Red Hat Enterprise Linux 7

Guide de Gestion des réseaux

Configuration et Gestion des réseaux dans Red Hat Enterprise Linux 7

Last Updated: 2018-04-17

Red Hat Enterprise Linux 7 Guide de Gestion des réseaux

Configuration et Gestion des réseaux dans Red Hat Enterprise Linux 7

Mirek Jahoda
Red Hat Customer Content Services
mjahoda@redhat.com

Jana Heves
Red Hat Customer Content Services

Stephen Wadeley
Red Hat Customer Content Services

Christian Huffman
Red Hat Customer Content Services

Notice légale

Résumé

Le Guide de Gestion des réseaux de Red Hat Enterprise Linux 7procure des informations importantes de configuration et d'administration des interfaces réseau, des réseaux et des services de réseau de Red Hat Enterprise Linux 7. Il s'adresse aux administrateurs de systèmes possédant une compréhension de base de Linux et de la gestion des réseaux. Cet ouvrage est basé sur le Guide de Déploiement de Red Hat Enterprise Linux 6 Les chapitres relatifs au réseautage sont extraits du Guide de développement et forment la base de cet ouvrage.

Table des matières

PARTIE I. IP NETWORKING

Cette partie décrit comment configurer le réseau sur Red Hat Enterprise Linux.

CHAPITRE 1. INTRODUCTION À RED HAT ENTERPRISE LINUX NETWORKING

1.1. STRUCTURE DE L'OUVRAGE

Toutes les nouveautés dans cet ouvrage ont été rédigées et organisées clairement en séparant la partie introductive comprenant les explications des concepts et les cas d'utilisation, des tâches de configuration. Red Hat Engineering Content Services espère que vous pourrez ainsi trouver rapidement les instructions de configuration dont vous avez besoin, tout en continuant de fournir quelques explications pertinentes et du matériel conceptuel qui puisse vous aider à comprendre et décider quelles sont les tâches importantes en fonction de vos besoins. Lorsque le matériel en provenance du guide *Guide de déploiement de Red Hat Enterprise Linux 6* a été réutilisé , il aura été examiné et modifié, si possible, pour s'adapter à cette nouvelle idée de séparer les concepts des tâches.

Le matériel est regroupé par objectifs plutôt que par méthodes. Des instructions sur la façon de réaliser une tâche spécifique à l'aide de différentes méthodes sont regroupées entre elles. Ceci a pour but de vous faciliter la tâche en matière de recherche d'informations sur la façon de réaliser une tâche particulière ou un objectif, tout en vous permettant de voir rapidement les différentes méthodes disponibles.

Dans chaque chapitre, les méthodes de configuration seront présentées dans l'ordre suivant :

- l'outil d'interface utilisateur de texte, **nmtui**,
- l'outil en ligne de commandes du **NetworkManager nmcli**,
- autres méthodes de lignes de commandes et utilisation des fichiers de configuration,
- une méthode d'interface utilisateur graphique (GUI), comme **nm-connection-editor** ou **control-network** pour diriger le **`NetworkManager`**.

L'interface en ligne de commandes peut être utilisée pour exécuter des commandes, d'où le terme *command-line interface* (ou CLI) mais la ligne de commandes peut également démarrer un éditeur, pour composer ou modifier des fichiers de configuration. C'est pourquoi l'utilisation des commandes **ip** et les fichiers de configuration, tels que fichiers **`ifcfg`**, sont documentés ensemble.

1.2. RÉSEAUX IP VERSUS RÉSEAUX NON-IP

La plupart des réseaux modernes appartiennent à l'une de grandes catégories. D'une part, les réseaux basés IP. Ce sont tous les réseaux qui communiquent via adresses de protocole Internet (IP), ce qui est la norme pour l'Internet et pour la plupart des réseaux internes aujourd'hui. En général, cela comprend l'Ethernet, les Modems Câble/DSL/Dial up, le Wi-Fi, et les connexions VPN, etc.

Il y a des réseaux non-IP de base. Ce sont des réseaux niches, habituellement très spécifiques, mais l'un d'entre eux en particulier a pris en importance, InfiniBand, à tel point que nous prenons la peine de le mentionner ici. InfiniBand n'étant pas un réseau IP, de nombreuses fonctionnalités et configurations habituellement utilisées sur les réseaux IP ne sont pas applicables à InfiniBand. La section Chapitre 9, *Configuration d'InfiniBand et des Réseaux RDMA* de ce guide couvre les prérequis de configuration et d'administration d'un réseau InfiniBand et également de la plus large classe de périphériques compatibles RDMA.

1.3. INTRODUCTION AU NETWORKMANAGER

Dans Red Hat Enterprise Linux 7, le service de réseautage est fourni par le **`NetworkManager`**, qui est

un démon de configuration et de contrôle de réseaux dynamiques, qui tente de conserver les périphériques réseau et les connexions actives quand elles sont disponibles. Les fichiers de configuration traditionnels de type **`ifcfg`** sont toujours prises en charge. Voir Section 1.8, « NetworkManager et les Scripts réseau » pour plus d'informations.

