


How to create and attach a volume

1. Log into the web dashboard (<https://prisma-cloud.ba.infn.it/>) with your username and password



The PRISMA logo features a stylized blue and yellow geometric design on the left, followed by the word "PRISMA" in a blue, sans-serif font.

Log In

User Name:

Password:

[Sign In](#)

2. In the left panel, click on **Project** → **Volumes** → **Create Volume**

The screenshot shows the PRISMA web interface. The left sidebar contains a navigation menu with the following items: Project, Compute, Overview, Instances, Volumes (highlighted with a blue bar), Images, Access & Security, Network, Object Store, and Orchestration. The main content area is titled 'Volumes & Snapshots' and has two tabs: 'Volumes' (active) and 'Volume Snapshots'. Below the tabs, there is a 'Volumes' section with a search filter and a '+ Create Volume' button. A table below the button lists the existing volumes. The table has columns: Name, Description, Size, Status, Type, Attached To, Availability Zone, and Actions. The first row shows a volume named 'volume-demo' with a size of 100GB, status 'Available', and type 'encrypted-data'. The 'Actions' column for this volume contains 'Edit Volume' and 'More' buttons. At the bottom of the table, it says 'Displaying 1 item'. Red arrows point from the text '2. In the left panel, click on Project → Volumes → Create Volume' to the 'Project' menu item, the 'Volumes' menu item, and the '+ Create Volume' button respectively.

PRISMA demo_outreach demo_user Sign Out

Project

- Compute
- Overview
- Instances
- Volumes**
- Images
- Access & Security
- Network
- Object Store
- Orchestration

Volumes & Snapshots

Volumes Volume Snapshots

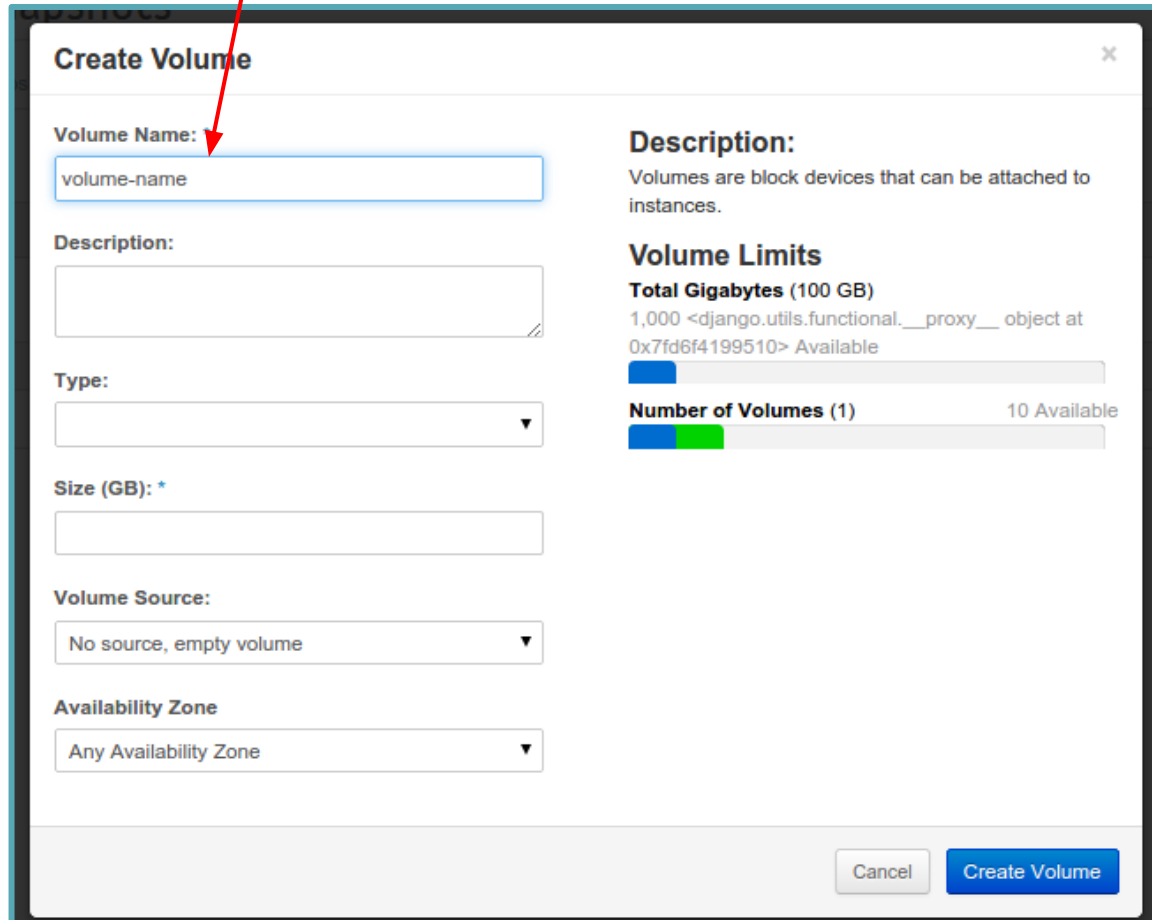
Volumes

Filter Filter

	Name	Description	Size	Status	Type	Attached To	Availability Zone	Actions
<input type="checkbox"/>	volume-demo		100GB	Available	encrypted-data		nova	<input type="button" value="Edit Volume"/> <input type="button" value="More"/>

