Dell EMC OpenStack Data Protection Extension

Version 19.2

Installation and User Guide

302-005-845
Rev 01
November 2019

Copyright © 2016-2019 Dell Inc. or its subsidiaries. All rights reserved.

Dell believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

THE INFORMATION IN THIS PUBLICATION IS PROVIDED "AS-IS." DELL MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WITH RESPECT TO THE INFORMATION IN THIS PUBLICATION, AND SPECIFICALLY DISCLAIMS IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. USE, COPYING, AND DISTRIBUTION OF ANY DELL SOFTWARE DESCRIBED IN THIS PUBLICATION REQUIRES AN APPLICABLE SOFTWARE LICENSE.

Dell Technologies, Dell, EMC, Dell EMC and other trademarks are trademarks of Dell Inc. or its subsidiaries. Other trademarks may be the property of their respective owners. Published in the USA.

Dell EMC
Hopkinton, Massachusetts 01748-9103
1-508-435-1000 In North America 1-866-464-7381
www.DellEMC.com

CONTENTS

Figures 5

Tables 7

Preface 9

Chapter 1 Introduction 13

OpenStack Data Protection Extension 14
Proxy instances 14
Proxy deployment considerations 14
Volume copies 15
Architecture and system requirements 16
Components of the OpenStack DPE product 16
System requirements 16
Resource requirements 17
DNS and time sync requirements 17
Network connections and port usage 17
Multi-domain support 18

Chapter 2 Installation 19

Overview 20
Installation prerequisites 20
OpenStack prerequisites 21
Backup-service project and user 22
Networking prerequisites 24
Avamar prerequisites 25
Installation checklist 25
Unpacking the OpenStack DPE installation files 26
Installing the OpenStack DPE service 27
Configuring the proxy service configuration file 27
Configuring the `proxy-owner1.conf` file 30
Starting the OpenStack DPE service 31
Installing the OpenStack DPE proxy 31
Installing the Avamar REST API 34
Installing the OpenStack DPE API 35
Creating the DPE-API service entity and endpoint on the controller node 38
Installing the CBT driver and agent 39
Installing the CBT driver and agent on RHEL 39
Installing the CBT driver and agent on Ubuntu 39
Creating the DPE service entity and API endpoint on the controller node 40
Installing the OpenStack DPE Horizon plug-in 40
Enabling multi-domain support 42
Post-installation checklist 43
Verifying that the services are running 43
Verifying component communication 44

Verifying the communication ports..... 44

Chapter 3 Using the OpenStack DPE Horizon plug-in 47
Overview of the OpenStack DPE Horizon plug-in..... 48
OpenStack DPE Horizon plug-in interface..... 48
Operations for cloud administrators.....49
Protection providers..... 49
Projects.....51
Group policies..... 53
Schedules..... 53
Retention policies..... 55
Datasets.....57
Group policies..... 59
Operations for project administrators..... 62
Instances 62
Activities.....66
Policies.....67
File-level restoration..... 68
FLR road map.....68
Feature limitations..... 69
FLR session timers.....69
Creating a file-level restore session..... 69
Listing the active FLR sessions and retrieving the FLR session objects.... 70
Using an FLR session to retrieve files from an instance backup.....70
Deleting an FLR session..... 72

Appendix A Additional Information 73
Proxy performance 74
OpenStack availability zones.....74
Cinder drivers.....74
VNX snap copy.....74

Appendix B Troubleshooting 75
OpenStack cloud infrastructure issues.....76
The metadata_host parameter is missing from the nova configuration file.....76
cloud-init is not configured 76
An Oslo-related error causes the OpenStack DPE service to fail.....77
Networking issues.....77
Ports for OpenStack DPE component communication..... 77
Hostname resolution..... 78
OpenStack DPE instances cannot ping the OpenStack controller node... 78
Installation troubleshooting.....79
The OpenStack DPE proxy instance fails to register to the protection provider.....79
The OpenStack DPE API instance did not install completely.....80
Management troubleshooting.....81
OpenStack DPE lists a project as Not Available and you cannot unregister the project..... 81
FLR troubleshooting..... 81
OpenStack DPE FLR cannot mount an XFS file system.....81
OpenStack DPE fails to create an FLR session.....82

FIGURES

1 The OpenStack DPE Horizon plug-in..... 48
2 Batch operations.....49
3 OpenStack File Level Restore page..... 71

TABLES

1 Typographical conventions....10
2 Components of the OpenStack DPE product.... 16
3 Supported operating system versions....17
4 CBT kernel requirements.... 17
5 Resources requirements for components of the OpenStack DPE system....17
6 OpenStack DPE port requirements....17
7 Package requirements.... 25
8 [Service] settings....27
9 [Setting] (common) settings.... 28
10 [Watchdog] settings.... 28
11 [Keystone] settings....30
12 [Tenant] setting....31

Preface

As part of an effort to improve the product lines, revisions of the software and hardware are periodically released. Therefore, some functions that are described in this document might not be supported by all versions of the software or hardware currently in use. The product release notes provide the most up-to-date information on product features.

Contact the technical support professional when a product does not function correctly or does not function as described in this document.

Note: This document was accurate at publication time. To find the latest version of this document, go to Online Support (https://support.EMC.com).

Purpose

This guide describes how to install, configure, and use the OpenStack Data Protection Extension (OpenStack DPE).

Audience

This document is intended for system administrators and programmers who will be installing the OpenStack Data Protection Extension and the OpenStack DPE API. A high degree of knowledge regarding Avamar and OpenStack administration is required.

Revision history

The following table prevents the revision history of this document.

Revision	Date	Description
01	November 15, 2019	GA release of Avamar 19.2

Related documentation

The following publications available at https://support.emc.com provide additional information:

- *OpenStack Data Protection Extension Release Notes*
- *OpenStack Data Protection Extension REST API Getting Started Guide*

Special notice conventions used in this document

These conventions are used for special notices.

DANGER Indicates a hazardous situation which, if not avoided, results in death or serious injury.

WARNING Indicates a hazardous situation which, if not avoided, could result in death or serious injury.

CAUTION Indicates a hazardous situation which, if not avoided, could result in minor or moderate injury.

NOTICE Addresses practices that are not related to personal injury.

Note: Presents information that is important, but not hazard-related.

Typographical conventions

These type style conventions are used in this document.

Table 1 Typographical conventions

Bold	Used for names of interface elements, such as names of windows, dialog boxes, buttons, fields, tab names, key names, and menu paths (what the user specifically selects or clicks)
Italic	Used for full titles of publications that are referenced in text
`Monospace`	Used for: • System code • System output, such as an error message or script • Pathnames, filenames, prompts, and syntax • Commands and options
Monospace italic	Used for variables
`Monospace bold`	Used for user input
[ ]	Square brackets enclose optional values
|	Vertical bar indicates alternate selections - the bar means “or”
{ }	Braces enclose content that the user must specify, such as x or y or z
...	Ellipses indicate nonessential information that is omitted from the example

Where to get help

The Avamar support page provides access to licensing information, product documentation, advisories, and downloads, as well as how-to and troubleshooting information. This information may resolve a product issue before contacting Customer Support.

To access the Avamar support page:

1. Go to https://www.dell.com/support/home/us/en/19.
2. Type a product name in the **Enter a Service Tag, Serial Number, Service Request, Model, or Keyword** search box.
3. Select the product from the list that appears. When you select a product, the **Product Support** page loads automatically.
4. (Optional) Add the product to the **My Products** list by clicking **Add to My Saved Products** in the upper right corner of the **Product Support** page.

Documentation

The Avamar product documentation provides a comprehensive set of feature overview, operational task, and technical reference information. To supplement the information in product administration and user guides, review the following documents:

- Release notes provide an overview of new features and known limitations for a release.
- Technical notes provide technical details about specific product features, including step-by-step tasks, where necessary.
- White papers provide an in-depth technical perspective of a product or products as applied to critical business issues or requirements.

Knowledgebase

The Knowledgebase contains applicable solutions that you can search for either by solution number (for example, KB000xxxxxx) or by keyword.

To search the Knowledgebase:

1. Go to https://www.dell.com/support/home/us/en/19.
2. Under the **Support** tab, click **Knowledge Base**.
3. Type either the solution number or keywords in the search box. Optionally, you can limit the search to specific products by typing a product name in the search box and then selecting the product from the list that appears.

Online communities

Go to Community Network at http://community.EMC.com for peer contacts, conversations, and content on product support and solutions. Interactively engage online with customers, partners, and certified professionals for all products.

Live chat

To engage Customer Support by using live interactive chat, click **Join Live Chat** on the **Service Center** panel of the Avamar support page.

Service Requests

For in-depth help from Customer Support, submit a service request by clicking **Create Service Requests** on the **Service Center** panel of the Avamar support page.

> **Note:** To open a service request, you must have a valid support agreement. Contact a sales representative for details about obtaining a valid support agreement or with questions about an account.

To review an open service request, click the **Service Center** link on the **Service Center** panel, and then click **View and manage service requests**.

Enhancing support

It is recommended to enable ConnectEMC and Email Home on all Avamar systems:

- ConnectEMC automatically generates service requests for high priority events.
- Email Home sends configuration, capacity, and general system information to Customer Support.

Comments and suggestions

Comments and suggestions help to continue to improve the accuracy, organization, and overall quality of the user publications. Send comments and suggestions about this document to DPAD.Doc.Feedback@emc.com.

Please include the following information:

- Product name and version
- Document name, part number, and revision (for example, 01)
- Page numbers
- Other details to help address documentation issues

CHAPTER 1

Introduction

This chapter includes the following topics:

- OpenStack Data Protection Extension 14
- Proxy instances 14
- Volume copies 15
- Architecture and system requirements 16

OpenStack Data Protection Extension

The OpenStack Data Protection Extension (OpenStack DPE) allows backup administrators to manage backup and restore operations for projects in an OpenStack cloud infrastructure. The backup administrator role is performed by an OpenStack administrator who has access rights to projects and associated instances to be backed up or restored. The backup administrator can manage the protection provider (currently an Avamar server), all projects that are protected by the protection providers, and configure backup policies for scheduling backups of a particular project. The backup administrator also manages the backup proxies that are deployed in the OpenStack cloud and are used to perform backup and restore operations.

The OpenStack DPE provides project administrators the ability to manage instances they want to be protected, and browse the backup inventory of a protected instance. The project administrator can then select a backup and restore it to replace the original instance, or restore it to a new location. Progress of the backup or restore operation can be monitored. Project administrators can also add instances to a backup policy created by the backup administrator for scheduled backups.

OpenStack DPE provides a plug-in to integrate with the OpenStack Horizon dashboard for backup management tasks. Support for multiple domains allows the OpenStack DPE to back up tenant projects. OpenStack DPE 19.2 and later supports OpenStack clouds deployed with Kolla.

File-level restoration

OpenStack DPE also delivers support for file-level restoration (FLR) from instance backups to allow users to retrieve files from a backup without the need to complete a full restore operation. This capability provides the ability to restore specific files from a volume in a particular backup or browse the files contained in the backup volumes. The FLR service resides on the OpenStack DPE API VM, and controls FLR session management functions and scheduling.

Note: The FLR feature does not support restoring files directly into a running instance. Instead, the OpenStack DPE provides a graphical interface to browse and download files directly from the image backup.

Proxy instances

Instance backups and restores require deployment of proxy instances within the OpenStack compute nodes. An OpenStack DPE proxy instance is also referred to as `dpe-avamar-proxy`. Proxies run Avamar software inside a Linux instance, and are deployed using a qcow2 image file.

Once deployed, each proxy provides these capabilities:

- Backup of instances and associated volumes attached to the instances.
- Restore of instances and associated volumes attached to the instances.
- Up to eight concurrent backup operations with eight proxies.
- Up to eight concurrent restore operations with eight proxies.
- Up to eight concurrent mixed backup and restore operations with eight proxies.

Proxy deployment considerations

The follow factors should be taken into account by a backup administrator when deploying proxies. The number of proxies to deploy in an OpenStack environment depends on the following:

- The number of availability zones and compute nodes in the protected OpenStack environment.
- The number of projects to be protected
- The number of instances within each project to be protected

- The size of the instances and associated volumes to be protected

Guidelines for proxy deployments

The following guidelines should be followed when deploying proxies:

- There must be at least one proxy per availability zone (for zones that have any instances that should be backed up).
- For optimal performance, as many as one proxy can be installed per Nova compute physical host. Fewer proxies are required if less performance is needed.
- Distribute proxies across physical hosts to optimize network load.
- Each proxy can back up multiple projects. However, suggested practice is to deploy one or more proxies to protect a single project.
- Eight proxies can perform up to eight concurrent backups.
- One proxy can back up approximately 180 GB of data per hour. Proxy performance on page 74 contains further information.
- For scaled environments, multiple proxies should be deployed to ensure that all backups are completed within the backup window of the Avamar server, as defined when creating a scheduled backup policy for a particular project.

ZoneIDs

When a proxy is deployed and configured, the backup administrator must define a ZoneID for each proxy. The ZoneID allows a proxy to be associated to an OpenStack availability zone. OpenStack availability zones on page 74 contains more information.
The following items should be considered when creating ZoneIDs with OpenStack DPE:

- If the OpenStack cloud contains multiple availability zones, a proxy must be deployed for each zone.
- The backup administrator is responsible for managing the ZoneID of all deployed proxies.
- All proxies defined by a ZoneID can service all the projects and instances in the availability zone of the ZoneID.
- When an instance is added for protection, the project administrator must specify a valid ZoneID.
- An Instance can only be associated to a single ZoneID.
- The default ZoneID is named **nova**.
- The ZoneID name is case sensitive.

Examples of ZoneID usage:

- A backup admin must define a single ZoneID (for example, **nova**) and deploy all proxies with the same ZoneID. This means all instances, regardless of which project they are is in, have access to all proxies in **nova**.
- A backup admin can define a ZoneID for each OpenStack availability zone, and associate one or more proxies per ZoneID. This allows the backup administrator to group all instances in an availability zone that use a particular set of proxies.
- A backup admin can define a ZoneID for each project to be protected, and associate one or more proxies for each ZoneID. This allows the backup administrator to define a group of proxies at the project level.

Volume copies

The backup process requires temporary creation of a volume copy. These volume copies are managed by the Cinder drivers of the particular storage array in which the instances and

associated volumes are created. The backup performance can be greatly impacted depending on the Cinder driver's feature that accelerates the process for creating a copied volume (for example, VNX Snap Copy).

Drivers that do not support some form of accelerated copied volume perform a full data copy when creating the volume from a snapshot or cloning a volume, which is time-consuming, especially for large volumes.

Cinder drivers on page 74 contains more information.

Architecture and system requirements

This section details the components of the OpenStack DPE product and system requirements.

Components of the OpenStack DPE product

The following table lists the components of the OpenStack DPE product.

Table 2 Components of the OpenStack DPE product

Component	Description
Avamar server	Also referred to as the Protection Provider.
Data Domain	Provides scalable storage for backups, with features including source data deduplication.
OpenStack DPE API	Management API for OpenStack DPE, provided as a qcow2 image. Includes the following: • API reference documentation via the Swagger UI. • OpenStack DPE file-level restore (FLR) module, which provides FLR capability and a web interface.
OpenStack DPE UI	OpenStack Horizon management plug-in for graphical management of OpenStack DPE.
OpenStack DPE proxy (`dpe-avamar-proxy`)	An Avamar proxy or worker, provided as a qcow2 image and registered to the Avamar server.
OpenStack DPE service (`dpe-proxy-service`)	Integrates with OpenStack controller or compute nodes to provide the ability to clone, attach, and detach Cinder volumes for backup. Provided as an RPM package for RHEL distributions and as a DEB package for Ubuntu.
CBT driver (`dpe-cbt-driver`) and agent (`dpe-cbt-agent`)	Provides the ability to complete incremental backups of changed data since the last full backup.
Avamar REST API	The Avamar REST API provides middleware services between the OpenStack DPE API and the Avamar server. These services provide the interface to the Avamar server.

System requirements

Review the following requirements before installation.

The OpenStack DPE supports the following operating systems on the compute node:

Table 3 Supported operating system versions

RHEL	CentOS	Ubuntu
7.6	7.6-1810	16.04 LTS
7.7	7.7-1908	

The CBT driver supports the following kernel versions:

Table 4 CBT kernel requirements

RHEL	Ubuntu 16.04
3.10.0-957.el7.x86_64	4.15.0-54-generic
3.10.0-1062.el7.x86_64	

Use `uname -r` to determine the kernel version on the compute node.