Tableau 1.1. Récapitulatif des applications et des outils de réseautage

Application ou Outil	Description
NetworkManager	Le démon de réseautage par défaut
nmtui	Une simple interface texte utilisateur (de l'anglais Text User Interface) (TUI) basée Curses pour le **NetworkManager**
nmcli	L'outil d'interface en ligne de commandes est offert afin de permettre aux utilisateurs et aux scripts d'interagir avec le **NetworkManager**.
control-center	Un outil d'interface utilisateur graphique fourni par le gnome-shell
nm-connection-editor	Une application GTK+ 3 disponible pour certaines tâches qui ne sont pas encore gérées par le **control-center**

Le **`NetworkManager`** peut configurer des alias de réseau, des adresses **`IP`**, des itinéraires statiques, les infos **`DNS`**, les connexions VPN, ainsi que de nombreux paramètres spécifiques aux connexions. Finalement, **`NetworkManager`** fournit un API via D-Bus qui permet aux applications de chercher et de contrôler la configuration et l'état du réseau.

1.4. INSTALLATION DU NETWORKMANAGER

Le **NetworkManager** est installé par défaut dans Red Hat Enterprise Linux. Si nécessaire, pour vous assurer que c'est bien le cas, exécutez la commande suivante en tant qu'utilisateur **`root`** :

```
~]# yum install NetworkManager
```

Pour obtenir des informations sur les privilèges utilisateur et sur la façon d'obtenir des privilèges, consulter le guide *Red Hat Enterprise Linux 7 System Administrator's Guide*.

1.4.1. Le démon du NetworkManager

Le démon du **NetworkManager** exécute avec les privilèges root et il est, par défaut, configuré pour démarrer dès l'amorçage. Vous pouvez déterminer si le démon **NetworkManager** est en cours d'exécution en saisissant cette commande :

```
~]$ systemctl status NetworkManager
NetworkManager.service - Network Manager
   Loaded: loaded (/lib/systemd/system/NetworkManager.service; enabled)
   Active: active (running) since Fri, 08 Mar 2013 12:50:04 +0100; 3 days
ago
```

La commande **systemctl status** rapportera que le **NetworkManager** est **Active: inactive (mort)** si le service **NetworkManager** n'est pas en cours d'exécution. Pour le démarrer pour la session en cours, exécutez la commande suivante en tant qu'utilisateur root :

```
~]# systemctl start NetworkManager
```

Exécutez la commande **systemctl enable** pour vous assurer que le **NetworkManager** démarre bien à chaque fois que le système est amorcé :

```
~]# systemctl enable NetworkManager
```

Pour obtenir plus d'informations sur le démarrage, l'arrêt et la gestion des services, voir le guide *Red Hat Enterprise Linux 7 System Administrator's Guide*.

1.4.2. Interactions avec le NetworkManager

Les utilisateurs n'ont pas d'interractions avec le service système du **NetworkManager** directement. Au lieu de cela, les utilisateurs effectuent des tâches de configuration de réseau par des outils d'interface utilisateur graphiques ou en ligne de commandes. Les outils suivants sont disponibles dans Red Hat Enterprise Linux 7:

1. Il existe une simple interface texte utilisateur (de l'anglais Text User Interface) (TUI) basée curses pour le **NetworkManager**, **nmtui**.

2. Un outil de ligne de commandes, **nmcli**, est fourni pour permettre aux utilisateurs et aux scripts d'interagir avec le **NetworkManager**. Notez que le **nmcli** peut être utilisé sur les systèmes de GUI-less comme serveurs pour contrôler tous les aspects de **NetworkManager**. Cet outil est sur un même pied d'égalité que les outils graphiques (GUI).

3. Le gnome-shell fournit également une icône de réseau dans sa zone de Notification qui représente les états de connexion du réseau, ainsi rapportés par le **NetworkManager**. L'icône possède plusieurs états qui servent d'indicateurs visuels pour le type de connexion que vous utilisez actuellement.

4. Un outil d'interface utilisateur graphique appelé **control-center**, fourni par le gnome-shell, est à la disposition des utilisateurs de bureaux. Il intègre un outil de configuration **réseau**. Pour le démarrer, appuyez sur la touche de **Super** pour afficher la vue d'ensemble des activités, saisir **control network** et appuyez sur la touche **Entrée**. La touche **Super** apparaîtra sous diverses formes, selon le clavier ou autre matériel, mais souvent sous la touche Windows ou la commande, et généralement à gauche de la barre d'espace.

5. Un outil d'interface utilisateur graphique, appelé **nm-connection-editor**, est disponible pour certaines tâches qui ne sont pas encore gérées par le **control-center**. Pour le démarrer, appuyer sur la touche **Super** pour accéder à la Vue d'ensemble des activités. et saisir **connexions de réseau** ou **nm-connection-editor**, puis **Entrée**.