Displaying 1 item

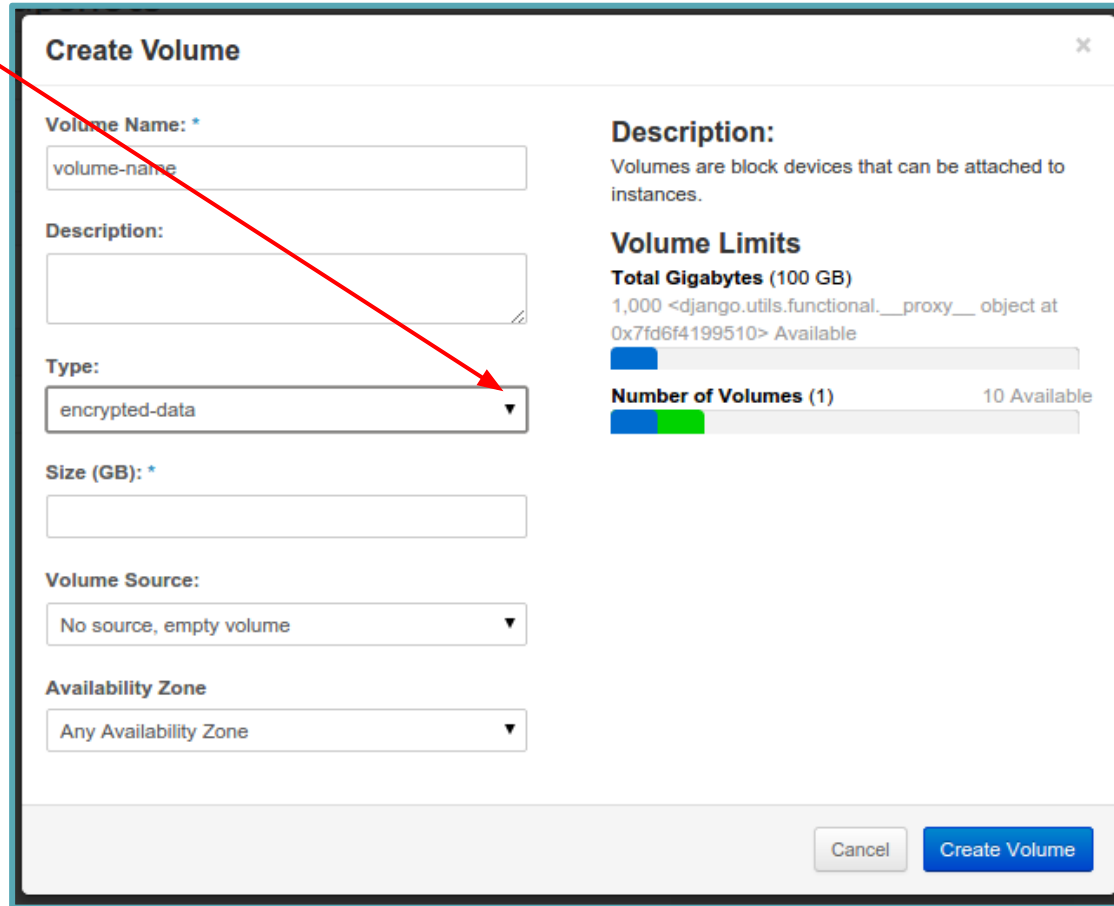
3. In the new window, type a name for the volume



The screenshot shows a 'Create Volume' dialog box with a red arrow pointing to the 'Volume Name' input field. The dialog box contains the following fields and sections:

- Volume Name:** A text input field containing 'volume-name'.
- Description:** A text input field.
- Type:** A dropdown menu.
- Size (GB): *** A text input field.
- Volume Source:** A dropdown menu with the option 'No source, empty volume'.
- Availability Zone:** A dropdown menu with the option 'Any Availability Zone'.
- Description:** A text area with the text 'Volumes are block devices that can be attached to instances.'
- Volume Limits:** A section containing two progress bars:
 - Total Gigabytes (100 GB):** A progress bar showing 1,000 <django.utils.functional.__proxy__ object at 0x7fd6f4199510> Available.
 - Number of Volumes (1):** A progress bar showing 10 Available.
- Buttons:** 'Cancel' and 'Create Volume' buttons at the bottom right.

4. Select a type (select “encrypted-data” if you want the volume to be encrypted)




The screenshot shows a 'Create Volume' dialog box with the following fields and sections:

- Volume Name:** *
Input field containing 'volume-name'.
- Description:**
Input field.
- Type:**
Dropdown menu with 'encrypted-data' selected. A red arrow points to this dropdown.
- Size (GB):** *
Input field.
- Volume Source:**
Dropdown menu with 'No source, empty volume' selected.
- Availability Zone:**
Dropdown menu with 'Any Availability Zone' selected.
- Description:**
Text: 'Volumes are block devices that can be attached to instances.'
- Volume Limits**
 - Total Gigabytes (100 GB)**
1,000 <django.utils.functional.__proxy__ object at 0x7fd6f4199510> Available
Progress bar showing 1,000 / 100 GB.
 - Number of Volumes (1)**
10 Available
Progress bar showing 1 / 10.

Buttons at the bottom: Cancel, Create Volume.

5. Insert the size you need for your volume



Create Volume

Volume Name: *

Description:

Type:

encrypted-data ▼

Size (GB): *

20

Volume Source:

No source, empty volume ▼

Availability Zone

Any Availability Zone ▼

Description:

Volumes are block devices that can be attached to instances.

Volume Limits

Total Gigabytes (100 GB)

