Oracle® Cloud

Using Oracle Cloud Infrastructure Storage Software Appliance

Release 16.3.1.4
E63845-19
June 2020

Oracle Cloud Using Oracle Cloud Infrastructure Storage Software Appliance, Release 16.3.1.4

E63845-19

Primary Author: Oracle Corporation

Contents

Preface

Using Oracle Infrastructure Storage Software Appliance describes how to install Oracle Cloud Infrastructure Storage Software Appliance and manage content in the cloud.

Topics:

Audience

Using Oracle Infrastructure Storage Software Appliance is intended for administrators and users who want to install Oracle Cloud Infrastructure Storage Software Appliance and manage content in the cloud.

Related Resources

For more information, see *Using Oracle Cloud Infrastructure Object Storage Classic*.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1

Getting Started with Storage Software Appliance

Oracle Cloud Infrastructure Storage Software Appliance, on-premise distribution, is a cloud storage gateway that you can install on premises and then use to easily connect your on-premises applications and workflows to your Oracle Cloud Infrastructure Object Storage Classic instance in the cloud.

> **Note:**
>
> If you are using Oracle Cloud Infrastructure Object Storage Service instances in the cloud, use Oracle Cloud Infrastructure Storage Gateway. See Overview of Storage Gateway in *Oracle Cloud Infrastructure Documentation*.

Topics:

About Oracle Cloud Infrastructure Storage Software Appliance

Oracle Cloud Infrastructure Storage Software Appliance is a cloud storage gateway that you can install and then use to easily connect your on-premises applications and workflows to your service instance. It provides the following benefits:

- Applications can interact with your service instance over a secure HTTPS connection using the REST API or the Java SDK. For legacy applications and environments that can't use the REST API or Java SDK, Oracle Cloud Infrastructure Storage Software Appliance provides POSIX-compliant file access to the account or tenancy in your service instance using the NFSv4 protocol.

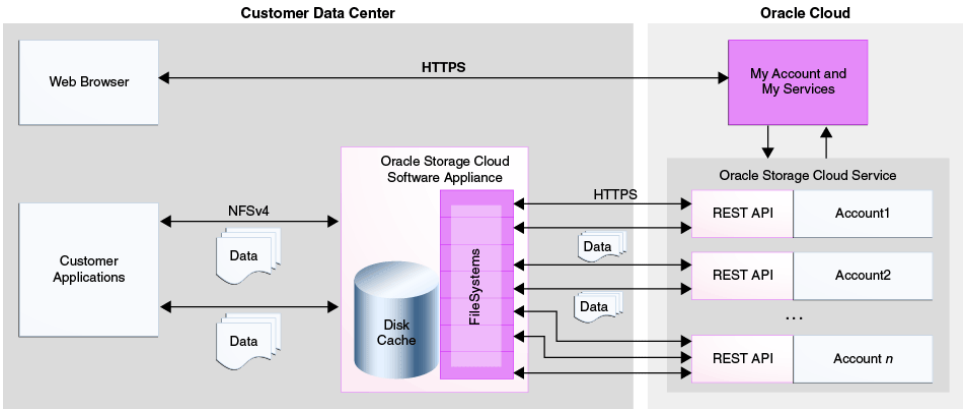
> **Note:**
>
> The appliance supports NFS v4 I/O in 2 modes: asynchronous and POSIX Sync. For more information about the modes, see the field **Sync Policy** in the table in Step 9 in Adding a FileSystem.
> Use the appliance as a data mover, to transfer files to and from your service instance. The appliance is not a replacement for general purpose network attached storage (NAS), though it behaves similarly to NAS.
>
> Content repositories and storing secondary copies of data are ideal use cases for the appliance. Don't run applications and executables directly from the appliance mount points, particularly if the appliance cache is not large enough for all the files that the applications will access. Applications typically create temporary files and modify them often, affecting the operational efficiency of the appliance.

- Files in your service instance are stored as objects in a flat namespace within a container or bucket. However, enterprise applications typically work with files in nested directories. Oracle Cloud Infrastructure Storage Software Appliance transparently handles the storage of files as objects in your account or tenancy.

  Your applications can store files in and retrieve files from your account or tenancy through *filesystems* that you create in Oracle Cloud Infrastructure Storage Software Appliance and mount on your local host. A filesystem in this context represents a mapping between a directory on a local host and a container or bucket in your service instance, depending on the storage service. It defines the connection credentials that Oracle Cloud Infrastructure Storage Software Appliance must use to connect to an account or tenancy.
- To ensure that your data is secure, you can configure Oracle Cloud Infrastructure Storage Software Appliance to encrypt files when they’re being stored and decrypt them when they have to be retrieved.
- Oracle Cloud Infrastructure Storage Software Appliance caches frequently retrieved data on the local host, minimizing the number of REST API calls to your service instance and enabling low-latency, high-throughput file I/O.

Oracle Cloud Infrastructure Storage Software Appliance doesn’t support third-party storage cloud services.

The following figure shows the flow of data between your on-premises applications and Oracle Cloud Infrastructure Object Storage Classic through Oracle Cloud Infrastructure Storage Software Appliance.



Caution:

Don't use the REST API, Java library, or any other client to retrieve, create, update, or delete objects in a container or bucket that's mapped to a filesystem in Oracle Cloud Infrastructure Storage Software Appliance. Doing so will cause the data in the appliance to become inconsistent with data in your storage service. You can't recover from this inconsistency.

To prevent unauthorized users from retrieving, creating, updating, or deleting objects in a container or bucket that's connected to a filesystem in Oracle Cloud Infrastructure Storage Software Appliance, define custom roles, assign them to the appropriate container, assign the roles to only the users that should have access to the container, and specify only one of these users when defining the filesystem to be connected to the container or bucket.

Oracle Cloud Infrastructure Storage Software Appliance automatically backs up and stores all of its important configuration data in your service instance. If the host on which Oracle Cloud Infrastructure Storage Software Appliance is installed crashes, another instance of the appliance can be started quickly using the stored configuration data.

Features of Storage Software Appliance

Oracle Cloud Infrastructure Storage Software Appliance facilitates easy, secure, reliable data storage and retrieval from your service instance.

The following are the features of the appliance:

- POSIX-Compliant NFS Access to Oracle Cloud Infrastructure Object Storage Classic and Oracle Cloud Infrastructure Object Storage
- Granular Encryption to Enable Data Security and Storage Efficiency
- End-to-end Data Integrity with Checksum Verification
- Efficient Handing of Large Files
- Support for Data Archival
- Support for File Versions Compaction and End-to-End Delete

- Quick Access to Select Files with Cache Pinning
- Appliance Health Check

POSIX-Compliant NFS Access to Oracle Cloud Infrastructure Object Storage Classic and Oracle Cloud Infrastructure Object Storage

Using Oracle Cloud Infrastructure Storage Software Appliance, your applications can interact with Oracle Cloud Infrastructure Object Storage Classic through standard protocols, without invoking direct REST API calls to the services. The appliance is compliant with POSIX standards. You can create multiple NFS shares within a single appliance instance. Using a single appliance instance, you can connect to multiple containers. The files are copied to the appliance filesystem by using the NFSv4 protocol. The appliance supports NFS in asynchronous and POSIX-Sync modes. The appliance stores the files as objects in Oracle Cloud Infrastructure Object Storage Classic by using the HTTPS protocol.

Using Oracle Cloud Infrastructure Storage Software Appliance, your applications can now also interact with Oracle Cloud Infrastructure Object Storage through standard protocols. You can connect to multiple buckets using the appliance instance, and copy the files to the appliance filesystems. The appliance stores the files as objects in the Oracle Cloud Infrastructure Object Storage tenancy and performs multipart uploads for large objects.

> **Caution:**
>
> Don’t use the REST API, Java library, or any other client to retrieve, create, update, or delete objects in a container or bucket that’s mapped to a filesystem in Oracle Cloud Infrastructure Storage Software Appliance. Doing so will cause the data in the appliance to become inconsistent with data in your storage service. You can’t recover from this inconsistency.
>
> To prevent unauthorized users from retrieving, creating, updating, or deleting objects in a container or bucket that’s connected to a filesystem in Oracle Cloud Infrastructure Storage Software Appliance, define custom roles, assign them to the appropriate container, assign the roles to only the users that should have access to the container, and specify only one of these users when defining the filesystem to be connected to the container or bucket.

Granular Encryption to Enable Data Security and Storage Efficiency

Oracle Cloud Infrastructure Storage Software Appliance stores data securely in your account or tenancy. The appliance transfers data using HTTPS which encrypts data packets in flight between the appliance and the cloud. The appliance also provides the option to enable encryption of data during upload to Oracle Cloud Infrastructure Object Storage Classic, so that it remains encrypted while at rest in the cloud. Data written to Oracle Cloud Infrastructure Object Storage is always automatically encrypted in the cloud. To ensure data security, you can configure the appliance to encrypt data on premises before the data is stored in your account or tenancy, and decrypt files when they are retrieved. You can update the encryption keys at any point in time. By having granular control that enables encryption at the NFS share level, you can ensure that only sensitive data is being encrypted. This minimizes the performance cost associated with encryption. Encryption is supported at the filesystem level. You can configure encryption for each configured filesystem, which ensures that sensitive

data is secured in your account or tenancy. By using more granular controls, you can increase storage efficiency.

End-to-end Data Integrity with Checksum Verification

The built-in data integrity checks ensure that data is validated as it moves through the data path, from Oracle Cloud Infrastructure Storage Software Appliance, to your account or tenancy, enabling seamless end-to-end data integrity. Checksum verification helps in ensuring the data integrity. Metadata integrity checks are performed to ensure that the metadata is in consistent state. The checksum for each file can be read using a custom interface or API.

Efficient Handing of Large Files

Oracle Cloud Infrastructure Storage Software Appliance supports large files that exceed the maximum size allowed by Oracle Cloud Infrastructure Object Storage Classic. Large files are sliced into 1 GB segments and each segment is stored as a separate object. The metadata database maintains the manifest of the segments that comprise a given object, so that multi-segment files can be reconstructed automatically when read through the appliance. The segments are uploaded sequentially.

Oracle Cloud Infrastructure Storage Software Appliance uploads large files (with file size higher than 128 MB) as multipart upload objects in Oracle Cloud Infrastructure accounts.

Support for Data Archival

Oracle Cloud Infrastructure Storage Software Appliance supports uploading and restoring objects in containers of the `Archive` storage class.

In metered accounts, you can create containers of two *storage classes*, `Standard` (default) and `Archive`. You can use `Archive` containers to store large data sets that you don't need to access frequently, at a fraction of the cost of storing data in `Standard` containers. Note that to download data stored in `Archive` containers, you must first *restore* the objects. The restoration process can take up to four hours depending on the size of the object. A few features, such as bulk upload and deletion are not supported for `Archive` containers. `Archive` containers are ideal for storing data such as email archives, data backups, and digital video masters. For information about the pricing and other terms for the `Archive` storage class in Oracle Cloud Infrastructure Object Storage Classic, go to https://cloud.oracle.com/storage-classic?tabID=1406491833493.

For information about the pricing and other terms for the `Archive` storage class in Oracle Cloud Infrastructure Object Storage, go to https://cloud.oracle.com/en_US/infrastructure/storage.

Support for File Versions Compaction and End-to-End Delete

Oracle Cloud Infrastructure Storage Software Appliance supports deletion of old file versions from your service instance.

Oracle Cloud Infrastructure Storage Software Appliance provides a traditional file system interface for the storage services. It allows file operations with byte-level granularity, such as append, re-write, over-write, and truncate. When a file is modified in an appliance filesystem, it results in a new version of the file being created and uploaded to the service instance.

When a file that contains multiple versions exists, the latest or most recent version of the file will always be returned when the file is read. The administrator can configure the number of versions of a file that will be retained in your service instance. File Version Compaction allows the permanent deletion of unwanted versions. Also, if a file is deleted from the filesystem, then the corresponding object(s) in the service instance will also be deleted, if file version compaction is enabled in the appliance.

Quick Access to Select Files with Cache Pinning

Oracle Cloud Infrastructure Storage Software Appliance allows you to pin select files to the filesystem cache for quick access. You can pin files to the cache for filesystems connected to any storage class, `Standard` or `Archive`.

