VULTR

Vultr Marketplace > Article

How to use Nextcloud with Vultr Block Storage and Object Storage

Learn how to integrate Nextcloud with Vultr Block Storage and Object Storage for enhanced file management, scalability, and reliable data backup solutions.

Contents

Introduction

Nextcloud is a self-hosted open-source file hosting application that lets you store, edit and share files privately using your own server. This article explains how to use Nextcloud with Vultr's Block Storage and Object Storage to maximize the application's storage potential.

Prerequisites

- Deploy a One-Click Nextcloud app from the Vultr Marketplace.
- Deploy a Block Storage volume and attach it to your server.
- Deploy an Object Storage Volume and create a new bucket.
- Create a domain pointing to the server.

1. Setup the Server

- Connect to the server using SSH and login as root.

1. Create a new server user account.

```
# adduser example-user
```

2. Grant the new user sudo privileges.

```
# adduser example-user sudo
```

3. Switch to the new user.

```
# su example-user
```

4. Request a Free Let's Encrypt SSL certificate using Certbot. Replace `cloud.example.com` with your linked domain.

```
$ sudo certbot --nginx -d cloud.example.com --agree-tos
```

5. To access Nextcloud using the domain, edit the `config.php` file.

```
$ sudo nano /var/www/html/config/config.php
```

6. Find the `trusted_domain` array.

```
'trusted_domains' =>
 array (
 0 => '<your-server-ip-address',
),
```

7. Add your domain as a new entry.

```
 'trusted_domains' =>
array (
0 => '<your-server-ip-address>',
1 => 'cloud.example.com',
),
```

   Save and exit the file.

8. Restart Nginx to load changes.

```
$ sudo systemctl restart nginx
```

Setup Nextcloud with Vultr Block Storage

2. Setup the Block Storage Volume

1. List all storage devices attached to your server.

```
$ lsblk
```

   Output:

```
 NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop1    7:1    0 70.3M  1 loop /snap/lxd/21029
loop2    7:2    0 55.4M  1 loop /snap/core18/2128
loop3    7:3    0 44.7M  1 loop /snap/snapd/15904
loop4    7:4    0 55.5M  1 loop /snap/core18/2409
loop5    7:5    0 61.9M  1 loop /snap/core20/1494
loop6    7:6    0 67.8M  1 loop /snap/lxd/22753
sr0     11:0    1 1024M  0 rom
vda    252:0    0   25G  0 disk
└─vda1 252:1    0   25G  0 part /
vdb    252:16   0   40G  0 disk
```

Your first block storage device displays as `vdb`. Any additional attachments display as `vdc`, `vdd`, and so on.

2. Set the volume disk label to `gpt`.

```
$ sudo parted -s /dev/vdb mklabel gpt
```

3. Create a new primary partition occupying the full disk space.

```
$ sudo parted -s /dev/vdb unit mib mkpart primary 0% 100%
```

4. Verify the new partition.

```
$ lsblk
```

Output:

```
 NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop1    7:1    0 70.3M  1 loop /snap/lxd/21029
loop2    7:2    0 55.4M  1 loop /snap/core18/2128
loop3    7:3    0 44.7M  1 loop /snap/snapd/15904
loop4    7:4    0 55.5M  1 loop /snap/core18/2409
loop5    7:5    0 61.9M  1 loop /snap/core20/1494
loop6    7:6    0 67.8M  1 loop /snap/lxd/22753
sr0     11:0    1 1024M  0 rom
vda    252:0    0   25G  0 disk
└─vda1 252:1    0   25G  0 part /
```

```
vdb    252:48   0   40G  0 disk
└─vdb1 252:49   0   40G  0 part
```

A new `vdb1` volume with the type `part` should display below the main `vdb` disk.

