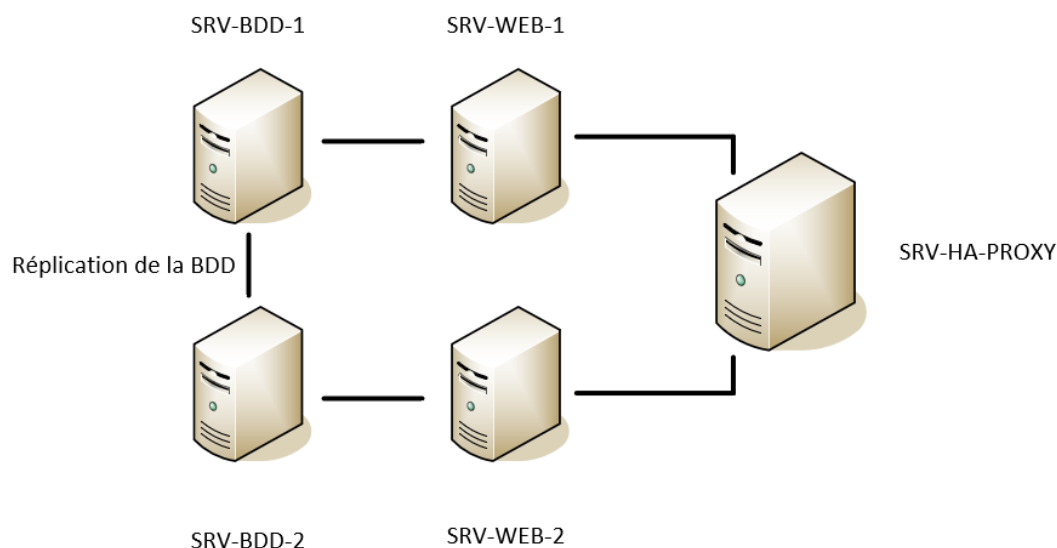


Mise en place d'une infrastructure web redondée avec bases de données répliquées

Schéma d'architecture finale



Introduction

Ce projet a pour objectif de mettre en œuvre une infrastructure web avec répartition de charge et haute disponibilité. Cela inclut deux serveurs web Apache, deux bases de données MariaDB synchronisées entre elles et un proxy de répartition de charge basé sur HAProxy. Le site déployé permet la prise de rendez-vous via un formulaire PHP connecté à la base distante.

1. Préparer l'infrastructure

Avant tout, vous devez préparer cinq machines Debian 12 avec les rôles suivants :

- WebSrv1 : 172.16.1.20
- WebSrv2 : 172.16.1.21
- HAProxy : 172.16.1.22
- DB1 : 172.16.1.23
- DB2 : 172.16.1.24

Chacune doit avoir un accès administrateur et être joignable en réseau. Assurez-vous que tous les hôtes ont des IP fixes ou sont bien définis dans votre DNS/local.

2. Déployer les serveurs Web

Chaque serveur Web doit héberger un site PHP identique. Celui-ci contiendra un formulaire HTML permettant d'enregistrer des rendez-vous dans une base de données distante. Apache2 et PHP doivent être installés.

 **Voir la documentation : *Création d'un serveur web Apache***

- Créez un fichier `index.php` avec formulaire
- Ajoutez le code PHP pour insérer dans la base distante
- Testez l'accessibilité de WebSrv1 et WebSrv2 via leurs IP respectives