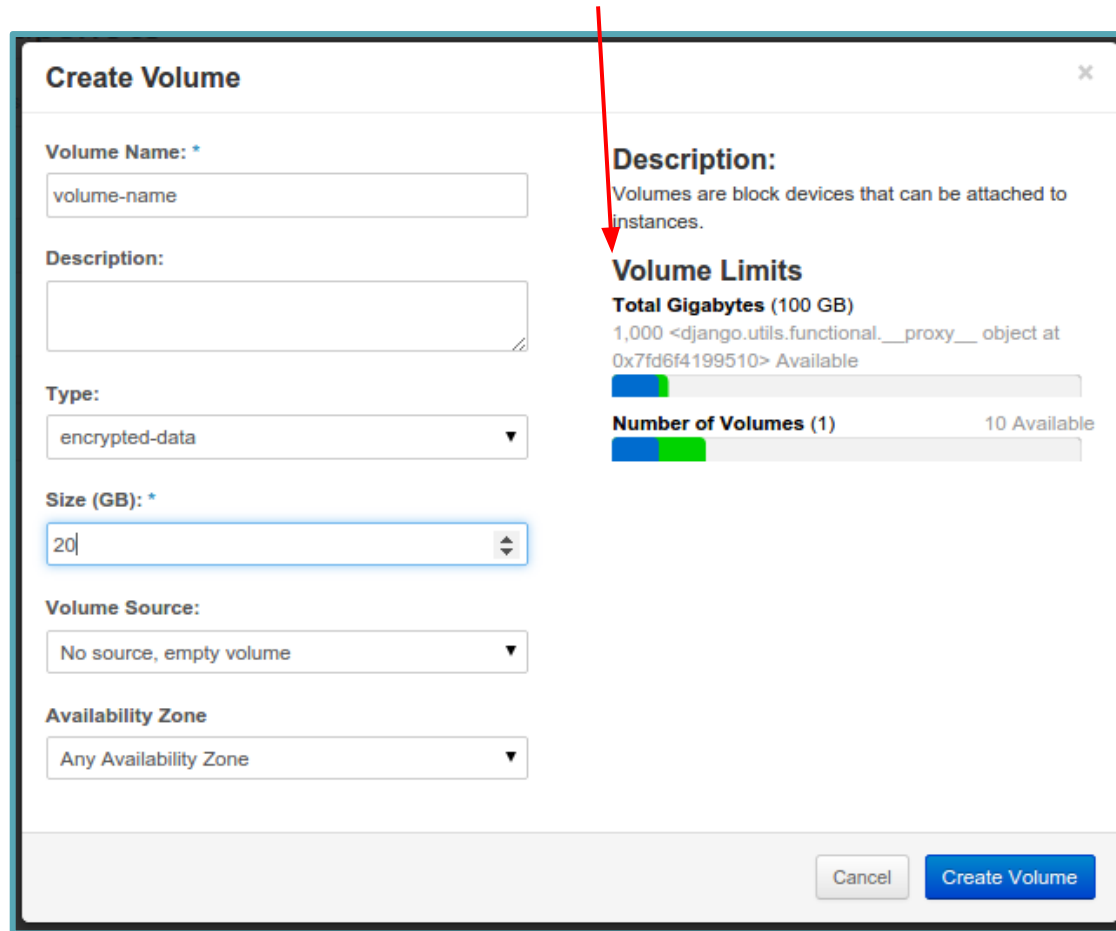
1,000 <django.utils.functional.__proxy__ object at 0x7fd6f4199510> Available

Number of Volumes (1) 10 Available

Cancel

Create Volume

NB: notice the size must be within available resources



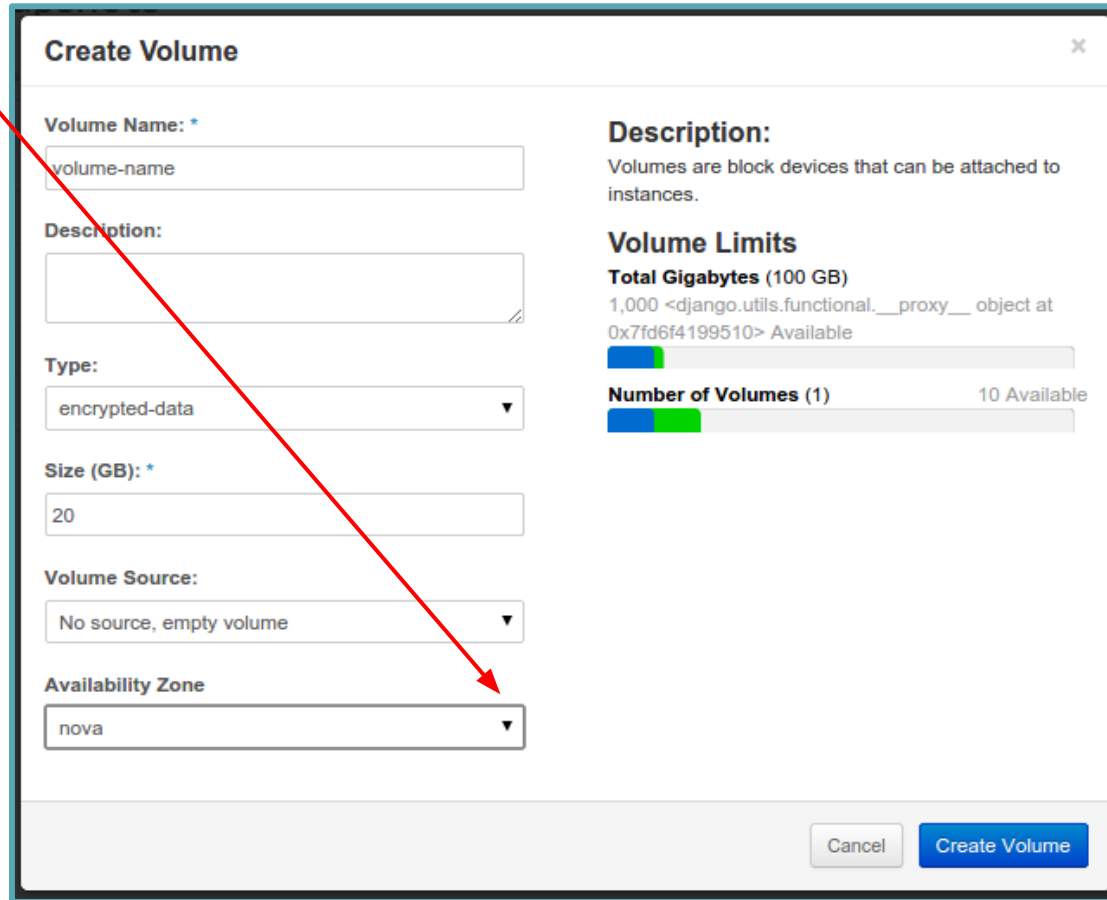
The screenshot shows a 'Create Volume' dialog box with the following fields and sections:

- Volume Name:** *
Input field containing 'volume-name'.
- Description:**
Input field.
- Type:**
Dropdown menu showing 'encrypted-data'.
- Size (GB):** *
Input field containing '20'.
- Volume Source:**
Dropdown menu showing 'No source, empty volume'.
- Availability Zone**
Dropdown menu showing 'Any Availability Zone'.
- Description:**
Text: 'Volumes are block devices that can be attached to instances.'
- Volume Limits**
 - Total Gigabytes (100 GB)**
Text: '1,000 <django.utils.functional.__proxy__ object at 0x7fd6f4199510> Available'
Progress bar showing approximately 2% usage (blue and green segments).
 - Number of Volumes (1)**
Text: '10 Available'
Progress bar showing approximately 10% usage (blue and green segments).