When you upload a file to your filesystem, it's initially stored in the filesystem cache, and then uploaded to the container or bucket, depending on the storage service. After the file has been uploaded, it may get removed from the filesystem cache by the cache manager. The cache is reclaimed using the *Least Recently Used* (LRU) cache management policy to meet the cache threshold that's specified in the filesystem advanced settings. If you want specific files to be always available in the cache for quick access, you can preserve them in the filesystem cache by pinning them to the cache. Once pinned, the files are not removed from the filesystem cache, except if you specifically unpin them.

Appliance Health Check

The appliance health check service is an automated process run on Oracle Cloud Infrastructure Storage Software Appliance. You can monitor the overall system status through the health check and get insights on the appliance performance like local storage usage.

Terminology

The following table defines the key terms used in the context of Oracle Cloud Infrastructure Storage Software Appliance.

Term	Description
Account	An **Oracle Account** is a unique customer account and can correspond to an individual, an organization, or a company that is an Oracle customer. Each account has one or more identity domains.
Bucket	A **bucket** is a user-created resource in Oracle Cloud Infrastructure. It can hold objects in a compartment within a namespace. A bucket is associated with a single compartment.
Compartment	A **compartment** is a collection of related resources (such as instances, virtual cloud networks, block volumes) that can be accessed only by certain groups authorized by an administrator. A compartment allows you to organize and control access to your cloud resources.
Container	A **container** is a user-created resource in Oracle Cloud Infrastructure Object Storage Classic. It can hold an unlimited number of objects, unless you specify a quota for the container. Note that containers cannot be nested.

Term	Description
FileSystem (or filesystem)	A **FileSystem** in Oracle Cloud Infrastructure Storage Software Appliance connects a directory on a local host to a container or bucket in your service instance, depending on the storage service. Generally, *file system* (two words) means the mechanism that operating systems use to manage files on disks. This general meaning is distinct from the meaning of *filesystem* (one word) in the context of Oracle Cloud Infrastructure Storage Software Appliance.
Metadata	Metadata refers to information that is specific to a given file or object. Examples include: filename, object id, creation date, modification date, size, permissions. The appliance caches all metadata for the filesystem locally, as well as backs it up to the cloud periodically.
NFS v4	**NFS v4** is version 4 of NFS (network file system), a distributed file system protocol defined in RFC 3530 (https://www.ietf.org/rfc/rfc3530.txt). It enables client computers to mount file systems that exist on remote servers and access those remote file systems over the network as though they were local file systems.
Object storage	**Object storage** provides an optimal blend of performance, scalability, and manageability when storing large amounts of unstructured data. Multiple storage nodes form a single, shared, horizontally scalable pool in which data is stored as objects (blobs of data) in a flat hierarchy of containers. Each object stores data, the associated metadata, and a unique ID. You can assign custom metadata to containers and objects, making it easier to find, analyze, and manage data. Applications use the unique object IDs to access data directly via REST API calls. Object storage is simple to use, performs well, and scales to a virtually unlimited capacity.
Oracle Cloud Infrastructure Object Storage Classic	**Oracle Cloud Infrastructure Object Storage Classic** provides a low cost, reliable, secure, and scalable object-storage solution for storing unstructured data and accessing it anytime from anywhere. It is ideal for data backup, archival, file sharing, and storing large amounts of unstructured data like logs, sensor-generated data, and VM images.
Oracle Cloud Infrastructure	Oracle Cloud Infrastructure is a set of complementary cloud services that enable you to build and run a wide range of applications and services in a highly-available hosted environment. It offers high-performance compute capabilities (as physical hardware instances) and storage capacity in a flexible overlay virtual network that is securely accessible from your on-premises network.
Tenancy	A **tenancy** is a secure and isolated partition within Oracle Cloud Infrastructure where you can create, organize, and administer your cloud resources.

System Requirements for Installing Storage Software Appliance

This section provides details about the hardware and software that are required to install Oracle Cloud Infrastructure Storage Software Appliance.

Hardware Requirements

A server with:

- Two dual-core CPUs (4-core CPUs recommended)
- Recommended disk size: 300 GB
- Minimum memory requirements (based on the maximum number of files that can be uploaded to the appliance filesystem):
  - 16 GB for filesystems up to 1 million files
  - 32 GB for filesystems up to 5 million files
  - 64 GB for filesystems up to 10 million files

Software Requirements

- Oracle Linux 7 with UEK Release 4 or later
- Docker 1.12.6
  Docker is an open platform for building, shipping and running distributed applications. For more information, see https://www.docker.com/.
- NFS version 4.0

> **Note:**
>
> - Docker and NFS protocol are installed automatically during the appliance installation on a host running on Oracle Linux.
> - After installing the appliance, ensure that the storage driver in Docker is `devicemapper`. See Verifying and Updating the Storage Driver in Docker.

Booting the Linux Host Using the UEK4 Kernel

1. Edit `/etc/yum.repos.d/public-yum-ol7.repo` on the host on which you want to install the appliance:
   a. Look for the `[ol7_UEKR3]` section:
      - `[ol7_UEKR3] name=Latest Unbreakable Enterprise Kernel Release 3 for Oracle Linux $releasever ($basearch)`
      - `baseurl=http://public-yum.oracle.com/repo/OracleLinux/OL7/UEKR3/$basearch/`
      - `gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-oracle`
      - `gpgcheck=1`
      - `enabled=1`
   b. If the `[ol7_UEKR3]` section doesn't exist, then add the following section:
      - `[ol7_UEKR4] name=Latest Unbreakable Enterprise Kernel Release 4 for Oracle Linux $releasever ($basearch)`
      - `baseurl=http://public-yum.oracle.com/repo/OracleLinux/OL7/UEKR4/$basearch/`
      - `gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-oracle`
      - `gpgcheck=1`

- `enabled=1`

If the `[ol7_UEKR3]` section exists, then replace it with the `[ol7_UEKR4]` section.

2. List all the 3.8 kernels by entering the following command:
   `rpm -qa | grep kernel`
3. Remove all the 3.8 kernels from the host on which you want to install the appliance by entering the following command:
   `sudo rpm -e kernel`

> **Note:**
>
> You might get errors about the packages that the 3.8 kernels might depend on. If you see such errors, then remove the packages in addition to the 3.8 kernels.

4. (Optional) Update to the latest kernel:
   `sudo yum update`
5. Generate a new `grub.cfg` file:
   `sudo grub2-mkconfig > grub.cfg`
6. Move the updated `grub.cfg` file to the `grub2` directory:
   `sudo mv grub.cfg /boot/grub2`

Before you install Oracle Cloud Infrastructure Storage Software Appliance, you must install `docker` and `nfs-utils` package. For more information, see Installing Storage Software Appliance.

Verifying and Updating the Storage Driver in Docker

1. Start `docker`:
   `sudo systemctl start docker`
2. Verify the information in `docker`:
   `sudo docker info`

   Sample output

```
Containers: 0
Images: 0
Storage Driver: devicemapper
Backing Filesystem: extfs
Server Version: 17.03.1-ce
Storage Driver: devicemapper
Pool Name: docker-202:1-8413957-pool
Pool Blocksize: 65.54 kB
Base Device Size: 10.74 GB
Backing Filesystem: xfs
Data file: /dev/loop0
Metadata file: /dev/loop1
Data Space Used: 11.8 MB
Data Space Total: 107.4 GB
Data Space Available: 7.44 GB
Metadata Space Used: 581.6 kB
Metadata Space Total: 2.147 GB
Metadata Space Available: 2.147 GB
Thin Pool Minimum Free Space: 10.74 GB
Udev Sync Supported: true
```

```
Deferred Removal Enabled: false
Deferred Deletion Enabled: false
Deferred Deleted Device Count: 0
Data loop file: /var/lib/docker/devicemapper/data
Metadata loop file: /var/lib/docker/devicemapper/metadata
Library Version: 1.02.135-RHEL7 (2016-11-16)
```

3. Look for `Storage Driver` in the output.
   If `Storage Driver` is *not* `devicemapper`, then perform the following steps:

   a. Stop `docker`:
   `sudo systemctl stop docker`

   b. Look for `/etc/docker/daemon.json` in the host.
   If the file `daemon.json` does not exist, then create it.

   c. Add the following text in `daemon.json` to set the variable `storage-driver` to `devicemapper` :

   ```
   {
   "storage-driver": "devicemapper"
   }
   ```

   > **Note:**
   >
   > Also, see Use the Device Mapper storage driver and update the `docker` configuration, if required.

   d. Restart `docker`:
   `sudo systemctl start docker`

   e. Verify the information in `docker`:
   `sudo docker info`

   Look for `Storage Driver` in the output and verify that the storage driver is `devicemapper`.

Installing Storage Software Appliance

Before You Begin

- Fulfill the system requirements. See System Requirements for Installing Storage Software Appliance.
- Read Best Practices for Using Storage Software Appliance.
- Make sure that you have a subscription to Oracle Cloud Infrastructure Object Storage Classic or Oracle Cloud Infrastructure Object Storage.
- Go to http://www.oracle.com/technetwork/topics/cloud/downloads/index.html, look for the appliance installer in the Oracle Cloud Infrastructure Object Storage Classic section, and download the installer.

Steps for Installing Oracle Cloud Infrastructure Storage Software Appliance

> **Note:**
>
> If the server is running on Oracle Linux, go to Step 11 to install the appliance. Docker and NFS protocol are installed automatically on the server.

1. Log in to the server on which you want to install Oracle Cloud Infrastructure Storage Software Appliance.
2. Edit `/etc/yum.repos.d/public-yum-ol7.repo` on the host on which you want to install the appliance.
   - Change the value of `enable` to `1` in `addons`.
   - Change the value of `enable` to `1` in `optional_latest`.
3. (Optional) Install `docker` using `yum`:
   ```
   sudo yum install docker-engine
   ```
   Run this command only if you have to install `docker` manually.
4. Restart the server:
   ```
   sudo shutdown -r now
   ```
5. Enable non-root users to use `docker` client.
   a. Add a new group (`docker`) to the host:
      ```
      sudo groupadd docker
      ```
   b. Modify your user account and add your user name to the `docker` group:
      ```
      sudo usermod -a –G docker username
      ```
   c. Log out and log in again.
6. Run the following commands to set up the appliance host to work around the `docker` socket:
   ```
   NO_PROXY=localhost,127.0.0.1/8,/var/run/docker.sock

   no_proxy=$NO_PROXY
   ```
7. Start `docker` and ensure that `docker` is running by entering the following commands:
   - `sudo systemctl start docker`
   - `sudo systemctl enable docker`
8. (Optional) Install `nfs-utils` using `yum`:
   ```
   sudo yum install nfs-utils
   ```
   Run this command only if you have to install `nfs-utils` manually.

   The `nfs-utils` package enables NFS protocol on your host.
9. Start `rpcbind` and `nfs-server` and ensure the NFS protocol is running by entering the following commands:
   - `sudo systemctl start rpcbind`
   - `sudo systemctl start nfs-server`

- `sudo systemctl enable rpcbind`
- `sudo systemctl enable nfs-server`

10. Check if the NFS protocol version is 4:
    `sudo rpcinfo -p | grep nfs`
11. Extract the files from the installer to a directory on the appliance host by entering the command:
    `tar -xvf installer_tar.gz`
12. Change the permission of the `oscsa-install.sh` file:
    `chmod +x oscsa-install.sh`
13. Check the status of `docker`:
    `sudo systemctl status docker`
14. Verify the information in `docker`:
    `sudo docker info`
15. Run the `oscsa-install.sh` file:
    `sudo ./oscsa-install.sh`

    The default installation location on the appliance host is `/opt/oscsa_gateway`.

    The appliance creates local copies of the files in the local buffer until the files are copied to your account. By default, the location of the buffer is in a subdirectory of `/var/lib`. However, if there isn't sufficient storage space in `/var/lib`, then use this option and specify a location for data storage, metadata, and log storage.

    (Optional) Alternatively, you can run the `oscsa-install.sh` file with any of the following options:

    - `-a`: To run the installation in advanced mode.
    - `-p`: To specify that Oracle Cloud Infrastructure Storage Software Appliance is running with a proxy server.
    - `-q`: To run the installation in quiet mode.
    - `-d`: To install at the specified installation path.
    - `-h`: To display the help information.