Before upgrading the kernel on the compute node, determine whether the OpenStack DPE supports the new kernel and review the notes in Installing the CBT driver and agent on page 39.

Resource requirements

The following components should meet these resource requirements:

Table 5 Resources requirements for components of the OpenStack DPE system

Component	Virtual CPUs	Virtual disk size	Virtual RAM
OpenStack DPE proxy	2	16 GB	2 GB
OpenStack DPE API	2	16 GB	6 GB[a]
Avamar REST API	2	4 GB	6 GB[a]

a. Additional RAM may provide greater performance in high-demand environments.

DNS and time sync requirements

Secure, reliable operation of OpenStack DPE depends on a network that is also secure and reliable, and that supports forward and reverse lookup of hostnames, network time service, and various other services. Your network must meet these requirements before you begin installing OpenStack DPE.

Network connections and port usage

The following table provides a summary of the ports that are required by the OpenStack DPE.

Table 6 OpenStack DPE port requirements

Initiator	Target	Protocol	Port	Notes
FLR clients	OpenStack DPE API	TCP	443	FLR user interface.
OpenStack DPE proxy	`dpe-proxy-service`	TCP	1947	Default binding port.

Table 6 OpenStack DPE port requirements (continued)

Initiator	Target	Protocol	Port	Notes
OpenStack DPE service	OpenStack DPE API (`dpe-api`)	TCP	8443	Default binding port.
Avamar REST API	Avamar server	TCP	8543	Default binding port.
Proxy (`avagent`)	Avamar server	TCP	28001	
Avamar server	Proxy instance	TCP	28000–30109	Various ports are required for communication between the Avamar server and the `dpe-avamar-proxy` service. See the *Avamar Product Security Guide* for details.

The OpenStack DPE graphical management plug-in uses the existing OpenStack Horizon HTTP connections.

The default security group disables HTTP connections to the OpenStack DPE API on port 8080. Using HTTPS on port 8443 is strongly advised.

Multi-domain support

OpenStack DPE supports backing up projects across different domains that can be used to provide additional access control.

For example, domains can be used to segregate multiple tenants. After you configure Horizon for domain support, the OpenStack DPE allows administrators to independently manage projects in each domain, using the permissions defined in OpenStack.

CHAPTER 2

Installation

This chapter includes the following topics:

- Overview 20
- Installation prerequisites 20
- Installation checklist 25
- Unpacking the OpenStack DPE installation files 26
- Installing the OpenStack DPE service 27
- Starting the OpenStack DPE service 31
- Installing the OpenStack DPE proxy 31
- Installing the Avamar REST API 34
- Installing the OpenStack DPE API 35
- Installing the CBT driver and agent 39
- Installing the OpenStack DPE Horizon plug-in 40
- Enabling multi-domain support 42
- Post-installation checklist 43

Overview

This section provides an overview of the installation of the OpenStack DPE product.

OpenStack DPE requires the deployment of five components in the OpenStack cloud infrastructure.

- OpenStack DPE service (`dpe-proxy-service`), provided as an rpm for RHEL or deb for Ubuntu installation file.
- OpenStack DPE proxy (`dpe-avamar-proxy`), provided as a qcow2 image.
- OpenStack DPE API (`dpe-api`), provided as a qcow2 image.
- OpenStack DPE Horizon plug-in (`dpe-horizon-plugin`), which provides a graphical interface for backup management services.
- Avamar REST API, provided as a qcow2 image.

The Avamar REST API provides middleware services between the OpenStack DPE API and the Avamar server. These services provide the interface to the Avamar server.

Overview of the installation:

1. Create a backup-service project and backup administrator user for hosting the OpenStack DPE components.
2. Install and configure the OpenStack DPE service on each of the OpenStack controller nodes (multiple nodes for High Availability support), or on any compute node.
3. Install the CBT driver and agent on each of the OpenStack compute nodes.
4. Within the **avamar** project, deploy and configure one or more OpenStack DPE proxy instances.
5. Within the **avamar** project, deploy and configure an OpenStack DPE API instance (REST API).
6. Within the **avamar** project, deploy and configure an Avamar REST API instance.
7. Install the OpenStack DPE Horizon plug-in in the container service for OpenStack Horizon.

   The OpenStack Horizon container service normally runs on a controller node.

Overview of configuration:

Note: Refer to the *OpenStack Data Protection Extension REST API Getting Started Guide* for more details on configuring the OpenStack DPE API.

1. After installation is complete, the OpenStack DPE Horizon plug-in or OpenStack DPE API can be used to register an Avamar server as a protection provider.
2. OpenStack projects are registered and associated with one protection provider.
3. Instances within those projects are registered for protection.
4. Policies may also be created for scheduled backups.

Installation prerequisites

Before installing the OpenStack DPE, ensure that you have met all prerequisites.

OpenStack prerequisites

OpenStack DPE is designed to work with OpenStack Queens and Kolla deployment, and is verified against the Red Hat OpenStack Platform 13. The OpenStack DPE supports dockerized deployment.

nova.conf:

The OpenStack DPE service looks for certain parameters in the `/etc/nova/nova.conf` file on the node that runs the OpenStack DPE service. Specifically, the OpenStack DPE service looks for `metadata_host` and related parameters to discover instances and obtain the paths to compute services.

Some distributions do not provide this folder, while other distributions comment out some of these parameters. Before you begin, ensure that this file exists and that the permissions permit read access for all users.

In the `Default` section, configure the following parameters:

```
metadata_host = <controller-node-ip>
metadata_listen_port = 8775
metadata_listen = <controller-node-ip>
transport_url = rabbit://<user>:<credential>@<rabbitmq-ip>:5672
```

where:

- *<controller-node-ip>* is the FQDN or IP address of the OpenStack controller node. If the controller node has a high-availability (HA) configuration, use the HA FQDN or IP address instead.
- *<user>* is the RabbitMQ username that the OpenStack DPE should use.
- *<credential>* is the RabbitMQ password that the OpenStack DPE should use.
- *<rabbitmq-ip>* is the FQDN or IP address of the RabbitMQ server.

In the `Neutron` section, configure the following parameters:

```
metadata_proxy_shared_secret = <shared-secret>
service_metadata_proxy = true
```

where *<shared-secret>* is the value that allows the OpenStack DPE to connect.

cloud-init:

`cloud-init` must be correctly configured, or deployment of OpenStack DPE components fails. By default, certain settings are not enabled which are required for `cloud-init` to function.

Identify the Neutron configuration path and locate the file `dhcp_agent.ini`. For example, under RHEL, the file resides at `/var/lib/configdata/puppet-generated/neutron/etc/neutron/` on the controller node.

Locate the following settings in the `DEFAULT` section of `dhcp_agent.ini` and ensure that all settings are set to `true`:

```
enable_isolated_metadata = true
enable_metadata_network = true
```

After changing these settings, restart the Neutron services:

1. Type **`docker ps | grep neutron`**.
2. Note the container names for the Neutron container services.
3. Restart the Neutron container services by typing **`docker restart <container>`**.

   where *<container>* is the name of each container service. Repeat this step for all Neutron container services.

Cinder volume API v2:

OpenStack DPE requires Cinder volume API v2. To verify the Cinder volume API version, type the following command:

```
openstack endpoint list | grep volume
```

Output similar to the following appears:

```
0e67c64387fa4356bb9738c839e150ab | RegionOne | cinderv2     | volumev2     |
True    | internal  | http://10.3.4.5:8776/v2/%(tenant_id)
```

Inspect the output for the values `cinderv2` and `volumev2`. If Cinder volume API v2 is not present, install it before proceeding.

Backup-service project and user

A backup-service project and backup administrator user account must be created for hosting the OpenStack DPE components. The following guidelines should be followed when creating this project and user account:

- Create a backup-service project solely for use by the backup service. The suggested name for this project is **avamar**, for clarity purposes.
  - The backup-service project name will go into the backup-service configuration.
  - Proxy instances will run within the backup-service project.
  - You cannot perform backups of instances within the backup-service project.
  - Temporary volumes and snapshots for backup and restore operations will be attached to the backup-service project.
  - Quotas for the backup-service project should be fairly large.
    - The number of volumes quota should be greater than the maximum number of volumes in all active backups and restores. One proxy instance can perform a single backup or restore. Eight proxy instances can perform up to eight concurrent backups or restores. Queued (but not active) backup or restore jobs do not count against this limit.
    - The total size of the volumes quota should be greater than the total size of all volumes in all active backups and restores. Queued jobs do not count against this limit.
- The suggested name for the user is backup_admin, for clarity purposes. Do not use an existing user.
  - The username and password will go into the backup-service configuration file.
  - The user must have the admin role for the backup-service project.
  - The user must have the member role of all projects with instances you wish to back up.

Example backup-service project and user configuration

The following task describes how to configure an example backup-service project, named **avamar**, and a user, named **backup_admin**.

Before you begin

In the root user home directory of the OpenStack controller node, there should be a `keystonerc_admin` or `openrc` file (or a similar file for various different distributions). This file contains the environment variables that are needed by the OpenStack CLI tools, and its contents should resemble the following:

```
# unset OS_SERVICE_TOKEN
# export OS_USERNAME=admin
# export OS_PASSWORD=yourpassword
# export OS_AUTH_URL=http://yourhost:5000/v3.0
# export PS1='[\u@\h \W(keystone_admin)]\$ '
```

```
# export OS_TENANT_NAME=admin
# export OS_REGION_NAME=RegionOne
```

These environment variables must be exported by using the shell's `source` command:

```
source keystonerc_admin
```

Procedure

1. Create a backup-service project by performing the following steps at the command prompt:

   a. Create a backup service tenant:

   ```
   openstack project create avamar
   ```

   ```
   +-------------+----------------------------------+
   | Field       | Value                            |
   +-------------+----------------------------------+
   | description | None                             |
   | enabled     | True                             |
   | id          | f1210c6bc95043e8acbe10b2c29675ac |
   | name        | avamar                           |
   +-------------+----------------------------------+
   ```

   b. Determine the volume quotas:

   ```
   cinder quota-show avamar
   ```

   ```
   +------------------+-------+
   |     Property     | Value |
   +------------------+-------+
   | backup_gigabytes |  1000 |
   |     backups      |   10  |
   |    gigabytes     |  1000 |
   | gigabytes_iscsi  |   -1  |
   |    snapshots     |   10  |
   | snapshots_iscsi  |   -1  |
   |     volumes      |   10  |
   |  volumes_iscsi   |   -1  |
   +------------------+-------+
   ```

   The OpenStack DPE requires enough quotas for the volume number and size to allow for the largest possible number and size of concurrent backups that will be executed. For example, the next command changes the number of volumes to 100 and the volume size to 1000 GB.

   c. Set the necessary volume number and size:

   ```
   cinder quota-update --volumes 100 --gigabytes 1000 avamar
   ```

   ```
   +------------------+-------+
   |     Property     | Value |
   +------------------+-------+
   | backup_gigabytes |  1000 |
   |     backups      |   10  |
   |    gigabytes     |  1000 |
   | gigabytes_iscsi  |   -1  |
   |    snapshots     |   10  |
   | snapshots_iscsi  |   -1  |
   |     volumes      |  100  |
   |  volumes_iscsi   |   -1  |
   +------------------+-------+
   ```

   Note: `backup_gigabytes` and `backups` quotas are not related to OpenStack DPE requirements.

2. Create the backup_admin user by typing the following command on one line:

   ```
   openstack user create --domain domain_id --project avamar backup_admin
   ```

   where *domain_id* is an appropriate domain for the backup project.

3. Add roles to the backup_admin user by performing the following steps:

   a. List the available projects:

   openstack project list

```
+----------------------------------+--------------------+
| ID                               | Name               |
+----------------------------------+--------------------+
| 15e3219df829442790495ba17d8cef37 | services           |
| d224c02464c147dfa8e86f6c1209a814 | admin              |
| daf2ec458c4548e6be18f4aff3d01948 | project-ABC        |
| ec0ad980acb64dfa98966d1fb899ed23 | project-XYZ        |
| f1210c6bc95043e8acbe10b2c29675ac | avamar             |
+----------------------------------+--------------------+
```

   b. List the available roles:

   openstack role list

```
+----------------------------------+---------------+
| ID                               | Name          |
+----------------------------------+---------------+
| 6b9489f52a364dceb81ef5e957097b3c | admin         |
| 773822c030ef4dff93997e92e56ab2c2 | ResellerAdmin |
| 9fe2ff9ee4384b1894a90878d3e92bab | _member_      |
| da2f7af3bcf042459294156a9e4d12fe | SwiftOperator |
+----------------------------------+---------------+
```

   c. Add the backup_admin user as an admin to the **avamar** project:

   openstack role add --project avamar --user backup_admin admin

```
+-------+----------------------------------+
| Field | Value                            |
+-------+----------------------------------+
| id    | 6b9489f52a364dceb81ef5e957097b3c |
| name  | admin                            |
+-------+----------------------------------+
```

   d. Add the backup_admin user as a member to other projects:

   Example 1:

   openstack role add --project project-ABC --user backup_admin _member_

```
+-------+----------------------------------+
| Field | Value                            |
+-------+----------------------------------+
| id    | 9fe2ff9ee4384b1894a90878d3e92bab |
| name  | _member_                         |
+-------+----------------------------------+
```

   Example 2:

   openstack role add --project project-XYZ --user backup_admin _member_

```
+-------+----------------------------------+
| Field | Value                            |
+-------+----------------------------------+
| id    | 9fe2ff9ee4384b1894a90878d3e92bab |
| name  | _member_                         |
+-------+----------------------------------+
```

Networking prerequisites

The following prerequisites apply to networking issues with the OpenStack DPE:

- OpenStack DPE supports Neutron Configuration for FLAT, VLAN, GRE, & VXLAN segmentation.

- This documentation may refer to a flat network (which may also be referred to as a provider network) for the **avamar** project to use.
- If deploying proxies through NAT or floating IP address configuration, the IP address assigned to the proxy instances must be registered to your DNS (that is, the fully-qualified domain name must be resolvable to an IP address based on `nslookup`).
- The proxy instance must be able to connect to the Avamar server and the OpenStack DPE service running on the controller node or compute node.
- The Avamar server must be able to connect to the proxy instance.
- The Avamar REST API must be able to connect and issue commands to the Avamar server.
- The OpenStack DPE API must be able to connect to the Avamar server.
- The OpenStack DPE service and the OpenStack DPE API must be able to connect to OpenStack.

Avamar prerequisites

The following Avamar prerequisites apply to installations of OpenStack DPE:

- An Avamar server that is installed and running Avamar 19.2 or later.
- Data Domain can be used as back-end storage and must be configured within the Avamar Administrator GUI.
- The Data Domain must run a version of DDOS that is supported by the Avamar server.

Only one protection provider is supported per OpenStack installation.

Installation checklist

The following is a checklist for installation activities of the various components of OpenStack DPE.

OpenStack DPE service installation checklist:

- The OpenStack DPE service requires the installation of the following packages on its host OpenStack controller or compute node:

Table 7 Package requirements

Package	Version	Notes
python-suds	0.4 or greater	
python-requests	2.7.0 or greater	
python-bottle	0.12.6 or greater	
python-support	1.0.15 or greater	Only required for installation on Ubuntu 16.04. This package cannot be installed automatically by using `apt-get install`. Instead, obtain the package from Launchpad here.

- Configure any installed firewall to allow communication on the ports that are listed in Network connections and port usage on page 17.

OpenStack DPE proxy installation checklist:

- Configure the **avamar** project according to instructions in Example backup-service project and user configuration on page 22.

- Configure a flat or VLAN external network for the **avamar** project to use.
- Configure Keystone environment variables for admin access to the **avamar** project by adding the following to the `/root/keystonerc_backup_admin` file on the OpenStack controller node:

```
unset OS_SERVICE_TOKEN
export OS_USERNAME=backup_admin
export OS_PASSWORD=changeme
export OS_AUTH_URL=http://keystone.example.com:5000/v3.0
export PS1='[\u@\h \W(keystone_avamar)]\$ '
export OS_TENANT_NAME=avamar
export OS_REGION_NAME=RegionOne
```

OpenStack DPE API installation checklist:

- Configure the **avamar** project according to instructions in Example backup-service project and user configuration on page 22.
- The OpenStack DPE API instance can use either the same flat network that the OpenStack DPE proxy uses, or it can use a floating IP address on a private network.
- The OpenStack DPE API connect to the OpenStack DPE service on a single TCP port. The default port number is 8443. Configure any installed firewall to allow communication on this port.

