Manila OpenStack service - How to share filesystem in OpenStack

- Introduction
- How to create and use a CEPHFS_TYPE share
 - Request to be enabled to the service
 - Create the share
 - ° Mount the share on the VMs

Introduction

This guide is for users that aim to create a **shared filesystem in their virtual machines** using the OpenStack Manila service. The OpenStack Manila service allows the creation of a filesystem that can be shared among virtual machines in the same tenant (intra-tenant) or in different tenants (extra-tenant).

More details on Manila's service can be found here.

Important: In ADA Cloud, it is enabled only the CEPHFS_TYPE share, that allows the creation of a share in the CEPH storage. Important: Only intra-tenant shares are allowed on ADA Cloud.

How to create and use a CEPHFS_TYPE share

Request to be enabled to the service

A user that would like to make use of the Manila service needs to send an email to superc@cineca.it, communicating how many shares are needed and f or each share:

- its dimensions (GB)
- the instance name of the virtual machines (VMs) that will share that filesystem
- the tenant's name.

Once enabled by the User Support Team, the user needs to create the share and mount it in a special network interface on the VMs attached to a dedicated storage network.

As an example, in what follows, we will create a share in common between VM_alice and VM_bob belonging to the same tenant.

Create the share

You can create a share by following the below steps:

- Create the share by clicking on "Project Share Shares "Create Share" and set:
 - share name
 - share protocol == "CephFS" (other cases not allowed)
 - size (on the right side is visualized information about the actual available and used space within the tenant)
 - o Type == "cephfs_type"
 - ° Leave blank the option "Make visible for all projects" because it is not enabled
 - In the end, click on the "save" button.

	CIN	NECA • cin_US_	Create Share			×		🛔 dmolina1 ·
Progetto	So API	Progetto /	Share Name *					
Compute	>	Share	Share_test		Descrizione: Select parameters of share you	want to create.		
Volumi	>		Descrizione		Metadata:			+ Create Share
Container Infra Rete	>	Name		lte	One line - one action. Empty str To add metadata use:	rings will be ignored.	Share Group	Actions
Orchestrazione	>		Share Protocol *	~	key=value			
Database	>		Dimensione (GiB) *		Share Limits GB totali	0 of 1.000 GiB Used		
Data Processing Share	> ~		100 Share Type *	-	Number of Shares	0 of 50 Used		
:	Shares		cephfs_type	~				
Share Sna	pshots		Zona di Disponibilità	~				
Snare Ne Security Se	ervices		Share Group					
Share C	Groups		Notadata	~				
			metauata					

• Set the access rule on the share just created. On the OpenStack dashboard click on "Project Share Shares", then select the share just created and in the menu on the right select "Manage Rules".

Volumes	>										
Container Infra	>							Filter	Q	+ Create Share	🛍 Delete Shares
Network	>	Displaying 1 item									
Orchestration	>	□ Name	Description	Metadata	Size	Status	Protocol	Visibility 🕜	Share Network	Share Group	Actions
Database	>	□ Share_test	-		100GiB	Available	CEPHFS	private			Edit Share 🔻
Data Processing	>	Displaying 1 item								Res	ize Share
Share	~									Mar	nage Rules
Sha	ares									Del	ete Share
Share Snaps	nots										
Share Netwo	orks										
Security Servi	ices										

- Click on "Add rule" and set:
 - o access type: cephx
 - access level: read-write or read-only (depending on your needs)
 access to: write the name of the client (in our example "charlie")

		NECA • cin_US_	_clo_X 👻		a dm
roject	~	Project / S	Add Rule	3	C
API	I Access	Share	Access Type *		
Compute		Cephx Description:	Description:		
Volumes	>		Access Level *	Add policy rule to share, 'p' rule represents IPv4 or IPv6 address, 'user' rule represents username or usergroup, 'cephx' rule represents ceph auth ID.	
Container Infra	>		read-write ~		
Network	>	Access Typ	Access To *		Updated At Actions
Orchestration	>		charlie		
Database	>				
Data Processing	>			Cancel	
Share	~				
Share Sn	apshots				
Share N	letworks				
Security 5	Services				

By clicking on the "add" button the dashboard will show the "access key" and "access to" keys that must be used to mount the share on the virtual machines.

Mount the share on the VMs

The user is now ready to mount the share on VMs. In the following example, we will consider two VM with Ubuntu 20.04 OS. Please refer to the network guide of the operating system of your VM to be sure about the operations to be done.

Login in the first VM, then:

- configure the network interface attached to the storage network
 - "ip a" command lists all the network interfaces. Find the new interface, attached to the storage network, and refer to the mac-address of the interface to be sure.

Compute 🗸	VM_bob						Create Snapshot 👻
Overview		Luc Counts	Andrea Lana				
Instances	Overview	Log Console					
Images	Displaying 2 items						
Key Pairs	Name	Network	Fixed IPs	MAC Address	Status	Admin State	Actions
Server Groups	(c6970aa1-91b5)		• 10.47.4.18	fa:16:3e:a4:b3:66	Active	UP	Edit Security Groups 💌
Volumes >	(d9d545e8-dd99)	tenant_admin_net	• 192.168.0.116	fa:16:3e:4f:fb:f1	Active	UP	Edit Security Groups 👻
Network >	Displaying 2 items	<pre>buntugw-bob: : lo: <loopbac glen 1000 link/Loopbac inet 127.0. valid if : ens3: <broad default glen link/ether valid if : ens3: <broad ints/ether link/ether link/ether</broad </broad </loopbac </pre>	<pre>\$ ip a K, UP,LOWER_UP> ck 00:00:00:00 0.1/8 scope ho t forever pref CAST,MULTICAST 1000 fa:16:3e:4f:fb 8.0.116/24 brd t 830:77sec pref fa:16:3e:ff:f4 t forever pref CAST,MULTICAST fa:16:3e:a4:b3</pre>	mtu 65536 qdisc :00:00 brd 00:00; st lo erred_lft forever erred_lft forever uP,LOWER_UP> mtu :f1 brd ff:ff:ff: 192.108.0255 sc ferred_lft 839772 iffb1/04 scope l erred_lft forever mtu 1500 ddisc iso brd ff:ff:ff:	noqueue s 00:00:00:00:0 1450 qdi: ff:ff:ff ope globa ec ink noop state ff:ff:ff	tate UNKNOWN 90 sc fq_codel s l dynamic ens e DOWN group	group defaul tate UP grou 3 default qlen