A red arrow points from the top center to the 'Volume Limits' section.

Buttons at the bottom: 'Cancel' and 'Create Volume'.

6. Select “Availability Zone” (recommended choice: “nova”)



The screenshot shows a 'Create Volume' dialog box with the following fields and options:

- Volume Name:** *
Input field containing 'volume-name'.
- Description:**
Input field.
- Type:**
Dropdown menu showing 'encrypted-data'.
- Size (GB):** *
Input field containing '20'.
- Volume Source:**
Dropdown menu showing 'No source, empty volume'.
- Availability Zone**
Dropdown menu showing 'nova'.

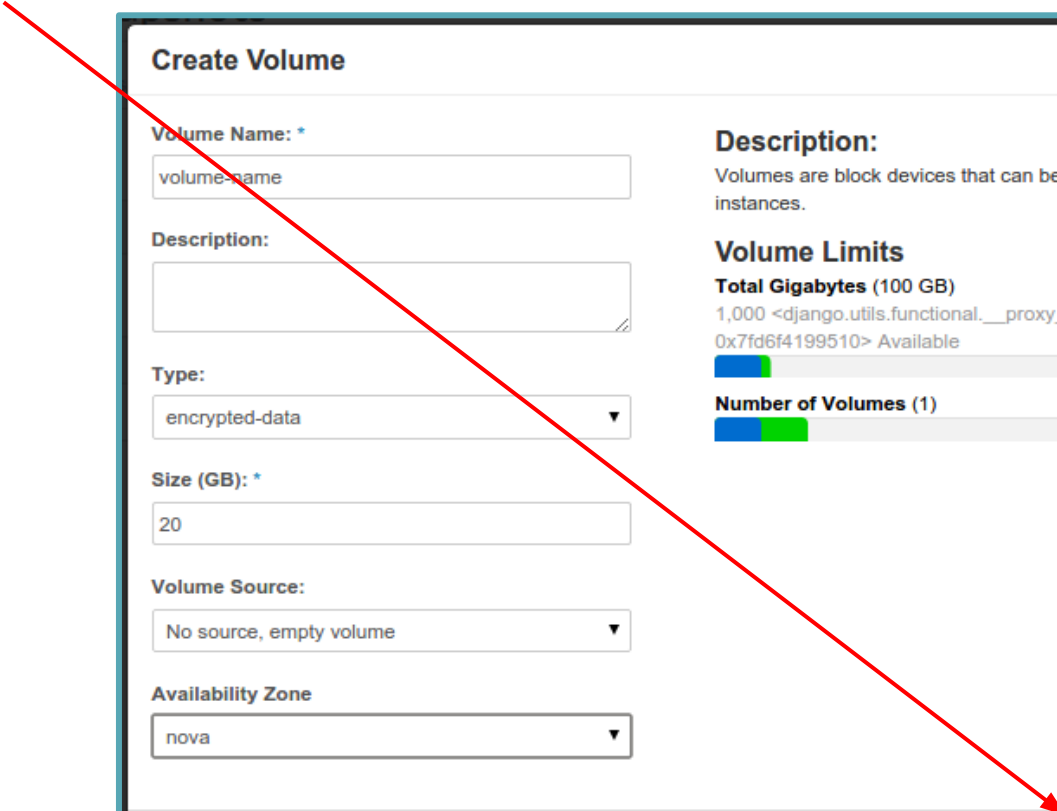
On the right side of the dialog, there is a **Description:** section stating 'Volumes are block devices that can be attached to instances.' and a **Volume Limits** section. The limits section includes:

- Total Gigabytes (100 GB)**
1,000 <django.utils.functional.__proxy__ object at 0x7fd6f4199510> Available
Progress bar showing approximately 10% usage.
- Number of Volumes (1)**
10 Available
Progress bar showing approximately 10% usage.

A red arrow points from the top left towards the 'Availability Zone' dropdown menu.

At the bottom right, there are two buttons: 'Cancel' and 'Create Volume'.

7. Click on Create Volume



Create Volume

Volume Name: *

Description:

Type:

encrypted-data ▼

Size (GB): *

Volume Source:

No source, empty volume ▼

Availability Zone

nova ▼

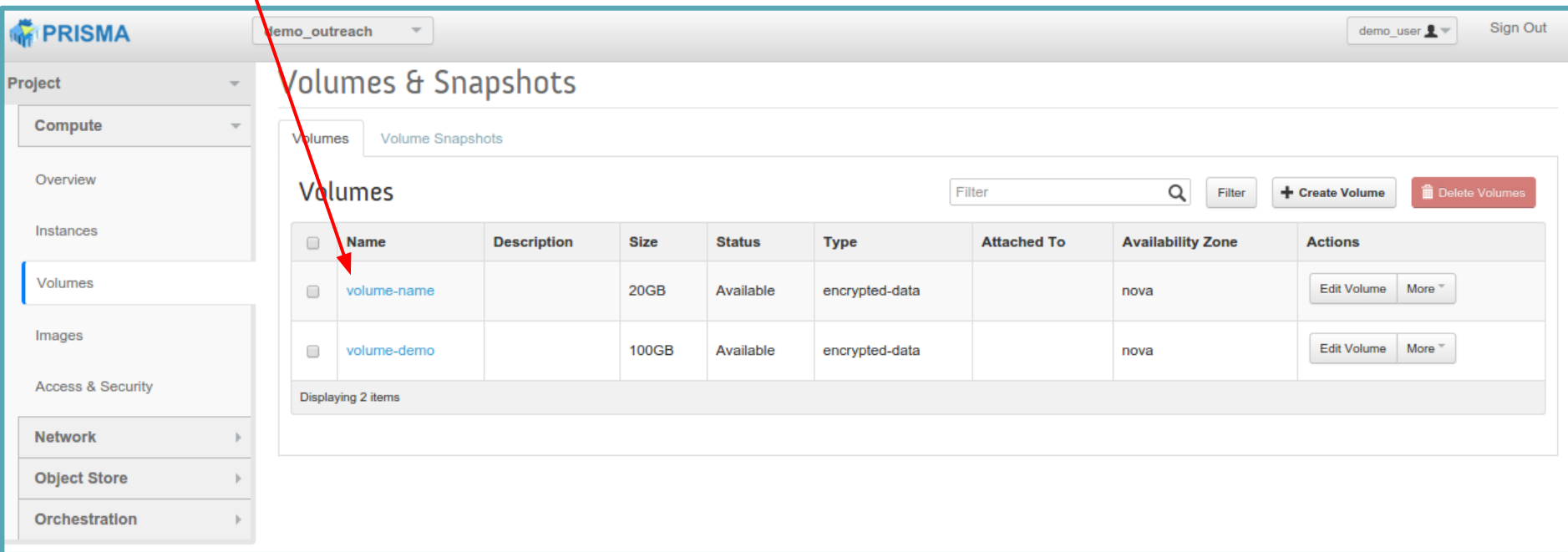
Description:
Volumes are block devices that can be attached to instances.

Volume Limits
Total Gigabytes (100 GB)
1,000 <django.utils.functional.__proxy__ object at 0x7fd6f4199510> Available

Number of Volumes (1) 10 Available

Cancel Create Volume

8. Check that the new volume is in the list in panel “Volumes”



The screenshot shows the PRISMA web interface. The left sidebar contains a navigation menu with the following items: Project, Compute, Overview, Instances, Volumes (highlighted), Images, Access & Security, Network, Object Store, and Orchestration. The main content area is titled "Volumes & Snapshots" and has two tabs: "Volumes" (active) and "Volume Snapshots". Below the tabs, there is a "Volumes" section with a search filter, a "Filter" button, a "+ Create Volume" button, and a "Delete Volumes" button. A table lists the volumes:

<input type="checkbox"/>	Name	Description	Size	Status	Type	Attached To	Availability Zone	Actions
<input type="checkbox"/>	volume-name		20GB	Available	encrypted-data		nova	Edit Volume More
<input type="checkbox"/>	volume-demo		100GB	Available	encrypted-data		nova	Edit Volume More

Below the table, it says "Displaying 2 items". A red arrow points from the instruction text to the "volume-name" entry in the table.

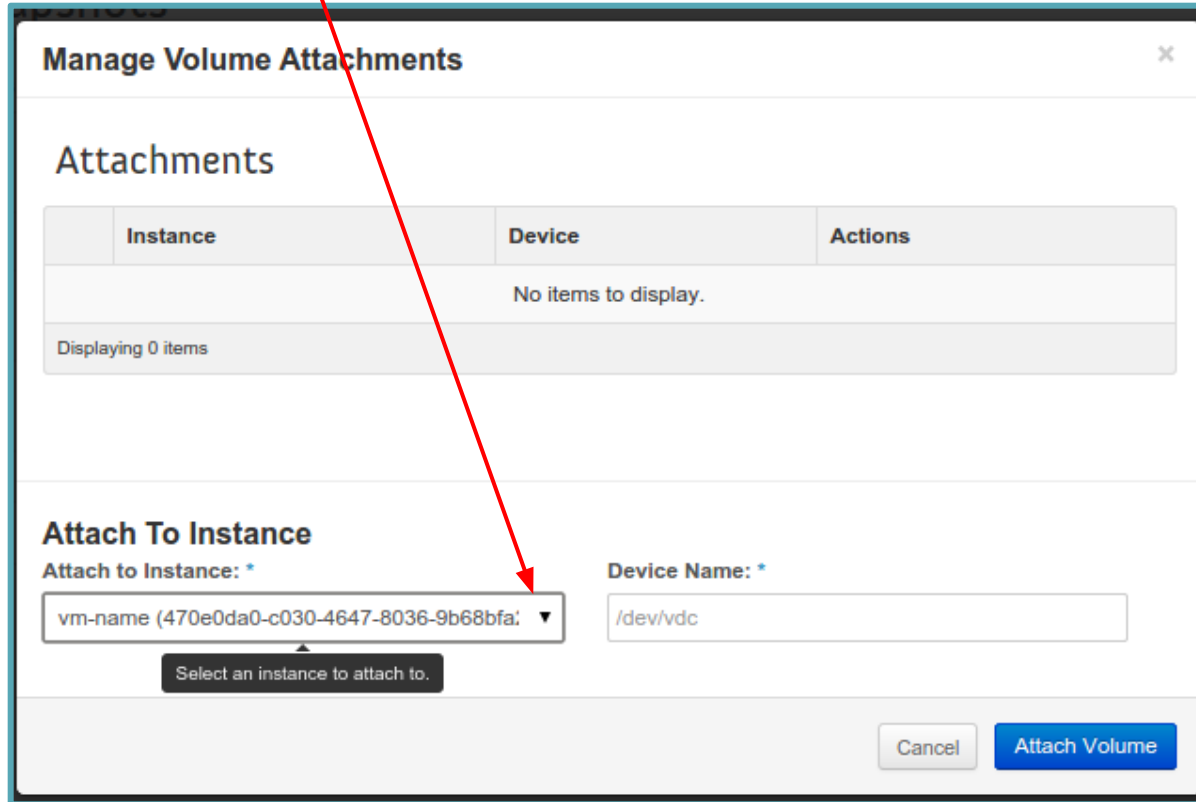
9. To attach it to a vm in the same tenant, click on
Edit Volume → **Edit Attachments**