Unpacking the OpenStack DPE installation files

Procedure

1. From the Online Support website, obtain a copy of the OpenStack DPE installation ZIP file (`OpenStack_DPE_`*`version`*`.zip`), where *version* is the version of the OpenStack DPE.
2. Use a standard unzip program to unpack the installation files to a temporary location.

Results

The OpenStack DPE installation ZIP file contains the following files:

- The OpenStack DPE service installation package for Ubuntu

  (`dpe-proxy-service-`*`version`*`.deb`)
- The OpenStack DPE service installation package for RedHat

  (`dpe-proxy-service-`*`version`*`.rpm`)
- The OpenStack DPE proxy image file

  (`dpe-avamar-proxy.`*`version`*`.qcow2`)
- The OpenStack DPE API image file

  (`dpe-api.`*`version`*`.qcow2`)
- The Avamar REST API image file

  (`avamar_rest_api.x86_64-`*`version`*`.qcow2`)
- The CBT driver installation package for RedHat

  (`dpe-cbt-driver-linux-el7-x86_64-`*`version`*`.rpm`)
- The CBT driver installation package for Ubuntu 16.04

  (`dpe-cbt-driver-linux-ub16-x86_64-`*`version`*`.deb`)
- The CBT agent installation package for RedHat

  (`dpe-cbt-agent-`*`version`*`.noarch.rpm`)

- The CBT agent installation package for Ubuntu

  (`dpe-cbt-agent_version_all.deb`)
- The OpenStack DPE Horizon plug-in installation package

  (`dpe-horizon-plugin-version.tar.gz`)

The following sections detail how to install and configure OpenStack DPE components by using these files.

Installing the OpenStack DPE service

This section provides information about installing and configuring the OpenStack DPE service.

Procedure

1. Install the OpenStack DPE service RPM or DEB file from the temporary location that is described in Unpacking the OpenStack DPE installation files on page 26:
   - On RHEL, install the RPM by typing the following command:

     `yum localinstall dpe-proxy-service-version.rpm`
   - On Ubuntu, install the DEB by typing the following command:

     `dpkg -i dpe-proxy-service-version.deb`

   where *version* is the version of the OpenStack DPE.

Configuring the proxy service configuration file

The proxy service configuration (`/etc/avamar/proxy-service1.conf`) contains configurable settings that are related to the proxy service. All of these settings have default values, and therefore the `proxy-service1.conf` file can be empty or have all values commented out.

About this task

The following tables describe each setting in the `proxy-service1.conf` file.

Table 8 [Service] settings

Parameter	Description
`bind`	The network interface to which the proxy service binds. Default is `0.0.0.0` (which is usually the correct setting).
`port`	The port to which the proxy service binds. Default is `1947`.
`server`	The type of Python server implementation to use. Default is `paste`.
`ssl_pem`	The path to an SSL `.pem` file (or `*` to accept all). Required to use SSL, otherwise not required. Default is `null`.
`debug`	Enables troubleshooting output from the server. Default is `false`.
`max_body_size`	The maximum body size of the client web request for the Python bottle server. Default is `1048576` bytes.

Table 9 [Setting] (common) settings

Parameter	Description
`interface`	Selects which set of OpenStack endpoints are expected on the same network as the proxy service. Depending on the network topology, only public or internal endpoints may be visible. Highly-secure customer sites will most likely install the OpenStack DPE on the internal network. For most other configurations, the public endpoints are appropriate. Defaults to `public`.
`timeout_volume_clone`	The maximum time (in seconds) that the proxy service waits for a volume clone operation to complete. Default is `7200` (two hours).
`timeout_volume_create`	The maximum time (in seconds) that the proxy service waits for a volume create operation to complete. Default is `3600` (one hour).
`timeout_volume_detach`	The maximum time (in seconds) that the proxy service waits for a volume detach operation to complete. Default is `3600` (one hour).
`timeout_proxy_request`	The maximum time (in seconds) that the proxy plug-in waits for a response from the proxy service. Default is `7200` (two hours).

Table 10 [Watchdog] settings

Parameter	Description
`interval_poll`	The interval (in seconds) at which the watchdog wakes. Default is `3`.
`interval_inspect`	The interval (in seconds) at which the watchdog inspects tenants and volumes for stray volumes. Default is `3600` (one hour).
`volume_expiration`	How old (in seconds) a volume must be before it is considered for cleanup. Default is `86400` (24 hours).

Procedure

1. Using a text editor, edit the `proxy-service1.conf` file and change the values for any applicable settings.

   For any value that you set in the file, remember to uncomment (remove the `#` character) at the beginning of the line.

   Note: There is no enforcement mechanism for reasonable values in the `proxy-service1.conf` file. The defaults are generally reasonable. Choose over-rides carefully.

Example 1 Sample `proxy-service1.conf` file

```
[Service]
# How should the proxy-service appear as a web-service.

# Specify if you want the proxy-service bound to a particular
```

Example 1 Sample `proxy-service1.conf` file (continued)

```
interface.
#bind=0.0.0.0

# If running more than one proxy-service, each bind/port must be
unique.
#port=1947

# The default WSGI server to use for the proxy-service.
#server=paste

# Must specify if use of SSL is wanted between proxy-instance and
proxy-service.
#ssl_pem=

# Primarily for development (or debugging).
#debug=False

# Max body size for bottle server(bytes). set default as 1M(1024*1024)
max_body_size=1048576

[Settings]
# Control over aspects of the proxy-service behavior.

# Which OpenStack API interfaces should we use?
#interface=public

# Allow time needed for large volumes and slow/dumb storage.
#timeout_volume_clone=7200

# Allow time needed for slow storage.
#timeout_volume_create=3600

# Primarily a workaround for the Kilo (and prior) bug with
multipathed storage.
#timeout_volume_detach=3600

# How long should the proxy-instance wait for the processing of a
request by the proxy-service?
#timeout_proxy_request=7200

[Watchdog]
# Control over the background watchdog that (eventually) cleans up
volumes not processed properly.

# How often should the watchdog wake (in seconds)?
#interval_poll=3

# How often should the watchdog inspect tenants/volumes (in seconds)?
#interval_inspect=3600

# How old a disused/marked volume must be before forced deletion?
#volume_expiration=86400
```

Configuring the proxy service for SSL

This section describes how to configure the `dpe-proxy-service` for SSL. This is an optional task.

Procedure

1. Create a valid `.pem` file:

   a. Generate a new key by typing the following command:

   `openssl genrsa 1024 > host.key`

b. Change permissions on the key by typing the following command:

```
chmod 400 host.key
```

c. Create a certificate by typing the following command:

```
openssl req -new -x509 -nodes -sha1 -days 365 -key host.key > host.cert
```

d. Create the `.pem` file from the certificate and key by typing the following command:

```
cat host.cert host.key > host.pem
```

e. Change permissions on the `.pem` file by typing the following command:

```
chmod 400 host.pem
```

f. Change ownership of the `.pem` file by typing the following command:

```
chown avamar:avamar host.pem
```

2. Use a text editor to edit the `proxy-service1.conf` file and change the values of these two settings:

[Service] `server`

Set to `paste`. This is the `wsgi` server that OpenStack uses and should be available if the proxy service is installed on the OpenStack controller node or compute node. Otherwise, the server needs to be installed on the system on which the proxy service runs.

[Service] `ssl_pem`

Set to the location of a valid `.pem` file. For example, set this to **`ssl_pem="/etc/avamar/host.pem"`**.

Configuring the `proxy-owner1.conf` file

The `proxy-owner1.conf` specifies the owner of proxy instances and working quota for use by the proxy service.

About this task

The following tables describe each setting in the `proxy-owner1.conf` file.

Table 11 [Keystone] settings

Parameter	Description
`href`	Required. The URL of the OpenStack Keystone implementation. From this URL, the OpenStack DPE can discover all other OpenStack services.
`username`	Required. The OpenStack user login that is used for both client registration and the proxy service. This value should be the backup_admin user, as described in Backup-service project and user on page 22.
`password`	Required. The password for the backup_admin user.
`domain_id`	The ID of the domain to which the backup_admin user belongs, as listed in Horizon under **Identity** > **User backup_admin**. If no information is listed there for **Domain ID**, leave this value as `default`.
`ssl_verify`	Determines whether to verify the SSL certificate and host of the OpenStack Keystone implementation. Setting this value to `false` disables verification.

Table 11 [Keystone] settings (continued)

Parameter	Description
	Otherwise, set this value to the path of a valid SSL certificate for the Keystone host.

Table 12 [Tenant] setting

Parameter	Description
`name`	Required. The name of the backup service project that contains proxies and working volume quotas.

Procedure

1. Using a text editor, edit the `proxy-owner1.conf` file and change the values for any applicable settings.

   For any value that you set in the file, remember to uncomment (remove the `#` character) at the beginning of the line.

Example 2 A sample `proxy-owner1.conf` file

```
[Keystone]
#href=http://localhost:5000/
#username=backup_admin
#password=secret
#domain_id=default
#ssl_verify=/path/to/sslcert.pem

[Tenant]
#name=avamar
```

Starting the OpenStack DPE service

After installing and configuring the OpenStack DPE service, use this procedure to start it.

Procedure

1. Start the OpenStack DPE service by typing the following commands:

   ```
   systemctl enable dpe-proxy-service
   systemctl start dpe-proxy-service
   ```

Installing the OpenStack DPE proxy

This section provides information about installing and configuring the OpenStack DPE proxy.

About this task

Note: Space limitations may prevent some commands in this task from being printed on one line. Use the clipboard, where available, to copy and paste commands, and ensure that you type all commands as directed.

Procedure

1. Create an image in the **avamar** project of the OpenStack cloud using the OpenStack DPE proxy image file from the temporary location that is described in Unpacking the OpenStack DPE installation files on page 26. You can use Horizon or type the following command on one line at the command prompt:

   `openstack image create --disk-format qcow2 --file dpe-avamar-proxy.`*`version`*`.qcow2 --min-disk 16 --min-ram 2048 --private avamar-proxy`

   where *version* is the version of the OpenStack DPE.

2. Create a flavor for the **avamar** proxies by logging in to Keystone as the backup_admin user and typing the following command on one line at the command prompt:

   `openstack flavor create --disk 16 --ram 2048 --vcpus 2 --swap 0 --ephemeral 0 --private avamar-proxy`

   Note the unique ID that OpenStack returns for the new flavor.

3. Assign the flavor to the project by typing the following command at the command prompt:

   `nova flavor-access-add` ***`FLAVOR_ID PROJECT_ID`***

   where:

   - *FLAVOR_ID* is the unique ID for the new flavor.
   - *PROJECT_ID* is the **avamar** project ID.

4. Use one of the following commands to create or import a keypair for ssh access:

   Note: You can also use the same keypair as the OpenStack DPE API image.

   - To import a keypair, type the following command on one line:

     `openstack keypair create --public-key ~/.ssh/id_rsa.pub avamar_kp`

   - To create a keypair, type the following command:

     `openstack keypair create keypair_name > private_key.pem`

5. Create the security group and rules that define the access that the proxy instance permits to the network. The following uses the proxy-security group name:

```
SECURITY_GROUP_NAME=proxy-security
openstack security group create $SECURITY_GROUP_NAME
openstack security group rule create --proto icmp --src-ip \
   0.0.0.0/0 $SECURITY_GROUP_NAME
openstack security group rule create --proto tcp --src-ip \
   0.0.0.0/0 --dst-port 22:22 $SECURITY_GROUP_NAME
openstack security group rule create --proto tcp --src-ip \
   0.0.0.0/0 --dst-port 28000:30100 $SECURITY_GROUP_NAME
```

6. Use the following script to deploy an instance of the OpenStack DPE API inside the **avamar** project using the image that was created in step 1 on page 32 as the boot source. Do not create a volume from the image. Replace all variables in the script with valid parameter values.

```
#Username of 'avamar' project
USERNAME_PROXY=backup_admin

#Password of backup admin of 'avamar' project
PASSWORD_PROXY=changeme

#IP or FQDN of controller node
HOSTNAME_CONTROLLER=10.2.3.xx
```

```
#IP or FQDN of Avamar server
HOSTNAME_AVAMAR=10.2.3.xx

#Default ZoneID is 'nova'.  Set according to the
#Guidelines defined in the Proxy section
ZONEID=nova

URL_PROXY_SERVICE="http://$USERNAME_PROXY:$PASSWORD_PROXY@
$HOSTNAME_CONTROLLER:1947"

PROXY_FLAVOR=avamar-proxy
PROXY_NAME=proxy-1
PROXY_CONFIG_FILE=/tmp/user-data
IMAGE_PROXY=avamar-proxy

KEYPAIR_NAME=avamar_kp
SECURITY_GROUP_NAME=proxy-security

cat > $PROXY_CONFIG_FILE <<XXXX
#cloud-config
bootcmd:
- /tmp/postconfig.sh $HOSTNAME_AVAMAR /clients/$ZONEID \
  "$URL_PROXY_SERVICE"
XXXX

openstack server create \
  --image $IMAGE_PROXY \
  --key-name $KEYPAIR_NAME \
  --flavor $PROXY_FLAVOR \
  --security-group $SECURITY_GROUP_NAME \
  --user-data $PROXY_CONFIG_FILE \
  $PROXY_NAME
```

Note: The ZoneID `nova` in the script must match the ZoneID field that is used to register instances via the OpenStack DPE API. If the instance and proxy are not in the same availability zone, backup/restore operations fail. The default ZoneID is `nova`.

a. If the OpenStack DPE proxy service is not installed on the controller node, change the `URL_PROXY_SERVICE` parameter to:

```
URL_PROXY_SERVICE="http://$USERNAME_PROXY:$PASSWORD_PROXY@$<node-
ip>:1947"
```

where *<node-ip>* is the hostname or IP address of the node on which the OpenStack DPE proxy service is installed.

7. If you changed the password for the backup_admin user in OpenStack, perform the following substeps to update the encrypted URL for the OpenStack DPE service:

a. SSH to the OpenStack DPE proxy instance.

b. Generate a new encrypted URL by typing the following command:

/usr/local/avamarclient/bin/ostackimage --encrypt_dpservice_url=http://backup_admin:*new-password*@*node*:1947

where *node* is the hostname or IP address of the node on which the OpenStack DPE service service is installed.

Output similar to the following appears:

```
http://backup_admin:zCsKdekSODA=@10.2.3.4:1947
```

Record the output.

c. Using a Linux text editor, such as `vi`, open `/usr/local/avamarclient/var/ostackimage.cmd`.

d. Replace the existing OpenStack DPE service URL with the value that you recorded in the previous substep.

e. Save and close the file.

After you finish

NOTICE Do not change the value of `DHCLIENT_SET_HOSTNAME` in `/etc/sysconfig/network/dhcp` to anything other than `no`. Changing this value may cause the DHCP client to change the OpenStack DPE proxy hostname.

Installing the Avamar REST API

This section provides information about installing and configuring the Avamar REST API.

About this task

Note: Space limitations may prevent some commands in this task from being printed on one line. Use the clipboard, where available, to copy and paste commands, and ensure that you type all commands as directed.

Procedure

1. Create an image in the **avamar** project of the OpenStack cloud, using the Avamar REST API image file from the temporary location that is described in Unpacking the OpenStack DPE installation files on page 26, by typing the following command on one line at the command prompt:

   ```
   openstack image create --disk-format qcow2 --file avamar_rest_api.x86_64-version.qcow2 --min-disk 4 --min-ram 6144 --public avamar-rest-api
   ```

   where *version* is the version of the OpenStack DPE.

2. Create a flavor for the **avamar** REST API by logging in to Keystone as the backup_admin user and typing the following command on one line at the command prompt:

   ```
   openstack flavor create --disk 4 --ram 6144 --vcpus 2 --swap 0 --ephemeral 0 --private avamar-rest-api
   ```

   Note the unique ID that OpenStack returns for the new flavor.