Figure 1.1. États d'icône de connexion réseau

1.5. CONFIGURATION RÉSEAU UTILISANT UNE INTERFACE UTILISATEUR TEXTE (NMTUI)

L'outil d'interface utilisateur de texte (TUI) du **NetworkManager**, **nmtui**, fournit une interface texte pour configurer le réseau en contrôlant le **NetworkManager**. L'outil est contenu dans le paquet NetworkManager-tui. Au moment de la rédaction, il n'est pas installé dans le **NetworkManager** par défaut. Pour installer le NetworkManager-tui, émettez la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install NetworkManager-tui
```

Si besoin est, pour obtenir des informations sur la façon de vérifier que le **NetworkManager** est en cours d'exécution, consulter Section 1.4.1, « Le démon du NetworkManager ».

Pour démarrer le **nmtui**, exécutez la commande suivante :

```
~]$ nmtui
```

L'interface utilisateur texte apparaît. Pour naviguer, utiliser les flèches ou appuyer sur **Tab** pour continuer et appuyer sur la combinaision de touches **Maj**+**Tab** pour revenir aux options. Appuyer sur la touche **Entrée** pour sélectionner une option. La barre **Espace** active/désactive le statut d'une case à cocher.

Les commandes suivantes sont disponibles :

- **nmtui edit*connection-name***

  Si aucun nom de connexion n'est fourni, le menu de sélection s'affiche. Si le nom de la connexion est fourni et est correctement identifié, l'écran **Modifier connexion** s'affichera.

- **nmtui connect *connection-name***

  Si aucun nom de connexion n'est fourni, le menu de sélection s'affichera. Si le nom de connexion est fourni et est correctement identifié, la connexion qui convient sera activée. Toute commande non valide affiche un message d'utilisation.

Au moment de la rédaction, **nmtui** ne supporte pas tous les types de connexions. En particulier, vous ne pouvez pas modifier les VPN, connexions Wi-Fi utilisant WPA Enterprise ou des connexions Ethernet à l'aide de **802. 1X**.

1.6. CONFIGURATION DE RÉSEAU AVEC L'INTERFACE CLI (NMCLI) DU NETWORKMANAGER

L'outil de ligne de commandes du **NetworkManager**, **nmcli**, fournit une façon de configurer le réseau en contrôland le **NetworkManager** par ligne de commandes. Il est installé avec le **NetworkManager** par défaut. Si nécessaire, pour savoir comment vérifier si le **NetworkManager** est en cours d'exécution, consulter Section 1.4.1, « Le démon du NetworkManager ».

Des exemples d'utilisation du l'outil **nmcli** pour chaque tâche seront inclus quand c'est possible, avant d'expliquer l'utilisation des autres méthodes en ligne de commandes ou en interfaces graphiques. Voir Section 2.3, « Utiliser l'outil de ligne de commandes du NetworkManager, nmcli » pour obtenir une introduction à **nmcli**, et consulter la page **man nmcli-examples(7)** pour trouver des exemples. Pour trouver les propriétés disponibles pour les commandes **nmcli c add** et **nmcli c modify**, consulter la page man **man nm-settings(5)**.

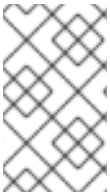
1.7. CONFIGURATION RÉSEAU PAR INTERFACE EN LIGNE DE COMMANDES (CLI)

Les commandes de l'utilitaire **ip**, parfois dénommé iproute2 d'après le nom du package en amont, sont documentés dans la page de **`man ip(8)`**. Le nom du package dans Red Hat Enterprise Linux 7 est iproute. Si nécessaire, vous pouvez vérifier que l'utilitaire **ip** est bien installé, en vérifiant son numéro de version comme suit :

```
~]$ ip -V
ip utility, iproute2-ss130716
```

Les commandes **ip** peuvent être utilisées pour ajouter ou supprimer des adresses et des itinéraires pour les interfaces en parallèle, avec le **NetworkManager**, qui va veiller à leur conservation et les reconnaître dans **nmcli**, **nmtui**, **control-center** et l'API D-Bus.

Notez que l'utilitaire **ip** remplace l'utilitaire **`ifconfig`**, car le paquet net-tools (qui fournit un **`ifconfig`**) ne supporte pas les adresses InfiniBand. La commande **`ip help`** imprime un message d'utilisation. Il y a une aide spécifique aux OBJETS, par exemple : **`ip link help`** et **`ip addr help`**.



NOTE

Les commandes **ip** données en ligne de commandess ne seront pas persitantes après le redémarrage du système. Si la persistance est requise, utiliser des fichiers de configuration (fichiers **`ifcfg`**) ou ajouter les commandes à un script.

Des exemples d'utilisation du l'outil en ligne de commandes et des fichiers de configuration pour chaque tâche seront inclus après les exemples **nmtui** et **nmcli**, mais avant d'expliquer l'utilisation d'une des autres méthodes en interface graphique du **NetworkManager**, plus précisément, **control-center** et **nm-connection-editor**.

1.8. NETWORKMANAGER ET LES SCRIPTS RÉSEAU

Dans les versions précédentes de Red Hat Enterprise Linux, le mode par défaut pour configurer le réseau utilisait des *scripts réseau*. Le terme *script réseau* est couramment utilisé pour décrire le script **`/etc/init.d/network`** et tout autre script installé qu'il évoque. Les fichiers fournis par l'utilisateur sont généralement considérés comme fichiers de configuration, mais peuvent aussi être interprétés comme un amendement aux scripts.

Bien que le **NetworkManager** fournisse le service de réseautage par défaut, les développeurs de Red Hat ont travaillé dur pour s'assurer que les scripts et le **NetworkManager** collaborent. Les administrateurs, qui sont habitués aux scripts, peuvent continuer à les utiliser. Nous espérons que les deux systèmes vont pouvoir exécuter en parallèle et bien coopérer. Il est prévu que la plupart des scripts shell utilisateur des versions précédentes continueront de fonctionner. Red Hat recommande que vous les testiez tout d'abord.

Exécuter le script de réseau

Exécutez le script **uniquement** avec l'utilitaire **systemctl** qui supprimera les variables d'environnement existantes, et qui veillera à une exécution sans reproche. La commande prend la forme suivante :