• Create a new file in the /etc/netplan directory to configure such new interface (in our example "ens7"), and enable it

```
$ sudo su
$ cd /etc/netplan
$ cp <file-cloud-init> <file-ens-number>  - example: cp 50-cloud-init.yaml ens7.yaml -
$ vim <file-ens-number> ==> Modify the value of the fields "ens", "mtu <value>", "macaddress <value>" and "set-
name < value>" with the values shown by "ip a" command.
-- example of the ens7.yaml is --
network:
   version: 2
   ethernets:
       ens7:
           dhcp4: true
           match:
              macaddress: <your MAC address>
           mtu: <MTU value>
           set-name: ens7
$ netplan apply ==> to enable the new interface
$ ip a ==> check that the interface is enabled
   This file is generated from information provided by the datasource.
                                                                                             Changes
```

```
# to it will not persist across an instance reboot. To disable cloud-init's
# network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
network:
    version: 2
    ethernets:
        ens7:
            dhcp4: true
            macaddress: fa:16:3e:a4:b3:66
            mtu: 1500
            set-name: ens7
~
```

• Install the client, by installing the package named "ceph-common" (on Ubuntu)

• Create the mount point in the virtual machines (in our example "/mnt/share_manila") and mount the share

To mount the share you will need some informations contained in the "Share Overview" page on OpenStack dashboard, in particular you will need the values of PATH, ACCESS_TO and ACCESS KEY (here an example):

	Container Infra	>	Share Overview	
	Network	>	Name	Share_test
	Orchestration > ID Status Database > Export locations		ID Status	7243b403-b5cc-4058-ad25-a551479821da Available
			Export locations	Path:
	Data Processing	>		Professed: False
	Share	~	Visibility Availability zone Size	private nova 100 G/B
		Shares	Protocol	CEPHFS
	Share Snapshots		Share type	Name: cephfs_type ID: 15b58936-id3d-46a7-9758-98221e77d30e
Share Networks		tworks	Mount snapshot support	False
	Security Sec	ervices	Task state	None
	Share (Groups	Access Rules	
	Share Group Sna	pshots	cephx	Access for charling
	User Me	ssages		Access Lever. IW
lder	ntity	>	¢	Access Key: AQCelZvj+YIzCxAAM8VJv16GXk2PguzgyH/F/w== Created At: Dec. 16, 2022, 3:07 p.m. Updated At: Dec. 16, 2022, 3:07 p.m.

Beware that different versions of ceph-common are available for different versions of Ubuntu and the syntax of the mount command could change.

For Ubuntu 20.04, the command is:

```
sudo mount -t ceph -v <mount PATH> <mount point on the VM> -o name=<CLIENT_ACCESS_TO>,secret=<ACCESS_KEY>
```

An example of the complete command is:

```
sudo mount -t ceph -v 10.35.1.9:6789,10.35.1.10:6789,10.35.1.11:6789,10.35.1.12:6789,10.35.1.13:6789:/volumes
/_nogroup/43aa4ecc-1db6-4952-b2dd-6336b45075d5 /mnt/share_manila/ -o name=my-client-name,secret=AQBP07Nejv
/RLhAABYqQ5tvgePh2EP7EL0UuhQ==
```

While, for Ubuntu 22.04 and higher the command is:

```
sudo mount.ceph <CLIENT_ACCESS_TO>@482d24d4-df47-11eb-8d80-0c42alf53648.g100_fs=<FS_ADDRESS> <mount point on
the VM> -o mon_addr=<MON_IPS>,secretfile=<SECRET_FILE>
```

Where <FS_ADDRESS> and <MON_IPS> are the two parts of the "Path" string on OpenStack:

- <MON_IPS> is the first numeric part of the "Path" string, up to ":/volumes", where each IP has to be separated using the character "/" instead of "."
- ° <FS_ADDRESS> is everything else, from "/volumes/" to the end of the string

Finally, the <SECRET_FILE> is the path to a text file that contains the string <ACCESS_KEY>. Following the same example that uses the picture from above:

```
sudo mount.ceph charlie@482d24d4-df47-11eb-8d80-0c42a1f53648.g100_fs=/volumes/_nogroup/43aa4ecc-1db6-4952-b2dd-
6336b45075d5 /mnt/share_manila/ -o mon_addr=10.35.1.9:6789/10.35.1.10:6789/10.35.1.11:6789/10.35.1.12:6789/10.
35.1.13:6789,secretfile=/home/ubuntu/my_secret_file.txt
```

NB: If you are using a different Linux distribution, please refer to the ceph user manual to be sure that the syntax you are using is appropriate for the ceph version installed.

Then repeat the same steps for the second VM as well. Now the two VMs share the same filesystem.