3. Assign the flavor to the project by typing the following command at the command prompt:

   ```
   nova flavor-access-add FLAVOR_ID PROJECT_ID
   ```

   where:

   - *FLAVOR_ID* is the unique ID for the new flavor.
   - *PROJECT_ID* is the **avamar** project ID.

4. Use one of the following commands to create or import a keypair for SSH access:

   Note: You can also use the same keypair as the proxy image.

   - To import a keypair, type the following command on one line:

     ```
     openstack keypair create --public-key ~/.ssh/id_rsa.pub avamar_kp
     ```

   - To create a keypair, type the following command:

     ```
     openstack keypair create --public-key > keypair.pem avamar_kp
     ```

5. Create the security group and rules that define the access that the instance permits to the network. The following example uses the `avamar-rest-api-security` group name:

```
SECURITY_GROUP_NAME=avamar-rest-api-security
openstack security group create $SECURITY_GROUP_NAME
openstack security group rule create --proto icmp --src-ip \
  0.0.0.0/0 $SECURITY_GROUP_NAME
openstack security group rule create --proto tcp --src-ip \
  0.0.0.0/0 --dst-port 22:22 $SECURITY_GROUP_NAME
openstack security group rule create --proto tcp --src-ip \
  0.0.0.0/0 --dst-port 8543:8543 $SECURITY_GROUP_NAME
openstack security group rule create --proto tcp --src-ip \
  0.0.0.0/0 --dst-port 443:443 $SECURITY_GROUP_NAME
```

6. Use the following script to deploy an instance of the Avamar REST API inside the **avamar** project using the image that was created in step 1 on page 35 as the boot source. Do not create a volume from the image.

```
IMAGE=avamar-rest-api
FLAVOR=avamar-rest-api
INSTANCE_NAME=avamar-rest-api
CONFIG_FILE=/tmp/user-data
KEYPAIR=avamar_kp
SECURITY_GROUP_NAME=avamar-rest-api-security

cat > $CONFIG_FILE <<XXXX
#cloud-config
bootcmd:
 - /tmp/avamar_rest_postconfig.sh
XXXX

openstack server create \
  --image $IMAGE \
  --key-name $KEYPAIR \
  --security-group $SECURITY_GROUP_NAME \
  --flavor $FLAVOR \
  --user-data $CONFIG_FILE \
  $INSTANCE_NAME
```

Installing the OpenStack DPE API

This section provides information about installing and configuring the OpenStack DPE API.

About this task

Note: Space limitations may prevent some commands in this task from being printed on one line. Use the clipboard, where available, to copy and paste commands, and ensure that you type all commands as directed.

Procedure

1. Create an image in the **avamar** project of the OpenStack cloud, using the OpenStack DPE API image file from the temporary location that is described in Unpacking the OpenStack DPE installation files on page 26, by typing the following command on one line at the command prompt:

   `openstack image create --disk-format qcow2 --file dpe-api.`*`version`*`.qcow2 --min-disk 16 --min-ram 6144 --public avamar-api`

   where *version* is the version of the OpenStack DPE.

2. Create a flavor for the **avamar** management service by logging in to Keystone as the backup_admin user and typing the following command on one line at the command prompt:

   `openstack flavor create --disk 16 --ram 6144 --vcpus 2 --swap 0 --ephemeral 0 --private avamar-api`

   Note the unique ID that OpenStack returns for the new flavor.

3. Assign the flavor to the project by typing the following command on one line at the command prompt:

   nova flavor-access-add *FLAVOR_ID PROJECT_ID*

   where:

   - *FLAVOR_ID* is the unique ID for the new flavor.
   - *PROJECT_ID* is the **avamar** project ID.

4. Use one of the following commands to create or import a keypair for SSH access:

   Note: You can also use the same keypair as the proxy image.

   - To import a keypair, type the following command on one line:

     openstack keypair create --public-key ~/.ssh/id_rsa.pub avamar_kp

   - To create a keypair, type the following command:

     openstack keypair create --public-key > keypair.pem avamar_kp

5. Create the security group and rules that define the access that the proxy instance permits to the network. The following uses the `avamar-api-security` group name:

```
SECURITY_GROUP_NAME=avamar-api-security
openstack security group create $SECURITY_GROUP_NAME
openstack security group rule create --proto icmp --src-ip \
  0.0.0.0/0 $SECURITY_GROUP_NAME
openstack security group rule create --proto tcp --src-ip \
  0.0.0.0/0 --dst-port 22:22 $SECURITY_GROUP_NAME
openstack security group rule create --proto tcp --src-ip \
  0.0.0.0/0 --dst-port 8443:8443 $SECURITY_GROUP_NAME
openstack security group rule create --proto tcp --src-ip \
  0.0.0.0/0 --dst-port 443:443 $SECURITY_GROUP_NAME
```

6. Use the following script to deploy an instance of the OpenStack DPE API inside the **avamar** project using the image that was created in step 1 on page 35 as the boot source. Do not create a volume from the image. The username and password must be for a keystone user in the admin project with the admin privilege. Replace all variables in the script with valid parameter values.

```
KEYSTONE="http://keystone.example.com:5000"
AVAMAR_REST_API=avamar-rest-api-host

IMAGE=avamar-api
FLAVOR=avamar-api
INSTANCE_NAME=avamar-api
CONFIG_FILE=/tmp/user-data
KEYPAIR=avamar_kp
SECURITY_GROUP_NAME=avamar-api-security

cat > $CONFIG_FILE <<XXXX
#cloud-config
bootcmd:
 - /tmp/postconfig.sh "$KEYSTONE" $AVAMAR_REST_API
XXXX

openstack server create \
  --image $IMAGE \
  --key-name $KEYPAIR \
  --security-group $SECURITY_GROUP_NAME \
  --flavor $FLAVOR \
  --user-data $CONFIG_FILE \
  $INSTANCE_NAME
```

   where *avamar-rest-api-host* is the IP address or hostname of the Avamar REST API instance.

7. Optional. If using SSL, complete the following substeps to configure SSL:
   a. SSH to the management instance.
   b. Change directory by typing the following command:

      cd /etc/avamar

   c. Generate a private key by typing the following command:

      openssl genrsa 1024 > host.key

   d. Change permissions on the key by typing the following command:

      chmod 400 host.key

   e. Create a server certificate for the OpenStack DPE API by typing the following command on one line:

      openssl req -new -x509 -nodes -sha1 -days 365 -key host.key > host.cert

   f. Create the `.pem` file from the certificate and key by typing the following command:

      cat host.cert host.key > /etc/avamar/host.pem

   g. Change permissions on the `.pem` file by typing the following command:

      chmod 400 /etc/avamar/host.pem

   h. Change ownership of the `.pem` file by typing the following command:

      chown avamar:avamar /etc/avamar/host.pem

   i. Type the following command on one line to validate the SSL public certificate for Keystone:

      openssl s_client -showcerts -connect host:port /null | openssl x509 -outform PEM > keystone.pem

   j. Using a Linux text editor, such as `vi`, edit `/etc/dpe/dpe-api.conf`.
      - In the `[keystone]` section, change the `url` setting to the URL of the Keystone instance.
      - If the SSL public certificate should be validated:
        - Uncomment the line for the **ssl_pem** setting and change the value to the `host.pem` file created in substep 7.f on page 37.
        - Change the instances of the `ssl_verify` setting to **True**.
      - If validation of the SSL public certificate is not required:
      - Comment the line for the **ssl_pem** setting by typing a **#** at the beginning of the line.
      - Change the instances of the `ssl_verify` setting to **False**.

```
[DEFAULT]
host=0.0.0.0
port=8080
# ssl_pem=
policy=/etc/dpe/dpe_api_policy.json
resourceShareCapacityMB=10000
default_cbt_enable=True

[provider]
url=https://localhost:8543/rest-api
username=admin
password=changeme
ssl_verify=False
```

```
[keystone]
url=http://keystone.example.com:5000
interface=public
ssl_verify=False

[flr]
url=http://localhost:8081/v1
username=test
password=test
interface=public
ssl_verify=False

[logging]
format=%(asctime)s - %(levelname)-8s %(name)-15s %(message)s
dateformat=%m/%d/%Y  %H:%M:%S
level=INFO
# filename=/var/log/dpe_api/dpe_api.log

[extra]
# the below options are auto-generated by DPE-API
# do NOT modify any of the below options unless you are
# aware of the sequence.
```

8. SSH to the OpenStack DPE API instance.
9. On the OpenStack DPE API instance, restart the management service:

   `systemctl restart dpe-api.service`

After you finish

NOTICE Do not change the value of `DHCLIENT_SET_HOSTNAME` in `/etc/sysconfig/network/dhcp` to anything other than `no`. Changing this value may cause the DHCP client to change the OpenStack DPE API hostname.

Creating the DPE-API service entity and endpoint on the controller node

The following task describes how to create the DPE-API service entity and endpoint on the OpenStack controller node. Dell EMC recommends that you create two different endpoints for the service, as different versions of the OpenStack Dashboard can refer to different endpoints.

Before you begin

In the root user home directory of the OpenStack controller node, there should be a `keystonerc_admin` or `openrc` file (or a similar file for various different distributions). This file contains the environment variables that are needed by the OpenStack CLI tools, and its contents should resemble the following:

```
# unset OS_SERVICE_TOKEN
# export OS_USERNAME=admin
# export OS_PASSWORD=yourpassword
# export OS_AUTH_URL=http://yourhost:5000/v3.0
# export PS1='[\u@\h \W(keystone_admin)]\$ '
# export OS_TENANT_NAME=admin
# export OS_REGION_NAME=RegionOne
```

These environment variables must be exported by using the shell's `source` command:

`source keystonerc_admin`

Procedure

1. Create the service entity by typing the following command on one line:

   `openstack service create --name emc-dpe --description "OpenStack Data Protection Extension" emc-dpe`

2. Create the endpoints by typing the following commands, each on one line:

   ```
   openstack endpoint create --region RegionOne emc-dpe public https://dpe-api-ip:8443/v1
   ```

   ```
   openstack endpoint create --region RegionOne emc-dpe internal https://dpe-api-ip:8443/v1
   ```

   where *dpe-api-ip* is the IP address or hostname of the OpenStack DPE API instance.

Installing the CBT driver and agent

Install the CBT driver and agent, plus any Linux package dependencies, on each compute node. Review the notes and select the procedures that apply to your Linux distribution.

Note: Kernel upgrades between supported versions do not require additional steps. The upgrade preserves CBT records and incremental backups can continue.

Installing the CBT driver and agent on RHEL

Install the CBT driver before installing the CBT agent.

Procedure

1. Install the CBT driver package from the temporary location that is described in Unpacking the OpenStack DPE installation files on page 26 by typing the following command:

   ```
   rpm -ivh dpe-cbt-driver-linux-el7-x86_64-version.rpm
   ```

   where *version* is the version of the CBT driver.

2. Install the required dependency by typing the following command:

   ```
   yum install python-suds
   ```

3. Install the CBT agent package from the temporary location that is described in Unpacking the OpenStack DPE installation files on page 26 by typing the following command:

   ```
   rpm -ivh dpe-cbt-agent-version.noarch.rpm
   ```

   where *version* is the version of the OpenStack DPE.

Installing the CBT driver and agent on Ubuntu

Install the CBT driver before installing the CBT agent.

Procedure

1. Install the CBT driver package from the temporary location that is described in Unpacking the OpenStack DPE installation files on page 26 by typing the following command:

   ```
   dpkg -i dpe-cbt-driver-linux-ub16-x86_64-version.deb
   ```

   where *version* is the version of the CBT driver.

2. Install the required dependency by typing the following command:

   ```
   apt-get install python-support
   ```

3. Install the CBT agent package from the temporary location that is described in Unpacking the OpenStack DPE installation files on page 26 by typing the following command:

   ```
   dpkg -i dpe-cbt-agent_version_all.deb
   ```

where *version* is the version of the OpenStack DPE.

Creating the DPE service entity and API endpoint on the controller node

The following task describes how to create the DPE service entity and API endpoint on the OpenStack controller node.

Before you begin

In the root user home directory of the OpenStack controller node, there should be a `keystonerc_admin` or `openrc` file (or a similar file for various different distributions). This file contains the environment variables that are needed by the OpenStack CLI tools, and its contents should resemble the following:

```
# unset OS_SERVICE_TOKEN
# export OS_USERNAME=admin
# export OS_PASSWORD=yourpassword
# export OS_AUTH_URL=http://yourhost:5000/v3.0
# export PS1='[\u@\h \W(keystone_admin)]\$ '
# export OS_TENANT_NAME=admin
# export OS_REGION_NAME=RegionOne
```

These environment variables must be exported by using the shell's `source` command:

`source keystonerc_admin`

Procedure

1. Create the service entity by typing the following command on one line:

   `openstack service create --name=avamar --description="Avamar DPE Backup Service" backup`

   Note the value that OpenStack returns for the `id` field.

2. Create the API endpoint by typing the following command on one line:

   `openstack endpoint create --region RegionOne` *`service-id`* `public http://` *`controller-node-ip`*`:1947`

   where:

   - *service-id* is the `id` value that OpenStack returned in the previous step.
   - *controller-node-ip* is the IP address of the OpenStack controller node.

Installing the OpenStack DPE Horizon plug-in

This section provides information about installing the OpenStack DPE Horizon plug-in in a containerized environment.

About this task

Throughout this task, *version* is the version of the OpenStack DPE.

Procedure

1. Copy the OpenStack DPE Horizon plug-in package (`dpe-horizon-plugin-`*`version`*`.tar.gz`) to a temporary directory on the OpenStack node that runs the Horizon container service.
2. Copy the package into the Horizon container by typing the following command:

   `docker cp dpe-horizon-plugin-`*`version`*`.tar.gz horizon:/root`

3. Start a Bash shell in the Horizon container by typing the following command:

   `docker exec -it horizon bash`

4. Unzip the package by typing the following command:

   `tar -zxvf dpe-horizon-plugin-`*`version`*`.tar.gz`

5. Change directory by typing the following command:

   `cd dpe-horizon-plugin`

6. Install the OpenStack DPE Horizon plug-in by typing the following command:

   `./install.sh`

   The script automatically installs the OpenStack DPE Horizon plug-in into the `/usr/share/openstack-dashboard/openstack_dashboard/dashboards/emc_dpe` directory.

   If the script returns an error that asks you to specify the OpenStack Horizon location, then the OpenStack Dashboard installation path is different for this container. Complete the following substeps:

   a. Verify and record the correct installation path for OpenStack Dashboard.

   b. Install the OpenStack DPE Horizon plug-in with the correct path by typing the following command:

      `./install.sh --horizon_dir <`*`dashboard-path`*`>`

      where *<dashboard-path>* is the correct path to the OpenStack Dashboard.

7. Exit the Bash shell for the Horizon container by typing the following command:

   `exit`

8. Restart the Horizon container by typing the following command:

   `docker restart horizon`

9. Optional. To configure the OpenStack DPE Horizon plug-in to support only trusted certificates, perform the following substeps:

   Note:
   This step secures communication between the OpenStack DPE Horizon plug-in and the OpenStack DPE API by ensuring that Horizon verifies the trusted certificate before accepting a response.

   Configuring SSL during the installation of the OpenStack DPE API with a trusted certificate is a prerequisite.

   a. Start a Bash shell in the Horizon container by typing the following command:

      `docker exec -it horizon bash`

   b. Verify and record the correct installation path for OpenStack Dashboard.

   c. Using a Linux text editor, such as `vi`, edit *<dashboard-path>*`/local/local_settings.d/_50_dpe_api_settings.py`.

      where *<dashboard-path>* is the full path to the OpenStack Dashboard. For example: `/usr/share/openstack-dashboard/openstack_dashboard`.

   d. Locate the following lines near the end of the file:

```
# By default do NOT verify SSL certificate when invoking
# HTTPS request to DPE-API
DPE_SSL_VERIFY = False
```

e. Change the value of `DPE_SSL_VERIFY` to **`True`**.

f. Save and close the file.

g. Exit the Bash shell for the Horizon container by typing the following command:

```
exit
```

h. Restart the Horizon container by typing the following command:

```
docker restart horizon
```

Enabling multi-domain support

After installing the OpenStack DPE, enable domain support for Horizon and Keystone v3.