```
systemctl start|stop|restart|status network
```

Notez que dans Red Hat Enterprise Linux 7, le **NetworkManager** est tout d'abord démarré, et **`/etc/init.d/network`** vérifiera avec le **NetworkManager** qu'il n'y a pas d'interférence avec les

connexions du **NetworkManager**. Le **NetworkManager** est sensé être la première application qui utilise les fichiers de configuration sysconfig et **`/etc/init.d/network`** est secondaire, jouant un rôle de soutien en cas d'échec.

Le script **`/etc/init.d/network`** n'est pas lié aux événements, il exécute soit :

1. manuellement (par l'une des commandes **`systemctl start|stop|restart network`**),

2. au démarrage et à l'arrêt si le service réseau est actif (suite à la commande **`systemctl enable network`**).

C'est un processus manuel qui ne réagit pas à des événements qui ont eu lieu après le démarrage. Les utilisateurs peuvent aussi appeler les scripts **`ifup`** et **`ifdown`** manuellement.

Commandes personnalisées et Scripts de réseau

On exécute des commandes personnalisées dans les scripts **`/sbin/ifup-local`** **`ifdown-pre-local`** et **`ifdown locale`** uniquement lorsque ces périphériques sont contrôlés par le service **`/etc/init.d/network`**. Si vous avez modifié les initscripts eux-mêmes (par exemple, **`/etc/sysconfig/network-scripts/ifup-eth`**) alors ces modifications sont remplacées par une mise à jour du paquet initscripts. Il est donc recommandé de ne pas modifier les initscripts directement et de faire usage des scripts **`/sbin/if*local`**, afin que vos modifications personnalisées puissent survivre aux mises à jour des packages. Les initscripts vérifient juste la présence de ladite **`/sbin/if*local`** et les exécutent s'ils existent. Les initscripts ne mettent rien dans les scripts **`/sbin/if*local`**, et le RPM initscripts (ou tout autre paquet) ne posséde, ni ne peut modifier ces fichiers.

Il y a plusieurs façons d'exécuter des tâches personnalisées lorsque les connexions réseau sont up ou down, avec les anciens scripts réseau, ainsi qu'avec le **NetworkManager**. Lorsque le **NetworkManager** est activé, le script **`ifup`** et **`ifdown`** demandera au **NetworkManager** si le **NetworkManager** gère l'interface en question, qui se trouve sur la ligne « DEVICE= » dans le fichier **`ifcfg`**. Si le **NetworkManager** gère bien ce périphérique et que le périphérique n'est pas déjà connecté, alors **`ifup`** demandera au **NetworkManager** de démarrer la connexion.

- Si le périphérique est géré par le **NetworkManager** et qu'il **est** déjà connecté, rien ne se produira.

- Si le périphérique n'est pas géré par le **NetworkManager**, alors les scripts démarreront une connexion par les mécanismes anciens, non-**NetworkManager**, qui étaient utilisés avant la venue du **NetworkManager**.

Si vous évoquez "**`ifdown`**" et que le périphérique est géré par le **NetworkManager**, alors **`ifdown`** demandera au **NetworkManager** de terminer la connexion.

Les scripts surveillent le **NetworkManager** de façon dynamique, donc, si le **NetworkManager** n'est pas en cours d'exécution, les scripts se retourneront sur les anciens mécanismes basés scripts pré-**NetworkManager**.

1.9. CONFIGURATION DE RÉSEAU PAR LES FICHIERS SYSCONFIG

Le répertoire **`/etc/sysconfig/`** est l'emplacement où se trouvent les scripts et les fichiers de configuration. La plupart des informations de configuration de réseaux se situent à cet endroit, à l'exception des configurations VPN, mobiles haut débit, et PPPoE qui se trouvent dans les sous-répertoires **`/etc/NetworkManager/`**. Par exemple, les informations spécifiques interfaces se trouvent dans le répertoire **`ifcfg`** du répertoire **`/etc/sysconfig/network-scripts/`**.

Le fichier **/etc/sysconfig/network** est utilisé pour les configurations globales. Les informations sur les connexions PPPoE, mobiles haut débit, et VPN sont stockées dans le fichier **/etc/NetworkManager/system-connections/**.

Dans Red Hat Enterprise Linux 7, quand vous modifiez un fichier **ifcfg**, le **NetworkManager** n'est pas automatiquement mis au courant du changement et doit être mis au courant du changement par une notification. Si vous utilisez un des outils pour mettre le profil du **NetworkManager** à jour, alors, le **NetworkManager** n'implémentera pas ces changements tant que vous ne vous serez pas reconnecté en utilisant ce profil. Par exemple, si les fichiers de configuration ont été changés en utilisant un éditeur, le **NetworkManager** devra être instruit de lire les fichiers de configuration à nouveau. Pour cela, exécutez la commande siuvante en tant qu'utilisateur **root** :