    Sample output

```
**********************************
Imported temporary env vars from opc to this install session
**********************************
Checking that docker is installed and using the correct version
Pass found docker version Docker version 17.06.2-ol, build d02b7ab
WARNING: devicemapper: usage of loopback devices is strongly discouraged for
production use.
         Use '--storage-opt dm.thinpooldev' to specify a custom block
storage device.
WARNING: devicemapper: usage of loopback devices is strongly discouraged for
production use.
         Use '--storage-opt dm.thinpooldev' to specify a custom block
storage device.

**********************************
Checking host prerequisites
**********************************

Detected linux operating system
```

```
Checking kernel version
Pass kernel version 4.1.12-61.1.18.el7uek.x86_64 found
Checking NFS version
Pass found NFS version 4

************************************
All prerequisites have been met
************************************


************************************
Begin installation
************************************

Enter the install location press enter for default (/opt/oscsa_gateway/) :
Installing to destination /opt/oscsa_gateway/
Copied install scripts
Copied OCISSA image
Starting configuration script

Enter the path for OCISSA Cache storage : /oscsa/cache

Enter the path for OCISSA Metadata storage : /oscsa/metadata

Enter the path for OCISSA Log storage : /oscsa/log
Writing configuration
Importing image
d0c367ad7015: Loading layer
[==================================================>]  233.7MB/233.7MB
1bef79313f6a: Loading layer
[==================================================>]  395.6MB/395.6MB
50affcccd4ca: Loading layer
[==================================================>]  3.072kB/3.072kB
02651725b81e: Loading layer
[==================================================>]  3.072kB/3.072kB
9d65e9622255: Loading layer
[==================================================>]  62.21MB/62.21MB
...
Loaded image: oscsa_gw:1.4
Loaded image: oraclelinux:7.3
Creating OCISSA Volume with args -v /oscsa/cache:/usr/share/oracle/ -v /
oscsa/metadata:/usr/share/oracle/system/ -v /oscsa/log:/var/log/gateway
Applying configuration file to container
Starting OCISSA [oscsa_gw:1.4]
Setting up config file port with nfs
Setting up config file port with rest
Management Console: https://prd-linux-srv5:443
If you have already configured an OCISSA FileSystem via the Management
Console,
you can access the NFS share using the following port.

NFS Port: 32769
Example: mount -t nfs -o vers=4,port=32769 appliance_host_name:/OCISSA
FileSystem name/local_mount_point
```

In the advanced setup, you can configure the following additional parameters:

- `DATASTORAGE`: Specifies the disk location or mount path where the Oracle Cloud Infrastructure Storage Software Appliance disk cache will be stored.

- `MDSTORAGE`: Specifies the disk location or mount path where the Oracle Cloud Infrastructure Storage Software Appliance metadata will be stored.
- `LOGSTORAGE`: Specifies the disk location or mount path where the Oracle Cloud Infrastructure Storage Software Appliance logs will be stored.
- `ADMINPORT`: Specifies the port on the appliance host to access the management console.
- `NFSPORT`: Specifies the port on the appliance host to access NFS.

> **Note:**
>
> During the appliance installation, the directories for storing data, metadata and log storage are automatically created if they don't exist

16. Check the status of the firewall:
    `sudo firewall-cmd --state`
17. Open the firewall ports on the appliance host by entering the following commands:
    - `sudo firewall-cmd --zone=public --add-port=`*`nfs_port`*`/tcp --permanent`
    - `sudo firewall-cmd --zone=public --add-port=`*`management_web_ui_port`*`/tcp --permanent`
    - `sudo firewall-cmd --reload`

    Example:

    ```
    sudo firewall-cmd --zone=public --add-port= 32774/tcp --permanent

    sudo firewall-cmd --zone=public --add-port= 32775/tcp --permanent

    sudo firewall-cmd --zone=public --add-port= 32776/tcp --permanent

    sudo firewall-cmd --reload
    ```
18. Start the appliance:
    `oscsa up`

The installation might take up to 30 minutes to run, download images, and import the images into the `docker`.

When the installation is complete, you can see the details of Oracle Cloud Infrastructure Storage Software Appliance as shown in the following sample output.

Make a note of the management console URL, which has the following format:

`https://`*`appliance_host_name`*`:`*`port_number`*

Sample output

```
Starting NFS on docker host
Applying configuration file to container
Starting OSCSA [oscsa_gw:1.4]

Management Console: https://myApplianceHost.example.com:32771


If you have already configured an OCISSA FileSystem via the Management Console,
you can access the NFS share using the following port.
```

```
NFS Port: 32770

Example: mount -t nfs -o vers=4,port=32770 myApplianceHost.example.com:/
OCISSA_filesystem_name /local_mount_point
```

In the sample output,

- *myApplianceHost.example.com* is the appliance host name
- *32771* is the management console port number

> **Tip:**
>
> You can use either the appliance host name or the IP address in the management console URL.
>
> - You can find the name of the appliance host by entering the following command:
>   `hostname`
> - You can find the IP address of the appliance host by entering the following command:
>   `ip addr`

The management console is a web application running within Oracle Cloud Infrastructure Storage Software Appliance. You can open the management console by using a web browser. Then you can create your administrator and log in to the management console to create your first filesystem. For more information, see Creating Your First FileSystem.

Creating Your First FileSystem

When you access the management console for the first time, the management console invokes a wizard prompting you to create the administrator credentials and your first filesystem. A filesystem is similar to a namespace containing a set of data that's accessible through Oracle Cloud Infrastructure Storage Software Appliance. A filesystem in this context represents a mapping between a directory on a local host and a container in your account or bucket in your tenancy, depending on the storage service. The filesystem defines the connection credentials that Oracle Cloud Infrastructure Storage Software Appliance must use to connect to an account or tenancy.

Using a filesystem in the appliance, you can store and retrieve files from a directory on the appliance host (or any host from where you want to upload data) to a container or bucket.

1. Log in to your host.
2. Check if Oracle Cloud Infrastructure Storage Software Appliance is running by entering the following command:
   `sudo docker ps`

   Sample output:

   ```
   CONTAINER ID
   a4e254d80472
   IMAGE
   ```

```
oscsa_gw:1.3
COMMAND
"/sbin/init"
CREATED 38 minutes ago
STATUS
Up 38 minutes
PORTS
0.0.0.0:32775->111/udp, 0.0.0.0:32782->2049/tcp, 0.0.0.0:32783->3333/tcp
NAMES
oscsa_gw
```

3. If Oracle Cloud Infrastructure Storage Software Appliance isn't running in `docker`, then start the appliance by entering the following command:
   `oscsa up`

   Sample output:

   ```
   Starting NFS on docker host
   Applying configuration file to container
   Starting OSCSA [oscsa_gw:1.3]
   Management Console: https://myApplianceHost.example.com:32771

   If you have already configured an OCISSA FileSystem via the Management
   Console,
   you can access the NFS share using the following port.

   NFS Port: 32770

   Example: mount -t nfs -o vers=4,port=32770 myApplianceHost.example.com:/
   OCISSA_filesystem_name/local_mount_point
   ```

4. Open the management console in a web browser by entering the management console URL:
   `https://`*`appliance_host_name`*`:`*`port_number`*

   For example, `https://myApplianceHost.example.com:32771`. The management console is displayed.

5. Create the administrator credentials and log in to the management console.
   A wizard is displayed with a message:

   `No FileSystems are created yet.`

6. Click **Create a FileSystem**.

7. Enter the name of your filesystem.

   > **Note:**
   >
   > For the character restrictions applicable when you enter a filesystem name, see Character Restrictions.

   Click **Next**.

8. Enter the object storage API endpoint of your account.

> **Note:**
>
> - To find the object storage API endpoint for your Oracle Cloud Infrastructure Object Storage Classic account, see About REST URLs for Oracle Cloud Infrastructure Object Storage Classic Resources in *Using Oracle Cloud Infrastructure Object Storage Classic*.
> - To find the object storage API endpoint for your Oracle Cloud Infrastructure tenancy, see API documentation for Oracle Cloud Infrastructure.

9. If you'd like to create a filesystem in Oracle Cloud Infrastructure Object Storage Classic, go to step 9a. If you'd like to create a filesystem in Oracle Cloud Infrastructure, go to step 9b.
   a. Create a filesystem using your Oracle Cloud Infrastructure Object Storage Classic account details:
      - User name of your account (for example: `jack.jones@example.com`).
      - Password of your account.
      - Click **Validate**.
   b. Create a filesystem using your Oracle Cloud Infrastructure Object Storage tenancy details:
      - Tenant ID
      - User ID
      - Compartment ID (Optional)
      - Public key finger print
      - Private key
      - Private key passphrase
10. Set the following options as required:
    - **Enable Archive**: Select this check box if you want to connect this filesystem to a container or bucket of the `Archive` storage class.
      For more information, see About Archive FileSystems.
11. Click **Show Advanced**, and enter the required information in the advanced configuration fields. For more information, see the table in Step 9 in Adding a FileSystem.
12. Click **Save**.
    The filesystem is created and the details of the filesystem are displayed on the **Dashboard** page.

About Archive FileSystems

When you create an `Archive` filesystem, if a container or bucket by the same name doesn't exist in your account, then the container or bucket will be created. In addition, another container or buclet named *`filesystem_name`*`-archive` is created.

For example, if you create an `Archive` filesystem with the name `myFirstArchiveFS`, then the following two containers or buckets are created:

- `myFirstArchiveFS-archive`
  This container or bucket is of the storage class `archive`. All the files that are uploaded in the mounted directory on the appliance host are stored as objects in the `myFirstArchiveFS` filesystem and then asynchronously copied to this container. Metadata backups are also stored in this container.
- `myFirstArchiveFS`
  Metadata synchronization objects are stored in this container or bucket. Metadata synchronization objects enable metadata information to be restored if the appliance installation is lost.

  In addition to metadata synchronization objects, the metadata backups are stored in an `Archive` container or bucket periodically.

If the filesystem name matches an existing container or bucket, and if the storage class of the container is `standard`, then you cannot mount the filesystem and an error message is displayed.

The appliance automatically performs daily, weekly, and monthly metadata backups.

Next Steps

Connect the filesystem to a directory on the appliance host. For more information, see Connecting a FileSystem.

To manage Oracle Cloud Infrastructure Storage Software Appliance by using the management console, see Managing and Monitoring the Appliance.
You can also do the following tasks in the management console:

- Set up the NFS export. This directory will act as a mount point. For more information, see Mounting FileSystems on Clients.
- Add more filesystems. For more information, see Adding a FileSystem.
- View the details of a filesystem. For more information, see Viewing the Details of a FileSystem.
- Connect or disconnect a filesystem. For more information, see Connecting a FileSystem.

Mounting FileSystems on Clients

Each filesystem in Oracle Cloud Infrastructure Storage Software Appliance maps a directory on the appliance host to a container or bucket. To establish the connection between Oracle Cloud Infrastructure Storage Software Appliance and an NFS client, you must mount the appliance filesystem on the NFS client.

Verified NFS Clients

- Oracle Linux 6.4, 6.6 and 7.3
- Ubuntu 14.04 and 16.04
- CentOS 7
- Debian 8 Jessie

> **Note:**
>
> The above list of NFS clients have been tested and verified. However, you can use any other NFS client apart from this list.

Steps

1. Log in to the appliance host.
2. Start the appliance by entering the following command:
   `oscsa up`
3. Find out the NFS port number:

   `oscsa info`

   Make a note of the NFS port number from the output.

   Sample output:

   ```
   Management Console: https://myApplianceHost.example.com:32775

   If you have already configured an OSCSA FileSystem via the
   Management Console,
   you can access the NFS share using the following port.

   NFS Port: 32774

   Example: mount -t nfs -o vers=4,port=32774
   myApplianceHost.example.com:/filesystem_name /local_mount_point
   ```

   In the sample output,

   - *myApplianceHost.example.com* is the appliance host name
   - *32775* is the management console port number
   - *32774* is the NFS port
4. Log in to the NFS client from which you want to access your service instance through the appliance.
5. Create a directory on the NFS client.
6. Mount the filesystem on the directory which you created on the NFS client:
   `sudo mount -t nfs -o vers=4, port=NFS_port` *`appliance_host:/OSCSA_filesystem_name /local_mount_point_on_NFS_client`*

   In this command,

   - Replace *appliance_host* with the server name or IP address of the server on which the appliance is installed.
   - Replace *OSCSA_filesystem_name* with the filesystem name that you want to mount.
   - Replace */local_mount_point_on_NFS_client* with the path to the directory you created on the NFS client.

