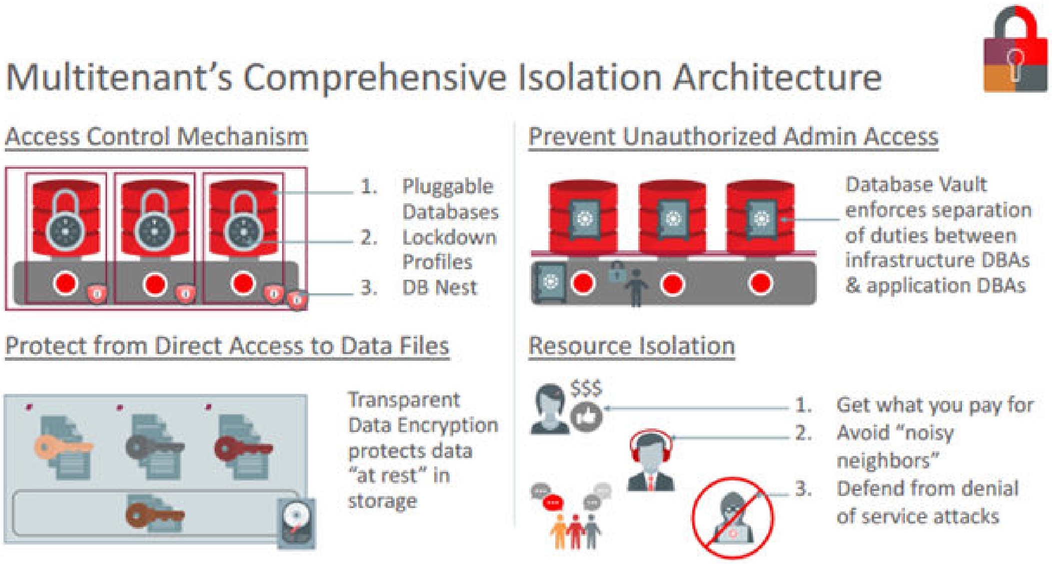


- **Multitenant Security**
  Consistent with our security philosophy of Defense in Depth, Multitenant has a comprehensive isolation architecture.

  There are four major categories to this, with several important features in each category.

  1. Access Control Mechanism
  2. Prevent Unauthorized Admin Access
  3. Protect from direct access to Data Files
  4. Resource Isolation

Figure 5-2 Multitenant's Comprehensive Isolation Architecture


Multitenant's Comprehensive Isolation Architecture
Access Control Mechanism
1. Pluggable Databases
2. Lockdown Profiles
3. DB Nest
Prevent Unauthorized Admin Access
Database Vault enforces separation of duties between infrastructure DBAs & application DBAs
Protect from Direct Access to Data Files
Transparent Data Encryption protects data "at rest" in storage
Resource Isolation
$$$
1. Get what you pay for
2. Avoid "noisy neighbors"
3. Defend from denial of service attacks

Related Topics

- Oracle Cloud Infrastructure Security Architecture
- Oracle Cloud Infrastructure Security Guide
- Data Security: Physical and Environmental Controls
- Oracle Multitenant with Oracle Database 19c
- Oracle Cloud Compliance

Guiding Principles Followed for Security Configuration Defaults

- **Defense in Depth** Oracle Exadata Database Service on Exascale Infrastructure offers a number of controls to ensure confidentiality, integrity, and availability throughout the service.
  First, Oracle Exadata Database Service on Exascale Infrastructure is built from the hardened operating system image provided by Exadata Database Machine (https://docs.oracle.com/en/engineered-systems/exadata-database-machine/dbmsq/exadata-security-overview.html). This secures the core operating environment by restricting the installation image to only the required software packages, disabling unnecessary services, and implementing secure configuration parameters throughout the system.

  In addition to inheriting all the strength of Exadata Database Machine's mature platform, because Oracle Exadata Database Service on Exascale Infrastructure provisions systems for customers, additional secure default configuration choices are implemented in the service instances. For example, all database tablespaces require transparent data encryption (TDE), strong password enforcement for initial database users and superusers, and enhanced audit and event rules.

  Oracle Exadata Database Service on Exascale Infrastructure also constitutes a complete deployment and service, so it is subjected to industry-standard external audits such as PCI, HIPPA and ISO27001. These external audit requirements impose additional value-added service features such as antivirus scanning, automated alerting for unexpected changes to the system, and daily vulnerability scans for all Oracle-managed infrastructure systems in the fleet.
- **Least privilege**

Oracle Secure Coding Standards require software processes run at the minimum privilege level to implement their functionality.

Each process and daemon, must run as a normal, unprivileged user unless it can prove a requirement for a higher level of privilege. This helps contain any unforeseen issues or vulnerabilities to unprivileged user space and not compromise an entire system.

This principle also applies to Oracle operations team members who use individual named accounts to access the Oracle Exadata Database Service on Exascale Infrastructure for maintenance or troubleshooting. Only when necessary will they use the audited access to higher levels of privilege to solve or resolve an issue. Most issues are resolved through automation, so we also employ least privilege by not permitting human operators to access a system unless the automation is unable to resolve the issue.

- **Auditing and accountability**
  When required, access to the system is allowed, but all access and actions are logged and tracked for accountability.

