Red Hat Enterprise Linux 9

Gestion des services d'infrastructure de réseau

Guide de gestion des services d'infrastructure réseau dans Red Hat Enterprise Linux 9

Last Updated: 2024-01-18

Red Hat Enterprise Linux 9 Gestion des services d'infrastructure de réseau

Guide de gestion des services d'infrastructure réseau dans Red Hat Enterprise Linux 9

Notice légale

Résumé

Ce document décrit comment configurer et gérer les services d'infrastructure réseau de base, tels que DNS et DHCP, sur Red Hat Enterprise Linux 9.

Table des matières

RENDRE L'OPEN SOURCE PLUS INCLUSIF

Red Hat s'engage à remplacer les termes problématiques dans son code, sa documentation et ses propriétés Web. Nous commençons par ces quatre termes : master, slave, blacklist et whitelist. En raison de l'ampleur de cette entreprise, ces changements seront mis en œuvre progressivement au cours de plusieurs versions à venir. Pour plus de détails, voir le message de notre directeur technique Chris Wright.

FOURNIR UN RETOUR D'INFORMATION SUR LA DOCUMENTATION DE RED HAT

Nous apprécions vos commentaires sur notre documentation. Faites-nous savoir comment nous pouvons l'améliorer.

Soumettre des commentaires sur des passages spécifiques

1. Consultez la documentation au format **Multi-page HTML** et assurez-vous que le bouton **Feedback** apparaît dans le coin supérieur droit après le chargement complet de la page.
2. Utilisez votre curseur pour mettre en évidence la partie du texte que vous souhaitez commenter.
3. Cliquez sur le bouton **Add Feedback** qui apparaît près du texte en surbrillance.
4. Ajoutez vos commentaires et cliquez sur **Submit**.

Soumettre des commentaires via Bugzilla (compte requis)

1. Connectez-vous au site Web de Bugzilla.
2. Sélectionnez la version correcte dans le menu **Version**.
3. Saisissez un titre descriptif dans le champ **Summary**.
4. Saisissez votre suggestion d'amélioration dans le champ **Description**. Incluez des liens vers les parties pertinentes de la documentation.
5. Cliquez sur **Submit Bug**.

CHAPITRE 1. MISE EN PLACE ET CONFIGURATION D'UN SERVEUR DNS BIND

BIND est un serveur DNS riche en fonctionnalités et entièrement conforme aux normes et projets de normes DNS de l'IETF (Internet Engineering Task Force). Par exemple, les administrateurs utilisent fréquemment BIND comme :

- Serveur DNS en cache dans le réseau local
- Serveur DNS autoritaire pour les zones
- Serveur secondaire pour assurer la haute disponibilité des zones

1.1. CONSIDÉRATIONS SUR LA PROTECTION DE BIND AVEC SELINUX OU SON UTILISATION DANS UN ENVIRONNEMENT CHANGE-ROOT

Pour sécuriser une installation BIND, vous pouvez :

- Exécutez le service **named** sans environnement change-root. Dans ce cas, SELinux en mode **enforcing** empêche l'exploitation des vulnérabilités de sécurité connues de BIND. Par défaut, Red Hat Enterprise Linux utilise SELinux en mode **enforcing**.

IMPORTANT

L'exécution de BIND sur RHEL avec SELinux en mode **enforcing** est plus sûre que l'exécution de BIND dans un environnement change-root.

- Exécutez le service **named-chroot** dans un environnement change-root.
  En utilisant la fonctionnalité change-root, les administrateurs peuvent définir que le répertoire racine d'un processus et de ses sous-processus est différent du répertoire /. Lorsque vous démarrez le service **named-chroot**, BIND change son répertoire racine en **/var/named/chroot/**. Par conséquent, le service utilise les commandes **mount --bind** pour rendre les fichiers et les répertoires répertoriés dans **/etc/named-chroot.files** disponibles dans **/var/named/chroot/**, et le processus n'a pas accès aux fichiers situés en dehors de **/var/named/chroot/**.

Si vous décidez d'utiliser BIND :

- En mode normal, utilisez le service **named**.
- Dans un environnement change-root, utilisez le service **named-chroot**. Pour ce faire, vous devez installer en plus le paquetage **named-chroot**.

1.2. CONFIGURATION DE BIND EN TANT QUE SERVEUR DNS DE MISE EN CACHE

Par défaut, le serveur DNS BIND résout et met en cache les recherches réussies et celles qui ont échoué. Le service répond ensuite aux requêtes portant sur les mêmes enregistrements à partir de son cache. Cela permet d'améliorer considérablement la vitesse des recherches DNS.

Conditions préalables

- L'adresse IP du serveur est statique.

Procédure

1. Installez les paquets **bind** et **bind-utils**:

```
# dnf install bind bind-utils
```

2. Si vous souhaitez exécuter BIND dans un environnement change-root, installez le paquetage **bind-chroot**:

```
# dnf install bind-chroot
```

Notez que l'exécution de BIND sur un hôte avec SELinux en mode **enforcing**, qui est le mode par défaut, est plus sûre.

3. Modifiez le fichier **/etc/named.conf** et apportez les modifications suivantes à l'instruction **options**:

   a. Mettez à jour les instructions **listen-on** et **listen-on-v6** pour spécifier les interfaces IPv4 et IPv6 sur lesquelles BIND doit écouter :

```
listen-on port 53 { 127.0.0.1; 192.0.2.1; };
listen-on-v6 port 53 { ::1; 2001:db8:1::1; };
```

   b. Mettez à jour l'instruction **allow-query** pour configurer les adresses et plages IP à partir desquelles les clients peuvent interroger ce serveur DNS :

```
allow-query { localhost; 192.0.2.0/24; 2001:db8:1::/64; };
```

   c. Ajoutez une déclaration **allow-recursion** pour définir à partir de quelles adresses IP et de quelles plages BIND accepte les requêtes récursives :

```
allow-recursion { localhost; 192.0.2.0/24; 2001:db8:1::/64; };
```

AVERTISSEMENT

N'autorisez pas la récursivité sur les adresses IP publiques du serveur. Dans le cas contraire, le serveur peut être impliqué dans des attaques d'amplification DNS à grande échelle.

   d. Par défaut, BIND résout les requêtes en interrogeant de manière récursive un serveur DNS faisant autorité à partir des serveurs racine. Vous pouvez également configurer BIND pour qu'il transmette les requêtes à d'autres serveurs DNS, tels que ceux de votre fournisseur d'accès. Dans ce cas, ajoutez une déclaration **forwarders** avec la liste des adresses IP des serveurs DNS vers lesquels BIND doit transférer les requêtes :

```
forwarders { 198.51.100.1; 203.0.113.5; };
```

En guise de solution de repli, BIND résout les requêtes de manière récursive si les serveurs de transfert ne répondent pas. Pour désactiver ce comportement, ajoutez une déclaration **forward only;**.

4. Vérifier la syntaxe du fichier **/etc/named.conf**:

```
# named-checkconf
```

Si la commande n'affiche aucune sortie, la syntaxe est correcte.

5. Mettez à jour les règles **firewalld** pour autoriser le trafic DNS entrant :

```
# firewall-cmd --permanent --add-service=dns
# firewall-cmd --reload
```

6. Démarrer et activer BIND :

```
# systemctl enable --now named
```

Si vous souhaitez exécuter BIND dans un environnement change-root, utilisez la commande **systemctl enable --now named-chroot** pour activer et démarrer le service.

Vérification

1. Utiliser le serveur DNS nouvellement configuré pour résoudre un domaine :

```
# dig @localhost www.example.org
...
www.example.org.    86400    IN    A    198.51.100.34

;; Query time: 917 msec
...
```

Cet exemple suppose que BIND fonctionne sur le même hôte et répond aux requêtes sur l'interface **localhost**.

Après avoir interrogé un enregistrement pour la première fois, BIND ajoute l'entrée à son cache.

2. Répétez la requête précédente :

```
# dig @localhost www.example.org
...
www.example.org.    85332    IN    A    198.51.100.34

;; Query time: 1 msec
...
```

Grâce à l'entrée mise en cache, les requêtes ultérieures pour le même enregistrement sont nettement plus rapides jusqu'à l'expiration de l'entrée.

Prochaines étapes

- Configurez les clients de votre réseau pour qu'ils utilisent ce serveur DNS. Si un serveur DHCP fournit les paramètres du serveur DNS aux clients, mettez à jour la configuration du serveur DHCP en conséquence.

Ressources supplémentaires

- Considérations sur la protection de BIND avec SELinux ou son utilisation dans un environnement change-root
- **named.conf(5)** page de manuel
- **/usr/share/doc/bind/sample/etc/named.conf**

1.3. CONFIGURATION DE LA JOURNALISATION SUR UN SERVEUR DNS BIND

La configuration du fichier par défaut **/etc/named.conf**, telle que fournie par le paquetage **bind**, utilise le canal **default_debug** et enregistre les messages dans le fichier **/var/named/data/named.run**. Le canal **default_debug** n'enregistre les entrées que lorsque le niveau de débogage du serveur est différent de zéro.

En utilisant différents canaux et catégories, vous pouvez configurer BIND pour qu'il écrive dans des fichiers distincts différents événements ayant une gravité définie.

Conditions préalables

- BIND est déjà configuré, par exemple, en tant que serveur de noms avec mise en cache.
- Le service **named** ou **named-chroot** est en cours d'exécution.

Procédure

1. Modifiez le fichier **/etc/named.conf** et ajoutez les phrases **category** et **channel** à la déclaration **logging**, par exemple :

```
logging {
    ...

    category notify { zone_transfer_log; };
    category xfer-in { zone_transfer_log; };
    category xfer-out { zone_transfer_log; };
    channel zone_transfer_log {
        file "/var/named/log/transfer.log" versions 10 size 50m;
        print-time yes;
        print-category yes;
        print-severity yes;
        severity info;
    };

    ...
};
```

Avec cet exemple de configuration, BIND enregistre les messages relatifs aux transferts de zone sur **/var/named/log/transfer.log**. BIND crée jusqu'à **10** versions du fichier journal et les fait tourner s'ils atteignent une taille maximale de **50** MB.

La phrase **category** définit les canaux auxquels BIND envoie les messages d'une catégorie.

La phrase **channel** définit la destination des messages de log, y compris le nombre de versions,

la taille maximale du fichier et le niveau de sévérité que BIND doit enregistrer dans un canal. Des paramètres supplémentaires, tels que l'activation de l'enregistrement de l'horodatage, de la catégorie et de la gravité d'un événement, sont facultatifs, mais utiles à des fins de débogage.

2. Créez le répertoire log s'il n'existe pas, et accordez des droits d'écriture à l'utilisateur **named** sur ce répertoire :

```
# mkdir /var/named/log/
# chown named:named /var/named/log/
# chmod 700 /var/named/log/
```

3. Vérifier la syntaxe du fichier **/etc/named.conf**:

```
# named-checkconf
```

Si la commande n'affiche aucune sortie, la syntaxe est correcte.