About this task

Note: Space limitations may prevent some commands in this task from being printed on one line. Use the clipboard, where available, to copy and paste commands, and ensure that you type all commands as directed.

Procedure

1. Start a Bash shell in the Horizon container by typing the following command:

   ```
   docker exec -it horizon bash
   ```

2. Verify and record the correct installation path for OpenStack Dashboard.

   The default path for RedHat is `/usr/share/openstack-dashboard/openstack_dashboard`.

3. Use a Linux text editor, such as `vi`, to edit `<dashboard-path>/local/local_settings.py`.

4. Locate the following lines in the file:

   ```
   # Overrides for OpenStack API versions. Use this setting to force the
   # OpenStack dashboard to use a specific API version for a given service
   API.
   # Versions specified here should be integers or floats, not strings.
   # NOTE: The version should be formatted as it appears in the URL for the
   # service API. For example, The identity service APIs have inconsistent
   # use of the decimal point, so valid options would be 2.0 or 3.
   # Minimum compute version to get the instance locked status is 2.9.
   #OPENSTACK_API_VERSIONS = {
   #    "data-processing": 1.1,
   #    "identity": 3,
   #    "image": 2,
   #    "volume": 2,
   #    "compute": 2,
   #}

   # Set this to True if running on a multi-domain model. When this is
   enabled, it
   # will require the user to enter the Domain name in addition to the
   username
   # for login.
   #OPENSTACK_KEYSTONE_MULTIDOMAIN_SUPPORT = False
   ```

   and

   ```
   OPENSTACK_HOST = "127.0.0.1"
   OPENSTACK_KEYSTONE_URL = "http://%s:5000/v2.0" % OPENSTACK_HOST
   OPENSTACK_KEYSTONE_DEFAULT_ROLE = "_member_"
   ```

5. Uncomment the appropriate lines for *OPENSTACK_API_VERSIONS* and change the value of *identity* to `3`.

   For example:

   ```
   OPENSTACK_API_VERSIONS = {
       "identity": 3,
   }
   ```

6. Uncomment the line for *OPENSTACK_KEYSTONE_MULTIDOMAIN_SUPPORT* and change the value to **`True`**.
7. Change the value of *OPENSTACK_KEYSTONE_URL* to **`"http://keystone-host:5000/v3"`**

   where *keystone-host* is the IP address or hostname of the Keystone node.
8. Save and close the file.
9. Exit the Bash shell for the Horizon container by typing the following command:

   `exit`
10. Restart the Horizon container by typing the following command:

    `docker restart horizon`

Results

The OpenStack login screen now contains an additional **Domain** field.

Post-installation checklist

Use the following checklists to ensure that the systems are working correctly after installation.

Verifying that the services are running

Procedure

1. Verify that the OpenStack DPE service is running:

   `systemctl status dpe-proxy-service`

   This command should return a status of `active (running)` or `dpe-proxy-service start/running`.
2. Verify that the OpenStack DPE API service is running.

   SSH to the OpenStack DPE API instance and type the following at the command prompt:

   `systemctl status dpe-api`

   This command should return a status of `active (running)` or `dpe-api start/running`.
3. Verify that the OpenStack DPE proxy is running.

   SSH to the OpenStack DPE proxy instances and type the following at the command prompt:

   `systemctl status avagent`

   This command should return a status of `active (running)` or `avagent start/running`.
4. On each compute node, verify that the CBT driver service is running:

   `systemctl status lfi`

These commands should return a status of `active (running)` or `lfi start/running`.

5. On each compute node, verify that the CBT agent service is running:

   `systemctl status dpe-cbt-agent`

   These commands should return a status of `active (running)` or `dpe-cbt-agent start/running`.

6. Verify that the OpenStack DPE FLR service is running.

   SSH to the OpenStack DPE API instance and type the following at the command prompt:

   `systemctl status flrservice`

   This command should return a status of `active (running)` or `flrservice start/running`.

   In a web browser, type **`https://`*`dpe-api-host`*`/flrui`**. Verify that the OpenStack DPE FLR login page displays.

7. Verify that the Avamar REST API service is running.

   SSH to the Avamar REST API instance and type the following at the command prompt:

   `systemctl status concerto`

   This command should return a status of `active (running)` or `concerto start/running`.

Verifying component communication

Procedure

1. Verify that the Avamar server can ping the FQDN of all deployed `dpe-avamar-proxy` instances.
2. Verify that the OpenStack DPE proxy instances can ping the Avamar server.
3. Verify that the OpenStack DPE proxy instances can ping the controller node (the OpenStack DPE service).
4. Verify that the OpenStack DPE API instance can ping Keystone.
5. Verify that the OpenStack DPE API instance can ping the Avamar server.
6. Verify that the Avamar REST API instance can ping the Avamar server.
7. Verify that the OpenStack DPE service node can ping the OpenStack DPE proxy instances.
8. Verify that the OpenStack DPE Horizon plug-in appears in the Horizon dashboard.

Verifying the communication ports

Procedure

1. SSH to the OpenStack DPE proxy instance.
2. Check the communication port of the OpenStack DPE proxy instance by typing the following command:

   `curl http://`*`dpe-proxy-service-IP`*`:1947`

   This command should not return a `couldn't connect to host` error.

3. From the controller node, check the communication port of the OpenStack DPE API instance by typing the following command:

   `curl https://`*`dpe-api-service-IP`*`:8443`

This command should not return a `couldn't connect to host` error.

CHAPTER 3

Using the OpenStack DPE Horizon plug-in

This chapter includes the following topics:

- Overview of the OpenStack DPE Horizon plug-in 48
- Operations for cloud administrators 49
- Operations for project administrators 62
- File-level restoration 68

Overview of the OpenStack DPE Horizon plug-in

The OpenStack DPE Horizon plug-in is an addition to the OpenStack Horizon web-based management application. The plug-in allows users to administer the OpenStack DPE from a computer with a supported browser without having to issue commands directly via the REST API.

Use the OpenStack DPE Horizon plug-in to monitor backup and restore activities, as well as to configure backup policies, manage instances, and configure other settings. Any command that can be issued via the REST API can be performed as an operation in the OpenStack DPE Horizon plug-in.

Access the OpenStack DPE Horizon plug-in by logging in to OpenStack Horizon as any user with appropriate privileges. If domain support was configured during installation, log in to an appropriate OpenStack domain as well. Some operations may be restricted to users with particular privileges.

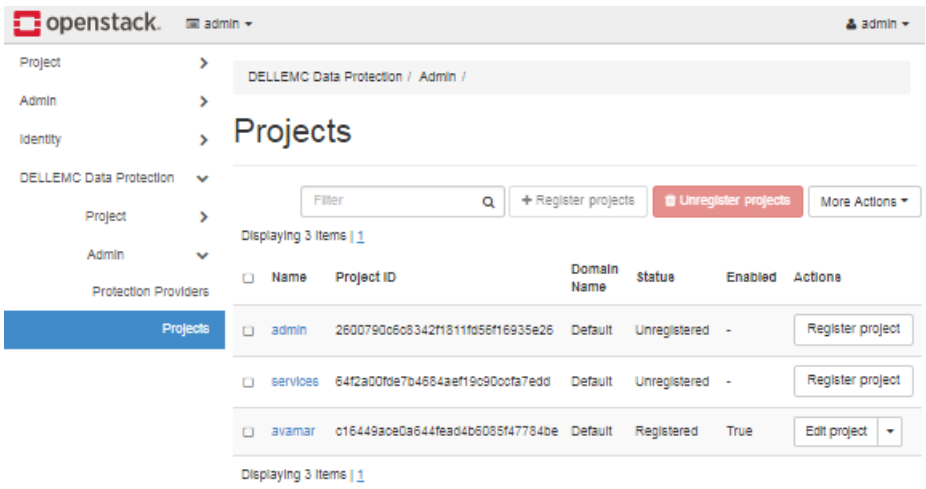
Note: Some steps in this chapter may assume that you have completed the optional installation tasks in the previous chapters.

OpenStack DPE Horizon plug-in interface

The OpenStack DPE Horizon plug-in appears as a new **DellEMC Data Protection** dashboard in OpenStack Horizon. The dashboard provides access to all functionality through panel groups and panels.

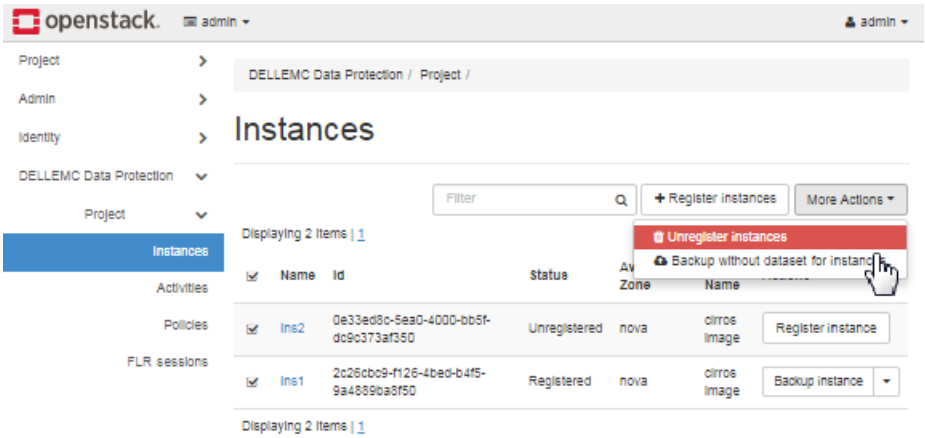
The **Admin** panel group is only visible to cloud administrators.

Figure 1 The OpenStack DPE Horizon plug-in



Many operations can be performed simultaneously on multiple list items by using the checkboxes for each row and the action buttons above the list.

Figure 2 Batch operations



If a panel displays a list with many results, the OpenStack DPE separates the results into discrete pages for ease of navigation.

Additionally, you can search or filter the results on most panels by keywords or other criteria. For example, filtering the **Projects** panel using the criteria `True` returns only the projects that are enabled.

Operations for cloud administrators

The OpenStack DPE **Admin** panel group provides functionality for cloud administrators to create and manage protection providers where backups are stored, register projects to which protection applies, and create and manage the policies that control backup operations.

Operations on the **Admin** panel group can only be performed by cloud administrators. Project administrators implement the policies that are defined on the **Admin** panel group.

Protection providers

A protection provider is a reference to a backup appliance, which is either a physical Avamar server or an Avamar Virtual Edition (AVE) instance, and which may also have an attached Data Domain system.

A protection provider is registered by URL and exists outside of projects. One protection provider may be shared among many projects or may be assigned exclusively to a single project, and a protection provider cannot be removed unless all projects registered to the protection provider have been removed first. A protection provider can be suspended (for example, for maintenance).

The OpenStack DPE supports one protection provider at a time.

Adding a protection provider

Before you begin

Ensure that communication is possible on port 9443 between OpenStack and the Avamar server. Port 9443 is the Management Console web services interface.

Procedure

1. From the OpenStack DPE dashboard, select **Admin** > **Protection Providers**.

   The **Protection providers** panel appears.

2. Click **Add provider**.

   The **Add provider** dialog box appears.

3. Type a name and a description for the protection provider.
4. Type the URL that provides the path to the Avamar server.

   Supply the **Avamar RMI URL** in the form `https://host:9443`, where *host* is the hostname or IP address of the Avamar server.
5. Type the name of a user with administrator privileges for the Avamar software.

   For example, the Avamar `MCUser` account.

   Note: This user must be configured for the Avamar software, not the Avamar appliance operating system.
6. Type the password for the administrator user.
7. Click **Add provider**.

Results

The new protection provider appears on the **Protection providers** panel.

Editing a protection provider

Editing an existing protection provider allows you to change a username or password, or the associated Avamar server.

Procedure

1. From the OpenStack DPE dashboard, select **Admin** > **Protection Providers**.

   The **Protection providers** panel appears.
2. Locate the protection provider in the list and then click **Edit provider**.
3. Edit the name or description for the protection provider, the Avamar RMI URL, the username, or the password.
4. Click **Save protection provider**.

Results

The **Protection providers** panel displays the changed information.

Suspending and resuming a protection provider

You can suspend a protection provider to stop backups on that appliance. One example would be as part of routine server maintenance.

Procedure

1. From the OpenStack DPE dashboard, select **Admin** > **Protection Providers**.

   The **Protection providers** panel appears.
2. Locate the protection provider in the list and then click **Suspend provider** from the **Actions** drop-down.

   On the **Protection providers** panel, the **State** for this protection provider changes to `Suspended`.
3. When you are ready to resume protection, click **Resume provider** from the **Actions** drop-down.

   On the **Protection providers** panel, the **State** for this protection provider changes to `Active`.

Deleting a protection provider

The OpenStack DPE supports one protection provider at a time. You can remove the current protection provider and add another.

Before you begin

Perform the following actions before deleting a protection provider:

- Delete all associated projects and their group policies, schedules, retention policies, and datasets.
- Unregister all associated instances.

Note: Unregistering an instance deletes the instance backups from the protection provider.

Procedure

1. From the OpenStack DPE dashboard, select **Admin** > **Protection Providers**.

   The **Protection providers** panel appears.

2. Locate the protection provider in the list and then click **Delete provider** from the **Actions** drop-down.

   The current protection provider disappears from the **Protection providers** panel.

Projects

A project represents an OpenStack tenant or project. When a project is registered, at least one protection provider must be associated with it. More than one protection provider may be associated with one project, but the set of associated protection provider cannot be modified after registration. The list of providers may either be shared with other projects, or be exclusive to a single project.

A project contains instances, group policies, retention policies, schedules and datasets. A project can be suspended similarly to a protection provider. A project can only be removed only after all of its contents have been removed.

If a tenant or project is deleted from OpenStack, it remains registered in the OpenStack DPE API and may be used to restore the contents of the project.

Note: The OpenStack DPE retrieves all of the projects on the **Projects** panel from the list on the OpenStack **Identity** panel group.

Registering a project

Registering a project allows the OpenStack DPE to manage the protection of instances within the project.

Procedure

1. From the OpenStack DPE dashboard, select **Admin** > **Projects**.

   The **Projects** panel appears.

2. Locate an unregistered project in the list and then click **Register project**.

Results

The OpenStack DPE signals the success of the registration operation. The status of the project changes to `Registered`.

Editing a project

Editing a project allows you to rename the project, change the assigned quota, or enable and disable the project.

Procedure

1. From the OpenStack DPE dashboard, select **Admin** > **Projects**.

   The **Projects** panel appears.

2. Locate the project in the list and then click **Edit project**.

   The **Edit project** dialog box opens.

3. Edit the name for this project.
4. Edit the value in the **Capacity** field.

   This value controls the quota assigned to the project.

5. Check or uncheck **Enabled**.

   This checkbox controls whether protection is applied to the project.

6. Click **Save project**.

Results

Click the name of the modified project to review the details.

Unregistering a project

Unregistering a project makes the project no longer eligible to receive protection from the OpenStack DPE. The project can only be unregistered after you remove its contents.

Procedure

1. From the OpenStack DPE dashboard, select **Admin** > **Projects**.

   The **Projects** panel appears.

2. Locate a registered project in the list and then click the name of the project to review the project details.

   Ensure that there are no remaining group policies, schedules, retention policies, or datasets.

3. Select **Admin** > **Projects** to return to the **Projects** panel.
4. Locate the project in the list again and then click **Unregister project** from the **Actions** drop-down.

   A confirmation dialog box appears. Click **Unregister project**.

Results

The OpenStack DPE signals the success of the deregistration operation. The status of the project changes to `Unregistered`.

Suspending and resuming a project

You can suspend a project to stop backups on the instances in that project. One example would be as part of routine server maintenance.

Procedure

1. From the OpenStack DPE dashboard, select **Admin** > **Projects**.

   The **Projects** panel appears.

2. Locate the project in the list and then click **Suspend project** from the **Actions** drop-down.

   On the **Projects** panel, the **Enabled** attribute for this project changes to `False`.

3. When you are ready to resume the project, click **Resume project** from the **Actions** drop-down.

   On the **Projects** panel, the **Enabled** attribute for this project changes to `True`.

Group policies

A group policy contains all the information necessary to perform backups of groups of instances.