   Example

```
mount -t nfs -o vers=4, port=32774 myApplianceHost.example.com:/
myFirstFS /home/xyz/abc
```

In this example,

- `32774` is the NFS port
- `myApplianceHost.example.com` is the appliance host name
- `myFirstFS` is the filesystem name
- `/home/xyz/abc` is the path to the directory `abc` on the NFS client

The appliance filesystem is now mounted on the NFS client directory. You can now access the appliance filesystem from the NFS client.

For more information, see Using the Appliance to Store and Retrieve Data.

> **Tip:**
>
> To learn how to back up data on a Linux host and store it on your Oracle Cloud Infrastructure Object Storage Classic account using the appliance, see the tutorial Backing Up Data Using Oracle Cloud Infrastructure Storage Software Appliance.

Workflow for Using Storage Software Appliance

Task	Description	More Information
Creating and managing a filesystem	Create a filesystem, encrypt data in a filesystem, and connect the filesystem to a container or bucket on your account or tenancy.	• Managing FileSystems
Storing and retrieving data using Oracle Cloud Infrastructure Storage Software Appliance	Store files from the directories on the server to the filesystems in Oracle Cloud Infrastructure Storage Software Appliance. Retrieve the files from the filesystems to the directories on the server.	• Using the Appliance to Store and Retrieve Data
Managing and monitoring Oracle Cloud Infrastructure Storage Software Appliance	Start or stop Oracle Cloud Infrastructure Storage Software Appliance. View the logs in Oracle Cloud Infrastructure Storage Software Appliance.	• Managing and Monitoring the Appliance

2
Managing FileSystems

You can manage the appliance filesystems and store data in your account.

Topics

Adding a FileSystem

You can add one or more filesystems in Oracle Cloud Infrastructure Storage Software Appliance and connect each filesystem to a container or bucket in your account or tenancy.

1. Log in to the management console.
   The available filesystems are displayed.
2. Click **Create Filesystem** in the navigation pane on the left.
   The **Create a FileSystem** page is displayed.
3. Enter the required information in the following fields:

Field	Description
FileSystem Name	Name of the filesystem. If a container or bucket by the same name doesn't exist in your account or tenancy, then it will be created. Enter a name that is meaningful to you and unique. **Note**: • For the character restrictions applicable when you enter a filesystem name, see Character Restrictions. If a container or bucket with the same name as the filesystem already exists, and if that container or bucket isn't empty, then the data cached in the Oracle Cloud Infrastructure Storage Software Appliance filesystem may not be consistent with data stored in the container or bucket.

Field	Description
Object Storage API Endpoint	The object storage API endpoint for your service instance. • To find out the object storage API endpoint for your Oracle Cloud Infrastructure Object Storage Classic instance, see About REST URLs for Oracle Cloud Infrastructure Object Storage Classic Resources in *Using Oracle Cloud Infrastructure Object Storage Classic*. (Not available on Oracle Cloud at Customer) • To find out the object storage API endpoint for your Oracle Cloud Infrastructure Object Storage tenancy, see API documentation for Oracle Cloud Infrastructure. Go to Step 4a.
User Name	Oracle Cloud Infrastructure Object Storage Classic user name. **Note**: This field is displayed only for **Oracle Cloud Infrastructure Object Storage Classic** accounts.
Account Password	Oracle Cloud Infrastructure Object Storage Classic password. **Note**: This field is displayed only for **Oracle Cloud Infrastructure Object Storage Classic** accounts.
Compartment ID	Oracle Cloud Infrastructure Object Storage Compartment ID **Note**: This field is displayed only for Oracle Cloud Infrastructure tenancy.
Tenant ID	Oracle Cloud Infrastructure Object Storage Tenant ID **Note**: This field is displayed only for Oracle Cloud Infrastructure tenancy.
User ID	Oracle Cloud Infrastructure Object Storage User ID **Note**: This field is displayed only for Oracle Cloud Infrastructure tenancy.
Public Key Finger Print	Oracle Cloud Infrastructure Object Storage Public Key Finger Print **Note**: This field is displayed only for Oracle Cloud Infrastructure tenancy.
Private Key	Oracle Cloud Infrastructure Object Storage Private Key **Note**: This field is displayed only for Oracle Cloud Infrastructure tenancy.
Private Key Passphrase	Oracle Cloud Infrastructure Object Storage Private Key Passphrase **Note**: This field is displayed only for Oracle Cloud Infrastructure tenancy. Go to Step 4b.

4. a. Click **Validate**.
   If any of the values entered above do not match your Oracle Cloud Infrastructure Object Storage Classic account credentials, then an error message is displayed. Recheck and enter the appropriate values in the respective fields.

   b. Click **Save**.

5. If you want to create an `Archive` filesystem, then select the **Enable Archive** check box.

   For more information, see About Archive FileSystems.

   If the filesystem name matches an existing container, and if the storage class of the container is `standard`, then you cannot mount the filesystem and an error message is displayed.

6. Click **Show Advanced**, and enter the required information in the following fields:

Field	Description
NFS Allowed Hosts	The hosts allowed to connect to the NFS export. Example: `2001:db8:9:e54::/64`, `192.0.2.0/24`
NFS Export Options	The NFS export options. Example: `rw`, `sync`, `insecure`, `no_subtree_check`, `no_root_squash` Don't specify the `fsid` option.
Maximum Local Cache Size in GiB	The maximum number of bytes that can be cached. When the data in the cache reaches the specified limit or the cache is full, the appliance removes files from the cache based on a least recently used (LRU) algorithm. The files that are yet to be uploaded to the account are not removed from the cache. The preserved files (by cache pinning) are also not removed from the cache. See Configuring the Cache for FileSystems. **Note**: The number of files in cache is limited to 20,000, regardless of the specified cache size in bytes.
Concurrent Uploads	The number of concurrent uploads to the cloud. Allowed range: 1 to 30 This field indicates the maximum number of files that can be concurrently uploaded in the appliance. If the value is 5, the concurrent file uploads can be between 0-5.
Delete Old File Versions	If you select this check box for a filesystem, then every time you start the appliance and once every 24 hours after that, all the older versions of all the objects are deleted in the container or bucket that's connected to the filesystem. Only the latest version of each object is retained. For more information, see Enabling File Versions Compaction. This option is disabled by default.
Restore Object Retention	The number of days a file will remain in the restored state. **Note**: This field is displayed only for an `archive` filesystems. Default value: 10
Sync Policy	The metadata operations are flushed to the disk based on the following modes. Select one of the following modes based on your requirement: • **Asynchronous** In this mode, the filesystem operations are time-based and are persisted asynchronously. This mode offers the best performance. **Note**: This mode is not suitable for any filesystem operation that depends on synchronous transactions. • **Posix Standard** This mode is enabled by default. Only the synchronous transactions (like fsync, ODSYNC and OSYNC) are committed to the disk. All the other transactions are handled asynchronously. These sync modes do not affect on-disk consistency.
Cloud Read-ahead	The number of 1-MB blocks to be downloaded and used to *read ahead* when reading files. Use this setting to improve the read performance for large files that aren't cached. Default value: 0 (prefetching is disabled)

7. Click **Save**.

Next Task

Connect the filesystem. See Connecting a FileSystem.

Importing an Existing Filesystem

Prerequisite

Before you import an existing filesystem from another appliance, ensure that all the pending uploads of the files residing on the appliance which last owned the filesystem, are completed and the files are uploaded to your account.

> **Note:**
>
> There may be pending or interrupted file uploads in a failed appliance. If you're importing an existing filesystem from a failed appliance, you must re-create those files in the filesystem on the recovery appliance.

1. Log in to the management console.
   The available filesystems are displayed.
2. Click **Create FileSystem** in the navigation pane on the left.
   The **Create a FileSystem** page is displayed.
3. Enter the required information in the **Required** tab.
   For the filesystem name, enter the name of the existing filesystem that you want to import to this appliance.
4. To continue importing a filesystem for your Oracle Cloud Infrastructure Object Storage Classic account, go to Step 4a.
   a. Click **Validate**.
      If any of the values entered do not match your Oracle Cloud Infrastructure Object Storage Classic account credentials, then an error message is displayed. Recheck and enter the appropriate values in the respective fields.
   b. Click **Save**.
5. Select the options that you'd like to enable in the filesystem.
6. Click **Show Advanced**, and enter the required information.
7. Click **Save**.
   The filesystem is created and displayed on the **Dashboard** tab.
8. Click **Connect** for the filesystem that you want to import.
   If the filesystem that you're importing is connected to another appliance, then **FileSystem: Claim Ownership** window is displayed, prompting you to confirm whether the other appliance must be disconnected.

   If you opt to proceed, then take necessary action, depending on the storage service type:

   - For an Oracle Cloud Infrastructure Object Storage Classic account, re-enter your Oracle Cloud Infrastructure Object Storage Classic password and select **Claim Ownership**.

- For an Oracle Cloud Infrastructure Object Storage tenancy, re-enter the values for the following fields in the **FileSystem: Claim Ownership** window and select **Claim Ownership**.
  - **Public Key Finger Print**
  - **Private Key**
  - **Private Key Passphrase**

A filesystem may be mounted for read/write on only one appliance at a time.

If a container with the same name as the filesystem exists in your Oracle Cloud Infrastructure Object Storage Classic account, then the filesystem is connected to that container. If a container by that name doesn't exist, then it's created and the filesystem is connected to the container.

If a bucket with the same name as the filesystem exists in your Oracle Cloud Infrastructure Object Storage tenancy, then the filesystem is connected to that bucket. If a bucket by that name doesn't exist, then it's created and the filesystem is connected to the bucket.

9. Mount the filesystem to a directory on the appliance host and set up the NFS export.
   `sudo mount -t nfs -o vers=4,port=`*`NFS_port_number appliance_host`*`:/`*`filesystem_name /path/to/directory`*

Viewing the Details of a FileSystem

You can view the configuration details of a filesystem and also monitor the upload activity, through the management console of Oracle Cloud Infrastructure Storage Software Appliance.

To view the details of a filesystem, log in to the management console, and click the name of the filesystem:

- The **Details** tab displays the storage service type.
  For Oracle Cloud Infrastructure Object Storage Classic accounts, the identity domain associated with your account is also displayed.

  For Oracle Cloud Infrastructure Object Storage Classic accounts, you can also view a graphical representation of the amount of cloud storage being used by the filesystem and available free space on the appliance host, with the following details:
  - Current Usage
  - Free Space
- The **Settings** tab displays the following details:
  - Details of the account or tenancy specified for the filesystem
  - Enabled filesystem properties (such as encryption, `Archive` storage class and deleting old file versions)
  - NFS and cache settings for the filesystem

  You can edit these settings. If you make any changes, remember to click **Save**.
- The **Activity** tab shows the ongoing and pending upload activity.
  If you contact Oracle Support Services about any issue with the filesystem, you may need to provide the filesystem log to help the Oracle Support Services

technician diagnose the issue. To view or download the filesystem log, click **View Streaming Logs** near the lower-right corner of the **Details** tab.

- The **Completed Uploads** tab shows the last 100 files that were uploaded to your account during the current browser session. Note that this list doesn't persist across browser sessions. If you refresh the page or if you open the **Completed Uploads** tab in another browser after the files are uploaded, then the list will be empty.
- You can also disconnect the filesystem. See Connecting a FileSystem.

Next Task

You can edit the properties of a filesystem. See Changing the Properties of a FileSystem.

Changing the Properties of a FileSystem

You can change the properties of a filesystem, through the management console of Oracle Cloud Infrastructure Storage Software Appliance.

> **Note:**
>
> You can't enable or disable encryption, and you can't change the storage class of the filesystem.

To change the properties of a filesystem, log in to the management console, and click the name of the filesystem in the **Dashboard** pane:

- You can edit the filesystem properties and advanced settings (such as the cache limits) of the service instance specified for the filesystem in the **Settings** tab.