4. Redémarrer BIND :

```
# systemctl restart named
```

Si vous exécutez BIND dans un environnement change-root, utilisez la commande **systemctl restart named-chroot** pour redémarrer le service.

Vérification

- Affiche le contenu du fichier journal :

```
# cat /var/named/log/transfer.log
...
06-Jul-2022 15:08:51.261 xfer-out: info: client @0x7fecbc0b0700 192.0.2.2#36121/key
example-transfer-key (example.com): transfer of 'example.com/IN': AXFR started: TSIG
example-transfer-key (serial 2022070603)
06-Jul-2022 15:08:51.261 xfer-out: info: client @0x7fecbc0b0700 192.0.2.2#36121/key
example-transfer-key (example.com): transfer of 'example.com/IN': AXFR ended
```

Ressources supplémentaires

- **named.conf(5)** page de manuel

1.4. ÉCRITURE D'ACLS BIND

Le contrôle de l'accès à certaines fonctionnalités de BIND peut empêcher les accès non autorisés et les attaques, telles que les dénis de service (DoS). Les listes de contrôle d'accès (**acl**) de BIND sont des listes d'adresses IP et de plages d'adresses. Chaque ACL possède un surnom que vous pouvez utiliser dans plusieurs instructions, comme **allow-query**, pour faire référence aux adresses IP et aux plages spécifiées.

AVERTISSEMENT

BIND n'utilise que la première entrée correspondante d'une ACL. Par exemple, si vous définissez une ACL **{ 192.0.2/24; !192.0.2.1; }** et que l'hôte ayant l'adresse IP **192.0.2.1** se connecte, l'accès est accordé même si la deuxième entrée exclut cette adresse.

BIND dispose des listes de contrôle d'accès (ACL) intégrées suivantes :

- **none**: Ne correspond à aucun hôte.
- **any**: Correspond à tous les hôtes.
- **localhost**: Correspond aux adresses de loopback **127.0.0.1** et **::1**, ainsi qu'aux adresses IP de toutes les interfaces du serveur qui exécute BIND.
- **localnets**: Correspond aux adresses de bouclage **127.0.0.1** et **::1**, ainsi qu'à tous les sous-réseaux auxquels le serveur qui exécute BIND est directement connecté.

Conditions préalables

- BIND est déjà configuré, par exemple, en tant que serveur de noms avec mise en cache.
- Le service **named** ou **named-chroot** est en cours d'exécution.

Procédure

1. Modifiez le fichier **/etc/named.conf** et effectuez les changements suivants :
   a. Ajoutez les déclarations **acl** au fichier. Par exemple, pour créer une ACL nommée **internal-networks** pour **127.0.0.1**, **192.0.2.0/24**, et **2001:db8:1::/64**, entrez :

```
acl internal-networks { 127.0.0.1; 192.0.2.0/24; 2001:db8:1::/64; };
acl dmz-networks { 198.51.100.0/24; 2001:db8:2::/64; };
```

   b. Utilisez le surnom de l'ACL dans les déclarations qui les prennent en charge, par exemple :

```
allow-query { internal-networks; dmz-networks; };
allow-recursion { internal-networks; };
```

2. Vérifier la syntaxe du fichier **/etc/named.conf**:

```
# named-checkconf
```

   Si la commande n'affiche aucune sortie, la syntaxe est correcte.

3. Recharger BIND :

```
# systemctl reload named
```

Si vous exécutez BIND dans un environnement change-root, utilisez la commande **systemctl reload named-chroot** pour recharger le service.

Vérification

- Exécuter une action qui déclenche une fonction utilisant la liste de contrôle d'accès configurée. Par exemple, l'ACL de cette procédure n'autorise que les requêtes récursives à partir des adresses IP définies. Dans ce cas, entrez la commande suivante sur un hôte qui ne fait pas partie de la définition de l'ACL pour tenter de résoudre un domaine externe :

```
# dig short @192.0.2.1 www.example.com
```

Si la commande ne renvoie aucun résultat, BIND a refusé l'accès et l'ACL fonctionne. Pour obtenir une sortie verbeuse sur le client, utilisez la commande sans l'option **short**:

```
# dig @192.0.2.1 www.example.com
...
;; WARNING: recursion requested but not available
...
```

1.5. CONFIGURATION DES ZONES SUR UN SERVEUR DNS BIND

Une zone DNS est une base de données contenant des enregistrements de ressources pour un sous-arbre spécifique de l'espace de domaine. Par exemple, si vous êtes responsable du domaine **example.com**, vous pouvez configurer une zone pour celui-ci dans BIND. Par conséquent, les clients peuvent résoudre **www.example.com** à l'adresse IP configurée dans cette zone.

1.5.1. L'enregistrement SOA dans les fichiers de zone

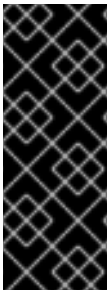
L'enregistrement de début d'autorité (SOA) est un enregistrement obligatoire dans une zone DNS. Cet enregistrement est important, par exemple, si plusieurs serveurs DNS font autorité pour une zone, mais aussi pour les résolveurs DNS.

Un enregistrement SOA dans BIND a la syntaxe suivante :

```
name class type mname rname serial refresh retry expire minimum
```

Pour une meilleure lisibilité, les administrateurs divisent généralement l'enregistrement dans les fichiers de zone en plusieurs lignes avec des commentaires qui commencent par un point-virgule (**;**). Notez que si vous divisez un enregistrement SOA, les parenthèses maintiennent l'enregistrement ensemble :

```
@ IN SOA ns1.example.com. hostmaster.example.com. (
                2022070601 ; serial number
                1d         ; refresh period
                3h         ; retry period
                3d         ; expire time
                3h )       ; minimum TTL
```

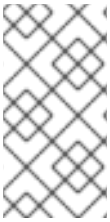


IMPORTANT

Notez le point à la fin des noms de domaine pleinement qualifiés (FQDN). Les FQDN sont constitués de plusieurs étiquettes de domaine, séparées par des points. Comme la racine du DNS a une étiquette vide, les FQDNs se terminent par un point. Par conséquent, BIND ajoute le nom de la zone aux noms sans point final. Un nom d'hôte sans point final, par exemple **ns1.example.com**, sera étendu à **ns1.example.com.example.com.**, ce qui n'est pas l'adresse correcte du serveur de noms primaire.

Il s'agit des champs d'un enregistrement SOA :

- **name**: Le nom de la zone, appelé **origin**. Si vous attribuez la valeur **@** à ce champ, BIND l'étend au nom de la zone défini dans **/etc/named.conf**.
- **class**: Dans les enregistrements SOA, vous devez toujours définir ce champ sur Internet (**IN**).
- **type**: Dans les enregistrements SOA, ce champ doit toujours avoir la valeur **SOA**.
- **mname** (nom principal) : Le nom d'hôte du serveur de noms primaire de cette zone.
- **rname** (nom du responsable) : L'adresse électronique de la personne responsable de cette zone. Notez que le format est différent. Vous devez remplacer le signe at (**@**) par un point (**.**).
- **serial**: Le numéro de version de ce fichier de zone. Les serveurs de noms secondaires ne mettent à jour leurs copies de la zone que si le numéro de série du serveur primaire est plus élevé.
  Le format peut être n'importe quelle valeur numérique. Un format couramment utilisé est **<year><month><day><two-digit-number>**. Avec ce format, vous pouvez théoriquement modifier le fichier de zone jusqu'à cent fois par jour.
- **refresh**: Délai pendant lequel les serveurs secondaires doivent attendre avant de vérifier auprès du serveur primaire si la zone a été mise à jour.
- **retry**: Délai pendant lequel un serveur secondaire tente à nouveau d'interroger le serveur primaire après une tentative infructueuse.
- **expire**: Le temps après lequel un serveur secondaire cesse d'interroger le serveur primaire, si toutes les tentatives précédentes ont échoué.
- **minimum**: La RFC 2308 a modifié la signification de ce champ en le remplaçant par le temps de mise en cache négatif. Les résolveurs conformes l'utilisent pour déterminer la durée de mise en cache des erreurs de noms **NXDOMAIN**.



NOTE

Une valeur numérique dans les champs **refresh**, **retry**, **expire**, et **minimum** définit une heure en secondes. Toutefois, pour une meilleure lisibilité, utilisez des suffixes temporels, tels que **m** pour les minutes, **h** pour les heures et **d** pour les jours. Par exemple, **3h** correspond à 3 heures.

Ressources supplémentaires

- RFC 1035: Noms de domaine - mise en œuvre et spécification
- RFC 1034: Noms de domaine - concepts et facilités

- RFC 2308: Mise en cache négative des requêtes DNS (cache DNS)

1.5.2. Configuration d'une zone de transfert sur un serveur primaire BIND

Les zones de transfert associent les noms aux adresses IP et à d'autres informations. Par exemple, si vous êtes responsable du domaine **example.com**, vous pouvez configurer une zone de transfert dans BIND pour résoudre des noms tels que **www.example.com**.

Conditions préalables

- BIND est déjà configuré, par exemple, en tant que serveur de noms avec mise en cache.
- Le service **named** ou **named-chroot** est en cours d'exécution.

Procédure

1. Ajouter une définition de zone au fichier **/etc/named.conf**:

```
zone "example.com" {
    type master;
    file "example.com.zone";
    allow-query { any; };
    allow-transfer { none; };
};
```

   Ces paramètres définissent

   - Ce serveur est le serveur primaire (**type master**) pour la zone **example.com**.
   - Le fichier **/var/named/example.com.zone** est le fichier de zone. Si vous définissez un chemin d'accès relatif, comme dans cet exemple, ce chemin d'accès est relatif au répertoire que vous avez défini dans **directory** dans la déclaration **options**.
   - Tout hôte peut interroger cette zone. Il est également possible de spécifier des plages d'adresses IP ou des surnoms de liste de contrôle d'accès (ACL) BIND pour limiter l'accès.
   - Aucun hôte ne peut transférer la zone. N'autorisez les transferts de zone que lorsque vous configurez des serveurs secondaires et uniquement pour les adresses IP des serveurs secondaires.

2. Vérifier la syntaxe du fichier **/etc/named.conf**:

```
# named-checkconf
```

   Si la commande n'affiche aucune sortie, la syntaxe est correcte.

