

Need live support within 30 minutes for mission-critical emergencies? Sign up for Premium Support! →

We're hiring

[Blog](#)

[Docs](#)

[Get Support](#)

[Sales](#)



[Tutorials](#) [Questions](#) [Learning Paths](#) [For Businesses](#) [Product Docs](#) [Social Impact](#)

## CONTENTS

Conditions préalables

Étape 1 — Téléchargement et installation des composants

Étape 2 — Création des répertoires de partage sur l'hôte

Étape 3 — Configuration des exportations NFS sur le serveur hôte

Étape 4 — Réglage du pare-feu sur l'hôte

Étape 5 — Création de points de montage et de répertoires de montage sur le client

Étape 6 — Test de l'accès NFS

Étape 7 — Montage des répertoires NFS à distance au démarrage

Étape 8 — Démontage d'un partage distant NFS

Conclusion

## RELATED

Comment passer à Ubuntu 20.04 Focal Fossa

[View](#) [↗](#)

Comment installer la distribution Anaconda pour Python sur Ubuntu 20.04

[View](#) [↗](#)



This site uses cookies and related technologies, as described in our [privacy policy](#), for purposes that may include site operation, analytics, enhanced user experience, or advertising. You may choose to consent to our use of these technologies, or manage your own preferences.

MANAGE CHOICES

AGREE & PROCEED

Published on June 11, 2020

Ubuntu   Ubuntu 20.04   Networking



By [Brian Boucheron](#)

Developer and author at DigitalOcean.

Français



## Introduction

NFS, ou Network File System, est un protocole de système de fichiers distribué qui vous permet de monter des répertoires distants sur votre serveur. Cela vous permet de gérer un espace de stockage dans un autre emplacement et d'y écrire depuis plusieurs clients. NFS fournit un moyen relativement standard et performant d'accéder à des systèmes distants sur réseau, et fonctionne bien dans les situations où un accès régulier aux ressources partagées est nécessaire.

Dans ce guide, nous allons voir comment installer le logiciel nécessaire à la fonctionnalité NFS sur Ubuntu 20.04, configurer deux montages NFS sur un serveur et un client, et monter et démonter les partages distants.

## Conditions préalables



### Try DigitalOcean for free

Click here to [Sign up](#) and get \$200 of credit to try our products over 60 days!

- Pour obtenir de l'aide pour configurer un utilisateur non **root** avec des privilèges `sudo` et un pare-feu, suivez notre guide de [configuration initiale du serveur avec Ubuntu 20.04](#).
- Si vous utilisez les DigitalOcean Droplets pour votre serveur et votre client, vous pouvez en savoir plus sur la mise en place d'un réseau privé dans notre documentation sur [la manière de créer un VPC](#).

Tout au long de ce tutoriel, nous faisons référence au serveur qui partage ses répertoires comme étant l'**hôte** et au serveur qui monte ces répertoires comme étant le **client**. Vous devrez connaître l'adresse IP de ces deux serveurs. Assurez-vous d'utiliser l'adresse de réseau *privée*, si elle est disponible.

Tout au long de ce tutoriel, nous ferons référence à ces adresses IP par les substituants `host_ip` et `client_ip`. Veuillez les remplacer si nécessaire.

## Étape 1 – Téléchargement et installation des composants

Nous commencerons par installer les composants nécessaires sur chaque serveur.

### Sur l'hôte

Sur le serveur **hôte**, installez le paquet `nfs-kernel-server`, qui vous permettra de partager vos répertoires. Comme c'est la première opération que vous effectuez avec `apt` dans cette session, rafraîchissez l'index de votre paquet local avant l'installation :

```
host:$ sudo apt update
host:$ sudo apt install nfs-kernel-server
```

Copy

Une fois que ces paquets sont installés, accédez au serveur **client**.