  Oracle Exadata Database Service on Exascale Infrastructure audit logs are controlled by Oracle and used for security monitoring and compliance purposes. Oracle can share relevant audit logs with customers per Oracle Incident Response Practices and the Oracle Data Processing Agreement.

  Auditing capabilities are provided at all infrastructure components to ensure all actions are captured. Customers also have ability to configure auditing for their database and guest VM configuration and may choose to integrate those with other enterprise auditing systems.
- **Automating cloud operations**
  By eliminating manual operations required to provision, patch, maintain, troubleshoot, and configure systems, the opportunity for error is reduced.

Security Features

- **Hardened OS image**
  - Minimal package installation:

    Only the necessary packages required to run an efficient system are installed. By installing a smaller set of packages, the attack surface of the operating system is reduced and the system remains more secure.
  - Secure configuration:

    Many non-default configuration parameters are set during installation to enhance the security posture of the system and its content. For example, SSH is configured to only listen on certain network interfaces, sendmail is configured to only accept localhost connections, and many other similar restrictions are implemented during installation.
  - Run only necessary services:

    Any services that may be installed on the system, but not required for normal operation, are disabled by default. For example, while NFS is a service often configured by customers for various application purposes, it is disabled by default as it is not required for normal database operations. Customers may choose to optionally configure services per their requirements.
- **Minimized attack surface**

  As part of the hardened image, attack surface is reduced installing and running only the software required to deliver the service.
- **Additional security features enabled (grub passwords, secure boot)**

    - Leveraging Exadata image capabilities, ExaDB-XS enjoys the features integrated into the base image such as grub passwords and secure boot in addition to many others.
    - Through customization, customers may wish to further enhance their security posture with additional configurations.
- **Secure access methods**
    - Accessing database servers via SSH using strong cryptographic ciphers. Weak ciphers are disabled by default.
    - Accessing databases via encrypted Oracle Net connections. By default, our services are available using encrypted channels and a default configured Oracle Net client will use encrypted sessions.
    - Accessing diagnostics via Exadata MS web interface (https).
- **Auditing and logging**
    - By default, auditing is enabled for administrative operations and those audit records are communicated to OCI internal systems for automated review and alerting when required.

Guest VM Default Fixed Users

Several user accounts regularly manage the components of Oracle Exadata Database Service on Exascale Infrastructure. These users are required and may not be modified.

In all ExaDB-XS machines, Oracle uses and recommends token-based SSH login.

There are three classes of users:

- Default Users: No Logon Privileges
- Default Users WITH RESTRICTED SHELL Access
  These users are used for accomplishing a defined task with a restricted shell login. These users should not be removed as the defined task will fail in case these users are deleted.
- Default Users with Login Permissions
  These privileged users are used for accomplishing most of the tasks in the system. These users should never be altered or deleted as it would have significant impact on the running system.

Default Users: No Logon Privileges

This user list consists of default operating system users along with some specialized users like exawatch and dbmsvc. These users should not be altered. These users cannot login to the system as all are set to /sbin/nologin.

In the list of users below, most are either standard Linux OS users or related to standard Linux packages except for the exawatch and dbmsvc users.

- exawatch: The exawatch user is responsible for collecting and archiving system statistics on both the database servers and the storage servers
- dbmsvc: User is used for Management Server on the database node service in Oracle Exadata System

NOLOGIN Users

```
bin:x:1:1:bin:/bin:/sbin/nologin
Daemon:x:2:2:daemon:/sbin:/sbin/nologin
```

```
adm:x:3:4:adm:/dev/null:/sbin/nologin
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
systemd-network:x:192:192:systemd Network Management:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
rpm:x:37:37::/var/lib/rpm:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
rpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/sbin/nologin
unbound:x:999:997:Unbound DNS resolver:/etc/unbound:/sbin/nologin
nscd:x:28:28:NSCD Daemon:/:/sbin/nologin
tss:x:59:59:Account used by the trousers package to sandbox the tcsd
daemon:/dev/null:/sbin/nologin
saslauth:x:998:76:Saslauthd user:/run/saslauthd:/sbin/nologin
mailnull:x:47:47::/var/spool/mqueue:/sbin/nologin
smmsp:x:51:51::/var/spool/mqueue:/sbin/nologin
chrony:x:997:996::/var/lib/chrony:/sbin/nologin
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
nslcd:x:65:55:LDAP Client User:/:/sbin/nologin
uucp:x:10:14:Uucp user:/var/spool/uucp:/sbin/nologin
tcpdump:x:72:72::/:/sbin/nologin
exawatch:x:1010:510::/opt/oracle.ExaWatcher:/sbin/nologin
sssd:x:996:508:User forsssd:/:/sbin/nologin
dbmsvc:x:2001:2001::/:/sbin/nologin
clamupdate:x:995:504:Clamav database update user:/var/lib/clamav:/sbin/nologin
```

Default Users WITH RESTRICTED SHELL Access

These users are used for accomplishing a defined task with a restricted shell login. These users should not be removed as the defined task will fail in case these users are deleted.

`dbmmonitor` password is set to a random string during deployment, which must change on first use.