3. Créez le fichier **/var/named/example.com.zone**, par exemple, avec le contenu suivant :

```
$TTL 8h
@ IN SOA ns1.example.com. hostmaster.example.com. (
                    2022070601 ; serial number
                    1d         ; refresh period
                    3h         ; retry period
                    3d         ; expire time
                    3h )       ; minimum TTL
```

```
                IN NS   ns1.example.com.
                IN MX   10 mail.example.com.

www             IN A    192.0.2.30
www             IN AAAA 2001:db8:1::30
ns1             IN A    192.0.2.1
ns1             IN AAAA 2001:db8:1::1
mail            IN A    192.0.2.20
mail            IN AAAA 2001:db8:1::20
```

Ce fichier de zone :

- Définit la valeur par défaut du time-to-live (TTL) pour les enregistrements de ressources à 8 heures. Sans suffixe temporel, tel que **h** pour heure, BIND interprète la valeur en secondes.
- Contient l'enregistrement de ressource SOA requis avec des détails sur la zone.
- Définit **ns1.example.com** comme serveur DNS faisant autorité pour cette zone. Pour être fonctionnelle, une zone nécessite au moins un enregistrement de serveur de noms (**NS**). Toutefois, pour être conforme à la RFC 1912, il faut au moins deux serveurs de noms.
- Définit **mail.example.com** comme l'échangeur de courrier (**MX**) du domaine **example.com**. La valeur numérique devant le nom d'hôte est la priorité de l'enregistrement. Les entrées ayant une valeur inférieure ont une priorité plus élevée.
- Définit les adresses IPv4 et IPv6 de **www.example.com**, **mail.example.com**, et **ns1.example.com**.

4. Définissez des autorisations sécurisées sur le fichier de zone qui permettent uniquement au groupe **named** de le lire :

```
# chown root:named /var/named/example.com.zone
# chmod 640 /var/named/example.com.zone
```

5. Vérifier la syntaxe du fichier **/var/named/example.com.zone**:

```
# named-checkzone example.com /var/named/example.com.zone
zone example.com/IN: loaded serial 2022070601
OK
```

6. Recharger BIND :

```
# systemctl reload named
```

Si vous exécutez BIND dans un environnement change-root, utilisez la commande **systemctl reload named-chroot** pour recharger le service.

Vérification

- Interrogez différents enregistrements de la zone **example.com** et vérifiez que le résultat correspond aux enregistrements que vous avez configurés dans le fichier de zone :

```
# dig +short @localhost AAAA www.example.com
```

```
2001:db8:1::30

# dig +short @localhost NS example.com
ns1.example.com.

# dig +short @localhost A ns1.example.com
192.0.2.1
```

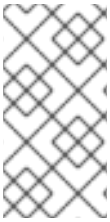
Cet exemple suppose que BIND fonctionne sur le même hôte et répond aux requêtes sur l'interface **localhost**.

Ressources supplémentaires

- L'enregistrement SOA dans les fichiers de zone
- Écriture d'ACLs BIND
- RFC 1912 - Erreurs courantes de fonctionnement et de configuration du DNS

1.5.3. Mise en place d'une zone inverse sur un serveur primaire BIND

Les zones inversées transforment les adresses IP en noms. Par exemple, si vous êtes responsable de la plage d'adresses IP **192.0.2.0/24**, vous pouvez configurer une zone inverse dans BIND pour résoudre les adresses IP de cette plage en noms d'hôtes.



NOTE

Si vous créez une zone inversée pour des réseaux de classe entière, nommez la zone en conséquence. Par exemple, pour le réseau de classe C **192.0.2.0/24**, le nom de la zone est **2.0.192.in-addr.arpa**. Si vous souhaitez créer une zone inverse pour un réseau de taille différente, par exemple **192.0.2.0/28**, le nom de la zone est **28-2.0.192.in-addr.arpa**.

Conditions préalables

- BIND est déjà configuré, par exemple, en tant que serveur de noms avec mise en cache.
- Le service **named** ou **named-chroot** est en cours d'exécution.

Procédure

1. Ajouter une définition de zone au fichier **/etc/named.conf**:

```
zone "2.0.192.in-addr.arpa" {
    type master;
    file "2.0.192.in-addr.arpa.zone";
    allow-query { any; };
    allow-transfer { none; };
};
```

Ces paramètres définissent

- Ce serveur est le serveur primaire (**type master**) pour la zone inversée **2.0.192.in-addr.arpa**.

- Le fichier **/var/named/2.0.192.in-addr.arpa.zone** est le fichier de zone. Si vous définissez un chemin d'accès relatif, comme dans cet exemple, ce chemin d'accès est relatif au répertoire que vous avez défini dans **directory** dans la déclaration **options**.
- Tout hôte peut interroger cette zone. Il est également possible de spécifier des plages d'adresses IP ou des surnoms de liste de contrôle d'accès (ACL) BIND pour limiter l'accès.
- Aucun hôte ne peut transférer la zone. N'autorisez les transferts de zone que lorsque vous configurez des serveurs secondaires et uniquement pour les adresses IP des serveurs secondaires.

2. Vérifier la syntaxe du fichier **/etc/named.conf**:

```
# named-checkconf
```

Si la commande n'affiche aucune sortie, la syntaxe est correcte.

3. Créez le fichier **/var/named/2.0.192.in-addr.arpa.zone**, par exemple, avec le contenu suivant :

```
$TTL 8h
@ IN SOA ns1.example.com. hostmaster.example.com. (
                2022070601 ; serial number
                1d         ; refresh period
                3h         ; retry period
                3d         ; expire time
                3h )       ; minimum TTL

            IN NS   ns1.example.com.

1           IN PTR  ns1.example.com.
30          IN PTR  www.example.com.
```

Ce fichier de zone :

- Définit la valeur par défaut du time-to-live (TTL) pour les enregistrements de ressources à 8 heures. Sans suffixe temporel, tel que **h** pour heure, BIND interprète la valeur en secondes.
- Contient l'enregistrement de ressource SOA requis avec des détails sur la zone.
- Définit **ns1.example.com** comme serveur DNS faisant autorité pour cette zone inversée. Pour être fonctionnelle, une zone nécessite au moins un enregistrement de serveur de noms (**NS**). Toutefois, pour être conforme à la RFC 1912, il faut au moins deux serveurs de noms.
- Définit l'enregistrement du pointeur (**PTR**) pour les adresses **192.0.2.1** et **192.0.2.30**.

4. Définissez des autorisations sécurisées sur le fichier de zone qui n'autorisent que le groupe **named** à le lire :

```
# chown root:named /var/named/2.0.192.in-addr.arpa.zone
# chmod 640 /var/named/2.0.192.in-addr.arpa.zone
```

5. Vérifier la syntaxe du fichier **/var/named/2.0.192.in-addr.arpa.zone**:

```
# named-checkzone 2.0.192.in-addr.arpa /var/named/2.0.192.in-addr.arpa.zone
zone 2.0.192.in-addr.arpa/IN: loaded serial 2022070601
OK
```

6. Recharger BIND :

```
# systemctl reload named
```

Si vous exécutez BIND dans un environnement change-root, utilisez la commande **systemctl reload named-chroot** pour recharger le service.

Vérification

- Interrogez différents enregistrements de la zone inversée et vérifiez que les résultats correspondent aux enregistrements configurés dans le fichier de zone :

```
# dig +short @localhost -x 192.0.2.1
ns1.example.com.

# dig +short @localhost -x 192.0.2.30
www.example.com.
```

Cet exemple suppose que BIND fonctionne sur le même hôte et répond aux requêtes sur l'interface **localhost**.

Ressources supplémentaires

- L'enregistrement SOA dans les fichiers de zone
- Écriture d'ACLs BIND
- RFC 1912 - Erreurs courantes de fonctionnement et de configuration du DNS

1.5.4. Mise à jour d'un fichier de zone BIND

Dans certaines situations, par exemple si l'adresse IP d'un serveur change, vous devez mettre à jour un fichier de zone. Si plusieurs serveurs DNS sont responsables d'une zone, n'effectuez cette procédure que sur le serveur principal. Les autres serveurs DNS qui stockent une copie de la zone recevront la mise à jour par le biais d'un transfert de zone.

Conditions préalables

- La zone est configurée.
- Le service **named** ou **named-chroot** est en cours d'exécution.

Procédure

1. Facultatif : indiquez le chemin d'accès au fichier de zone dans le fichier **/etc/named.conf**:

```
options {
    ...
    directory       "/var/named";
}
```

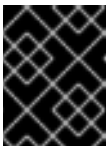
```
zone "example.com" {
    ...
    file "example.com.zone";
};
```

Le chemin d'accès au fichier de la zone est indiqué dans la déclaration **file** de la définition de la zone. Un chemin relatif est relatif au répertoire défini dans **directory** dans la déclaration **options**.

2. Modifier le fichier de zone :

   a. Effectuez les modifications nécessaires.

   b. Incrémenter le numéro de série dans l'enregistrement de début d'autorité (SOA).



IMPORTANT

Si le numéro de série est égal ou inférieur à la valeur précédente, les serveurs secondaires ne mettront pas à jour leur copie de la zone.

3. Vérifier la syntaxe du fichier de zone :

```
# named-checkzone example.com /var/named/example.com.zone
zone example.com/IN: loaded serial 2022062802
OK
```

4. Recharger BIND :

```
# systemctl reload named
```

Si vous exécutez BIND dans un environnement change-root, utilisez la commande **systemctl reload named-chroot** pour recharger le service.

Vérification

- Interroger l'enregistrement que vous avez ajouté, modifié ou supprimé, par exemple :

```
# dig +short @localhost A ns2.example.com
192.0.2.2
```

Cet exemple suppose que BIND fonctionne sur le même hôte et répond aux requêtes sur l'interface **localhost**.

Ressources supplémentaires

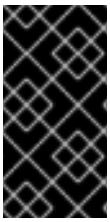
- L'enregistrement SOA dans les fichiers de zone
- Configuration d'une zone de transfert sur un serveur primaire BIND
- Mise en place d'une zone inverse sur un serveur primaire BIND

1.5.5. Signature de la zone DNSSEC à l'aide des fonctions de génération automatique de clés et de maintenance de la zone

Vous pouvez signer les zones avec des extensions de sécurité du système de noms de domaine (DNSSEC) pour garantir l'authentification et l'intégrité des données. Ces zones contiennent des enregistrements de ressources supplémentaires. Les clients peuvent les utiliser pour vérifier l'authenticité des informations de la zone.

Si vous activez la fonctionnalité de politique DNSSEC pour une zone, BIND effectue automatiquement les actions suivantes :

- Crée les clés
- Signes de la zone
- Entretient la zone, y compris la re-signature et le remplacement périodique des clés.



IMPORTANT

Pour permettre aux serveurs DNS externes de vérifier l'authenticité d'une zone, vous devez ajouter la clé publique de la zone à la zone mère. Contactez votre fournisseur de domaine ou votre bureau d'enregistrement pour plus de détails sur la manière de procéder.

Cette procédure utilise la politique DNSSEC intégrée à **default** dans BIND. Cette politique utilise des signatures de clés uniques **ECDSAP256SHA**. Vous pouvez également créer votre propre politique pour utiliser des clés, des algorithmes et des délais personnalisés.

Conditions préalables

- La zone pour laquelle vous souhaitez activer DNSSEC est configurée.
- Le service **named** ou **named-chroot** est en cours d'exécution.
- Le serveur synchronise l'heure avec un serveur de temps. Une heure système précise est importante pour la validation DNSSEC.