### Sur le client

Sur le serveur **client**, nous devons installer un paquet appelé `nfs-common`, qui fournit la fonctionnalité NFS sans inclure aucun composant serveur. Encore une fois, rafraîchissez l'index local du paquet avant l'installation pour vous assurer que vous



**Try DigitalOcean for free**

Click here to [Sign up](#) and get \$200 of credit to try our products over 60 days!

commencer à les configurer.

## Étape 2 – Création des répertoires de partage sur l'hôte

Nous allons partager deux répertoires séparés, avec des paramètres de configuration différents, afin d'illustrer deux façons principales dont les montages NFS peuvent être configurés en tenant compte de l'accès des super utilisateurs.

Les super-utilisateurs peuvent faire n'importe quoi n'importe où sur leur système. Cependant, les répertoires montés par NFS ne font pas partie du système sur lequel ils sont montés, donc par défaut, le serveur NFS refuse d'effectuer des opérations qui nécessitent des privilèges de super-utilisateur. Cette restriction par défaut signifie que les super-utilisateurs sur le **client** ne peuvent pas écrire des fichiers en tant que **root**, réassigner la propriété, ou effectuer toute autre tâche de super-utilisateur sur le montage NFS.

Parfois, cependant, il existe des utilisateurs de confiance sur le système **client** qui doivent effectuer ces actions sur le système de fichiers monté mais qui n'ont pas besoin d'un accès super-utilisateur sur l'**hôte**. Vous pouvez configurer le serveur NFS pour permettre cela, bien que cela introduise un élément de risque, car un tel utilisateur *pourrait* obtenir un accès root à l'ensemble du système **hôte**.

### Exemple 1 : Exportation d'un montage à usage général

Dans le premier exemple, nous allons créer un montage NFS polyvalent qui utilise le comportement par défaut de NFS pour rendre difficile pour un utilisateur ayant les privilèges root sur la machine **cliente** d'interagir avec l'**hôte** en utilisant ces privilèges de super-utilisateur **client**. Vous pourriez utiliser quelque chose de ce genre pour stocker des fichiers qui ont été téléchargés à l'aide d'un système de gestion de contenu ou pour créer un espace permettant aux utilisateurs de partager facilement des fichiers de projet.

Tout d'abord, créez le répertoire de partage :

```
host:$ sudo mkdir /var/nfs/general -p
```

Copy



**Try DigitalOcean for free**

Click here to [Sign up](#) and get \$200 of credit to try our products over 60 days!

### Output

```
drwxr-xr-x 2 root root 4096 May 14 18:36 .
```

Par mesure de sécurité, NFS traduira toutes les opérations **root** sur le **client** par les identifiants `nobody:nogroup`. Par conséquent, nous devons modifier la propriété du répertoire pour qu'il corresponde à ces identifiants.

```
host:$ sudo chown nobody:nogroup /var/nfs/general
```

[Copy](#)

Vous êtes maintenant prêt à exporter ce répertoire.

## Exemple 2 Exportation du répertoire d'accueil

Dans notre deuxième exemple, l'objectif est de rendre les répertoires personnels des utilisateurs stockés sur l'**hôte** disponibles sur les serveurs **clients**, tout en permettant aux administrateurs de confiance de ces serveurs **clients** l'accès dont ils ont besoin pour gérer les utilisateurs de manière pratique.

Pour ce faire, nous allons exporter le répertoire `/home`. Comme il existe déjà, nous n'avons pas besoin de le créer. Nous ne modifierons pas non plus les autorisations. Si nous *le faisons*, cela pourrait entraîner toute une série de problèmes pour quiconque possède un répertoire `home` sur la machine **hôte**.

## Étape 3 – Configuration des exportations NFS sur le serveur hôte

Ensuite, nous allons nous plonger dans le fichier de configuration NFS pour configurer le partage de ces ressources.

Sur la machine **hôte**, ouvrez le fichier `/etc/exports` dans votre éditeur de texte avec les privilèges **root** :

```
host:$ sudo nano /etc/exports
```

[Copy](#)

**Try DigitalOcean for free**

Click here to [Sign up](#) and get \$200 of credit to try our products over 60 days!