5. Create a new directory in `/mnt` to use as a mount point for volume.

```
$ sudo mkdir /mnt/nextcloud-data
```

6. Mount the block storage volume.

```
$ sudo mount -o defaults /dev/vdb1 /mnt/nextcloud-data/
```

7. Verify the new volume mount point.

```
$ lsblk
```

Output:

```
 NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop1    7:1    0 70.3M  1 loop /snap/lxd/21029
loop2    7:2    0 55.4M  1 loop /snap/core18/2128
loop3    7:3    0 44.7M  1 loop /snap/snapd/15904
loop4    7:4    0 55.5M  1 loop /snap/core18/2409
loop5    7:5    0 61.9M  1 loop /snap/core20/1494
loop6    7:6    0 67.8M  1 loop /snap/lxd/22753
sr0     11:0    1 1024M  0 rom
vda    252:0    0   25G  0 disk
 └─vda1 252:1    0   25G  0 part /
vdb    252:48   0   40G  0 disk
 └─vdb1 252:49   0   40G  0 part /mnt/nextdata
```

8. Enable persistent mounting to automatically mount the block storage volume at boot time.

```
$ sudo echo /dev/vdb1 /mnt/nextcloud-data ext4 defaults,noatime,nofail 0 0 >> /
etc/fstab
```


© Vultr 2025 | Page 6

9. Create a sample text file in the mount directory to confirm read and write access.

```
$ sudo echo "hello world, your nextcloud intergration works"> /mnt/nextcloud-data/sample.txt
```

3. Configure Nextcloud to use Block Storage

1. Using a web browser of your choice, visit your domain.

```
https://cloud.example.com
```

2. Login to Nextcloud.

> If you deployed Nextcloud as a OneClick Vultr App, you can find your default credentials on the cloud instance's panel.

3. Click your username in the upper right corner and select **Apps** from the drop-down list.
4. Find **External storage support** on the Apps list and enable it.
5. Navigate to your Username again and select **Settings**.
6. Under the Administration section, navigate to **External Storage**.
7. Click **Add storage**and select **Local** from the drop-down list.
8. Enter your **Mount Point** in the **Configuration** field and click the checkmark symbol ✓ to save changes.
9. A **green tick** indicates a successful connection to the mount point.

4. Test

1. Navigate to **Files**.

© Vultr 2025 | Page 7

2. Open the **Local** directory, or, select **External Storage** and open the block storage mount point.
3. Files stored on the block storage volume display, open the `sample.txt` file to confirm read access.
4. Upload files to the directory to confirm write access.

5. Setup Nextcloud with Vultr Object Storage

- Copy the Vultr Object Storage hostname.
- Copy your object storage Access and Secret Keys.

1. Through a web browser, visit your domain and Login to Nextcloud.
2. Enable the **External Storage** App if not already enabled.
3. Navigate to Settings, and access **External Storage** under the Administration section.
4. Click **Add Storage** and select **Amazon S3** from the drop-down list.
5. Enter your **Bucket** name, **Hostname** as displayed on your Object Storage instance panel, enable **SSL** and leave the **Port**, **Region** fields blank.
6. Toggle the **Authentication** drop down and select **Access Key**.
7. Paste the Access Key, and Secret Key as copied from your Vultr Object Storage instance panel.
8. For easy identification, change the Folder name to **Vultr Object Storage** or your preferred custom label.
9. Click the checkmark symbol ✓ to save changes.
10. A **green tick** indicates that your Vultr Object Storage is successfully connected.

6. Test the Object Storage

1. Navigate to **Files**.
2. Open the **Vultr Object Storage** directory (as set earlier), or, navigate to **External Storage** and select your S3 object storage.
3. Upload or create a sample file in the directory.
4. Log in to your Vultr account, open the **Object Storage** panel, and confirm that the new file is available in your bucket.

© Vultr 2025

7. More Information

You have successfully configured Nextcloud to use Vultr Block Storage and Object Storage volumes. You can connect multiple volumes to store different files per attachment. For more information, please visit the following resources.

- Vultr Object Storage documentation.
- Vultr Block Storage documentation.
- Nextcloud Documentation.

© Vultr 2025 | Page 9


VULTR