After updating the filesystem properties, click **Save**.

> **Note:**
>
> For the changes to take effect, you must disconnect and reconnect the filesystem. See Connecting a FileSystem.

Configuring the Cache for FileSystems

Oracle Cloud Infrastructure Storage Software Appliance caches frequently retrieved data on the local host, minimizing the number of REST API calls to your service instance and enabling faster data retrieval. The appliance uses an upload buffer and a read cache for data storage and retrieval.

Topics

- About the FileSystem Cache
- Guidelines for Sizing and Configuring the Cache
- Configuring the FileSystem Cache

About the FileSystem Cache

The filesystem cache serves two roles: an upload buffer and a read cache. The upload buffer contains data that has been copied to the disk cache and is queued to be stored in your account or tenancy. The read cache contains frequently retrieved data that's accessible locally for read operations.

When an application transfers a file through an NFS share, the file is queued to be stored in your account or tenancy. The upload buffer might contain many files. If the host on which Oracle Cloud Infrastructure Storage Software Appliance is installed fails, or if the appliance stops abruptly, the pending upload operations are not lost because they are persisted on the local disk. When the appliance restarts, the pending upload operations resume and the data is stored in your account or tenancy.

When you retrieve data, the data is stored in the read cache of the appliance. This allows subsequent I/O operations to that file to be done at local disk speed.

When the data in the cache reaches the specified limit or the cache is full, the appliance removes files from the cache based on a least recently used (LRU) algorithm. The files that are yet to be uploaded to the account are not removed from the cache. The preserved files (by cache pinning) are also not removed from the cache.

For more information on how to preserve files in the cache, see Preserving Files in the FileSystem Cache.

Guidelines for Sizing and Configuring the Cache

See Best Practices – Configuring Cache Storage and Best Practices – Determining the Cache Size.

Configuring the FileSystem Cache

You can configure the cache for a filesystem while adding the filesystem. See Adding a FileSystem.

Preserving Files in the FileSystem Cache

When you write a file to your filesystem, it's initially stored in the filesystem cache, and then uploaded to your account. Once a file has been uploaded to your account, it may get removed from the filesystem cache by the cache manager. The cache is reclaimed using the *Least Recently Used* (LRU) cache management policy to meet the cache threshold that's specified in the filesystem advanced settings. If you want specific files to be always available in the cache for quick access, you can preserve them in the filesystem cache by pinning them to the cache. Once pinned, the files are not removed from the filesystem cache, except if you specifically unpin them.

You can pin files to the cache for filesystems connected to any storage class, `Standard` or `Archive`. Files that you write to a filesystem are uploaded to your account, regardless of whether the files are pinned to the cache.

If the file that you want to pin to the filesystem cache is not present in the cache, then it's automatically downloaded to the cache from the account. If that file belongs to a filesystem of the `Archive` storage class, then it's first restored, and then downloaded.

Note:

- When selecting the files for cache pinning, consider the overall cache threshold and calculate the residual cache space that would be available for normal cache operations. For example, if your cache threshold is 1 TB, and you estimate files that are pinned to the cache to occupy 300 GB, then you'd have 700 GB usable space on your cache after pinning the files. See Best Practices – Configuring Cache Storage and Best Practices – Determining the Cache Size.
- By default, the cache pinning feature is enabled on all filesystems.
- When you restore a file that belongs to a filesystem of the `Archive` storage class, the file will remain in the corresponding container of the `Standard` storage class for the duration specified in the **Restore Object Retention** field for any filesystem. Its continued availability in the cache will depend on the LRU operation. However, when you pin such a file to the cache, the restored file will remain in the cache, until if you specifically unpin it.
- You can pin files to the cache and restore them in filesystems of the `Archive` storage class only for Oracle Cloud Infrastructure Object Storage Classic accounts.

Enabling and Managing Cache Pinning

To perform cache pinning operations for a filesystem, run the following command from the NFS client on which the filesystem is mounted:

```
cat /path/to/mountpoint:::cache:cache_command[:argument]
```

The following table lists the cache pinning operations and the corresponding command and argument for each operation:

Operation	Cache Command	Argument
Enable cache pinning for a filesystem Note that, by default, cache pinning is enabled for all filesystems.	`set-preserve-option`	`true`
Get the cache pinning status for a filesystem	`get-preserve-option`	No argument
Disable cache pinning for a filesystem	`set-preserve-option`	`false`
List the files that are pinned to the cache	`list-preserve`	No argument
Remove any files from the preserve list that have been deleted	`list-preserve-update`	No argument
Add a file to the preserve list	`add-preserve`	No argument
Remove a file from the preserve list	`remove-preserve`	No argument
Clear the preserve list	`clear-preserve`	No argument

Example Commands

- To enable cache pinning for the `myFS` filesystem:

  ```
  cat /mnt/gateway/myFS/:::cache:set-preserve-option:true
  ```

- To get the cache pinning status for `myFS`:

  ```
  cat /mnt/gateway/myFS/:::cache:get-preserve-option
  ```

  The output of this command is `true` if cache pinning is enabled for the filesystem. Otherwise, `false`.

- To disable cache pinning for the `myFS` filesystem:

  ```
  cat /mnt/gateway/myFS/:::cache:set-preserve-option:false
  ```

- To add a file `myFile` of the `myFS` filesystem to the preserve list:

  ```
  cat /mnt/gateway/myFS/myFile:::cache:add-preserve
  ```

- To find out which files are added to the preserve list of the `myFS` filesystem:

  ```
  cat /mnt/gateway/myFS/:::cache:list-preserve
  ```

  A sample output of the above command:

  ```
  ["/doNotDelete.txt", "/myFileMetadata", "/myFile"]
  ```

- To remove the file `myFile` from the preserve list

  ```
  cat /mnt/gateway/myFS/myFile:::cache:remove-preserve
  ```

- To update the preserve list when the output of the `cache:list-preserve` command indicates that a pinned file has been removed from the filesystem:

  ```
  cat /mnt/gateway/myFS/:::cache:list-preserve-update
  ```

  A sample of the original preserve list:

  ```
  ["/doNotDelete.txt", "/myFileMetadata"]
  ```

  Output of the `cache:list-preserve` command after the file `myFileMetadata` is removed from the cache:

  ```
  ["/doNotDelete.txt", "Status: 1 files appear to no longer exist. Please run
  list-preserve-update"]
  ```

  Output of the `cache:list-preserve-update` command:

  ```
  ["/doNotDelete.txt"]
  ```

- To clear the preserve list for a filesystem:

  ```
  cat /mnt/gateway/myFS/:::cache:clear-preserve
  ```

Connecting a FileSystem

Connecting a FileSystem

After you create a filesystem, you must connect it to a container or bucket in your account before you can store and retrieve data through the filesystem.

> **Caution:**
>
> If a container or bucket with the same name as the filesystem already exists, and if that container or bucket isn't empty, then the data cached in the Oracle Cloud Infrastructure Storage Software Appliance filesystem may not be consistent with data stored in the container or bucket.

1. Log in to the management console of Oracle Cloud Infrastructure Storage Software Appliance.
2. On the **Dashboard** tab, identify the filesystem that you want to connect to your account.
3. Click **Connect**.
   If a container with the same name as the filesystem exists in Oracle Cloud Infrastructure Object Storage Classic, then the filesystem is connected to that container. If a container by that name doesn't exist, then it's created and the filesystem is connected to the container.

   If a bucket with the same name as the filesystem exists in Oracle Cloud Infrastructure, then the filesystem is connected to that bucket. If a bucket by that name doesn't exist, then it's created and the filesystem is connected to the bucket. The bucket is created in the `root` compartment by default.

   > **Note:**
   >
   > A filesystem may be mounted for read/write on only one appliance at a time.
   >
   > - If the filesystem name that you've specified matches the name of an existing container in your Oracle Cloud Infrastructure Object Storage Classic account, and if that container is connected to another appliance filesystem, then **FileSystem: Claim Ownership** window is displayed, prompting you to confirm whether the other filesystem must be disconnected. If you opt to proceed, then you must re-enter your Oracle Cloud Infrastructure Object Storage Classic password and select **Claim Ownership**.
   > - If the filesystem name that you've specified matches the name of an existing bucket in your Oracle Cloud Infrastructure Object Storage tenancy, and if that bucket is connected to another appliance filesystem, then re-enter the values for the following fields in the **FileSystem: Claim Ownership** window and select **Claim Ownership**.
   >   - **Public Key Finger Print**
   >   - **Private Key**
   >   - **Private Key Passphrase**
   >
   > This check ensures that you don't inadvertently connect the new filesystem to a container or bucket that's already connected to another appliance filesystem.

4. Mount the filesystem to a directory on the appliance host and set up the NFS export.
   ```
   sudo mount -t nfs -o vers=4,port=NFS_port_number appliance_host:/
   filesystem_name /path/to/directory
   ```

Disconnecting a FileSystem

To disconnect a filesystem, select it on the dashboard of the Oracle Cloud Infrastructure Storage Software Appliance management console, and click **Disconnect**.

The container or bucket to which the filesystem was previously connected and the stored data remain intact even after the filesystem is disconnected.

At any time, you can resume storing and retrieving data through the filesystem by connecting it again. If you no longer need the disconnected filesystem, then you can delete it. See Deleting a FileSystem.

Enabling File Versions Compaction

Oracle Cloud Infrastructure Storage Software Appliance allows file operations with byte-level granularity, such as append, re-write, over-write, and truncate. When a file is modified in an appliance filesystem, it results in a new version of the file being created and uploaded to the account.

When you create a filesystem, you can choose whether older versions of an object stored in the cloud must be retained whenever the corresponding file is updated or deleted in the filesystem.

- If you **select** the **Delete Old File Versions** check box for a filesystem, then a version compaction process runs in the background when you start the appliance. Upon completion of one cycle, the process sleeps for 24 hours before starting the next version compaction cycle. During this process, the older versions of all the objects in the container or bucket connected to the filesystem are removed. Only the latest version of the objects are retained.
- If you don't select the **Delete Old File Versions** check box:
  - When a file is updated in the filesystem, a new version of the corresponding object in the cloud is created. Additional capacity is consumed in the cloud after each such update operation.
  - When a file is deleted in the filesystem, all older versions of the object in the cloud are retained. Capacity continues to be used in the cloud for the file that you deleted in the appliance.

Deleting a FileSystem

When you no longer need a filesystem, you can delete it from Oracle Cloud Infrastructure Storage Software Appliance.

To delete a filesystem:

1. Log in to the management console.
2. On the **Dashboard** tab, identify the filesystem that you want to delete.

3. Make sure that the filesystem is disconnected. If it's still connected, then click **Disconnect**.

> **Note:**
>
> The container or bucket to which the filesystem was previously connected and the stored data remain intact even after the filesystem is disconnected.

4. After the filesystem is disconnected, click its name.
5. On the page that displays the details of the filesystem, click **Delete**.

The filesystem is deleted from Oracle Cloud Infrastructure Storage Software Appliance. Deleting a filesystem does not automatically delete the objects in the container or bucket. If you'd like to remove the objects from the container or bucket, all the files should be deleted from the filesystem prior to disconnecting the filesystem, with version compaction enabled.

3
Managing and Monitoring the Appliance

You can manage Oracle Cloud Infrastructure Storage Software Appliance and view the system status, logs in the management console.

Topics:

Managing the Appliance

You can use the `oscsa` command-line tool to manage Oracle Cloud Infrastructure Storage Software Appliance. Log in to the host on which you installed the appliance and enter:

- To start Oracle Cloud Infrastructure Storage Software Appliance:
  ```
  oscsa up
  ```
- To stop Oracle Cloud Infrastructure Storage Software Appliance:
  ```
  oscsa down
  ```

> **Note:**
>
> If the server with an Oracle Cloud Infrastructure Storage Software Appliance instance fails, you can reinstall and start another instance. All the configuration and system data is automatically downloaded and applied. The pending upload and download activities are resumed when the Oracle Cloud Infrastructure Storage Software Appliance instance is running again.
> If a disk cache is irrecoverable on the server with the Oracle Cloud Infrastructure Storage Software Appliance instance, then data might be lost, as the file might not have been transferred to the container or bucket in your account. To ensure efficient data protection, see Best Practices for Using Storage Software Appliance.