- `dbmmonitor`: The `dbmmonitor` user is used for DBMCLI Utility. For more details refer to Using the DBMCLI Utility

Restricted Shell Users

```
dbmmonitor:x:2003:2003::/home/dbmmonitor:/bin/rbash
```

Default Users with Login Permissions

These privileged users are used for accomplishing most of the tasks in the system. These users should never be altered or deleted as it would have significant impact on the running system.

SSH keys are used for login by customer staff and cloud automation software.

Customer-added SSH keys may be added by the `UpdateVmCluster` method, or by customers directly accessing the customer VM and managing SSH keys inside of the customer VM. Customers are responsible for adding comments to keys to make them identifiable. When a customer adds the SSH key by the `UpdateVmCluster` method, the key is only added to the `authorized_keys` file of the `opc` user.

Cloud automation keys are temporary, specific to a given cloud automation task, for example, VM Cluster Memory resize, and unique. Cloud automation access keys can be identified by the following comments: `OEDA_PUB`, `EXACLOUD_KEY`, `ControlPlane`. Cloud automation keys are removed after the cloud automation task completes so the `authorized_keys` files of the `root`, `opc`, `oracle`, and `grid` accounts should only contain cloud automation keys while the cloud automation actions are running.

Privileged Users

```
root:x:0:0:root:/root:/bin/bash
oracle:x:1001:1001::/home/oracle:/bin/bash
grid:x:1000:1001::/home/grid:/bin/bash
opc:x:2000:2000::/home/opc:/bin/bash
dbmadmin:x:2002:2002::/home/dbmadmin:/bin/bash
```

- `root`: Linux requirement, used sparingly to run local privileged commands. `root` is also used for some processes like Oracle Trace File Analyzer Agent and `ExaWatcher`.
- `grid`: Owns Oracle Grid Infrastructure software installation and runs Grid Infastructure processes.
- `oracle`: Owns Oracle database software installation and runs Oracle Database processes.
- `opc`:
  - Used by Oracle Cloud automation for automation tasks.
  - Has the ability to run certain privileged commands without further authentication (to support automation functions).
  - Runs the local agent, also known as “DCS Agent” that performs lifecycle operations for Oracle Database and Oracle Grid Infastructure software (patching, create database, and so on).
- `dbmadmin`:
  - The `dbmadmin` user is used for Oracle Exadata Database Machine Command-Line Interface (DBMCLI) utility.
  - The `dbmadmin` user should be used to run all services on the database server. For more information, see Using the DBMCLI Utility.

Related Topics

- Using the DBMCLI Utility

Default Security Settings: Customer VM

Learn about the default security settings used with Oracle Exadata Database Service on Exascale Infrastructure instances.

- Custom database deployment with non-default parameters.
  The command `host_access_control` is to configure Exadata security settings:
  - Implementing password aging and complexity policies.
  - Defining account lockout and session timeout policies.
  - Restricting remote root access.
  - Restricting network access to certain accounts.
  - Implementing login warning banner.
- `account-disable`: Disables a user account when certain configured conditions are met.
- `pam-auth`: Various PAM settings for password changes and password authentication.

- `rootssh`: Adjusts the `PermitRootLogin` value in `/etc/ssh/sshd_config`, which permits or denies the `root` user to login through SSH.
  - By default, `PermitRootLogin` is set to `no`.
  - PermitRootLogin=without-password is required for the cloud automation to perform some lifecycle management operations, disabling root login will cause that service functionality to fail.
- `session-limit`: Sets the `* hard maxlogins` parameter in `/etc/security/limits.conf`, which is the maximum number of logins for all users. This limit does not apply to a user with `uid=0`.
  Defaults to `* hard maxlogins 10` and it is the recommended secure value.
- `ssh-macs`: Specifies the available Message Authentication Code (MAC) algorithms.
  - The MAC algorithm is used in protocol version 2 for data integrity protection.
  - Defaults to `hmac-sha1`, `hmac-sha2-256`, `hmac-sha2-512` for both server and client.
  - Secure recommended values: `hmac-sha2-256`, `hmac-sha2-512` for both server and client.
- `password-aging`: Sets or displays the current password aging for interactive user accounts.
  - `-M`: Maximum number of days a password may be used.
  - `-m`: Minimum number of days allowed between password changes.
  - `-W`: Number of days warning given before a password expires.
  - Defaults to `-M 99999`, `-m 0`, `-W 7`
  - `--strict_compliance_only-M 60`, `-m 1`, `-W 7`
  - Secure recommended values: `-M 60`, `-m 1`, `-W 7`

Default Processes on Customer VM

A list of the processes that run by default on the customer VM, also called DOMU, or Guest VM and Guest OS

- **Oracle Exadata Database Service on Exascale Infrastructure VM agent:**
  Cloud agent for handling database lifecycle operations.
  - Runs as `opc` user
  - Process table shows it running as a Java process with `jar` names - `dbcs-agent-VersionNumber-SNAPSHOT.jar` and `dbcs-admin-VersionNumver-SNAPSHOT.jar`.
- **Oracle Trace File Analyzer agent:**
  Oracle Trace File Analyzer (TFA) provides a number of diagnostic tools in a single bundle, making it easy to gather diagnostic information about the Oracle database and clusterware, which in turn helps with problem resolution when dealing with Oracle Support
  - Runs as `root` user
  - Runs as initd demon (`/etc/init.d/init.tfa`)
  - Process tables show a Java application (`oracle.rat.tfa.TFAMain`)
  - Runs as `root` and `exawatch` users.
  - Runs as backgroud script, `ExaWatcher.sh` and all its child process run as a Perl process.