Procédure

1. Modifiez le fichier **/etc/named.conf** et ajoutez **dnssec-policy default;** à la zone pour laquelle vous souhaitez activer le DNSSEC :

```
zone "example.com" {
    ...
    dnssec-policy default;
};
```

2. Recharger BIND :

```
# systemctl reload named
```

   Si vous exécutez BIND dans un environnement change-root, utilisez la commande **systemctl reload named-chroot** pour recharger le service.

3. BIND stocke la clé publique dans le fichier **/var/named/K*<zone_name>*. *<algorithm>* *<key_ID>*.key** dans le fichier Ce fichier permet d'afficher la clé publique de la zone dans le format requis par la zone mère :

   - Format d'enregistrement DS :

- Format d enregistrement DS :

```
# dnssec-dsfromkey /var/named/Kexample.com.+013+61141.key
example.com. IN DS 61141 13 2
3E184188CF6D2521EDFDC3F07CFEE8D0195AACBD85E68BAE0620F638B4B1B027
```

- Format DNSKEY :

```
# grep DNSKEY /var/named/Kexample.com.+013+61141.key
example.com. 3600 IN DNSKEY 257 3 13
sjzT3jNEp120aSO4mPEHHSkReHUf7AABNnT8hNRTzD5cKMQSjDJin2I3
5CaKVcWO1pm+HltxUEt+X9dfp8OZkg==
```

4. Demande d'ajout de la clé publique de la zone à la zone parente. Contactez votre fournisseur de domaine ou votre bureau d'enregistrement pour plus de détails sur la manière de procéder.

Vérification

1. Interrogez votre propre serveur DNS pour obtenir un enregistrement de la zone pour laquelle vous avez activé la signature DNSSEC :

```
# dig +dnssec +short @localhost A www.example.com
192.0.2.30
A 13 3 28800 20220718081258 20220705120353 61141 example.com.
e7Cfh6GuOBMAWsgsHSVTPh+JJSOI/Y6zctzIuqIU1JqEgOOAfL/Qz474
M0sgi54m1Kmnr2ANBKJN9uvOs5eXYw==
```

   Cet exemple suppose que BIND fonctionne sur le même hôte et répond aux requêtes sur l'interface **localhost**.

2. Une fois que la clé publique a été ajoutée à la zone parentale et propagée à d'autres serveurs, vérifiez que le serveur active l'indicateur de données authentifiées (**ad**) lors des requêtes adressées à la zone signée :

```
#  dig @localhost example.com +dnssec
...
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1
...
```

Ressources supplémentaires

- Configuration d'une zone de transfert sur un serveur primaire BIND
- Mise en place d'une zone inverse sur un serveur primaire BIND

1.6. CONFIGURATION DES TRANSFERTS DE ZONE ENTRE SERVEURS DNS BIND

Les transferts de zone garantissent que tous les serveurs DNS disposant d'une copie de la zone utilisent des données actualisées.

Conditions préalables

- Sur le futur serveur primaire, la zone pour laquelle vous souhaitez mettre en place des transferts de zone est déjà configurée.
- Sur le futur serveur secondaire, BIND est déjà configuré, par exemple en tant que serveur de noms de cache.
- Sur les deux serveurs, le service **named** ou **named-chroot** est en cours d'exécution.

Procédure

1. Sur le serveur primaire existant :
   a. Créez une clé partagée et ajoutez-la au fichier **/etc/named.conf**:

```
# tsig-keygen example-transfer-key | tee -a /etc/named.conf
key "example-transfer-key" {
        algorithm hmac-sha256;
        secret "q7ANbnyliDMuvWgnKOxMLi313JGcTZB5ydMW5CyUGXQ=";
};
```

   Cette commande affiche la sortie de la commande **tsig-keygen** et l'ajoute automatiquement à **/etc/named.conf**.

   Vous aurez besoin de la sortie de la commande ultérieurement sur le serveur secondaire également.

   b. Modifiez la définition de la zone dans le fichier **/etc/named.conf**:

      i. Dans la déclaration **allow-transfer**, définissez que les serveurs doivent fournir la clé spécifiée dans la déclaration **example-transfer-key** pour transférer une zone :

```
zone "example.com" {
    ...
    allow-transfer { key example-transfer-key; };
};
```

      Il est également possible d'utiliser les surnoms de la liste de contrôle d'accès (ACL) de BIND dans l'instruction **allow-transfer**.

      ii. Par défaut, après la mise à jour d'une zone, BIND notifie tous les serveurs de noms qui ont un enregistrement de serveur de noms (**NS**) dans cette zone. Si vous n'avez pas l'intention d'ajouter un enregistrement **NS** pour le serveur secondaire dans la zone, vous pouvez configurer BIND pour qu'il notifie quand même ce serveur. Pour ce faire, ajoutez la déclaration **also-notify** avec les adresses IP de ce serveur secondaire dans la zone :

```
zone "example.com" {
    ...
    also-notify { 192.0.2.2; 2001:db8:1::2; };
};
```

   c. Vérifier la syntaxe du fichier **/etc/named.conf**:

```
# named-checkconf
```

   Si la commande n'affiche aucune sortie, la syntaxe est correcte.

d. Recharger BIND :

```
# systemctl reload named
```

Si vous exécutez BIND dans un environnement change-root, utilisez la commande **systemctl reload named-chroot** pour recharger le service.

2. Sur le futur serveur secondaire :

a. Modifiez le fichier /**etc**/**named.conf** comme suit :

i. Ajoutez la même définition de clé que sur le serveur primaire :

```
key "example-transfer-key" {
        algorithm hmac-sha256;
        secret "q7ANbnyliDMuvWgnKOxMLi313JGcTZB5ydMW5CyUGXQ=";
};
```

ii. Ajouter la définition de la zone au fichier /**etc**/**named.conf**:

```
zone "example.com" {
   type slave;
   file "slaves/example.com.zone";
   allow-query { any; };
   allow-transfer { none; };
   masters {
     192.0.2.1 key example-transfer-key;
     2001:db8:1::1 key example-transfer-key;
   };
};
```

Ces paramètres sont les suivants :

- Ce serveur est un serveur secondaire (**type slave**) pour la zone **example.com**.
- Le fichier /**var**/**named**/**slaves**/**example.com.zone** est le fichier de zone. Si vous définissez un chemin d'accès relatif, comme dans cet exemple, ce chemin d'accès est relatif au répertoire que vous avez défini dans **directory** dans l'instruction **options**. Pour séparer les fichiers de zone pour lesquels ce serveur est secondaire des fichiers primaires, vous pouvez les stocker, par exemple, dans le répertoire /**var**/**named**/**slaves**/.
- Tout hôte peut interroger cette zone. Vous pouvez également spécifier des plages d'adresses IP ou des pseudonymes ACL pour limiter l'accès.
- Aucun hôte ne peut transférer la zone à partir de ce serveur.
- Les adresses IP du serveur primaire de cette zone sont **192.0.2.1** et **2001:db8:1::2**. Vous pouvez également spécifier des surnoms ACL. Ce serveur secondaire utilisera la clé nommée **example-transfer-key** pour s'authentifier auprès du serveur primaire.

b. Vérifier la syntaxe du fichier /**etc**/**named.conf**:

```
# named-checkconf
```

c. Recharger BIND :

```
# systemctl reload named
```

Si vous exécutez BIND dans un environnement change-root, utilisez la commande **systemctl reload named-chroot** pour recharger le service.

3. Facultatif : Modifiez le fichier de zone sur le serveur primaire et ajoutez un enregistrement **NS** pour le nouveau serveur secondaire.

Vérification

Sur le serveur secondaire :

1. Afficher les écritures du journal **systemd** du service **named**:

```
# journalctl -u named
...
Jul 06 15:08:51 ns2.example.com named[2024]: zone example.com/IN: Transfer started.
Jul 06 15:08:51 ns2.example.com named[2024]: transfer of 'example.com/IN' from
192.0.2.1#53: connected using 192.0.2.2#45803
Jul 06 15:08:51 ns2.example.com named[2024]: zone example.com/IN: transferred serial
2022070101
Jul 06 15:08:51 ns2.example.com named[2024]: transfer of 'example.com/IN' from
192.0.2.1#53: Transfer status: success
Jul 06 15:08:51 ns2.example.com named[2024]: transfer of 'example.com/IN' from
192.0.2.1#53: Transfer completed: 1 messages, 29 records, 2002 bytes, 0.003 secs (667333
bytes/sec)
```

Si vous exécutez BIND dans un environnement change-root, utilisez la commande **journalctl -u named-chroot** pour afficher les entrées du journal.

2. Vérifiez que BIND a créé le fichier de zone :

```
# ls -l /var/named/slaves/
total 4
-rw-r--r--. 1 named named 2736 Jul  6 15:08 example.com.zone
```

Notez que, par défaut, les serveurs secondaires stockent les fichiers de zone dans un format binaire brut.

3. Interroger un enregistrement de la zone transférée à partir du serveur secondaire :

```
# dig +short @192.0.2.2 AAAA www.example.com
2001:db8:1::30
```

Cet exemple suppose que le serveur secondaire que vous avez configuré dans cette procédure écoute sur l'adresse IP **192.0.2.2**.

Ressources supplémentaires

- Configuration d'une zone de transfert sur un serveur primaire BIND
- Mise en place d'une zone inverse sur un serveur primaire BIND
- Écriture d'ACLs BIND

- Mise à jour d'un fichier de zone BIND

1.7. CONFIGURATION DES ZONES DE POLITIQUE DE RÉPONSE DANS BIND POUR REMPLACER LES ENREGISTREMENTS DNS

En utilisant le blocage et le filtrage DNS, les administrateurs peuvent réécrire une réponse DNS pour bloquer l'accès à certains domaines ou hôtes. Dans BIND, les zones de politique de réponse (RPZ) offrent cette fonctionnalité. Vous pouvez configurer différentes actions pour les entrées bloquées, comme renvoyer une erreur **NXDOMAIN** ou ne pas répondre à la requête.

Si vous disposez de plusieurs serveurs DNS dans votre environnement, utilisez cette procédure pour configurer le RPZ sur le serveur principal, puis configurez les transferts de zone pour rendre le RPZ disponible sur vos serveurs secondaires.

Conditions préalables

- BIND est déjà configuré, par exemple, en tant que serveur de noms avec mise en cache.
- Le service **named** ou **named-chroot** est en cours d'exécution.

Procédure

1. Modifiez le fichier **/etc/named.conf** et effectuez les changements suivants :
   a. Ajouter une définition **response-policy** à la déclaration **options**:

```
options {
    ...

    response-policy {
        zone "rpz.local";
    };

    ...
}
```

   Vous pouvez définir un nom personnalisé pour la zone de protection contre les incendies dans la déclaration **zone** sur **response-policy**. Cependant, vous devez utiliser le même nom dans la définition de la zone à l'étape suivante.

   b. Ajoutez une définition **zone** pour la ZPR que vous avez définie à l'étape précédente :

```
zone "rpz.local" {
    type master;
    file "rpz.local";
    allow-query { localhost; 192.0.2.0/24; 2001:db8:1::/64; };
    allow-transfer { none; };
};
```

   Ces paramètres sont les suivants :

   - Ce serveur est le serveur primaire (**type master**) de la ZPR nommée **rpz.local**.