- To view details about Oracle Cloud Infrastructure Storage Software Appliance and how to access the management console:

`oscsa info`

- To find out the version of Oracle Cloud Infrastructure Storage Software Appliance:
  `oscsa version`
- To configure Oracle Cloud Infrastructure Storage Software Appliance to use a proxy server to connect to the storage service:
  `oscsa configure proxy [http_proxy_server https_proxy_server]`

> **Note:**
>
> After configuring the proxy server, you must stop and restart Oracle Cloud Infrastructure Storage Software Appliance.

By default, no proxy server is specified.

- To remove the proxy server details in Oracle Cloud Infrastructure Storage Software Appliance:
  `oscsa configure proxy [remove]`
- To configure Oracle Cloud Infrastructure Storage Software Appliance to use SSL to access the management console:
  `oscsa configure ssl true`

  SSL is enabled by default.

> **Note:**
>
> After configuring Oracle Cloud Infrastructure Storage Software Appliance to use SSL, you must stop and restart Oracle Cloud Infrastructure Storage Software Appliance.

  To disable SSL: `oscsa configure ssl false`
- To specify ports for the Oracle Cloud Infrastructure Storage Software Appliance services:
  `oscsa configure port service port_number`
  - *service*: Specify `admin` or `nfs`.
  - *port_number*: Ensure that the port number is not already in use on the appliance host.

  By default, the port number is assigned dynamically for the Oracle Cloud Infrastructure Storage Software Appliance services when you start the appliance.

  To remove the static port assignment for a service: `oscsa configure port service remove`

> **Note:**
>
> For the port assignment to take effect, you must stop and start the appliance.

- To allocate memory for Oracle Cloud Infrastructure Storage Software Appliance in the appliance host:
  `oscsa configure memory memory_in_GB`

  To remove the memory allocation:

  `oscsa configure memory remove`

  By default, Oracle Cloud Infrastructure Storage Software Appliance uses 4 GB from the available memory on the appliance host. You can delete the memory information by using the `remove` parameter.

  > **Note:**
  >
  > After configuring memory for Oracle Cloud Infrastructure Storage Software Appliance, you must stop and restart Oracle Cloud Infrastructure Storage Software Appliance.

- To specify the docker network mode:
  `oscsa configure network mode`

  The mode can be either *host* or *bridge*.

  The default mode is `bridge`. In this mode, you can run multiple instances of the appliance on your host.

  In the `host` mode, you can run only a single instance of the appliance. Network performance is better in this mode.

  > **Note:**
  >
  > After specifying the docker network mode, you must stop and start the appliance.

- To view help for the available commands:
  `oscsa help`

Monitoring Upload Activity

The **Activity** tab shows the ongoing and pending upload activity in a filesystem.

When you upload a file to a filesystem, you can view the status of the upload activity.

1. Log in to the management console.
2. Select the filesystem.
3. Click the **Activity** tab.
   You can see the upload progress of the file in the **Uploading** pane.

Monitoring System Status Health Check

Health Check

You can monitor the overall system status through the **System Status** pane in the right side of the management console.

The appliance health check service is an automated process run on the system to monitor the status of the following:

- Services – Databases used by the appliance, management console and other auxiliary processes
- Disk space – Local storage
  For example: If the available local storage is lesser than 10 GB, the health check service reports this as an alert.

Depending on the appliance health check analysis, the following status is displayed in the **System Status** pane:

- Healthy
- Unhealthy

The appliance health check service also displays the following details in the **System Status** pane and highlights potential issues:

- Throughput
- Available local cache
- Pending uploads

Local Cache Modes

The **Local I/O** mode might display any one of the following values based on the local disk usage:

- **Normal**
  The free space is higher than 10 GB in the appliance. You can upload files in the appliance and store them in your account.
- **Rejecting I/O**
  The free space is lower than 10 GB in the appliance. The appliance is running on protection mode and will not allow any writes to its local disk. All read operations will work as normal. All metadata operations will fail in the appliance except for deletion and truncation.

  To return to **Normal** mode, you must wait until all the ongoing upload activities are complete and the files are removed from the local cache.

> **Note:**
>
> For optimum local storage configuration, see Best Practices – Configuring Cache Storage.

You can also view the system status details and track any issues using the support bundle. See Viewing System Notifications.

Monitoring Appliance Storage Usage

The **System Stats** tab enables you to track the storage usage and availability.

1. Log in to the management console.
2. Select the **System** tab on the upper-right side of the management console.
3. Click the **System Stats** tab in the **System** pane.
   The system data is displayed in the following three panes:
   - Local Storage
   - Local I/O
   - Local Resources

Local Storage

In this pane, you can view a graphical representation of the amount of storage being used and available free storage on the appliance host, with the following details:

- Available local storage
- Storage used for pending uploads and preserved cache files
- Storage used for metadata
- Storage used for logging
- Storage used for other applications

Local I/O

This pane displays the local I/O mode of the appliance based on the local disk space usage in the appliance host.

Local Resources

In this pane, you can view the overall memory usage and memory availability for the appliance from the following fields:

- Available Cores -The number of CPUs being used by the appliance
- Maximum Memory Available to the Appliance - The total RAM available for the appliance
- Memory Used by the Appliance - The amount of memory being used by the filesystems in the appliance
- Free Memory - The amount of free RAM available in the appliance host

Viewing System Notifications

The **System Notifications** tab allows you to view the system notifications and track the overall system performance.

1. Log in to the management console.
2. Select the **System** tab on the upper-right side of the management console.
3. Click the **System Notifications** tab in the **System** pane.
   You can view the list of warnings or critical system notifications.

Configuring Email Notification

Alternatively, you can provide the required configuration details and get notified about the system health checks through email.

1. Log in to the management console.
2. Select the **System** tab on the upper-right side of the management console.
3. Click the **System Notifications** tab in the **System** pane.
   A message is displayed:

   Email notifications are not configured. Click here to configure.
4. Click on the **Click here to configure** link.
5. Enter the required information for the following fields:
   - **SMTP server**
   - **Email addresses to receive notifications**
6. Click **Show Advanced**, and enter the required information in the advanced configuration fields:
   - **SMTP port**
   - **SMTP User name**
   - **SMTP Password**
   - **Sender's Email Address**
     Default value: `noreply@oracle.com`
7. Click **Save**.
8. Click **Test Email Notification** to verify if a system notification email has been sent successfully to the email id you've provided in the management console.

Downloading Support Bundle

If you contact Oracle Support Services about any issue with the appliance, then you may need to provide the support bundle to help the Oracle Support Services technician diagnose the issue.

1. Log in to the management console.
2. Select the **System** tab on the upper-right side of the management console.
3. Select the **Help** tab.
4. Click **Download Support Bundle** in the **System Logs** pane.
   You can download and save the support bundle.

Support Bundle

The support bundle contains the following information:

- All necessary logs for diagnostics
- Local storage usage information
- Basic system information such as memory size, Docker version, appliance version etc.

- List of filesystems

Upgrading the Appliance

Appliance Upgrade Matrix

This table provides the available release versions to upgrade the appliance, depending on the current version.

Current Appliance Release Version	Available Appliance Release Versions for Upgrade
16.3.1.0.9 or earlier	16.3.1.4 **Note**: To upgrade to 16.3.1.4, you must **first** upgrade to 16.3.1.0.13, and then upgrade to 16.3.1.4.
• 16.3.1.0.10 • 16.3.1.0.12 • 16.3.1.0.13	16.3.1.4
16.3.1.2, 16.3.1.3	16.3.1.4

Before You Begin

- Check the table for the available appliance release versions to upgrade the appliance, depending on the current version.
- Ensure that the filesystems are mounted.
- Check if the file uploads that are in progress in the filesystems have been completed. Ensure that there is no ongoing activity in the **Activity** tab for a filesystem in the management console. See Viewing the Details of a FileSystem.
- Ensure that you plan the downtime appropriately, as the upgrade may take some time. The downtime varies, depending on the system resources and if there are any filesystems to be restored from the cloud. If there are no filesystems to be restored, approximately 1 million records per minute may be transferred during migration, depending on the available system resources. If the filesystem needs to be restored from the cloud, then additional time would be required to download the metadata information and prepare the metadata database for migration

  To minimize downtime, configure and connect the filesystems on the current appliance version before you upgrade the appliance.

Steps

1. Log in to the host on which you want to upgrade Oracle Cloud Infrastructure Storage Software Appliance.
2. Stop the appliance:
   ```
   oscsa down
   ```
3. Delete the following files in the directory where you downloaded and extracted the previous version of the appliance installer:
   - `OSCSA_GATEWAY_README.txt`
   - `OSCSA_RELEASE_NOTES.txt`
   - `oscsa`
   - `oscsa-config.sh`

- `oscsa-control.sh`
- `oscsa-install.sh`
- `oscsa-upgrade.sh`
- `oscsa_gw:1.0.x.tar`

4. Remove the existing `docker` version:
   `sudo yum remove docker`

   Install `docker` using `yum`:

   `sudo yum install docker-engine`
5. Extract the files from the latest version of the installer to a directory on the appliance host by entering the command:
   `tar -xvf installer_tar.gz`
6. Run `installer.sh` extracted from the latest version of the installer. For more information, see Installing Storage Software Appliance.

When you upgrade the appliance, the filesystem configuration is retained even after you delete the old installation files on the host. You can view all the filesystems that you created when you access the management console after the upgrade.

Release 16.3.1.2 included filesystem changes which require migration of filesystem internal data (metadata). The migration takes place automatically when your filesystems are reconnected for the first time during or after the upgrade. The duration of the migration process is dependent on the size of the filesystem and can range from a few minutes to an hour or more. During the migration, all filesystems are in read-only state.

> **Caution:**
>
> - Do not delete any filesystem or change the properties of a filesystem during the migration.
> - Do not reboot the appliance host during the migration.

If there is any interruption during the appliance upgrade, the migration will resume when the appliance upgrade is resumed.

After the migration is complete, the following message appears on the dashboard for the respective filesystem: `Migration completed. Please reconnect the filesystem.`

You can now reconnect the filesystem. See Connecting a FileSystem.

After the filesystems are reconnected, the filesystems that had read / write permissions before the migration will return to the read / write state.

Uninstalling the Appliance

1. Log in to the host on which you want to uninstall Oracle Cloud Infrastructure Storage Software Appliance.
2. Stop the appliance:

```
oscsa down
```

3. Delete the `oscsa_data` file in `docker`:
   ```
   sudo docker rm -v oscsa_data
   ```
4. Delete the image in `docker`:
   ```
   sudo docker rmi $(docker images| grep oscsa_gw | awk '{print $3}')
   ```
5. Delete all the files preceding with `oscsa` in `/usr/bin/`:
   ```
   sudo rm /usr/bin/oscsa*
   ```
6. View the contents of the file `gateway_config`:
   ```
   cat /etc/gateway_config
   ```

   The following custom host paths are listed in the `gateway_config` file, if the custom directories were created during the appliance installation:

   - `DATASTORAGE=/oscsa/data`
   - `MDSTORAGE=/oscsa/md`
   - `LOGSTORAGE=/oscsa/logs`

   a. Delete the directory `data`:
      ```
      sudo rm -rf /oscsa/data
      ```
   b. Delete the directory `md`:
      ```
      sudo rm -rf /oscsa/md
      ```
   c. Delete the directory `logs`:
      ```
      sudo rm -rf /oscsa/logs
      ```

   If there are no custom host paths in the `gateway_config` file, then proceed to the next step.
7. Delete the file `gateway_config`:
   ```
   sudo rm /etc/gateway_config
   ```
8. Delete the appliance installation directory `oscsa_gateway`:
   ```
   sudo rm -rf /opt/oscsa_gateway
   ```

4

Using the Appliance to Store and Retrieve Data

Topics:

Deleting Files

Remove the files that you no longer need from the NFS client by deleting them from the directory on which the filesystem is mounted.

> **Caution:**
>
> If you've enabled the check box to delete old file versions in the filesystem, then depending on the configuration settings, the older versions of the objects may be automatically removed from your account.

To delete the older versions of the objects and the corresponding metadata files, log in to the management console, click the filesystem name, and select the check box **Delete Old File Versions** in the **Advanced** section of the **Settings** tab.

