

Documentation d'Exploitation

Mise en Place d'un Serveur Web avec Apache, Reverse Proxy et Load Balancing



Introduction

Ce guide détaille la mise en place d'un serveur web utilisant Apache sur une machine Ubuntu. Nous allons configurer deux applications web avec des sous-domaines, sécuriser les communications avec des certificats HTTPS, et mettre en place un reverse proxy avec répartition de charge.

Configuration de l'Infrastructure

- Architecture :

- **Serveurs Backend** : Deux serveurs hébergeant les applications web.
- **Machine Reverse Proxy / Load Balancer** : Une machine qui répartit la charge et agit comme un point d'entrée sécurisé pour les utilisateurs.
- **Pré-requis** :
 - Deux serveurs backend sous Ubuntu Server.
 - Une machine dédiée pour le reverse proxy/load balancer sous Ubuntu Server.
 - Accès SSH à toutes les machines.

Installation et Configuration

Serveurs Backend

- **Installation d'Ubuntu Server**
 - Téléchargez et installez Ubuntu Server depuis ubuntu.com.
 - Suivez les instructions pour créer une clé USB bootable et installer Ubuntu Server sur vos machines backend.
- **Installation d'Apache**

```
sudo apt update  
  
sudo apt install apache2
```

- **Déploiement des Applications Web**
 - Pour b1.com :

```
sudo mkdir -p /var/www/b1  
  
sudo cp -r /path/to/your/b1/* /var/www/b1/
```

- Pour blog.b1.com :

```
sudo mkdir -p /var/www/blog.b1  
  
sudo cp -r /path/to/your/blog.b1/* /var/www/blog.b1/
```

- **Configuration d'Apache pour les Applications**
 - Créer un fichier de configuration pour b1.com :

```
sudo nano /etc/apache2/sites-available/b1.conf
```

Contenu du fichier :

```
<VirtualHost *:80>

    ServerName b1.com

    DocumentRoot /var/www/b1

</VirtualHost>
```

- Créer un fichier de configuration pour blog.b1.com :

```
sudo nano /etc/apache2/sites-available/blog.b1.conf
```

Contenu du fichier :

```
<VirtualHost *:80>

    ServerName blog.b1.com

    DocumentRoot /var/www/blog.b1

</VirtualHost>
```

- **Activation des Sites et Redémarrage d'Apache**

```
sudo a2ensite b1.conf

sudo a2ensite blog.b1.conf

sudo systemctl restart apache2
```

Machine Reverse Proxy / Load Balancer

- **Installation d'Ubuntu Server**
 - Téléchargez et installez Ubuntu Server depuis ubuntu.com.
- **Installation d'Apache et des Modules Nécessaires**

```
sudo apt update  
  
sudo apt install apache2  
  
sudo a2enmod proxy  
  
sudo a2enmod proxy_http  
  
sudo a2enmod ssl  
  
sudo a2enmod proxy_balancer  
  
sudo a2enmod lbmethod_byrequests
```

- **Génération des Certificats Auto-Signés**

```
sudo mkdir /etc/apache2/ssl  
  
# Pour b1  
  
sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/apache2/ssl/b1.key -  
out /etc/apache2/ssl/b1.crt  
  
# Pour blog.b1  
  
sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout  
/etc/apache2/ssl/blog.b1.key -out /etc/apache2/ssl/blog.b1.crt  
  
# Pour le load balancer  
  
sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout  
/etc/apache2/ssl/loadbalancer.key -out /etc/apache2/ssl/loadbalancer.crt
```

- **Configuration des Hôtes Virtuels pour les Sous-Domains**
 - Créer et configurer le fichier pour b1:

```
sudo nano /etc/apache2/sites-available/b1.conf
```

Contenu du fichier :

```
<VirtualHost *:80>

    ServerName b1.com

    Redirect / https://www.b1.com/

</VirtualHost>


<VirtualHost *:443>

    ServerName b1.com

    DocumentRoot /var/www/b1


    SSLEngine on

    SSLCertificateFile /etc/apache2/ssl/b1.crt

    SSLCertificateKeyFile /etc/apache2/ssl/b1.key


    ProxyPreserveHost On

    <Proxy "balancer://b1cluster">

        BalancerMember http://192.168.1.10:80/b1

        BalancerMember http://192.168.1.11:80/b1

        ProxySet lbmethod=byrequests

    </Proxy>


    ProxyPass / balancer://b1cluster/

    ProxyPassReverse / balancer://b1cluster/

</VirtualHost>
```