SRV-WEB-1



SRV-WEB-2



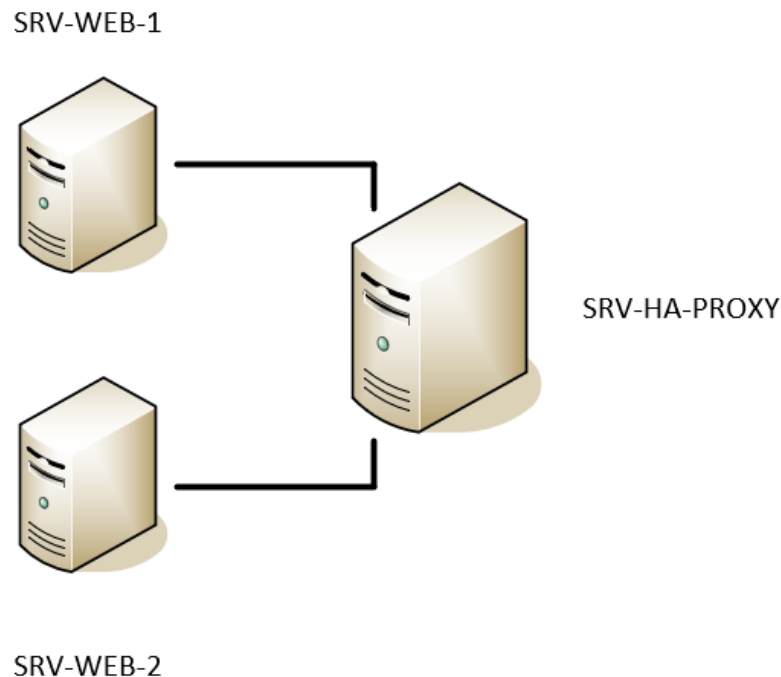
3. Configurer le proxy HAProxy

Le proxy HAProxy jouera le rôle de point d'entrée unique pour les utilisateurs. Il distribuera équitablement les requêtes vers WebSrv1 et WebSrv2.

 Voir la documentation : HAProxy – Répartition de charge HTTP

- Installez HAProxy
- Définissez une frontend HTTP écoutant sur le port 80
- Ajoutez une backend avec les deux serveurs web déclarés

Une fois en place, testez en accédant à l'adresse IP du proxy (172.16.1.22).

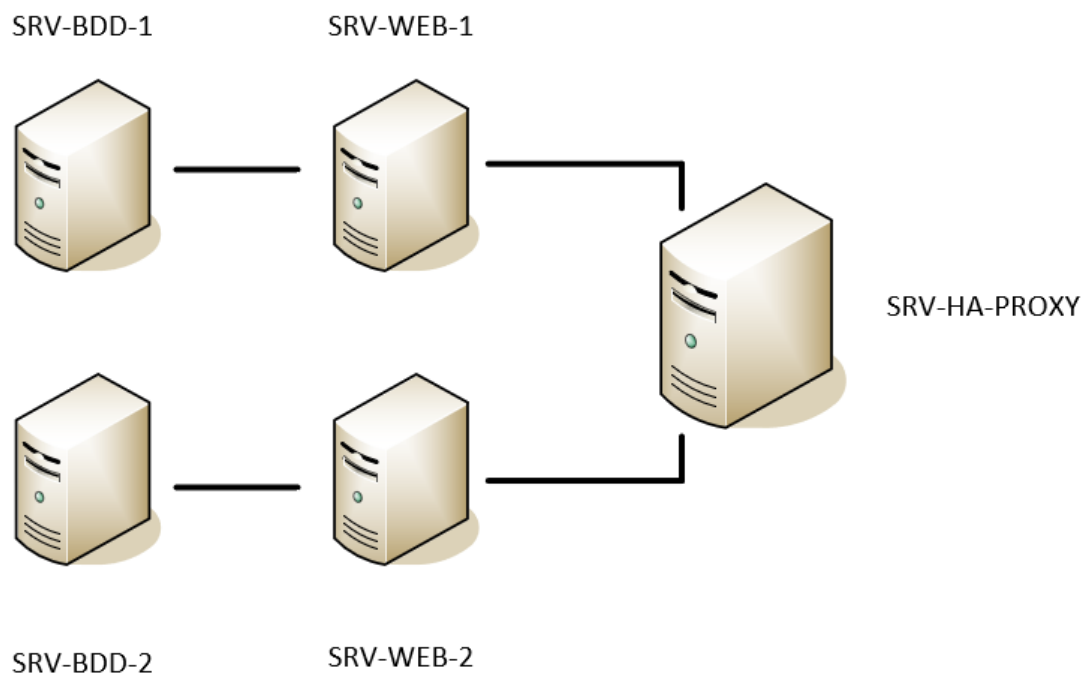


4. Lier les bases de données aux serveurs web

Chaque serveur Web doit être lié à sa propre base distante : WebSrv1 à DB1, WebSrv2 à DB2. Les deux bases auront exactement la même structure et le même nom.