- Le fichier **/var/named/rpz.local** est le fichier de zone. Si vous définissez un chemin d'accès relatif, comme dans cet exemple, ce chemin d'accès est relatif au répertoire que vous avez défini dans **directory** dans la déclaration **options**.
- Tous les hôtes définis dans **allow-query** peuvent interroger cette zone d'accès public. Vous pouvez également spécifier des plages d'adresses IP ou des surnoms de liste de contrôle d'accès (ACL) BIND pour limiter l'accès.
- Aucun hôte ne peut transférer la zone. N'autorisez les transferts de zone que lorsque vous configurez des serveurs secondaires et uniquement pour les adresses IP des serveurs secondaires.

2. Vérifier la syntaxe du fichier **/etc/named.conf**:

```
# named-checkconf
```

Si la commande n'affiche aucune sortie, la syntaxe est correcte.

3. Créez le fichier **/var/named/rpz.local**, par exemple, avec le contenu suivant :

```
$TTL 10m
@ IN SOA ns1.example.com. hostmaster.example.com. (
                2022070601 ; serial number
                1h         ; refresh period
                1m         ; retry period
                3d         ; expire time
                1m )       ; minimum TTL

          IN NS    ns1.example.com.

example.org     IN CNAME .
*.example.org   IN CNAME .
example.net     IN CNAME rpz-drop.
*.example.net   IN CNAME rpz-drop.
```

Ce fichier de zone :

- Définit la valeur par défaut du time-to-live (TTL) pour les enregistrements de ressources à 10 minutes. Sans suffixe temporel, tel que **h** pour heure, BIND interprète la valeur en secondes.
- Contient l'enregistrement de ressource de début d'autorité (SOA) requis avec des détails sur la zone.
- Définit **ns1.example.com** comme serveur DNS faisant autorité pour cette zone. Pour être fonctionnelle, une zone nécessite au moins un enregistrement de serveur de noms (**NS**). Toutefois, pour être conforme à la RFC 1912, il faut au moins deux serveurs de noms.
- Renvoyer une erreur **NXDOMAIN** pour les requêtes adressées à **example.org** et aux hôtes de ce domaine.
- Abandonner les requêtes vers **example.net** et les hôtes de ce domaine.

Pour une liste complète d'actions et d'exemples, voir le projet de l'IETF : DNS Response Policy Zones (RPZ).

4. Vérifier la syntaxe du fichier **/var/named/rpz.local**:

```
# named-checkzone rpz.local /var/named/rpz.local
zone rpz.local/IN: loaded serial 2022070601
OK
```

5. Recharger BIND :

```
# systemctl reload named
```

Si vous exécutez BIND dans un environnement change-root, utilisez la commande **systemctl reload named-chroot** pour recharger le service.

Vérification

1. Tentative de résolution d'un hôte dans **example.org**, qui est configuré dans le RPZ pour renvoyer une erreur **NXDOMAIN**:

```
# dig @localhost www.example.org
...
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 30286
...
```

Cet exemple suppose que BIND fonctionne sur le même hôte et répond aux requêtes sur l'interface **localhost**.

2. Tentative de résolution d'un hôte dans le domaine **example.net**, qui est configuré dans le RPZ pour rejeter les requêtes :

```
# dig @localhost www.example.net
...
;; connection timed out; no servers could be reached
...
```

Ressources supplémentaires

- Projet de l'IETF : Zones de politique de réponse DNS (RPZ)

CHAPITRE 2. MISE EN PLACE D'UN SERVEUR DNS NON LIÉ

Le serveur DNS **unbound** est un résolveur DNS validant, récursif et de mise en cache. En outre, **unbound** met l'accent sur la sécurité et a, par exemple, activé par défaut les extensions de sécurité du système de noms de domaine (DNSSEC).

2.1. CONFIGURATION D'UNBOUND EN TANT QUE SERVEUR DNS DE MISE EN CACHE

Par défaut, le service DNS **unbound** résout et met en cache les recherches réussies et celles qui ont échoué. Le service répond ensuite aux requêtes portant sur les mêmes enregistrements à partir de son cache.

Procédure

1. Installez le paquetage **unbound**:

```
# dnf install unbound
```

2. Modifiez le fichier **/etc/unbound/unbound.conf** et apportez les modifications suivantes à la clause **server**:

   a. Ajoutez les paramètres **interface** pour configurer les adresses IP sur lesquelles le service **unbound** écoute les requêtes, par exemple :

```
interface: 127.0.0.1
interface: 192.0.2.1
interface: 2001:db8:1::1
```

   Avec ces paramètres, **unbound** n'écoute que les adresses IPv4 et IPv6 spécifiées.

   Le fait de limiter les interfaces aux interfaces requises empêche les clients de réseaux non autorisés, tels que l'internet, d'envoyer des requêtes à ce serveur DNS.

   b. Ajoutez les paramètres **access-control** pour configurer les sous-réseaux à partir desquels les clients peuvent interroger le service DNS, par exemple :

```
access-control: 127.0.0.0/8 allow
access-control: 192.0.2.0/24 allow
access-control: 2001:db8:1::/64 allow
```

3. Créer des clés privées et des certificats pour gérer à distance le service **unbound**:

```
# systemctl restart unbound-keygen
```

   Si vous sautez cette étape, la vérification de la configuration à l'étape suivante signalera les fichiers manquants. Cependant, le service **unbound** crée automatiquement les fichiers s'ils sont manquants.

4. Vérifier le fichier de configuration :

```
# unbound-checkconf
unbound-checkconf: no errors in /etc/unbound/unbound.conf
```

5. Mettez à jour les règles firewalld pour autoriser le trafic DNS entrant :

```
# firewall-cmd --permanent --add-service=dns
# firewall-cmd --reload
```

6. Activez et démarrez le service **unbound**:

```
# systemctl enable --now unbound
```

Vérification

1. Demander au serveur DNS **unbound** écoutant sur l'interface **localhost** de résoudre un domaine :

```
# dig @localhost www.example.com
...
www.example.com.    86400    IN    A    198.51.100.34

;; Query time: 330 msec
...
```

Après avoir interrogé un enregistrement pour la première fois, **unbound** ajoute l'entrée à son cache.

2. Répétez la requête précédente :

```
# dig @localhost www.example.com
...
www.example.com.    85332    IN    A    198.51.100.34

;; Query time: 1 msec
...
```

Grâce à l'entrée mise en cache, les requêtes ultérieures pour le même enregistrement sont nettement plus rapides jusqu'à l'expiration de l'entrée.

Prochaines étapes

- Configurez les clients de votre réseau pour qu'ils utilisent ce serveur DNS. Par exemple, utilisez l'utilitaire **nmcli** pour définir l'IP du serveur DNS dans un profil de connexion NetworkManager :

```
# nmcli connection modify Example_Connection ipv4.dns 192.0.2.1
# nmcli connection modify Example_Connection ipv6.dns 2001:db8:1::1
```

Ressources supplémentaires

- **unbound.conf(5)** page de manuel

CHAPITRE 3. FOURNIR DES SERVICES DHCP

Le protocole de configuration dynamique des hôtes (DHCP) est un protocole réseau qui attribue automatiquement des informations IP aux clients. Vous pouvez configurer le service **dhcpd** pour qu'il fournisse un serveur DHCP et un relais DHCP dans votre réseau.

3.1. LA DIFFÉRENCE ENTRE L'ADRESSAGE IP STATIQUE ET DYNAMIQUE

Adressage IP statique

Lorsque vous attribuez une adresse IP statique à un appareil, cette adresse ne change pas au fil du temps, sauf si vous la modifiez manuellement. Utilisez l'adressage IP statique si vous le souhaitez :

- Pour assurer la cohérence des adresses réseau des serveurs tels que les serveurs DNS et les serveurs d'authentification.
- Pour utiliser des dispositifs de gestion hors bande qui fonctionnent indépendamment de l'infrastructure du réseau.

Adressage IP dynamique

Lorsque vous configurez un périphérique pour qu'il utilise une adresse IP dynamique, celle-ci peut changer au fil du temps. C'est pourquoi les adresses dynamiques sont généralement utilisées pour les périphériques qui se connectent occasionnellement au réseau, car l'adresse IP peut être différente après le redémarrage de l'hôte.
Les adresses IP dynamiques sont plus souples, plus faciles à configurer et à administrer. Le protocole DHCP (Dynamic Host Control Protocol) est une méthode traditionnelle d'attribution dynamique de configurations réseau aux hôtes.

NOTE

Il n'existe pas de règle stricte définissant quand utiliser des adresses IP statiques ou dynamiques. Cela dépend des besoins de l'utilisateur, de ses préférences et de l'environnement du réseau.

3.2. PHASES DE LA TRANSACTION DHCP

Le DHCP fonctionne en quatre phases : Découverte, Offre, Demande, Accusé de réception, également appelé processus DORA. Le DHCP utilise ce processus pour fournir des adresses IP aux clients.

Découverte

Le client DHCP envoie un message pour découvrir le serveur DHCP dans le réseau. Ce message est diffusé au niveau du réseau et de la liaison de données.

Offre

Le serveur DHCP reçoit les messages du client et lui offre une adresse IP. Ce message est unicast au niveau de la couche liaison de données, mais broadcast au niveau de la couche réseau.

Demande

Le client DHCP demande au serveur DHCP l'adresse IP offerte. Ce message est unicast au niveau de la couche liaison de données, mais broadcast au niveau de la couche réseau.

Remerciements

Le serveur DHCP envoie un accusé de réception au client DHCP. Ce message est unicast au niveau de la couche liaison de données, mais broadcast au niveau de la couche réseau. Il s'agit du message final du processus DHCP DORA.

3.3. LES DIFFÉRENCES ENTRE L'UTILISATION DE DHCPD POUR DHCPV4 ET DHCPV6

Le service **dhcpd** permet de fournir les protocoles DHCPv4 et DHCPv6 sur un même serveur. Cependant, vous avez besoin d'une instance distincte de **dhcpd** avec des fichiers de configuration distincts pour fournir le DHCP pour chaque protocole.

DHCPv4

- Fichier de configuration : **/etc/dhcp/dhcpd.conf**
- Nom du service Systemd : **dhcpd**

DHCPv6

- Fichier de configuration : **/etc/dhcp/dhcpd6.conf**
- Nom du service Systemd : **dhcpd6**

3.4. LA BASE DE DONNÉES DES BAUX DU SERVICE DHCPD

Un bail DHCP est la période pendant laquelle le service **dhcpd** attribue une adresse réseau à un client. Le service **dhcpd** stocke les baux DHCP dans les bases de données suivantes :

- Pour DHCPv4 : **/var/lib/dhcpd/dhcpd.leases**
- Pour DHCPv6 : **/var/lib/dhcpd/dhcpd6.leases**

AVERTISSEMENT

La mise à jour manuelle des fichiers de base de données peut corrompre les bases de données.