```
~]# nmcli connection reload
```

La commande ci-dessus interprétera tous les profils de connexion. Sinon, pour télécharger à nouveau un fichier modifé à la fois, **ifcfg-*ifname***, exécutez une commande ainsi :

```
~]# nmcli con load /etc/sysconfig/network-scripts/ifcfg-ifname
```

La commande accepte des noms de fichiers multiples. Ces commandes exigent les privilèges d'utilisateur **root**. Pour obtenir des informations sur les privilèges utilisateur et sur la façon d'obtenir des privilèges, consulter le guide *Red Hat Enterprise Linux 7 System Administrator's Guide*, et les pages man **su(1)** et **sudo(8)**

Les modifications apportées à l'aide d'outils tels que le **nmcli** n'ont pas besoin d'un chargement à nouveau, mais nécessitent que l'interface associée soit arrêtée (down) et qu'elle soit réactivée (up) à nouveau. Cela peut être fait en utilisant des commandes sous le format suivant :

```
nmcli dev disconnectinterface-name
```

suivi de :

```
nmcli con up interface-name
```

Le **NetworkManager** n'active aucun des scripts réseau, mais les scripts réseau vont essayer d'activer le **NetworkManager** s'il exécute quand les commandes **ifup** sont utilisées. Voir Section 1.8, « NetworkManager et les Scripts réseau » pour une explication sur les scripts réseau.

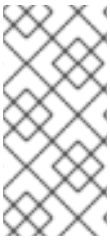
Le script **ifup** est un script générique qui fait un certain nombre de choses et faisant appel à des scripts spécifiques à une interface, comme **ifup-ethX**, **ifup-wireless**, **ifup-ppp**, etc. Quand un utilisateur exécute **ifup eth0** manuellement, voici ce qui se passe :

1. **ifup** cherche un fichier nommé **/etc/sysconfig/network-scripts/ifcfg-eth0**;
2. Si le fichier **ifcfg** existe, **ifup** ira chercher la clé **TYPE** dans ce fichier pour déterminer quel script spécifique à ce type appeler :
3. **ifup** appelle **ifup-wireless** ou **ifup-eth** ou **ifup-XXX** sur la base du **TYPE** ;
4. les scripts de type particulier exécutent des installations spécifiques :
5. et les scripts de type spécifiques laissent les fonctions courantes exécuter des tâches liées à l'**IP** comme **DHCP** ou une installation statique.

Au démarrage, **/etc/init.d/network** analyse tous les fichiers **ifcfg** et pour chacun d'eux ayant

ONBOOT=yes, il vérifie si le **NetworkManager** a déjà démarré le PÉRIPHÉRIQUE de ce fichier **ifcfg**. Si le **NetworkManager** démarre ce périphérique ou l'a déjà démarré, rien de plus n'est fait pour ce fichier, et le fichier **ONBOOT=yes** suivant est vérifié. Si le **NetworkManager** n'a pas encore démarré ce périphérique, les initscripts continueront avec leur comportement traditionnel et appeleront **ifup** pour ce fichier **ifcfg**.

Le résultat final est que n'importe quel fichier **ifcfg** qui a **ONBOOT=yes** devrait être lancé au démarrage du système, soit par le **NetworkManager** ou par les initscripts. De cette manière, certains types de réseaux hérités que le **NetworkManager** ne gère pas (comme RNIS ou les modems DIAL-UP analogues) ainsi que toute nouvelle application non encore supportée par le **NetworkManager** sont toujours correctement démarrés par les initscripts même si le **NetworkManager** est incapable de les gérer.



NOTE

Il est recommandé de ne pas stocker les fichiers de sauvegarde **ifcfg** au même endroit que les fichiers live. Le script fait littéralement **ifcfg-*** en excluant uniquement les extensions : **.old**, **.orig**, **.rpmnew**, **.rpmorig** et **.rpmsave**. Le meilleur moyen est ne pas stocker des fichiers de sauvegarde n'importe où dans le fichier **/etc /**.

1.10. RESSOURCES SUPPLÉMENTAIRES

Les sources d'informations suivantes vous donneront d'autres ressources sur le réseautage dans Red Hat Enterprise Linux 7.

1.10.1. Documentation installée

- Page man **man(1)** — décrit les pages de manuel et comment les trouver.
- Page man **NetworkManager(8)** — décrit le démon de gestion du réseau
- Page man **NetworkManager.conf(5)** — décrit le fichier de configuration du **NetworkManager**.
- **/usr/share/doc/initscripts-*version*/sysconfig.txt** — décrit les fichiers de configuration **ifcfg** et leurs directives telles qu'elles sont comprises dans l'ancien service réseau.

CHAPITRE 2. CONFIGURER LA GESTION DES RÉSEAUX IP

2.1. PARAMÈTRES D'INTERFACE STATIQUE ET DYNAMIQUE