Group policy objects contain three child objects:

- Schedule - when a scheduled backup is performed.
- Retention policy - how long the backup is stored in the backup appliance
- Dataset - optional backup parameters

Schedules

Note: The interface for the protection provider only supports time offsets based on GMT, not UTC. Ensure that you convert all time offsets to GMT when creating or editing a schedule, and allow for daylight saving time where applicable.

Creating a schedule

Schedules control when automated backups run on registered instances.

Procedure

1. From the OpenStack DPE dashboard, select **Admin** > **Projects**.

   The **Projects** panel appears.

2. Locate a project against which to define the schedule and then click on the project name.

   The project details page opens.

3. Click the **Schedules** tab.
4. Click **Create schedule**.

   The **Create schedule** dialog box opens.

5. Type a name and description for this schedule.
6. Type a date on which the schedule starts, and a date on which the schedule expires.

   Type dates in the format *`YYYY-MM-DD`*.

7. Select the time zone under which the schedule operates.

   Type time zones in the format `GMT-`*`OFFSET`* or `GMT+`*`OFFSET`*.

   For example, `GMT-8:00`, `GMT-5:00`, `GMT+5:30` or `GMT+8:00`.

8. Use the **Schedule type** drop-down to select how often the schedule runs.
9. Type the local time at which the schedule runs.

   Type the time in the format *`HH`*`:`*`MM`*.

10. Type the maximum amount of time for which the schedule is allowed to run, in hours.

    Scheduled runs that exceed this length are terminated.

11. Click **Create schedule**.

Results

The new schedule appears on the **Schedules** tab. Click the name of the new schedule to review the details.

Editing a schedule

Editing an existing schedule allows you to change details without re-creating the entire schedule. For example, changing the time zone for daylight saving time.

Procedure

1. From the OpenStack DPE dashboard, select **Admin** > **Projects**.

   The **Projects** panel appears.
2. Locate the project that contains the schedule and then click on the project name.

   The project details page opens.
3. Click the **Schedules** tab.
4. Locate the schedule in the list and then click **Edit schedule**.

   The **Edit schedule** dialog box opens.
5. Edit the name and description for this schedule.
6. Edit the date on which the schedule starts, or the date on which the schedule expires.

   Type dates in the format `YYYY-MM-DD`.
7. Edit the time zone in which the schedule operates.

   Type time zones in the format `GMT-OFFSET` or `GMT+OFFSET`.

   For example, `GMT-8:00`, `GMT-5:00`, `GMT+5:30` or `GMT+8:00`.
8. Use the **Schedule type** drop-down to edit how often the schedule runs.
9. Edit the local time at which the schedule runs.

   Type the time in the format `HH:MM`.
10. Edit the maximum amount of time for which the schedule is allowed to run, in hours.

    Scheduled runs that exceed this length are terminated.
11. Click **Save schedule**.

Results

Click the name of the modified schedule to review the details.

Deleting a schedule

Delete a schedule as part of deleting a group policy, or when you want to create an entirely new schedule.

Procedure

1. From the OpenStack DPE dashboard, select **Admin** > **Projects**.

   The **Projects** panel appears.
2. Locate the project that contains the schedule and then click the project name.

   The project details page opens.
3. Click the **Schedules** tab.
4. Locate the schedule in the list and then click **Delete schedule** from the **Actions** drop-down.

The schedule disappears from the **Schedules** tab.

5. To delete multiple schedules, select the checkbox next to the name of each schedule to delete and then click **Delete schedules**.

   A confirmation dialog box appears.

Results

The selected schedule disappears from the **Schedules** tab.

Retention policies

Creating a retention policy

Retention policies govern how long the data is kept after the scheduled backup.

Procedure

1. From the OpenStack DPE dashboard, select **Admin** > **Projects**.

   The **Projects** panel appears.

2. Locate a project against which to define the retention policy and then click the project name.

   The project details page opens.

3. Click the **Retention policies** tab.
4. Click **Create retention policy**.

   The **Create retention policy** dialog box opens.

5. Type a name and description for this retention policy.
6. Use the **Mode** drop-down to select `Backup`.
7. Use the **Retention type** drop-down to control how you specify the retention time.

   The drop-down has four options:

Option	Description
Retention period	Retains the backup data for a specified length of time after backup completion.
End date	Retains the backup data until a specified date.
No end date	Retains the backup data indefinitely.
Advanced retention	Retains the backup data for a custom period that is based on the backup schedule.

   Note: The selection of retention type controls the available fields in the **Create retention policy** dialog box.

8. Use the **Period**, **Unit**, and **End date** fields to select how long the OpenStack DPE should keep the backup data.
9. For advanced retention, type the number of periods to retain each type of backup in the **Keep daily duration**, **Keep weekly duration**, **Keep monthly duration**, and **Keep yearly duration** drop-downs.
10. Click **Create retention policy**.

Results

The new retention policy appears on the **Retention policies** tab. Click the name of the new retention policy to review the details.

Editing a retention policy

Editing an existing retention policy allows you to change how long the backup data resides on the protection provider, without manually deleting backups.

Procedure

1. From the OpenStack DPE dashboard, select **Admin** > **Projects**.

   The **Projects** panel appears.

2. Locate the project that contains the retention policy and then click the project name.

   The project details page opens.

3. Click the **Retention policies** tab.
4. Locate the retention policy in the list and then click **Edit retention policy**.

   The **Edit retention policy** dialog box opens.

5. Edit the name and description for this retention policy.
6. Use the **Retention type** drop-down to change the nature of the retention time, or specify a different length of time to retain backup data.

   The drop-down has four options:

Option	Description
Retention period	Retains the backup data for a specified length of time after backup completion.
End date	Retains the backup data until a specified date.
No end date	Retains the backup data indefinitely.
Advanced retention	Retains the backup data for a custom period that is based on the backup schedule.

   Note: The selection of retention type controls the available fields in the **Edit retention policy** dialog box.

7. Use the **Period**, **Unit**, and **End date** fields to change how long the OpenStack DPE should keep the backup data.
8. For advanced retention, change the number of periods to retain each type of backup in the **Keep daily duration**, **Keep weekly duration**, **Keep monthly duration**, and **Keep yearly duration** drop-downs.
9. Click **Save retention policy**.

Results

Click the name of the modified retention policy to review the details.

Deleting a retention policy

Delete a retention policy eriod as part of deleting a group policy, or when you want to create an entirely new retention policy.

Procedure

1. From the OpenStack DPE dashboard, select **Admin** > **Projects**.

   The **Projects** panel appears.

2. Locate the project that contains the retention policy and then click the project name.

   The project details page opens.

3. Click the **Retention policies** tab.
4. Locate the retention policy in the list and then click **Delete retention policy** from the **Actions** drop-down.

   The retention policy disappears from the **Retention policies** tab.

5. To delete multiple retention policies, select the checkbox next to the name of each retention policy to delete and then click **Delete retention policies**.

   A confirmation dialog box appears.

Results

The selected retention policy disappears from the **Retention policies** tab.

Datasets

Creating a dataset

Datasets control what data is protected when a scheduled or on-demand backup triggers.

Procedure

1. From the OpenStack DPE dashboard, select **Admin** > **Projects**.

   The **Projects** panel appears.

2. Locate a project against which to define the dataset and then click on the project name.

   The project details page opens.

3. Click the **Datasets** tab.
4. Click **Create dataset**.

   The **Create dataset** dialog box opens.

5. Type a name and description for this schedule.
6. Optional. Check **Use CBT** to perform incremental backups.

   Note: Incremental backups require the OpenStack DPE to track the changed block information for each instance. To track this information, ensure that you enable the **Enable CBT** attribute when registering any instance for which you require incremental backups.

7. Optional. Check **Use Data Domain** to store backups on a Data Domain system that is attached to the protection provider.

   Leaving **Use Data Domain** unchecked configures the OpenStack DPE to store backups on the protection provider appliance.

When **Use Data Domain** is checked, the **Data Domain index** field appears.

a. Type the index number of the Data Domain system in the **Data Domain index** field.

This value specifies which attached Data Domain system is the backup target, if more than one Data Domain system is attached to the protection provider.

Obtain this value from the protection provider.

8. Click **Create dataset**.

Results

The new dataset appears on the **Datasets** tab. Click the name of the new dataset to review the details.

Editing a dataset

Editing a dataset allows you to change the specific data that is protected on each instance.

Procedure

1. From the OpenStack DPE dashboard, select **Admin** > **Projects**.

   The **Projects** panel appears.

2. Locate the project that contains the dataset and then click the project name.

   The project details page opens.

3. Click the **Datasets** tab.
4. Locate the dataset in the list and then click **Edit dataset**.

   The **Edit dataset** dialog box opens.

5. Edit the name and description for this dataset.
6. Optional. Change the **Use CBT** setting to control incremental backups.

   Note: Incremental backups require the OpenStack DPE to track the changed block information for each instance. To track this information, ensure that you enable the **Enable CBT** attribute when registering any instance for which you require incremental backups.

7. Optional. Change the **Use Data Domain** setting to store backups on a Data Domain system that is attached to the protection provider.

   Leaving **Use Data Domain** unchecked configures the OpenStack DPE to store backups on the protection provider appliance.

   When **Use Data Domain** is checked, the **Data Domain index** field appears.

   a. Change the index number of the Data Domain system in the **Data Domain index** field.

   This value specifies which attached Data Domain system is the backup target, if more than one Data Domain system is attached to the protection provider.

   Obtain this value from the protection provider.

8. Click **Save dataset**.

Results

Click the name of the modified dataset to review the details.

Deleting a dataset

Delete a dataset as part of deleting a group policy, or when you no longer want to protect particular data.

Procedure

1. From the OpenStack DPE dashboard, select **Admin** > **Projects**.

   The **Projects** panel appears.

2. Locate the project that contains the dataset and then click the project name.

   The project details page opens.

3. Click the **Datasets** tab.
4. Locate the dataset in the list and then click **Delete dataset** from the **Actions** drop-down.

   The dataset disappears from the **Datasets** tab.

5. To delete multiple datasets, select the checkbox next to the name of each dataset to delete and then click **Delete datasets**.

   A confirmation dialog box appears.

Results

The selected dataset disappears from the **Datasets** tab.

Group policies

Working with group policies

As a cloud administrator, you define the group policy objects that project administrators use to back up instances.

Before you can create a group policy, define at least one schedule, retention policy, and dataset. If you have existing objects of these types, you can combine them into a new group policy.

To view the existing group policies, schedules, retention policies, and datasets, click the name of an existing project on the **Admin** > **Projects** panel to open the project details page. Select the tab that corresponds to the applicable object type.

Group policies, schedules, retention policies, and datasets can only be created, edited, or deleted, by the cloud administrator.

Creating a group policy

A group policy controls the application of predefined rules for backing up data from specified instances. Create group policies to simplify the backup process for groups of instances.

Before you begin

The project must have at least one associated schedule, retention policy, and dataset.

Procedure

1. From the OpenStack DPE dashboard, select **Admin** > **Projects**.

   The **Projects** panel appears.

2. Locate a project against which to define the group policy and then click the project name.

   The project details page opens.

3. Click the **Group policies** tab.

4. Click **Create group policy**.

   The **Create group policy** dialog box opens.

5. Type a name and description for this group policy.
6. Optional. Check the **Enabled** box to make this group policy active.

   Leaving the **Enabled** box unchecked prevents scheduled backups from running according to the terms that are defined in this group policy.

7. Select a value from the **Override schedule** drop-down.

Option	Description
Not overridden	The group policy does not override the assigned schedule.
Skip next backup	The group policy skips the next scheduled backup before following the schedule.
Run next backup once	The group policy performs the next scheduled backup one time only.

   The *Avamar Administration Guide* provides additional details.

8. Select an encryption type from the **Encryption type** drop-down.

   The encryption levels in the drop-down correspond to the encryption levels that are defined by the *Avamar Product Security Guide*.

9. Select a schedule from the **Schedule** drop-down.
10. Select a retention policy from the **Retention policy** drop-down.
11. Select a dataset from the **Dataset** drop-down.

   Note:
   OpenStack DPE populates the schedule, retention policy, and dataset drop-downs from their respective tabs for this project. If the object of your choice is not listed in the corresponding drop-down, click the **+** symbol next to that drop-down to create a new schedule, retention policy, or dataset for this project.

   Creating a schedule, retention policy, or dataset object returns you to the group policy creation dialog box.

12. Click **Create group policy**.

Results

The new group policy appears on the **Group policies** tab. Click the name of the new group policy to review the details.

Editing a group policy

Editing a group policy allows you to change the schedule, retention policy, or dataset without re-creating the entire group policy.

Procedure

1. From the OpenStack DPE dashboard, select **Admin** > **Projects**.

   The **Projects** panel appears.

2. Locate the project that contains the group policy and then click on the project name.

   The project details page opens.

3. Click the **Group policies** tab.
4. Locate the group policy in the list and then click **Edit group policy**.

   The **Edit group policy** dialog box opens.
5. Edit the name and description for the group policy.
6. Check the **Enabled** box to make the group policy active or uncheck the box to make the group policy inactive.
7. Select another value from the **Override schedule** drop-down.

Option	Description
Not overridden	The group policy does not override the assigned schedule.
Skip next backup	The group policy skips the next scheduled backup before following the schedule.
Run next backup once	The group policy performs the next scheduled backup one time only.

   The *Avamar Administration Guide* provides additional details.
8. Use the **Encryption type** drop-down to edit the encryption settings.

   The encryption levels in the drop-down correspond to the encryption levels that are defined by the *Avamar Product Security Guide*.
9. Select another schedule from the **Schedule** drop-down.
10. Select another retention policy from the **Retention policy** drop-down.
11. Select another dataset from the **Dataset** drop-down.

    Note:
    OpenStack DPE populates the schedule, retention policy, and dataset drop-downs from their respective tabs for this project. If the object of your choice is not listed in the corresponding drop-down, click the **+** symbol next to that drop-down to create a new schedule, retention policy, or dataset for this project.

    Creating a schedule, retention policy, or dataset object returns you to the group policy editing dialog box.
12. Click **Save group policy**.

Results

Click the name of the modified group policy to review the details.

Deleting a group policy

Deleting a group policy prevents that group policy from being applied to instances or projects.

Procedure

1. From the OpenStack DPE dashboard, select **Admin** > **Projects**.

   The **Projects** panel appears.
2. Locate the project that contains the group policy and then click the project name.

   The project details page opens.
3. Click the **Group policies** tab.
4. Locate the group policy in the list and then click **Delete group policy** from the **Actions** drop-down.

The group policy disappears from the **Group policies** tab.

5. To delete multiple group policies, select the checkbox next to the name of each group policy to delete and then click **Delete group policies**.

   A confirmation dialog box appears.

Results

The selected group policy disappears from the **Group policies** tab.

Operations for project administrators

The OpenStack DPE **Project** panel group provides functionality for project administrators to manage backup and restore operations on the projects and instances under their control. Project administrators can also apply policies that were created by the cloud administrator.

Operations on the **Project** panel group can only be performed by project administrators.

Note: The OpenStack DPE retrieves all of the instances on the **Instances** panel from the list on the OpenStack **Project** panel group.

Instances

OpenStack instances are registered under their respective projects. If the OpenStack instance is deleted, the OpenStack DPE API instance remains and can be used to restore the original instance. Instances may be assigned to policies for scheduled backups or they may be used for adhoc backups only.

Registering an instance

Registering an instance makes it available for backups via the OpenStack DPE and makes the instance known to the protection provider.

Before you begin

If you use the checkboxes and the **Register instances** button to perform a batch registration, OpenStack DPE registers the instances to the zone ID that matches the availability zone for each instance. OpenStack DPE also sets the **Enable CBT** attribute to the default value that is defined in `/etc/dpe/dpe-api.conf` on the OpenStack DPE API instance.

Procedure

1. From the OpenStack DPE dashboard, select **Project** > **Instances**.

   The **Instances** panel appears.

2. Locate an unregistered instance in the list and then click **Register instance**.

   OpenStack DPE displays the instance IDs, as OpenStack instances may have duplicate names.

   The **Register instance** dialog box appears.

3. Verify the instance name and zone ID.
4. Optional. Check the **Enable CBT** box to instruct the OpenStack DPE to start tracking changed block information for this instance.

   CBT is a method to identify and track only blocks that have changed since the latest backup of a virtual machine. Using CBT improves backup and proxy performance, while reducing the amount of data in subsequent backups after the initial full backup.