Les bases de données de baux contiennent des informations sur les baux attribués, telles que l'adresse IP attribuée à une adresse MAC (Media Access Control) ou l'heure d'expiration du bail. Notez que tous les horodatages des bases de données de baux sont exprimés en temps universel coordonné (UTC).

Le service **dhcpd** recrée périodiquement les bases de données :

1. Le service renomme les fichiers existants :
   - **/var/lib/dhcpd/dhcpd.leases** à **/var/lib/dhcpd/dhcpd.leases~**
   - **/var/lib/dhcpd/dhcpd6.leases** à **/var/lib/dhcpd/dhcpd6.leases~**

2. Le service écrit tous les baux connus dans les fichiers **/var/lib/dhcpd/dhcpd.leases** et **/var/lib/dhcpd/dhcpd6.leases** nouvellement créés.

Ressources supplémentaires

- **dhcpd.leases(5)** page de manuel
- Restauration d'une base de données de baux corrompue

3.5. COMPARAISON ENTRE DHCPV6 ET RADVD

Dans un réseau IPv6, seuls les messages d'annonce de routeur fournissent des informations sur la passerelle par défaut IPv6. Par conséquent, si vous souhaitez utiliser DHCPv6 dans des sous-réseaux qui nécessitent un réglage de la passerelle par défaut, vous devez également configurer un service d'annonce de routeur, tel que Router Advertisement Daemon (**radvd**).

Le service **radvd** utilise des drapeaux dans les paquets d'annonce de routeur pour annoncer la disponibilité d'un serveur DHCPv6.

Le tableau suivant compare les caractéristiques du DHCPv6 et de **radvd**:

	DHCPv6	radvd
Fournit des informations sur la passerelle par défaut	non	yes
Garantit des adresses aléatoires pour protéger la vie privée	yes	non
Envoi d'autres options de configuration du réseau	yes	non
Mappage des adresses de contrôle d'accès au support (MAC) en adresses IPv6	yes	non

3.6. CONFIGURATION DU SERVICE RADVD POUR LES ROUTEURS IPV6

Le démon d'annonce de routeur (**radvd**) envoie des messages d'annonce de routeur qui sont nécessaires à l'autoconfiguration sans état d'IPv6. Cela permet aux utilisateurs de configurer automatiquement leurs adresses, paramètres et itinéraires, et de choisir un routeur par défaut sur la base de ces annonces.

NOTE

Vous ne pouvez définir des préfixes **/64** que dans le service **radvd**. Pour utiliser d'autres préfixes, utilisez DHCPv6.

Conditions préalables

- Vous êtes connecté en tant qu'utilisateur **root**.

Procédure

1. Installez le paquetage **radvd**:

```
# dnf install radvd
```

2. Modifiez le fichier **/etc/radvd.conf** et ajoutez la configuration suivante :

```
interface enp1s0
{
  AdvSendAdvert on;
  AdvManagedFlag on;
  AdvOtherConfigFlag on;

  prefix 2001:db8:0:1::/64 {
  };
};
```

   Ces paramètres configurent **radvd** pour qu'il envoie des messages d'annonce de routeur sur le périphérique **enp1s0** pour le sous-réseau **2001:db8:0:1::/64**. Le paramètre **AdvManagedFlag on** définit que le client doit recevoir l'adresse IP d'un serveur DHCP, et le paramètre **AdvOtherConfigFlag** défini sur **on** définit que les clients doivent également recevoir des informations non liées à l'adresse du serveur DHCP.

3. En option, configurez **radvd** pour qu'il démarre automatiquement au démarrage du système :

```
# systemctl enable radvd
```

4. Démarrez le service **radvd**:

```
# systemctl start radvd
```

5. En option, afficher le contenu des paquets d'annonces du routeur et les valeurs configurées que **radvd** envoie :

```
# radvdump
```

Ressources supplémentaires

- **radvd.conf(5)** page de manuel
- **/usr/share/doc/radvd/radvd.conf.example** fichier
- Puis-je utiliser une longueur de préfixe autre que 64 bits dans les annonces de routeur IPv6 ?

3.7. CONFIGURATION DES INTERFACES RÉSEAU POUR LES SERVEURS DHCP

Par défaut, le service **dhcpd** traite les requêtes uniquement sur les interfaces réseau qui ont une adresse IP dans le sous-réseau défini dans le fichier de configuration du service.

Par exemple, dans le scénario suivant, **dhcpd** n'écoute que l'interface réseau **enp0s1**:

- Vous n'avez qu'une définition **subnet** pour le réseau 192.0.2.0/24 dans le fichier **/etc/dhcp/dhcpd.conf**.

- L'interface réseau **enp0s1** est connectée au sous-réseau 192.0.2.0/24.
- L'interface **enp7s0** est connectée à un sous-réseau différent.

Ne suivez cette procédure que si le serveur DHCP contient plusieurs interfaces réseau connectées au même réseau, mais que le service ne doit écouter que sur des interfaces spécifiques.

Selon que vous souhaitez fournir le DHCP pour IPv4, IPv6 ou les deux protocoles, voir la procédure pour :

- Réseaux IPv4
- Réseaux IPv6

Conditions préalables

- Vous êtes connecté en tant qu'utilisateur **root**.
- Le paquet **dhcp-server** est installé.

Procédure

- Pour les réseaux IPv4 :
  1. Copiez le fichier **/usr/lib/systemd/system/dhcpd.service** dans le répertoire **/etc/systemd/system/**:

     ```
     # cp /usr/lib/systemd/system/dhcpd.service /etc/systemd/system/
     ```

     Ne modifiez pas le fichier **/usr/lib/systemd/system/dhcpd.service**. Les futures mises à jour du paquet **dhcp-server** peuvent annuler les modifications.
  2. Modifiez le fichier **/etc/systemd/system/dhcpd.service** et ajoutez les noms des interfaces que **dhcpd** doit écouter à la commande du paramètre **ExecStart**:

     ```
     ExecStart=/usr/sbin/dhcpd -f -cf /etc/dhcp/dhcpd.conf -user dhcpd -group dhcpd --no-pid
     $DHCPDARGS enp0s1 enp7s0
     ```

     Cet exemple configure que **dhcpd** n'écoute que sur les interfaces ***enp0s1*** et **enp7s0**.
  3. Recharger la configuration du gestionnaire **systemd**:

     ```
     # systemctl daemon-reload
     ```
  4. Redémarrez le service **dhcpd**:

     ```
     # systemctl restart dhcpd.service
     ```
- Pour les réseaux IPv6 :
  1. Copiez le fichier **/usr/lib/systemd/system/dhcpd6.service** dans le répertoire **/etc/systemd/system/**:

     ```
     # cp /usr/lib/systemd/system/dhcpd6.service /etc/systemd/system/
     ```

Ne modifiez pas le fichier **/usr/lib/systemd/system/dhcpd6.service**. Les futures mises à jour du paquet **dhcp-server** peuvent annuler les modifications.

2. Modifiez le fichier **/etc/systemd/system/dhcpd6.service** et ajoutez les noms des interfaces que **dhcpd** doit écouter à la commande du paramètre **ExecStart**:

```
ExecStart=/usr/sbin/dhcpd -f -6 -cf /etc/dhcp/dhcpd6.conf -user dhcpd -group dhcpd --no-pid $DHCPDARGS enp0s1 enp7s0
```

Cet exemple configure que **dhcpd** n'écoute que sur les interfaces ***enp0s1*** et **enp7s0**.

3. Recharger la configuration du gestionnaire **systemd**:

```
# systemctl daemon-reload
```

4. Redémarrez le service **dhcpd6**:

```
# systemctl restart dhcpd6.service
```

3.8. CONFIGURATION DU SERVICE DHCP POUR LES SOUS-RÉSEAUX DIRECTEMENT CONNECTÉS AU SERVEUR DHCP

Utilisez la procédure suivante si le serveur DHCP est directement connecté au sous-réseau pour lequel le serveur doit répondre aux requêtes DHCP. C'est le cas si une adresse IP de ce sous-réseau est attribuée à une interface réseau du serveur.

Selon que vous souhaitez fournir le DHCP pour IPv4, IPv6 ou les deux protocoles, voir la procédure pour :

- Réseaux IPv4
- Réseaux IPv6

Conditions préalables

- Vous êtes connecté en tant qu'utilisateur **root**.
- Le paquet **dhcp-server** est installé.

Procédure

- Pour les réseaux IPv4 :

1. Modifiez le fichier **/etc/dhcp/dhcpd.conf**:

a. Il est possible d'ajouter des paramètres globaux que **dhcpd** utilise par défaut si aucune autre directive ne contient ces paramètres :

```
option domain-name "example.com";
default-lease-time 86400;
```

Cet exemple définit le nom de domaine par défaut pour la connexion à **example.com**, et le délai de location par défaut à **86400** secondes (1 jour).

b. Ajoutez la déclaration **authoritative** sur une nouvelle ligne :

```
faisant autorité ;
```

IMPORTANT

Sans la mention **authoritative**, le service **dhcpd** ne répond pas aux messages **DHCPREQUEST** par **DHCPNAK** si un client demande une adresse qui n'est pas dans le pool.

c. Pour chaque sous-réseau IPv4 directement connecté à une interface du serveur, ajoutez une déclaration **subnet**:

```
subnet 192.0.2.0 netmask 255.255.255.0 {
  range 192.0.2.20 192.0.2.100;
  option domain-name-servers 192.0.2.1;
  option routers 192.0.2.1;
  option broadcast-address 192.0.2.255;
  max-lease-time 172800;
}
```

Cet exemple ajoute une déclaration de sous-réseau pour le réseau 192.0.2.0/24. Avec cette configuration, le serveur DHCP attribue les paramètres suivants à un client qui envoie une requête DHCP à partir de ce sous-réseau :

- Une adresse IPv4 libre dans la plage définie dans le paramètre **range**
- IP du serveur DNS pour ce sous-réseau : **192.0.2.1**
- Passerelle par défaut pour ce sous-réseau : **192.0.2.1**
- Adresse de diffusion pour ce sous-réseau : **192.0.2.255**
- Durée maximale du bail, après laquelle les clients de ce sous-réseau libèrent l'IP et envoient une nouvelle demande au serveur : **172800** secondes (2 jours)

2. En option, configurez **dhcpd** pour qu'il démarre automatiquement au démarrage du système :

```
# systemctl enable dhcpd
```

3. Démarrez le service **dhcpd**:

```
# systemctl start dhcpd
```

- Pour les réseaux IPv6 :

1. Modifiez le fichier **/etc/dhcp/dhcpd6.conf**:

a. Il est possible d'ajouter des paramètres globaux que **dhcpd** utilise par défaut si aucune autre directive ne contient ces paramètres :

```
option dhcp6.domain-search "example.com";
default-lease-time 86400;
```

Cet exemple définit le nom de domaine par défaut pour la connexion à **example.com**, et le délai de location par défaut à **86400** secondes (1 jour).

b. Ajoutez la déclaration **authoritative** sur une nouvelle ligne :

```
faisant autorité ;
```

IMPORTANT

Sans la mention **authoritative**, le service **dhcpd** ne répond pas aux messages **DHCPREQUEST** par **DHCPNAK** si un client demande une adresse qui n'est pas dans le pool.

c. Pour chaque sous-réseau IPv6 directement connecté à une interface du serveur, ajoutez une déclaration **subnet**:

```
subnet6 2001:db8:0:1::/64 {
  range6 2001:db8:0:1::20 2001:db8:0:1::100;
  option dhcp6.name-servers 2001:db8:0:1::1;
  max-lease-time 172800;
}
```

Cet exemple ajoute une déclaration de sous-réseau pour le réseau 2001:db8:0:1::/64. Avec cette configuration, le serveur DHCP attribue les paramètres suivants à un client qui envoie une requête DHCP à partir de ce sous-réseau :

- Une adresse IPv6 libre dans la plage définie dans le paramètre **range6**.
- L'IP du serveur DNS pour ce sous-réseau est **2001:db8:0:1::1**.
- La durée maximale du bail, après laquelle les clients de ce sous-réseau libèrent l'IP et envoient une nouvelle demande au serveur, est de **172800** secondes (2 jours). Notez que l'IPv6 nécessite l'utilisation de messages d'annonce de routeur pour identifier la passerelle par défaut.