 Voir la documentation : Lier une base de données à un site web PHP

- Créez une base de données `INFRACT_RDV` sur DB1 et DB2
- Créez un utilisateur MariaDB ayant les droits nécessaires
- Configurez la connexion distante (bind-address, firewall, etc.)
- Dans chaque fichier PHP, modifiez l'IP cible de la base distante

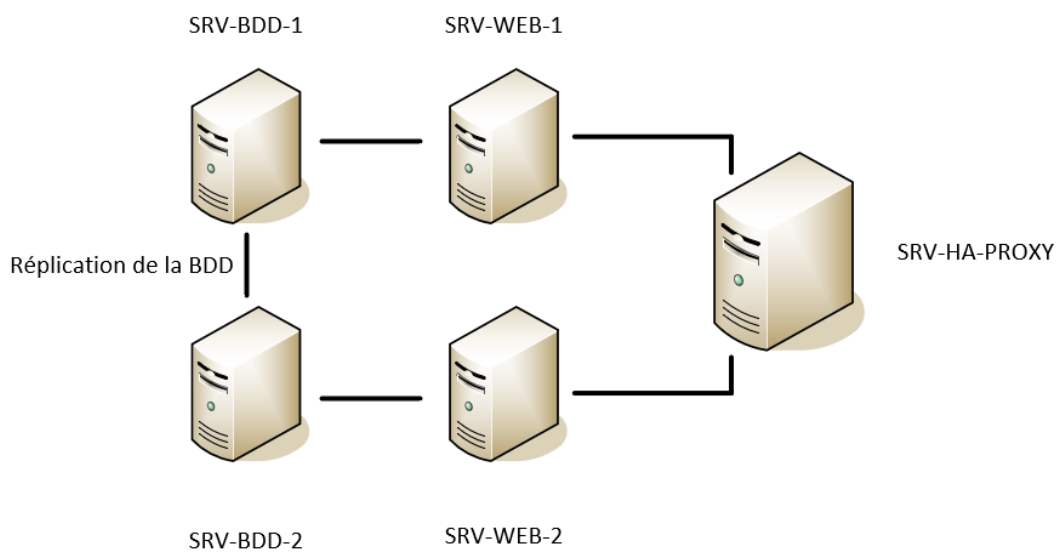


5. Mettre en place la réplication MariaDB

La réplication bidirectionnelle entre DB1 et DB2 permet de garder les données synchronisées. C'est ce qui garantit que les rendez-vous enregistrés sur un serveur soient visibles depuis l'autre.

 **Voir la documentation : *Réplication Master ↔ Master MariaDB***

- Modifiez les fichiers de configuration MariaDB sur DB1 et DB2
- Créez les utilisateurs de réplication
- Initialisez la réplication en croisant les deux bases
- Vérifiez avec les commandes `SHOW SLAVE STATUS\G` que tout fonctionne
- Mettez en place `auto_increment_offset` pour éviter les doublons



📅

Réservation de rendez-vous

Réserver

📅

Rendez-vous enregistrés

THOMAS DELZOR — 2025-04-04 à 14:00:00

🗑️

```

MariaDB [INFRAC_T_RDV]> SELECT * FROM rdv;
+----+-----+-----+-----+
| id | nom      | date      | heure |
+----+-----+-----+-----+
| 21 | THOMAS DELZOR | 2025-04-04 | 14:00:00 |
+----+-----+-----+-----+
  
```

✓ 6. Conclusion

✓ Ce que nous avons mis en place :

- Un site PHP accessible en haute disponibilité via HAProxy
- Deux serveurs web
- Deux bases de données MariaDB répliquées en temps réel
- Une séparation des rôles claire entre web, données et répartition

 **Pour reproduire le projet avec les commandes techniques :**

 Consultez la

documentation technique complète