Nous devons créer une ligne pour chacun des répertoires que nous prévoyons de partager. Assurez-vous de modifier le champ `client_ip` représenté ici par votre adresse IP réelle

```
/etc/exports
```

```
/var/nfs/general    client_ip(rw,sync,no_subtree_check)
/home               client_ip(rw,sync,no_root_squash,no_subtree_check)
```

Ici, nous utilisons les mêmes options de configuration pour les deux répertoires à l'exception de `no_root_squash`. Voyons ce que signifie chacune de ces options :

- `rw` : cette option donne à l'ordinateur **client** un accès en lecture et en écriture au volume.
- `sync` : cette option oblige NFS à écrire les modifications sur le disque avant de répondre. Il en résulte un environnement plus stable et plus cohérent puisque la réponse reflète l'état réel du volume distant. Cependant, cela réduit également la vitesse des opérations sur les fichiers.
- `no_subtree_check` : cette option empêche la vérification de sous-arborescence, qui est un processus où l'**hôte** doit vérifier si le fichier est réellement encore disponible dans l'arborescence exportée pour chaque requête. Cela peut causer de nombreux problèmes lorsqu'un fichier est renommé alors que le **client** l'a ouvert. Dans la plupart des cas, il est préférable de désactiver la vérification des sous-arborescences.
- `no_root_squash` : par défaut, NFS traduit les requêtes d'un utilisateur **root** à distance en un utilisateur non privilégié sur le serveur. Cette fonction a été conçue comme un dispositif de sécurité pour empêcher un compte **racine** sur le **client** d'utiliser le système de fichiers de l'**hôte** comme **racine**. `no_root_squash` désactive ce comportement pour certains partages.

Lorsque vous avez terminé vos modifications, enregistrez et fermez le fichier. Ensuite, pour rendre les parts disponibles aux clients que vous avez configurés, redémarrez le serveur NFS avec la commande suivante :

```
host:$ sudo systemctl restart nfs-kernel-server
```

Copy

Cependant, avant de pouvoir utiliser les nouveaux partages, vous devez vous assurer



**Try DigitalOcean for free**

Click here to [Sign up](#) and get \$200 of credit to try our products over 60 days!

```
host:$ sudo ufw status
```

[Copy](#)

#### Output

```
Status: active
```

To	Action	From
--	-----	----
OpenSSH	ALLOW	Anywhere
OpenSSH (v6)	ALLOW	Anywhere (v6)

Sur notre système, seul le trafic SSH est autorisé, nous devons donc ajouter une règle pour le trafic NFS.

Pour de nombreuses applications, vous pouvez utiliser `sudo ufw app list` et les activer par leur nom, mais `nfs` n'en fait pas partie. Cependant, comme `ufw` vérifie également `/etc/services` pour le port et le protocole d'un service, nous pouvons toujours ajouter NFS par nom. Les meilleures pratiques recommandent d'activer la règle la plus restrictive qui permettra toujours le trafic que vous souhaitez autoriser, donc plutôt que d'autoriser le trafic de n'importe où, nous nous montrerons spécifiques.

Utilisez la commande suivante pour ouvrir le port 2049 sur l'**hôte**, en vous assurant de substituer l'adresse IP de votre **client** :

```
host:$ sudo ufw allow from client_ip to any port nfs
```

[Copy](#)

Vous pouvez vérifier le changement en saisissant :

```
host:$ sudo ufw status
```

[Copy](#)

Vous devriez voir le trafic autorisé à partir du port 2049 dans la sortie :

#### Output

```
Status: active
```



**Try DigitalOcean for free**

Click here to [Sign up](#) and get \$200 of credit to try our products over 60 days!



ordinateur **client**.