2. En option, configurez **dhcpd6** pour qu'il démarre automatiquement au démarrage du système :

```
# systemctl enable dhcpd6
```

3. Démarrez le service **dhcpd6**:

```
# systemctl start dhcpd6
```

Ressources supplémentaires

- **dhcp-options(5)** page de manuel
- **dhcpd.conf(5)** page de manuel
- **/usr/share/doc/dhcp-server/dhcpd.conf.example** fichier
- **/usr/share/doc/dhcp-server/dhcpd6.conf.example** fichier

3.9. CONFIGURATION DU SERVICE DHCP POUR LES SOUS-RÉSEAUX QUI NE SONT PAS DIRECTEMENT CONNECTÉS AU SERVEUR DHCP

Utilisez la procédure suivante si le serveur DHCP n'est pas directement connecté au sous-réseau pour lequel il doit répondre aux requêtes DHCP. C'est le cas si un agent relais DHCP transmet les demandes au serveur DHCP, car aucune des interfaces du serveur DHCP n'est directement connectée au sous-réseau que le serveur doit desservir.

Selon que vous souhaitez fournir le DHCP pour IPv4, IPv6 ou les deux protocoles, voir la procédure pour :

- Réseaux IPv4
- Réseaux IPv6

Conditions préalables

- Vous êtes connecté en tant qu'utilisateur **root**.
- Le paquet **dhcp-server** est installé.

Procédure

- Pour les réseaux IPv4 :
  1. Modifiez le fichier **/etc/dhcp/dhcpd.conf**:
     a. Il est possible d'ajouter des paramètres globaux que **dhcpd** utilise par défaut si aucune autre directive ne contient ces paramètres :

```
option domain-name "example.com";
default-lease-time 86400;
```

     Cet exemple définit le nom de domaine par défaut pour la connexion à **example.com**, et le délai de location par défaut à **86400** secondes (1 jour).

     b. Ajoutez la déclaration **authoritative** sur une nouvelle ligne :

```
faisant autorité ;
```

IMPORTANT

Sans la mention **authoritative**, le service **dhcpd** ne répond pas aux messages **DHCPREQUEST** par **DHCPNAK** si un client demande une adresse qui n'est pas dans le pool.

     c. Ajoutez une déclaration **shared-network**, telle que la suivante, pour les sous-réseaux IPv4 qui ne sont pas directement connectés à une interface du serveur :

```
shared-network example {
  option domain-name-servers 192.0.2.1;
  ...

  subnet 192.0.2.0 netmask 255.255.255.0 {
```

```
    range 192.0.2.20 192.0.2.100;
    option routers 192.0.2.1;
  }

  subnet 198.51.100.0 netmask 255.255.255.0 {
    range 198.51.100.20 198.51.100.100;
    option routers 198.51.100.1;
  }
  ...
}
```

Cet exemple ajoute une déclaration de réseau partagé, qui contient une déclaration **subnet** pour les réseaux 192.0.2.0/24 et 198.51.100.0/24. Avec cette configuration, le serveur DHCP attribue les paramètres suivants à un client qui envoie une requête DHCP à partir de l'un de ces sous-réseaux :

- L'IP du serveur DNS pour les clients des deux sous-réseaux est : **192.0.2.1**.
- Une adresse IPv4 libre dans la plage définie dans le paramètre **range**, en fonction du sous-réseau à partir duquel le client a envoyé la demande.
- La passerelle par défaut est soit **192.0.2.1**, soit **198.51.100.1**, en fonction du sous-réseau à partir duquel le client a envoyé la demande.

d. Ajoutez une déclaration **subnet** pour le sous-réseau auquel le serveur est directement connecté et qui est utilisé pour atteindre les sous-réseaux distants spécifiés dans **shared-network** ci-dessus :

```
subnet 203.0.113.0 netmask 255.255.255.0 {
}
```

NOTE

Si le serveur ne fournit pas de service DHCP à ce sous-réseau, la déclaration **subnet** doit être vide, comme indiqué dans l'exemple. Sans déclaration pour le sous-réseau directement connecté, **dhcpd** ne démarre pas.

2. En option, configurez **dhcpd** pour qu'il démarre automatiquement au démarrage du système :

```
# systemctl enable dhcpd
```

3. Démarrez le service **dhcpd**:

```
# systemctl start dhcpd
```

- Pour les réseaux IPv6 :

1. Modifiez le fichier **/etc/dhcp/dhcpd6.conf**:

a. Il est possible d'ajouter des paramètres globaux que **dhcpd** utilise par défaut si aucune autre directive ne contient ces paramètres :

```
option dhcp6.domain-search "example.com";
default-lease-time 86400;
```

Cet exemple définit le nom de domaine par défaut pour la connexion à **example.com**, et le délai de location par défaut à **86400** secondes (1 jour).

b. Ajoutez la déclaration **authoritative** sur une nouvelle ligne :

```
faisant autorité ;
```

IMPORTANT

Sans la mention **authoritative**, le service **dhcpd** ne répond pas aux messages **DHCPREQUEST** par **DHCPNAK** si un client demande une adresse qui n'est pas dans le pool.

c. Ajoutez une déclaration **shared-network**, telle que la suivante, pour les sous-réseaux IPv6 qui ne sont pas directement connectés à une interface du serveur :

```
shared-network example {
  option domain-name-servers 2001:db8:0:1::1:1
  ...

  subnet6 2001:db8:0:1::1:0/120 {
    range6 2001:db8:0:1::1:20 2001:db8:0:1::1:100
  }

  subnet6 2001:db8:0:1::2:0/120 {
    range6 2001:db8:0:1::2:20 2001:db8:0:1::2:100
  }
  ...
}
```

Cet exemple ajoute une déclaration de réseau partagé qui contient une déclaration **subnet6** pour les réseaux 2001:db8:0:1::1:0/120 et 2001:db8:0:1::2:0/120. Avec cette configuration, le serveur DHCP attribue les paramètres suivants à un client qui envoie une requête DHCP à partir de l'un de ces sous-réseaux :

- L'IP du serveur DNS pour les clients des deux sous-réseaux est **2001:db8:0:1::1:1**.
- Une adresse IPv6 libre dans la plage définie dans le paramètre **range6**, en fonction du sous-réseau à partir duquel le client a envoyé la demande.
  Notez que l'IPv6 nécessite l'utilisation de messages d'annonce de routeur pour identifier la passerelle par défaut.

d. Ajoutez une déclaration **subnet6** pour le sous-réseau auquel le serveur est directement connecté et qui est utilisé pour atteindre les sous-réseaux distants spécifiés dans **shared-network** ci-dessus :

```
subnet6 2001:db8:0:1::50:0/120 {
}
```

NOTE

Si le serveur ne fournit pas de service DHCP à ce sous-réseau, la déclaration **subnet6** doit être vide, comme indiqué dans l'exemple. Sans déclaration pour le sous-réseau directement connecté, **dhcpd** ne démarre pas.

2. En option, configurez **dhcpd6** pour qu'il démarre automatiquement au démarrage du système :

```
# systemctl enable dhcpd6
```

3. Démarrez le service **dhcpd6**:

```
# systemctl start dhcpd6
```

Ressources supplémentaires

- **dhcp-options(5)** page de manuel
- **dhcpd.conf(5)** page de manuel
- **/usr/share/doc/dhcp-server/dhcpd.conf.example** fichier
- **/usr/share/doc/dhcp-server/dhcpd6.conf.example** fichier
- Mise en place d'un agent relais DHCP

3.10. ATTRIBUTION D'UNE ADRESSE STATIQUE À UN HÔTE À L'AIDE DE DHCP

En utilisant une déclaration **host**, vous pouvez configurer le serveur DHCP pour qu'il attribue une adresse IP fixe à l'adresse MAC (Media Access Control) d'un hôte. Cette méthode permet, par exemple, de toujours attribuer la même adresse IP à un serveur ou à un périphérique réseau.

Selon que vous souhaitez configurer des adresses fixes pour IPv4, IPv6 ou les deux protocoles, reportez-vous à la procédure décrite ci-dessous :

- Réseaux IPv4
- Réseaux IPv6

Conditions préalables

- Le service **dhcpd** est configuré et fonctionne.
- Vous êtes connecté en tant qu'utilisateur **root**.

Procédure

- Pour les réseaux IPv4 :
  1. Modifiez le fichier **/etc/dhcp/dhcpd.conf**:

a. Ajouter une déclaration **host**:

```
host server.example.com {
 hardware ethernet 52:54:00:72:2f:6e;
 fixed-address 192.0.2.130;
}
```

Cet exemple configure le serveur DHCP pour qu'il attribue toujours l'adresse IP **192.0.2.130** à l'hôte portant l'adresse MAC **52:54:00:72:2f:6e**.

Le service **dhcpd** identifie les systèmes par l'adresse MAC spécifiée dans le paramètre **fixed-address**, et non par le nom figurant dans la déclaration **host**. Par conséquent, vous pouvez attribuer à ce nom n'importe quelle chaîne de caractères qui ne correspond pas à d'autres déclarations **host**. Pour configurer le même système pour plusieurs réseaux, utilisez un nom différent, sinon **dhcpd** ne démarre pas.

b. Il est possible d'ajouter à la déclaration **host** d'autres paramètres spécifiques à cet hôte.

