



Documentation d'Architecture

Architecture Réseau et Serveurs pour une Infrastructure Web

Introduction

Cette documentation d'architecture détaille la configuration réseau, la répartition des services, et les bonnes pratiques mises en œuvre pour déployer une infrastructure web sécurisée et évolutive. L'architecture comprend des serveurs backend, un reverse proxy/load balancer, et l'utilisation de certificats auto-signés pour HTTPS.

Définition du Réseau

- **Réseau Local (LAN)** : Les serveurs backend et le reverse proxy/load balancer se trouvent dans le même réseau local.
- **Adresses IP** :
 - Serveur Backend 1 : 192.168.1.10
 - Serveur Backend 2 : 192.168.1.11
 - Reverse Proxy/Load Balancer : 192.168.1.20
- **Domaines** :
 - b1.com
 - blog.b1.com
 - loadbalancer.com

Configuration des Hôtes

Rôle	Adresse IP	Nom d'hôte	Services hébergés
Serveur Backend 1	192.168.1.10	backend1.com	b1.com, blog.b1.com
Serveur Backend 2	192.168.1.11	backend2.com	b1.com, blog.b1.com
Reverse Proxy/Load Balancer	192.168.1.20	loadbalancer.com	Apache, Proxy, Load Balancer

Répartition des Services

- **Sécurité :**
 - **HTTPS** : Utilisation de certificats SSL/TLS auto-signés pour sécuriser les communications.
 - **Pare-feu** : Configuration d'un pare-feu pour limiter les accès aux ports nécessaires (80, 443).
 - **Mises à Jour** : Maintenir les systèmes et les logiciels à jour pour bénéficier des dernières corrections de sécurité.
- **Performance :**
 - **Répartition de Charge** : Utilisation d'un load balancer pour répartir les requêtes entre les serveurs backend, améliorant la disponibilité et la performance.
 - **Caching** : Mise en place de mécanismes de cache (si nécessaire) pour réduire la charge sur les serveurs backend.
- **Fiabilité :**
 - **Redondance** : Deux serveurs backend pour assurer la continuité du service en cas de défaillance d'un serveur.
 - **Surveillance** : Mise en place de systèmes de monitoring pour surveiller la santé des serveurs et des services.

Configurations à Réaliser

Configuration Système

- **Installation d'Ubuntu Server**
 - Téléchargez et installez Ubuntu Server sur toutes les machines (backend et reverse proxy/load balancer).
 - Configurez les paramètres réseau (IP statiques) pour chaque serveur.

Configuration Réseau

- **Configuration des Adresses IP Statiques**
 - Éditez le fichier de configuration réseau `/etc/netplan/01-netcfg.yaml` sur chaque serveur :

network:

version: 2

ethernets:

eth0:

dhcp4: no

addresses:

- 192.168.1.10/24 # Backend 1

- 192.168.1.11/24 # Backend 2

- 192.168.1.20/24 # Reverse Proxy/Load Balancer

gateway4: 192.168.1.1

nameservers:

addresses:

- 8.8.8.8

- 8.8.4.4

- Appliquez la configuration.

```
sudo netplan apply
```

Configuration des Services

Serveurs Backend :

- **Installation d'Apache**

```
sudo apt update
```

```
sudo apt install apache2
```

- **Déploiement des Applications Web**

- Déployez b1 dans /var/www/b1 et blog.b1 dans /var/www/blog.b1.

- **Configuration d'Apache**

- Configurez des hôtes virtuels pour b1 et blog.b1.

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

- Activez les sites et redémarrez Apache.

```
sudo a2ensite b1.conf  
  
sudo a2ensite blog.b1.conf  
  
sudo systemctl restart apache2
```

Reverse Proxy/Load Balancer :

- **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 SSL/TLS Auto-Signés**

```
sudo mkdir /etc/apache2/ssl  
  
sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout  
/etc/apache2/ssl/b1.key -out /etc/apache2/ssl/b1.crt  
  
sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout  
/etc/apache2/ssl/blog.b1.key -out /etc/apache2/ssl/blog.b1.crt  
  
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**

- Configuration des Hôtes Virtuels pour les Sous-Domains

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

- **Activer les Sites et Redémarrer Apache**

```
sudo a2ensite b1.conf  
  
sudo a2ensite blog.b1.conf  
  
sudo a2ensite loadbalancer.conf  
  
sudo systemctl restart apache2
```

Exemple de Configuration de VirtualHost

- **Configuration de b1.conf**

```
<VirtualHost *:80>

    ServerName b1.com

    Redirect / https://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://app1cluster">

        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>
```

- **Configuration de blog.b1.conf**

```
<VirtualHost *:80>

    ServerName blog.b1.com

    Redirect / https://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>
```

- **Configuration de loadbalancer.conf**

```
<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>
```

Conclusion

Cette documentation d'architecture présente les éléments nécessaires pour déployer une infrastructure web sécurisée et performante, avec une répartition de charge et un reverse

proxy. En suivant ces étapes et configurations, vous assurerez une mise en œuvre réussie et maintenable de votre solution web.