## Étape 5 – Création de points de montage et de répertoires de montage sur le client

Maintenant que le serveur **hôte** est configuré et sert ses partages, nous allons préparer notre **client**.

Afin de rendre les partages distants disponibles sur le **client**, nous devons monter les répertoires sur l'**hôte** que nous voulons partager dans des répertoires vides sur le **client**.

**Remarque :** s'il existe des fichiers et des répertoires dans votre point de montage, ils seront cachés dès que vous monterez le partage NFS. Pour éviter la perte de fichiers importants, assurez-vous que si vous montez dans un répertoire qui existe déjà, ce répertoire est bien vide.

Nous allons créer deux répertoires pour nos montages :

```
client:$ sudo mkdir -p /nfs/general  
client:$ sudo mkdir -p /nfs/home
```

Copy

Maintenant que nous disposons d'un emplacement pour placer les partages distants et que nous avons ouvert le pare-feu, nous pouvons monter les partages en utilisant l'adresse IP de notre serveur **hôte** :

```
client:$ sudo mount host_ip:/var/nfs/general /nfs/general  
client:$ sudo mount host_ip:/home /nfs/home
```

Copy

Ces commandes monteront les partages de l'ordinateur hôte sur la machine **cliente**. Vous pouvez vérifier de plusieurs façons que le montage a réussi. Vous pouvez le vérifier avec une commande `mount` ou `findmnt`, mais `df -h` fournit une sortie plus lisible :



**Try DigitalOcean for free**

Click here to [Sign up](#) and get \$200 of credit to try our products over 60 days!



tmpfs	491M	0	491M	0%	/dev/shm
tmpfs	5.0M	0	5.0M	0%	/run/lock
tmpfs	491M	0	491M	0%	/sys/fs/cgroup
/dev/vda15	105M	3.9M	101M	4%	/boot/efi
tmpfs	99M	0	99M	0%	/run/user/1000
10.132.212.247:/var/nfs/general	25G	1.8G	23G	8%	/nfs/general
10.132.212.247:/home	25G	1.8G	23G	8%	/nfs/home

Les deux partages que nous avons montés apparaissent en bas. Comme ils ont été montés à partir du même système de fichiers, ils présentent la même utilisation du disque. Pour voir combien d'espace est réellement utilisé sous chaque point de montage, utilisez la commande d'utilisation du disque `du` et le chemin du montage. L'indicateur `-s` fournit un résumé de l'utilisation plutôt que d'afficher l'utilisation pour chaque fichier. L'indicateur `-h` génère une sortie lisible par l'utilisateur.

Par exemple :

```
client:$ du -sh /nfs/home
```

Copy

Output

```
36K      /nfs/home
```

Ceci nous montre que le contenu de l'ensemble du répertoire d'accueil n'utilise que 36 K de l'espace disponible.

## Étape 6 – Test de l'accès NFS

Ensuite, testons l'accès aux partages en écrivant quelque chose à chacune d'entre eux.

### Exemple 1 : Partage à usage général

Tout d'abord, créez un fichier test dans le partage `/var/nfs/general`

```
client:$ sudo touch /nfs/general/general.test
```

Copy



**Try DigitalOcean for free**

Click here to [Sign up](#) and get \$200 of credit to try our products over 60 days!

#### Output

```
-rw-r--r-- 1 nobody nogroup 0 Aug  1 13:31 /nfs/general/general.test
```

Comme nous avons monté ce volume sans modifier le comportement par défaut de NFS et que nous avons créé le fichier en tant qu'utilisateur **root** de la machine **cliente** via la commande `sudo`, la propriété du fichier est par défaut `nobody:nogroup`. Les super-utilisateurs **clients** ne pourront pas effectuer des actions administratives typiques (comme le changement de propriétaire d'un fichier ou la création d'un nouveau répertoire pour un groupe d'utilisateurs) sur ce partage monté par NFS.