  - Process table shows as multiple Perl applications.ExaWatcher:
- **Database and GI (clusterware):**
  - Runs as dbmsvc and grid users
  - Process table shows following applications:
    - oraagent.bin, apx_* and ams_* as grid user
    - dbrsMain, and Java applications - derbyclient.jar, weblogic.Server as oracle user.
- **Management Server (MS):**
  Part of Exadata image software for managing and monitoring the image functions.
  - Runs as dbmadmin.
  - Process table shows it running as a Java process.
- Guest VM Network Security
- Compliance Requirements

Guest VM Network Security

Table 5-34 Default Port Matrix for Guest VM Services

Type of interface	Name of interface	Port	Process running
Bridge on client VLAN	bondeth0	22	sshd
		1521 Optionally, customers can assign a SCAN listener port (TCP/IP) in the range between 1024 and 8999. Default is 1521.	Oracle TNS listener
		5000	Oracle Trace File Analyzer Collector
		7879	Jetty Management Server
	bondeth0:1	1521 Optionally, customers can assign a SCAN listener port (TCP/IP) in the range between 1024 and 8999. Default is 1521.	Oracle TNS listener
	bondeth0:2	1521 Optionally, customers can assign a SCAN listener port (TCP/IP) in the range between 1024 and 8999. Default is 1521.	Oracle TNS listener
Bridge on backup VLAN	bondeth1	7879	Jetty Management Server

Table 5-34 (Cont.) Default Port Matrix for Guest VM Services

Type of interface	Name of interface	Port	Process running
Oracle Clusterware running on each cluster node communicates through these interfaces.	`clib0/clre0`	1525	Oracle TNS listener
		3260	Synology DSM iSCSI
		5054	Oracle Grid Interprocess Communication
		7879	Jetty Management Server
		Dynamic Port: 9000-65500 Ports are controlled by the configured ephemeral range in the operating system and are dynamic.	System Monitor service (osysmond)
		Dynamic Port: 9000-65500 Ports are controlled by the configured ephemeral range in the operating system and are dynamic.	Cluster Logger service (ologgerd)
	`clib1/clre1`	5054	Oracle Grid Interprocess communication
		7879	Jetty Management Server
Cluster nodes use these interfaces to access storage cells (ASM disks).	`stib0/stre0`	7060	dbcs-admin
		7070	dbcs-agent
	`stib1/stre1`	7060	dbcs-admin
However, the IP/ports 7060/7070 attached to the storage interfaces are used to access DBCS agent from the Control Plane server.		7070	dbcs-agent
Control Plane server to domU	`eth0`	22	sshd
Loopback	`lo`	22	sshd
		2016	Oracle Grid Infrastructure
		6100	Oracle Notification Service (ONS), part of Oracle Grid Infrastructure
		7879	Jetty Management Server
		Dynamic Port 9000-65500	Oracle Trace File Analyzer

Note:

TNS listener opens dynamic ports after initial contact to well known ports (1521, 1525).

Default iptables rules for Guest VM:

The default iptables are setup to ACCEPT connections on input, forward, and output chains.

```
#iptables -L -n -v
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source
destination

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source
destination

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source
destination
```

Compliance Requirements

PII (Personally Identifiable Information) This information is considered confidential and sensitive, and must be protected to prevent unauthorized use of personal information for the purposes of legal regulation, financial liability, and personal reputation.

You must configure a set of explicit rules to prevent Personally Identifiable Information (PII) from being displayed in your data.

The default Application Performance Monitoring rules hide PII in URLs by recognizing monetary values, bank-account numbers, and dates. However, the default rules only catch obvious PII and are not exhaustive. You must evaluate the default rules and further configure rules to ensure correct reporting in your environment and ensure that PII is not displayed in your data.

For more information, see Hide Personally Identifiable Information and Security and Personally Identifiable Information

Backup Retention

When you enable the Automatic Backup feature, the service creates daily incremental backups of the database to Object Storage. The first backup created is a level 0 backup. Then, level 1 backups are created every day until the next weekend. Every weekend, the cycle repeats, starting with a new level 0 backup.

If you choose to enable automatic backups, you can choose one of the following preset retention periods: 7 days, 15 days, 30 days, 45 days, or 60 days. The system automatically deletes your incremental backups at the end of your chosen retention period.

For more information, see Manage Database Backup and Recovery on Oracle Exadata Database Service on Dedicated Infrastructure

Audit Log Retention Period

The OCI Audit service provides records of API operations performed against supported services as a list of log events. By default, Audit service records are retained for 365 days.

For more information, see Audit Log Retention Period

Service Log Retention

Oracle Cloud Infrastructure services, such as API Gateway, Events, Functions, Load Balancing, Object Storage, and VCN Flow Logs emit service logs. Each of these supported services has a Logs resource that allows you to enable or disable logging for that service. By default, Log retention is 1 month, but it can be set until 6 months.