The screenshot shows the PRISMA Volumes & Snapshots interface. The left sidebar contains navigation links: Project, Compute, Overview, Instances, Volumes (selected), Images, Access & Security, Network, Object Store, and Orchestration. The main content area is titled 'Volumes & Snapshots' and has tabs for 'Volumes' and 'Volume Snapshots'. Below the tabs is a table of volumes. The table has columns: Name, Description, Size, Status, Type, Attached To, Availability Zone, and Actions. Two volumes are listed: 'volume-name' (20GB, Available, encrypted-data) and 'volume-demo' (100GB, Available, encrypted-data). The 'Actions' column for the first volume shows a dropdown menu with options: Edit Volume, Extend Volume, Edit Attachments, Create Snapshot, and Delete Volume. Red arrows point from the text 'Edit Volume' and 'Edit Attachments' in the instruction above to the corresponding options in the dropdown menu.

	Name	Description	Size	Status	Type	Attached To	Availability Zone	Actions
<input type="checkbox"/>	volume-name		20GB	Available	encrypted-data		nova	Edit Volume More
<input type="checkbox"/>	volume-demo		100GB	Available	encrypted-data		nova	

Displaying 2 items

10. In the new window, select the vm to which you want to attach the volume



Manage Volume Attachments ×

Attachments

Instance	Device	Actions
No items to display.		
Displaying 0 items		

Attach To Instance

Attach to Instance: *

vm-name (470e0da0-c030-4647-8036-9b68bfa) ▼

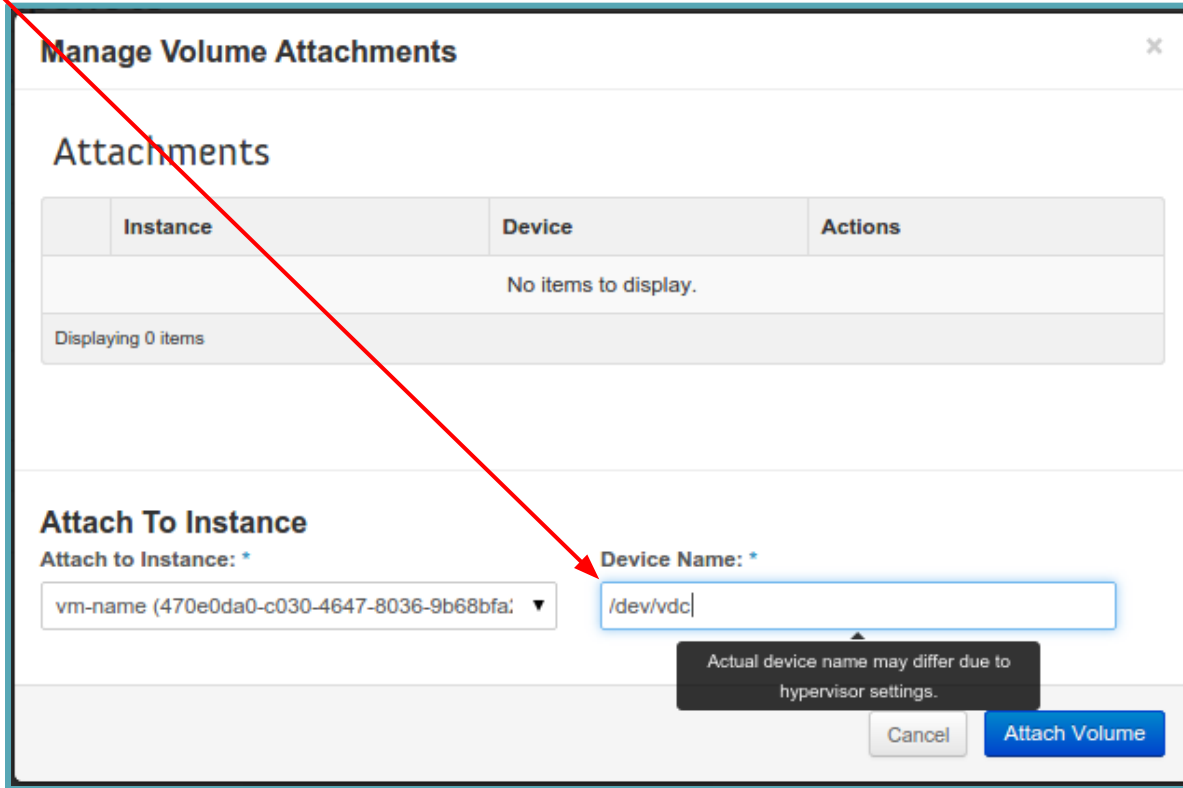
Select an instance to attach to.

Device Name: *

/dev/vdc

Cancel Attach Volume

11. Type a name for the device (notice that actual device name may differ)



Manage Volume Attachments ×

Attachments

Instance	Device	Actions
No items to display.		
Displaying 0 items		

Attach To Instance

Attach to Instance: *

vm-name (470e0da0-c030-4647-8036-9b68bfa: ▼

Device Name: *

/dev/vdc|

Actual device name may differ due to hypervisor settings.