## Exemple 2 : Partage du répertoire d'accueil

Pour comparer les autorisations du partage d'usage général avec le partage de répertoire d'accueil, créez un fichier dans `/nfs/home` de la même manière :

```
client:$ sudo touch /nfs/home/home.test
```

[Copy](#)

Ensuite, examinez la propriété du fichier :

```
client:$ ls -l /nfs/home/home.test
```

[Copy](#)

#### Output

```
-rw-r--r-- 1 root root 0 Aug  1 13:32 /nfs/home/home.test
```

Nous avons créé `home.test` en tant que **racine** en utilisant la commande `sudo`, exactement de la même manière que nous avons créé le fichier `general.test`. Cependant, dans ce cas, il est la propriété de la **racine**, car nous avons remplacé le comportement par défaut lorsque nous avons spécifié l'option `no_root_squash` sur ce montage. Cela permet à nos utilisateurs **root** sur la machine **cliente** d'agir en tant que **root** et rend l'administration des comptes utilisateurs beaucoup plus pratique. En même temps, cela signifie que nous n'avons pas besoin de donner à ces utilisateurs un accès root sur l'hôte.

## Étape 7 – Montage des répertoires NFS à distance au



**Try DigitalOcean for free**

Click here to [Sign up](#) and get \$200 of credit to try our products over 60 days!

```
client:$ sudo nano /etc/fstab
```

[Copy](#)

En bas du fichier, ajoutez une ligne pour chacun de nos partages. Ils ressembleront à ceci :

/etc/fstab

```
. . .
host_ip:/var/nfs/general    /nfs/general    nfs auto,nofail,noatime,nolock,i
host_ip:/home              /nfs/home       nfs auto,nofail,noatime,nolock,i
```

**Note** : Vous pouvez trouver plus d'informations sur les options que nous spécifions ici dans la page man de NFS. Vous pouvez y accéder en exécutant la commande suivante :

```
$ man nfs
```

[Copy](#)

Le **client** montera automatiquement les partitions distantes au démarrage, bien que cela puisse prendre quelques instants pour établir la connexion et pour que les partages soient disponibles.

## Étape 8 – Démontage d'un partage distant NFS

Si vous ne souhaitez plus que le répertoire distant soit monté sur votre système, vous pouvez le démonter en sortant de la structure du répertoire de partage et en le démontant, comme ceci :

```
client:$ cd ~
client:$ sudo umount /nfs/home
client:$ sudo umount /nfs/general
```

[Copy](#)

Notez que la commande est nommée `umount` et non `unmount`, comme vous pourriez vous y attendre.



**Try DigitalOcean for free**

Click here to [Sign up](#) and get \$200 of credit to try our products over 60 days!

#### Output

Filesystem	Size	Used	Avail	Use%	Mounted on
udev	474M	0	474M	0%	/dev
tmpfs	99M	936K	98M	1%	/run
/dev/vda1	25G	1.8G	23G	8%	/
tmpfs	491M	0	491M	0%	/dev/shm
tmpfs	5.0M	0	5.0M	0%	/run/lock
tmpfs	491M	0	491M	0%	/sys/fs/cgroup
/dev/vda15	105M	3.9M	101M	4%	/boot/efi
tmpfs	99M	0	99M	0%	/run/user/1000

Si vous voulez également éviter qu'ils ne soient remontés au prochain redémarrage, éditez `/etc/fstab` et supprimez la ligne ou commentez la en plaçant un caractère `#` au début de la ligne. Vous pouvez également empêcher le remontage automatique en supprimant l'option `auto`, ce qui vous permettra de toujours pouvoir le monter manuellement.

## Conclusion

Dans ce tutoriel, nous avons créé un hôte NFS et illustré certains comportements clés du NFS en créant deux montages NFS différents, que nous avons partagés avec un client NFS.

Thanks for learning with the DigitalOcean Community. Check out our offerings for compute, storage, networking, and managed databases.