Logs groups can be used to limit access to sensitive logs generated by services using IAM policy. You don't have to rely on complex compartment hierarchies to secure your logs. For example, say the default log group in a single compartment is where you store logs for the entire tenancy. You grant access to the compartment for log administrators with IAM policy as you normally would. However, let's say some projects contain personally identifiable information (PII) and those logs can only be viewed by a select group of log administrators. Log groups allow you to put logs that contain PII into a separate log group, and then use IAM policy to restrict access to all but a few log administrators.

For more information, see Service Logs and Managing Logs and Log Groups

Default Backup Security Configuration

Learn about the backup security policy for

OS/VM backups:

Oracle does a full backup of guest VM weekly and maintains one or more backup copies. These backups are full disk snapshots of the guest VM (local OS filesystems, not ASM disk groups which reside on Exadata storage). This backup is triggered at a preset time every week. The backups are stored in centralized storage by Oracle. Customers can request Oracle to restore the guest VM image from the most recent backup by filing a My Oracle Support (MOS) Service Request (SR). Oracle cannot restore specific files from the image backup. Customers should perform file level backups in the guest VM if they require the ability to perform single-file restore.

Managed DB backups:

- Weekly full backup (level 0)
- Daily rolling incremental backup (level 1) on seven day cycle
- Automatic backups daily at a specific time set during the database deployment creation process

Retention period for backups vary from 30 days (on Object Storage) to 7 days (on local storage)

Encryption:

- Both Object Storage and local storage: All backups to cloud storage are encrypted.
- Object Storage only: All backups to cloud storage are encrypted.

All backups can be configured via CP UI or CP API.

All backups are encrypted with the same master key used for Transparent Data Encryption (TDE) wallet encryption.

Operator Access to Customer System and Customer Data

Only automated tooling is permitted to access VM for lifecycle automation.

One specific use case is when a VM is unable to restart. In this case, customers must provide permission to access the VM for recovery. Details to handle this scenario are described in section "Exception Workflows" of Exadata Cloud Service Security Controls.

Customers control and monitor access to customer services, including network access to their VMs, authentication to access the VM, and authentication to access databases running in the VMs. Oracle controls and monitors access to Oracle-managed infrastructure components. Oracle staff are not authorized to access customer services, including VMs and databases.

Compliance Requirements

PII (Personally Identifiable Information) This information is considered confidential and sensitive, and must be protected to prevent unauthorized use of personal information for the purposes of legal regulation, financial liability, and personal reputation.

You must configure a set of explicit rules to prevent Personally Identifiable Information (PII) from being displayed in your data.

The default Application Performance Monitoring rules hide PII in URLs by recognizing monetary values, bank-account numbers, and dates. However, the default rules only catch obvious PII and are not exhaustive. You must evaluate the default rules and further configure rules to ensure correct reporting in your environment and ensure that PII is not displayed in your data.

For more information, see Hide Personally Identifiable Information and Security and Personally Identifiable Information

Backup Retention

When you enable the Automatic Backup feature, the service creates daily incremental backups of the database to Object Storage. The first backup created is a level 0 backup. Then, level 1 backups are created every day until the next weekend. Every weekend, the cycle repeats, starting with a new level 0 backup.

If you choose to enable automatic backups, you can choose one of the following preset retention periods: 7 days, 15 days, 30 days, 45 days, or 60 days. The system automatically deletes your incremental backups at the end of your chosen retention period.

For more information, see Manage Database Backup and Recovery on Oracle Exadata Database Service on Dedicated Infrastructure

Audit Log Retention Period

The OCI Audit service provides records of API operations performed against supported services as a list of log events. By default, Audit service records are retained for 365 days.

For more information, see Audit Log Retention Period

Service Log Retention

Oracle Cloud Infrastructure services, such as API Gateway, Events, Functions, Load Balancing, Object Storage, and VCN Flow Logs emit service logs. Each of these supported services has a Logs resource that allows you to enable or disable logging for that service. By default, Log retention is 1 month, but it can be set until 6 months.

Logs groups can be used to limit access to sensitive logs generated by services using IAM policy. You don't have to rely on complex compartment hierarchies to secure your logs. For example, say the default log group in a single compartment is where you store logs for the entire tenancy. You grant access to the compartment for log administrators with IAM policy as you normally would. However, let's say some projects contain personally identifiable information (PII) and those logs can only be viewed by a select group of log administrators. Log groups allow you to put logs that contain PII into a separate log group, and then use IAM policy to restrict access to all but a few log administrators.

For more information, see Service Logs and Managing Logs and Log Groups

Break Glass Procedure for Accessing Customer's Guest VM

There are situations where some problems can only be resolved by Oracle logging into the customer guest VM.

Below are situations where customer's guest VM access is require and recommended procedures for accessing guest VM:

1. Situations where the **starter database is not yet created and customer do not have ssh access to their guest VM yet.** An example would be SR opened by customer to troubleshoot why customer is unable to create a starter database. In this situation, customer never had access to guest VM and no database have yet been created and hence no customer data exists in guest VM.

   As per the security policy associated with ExaDB-XS service, Oracle personnel are prohibited to access customer guest VM without customer’s explicit permission. To comply with this policy, Oracle requires to get Customer permission to access guest VM by asking the following question.