The files in the filesystem are not deleted.

For more information, see Enabling File Versions Compaction.

> **Note:**
>
> Deletion of archived objects may result in an early-deletion fee. For more information, go to https://cloud.oracle.com/storage and see the **Pricing** tab.

Caution:

Don't use the REST API, Java library, or any other client to retrieve, create, update, or delete objects in a container or bucket that's mapped to a filesystem in Oracle Cloud Infrastructure Storage Software Appliance. Doing so will cause the data in the appliance to become inconsistent with data in your storage service. You can't recover from this inconsistency.

To prevent unauthorized users from retrieving, creating, updating, or deleting objects in a container or bucket that's connected to a filesystem in Oracle Cloud Infrastructure Storage Software Appliance, define custom roles, assign them to the appropriate container, assign the roles to only the users that should have access to the container, and specify only one of these users when defining the filesystem to be connected to the container or bucket.

Uploading Files

Topics

Uploading Files to Archive Containers

Caution:

Don't use the REST API, Java library, or any other client to retrieve, create, update, or delete objects in a container or bucket that's mapped to a filesystem in Oracle Cloud Infrastructure Storage Software Appliance. Doing so will cause the data in the appliance to become inconsistent with data in your storage service. You can't recover from this inconsistency.

To prevent unauthorized users from retrieving, creating, updating, or deleting objects in a container or bucket that's connected to a filesystem in Oracle Cloud Infrastructure Storage Software Appliance, define custom roles, assign them to the appropriate container, assign the roles to only the users that should have access to the container, and specify only one of these users when defining the filesystem to be connected to the container or bucket.

Prerequisite

Before you connect the filesystem to the container, ensure that the container is of the `Archive` storage class. To find out the storage class of the container, see Getting Container Metadata.

Ensure that the filesystem in Oracle Cloud Infrastructure Storage Software Appliance is connected to the container. See Connecting a FileSystem.

Procedure

Mount the appliance filesystem on the NFS client. Copy the files to the mount point. The appliance caches the files while they are queued and asynchronously uploads them to the corresponding `Archive` container.

> **Note:**
>
> For the character restrictions applicable when you must select a file name, see Character Restrictions.

Uploading Files to Standard Containers

> **Caution:**
>
> Don't use the REST API, Java library, or any other client to retrieve, create, update, or delete objects in a container or bucket that's mapped to a filesystem in Oracle Cloud Infrastructure Storage Software Appliance. Doing so will cause the data in the appliance to become inconsistent with data in your storage service. You can't recover from this inconsistency.
>
> To prevent unauthorized users from retrieving, creating, updating, or deleting objects in a container or bucket that's connected to a filesystem in Oracle Cloud Infrastructure Storage Software Appliance, define custom roles, assign them to the appropriate container, assign the roles to only the users that should have access to the container, and specify only one of these users when defining the filesystem to be connected to the container or bucket.

Prerequisite

Ensure that the filesystem in Oracle Cloud Infrastructure Storage Software Appliance is connected to the appliance host. See Connecting a FileSystem.

Procedure

Copy the files to the mounted directory on the appliance host or the NFS client host. Oracle Cloud Infrastructure Storage Software Appliance writes the files to the disk cache. The files are queued and then uploaded asynchronously to Oracle Cloud Infrastructure Object Storage Classic.

> **Note:**
>
> For the character restrictions applicable when you must select a file name, see Character Restrictions.

You can check the status of the files being uploaded in the management console. See Viewing the Details of a FileSystem.

Uploading Files to Buckets

> **Caution:**
>
> Don't use the REST API, Java library, or any other client to retrieve, create, update, or delete objects in a bucket that's mapped to a filesystem in the appliance. Doing so will cause the data in the appliance to become inconsistent with data in Oracle Cloud Infrastructure. You can't recover from this inconsistency.

Prerequisite

- Before you connect the filesystem to the bucket, check the storage class of the bucket.
- Ensure that the filesystem in Oracle Cloud Infrastructure Storage Software Appliance is connected to the appliance host. See Connecting a FileSystem.
- Make a note of the Oracle Cloud Infrastructure Object Storage tenancy details like namespace, tenant ID etc.

Procedure

Copy the files to the mounted directory on the appliance host or the NFS client host. Oracle Cloud Infrastructure Storage Software Appliance writes the files to the disk cache. The files are queued and then uploaded asynchronously to your Oracle Cloud Infrastructure account.

> **Note:**
>
> For the character restrictions applicable when you must select a file name, see Character Restrictions.

For large files (with file size higher than 128 MB), the appliance automatically performs multipart upload and stores the large object in a bucket in your Oracle Cloud Infrastructure Object Storage tenancy.

You can view the files that were uploaded to your account during the current browser session. For more information, see the **Completed Uploads** tab in Viewing the Details of a FileSystem.

Retrieving Files

Topics

- Reading Files
- Restoring Files From Archive Filesystems
- Tracking Restoration of a File in an Archive Filesystem
- Tracking Restoration of All Files in an Archive FileSystem

Reading Files

When a file is written to an appliance filesystem, it is stored in the local disk cache, and you can read the file directly from the mounted directory. The file is asynchronously copied to the corresponding container or bucket in your account. To retrieve the data from the container or bucket in your account by using the appliance, read the required files from the mounted directory. The appliance will automatically place the files in the local cache, if space is available.

> **Note:**
>
> When a file is copied to an `archive` filesystem, it is stored in the local disk cache. After the file is asynchronously copied to the corresponding `Archive` container in Oracle Cloud Infrastructure Object Storage Classic, it is stored as an archived object. If the file is in the local disk cache, then you can retrieve the file immediately. However, if the file is not available in the local disk cache and stored in the `Archive` container, then you must first restore the archived object. For more information, see Restoring Files From Archive Filesystems.
>
> If you try to download a file which does not exist in the local cache and is stored as an archived object, then an error message is displayed.

Reading the Checksum for a File

To read the checksum for a file in a filesystem, run the following command from the NFS client on which the filesystem is mounted:

```
cat /path/to/mountpoint/filename:::meta:csm
```

Restoring Files From Archive Filesystems

To download a file from an `Archive` filesystem, you must first restore the corresponding archived object in the container or bucket. The restored object is then downloaded and stored as a file in the appliance cache. Restoring archived files is an asynchronous operation.

To restore a file from an `Archive` filesystem, run the following command on the NFS client:

```
cat path_to_filename:::archive:restore
```

For example, a file `myFirstFile` is copied to a mounted directory `myArchiveDir` on the NFS client and is uploaded to an `Archive` filesystem `myFirstArchiveFS`. The file is asynchronously stored as an archived object in the `myFirstArchiveFS-archive` container or bucket in your account. To restore the archived object, enter the command:

```
cat /path_on_NFS_client/myArchiveDir/myFirstFile:::archive:restore
```

Sample Response:

For Oracle Cloud Infrastructure Object Storage accounts:

```
{"path":"/myFirstFile","restoreStatus":"inprogress","additionalInfo":""}
```

For Oracle Cloud Infrastructure Object Storage Classic accounts:

```
{"path":"/
myFirstFile","restoreStatus":"inprogress","restoreObjectPercent":
{"13456760 1079-11-v1":2},"additionalInfo":""}
```

> **Note:**
>
> You can restore an archived object in an `archive` container or bucket. If you try to restore an object in a `standard` container or bucket, then the following error message is displayed:
> `archive is not a valid command class`

You can now track the restoration progress of the object in the `myFirstArchiveFS-archive` container or bucket. To track the object's restoration progress, see Tracking Restoration of a File in an Archive Filesystem.

You can now track the restoration progress of the object in the `myFirstArchiveFS-archive` container. To track the object's restoration progress, see Tracking Restoration of a File in an Archive Filesystem.

> **Note:**
>
> If the filesystem is deleted and if you restore an object in the `Archive` container or bucket at the same time, the object restoration is not affected.

Tracking Restoration of a File in an Archive Filesystem

To track the restoration progress of the file in the `Archive` filesystem, run the following command on the NFS client:

`cat` *`path_to_filename`*`:::archive:restore-status`

For example, a file `myFirstFile` is copied to a mounted directory `myArchiveDir` on the NFS client and is uploaded to `myFirstArchiveFS`. The file is asynchronously stored as an archived object in the `myFirstArchiveFS-archive` container or bucket in your account and you've run the command to restore the object. To track the restoration status, enter the command:

`cat` *`/path_on_NFS_client/myArchiveDir/myFirstFile`*`:::archive:restore-status`

Sample Response:

For Oracle Cloud Infrastructure Object Storage accounts:

```
{"path":"/myFirstFile","restoreStatus":"restored","additionalInfo":""}
```

For Oracle Cloud Infrastructure Object Storage Classic accounts:

```
{"path":"/myFirstFile","restoreStatus":"restored","restoreObjectPercent":
{},"additionalInfo":""}
```

By default, a restored object will be downloaded and stored as a file in the `Archive` filesystem for one day. You can now read the file from the `Archive` filesystem before the restoration expires. For more information, see Reading Files.

Tracking Restoration of All Files in an Archive FileSystem

To track the restoration status of all the files in an `Archive` filesystem, run the following command:

```
cat /path_on_NFS_client_to_mounted_directory:::archive:jobs
```

The following is a sample response:

```
{"/myFirstFile":"restored","/mySecondFile":"inprogress", "/
myThirdFile":"inprogress"}
```

Example:

The following is an example to show the restoration and tracking the restoration status of the files in an `Archive` filesystem `myArchiveDir`:

1. Restoring the file `myFirstFile`:
   ```
   cat /mnt/dir1/myArchiveDir/myFirstFile:::archive:restore
   ```

   Output:

   For Oracle Cloud Infrastructure Object Storage accounts:

   ```
   {"path":"/
   myFirstFile","restoreStatus":"inprogress","additionalInfo":""}
   ```

   For Oracle Cloud Infrastructure Object Storage Classic accounts:

   ```
   {"path":"/
   myFirstFile","restoreStatus":"inprogress","restoreObjectPercent":
   {"1464707450825-13-v1":0},"additionalInfo":""}
   ```

2. Restoring the file `mySecondFile`:
   ```
   cat /mnt/dir1/myArchiveDir/mySecondFile:::archive:restore
   ```

   Output:

   For Oracle Cloud Infrastructure Object Storage accounts:

   ```
   {"path":"/
   mySecondFile","restoreStatus":"inprogress","additionalInfo":""}
   ```

   For Oracle Cloud Infrastructure Object Storage Classic accounts:

   ```
   {"path":"/
   mySecondFile","restoreStatus":"inprogress","restoreObjectPercent":
   {"1353643456634-13-v1":0},"additionalInfo":""}
   ```

3. Restoring the file `myThirdFile`:
   ```
   cat /mnt/dir1/myArchiveDir/myThirdFile:::archive:restore
   ```

   Output:

   For Oracle Cloud Infrastructure Object Storage accounts:

   ```
   {"path":"/
   myThirdFile","restoreStatus":"inprogress","additionalInfo":""}
   ```

   For Oracle Cloud Infrastructure Object Storage Classic accounts:

```
{"path":"/
myThirdFile","restoreStatus":"inprogress","restoreObjectPercent":
{"1734537242537-13-v1":0},"additionalInfo":""}
```

4. Tracking the restoration progress of all the files:
   ```
   cat /mnt/dir1/myArchiveDir/:::archive:jobs
   ```

   Output:

   ```
   {"/myFirstFile":"restored","/mySecondFile":"inprogress", "/
   myThirdFile":"inprogress”}
   ```

5

Best Practices for Using Storage Software Appliance

Follow the best practices described here to get maximum benefit from Oracle Cloud Infrastructure Storage Software Appliance in terms of manageability, performance, reliability, and security.

Topics

Best Practices – Configuring Cache Storage

Oracle Cloud Infrastructure Storage Software Appliance uses local storage attached to the server (or virtual server) for hosting the filesystems and cache. Files written to a filesystem in the appliance are uploaded to the associated container or bucket, with a portion of the file set maintained locally in the filesystem as a warm cache.