Si vous cherchez à implémenter NFS en production, il est important de souligner que le protocole lui-même n'est pas crypté. Dans les cas de partage sur un réseau privé, cela ne devrait pas poser de problème. Dans les autres cas, un VPN ou un autre type de tunnel crypté sera nécessaire pour protéger vos données

[Learn more about us →](#)

## Want to learn more? Join the DigitalOcean Community!



**Try DigitalOcean for free**

Click here to [Sign up](#) and get \$200 of credit to try our products over 60 days!

## About the authors



[Brian Boucheron](#) Author

Developer and author at DigitalOcean.

Still looking for an answer?

Ask a question

Search for more help

Was this helpful?

Yes

No



## Comments

### Leave a comment

**B** *I* U ☺ 📎 🖼️ ✎ H<sub>1</sub> H<sub>2</sub> H<sub>3</sub> ☰ ☷ ☶ ☵ ☴ ☳ ☲ ☱ “” ⓘ 🗑️ <>



Leave a comment...



#### Try DigitalOcean for free

Click here to [Sign up](#) and get \$200 of credit to try our products over 60 days!

## Sign In or Sign Up to Comment



This work is licensed under a Creative Commons Attribution-NonCommercial- ShareAlike 4.0 International License.

## Try DigitalOcean for free

Click below to sign up and get **\$200 of credit** to try our products over 60 days!

Sign up →

## Popular Topics

Ubuntu

Linux Basics

JavaScript

React

Python

Security

MySQL

Docker

Kubernetes

Browse all topic tags

[Free Managed Hosting →](#)



## Try DigitalOcean for free

Click here to [Sign up](#) and get \$200 of credit to try our products over 60 days!

Ask a question

DigitalOcean Support



Congratulations on unlocking the whale ambience easter egg! Click the whale button in the bottom left of your screen to toggle some ambient whale noises while you read.

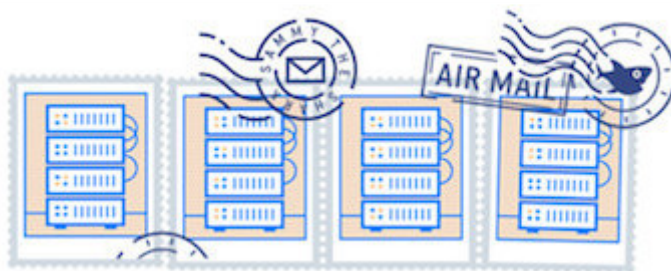


Thank you to the [Glacier Bay National Park & Preserve](#) and [Merrick079](#) for the sounds behind this easter egg.



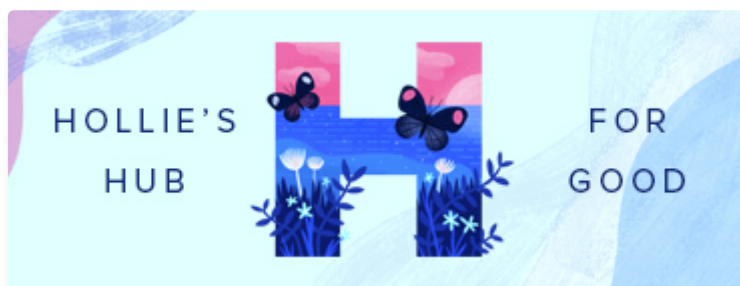
Interested in whales, protecting them, and their connection to helping prevent climate change? We recommend checking out the [Whale and Dolphin Conservation](#).