   “In order for Oracle to resolve the issue described *in this SR, we need customer’s explicit permission allowing us to login to customer guest VM. By giving us explicit permission to access guest VM, you are confirming that there is no confidential data that is stored in customer guest VM or associated databases and customer security team is authorizing Oracle to have access to customer guest VM in order for Oracle to help fix this issue. Do I have your explicit permission to access guest VM?"*

   After affirmative response by customer, Oracle support staff can login to customer guest VM to resolve the issue.

2. Situations where a **number of databases exist in customer system and customer have access to guest VM but now support needs to login to guest VM to resolve one of the many situations**
   We have encountered (Nodes doesn’t start because of changes on guest VM, eg. Non-existing mounts in fstab, need to run fsck, Hugepage / sysctl conf modification or lvm backup not completed successfully, fstab has wrong entries for non-existing mounts, customer changed the sshd configurations or permissions in /etc/ssh/sshd_config file, etc.) or simply because customer wants Oracle to help resolve the issue they are facing.

   This case is more serious than the first one as there could be some sensitive data in customer guest VM file system or database. In this case, our support staff will be required to ask the customer to open a new explicit SR specifically to get this permission with the following SR title and content.

   As per the security policy associated with ExaDB-XS service, Oracle personnel are prohibited to access customer guest VM without customer’s explicit permission. For Oracle to comply with this policy, We are required to ask you to open a new SR with exact language as shown below granting Oracle an explicit permission to access guest VM.Please note any modification to the language below may delay resolution of your SR.

New SR Title: SR granting Oracle explicit permission to access DomU of ExaDB-C@C with AK serial number AK99999999

New SR Content: We are opening this SR to grant explicit permission to Oracle to access our DomU in order for support to help resolve issue described in SR# 1-xxxxxxxx.

We acknowledge that by providing this permission, we understand that Oracle will have access to ALL FILES in DomU and agree that there are no confidential

files stored in any of the file systems in DomU. In addition, we also agree that customer security team has authorized Oracle to have access to customer DomU

in order to resolve the issue described in the above SR.

After affirmative response by customer in the above SR, Oracle support staff can login to customer guest VM to resolve the issue.

Enabling Additional Security Capabilities

- KMS Integration (HSM keys)
- Using Non-default Encryption Algorithms for TDE Tablespace Encryption

KMS Integration (HSM keys)

Oracle Exadata Cloud Service (ExaCS) has integration with the OCI Vault service to protect data at rest for its databases. Users now have the control to create and manage TDE master keys within the OCI Vault that protect your Exadata databases.

With this feature, users have the option to start using the OCI vault service to store and manage the master encryption keys. The OCI Vault keys used for protecting databases are stored in a highly available,

durable, and managed service. OCI vault integration for ExaCS is only available after Oracle Database 11g release 2 (11.2.0.4).

With OCI Vault integration with ExaDB-D, customers can now:

- Centrally control and manage your TDE master keys
- Have their TDE master keys stored in a highly available, durable and managed service wherein the keys are protected by hardware security modules (HSM) that meet Federal Information Processing Standards (FIPS) 140-2 Security Level 3 security certification.
- Rotate their encryption keys periodically to maintain security compliance and regulatory needs.
- Migrate from Oracle-managed keys to customer-managed keys for their existing databases.
- The Key version will only be assigned to the container database (CDB), and not to its pluggable database (PDB). PDB will be assigned an automatically generated new key version.

Related Topics

- Announcing Customer-Managed Encryption Keys for Oracle Exadata Cloud Service
- Manage Databases on Exadata Cloud Infrastructure

Using Non-default Encryption Algorithms for TDE Tablespace Encryption

In the published Oracle Advanced Security Guide (section Encrypting Columns in Tables the methodology to create a table to encrypt columns using a non-default encryption algorithm ids described.

Troubleshooting Oracle Exadata Database Service on Exascale Infrastructure Systems

These topics cover some common issues you might run into and how to address them.

- Known Issues for Exadata Database Service on Exascale Infrastructure
  General known issues.
- Troubleshooting Oracle Data Guard
  Learn to identify and resolve Oracle Data Guard issues.
- Obtaining Further Assistance

Known Issues for Exadata Database Service on Exascale Infrastructure

General known issues.

Troubleshooting Oracle Data Guard

Learn to identify and resolve Oracle Data Guard issues.

When troubleshooting Oracle Data Guard, you must first determine whether the problem occurs during the Data Guard setup and initialization or during Data Guard operation, when lifecycle commands are entered. The steps to identify and resolve the issues are different, depending on the scenario in which they are used.

There are three lifecycle operations: switchover, failover, and reinstate. The Data Guard broker is used for all of these commands. The broker command line interface (`dgmgrl`) is the main tool used to identify and troubleshoot the issues. Although you can use logfiles to identify root causes, `dgmgrl` is faster and easier to use to check and identify an issue.

Setting up and enabling Data Guard involves multiple steps. Log files are created for each step. If any of the steps fail, review the relevant log file to identify and fix the problem.