Quand utiliser l'adressage statique et quand utiliser l'adressage dynamique ? Ces décisions sont subjectives, elles dépendent de vos besoins d'accès, et de vos exigences spécifiques. Disposer d'une politique, la documenter et l'appliquer régulièrement est généralement plus important que les décisions spécifiques que vous prenez. Dans un réseau local de société traditionnelle, c'est une décision plus facile à prendre car que vous avez, en général, moins de serveurs que les autres hôtes. La mise à disposition et l'installation d'outils à cet effet, facilitent la mise en oeuvre de configurations statiques dans de nouveaux hôtes et l'utilisation de tels outils change votre flux de travail et ses exigences. Les deux sections suivantes visent à fournir des conseils de base à ceux qui ne sont pas encore passés par ce processus décisionnel. Les administrateurs système expérimentés risquent d'avoir déjà leur propre ensemble de règles et d'exigences et celles-ci peuvent différer de ce dont on parle à présent. Pour plus d'informations sur la configuration automatisée et de gestion, consultez la section **OpenLMI** du guide *Guide d'administration de systèmes de Red Hat Enterprise Linux 7*. Le *Guide d'Installation Red Hat Enterprise Linux 7* documente l'utilisation de **kickstart**, pouvant être également utilisé pour automatiser l'affectation de paramètres de réseaux.

2.1.1. Quand utiliser les paramètres de configuration de l'interface de réseau statique

Utiliser l'adressage statique **IP** sur les serveurs et sur les périphériques dont vous souhaitez contrôler la disponibilité de réseau, quand les méthodes d'affectation d'adresses automatiques, comme **DHCP** échouent. Les serveurs, **DHCP**, et **DNS** sont des exemples typiques. Les interfaces pour les périphériques de gestion out-of-band valent également la peine d'être configurés avec des paramètres statiques, car ces périphériques sont supposés fonctionner, dans la mesure du possible, indépendamment des autres infrastructures de réseau.

Pour les hôtes considérés comme étant non vitaux, mais pour lesquels l'adressage **IP** statique est toujours considéré comme étant souhaitable, utiliser une méthode d'affectation d'adresses automatique. Par exemple, les serveurs **DHCP** peuvent être configurés de manière à allouer la même adresse **IP** à un hôte à chaque fois. Cette méthode peut être utilisée, par exemple, pour les imprimantes en commun.

Tous les outils de configuration répertoriés dans Section 2.1.3, « Sélection des méthodes de configuration de réseau » permettent l'affectation d'adresses **IP** statiques manuelle. L'outil **nmcli** est également souhaitable pour une affectation de configuration de réseau par le biais d'un script.

2.1.2. Quand faut-il utliser les paramètres de configuration d'interface dynamique ?

Activer et utiliser une affectation dynamique des adresses **IP** et autres informations de réseau lorsqu'il n'y a aucune raison impérieuse de ne pas le faire. Le temps économisé à planifier et documenter les réglages manuels peut être mieux utilisé par ailleurs. Le *protocole de contrôle d'hôte dynamique* (DHCP) est une méthode traditionnelle pour assigner dynamiquement des configurations réseau aux hôtes. Voir Section 10.1, « Pourquoi utiliser DHCP ? » pour plus d'informations à ce sujet.

NetworkManager appelera le client **DHCP** par défaut, **dhclient**, quand un profil a été défini pour pouvoir recevoir des adresses automatiquement, ou quand un fichier de configuration d'interface a BOOTPROTO défini à **dhcp**. Quand **DHCP** est exigé, une instance du **dhclient** sera démarrée pour chaque protocole Internet, **IPv4** ou **IPv6**, sur une interface. Quand le **NetworkManager** n'est pas en cours d'exécution, ou n'exécute pas une interface, le service de réseautage d'origine appelera des instances du **dhclient** selon les besoins.

2.1.3. Sélection des méthodes de configuration de réseau

- **Pour configurer une interface avec l'outil d'interface d'utilisateur texte du NetworkManager**, **nmtui**, consulter Section 2.2, « Configurer l'interface d'utilisateur texte, nmtui »
- **Pour configurer une interface avec l'outil en ligne de commandes du NetworkManager**, **nmtui**, consulter Section 2.3, « Utiliser l'outil de ligne de commandes du NetworkManager, nmcli »
- **Pour configurer une interface de réseau manuellement**, voir Section 2.4, « Utilisation de l'interface en ligne de commandes (CLI) ».
- **Pour configurer un réseau à l'aide d'outils d'interface utilisateur graphiques**, voir Section 2.5, « Utiliser le NetworkManager avec l'interface graphique GNOME »

2.2. CONFIGURER L'INTERFACE D'UTILISATEUR TEXTE, NMTUI

L'outil d'interface utilisateur de texte **nmtui** peut être utilisé pour configurer une interface dans une fenêtre de terminal. Exécutez la commande suivante pour démarrer l'outil :