Cancel Attach Volume

12. Click on Attach Volume

Manage Volume Attachments ×

Attachments

Instance	Device	Actions
No items to display.		
Displaying 0 items		

Attach To Instance

Attach to Instance: *

vm-name (470e0da0-c030-4647-8036-9b68bfa: ▼

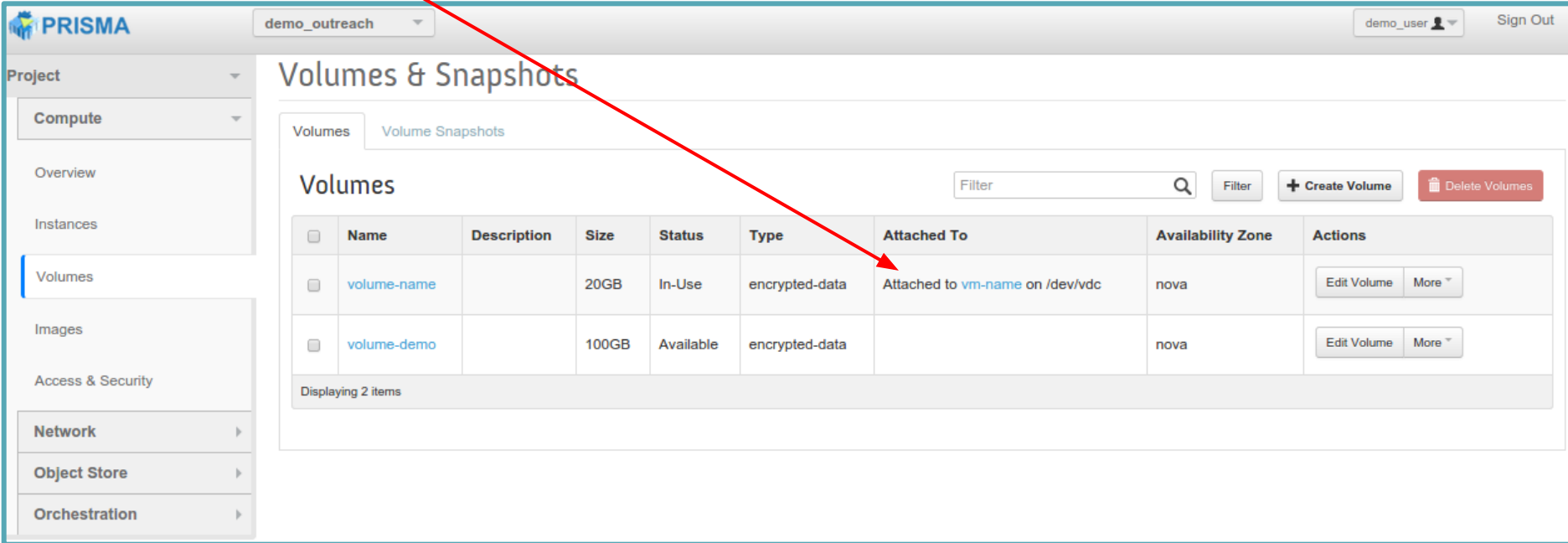
Device Name: *

/dev/vdc|

Actual device name may differ due to hypervisor settings.

Cancel Attach Volume

13. Check that the volume is now attached to the chosen vm



The screenshot shows the PRISMA web interface for managing Volumes and Snapshots. The left sidebar contains navigation links: Project, Compute, Overview, Instances, Volumes (selected), Images, Access & Security, Network, Object Store, and Orchestration. The main content area is titled 'Volumes & Snapshots' and has tabs for 'Volumes' and 'Volume Snapshots'. Below the tabs is a 'Volumes' section with a search filter and buttons for '+ Create Volume' and 'Delete Volumes'. A table lists the volumes:

<input type="checkbox"/>	Name	Description	Size	Status	Type	Attached To	Availability Zone	Actions
<input type="checkbox"/>	volume-name		20GB	In-Use	encrypted-data	Attached to vm-name on /dev/vdc	nova	Edit Volume More
<input type="checkbox"/>	volume-demo		100GB	Available	encrypted-data		nova	Edit Volume More

Displaying 2 items

14. Click on **Instances** and on the vm to which the volume is attached

PRISMA demo_outreach demo_user Sign Out

Project

Compute

Overview

Instances

Volumes

Images

Access & Security

Network

Object Store

Orchestration

Instances

Filter

+ Launch Instance

Soft Reboot Instances

Terminate Instances

	Instance Name	Image Name	IP Address	Size	Key Pair	Status	Availability Zone	Task	Power State	Uptime	Actions
<input type="checkbox"/>	vm-name	PRISMA-Ubuntu-14.04-x86_64	90.147.102.196	4cpu-4GB-20dsk 4GB RAM 4 VCPU 20.0GB Disk	nome_keypair	Active	nova	None	Running	1 day, 22 hours	Create Snapshot More
<input type="checkbox"/>	demo_desktop	PRISMA-Ubuntu-14.04-x86_64	90.147.102.80	4cpu-4GB-20dsk 4GB RAM 4 VCPU 20.0GB Disk	demo_keypair	Shutoff	nova	None	No State	1 week, 5 days	Start Instance More

Displaying 2 items

In the panel “Overview”, check that the volume is correctly attached

PRISMA demo_outreach demo_user Sign Out

Project Compute Overview Instances Volumes Images Access & Security Network Object Store Orchestration

Instance Details: vm-name

Overview Log Console

Instance Overview

Info

Name
vm-name

ID
470e0da0-c030-4647-8036-9b68bfa2facf

Status
Active

Availability Zone
nova

Created
March 24, 2015, 12:50 p.m.

Uptime
1 day, 22 hours

Specs

Flavor
4cpu-4GB-20disk

RAM
4GB

VCPUs
4 VCPU

Disk
20GB

IP Addresses

Public-Net
90.147.102.196

Security Groups

default
ALLOW IPv4 icmp from 0.0.0.0/0
ALLOW IPv6 to ::0
ALLOW IPv4 to 0.0.0.0/0
ALLOW IPv6 from default
ALLOW IPv4 22/tcp from 0.0.0.0/0

Meta

Key Name
nome_keypair

Image Name
PRISMA-Ubuntu-14.04-x86_64

Volumes Attached

Attached To
volume-name on /dev/vdc

How to make a partition, format and mount a device

1. As root, to see the list of devices, type

```
# fdisk -l
```

The result is similar to the following:

```
root@vm-name:~# fdisk -l

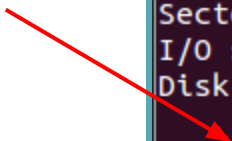
Disk /dev/vda: 21.5 GB, 21474836480 bytes
4 heads, 32 sectors/track, 327680 cylinders, total 41943040 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x000853f0

   Device Boot      Start         End      Blocks   Id  System
/dev/vda1    *        2048     41943039     20970496   83   Linux

Disk /dev/vdb: 21.5 GB, 21474836480 bytes
16 heads, 63 sectors/track, 41610 cylinders, total 41943040 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Disk /dev/vdb doesn't contain a valid partition table
```