For optimal performance, reliability, and fault tolerance, consider the following guidelines when configuring the local appliance storage:

- Allocate a dedicated volume for the appliance filesystem and metadata.
- Multiple disks (hard disk drives or solid state drives) in a RAID10 set provide an optimal balance of performance, reliability, and fault tolerance. Alternatively, RAID6 may be used. Avoid RAID0 or single disk (no RAID) due to the potential for data loss due to disk failure
- Enable read-ahead on the volume.
- Provision a volume that can accommodate the local cache **and** ingest new files (upload buffer) without ever becoming more than 80% full.
  A general guideline is to use a volume that is at least 1.5 times the size of the data set that you want to hold in local cache. For example, if the expected size of the entire file set is 50 TB and if 10% (5 TB) of that file set will be accessed frequently, then the cache storage volume should have at least 7.5 TB of usable capacity.

> **Note:**
>
> – If the cache size reaches a near-full threshold, any data ingest will result in **out of space** error
>
> in the appliance.

Best Practices – Determining the Cache Size

The local cache of Oracle Cloud Infrastructure Storage Software Appliance serves two roles: ingest cache (upload/write buffer) and read cache. You can specify the maximum size for the read cache. The write buffer will use any remaining available space on the local storage volume and does not have a cache size setting.

The maximum size of the write buffer is an important criterion to determine the cache size. The write buffer size increases when data is uploaded in the appliance. And the write buffer size decreases after the data is transferred to cloud. Write buffer cannot be removed from local cache. When the write buffer uses all the available local cache space, any data ingest will result in **out of space** error in the appliance.

Use the following guidelines to determine the appropriate setting for the ingest cache:

- Identify the amount of data to be uploaded in the appliance. If a large amount of data must be uploaded, the appliance write buffer may reach its maximum. This will lead to I/O failure as the local cache has no space. If the data transfer can be regulated, for example, by pausing after a certain amount of data is transferred or allow the uploads to complete periodically, the local cache space can be increased and I/O failure can be avoided. You can also follow this approach for backup/cron jobs if the local cache space is lesser than the amount of data to be uploaded.
- Calculate the amount of data that would be uploaded on any typical day or a week in the appliance. Also, calculate the amount of data that can be uploaded over a time period, based on the available bandwidth or historical data. The difference between former and latter data quantities should not exceed the write buffer size.
- If the application can handle I/O failure and resume the data transfer, set the write buffer size with the amount of data that you'd like to upload before the cache size decreases.

Configuring the read cache size is necessary only when the appliance must retrieve a significant amount of data from the cloud. Alternatively, you can preserve frequently accessed files in the local cache. Setting aside a large portion of the local cache for read cache might impact the amount of data that can be uploaded before the cache size decreases. Configuring the read cache is optional and depends on the appliance workload.

Use the following guidelines to determine the appropriate setting for the read cache:

- The default limit of the read cache size is the lower of 300 GB or the storage volume size.
- Do not set the read cache maximum to the size of the local storage volume. Doing so would allocate 100% of the volume for read cache and would not leave available capacity for ingest. If there is no available space for new file ingest, then the appliance might stop the data ingest and begin evicting files from the read cache to create space. This severely degrades ingest performance.
- Start with a read cache setting that is 50% of the size of the local storage volume (leaving 50% for ingest). Monitor the available capacity on the local storage volume over time, especially after periods of very high or sustained ingest activity. If the available capacity remains above 30% consistently, consider increasing the read cache size. If the available capacity is consistently below 20%, then consider decreasing the read cache size.

- The general strategy is to set the read cache size to equal the amount of data that you anticipate to be accessed frequently, while leaving enough capacity on the volume for the ingest cache (write buffer).

After you size the cache, you can choose to configure the read cache either while creating the filesystem or later. See Adding a FileSystem and Changing the Properties of a FileSystem.

Best Practices – Encrypting Data Using Keys

You can provide your own RSA asymmetric keys if you've enabled encryption for a filesystem. The symmetric key converts the data to a readable form called *cleartext*. If you lose the keys, you lose the data.

Asymmetric keys: There's a single key pair for every instance of the appliance. The same key pair is used to encrypt information related to local configuration. If you provide an asymmetric key pair, then the key pair is used to encrypt or decrypt the specified filesystem database configuration items. Ensure that the asymmetric keys are backed up.

Symmetric keys: The symmetric key is stored within the local filesystem database. Each filesystem can have its own unique symmetric encryption key. The symmetric key is encrypted using the asymmetric key that's stored locally on the disk.

At any time, you can download a `tar.gz` file containing the details of all the keys stored on the disk.

Key rotation enables data recovery if the appliance fails at any time.

Rotating Keys in the Appliance

1. Log in to the management console and select the filesystem.
2. Provide your asymmetric key pair. The appliance ensures that the keys are valid by encrypting and decrypting randomly generated sample data.
3. If the keys are valid, then they are saved in a temporary location on the disk. The old keys are moved to a backup location on the disk.
4. The database's encrypted configuration items are encrypted again in the filesystem by using the asymmetric key pair.
5. The new keys are saved to a permanent location on the disk.
6. Download the compressed key archive of the encryption keys. The compressed archive includes the new key details as well as the backup keys.

Best Practices – Scalability Recommendations

- Ensure that the number of objects stored in an appliance filesystem doesn't exceed 10 million (10000000). For data sets that consist of more than 10 million objects, ensure that the objects are distributed across multiple filesystems.
- The minimum amount of memory required for any appliance filesystem is 16 GB.
  - For filesystems with the number of files up to 5 million, the required amount of memory is 32 GB.
  - For large filesystems with the number of files up to 10 million, the required amount of memory is 64 GB

- To improve the efficiency of file ingest and cloud upload operations, and to reduce the number of objects in the namespace, bin-pack or zip small files before writing them to the appliance.
- Multiple filesystems can be created on a single appliance. However, for optimal performance, ensure that each filesystem is hosted on a dedicated appliance.

Best Practices – Recommended Workloads and Use Cases

Oracle Cloud Infrastructure Storage Software Appliance is an effective cloud gateway for many workloads. Use the following guidelines to determine whether the appliance is appropriate for your specific use cases and workloads:

- The appliance supports NFSv4 in asynchronous mode and POSIX Sync mode. The POSIX Sync mode is enabled in the appliance by default.
  In the asynchronous mode, there is scope for data loss in the event of a sudden server failure. Avoid using the appliance for workloads and use cases that require synchronous write behavior.
- The appliance is ideal for backup and archive use cases that require the replication of infrequently accessed data to cloud containers.
- Carefully consider use cases that involve frequent changes to existing files. Each time a file is modified and closed, the appliance creates a new version of the file, which is then uploaded to the container or bucket in your service instance, replacing the previous version. The appliance will be less efficient and may not perform optimally for this type of workload.
- Don't run applications and executables directly from the appliance mount points, particularly if the appliance cache is not large enough for all the files that the applications will access. Applications typically create temporary files and modify them often, affecting the operational efficiency of the appliance.

6 Troubleshooting Storage Software Appliance

This section provides solutions for problems you may encounter while using Oracle Cloud Infrastructure Storage Software Appliance.

Topics:

I installed docker and NFS in my host, but can't install Oracle Cloud Infrastructure Storage Software Appliance

1. Add the docker group to the existing groups in your host:
   ```
   sudo groupadd docker
   ```
2. Add your user id to the docker group:
   ```
   usermod -a -G docker username
   ```
3. Shut down your host:
   ```
   shutdown –r now
   ```
4. Log in to your host and run the Oracle Cloud Infrastructure Storage Software Appliance script:
   ```
   sudo ./oscsa-install.sh
   ```

Can't access the management console

1. Check if Oracle Cloud Infrastructure Storage Software Appliance is running:
   ```
   oscsa info
   ```
2. If the appliance is not running, then start the appliance:
   ```
   oscsa up
   ```

   Make a note of the management console port number.

   Sample output

   ```
   Creating OSCSA Volume
   Starting OSCSA [oscsa_gw:1.0]
   Management Console: https://myApplianceHost.example.com:32771

   If you have already configured an OCISSA FileSystem via the Management
   Console,
   you can access the NFS share using the following port.
   ```

```
NFS Port: 32770

Example: mount -t nfs -o vers=4,port=32770 myApplianceHost.example.com:/
OCISSA_filesystem_name/local_mount_point
```

In the sample output,

- *myApplianceHost.example.com* is the appliance host name

-*32771* is the management console port number

3. Check if the appliance is running on `docker` on the appliance host.
4. Check that the management console port number in the output (from `oscsa info`) matches the port you're using to access the management console.
5. Ensure that you are using `https` if you have enabled SSL. By default, SSL is enabled.

I provided my Oracle Cloud Infrastructure Object Storage Classic account details in the management console, but can't create a filesystem

Check your user credentials in your Oracle Cloud Infrastructure Object Storage Classic account and update them in the management console.

Unable to mount a filesystem

1. Check if Oracle Cloud Infrastructure Storage Software Appliance is running:
   `oscsa info`
2. If the appliance is not running, then start the appliance:
   `oscsa up`

   Make a note of the management console port number and NFS port number.

   Sample output

```
Creating OSCSA Volume
Starting OSCSA [oscsa_gw:1.0]
Management Console: https://myApplianceHost.example.com:32771

If you have already configured an OCISSA FileSystem via the Management
Console,
you can access the NFS share using the following port.

NFS Port: 32770

Example: mount -t nfs -o vers=4,port=32770 myApplianceHost.example.com:/
OCISSA_filesystem_name/local_mount_point
```

3. Check if the appliance is running on `docker` on the appliance host.
4. Ensure that the NFS protocol is running:
   `sudo systemctl enable nfs-server`
5. Check that the NFS port number in the output (from `oscsa info`) matches the port you're using to connect to with your NFS client.

Contacting Oracle for Support

1. Go to https://support.oracle.com.
2. In the **Sign In** pane, select **Cloud Portal** as the portal and click **Sign In**.

3. On the **Dashboard** page, click **Create Service Request**.
4. In the **Create Service Request** wizard, do the following:
    a. In the **Service Type** field, select **Oracle Cloud Infrastructure Object Storage Classic**.
    b. In the **Problem Type** field, Select **Issues accessing a Storage Software Appliance** or **Issues installing or upgrading a Storage Software Appliance**, and select the appropriate problem subtype.
5. Follow the prompts in the wizard to complete the service request.

A
Character Restrictions

This section lists the character restrictions when creating and updating resources in your storage service instance in the cloud.

Input Restrictions for FileSystem Name

Container Operation	Input Parameter	Input Restrictions	Unsupported Characters (If any)
• Create your first filesystem • Add a filesystem	Filesystem name	• Only UTF-8 characters • Maximum of 256 bytes • Can start with any character • Cannot contain a slash (/) character because this character delimits the filesystem name	• **Characters**: ', ", `, <, and >. For example: jack's_container, "Future_Use"_files, ToUs<Clients **Strings**: When the name contains `/./` or `/../`; When the name ends with `/.` or `/..`; For example: mymachine/./etc, current/../folder, download_directory/., root/misc/..
• Create a container • Update container metadata	Container custom metadata name (`X-Container-Meta-{name}`)	• Only ASCII characters • Maximum of 128 bytes	The following US-ASCII characters are not supported for use in the container metadata names: • Control characters (octet 0-31) and DEL (127) • Separators (,), <, >, @, ,, ;, :, \, ", /, [,], ?, =, {, }, space, horizontal-tab

Input Restrictions for File Name

Object Operation	Input Parameter	Input Restrictions	Unsupported Characters (If any)
• Upload files to `standard` containers • Upload files to `archive` containers	File name	• Only UTF-8 characters • Maximum of 1061 bytes • Can start with any character	• **Characters**: ', ", `, <, and >. For example: jane's_file, "Hello_World".txt, Send>Customers.pdf • **Strings**: When the name contains `/./` or `/../`; When the name ends with `/.` or `/..`; For example: mymachine/./etc, current/../file, php/., object/..
• Create or replace an object • Update object metadata	Object custom metadata name (`X-Object-Meta-`**`{name}`**)	• Only ASCII characters • Maximum of 128 bytes	The following US-ASCII characters are not supported for use in the object metadata names: • Control characters (octet 0-31) and `DEL` (127) • Separators `(`, `)`, `<`, `>`, `@`, `,`, `;`, `:`, `\`, ", `/`, `[`, `]`, `?`, `=`, `{`, `}`, space, horizontal-tab