- Validation of the primary cloud VM Cluster and database
- Validation of the standby cloud VM Cluster
- Recreating and copying files to the standby database (passwordfile and wallets)
- Creating Data Guard through Network (RMAN Duplicate command)
- Configuring Data Guard broker
- Finalizing the setup

- Troubleshooting Data Guard using logfiles
  The tools used to identify the issue and the locations of relevant logfiles are different, depending on the scenario in which they are used.
- Troubleshooting the Data Guard Setup Process
  Review errors that can occur in the different steps of the Data Guard setup process. While some errors are displayed within the Console, most of the root causes can be found in the logfiles

Troubleshooting Data Guard using logfiles

The tools used to identify the issue and the locations of relevant logfiles are different, depending on the scenario in which they are used.

Use the following procedures to collect relevant log files to investigate issues. If you are unable to resolve the problem after investigating the log files, contact My Oracle Support.

> **Note:**
>
> When preparing collected files for Oracle Support, bundle them into a compressed archive, such as a ZIP file.

NOT_SUPPORTED

On each compute node associated with the Data Guard configuration, gather log files pertaining to the problem you experienced.

- Enablement stage log files (such as those documenting the Create Standby Database operation) and the logs for the corresponding primary or standby system.
- Enablement job ID logfiles. For example: 23.
- Locations of enablement log files by enablement stage and Exadata system (primary or standby).
- Database name logfiles (`db_name` or `db_unique_name`, depending on the file path).

> **Note:**
>
> Check all nodes of the corresponding primary and standby Exadata systems. Commands executed on a system may have been run on any of its nodes.

NOT_SUPPORTED

Data Guard Deployer (`DGdeployer`) is the process that performs the configuration. When configuring the primary database, it creates the `/var/opt/oracle/log/<dbname>/dgdeployer/dgdeployer.log` file.

This log should contain the root cause of a failure to configure the primary database.

NOT_SUPPORTED

- The primary log from the `dbaasapi` command-line utility is: `/var/opt/oracle/log/dbaasapi/db/dg/<job_ID>.log`. Look for entries that contain `dg_api`.
- One standby log from the `dbaasapi` command-line utility is: `/var/opt/oracle/log/dbaasapi/db/dg/<job_ID>.log`. In this log, look for entries that contain `dg_api`.
- The other standby log is: `/var/opt/oracle/log/<dbname>/dgcc/dgcc.log`. This log is the Data Guard configuration log.

NOT_SUPPORTED

- The Oracle Cloud Deployment Engine (ODCE) creates the `/var/opt/oracle/log/<dbname>/ocde/ocde.log` file. This log should contain the cause of a failure to create the standby database.
- The `dbaasapi` command line utility creates the `var/opt/oracle/log/dbaasapi/db/dg/<job_ID>.log` file. Look for entries that contain `dg_api`.
- The Data Guard configuration log file is `/var/opt/oracle/log/<dbname>/dgcc/dgcc.log`.

NOT_SUPPORTED

- `DGdeployer` is the process that performs the configuration. It creates the following `/var/opt/oracle/log/<dbname>/dgdeployer/dgdeployer.log` file. This log should contain the root cause of a failure to configure the standby database.
- The `dbaasapi` command-line utility creates the `/var/opt/oracle/log/dbaasapi/db/dg/<job_ID>.log` file. Look for entries that contain `dg_api`.
- The Data Guard configuration log is `/var/opt/oracle/log/<dbname>/dgcc/dgcc.log`.

NOT_SUPPORTED

`DGdeployer` is the process that performs the configuration. While configuring Data Guard, it creates the `/var/opt/oracle/log/<dbname>/dgdeployer/dgdeployer.log` file. This log should contain the root cause of a failure to configure the primary database.

NOT_SUPPORTED

On each node of the primary and standby sites, gather log files for the related database name (`db_name`).

> **Note:**
>
> Check all nodes on both primary and standby Exadata systems. A lifecycle management operation may impact both primary and standby systems.

NOT_SUPPORTED

- **Database alert log:** `/u02/app/oracle/diag/rdbms/<dbname>/<dbinstance>/trace/alert_<dbinstance>.log`
- **Data Guard Broker log:** `/u02/app/oracle/diag/rdbms/<dbname>/<dbinstance>/trace/drc<dbinstance>.log`
- **Cloud tooling log file for Data Guard:** `/var/opt/oracle/log/<dbname>/odg/odg.log`

Troubleshooting the Data Guard Setup Process

Review errors that can occur in the different steps of the Data Guard setup process. While some errors are displayed within the Console, most of the root causes can be found in the logfiles

NOT_SUPPORTED

The password entered for enabling Data Guard didn't match the primary admin password for the SYS user. This error occurs during the Validate Primary stage of enablement.

NOT_SUPPORTED

The database may not be running. This error occurs during the Validate Primary stage of enablement. Check with `srvctl` and `sql` on the host to verify that the database is up and running on all nodes.

NOT_SUPPORTED

The primary database could not be configured. Invalid Data Guard commands or failed listener reconfiguration can cause this error.

NOT_SUPPORTED

The TDE wallet could not be created. The Oracle Transparent Database Encryption (TDE) keystore (wallet) files could not be prepared for transportation to the standby site. This error occurs during the create TDE Wallet stage of enablement. Either of the following items can cause failure at this stage:

- The TDE wallet files could not be accessed
- The enablement commands could not create an archive containing the wallet files

Troubleshooting procedure:

1. Ensure that the cluster is accessible. To check the status of a cluster, run the following command:

   ```
   crsctl check cluster -all
   ```

2. If the cluster is down, run the following command to restart it:

   ```
   crsctl start crs -wait
   ```

3. If this error occurs when the cluster is accessible, check the logs for create TDE Wallet (enablement stage) to determine cause and resolution for the error.