[Reset easter egg to be discovered again](#) / [Permanently dismiss and hide easter egg](#)



**GET OUR BIWEEKLY NEWSLETTER**

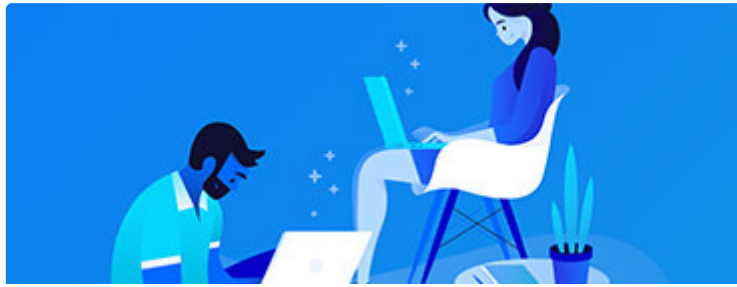
Sign up for Infrastructure as a  
Newsletter.



**Try DigitalOcean for free**

Click here to [Sign up](#) and get \$200 of credit to try our products over 60 days!





#### BECOME A CONTRIBUTOR

You get paid; we donate to tech nonprofits.

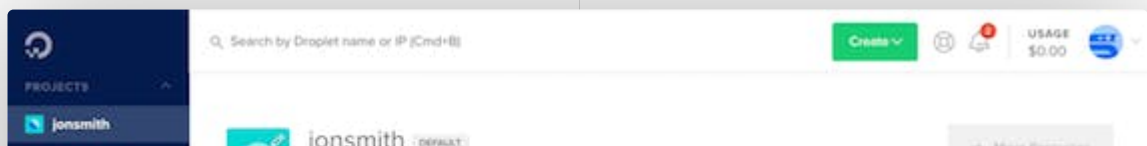
Featured on [Community](#) [Kubernetes Course](#) [Learn Python 3](#) [Machine Learning in Python](#)  
[Getting started with Go](#) [Intro to Kubernetes](#)

DigitalOcean Products [Virtual Machines](#) [Managed Databases](#) [Managed Kubernetes](#) [Block Storage](#)  
[Object Storage](#) [Marketplace](#) [VPC](#) [Load Balancers](#)

## Welcome to the developer cloud

DigitalOcean makes it simple to launch in the cloud and scale up as you grow – whether you're running one virtual machine or ten thousand.

[Learn More](#)

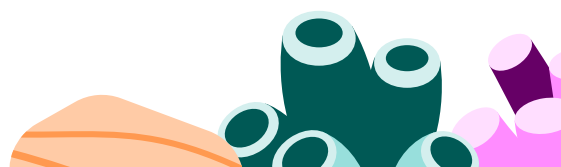
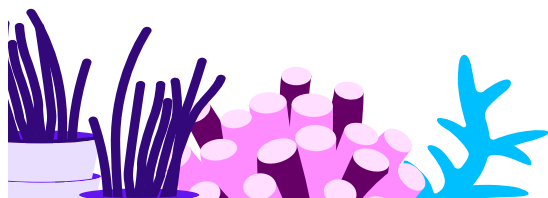


### Try DigitalOcean for free

Click here to [Sign up](#) and get \$200 of credit to try our products over 60 days!

Company	Products	Community	Solutions	Contact
About	Products Overview	Tutorials	Website Hosting	Support
Leadership		Q&A		Sales
Blog	Droplets	CSS-Tricks	VPS Hosting	Report Abuse
Careers	Kubernetes	Write for DO	Web & Mobile Apps	System Status
Customers	App Platform	Onations	Game Development	Share your ideas
Partners	Functions	Currents Research		
Channel Partners	Cloudways	Hatch Startup Program	Streaming VPN	
Referral Program	Managed Databases	deploy by DigitalOcean	SaaS Platforms	
Affiliate Program	Spaces	Shop Swag	Cloud Hosting for Blockchain	
Press	Load Balancers	Research Program	Startup Resources	
Legal	Block Storage	Open Source		
Security	Tools & Integrations	Code of Conduct		
Investor Relations	API	Newsletter Signup		
DO Impact	Pricing	Meetups		
	Documentation			
	Release Notes			
	Uptime			

© 2023 DigitalOcean, LLC. All rights reserved.



### Try DigitalOcean for free

Click here to [Sign up](#) and get \$200 of credit to try our products over 60 days!