> **Note:** To use this CBT information for incremental backups, create a dataset with the **Use CBT** attribute enabled.

5. Type a contact name, phone number, and email address for this instance.
6. Type a location for the instance. The location can correspond to anything.
7. Click **Register instance**.

Results

The instance status changes to `Registered`. Click the name of the instance to review the details.

The **Register instance** button for this instance changes to a drop-down with supported operations.

Editing an instance

Editing an instance allows you to change the contact information, or enable and disable CBT for that instance.

Procedure

1. From the OpenStack DPE dashboard, select **Project** > **Instances**.

   The **Instances** panel appears.
2. Locate the instance in the list and then click **Edit instance** from the **Actions** drop-down.
3. Edit the contact name, phone number, email address, or location for this instance.

   Other fields, such as zone ID, cannot be changed.
4. Optional. Check or uncheck the **Enable CBT** box to instruct the OpenStack DPE to start or stop tracking changed block information for this instance.

   If you disable CBT, future backups do not make use of the incremental feature. If you enable CBT, the next backup is always a full backup. Subsequent backups are incremental backups.
5. Click **Save instance**.

Results

The instance details page displays the changed information. Click the name of the instance to review the details.

Protecting an instance

Protecting an instance assigns the instance to a particular group policy. You can assign more than one group policy to an instance.

Procedure

1. From the OpenStack DPE dashboard, select **Project** > **Instances**.

   The **Instances** panel appears.
2. Locate the instance in the list and then click **Protect instance** from the **Actions** drop-down.

   The **Protect instance** dialog box appears.
3. Assign a group policy to this instance from the **Policy** drop-down.
4. Click **Protect instance**.

Results

The instance receives protection based upon the schedule and retention policy of the selected group policy.

Viewing a list of backups for an instance

The **Backups** tab provides an overview of successful backups for a particular instance that are stored on the protection provider.

Procedure

1. From the OpenStack DPE dashboard, select **Project** > **Instances**.

   The **Instances** panel appears.

2. Locate the instance in the list and then click the name of the instance.

   The instance details page appears.

3. Click the **Backups** tab.

Results

OpenStack DPE displays the existing backups in chronological order. Confirm the retention times and expiration dates.

Performing an on-demand backup of an instance

OpenStack DPE can perform an instance backup that is independent of existing schedules and policies.

Procedure

1. From the OpenStack DPE dashboard, select **Project** > **Instances**.

   The **Instances** panel appears.

2. Locate the instance in the list and ensure that the instance status is `Registered`.
3. Click **Backup instance** from the **Actions** drop-down.

   The **Backup instance** dialog box opens.

4. (Optional) Check **Backup with dataset** to specify a particular dataset for this on-demand backup.

   When you check **Backup with dataset**, the **Select dataset** drop-down appears.

   a. Select the name of a defined dataset from the **Select dataset** drop-down.

      Note: OpenStack DPE populates the drop-down from the datasets that are defined for this project.

   Leaving **Backup with dataset** unchecked configures OpenStack DPE to perform the backup with the default dataset. The default dataset has **Use CBT** checked and **Use Data Domain** unchecked.

5. Click **Backup instance**.

Results

OpenStack DPE signals the successful start of the backup operation. Review the progress under the **Activities** panel.

If you enabled CBT for this instance, OpenStack DPE performs an incremental backup.

Restoring an instance

Any successful instance backup can be used to restore a copy of that instance.

Before you begin

The restore operation offers the option of re-creating the original instance or creating a copy of the instance.

- Re-creating the original instance is equivalent to deleting the original instance through the OpenStack **Project** dashboard and then restoring to a new copy.
- Creating a copy of the instance leaves the original instance intact, but may result in duplicate naming. Check the OpenStack instances list (**Project** > **Compute** > **Instances**) for the name and ID of the new instance. Use the **Time since created** and **Key Pair** fields to distinguish between the original and restored instances.

Procedure

1. From the OpenStack DPE dashboard, select **Project** > **Instances**.

   The **Instances** panel appears.

2. Locate the instance in the list and then click the name of the instance.

   The instance details page appears.

3. Click the **Backups** tab.

   A list of completed backups for this instance appears. Any backup in this list can be used to restore the instance.

4. Select an appropriate backup in the list and then click **Restore instance**.

   The **Restore instance** dialog box appears.

5. Select a value from the **Restore type** drop-down.

Option	Description
Original	The OpenStack DPE deletes the original instance after the restore operation completes.
New	The OpenStack DPE preserves the original instance. You can delete the original instance at your convenience.

6. Click **Restore instance**.

   The restore operation is queued to start.

Results

OpenStack DPE signals the successful start of the restore operation. Review the progress under the **Activities** panel.

Deleting a backup

OpenStack DPE can delete existing instance backups. For example, backups can be deleted to sanitize systems after a security breach.

Procedure

1. From the OpenStack DPE dashboard, select **Project** > **Instances**.

   The **Instances** panel appears.

2. Locate the instance in the list and then click the name of the instance.

The instance details page appears.

3. Click the **Backups** tab.

   A list of completed backups for this instance appears.

4. Select the backup in the list and then click **Delete backup**.

   A confirmation dialog box appears.

Results

OpenStack DPE deletes the backup and removes it from the list on the **Backups** tab.

Activities

The **Activities** panel lists the running jobs and tasks. Use this panel to monitor job and task completion.

Viewing past activities

The **Jobs** tab provides information about running and completed operations.

Procedure

1. From the OpenStack DPE dashboard, select **Project** > **Activities**.

   The **Activities** panel appears.

2. Select the **Jobs** tab.

   A list of completed and running jobs appears.

3. To view the most recent jobs, click the **Start Time** column header to sort the jobs.

   To view only a subset of jobs, use the **Filter** field to narrow the list of results.

4. To obtain more information about any job, click the job name to open the details page.

Viewing current, pending, and canceled tasks

The **Tasks** tab provides information about operations that have yet to complete.

Procedure

1. From the OpenStack DPE dashboard, select **Project** > **Activities**.

   The **Activities** panel appears.

2. Select the **Tasks** tab.

   A list of running, scheduled and canceled tasks appears.

   Tasks that are listed here are displayed by ID and not the name assigned to the job. To see a list of running tasks with the associated names, click the **Jobs** tab.

3. To view the most recent running tasks, click the **Start Time** column header to sort the tasks.

   To view scheduled tasks, click the **State** column header to sort the tasks.

   To view only a subset of tasks, use the **Filter** field to narrow the list of results.

Canceling a current task

The **Tasks** tab provides a method to stop the execution of a current or pending task.

Procedure

1. From the OpenStack DPE dashboard, select **Project** > **Activities**.

   The **Activities** panel appears.

2. Select the **Tasks** tab.

   A list of running, scheduled and canceled tasks appears.

   Tasks that are listed here are displayed by ID and not the name assigned to the job. To see a list of running tasks with the associated names, click the **Jobs** tab.

3. Locate the task that requires cancellation.
4. Click **Cancel task**.

   The task state changes to `canceled` and the **Cancelled** field changes to `True`.

Policies

For project administrators, policies are systems of rules that govern backup operations and data retention. You can apply these policies to the project instances to schedule backups.

Policy objects contain three child objects:

- Schedule - when a scheduled backup is performed.
- Retention policy - how long the backup is stored in the backup appliance
- Dataset - optional backup parameters

Clicking the **Policies** panel displays a list of the policies that are available to each project administrator, and provides management functions. Click the name of any policy to view the details of that policy.

The **State** column indicates whether the cloud administrator has enabled that policy for use by the project administrator. You can only perform backups with policies that are marked as enabled.

Managing project instances

The **Policies** page allows project administrators to associate a particular instance with a backup policy and start scheduled backup operations.

About this task

On the **Manage instances** dialog box, the left pane displays a list of instances that currently have no assigned policy. Instances in this pane are considered unprotected and are available to apply a policy. The right pane displays a list of instances that are protected by the current policy and provides controls to manage these instances.

If the project or policy contains many instances, use the **Filter** fields to narrow the list of results.

Procedure

1. From the OpenStack DPE dashboard, select **Project** > **Policies**.

   The **Policies** panel appears.

2. Locate the policy for which you want to manage the assigned instances, and then click **Manage instances**.

   The **Manage instances** dialog box appears.

3. To remove an instance from this policy, locate the instance in the right pane under **Protected instances** and then click the **-** symbol.

   The instance moves to the left pane, under **Unprotected instances**.

   Note: The operation only removes the instance from the current policy. Other policies may continue to protect the instance.

4. To assign an instance to this policy, locate the instance in the left pane under **Unprotected instances** and then click the **+** symbol.

   The instance moves to the right pane, under **Protected instances**.

5. Click **Save instances**.

Performing an ad hoc backup of an entire policy

This option performs a backup of all instances assigned to this policy without waiting for the scheduled start.

Procedure

1. From the OpenStack DPE dashboard, select **Project** > **Policies**.

   The **Policies** panel appears.

2. Locate the policy for which you want to perform the backup, and then click **Ad hoc backup** from the **Actions** drop-down.

   The backup is queued to start.

Results

OpenStack DPE signals the successful start of the backup operation. Review the progress under the **Activities** panel. The **Operation Type** for this job is `adhocBackup`.

File-level restoration

OpenStack DPE delivers support for file-level restoration (FLR) from instance backups to allow users to retrieve files from a backup without the need to complete a full restore operation. This feature provides the ability to restore specific files from a volume in a particular backup, or browse the files that are contained in the backup volumes.

The FLR service resides on the OpenStack DPE API VM, and controls FLR session management functions and scheduling.

Note: The FLR feature does not support restoring files directly into a running instance. Instead, the OpenStack DPE provides a graphical interface to browse and download files directly from the image backup.

FLR road map

The following road map illustrates a typical sequence of operations. The project administrator creates and closes FLR sessions on behalf of the user.

Before you begin

Note: FLR sessions have timeout values that automatically close the session after periods of inactivity. However, anyone who has the objects for an active session may browse and download files from the instance backup.

Procedure

1. On receipt of a request to restore files, the project administrator logs in to the OpenStack DPE Horizon plug-in.

2. The project administrator obtains a list of instances and associated backups.
3. The project administrator selects a specific instance backup and then creates an FLR session.
4. The OpenStack DPE provides a URL for the new FLR session, along with an FLR session ID and token.
5. The project administrator provides the FLR session URL, ID, and token to the user.
6. Using the FLR session objects and web interface, the user browses the files and folders that are contained within the instance backup and starts a file restoration operation.
7. The user notifies the project administrator after the FLR session completes.
8. The project administrator uses the FLR session objects to close the session from the OpenStack DPE Horizon plug-in.

Feature limitations

Observe the following limitations before using the FLR feature:

- The OpenStack DPE does not track absolute file paths within the instance OS, and is unaware of the mount points for each file system that was mounted within the instance OS. Only relative paths are supported.
- The OpenStack DPE supports a maximum of eight concurrent FLR sessions.
- The selection of individual file systems within an instance backup is not possible. The OpenStack DPE makes all file systems within the instance backup available to browse via the FLR session.
- File system support is limited to the file systems that `libguestfs` supports.
- Authentication and authorization are provided by the FLR session objects. Any user with the FLR session objects, however obtained, can access all of the files within the corresponding instance backup without further challenge.
- By default, OpenStack DPE supports a maximum of approximately 100,000 files in one folder. This is a technical limitation of `libguestfs`.

FLR session timers

Three timers govern the longevity of FLR sessions. The expiration of any of the three timers ends the session.

1. The first-access timer tracks the elapsed time between the creation of the FLR session and the first time that a user opens the FLR session URL or downloads a file. The default value is 2 hours.
2. The inactivity timer tracks the elapsed time between navigation requests via the FLR session URL, or between download requests. The default value is 15 minutes.
3. The end-of-life timer tracks the elapsed time since FLR session creation to establish a maximum session lifetime. Even if the session is active, this timer can end the session after a specified length of time. The default value is 8 hours.

Creating a file-level restore session

Log in as a project administrator before creating an FLR session.

Procedure

1. From the OpenStack DPE dashboard, select **Project** > **Instances**.

   The **Instances** panel appears.

2. Locate the instance in the list and then click the name of the instance.

   The instance details page appears.

3. Click the **Backups** tab.

   A list of completed backups for this instance appears. Any backup in this list can be used to create the FLR session.

4. Select an appropriate backup in the list and then click **Create FLR session** from the **Actions** drop-down.

   The **Create FLR session** dialog box appears.

5. Type a description for this FLR session.
6. Optional. Type new values (in minutes) for the **End-of-life timer interval**, **First-access timer interval**, and **Inactivity timer interval** fields.

   The default values are 480 minutes, 120 minutes, and 15 minutes, respectively. FLR session timers on page 69 provides additional information.

7. Click **Create FLR session**.

Results

The OpenStack DPE returns the FLR session details, including the FLR session ID, token, and URL. These three values are the FLR session objects.

Save the FLR session objects and provide them to the user. The user can now log in to the new FLR session.

Listing the active FLR sessions and retrieving the FLR session objects

Use the session object values to log in to an existing, active FLR session.

About this task

Note: You cannot edit an existing FLR session or change the associated timer intervals. If the timer intervals are incorrect, delete the existing session and create another session with the correct timer intervals.

Procedure

1. From the OpenStack DPE dashboard, select **Project** > **FLR sessions**.

   The **FLR sessions** panel appears.

2. Using the descriptions, locate an active FLR session in the list and then click the ID of the FLR session.

Results

OpenStack DPE lists the FLR session details, including the session ID, token, and URL.

Save the FLR session objects and provide them to the user. The user can now log in to the FLR session.

Using an FLR session to retrieve files from an instance backup

The OpenStack DPE provides a web interface for users to browse the contents of instance backups and retrieve individual files.

Before you begin

Request the FLR session ID, token, and URL from the project administrator. These objects are the FLR session login credentials. Anyone with these objects can access the instance backup.

The project administrator may have set timeouts to limit the length of time that you can access the instance backup.

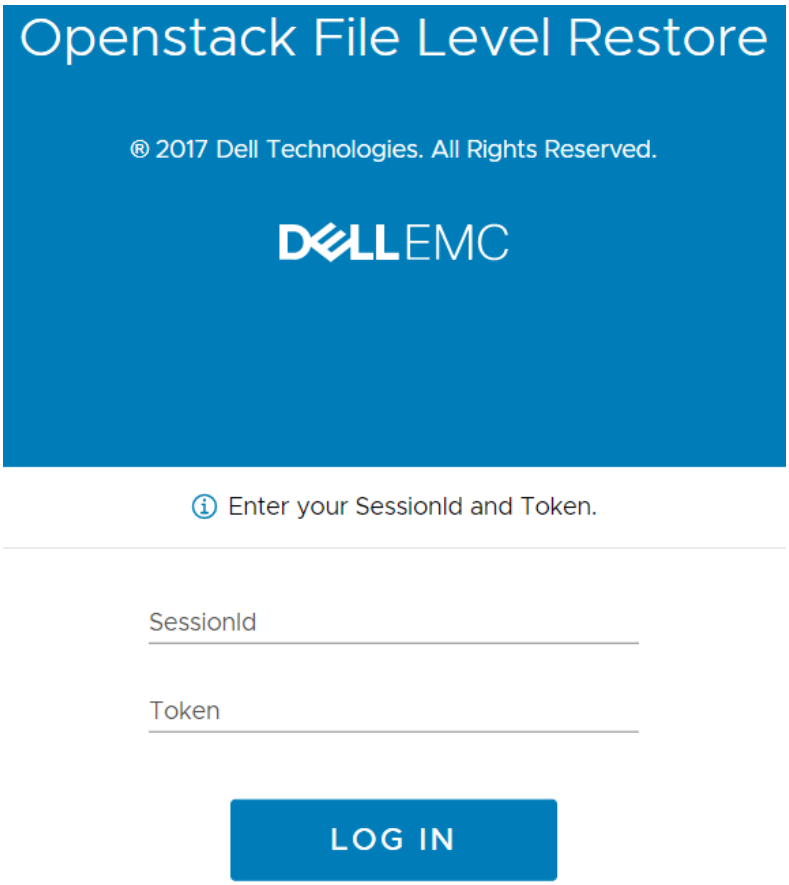
Procedure

1. In a web browser, type the FLR session URL that was provided by the project administrator.

   The **OpenStack File Level Restore** page appears.