NOT_SUPPORTED

The archive containing the TDE wallet was likely not transmitted to the standby site. Retrying usually solves the problem.

NOT_SUPPORTED

- The primary and standby sites may not be able to communicate with each other to configure the standby database. These errors occur during the configure standby database stage of enablement. In this stage, configurations are performed on the standby database, including the rman duplicate of the primary database. To resolve this issue:

1. Verify the connectivity status for the primary and standby sites.
2. Ensure that the host can communicate from port 1521 to all ports. Check the network setup, including Network Security Groups (NSGs), Network Security Lists, and the remote VCN peering setup (if applicable). The best way to test communication between the host and other nodes is to access the databases using SQL*PLUS from the primary to standby and from the standby to the primary.

- The SCAN VIPs or listeners may not be running. Use the test above to help identify the issue.

NOT_SUPPORTED

Possible causes:

- SCAN VIPs or listeners may not be running. You can confirm this issue by using the following commands on any cluster node.

  -
    ```
    [grid@exa1-****** ~]$ srvctl status
                scan
    ```

  -
    ```
    [grid@exa1-****** ~]$ srvctl status
                  scan_listener
    ```
- Databases may not be reachable. You can confirm this issue by attempting to connect using an existing Oracle Net alias.

Troubleshooting procedure:

1. As the oracle OS user, check for the existence of an Oracle Net alias for the container database (CDB). Look for an alias in $ORACLE_HOME/network/admin/<dbname>/tnsnames.ora.
   The following example shows an entry for a container database named db12c:

   ```
   cat $ORACLE_HOME/network/admin/db12c/tnsnames.ora
   DB12C = (DESCRIPTION =(ADDRESS = (PROTOCOL = TCP)(HOST = exa1-*****-
   scan.********.******.******.com)(PORT = 1521))
   (CONNECT_DATA = (SERVER = DEDICATED) (SERVICE_NAME =
   db12c.********.******.******.com)
   (FAILOVER_MODE = (TYPE = select) (METHOD = basic))))
   ```

2. Verify that you can use the alias to connect to the database. For example, as sysdba, enter the following command:

   ```
   sqlplus sys@db12c
   ```

NOT_SUPPORTED

A possible cause for this error is that the Oracle Database sys or system user passwords for the database and the TDE wallet may not be the same. To compare the passwords:

1. Connect to the database as the **sys user** and check the TDE status in

   ```
   V$ENCRYPTION_WALLET
   ```

   .

2. Connect to the database as the **system user** and check the TDE status in

   ```
   V$ENCRYPTION_WALLET
   ```

   .

3. Update the applicable passwords to match. Log on to the system host as **opc** and run the following commands:

   a. To change the SYS password:

      ```
      sudo dbaascli database changepassword --dbname <database_name>
      ```

   b. To change the TDE wallet password:

      ```
      sudo dbaascli tde changepassword --dbname <database_name>
      ```

NOT_SUPPORTED

When the switchover, failover, and reinstate commands are run, multiple error messages may occur. Refer to the Oracle Database documentation for these error messages.

Note

Oracle recommends using the Data Guard broker command line interface (dgmgrl) to validate the configurations.

1. As the Oracle User, connect to the primary or standby database with `dgmgrl` and verify the configuration and the database:

   ```
   dgmgrl sys/<pwd>@<database>
   DGMGRL> VALIDATE CONFIGURATION VERBOSE
   DGMGRL> VALIDATE DATABASE VERBOSE <PRIMARY>
   DGMGRL> VALIDATE DATABASE VERBOSE <STANDBY>
   ```

2. Consult the Oracle Database documentation to check for the respective error message. For example:

   - **ORA-16766:** Redo apply is stopped.
   - **ORA-16853**: Apply lag has exceeded specified threshold.
   - **ORA-16664**: Unable to receive the result from a member (under the standby database).
   - **ORA-12541**: TNS: no listener (under the primary database)

Obtaining Further Assistance

If you were unable to resolve the problem using the information in this topic, follow the procedures below to collect relevant database and diagnostic information. After you have collected this information, contact Oracle Support.

- Collecting Cloud Tooling Logs
  Use the relevant log files that could assist Oracle Support for further investigation and resolution of a given issue.
- Collecting Oracle Diagnostics

Related Topics

- Oracle Support

Collecting Cloud Tooling Logs

Use the relevant log files that could assist Oracle Support for further investigation and resolution of a given issue.

DBAASCLI Logs

`/var/opt/oracle/log/dbaascli`

- `dbaascli.log`

Collecting Oracle Diagnostics

To collect the relevant Oracle diagnostic information and logs, run the `dbaascli diag collect` command.

For more information about the usage of this utility, see *DBAAS Tooling: Using dbaascli to Collect Cloud Tooling Logs and Perform a Cloud Tooling Health Check*.

Related Topics

- DBAAS Tooling: Using dbaascli to Collect Cloud Tooling Logs and Perform a Cloud Tooling Health Check