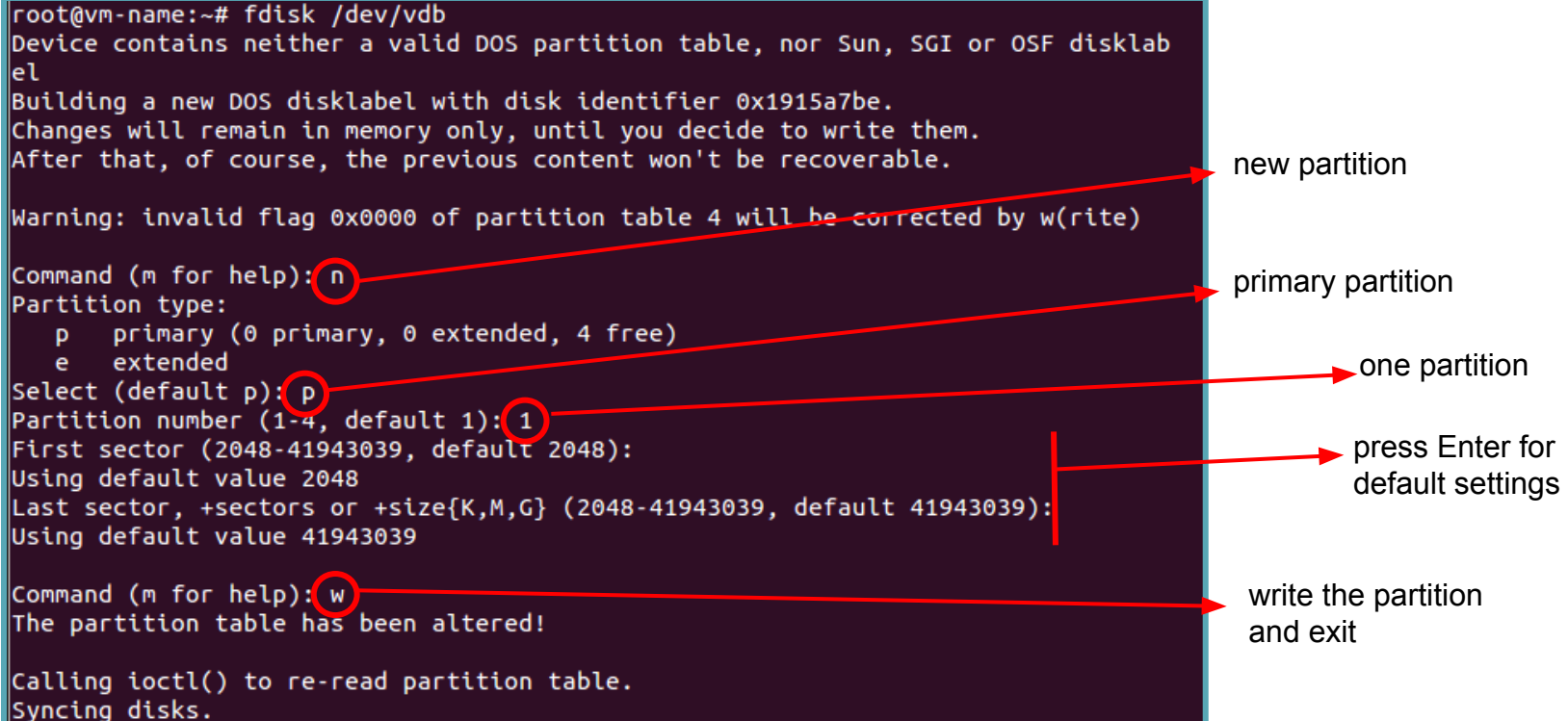
name of
disk



2. To make a partition, type (fdisk can make partitions up to 2 TB large)

```
# fdisk /dev/vdb
```

and answer the questions as shown here:



The screenshot shows a terminal window with the following text and annotations:

```
root@vm-name:~# fdisk /dev/vdb
Device contains neither a valid DOS partition table, nor Sun, SGI or OSF disklabel
Building a new DOS disklabel with disk identifier 0x1915a7be.
Changes will remain in memory only, until you decide to write them.
After that, of course, the previous content won't be recoverable.

Warning: invalid flag 0x0000 of partition table 4 will be corrected by w(rite)

Command (m for help): n
Partition type:
   p   primary (0 primary, 0 extended, 4 free)
   e   extended
Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-41943039, default 2048):
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-41943039, default 41943039):
Using default value 41943039

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
```

Annotations with red arrows pointing to the terminal output:

- new partition (points to 'n')
- primary partition (points to 'p')
- one partition (points to '1')
- press Enter for default settings (points to the empty line after 'Last sector...')
- write the partition and exit (points to 'w')

3. # fdisk -l

now gives:

```
root@vm-name:~# fdisk -l

Disk /dev/vda: 21.5 GB, 21474836480 bytes
4 heads, 32 sectors/track, 327680 cylinders, total 41943040 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x000853f0

   Device Boot      Start         End      Blocks   Id  System
/dev/vda1    *        2048     41943039     20970496   83   Linux

Disk /dev/vdb: 21.5 GB, 21474836480 bytes
3 heads, 34 sectors/track, 411206 cylinders, total 41943040 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x1915a7be

   Device Boot      Start         End      Blocks   Id  System
/dev/vdb1    *        2048     41943039     20970496   83   Linux
```

4. To format the device with an ext4 filesystem, type

```
# mkfs.ext4 /dev/vdb1
```



name of device

5. To mount the device, create a directory where the partition will be mounted:

```
# mkdir -p /disk/vdb1
```



6. and mount the device on that directory:

```
# mount /dev/vdb1 /disk/vdb1
```

7. Check your device has been mounted by looking at all the mounted partitions:

```
# mount
```