   You may be required to acknowledge a browser warning regarding self-signed certificates before continuing.

   Figure 3 OpenStack File Level Restore page



2. Type the FLR session ID and token in the **SessionId** and **Token** fields, respectively.
3. Click **Log In**.

   The OpenStack DPE displays a list of volumes that are contained within the instance backup. The volume names identify the original mount point, file system type, and size.
4. To view the contents of a volume, click the volume name.

   The OpenStack DPE displays a list of the files and directories at the root level. This list also includes the object type, UNIX permissions, the system IDs that correspond to the owner and group, the file size, and the last modification date.
5. To view the contents of any directory, click the name of the directory.

   To return to the previous directory, use the navigational breadcrumbs above the list.
6. To download any file to the local computer, click the name of the file.
7. After you have restored the necessary files, click **Log Out** to close the FLR session.

Results

The OpenStack DPE automatically closes the FLR session when any of the three timeouts expire.

Deleting an FLR session

Deleting an FLR session terminates a user's access to the files inside the instance backup.

Procedure

1. From the OpenStack DPE dashboard, select **Project** > **FLR sessions**.

   The **FLR sessions** panel appears.

2. Locate the FLR session in the list and then click **Delete FLR session**.

   A confirmation dialog box appears.

Results

OpenStack DPE deletes the FLR session and removes it from the list on the **FLR sessions** panel. The user's session is no longer valid and cannot be used to view the contents of the backup.

APPENDIX A

Additional Information

This appendix includes the following topics:

- Proxy performance 74
- OpenStack availability zones 74
- Cinder drivers 74

Proxy performance

In an optimized OpenStack cloud infrastructure, the OpenStack DPE proxy is capable of backing up approximately 180GB of data per hour.

OpenStack availability zones

OpenStack availability zones enable the cloud administrator to arrange OpenStack compute hosts into logical groups, and provides a form of physical isolation and redundancy from other availability zones, such as by using a separate power supply or separate network equipment.

The cloud administrator defines the availability zone in which a specified compute host resides locally on each server. An availability zone is commonly used to identify a set of servers that have a common attribute. For instance, if some of the racks in your data center are on a separate power source, you can put servers in those racks in their own availability zone. Availability zones can also help separate different classes of hardware.

When users provision resources, they can specify from which availability zone they want their instance to be built. This allows cloud consumers to ensure that their application resources are spread across disparate machines to achieve high availability in the event of hardware failure.

Cinder drivers

The OpenStack DPE supports all storage arrays.

VNX snap copy

The VNX driver supports snap copy, which dramatically accelerates the process for creating a copied volume. By default, the driver does full data copy when creating a volume from a snapshot or when cloning a volume, which is time-consuming especially for large volumes. When snap copy is used, the driver simply creates a snapshot and mounts it as a volume when creating or when cloning a volume, which is instantaneous even for large volumes.

To enable this functionality, include the `--metadata snapcopy=True` option when creating cloned volume or creating volume from a snapshot:

```
cinder create --source-volid <source-void> --name "cloned_volume" --metadata
snapcopy=True
```

or

```
cinder create --snapshot-id <snapshot-id> --name "vol_from_snapshot" --metadata
snapcopy=True
```

The newly created volume is actually a snap copy instead of a full copy. If a full copy is needed, retype or migration can be used to convert the snap-copy volume to a full-copy volume, which may be time-consuming.

Determine whether the volume is a snap-copy volume or a full-copy volume by showing its metadata:

```
cinder metadata-show volume
```

If the `snapcopy` entry in the metadata is `true`, the volume is a snap-copy volume. Otherwise, it is a full-copy volume.

APPENDIX B

Troubleshooting

This appendix includes the following topics:

- OpenStack cloud infrastructure issues 76
- Networking issues 77
- Installation troubleshooting 79
- Management troubleshooting 81
- FLR troubleshooting 81

OpenStack cloud infrastructure issues

This section addresses issues that are related to OpenStack infrastructure.

The metadata_host parameter is missing from the nova configuration file

The OpenStack DPE service looks for the `metadata_host` parameter in the `etc/nova/nova.conf` file to discover instances. This flag must be present in the `Default` section of `nova.conf` on the node that runs the OpenStack DPE service.

OpenStack prerequisites on page 21 provides more information.

cloud-init is not configured

By default, certain settings are not enabled which is required for `cloud-init` to function. Ensure that the following settings are set to **`true`** in the `/etc/neutron/dhcp_agent.ini` file on the controller node.

About this task

OpenStack prerequisites on page 21 provides more information.

Procedure

1. Identify the Neutron configuration path and locate the file `dhcp_agent.ini`.

   For example, under RHEL, the file resides at `/var/lib/configdata/puppet-generated/neutron/etc/neutron/` on the controller node.

2. Locate the following settings in the `DEFAULT` section of `dhcp_agent.ini` and ensure that all settings are set to `true`:

   ```
   enable_isolated_metadata = true
   enable_metadata_network = true
   ```

3. Obtain the Neutron container service names by typing the following command:

   `docker ps | grep neutron`

   Note the container names for the Neutron container services.

4. Restart the Neutron container services by typing the following command:

   `docker restart <container>`

   where *<container>* is the name of each container service. Repeat this step for all Neutron container services.

Results

Note: If `cloud-init` is not configured correctly, deployment of the OpenStack DPE components fails to run the proper post-configuration tasks and the installation does not successfully complete.

An Oslo-related error causes the OpenStack DPE service to fail

If the Oslo messaging library is installed on the OpenStack DPE service host, verify and correct the installed version. The OpenStack DPE requires Oslo 9.6.0.

About this task

This task assumes that you are logged in to the controller or compute node where the OpenStack DPE service is installed.

Procedure

1. Verify the installed Oslo version by typing the following command:

   ```
   pip show oslo.messaging
   ```

2. If the Oslo version is anything other than 9.6.0, uninstall the existing Oslo library by typing the following command:

   ```
   pip uninstall oslo.messaging
   ```

3. Install Oslo 9.6.0 by typing the following command:

   ```
   pip install oslo.messaging==9.6.0
   ```

Networking issues

This section describes networking issues that may occur with the OpenStack DPE.

Ports for OpenStack DPE component communication

About this task

The OpenStack DPE ports must be configured at the firewall to allow for communication.

- The OpenStack DPE proxy instance connects to the OpenStack DPE service on a single TCP port. The port is configurable and defaults to 1947.
- The OpenStack DPE API instance connects to the OpenStack DPE service on a single TCP port. The port is configurable and defaults to 8443.

Procedure

1. Add the ports to `iptables` by typing the following commands on the node that runs the OpenStack DPE service:

   ```
   iptables -I INPUT -p tcp --dport 1947 -j ACCEPT
   iptables -I OUTPUT -p tcp --dport 1947 -j ACCEPT
   iptables -I INPUT -p tcp --dport 8443 -j ACCEPT
   iptables -I OUTPUT -p tcp --dport 8443 -j ACCEPT
   ```

2. SSH to the OpenStack DPE proxy instance.
3. Confirm that the ports are correctly configured by typing the following commands:

   ```
   curl http://dpe-proxy-service-IP:1947
   curl http://dpe-proxy-service-IP:8443
   ```

   Both commands should indicate successful communication and should not return errors such as `couldn't connect to host`.

Hostname resolution

About this task

The OpenStack DPE requires proper hostname resolution for backup and restore operations. In certain OpenStack network configurations, the proxy instance may not have a resolvable hostname. Proper operation of the OpenStack DPE may require hostname resolution using the `/etc/hosts` file.

For example, deploying the OpenStack DPE proxy instance on a private subnet with a floating IP address requires hostname resolution. The Avamar server communicates with the proxy through the hostname.

Procedure

1. Log in to the Avamar server.
2. Try to ping the hostname of the deployed OpenStack DPE proxy instance.

   If it is not pingable, add an entry for the OpenStack DPE proxy instance to the `/etc/hosts` file. Add one entry for each deployed proxy instance.
3. SSH into the OpenStack DPE proxy.
4. Try to ping the hostname of the Avamar server.

   If it is not pingable, add an entry for the Avamar server to the `/etc/hosts` file. Repeat this step for all deployed proxy instances.
5. If the Avamar server is configured with a Data Domain, try to ping the hostname of the Data Domain system.

   If it is not pingable, add an entry for the Data Domain system to the `/etc/hosts` file. Repeat this step for all deployed proxy instances.

OpenStack DPE instances cannot ping the OpenStack controller node

The following requirements apply to communication between the OpenStack DPE system and the OpenStack controller node:

About this task

- The OpenStack DPE proxy must be able to communicate with the OpenStack DPE service running on the OpenStack controller or compute node.
- The OpenStack DPE API must be able to communicate with the Keystone services running on the OpenStack controller node.

The OpenStack DPE proxy instance (and optionally, the OpenStack DPE API instance) run in a flat network on an OpenStack cloud. The controller node is a multi-homed host with access to the flat network as well as the admin network (and possibly the storage network and other networks).

This may present issues if the controller node is not reachable from the OpenStack DPE instances using the default network settings in effect on the controller node. If the controller node is running RHEL6 or newer, default settings may need to be changed to fix this problem. This might also be an issue on other flavors and versions of Linux.

Note: If an OpenStack DPE instance is unable to ping the OpenStack controller node, installation completes but backup and restore operations fail.

Procedure

1. Edit `/etc/sysctl.conf` and set the following values:

```
net.ipv4.conf.default.rp_filter = 2
net.ipv4.conf.all.rp_filter = 2
```

2. Make the new values permanent by typing the following command:

   sysctl -p

   Instances on the OpenStack flat network should now be able to ping the controller node. Verify the resolution to this issue by pinging the controller node from the OpenStack DPE proxy instance.

Installation troubleshooting

This section provides troubleshooting steps for issues related to installation.

The OpenStack DPE proxy instance fails to register to the protection provider

After deploying the OpenStack DPE proxy instance, checking the status of the `avagent` service returns the following:

`active (exited)` or `avagent Info: Client Agent is not running.`

Select the appropriate procedure below, depending on your network configuration.

If your OpenStack network is mixed internal and external

Procedure

1. SSH to the OpenStack DPE proxy instance.
2. Add the IP address, fully qualified domain name, and hostname for both the OpenStack DPE proxy instance and the Avamar server to the IPv4 section of `/etc/hosts`. For example:

   ```
   127.0.0.1  localhost.localdomain localhost
   127.0.0.1  proxy-hostname.novalocal proxy-hostname
   Proxy_IP  proxy-hostname.novalocal proxy-hostname
   Avamar_IP  avamar-hostname.example.com avamar-hostname
   ```

   where:

   - *proxy-hostname* is the hostname for the OpenStack DPE proxy instance.
   - *proxy-hostname.novalocal* is the fully qualified domain name for the OpenStack DPE proxy instance.
   - *Proxy_IP* is the IP address for the OpenStack DPE proxy instance.
   - *Avamar_IP* is the IP address for the Avamar server.
   - *avamar-hostname* is the hostname for the Avamar server.
   - *avamar-hostname.example.com* is the fully qualified domain name for the Avamar server.

   The fully qualified domain name can be found by typing `hostname -f` in an SSH session to the OpenStack DPE proxy instance or the Avamar server.

3. Type the following command:

   /etc/init.d/avagent register *Avamar_IP* /clients/*ZoneID* http://*username*:*password*@*dpe-proxy-service-IP*:*port*

   where:

   - *Avamar_IP* is the IP address of the Avamar server.
   - *ZoneID* is the ZoneID of the instance being registered.

- http://*username*:*password*@*dpe-proxy-service-IP*:*port* represents the URL and login information for the OpenStack DPE service (`dpe-proxy-service`).

If your OpenStack network is external only

Procedure

1. Enable `cloud-init` on OpenStack as described in OpenStack prerequisites on page 21.
2. Redeploy the OpenStack DPE proxy instance.

   If the problem persists, use the following procedure to manually register the proxy instance:

3. ssh to the OpenStack DPE proxy instance.
4. Type the following command:

   `/etc/init.d/avagent register Avamar_IP /clients/ZoneID http://username:password@dpe-proxy-service-IP:port`

   where:

   - *Avamar_IP* is the IP address of the Avamar server.
   - *ZoneID* is the ZoneID of the instance being registered.
   - http://*username*:*password*@*dpe-proxy-service-IP*:*port* represents the URL and login information for the OpenStack DPE service (`dpe-proxy-service`).

The OpenStack DPE API instance did not install completely

About this task

After deploying the OpenStack DPE API instance, the `systemctl status dpe-api` command returns the status:

```
dpe-api.service
   Loaded: not-found (Reason: No such file or directory)
   Active: inactive (dead)
```

Procedure

1. Enable `cloud-init` on OpenStack, as described in OpenStack prerequisites on page 21.
2. Redeploy the OpenStack DPE API instance.

   If the problem persists, complete the following steps to manually configure the OpenStack DPE API installation:

3. SSH to the OpenStack DPE API instance.
4. Type the following command:

   `./tmp/postconfig.sh http://Keystone-IP:5000 Avamar-REST-API-IP`

   where:

   - *Keystone-IP* is the IP address of the Keystone server.
   - *Avamar-REST-API-IP* is the IP address of the instance that runs the Avamar REST API.

Management troubleshooting

This section provides troubleshooting steps for issues related to the management of virtual machine backups.

OpenStack DPE lists a project as Not Available and you cannot unregister the project

If you have registered a project and instances with the OpenStack DPE and then delete them directly through OpenStack without first unregistering them, the OpenStack DPE prevents you from unregistering the project.

Procedure

1. SSH to the OpenStack DPE API instance.
2. Using a Linux text editor, such as `vi`, edit `/etc/dpe/dpe_api_policy.json`.
3. Locate the line beginning with `instance:delete`.

   Change this line to:

   ```
   "instance:delete": "rule:cloud_or_project_admin",
   ```

4. Save and close the file.
5. Type the following command:

   ```
   systemctl restart dpe-api
   ```

Results

The cloud administrator should be able to delete the instances and unregister the project from the protection provider.

FLR troubleshooting

This section provides troubleshooting steps for issues related to the FLR functionality and restoring individual files from virtual machine backups.

OpenStack DPE FLR cannot mount an XFS file system

This issue occurs with an XFS volume that was created by `mkfs.xfs` from `xfsprogs` version 3.2.4 or later.

About this task

The FLR service runs on SLES 12 with kernel 3.16, which does not support the XFS superblock version 5 advanced features. By default, with `xfsprogs` version 3.2.4 or later, `mkfs.xfs` enables the metadata CRC and free inode btree features, which the FLR service does not support.

The XFS documentation provides further information.

Procedure

1. To use the FLR feature for XFS volumes, disable both features by using the `-m` option when creating the volume:

   ```
   mkfs.xfs -m crc=0,finobt=0 /path/device
   ```

2. To check the superblock version used by a particular XFS volume, perform the following substeps:
   a. Start the XFS debugging tool by typing the following command:

      xfs_db /*path/device*

   b. Set the address to the current allocation group number by typing the following command:

      sb

   c. Print all of the field values by typing the following command:

      print

   d. Search the output for a string that starts with `versionnum` and then record this string.

      For example: `versionnum = 0xb4a5`

   e. Exit the XFS tool by typing the following command:

      quit

   f. Check the last digit of the recorded string. The last digit indicates the superblock version.

OpenStack DPE fails to create an FLR session

The OpenStack DPE may be unable to create an FLR session or mount files from the backup if the network connection between the OpenStack DPE API VM and the Avamar server is unstable.

About this task

The error message includes the phrase `ERROR dpe_flr.flrworker.backup_mount [-] list mount backup error, increase mount duration?`

Procedure

1. SSH to the OpenStack DPE API VM.
2. Using a Linux text editor, such as `vi`, edit `/etc/hosts`.
3. Add the Avamar server to the `hosts` file.

   For example:

   ```
   10.2.3.4  avamar-server.test.com
   ```

4. Save and close the file.
5. Using a Linux text editor, such as `vi`, edit `/etc/flr/flr.conf`.
6. Insert a line to increase the backup mount wait duration from the default of 5 seconds.

   For example:

   ```
   backup_mount_duration = 15
   ```

7. Save and close the file.
8. Restart the FLR service by typing the following command:

   systemctl restart flrservice