- Créer et configurer le fichier pour app2 :

```
sudo nano /etc/apache2/sites-available/blog.b1.conf
```

Contenu du fichier :

```
<VirtualHost *:80>

    ServerName blog.b1.com

    Redirect / https://www.blog.b1.com/

</VirtualHost>


<VirtualHost *:443>

    ServerName blog.b1.com

    DocumentRoot /var/www/blog.b1


    SSLEngine on

    SSLCertificateFile /etc/apache2/ssl/blog.b1.crt
    SSLCertificateKeyFile /etc/apache2/ssl/blog.b1.key


    ProxyPreserveHost On

    <Proxy "balancer://blog.b1cluster">

        BalancerMember http://192.168.1.10:80/blog.b1
        BalancerMember http://192.168.1.11:80/blog.b1

        ProxySet lbmethod=byrequests
    </Proxy>


    ProxyPass / balancer://blog.b1cluster/
    ProxyPassReverse / balancer://blog.b1cluster/

</VirtualHost>
```

- Créer et configurer le fichier pour le load balancer :

```
sudo nano /etc/apache2/sites-available/loadbalancer.conf
```

Contenu du fichier :

```
<VirtualHost *:80>

    ServerName loadbalancer.example.com

    Redirect / https://loadbalancer.example.com/

</VirtualHost>


<VirtualHost *:443>

    ServerName loadbalancer.example.com


    SSLEngine on

    SSLCertificateFile /etc/apache2/ssl/loadbalancer.crt

    SSLCertificateKeyFile /etc/apache2/ssl/loadbalancer.key


    ProxyPreserveHost On

    <Proxy "balancer://mycluster">

        BalancerMember http://192.168.1.10

        BalancerMember http://192.168.1.11

        ProxySet lbmethod=byrequests

    </Proxy>


    ProxyPass / balancer://mycluster/

    ProxyPassReverse / balancer://mycluster/

</VirtualHost>
```


- **Activation des Sites et Redémarrage d'Apache**

```
sudo a2ensite app1.conf  
sudo a2ensite app2.conf  
sudo a2ensite loadbalancer.conf  
sudo systemctl restart apache2
```

Ajouter un Nouveau Site

- **Déployer l'Application Web**

- Créer le répertoire pour la nouvelle application (par exemple, myapp).

```
sudo mkdir -p /var/www/myapp  
sudo cp -r /path/to/your/myapp/* /var/www/myapp/
```

- **Générer un Certificat Auto-Signé pour le Nouveau Site**

```
sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout  
/etc/apache2/ssl/myapp.key -out /etc/apache2/ssl/myapp.crt
```

- **Configurer Apache pour le Nouveau Site**

- Créer un fichier de configuration pour myapp.

```
sudo nano /etc/apache2/sites-available/myapp.conf
```

Contenu du fichier :

```
<VirtualHost *:80>

    ServerName myapp.com

    Redirect / https://myapp.com/

</VirtualHost>


<VirtualHost *:443>

    ServerName myapp.com

    DocumentRoot /var/www/myapp


    SSLEngine on

    SSLCertificateFile /etc/apache2/ssl/myapp.crt

    SSLCertificateKeyFile /etc/apache2/ssl/myapp.key


    ProxyPreserveHost On

    <Proxy "balancer://myappcluster">

        BalancerMember http://192.168.1.10:80/myapp

        BalancerMember http://192.168.1.11:80/myapp

        ProxySet lbmethod=byrequests

    </Proxy>


    ProxyPass / balancer://myappcluster/

    ProxyPassReverse / balancer://myappcluster/

</VirtualHost>
```

- **Activer le Nouveau Site et Redémarrer Apache**

```
sudo a2ensite myapp.conf
```

```
sudo systemctl restart apache2
```

Conclusion

Ce guide couvre l'installation et la configuration d'un serveur web avec Apache, incluant un reverse proxy avec répartition de charge et l'utilisation de certificats auto-signés pour sécuriser les communications. En suivant ces étapes, vous pouvez déployer et gérer efficacement plusieurs applications web sur votre infrastructure.