2. Redémarrez le service **dhcpd**:

```
# systemctl start dhcpd
```

- Pour les réseaux IPv6 :

1. Modifiez le fichier **/etc/dhcp/dhcpd6.conf**:

a. Ajouter une déclaration **host**:

```
host server.example.com {
 hardware ethernet 52:54:00:72:2f:6e;
 fixed-address6 2001:db8:0:1::200;
}
```

Cet exemple configure le serveur DHCP pour qu'il attribue toujours l'adresse IP **2001:db8:0:1::20** à l'hôte portant l'adresse MAC **52:54:00:72:2f:6e**.

Le service **dhcpd** identifie les systèmes par l'adresse MAC spécifiée dans le paramètre **fixed-address6**, et non par le nom figurant dans la déclaration **host**. Par conséquent, vous pouvez donner à ce nom n'importe quelle chaîne de caractères, à condition qu'elle soit unique par rapport aux autres déclarations **host**. Pour configurer le même système pour plusieurs réseaux, utilisez un nom différent car, dans le cas contraire, **dhcpd** ne démarre pas.

b. Il est possible d'ajouter à la déclaration **host** d'autres paramètres spécifiques à cet hôte.

2. Redémarrez le service **dhcpd6**:

```
# systemctl start dhcpd6
```

Ressources supplémentaires

- **dhcp-options(5)** page de manuel
- **/usr/share/doc/dhcp-server/dhcpd.conf.example** fichier

- **/usr/share/doc/dhcp-server/dhcpd6.conf.example** fichier

3.11. UTILISATION D'UNE DÉCLARATION DE GROUPE POUR APPLIQUER DES PARAMÈTRES À PLUSIEURS HÔTES, SOUS-RÉSEAUX ET RÉSEAUX PARTAGÉS EN MÊME TEMPS

En utilisant une déclaration **group**, vous pouvez appliquer les mêmes paramètres à plusieurs hôtes, sous-réseaux et réseaux partagés.

Notez que la procédure décrit l'utilisation d'une déclaration **group** pour les hôtes, mais que les étapes sont les mêmes pour les sous-réseaux et les réseaux partagés.

Selon que vous souhaitez configurer un groupe pour IPv4, IPv6 ou les deux protocoles, reportez-vous à la procédure décrite ci-dessous :

- Réseaux IPv4
- Réseaux IPv6

Conditions préalables

- Le service **dhcpd** est configuré et fonctionne.
- Vous êtes connecté en tant qu'utilisateur **root**.

Procédure

- Pour les réseaux IPv4 :
  1. Modifiez le fichier **/etc/dhcp/dhcpd.conf**:
     a. Ajouter une déclaration **group**:

```
group {
  option domain-name-servers 192.0.2.1;

  host server1.example.com {
    hardware ethernet 52:54:00:72:2f:6e;
    fixed-address 192.0.2.130;
  }

  host server2.example.com {
    hardware ethernet 52:54:00:1b:f3:cf;
    fixed-address 192.0.2.140;
  }
}
```

     Cette définition de **group** regroupe deux entrées de **host**. Le service **dhcpd** applique la valeur définie dans le paramètre **option domain-name-servers** aux deux hôtes du groupe.

     b. En option, ajoutez d'autres paramètres à la déclaration **group** qui sont spécifiques à ces hôtes.

  2. Redémarrez le service **dhcpd**:

```
# systemctl start dhcpd
```

- Pour les réseaux IPv6 :

  1. Modifiez le fichier **/etc/dhcp/dhcpd6.conf**:

     a. Ajouter une déclaration **group**:

```
group {
  option dhcp6.domain-search "example.com";

  host server1.example.com {
    hardware ethernet 52:54:00:72:2f:6e;
    fixed-address 2001:db8:0:1::200;
  }

  host server2.example.com {
    hardware ethernet 52:54:00:1b:f3:cf;
    fixed-address 2001:db8:0:1::ba3;
  }
}
```

     Cette définition de **group** regroupe deux entrées de **host**. Le service **dhcpd** applique la valeur définie dans le paramètre **option dhcp6.domain-search** aux deux hôtes du groupe.

     b. En option, ajoutez d'autres paramètres à la déclaration **group** qui sont spécifiques à ces hôtes.

  2. Redémarrez le service **dhcpd6**:

```
# systemctl start dhcpd6
```

Ressources supplémentaires

- **dhcp-options(5)** page de manuel
- **/usr/share/doc/dhcp-server/dhcpd.conf.example** fichier
- **/usr/share/doc/dhcp-server/dhcpd6.conf.example** fichier

3.12. RESTAURATION D'UNE BASE DE DONNÉES DE BAUX CORROMPUE

Si le serveur DHCP enregistre une erreur liée à la base de données des baux, telle que **Corrupt lease file - possible data loss!**, vous pouvez restaurer la base de données des baux à partir de la copie créée par le service **dhcpd**. Notez que cette copie peut ne pas refléter le dernier état de la base de données.

AVERTISSEMENT

Si vous supprimez la base de données des baux au lieu de la remplacer par une sauvegarde, vous perdez toutes les informations relatives aux baux actuellement attribués. Par conséquent, le serveur DHCP peut attribuer à des clients des baux qui ont déjà été attribués à d'autres hôtes et qui n'ont pas encore expiré. Cela entraîne des conflits d'adresses IP.

Selon que vous souhaitez restaurer les bases de données DHCPv4, DHCPv6 ou les deux, reportez-vous à la procédure décrite ci-dessous :

- Restauration de la base de données des baux DHCPv4
- Restauration de la base de données des baux DHCPv6

Conditions préalables

- Vous êtes connecté en tant qu'utilisateur **root**.
- La base de données des baux est corrompue.

Procédure

- Restauration de la base de données des baux DHCPv4 :
  1. Arrêtez le service **dhcpd**:

     ```
     # systemctl stop dhcpd
     ```

  2. Renommer la base de données des baux corrompue :

     ```
     # mv /var/lib/dhcpd/dhcpd.leases /var/lib/dhcpd/dhcpd.leases.corrupt
     ```

  3. Restaurer la copie de la base de données des baux que le service **dhcp** a créée lorsqu'il a actualisé la base de données des baux :

     ```
     # cp -p /var/lib/dhcpd/dhcpd.leases~ /var/lib/dhcpd/dhcpd.leases
     ```

     IMPORTANT

     Si vous disposez d'une sauvegarde plus récente de la base de données des baux, restaurez plutôt cette sauvegarde.

  4. Démarrez le service **dhcpd**:

     ```
     # systemctl start dhcpd
     ```

- Restauration de la base de données des baux DHCPv6 :
  1. Arrêtez le service **dhcpd6**:

1. Arrêtez le service **dhcpd6**:

```
# systemctl stop dhcpd6
```

2. Renommer la base de données des baux corrompue :

```
# mv /var/lib/dhcpd/dhcpd6.leases /var/lib/dhcpd/dhcpd6.leases.corrupt
```

3. Restaurer la copie de la base de données des baux que le service **dhcp** a créée lorsqu'il a actualisé la base de données des baux :

```
# cp -p /var/lib/dhcpd/dhcpd6.leases~ /var/lib/dhcpd/dhcpd6.leases
```

IMPORTANT

Si vous disposez d'une sauvegarde plus récente de la base de données des baux, restaurez plutôt cette sauvegarde.

4. Démarrez le service **dhcpd6**:

```
# systemctl start dhcpd6
```

Ressources supplémentaires

- La base de données des baux du service dhcpd

3.13. MISE EN PLACE D'UN AGENT RELAIS DHCP

L'agent relais DHCP (**dhcrelay**) permet de relayer les requêtes DHCP et BOOTP d'un sous-réseau sans serveur DHCP vers un ou plusieurs serveurs DHCP situés sur d'autres sous-réseaux. Lorsqu'un client DHCP demande des informations, l'agent relais DHCP transmet la demande à la liste des serveurs DHCP spécifiés. Lorsqu'un serveur DHCP renvoie une réponse, l'agent relais DHCP transmet cette demande au client.

Selon que vous souhaitez configurer un relais DHCP pour IPv4, IPv6 ou les deux protocoles, reportez-vous à la procédure décrite ci-dessous :

- Réseaux IPv4
- Réseaux IPv6

Conditions préalables

- Vous êtes connecté en tant qu'utilisateur **root**.

Procédure

- Pour les réseaux IPv4 :
  1. Installez le paquetage **dhcp-relay**:

```
# dnf install dhcp-relay
```

2. Copiez le fichier **/lib/systemd/system/dhcrelay.service** dans le répertoire **/etc/systemd/system/**:

```
# cp /lib/systemd/system/dhcrelay.service /etc/systemd/system/
```

Ne modifiez pas le fichier **/usr/lib/systemd/system/dhcrelay.service**. Les futures mises à jour du paquet **dhcp-relay** peuvent annuler les modifications.

3. Modifiez le fichier **/etc/systemd/system/dhcrelay.service** et ajoutez le paramètre **-i *interface*** ainsi que la liste des adresses IP des serveurs DHCPv4 responsables du sous-réseau :

```
ExecStart=/usr/sbin/dhcrelay -d --no-pid -i enp1s0 192.0.2.1
```

Avec ces paramètres supplémentaires, **dhcrelay** écoute les requêtes DHCPv4 sur l'interface **enp1s0** et les transmet au serveur DHCP avec l'IP **192.0.2.1**.

4. Recharger la configuration du gestionnaire **systemd**:

```
# systemctl daemon-reload
```

5. En option, configurez le service **dhcrelay** pour qu'il démarre au démarrage du système :

```
# systemctl enable dhcrelay.service
```

6. Démarrez le service **dhcrelay**:

```
# systemctl start dhcrelay.service
```

- Pour les réseaux IPv6 :

1. Installez le paquetage **dhcp-relay**:

```
# dnf install dhcp-relay
```

2. Copiez le fichier **/lib/systemd/system/dhcrelay.service** dans le répertoire **/etc/systemd/system/** et nommez le fichier **dhcrelay6.service**:

```
# cp /lib/systemd/system/dhcrelay.service /etc/systemd/system/dhcrelay6.service
```

Ne modifiez pas le fichier **/usr/lib/systemd/system/dhcrelay.service**. Les futures mises à jour du paquet **dhcp-relay** peuvent annuler les modifications.

3. Modifiez le fichier **/etc/systemd/system/dhcrelay6.service** et ajoutez les éléments **-l *receiving_interface*** et **-u *outgoing_interface*** à la fin du fichier :

```
ExecStart=/usr/sbin/dhcrelay -d --no-pid -l enp1s0 -u enp7s0
```

Avec ces paramètres supplémentaires, **dhcrelay** écoute les demandes DHCPv6 sur l'interface **enp1s0** et les transmet au réseau connecté à l'interface **enp7s0**.

4. Recharger la configuration du gestionnaire **systemd**:

```
# systemctl daemon-reload
```

5. En option, configurez le service **dhcrelay6** pour qu'il démarre au démarrage du système :

```
# systemctl enable dhcrelay6.service
```

6. Démarrez le service **dhcrelay6**:

```
# systemctl start dhcrelay6.service
```

Ressources supplémentaires

- **dhcrelay(8)** page de manuel