```
~]$ nmtui
```

L'interface utilisateur texte apparaîtra. Tout commande non valide affichera un message d'utilisation.

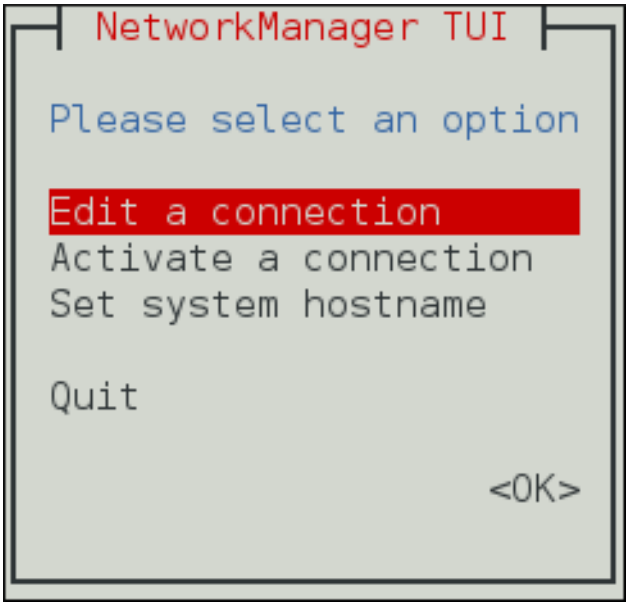


Figure 2.1. Le menu de démarrage de l'interface utilisateur texte du NetworkManager

Pour naviguer, utiliser les flèches ou appuyer sur **Tab** pour continuer et appuyer sur la combinaision de touches **Maj**+**Tab** pour revenir aux options. Appuyer sur la touche **Entrée** pour sélectionner une option. La barre **Espace** active/désactive le statut d'un case à cocher.

Voir Section 1.5, « Configuration réseau utilisant une interface utilisateur texte (nmtui) » pour obtenir des informations sur la façon d'installer **nmtui**.

2.3. UTILISER L'OUTIL DE LIGNE DE COMMANDES DU NETWORKMANAGER, NMCLI

L'outil de ligne de commandes **nmcli** peut être utilisé à la fois par les utilisateurs et les scripts pour contrôler le **NetworkManager**. Le format de base d'une commande est le suivant :

```
nmcli OPTIONS OBJECT { COMMAND | help }
```

ayant comme OBJECT pour **général**, **réseautage**, **radio**, **connexion**, ou **périphérique**. Les options les plus utilisées sont les suivantes : **-t, --terse** pour les scripts, l'option **-p, --pretty** pour les utilisateurs et l'option **-h, --help** pour assistance. La complétion de commande est maintenant en place pour **nmcli**, donc n'oubliez pas d'appuyer sur **Tab** (onglet) quand vous hésitez sur les commandes disponibles. Voir la page man **nmcli(1)** pour obtenir une liste complète des options et des commandes. Pour obtenir une liste des configurations des périphérique de réseau connus, exécuter la commande **nmcli device** sans arguments.

L'outil **nmcli** contient une assistance contectuelle intégrée. Pour l'utiliser, saisir les deux commande suivantes, et constatez la différence :

```
~]$ nmcli help
Usage: nmcli [OPTIONS] OBJECT { COMMAND | help }

OPTIONS
  -t[erse]                                  terse output
  -p[retty]                                 pretty output
  -m[ode] tabular|multiline                 output mode
  -f[ields] <field1,field2,...>|all|common   specify fields to output
  -e[scape] yes|no                          escape columns separators in
values
  -n[ocheck]                                don't check nmcli and
NetworkManager versions
  -a[sk]                                    ask for missing parameters
  -w[ait] <seconds>                         set timeout waiting for
finishing operations
  -v[ersion]                                show program version
  -h[elp]                                   print this help

OBJECT
  g[eneral]       NetworkManager's general status and operations
  n[etworking]    overall networking control
  r[adio]         NetworkManager radio switches
  c[onnection]    NetworkManager's connections
  d[evice]        devices managed by NetworkManager
  a[gent]         NetworkManager secret agent or polkit agent
  m[onitor]       monitor NetworkManager changes
```

```
~]$ nmcli general help
Usage: nmcli general { COMMAND | help }

  COMMAND := { status | hostname | permissions | logging }

  status

  hostname [<hostname>]

  permissions

  logging [level <log level>] [domains <log domains>]
```

Dans le deuxième exemple ci-dessus, l'aide disponible se rapporte à l'objet **general**.

La page man **nmcli-examples(5)** contient un certain nombre d'exemples utiles. En voici une brève sélection :

Pour afficher le statut global du **NetworkManager** :

```
nmcli statut general
```

. Pour contrôler la journalisation **NetworkManager** :

```
nmcli general logging
```

. Pour afficher toutes les connexions :

```
nmcli connection show
```

. Pour montrer les connexions actives uniquement, ajoutez l'option **- a,--active**, comme suit :

```
nmcli connection show --active
```

. Pour afficher les périphériques reconnus par le **NetworkManager** et leur état :

```
nmcli device status
```

Le commandes peuvent être écourtées et certaines options omises. Par exemple, la commande :

```
nmcli connection modify id 'MyCafe' 802-11-wireless.mtu 1350
```

peut être réduite à la commande suivante :

```
nmcli con mod MyCafe 802-11-wireless.mtu 1350
```

. L'option **id** peut été omise car l'ID de connexion (nom) est sans équivoque pour **nmcli** dans ce cas. Au fur et à mesure que vous vous familiariserez avec les commandes, d'autres abréviations peuvent être faites. Par exemple,

```
nmcli connection add type ethernet
```

peut être réduit à:

```
nmcli c a type eth
```

NOTE

N'oubliez pas de vous servir de la fonction de complétion de commande si vous avez un doute.

Lancer et arrêter une interface en utilisant nmcli

L'outil **nmcli** peut être utilisé pour démarrer et pour stopper une interface de réseau, y compris les masters. Ainsi :

```
nmcli con up id bond0
nmcli con up id port0
nmcli dev disconnect bond0
nmcli dev disconnect ens3
```

NOTE

Il est recommandé d'utiliser **nmcli dev disconnect *iface-name*** plutôt que **nmcli con down id *id-string***, car la disconnexion met l'interface en mode « manual », et aucune connexion automatique ne démarrera tant que l'utilisateur n'aura pas demandé au **NetworkManager** de démarrer une connexion ou tant qu'un événement externe tel qu'un changement d'opérateur, une hibernation,une mise en veille n'ait eu lieu.

L'éditeur de connexions interactives nmcli

L'outil **nmcli** contient un éditeur de connexions interactif. Pour l'utiliser, saisir la commande suivante :

```
~]$ nmcli con edit
```

On vous demandera de saisir un type de connexion valide à partir de la liste affichée. Après avoir entré un type de connexion, vous serez placé sur invite de **nmcli**. Si vous êtes familier avec les types de connexion, vous pouvez ajouter une option de **type** de connexion valide à la commande **nmcli con edit**, et être pris directement à l'invite de **nmcli**. Le format est le suivant pour l'édition d'un profil de connexion existant :

```
nmcli con edit [id | uuid | path] ID
```

. Pour ajouter ou modifier un nouveau profil de connexion, appliquer le format suivant :

```
nmcli con edit [saisir new-connection-type de type] [con-name new-
connexion- name]
```

Saisir **help** à l'invite de **nmcli** pour voir une liste de commandes valides. Utiliser la commande **describe** pour obtenir une liste des paramètres et de leurs propriétés. Le format est le suivant :

```
describe setting.property
```

. Par exemple :

```
nmcli> describe team.config
```

2.3.1. Comprendre les options nmcli

La plupart des commandes **nmcli** s'expliquent par elles-mêmes, mais pour certaines commandes, cela vaut la peine de se pencher un petit moment dessus :

type — le type de connexion.

Les valeurs autorisées sont les suivantes : **adsl**, **bond**, **bond-slave**, **bridge**, **bridge-slave**, **bluetooth**, **cdma**, **ethernet**, **gsm**, **infiniband**, **olpc-mesh**, **team**, **team-slave**, **vlan**, **wifi**, **wimax**.

Chaque type de connexion contient des options de commande spécifiques à un type. Appuyer sur

Tab pour en obtenir une liste ou pour voir la liste **TYPE_SPECIFIC_OPTIONS** dans la page man **nmcli(1)**. L'option **type** s'applique après les commandes suivantes : **nmcli connection add** et **nmcli connection edit**.

con-name — le nom assigné à un profil de connexion.

Si vous n'indiquez pas de nom de connexion, il sera généré suivant le format suivant :

```
type-ifname[-number]
```

Le nom de connexion est le nom d'un *profil de connexion* et ne doit pas être confondu avec le nom de l'interface qui représente un périphérique comme wlan0, ens3, em1 et ainsi de suite). Les utilisateurs peuvent toutefois nommer les connexions d'après les interfaces, mais ce n'est pas la même chose. Il peut y avoir plusieurs profils de connexion pour un périphérique. Ceci est particulièrement utile pour les appareils mobiles ou quand on fait des changements de câble réseau entre différents appareils. Plutôt que de modifier la configuration, créer des profils différents et les appliquer à l'interface selon les besoins. L'option **id** fait également référence au nom du profil de connexion.

id — une chaîne d'identification assignée par l'utilisateur à un profil de connexion.

L'ID peut être utilisé avec les commandes **nmcli connection** pour identifier une connexion. Le champ NAME figurant dans la sortie indique l'ID de connexion (nom). Il se rapporte au même nom de connexion que celui du **con-name**.

uuid — une chaîne d'identification unique assignée par le système à un profil de connexion.

L'UUID peut être utilisé par les commandes **nmcli connection** pour identifier une connexion.

2.3.2. Se connecter à réseau par nmcli

Pour obtenir une liste des connexions de réseau disponibles, lancez la commande suivante :

```
~]$ nmcli con show
NAME              UUID                                   TYPE
DEVICE
Auto Ethernet     9b7f2511-5432-40ae-b091-af2457dfd988  802-3-ethernet  --
ens3              fb157a65-ad32-47ed-858c-102a48e064a2  802-3-ethernet
ens3
MyWiFi            91451385-4eb8-4080-8b82-720aab8328dd  802-11-wireless
wlan0
```

Notez que le champ de nom NAME dans la sortie indique toujours l'ID de connexion (nom). Il ne correspond pas au nom de l'interface même s'il a le même aspect. Pour la seconde connexion mentionnée ci-dessus, **ens3** du champ NAME correspond à l'ID de connexion donné au profil qui s'applique à l'interface ens3. Dans la dernière connexion ci-dessus, l'utilisateur assigne l'ID de connexion **MyWiFi** à l'interface wlan0.

L'ajout d'une connexion Ethernet revient à créer un profil de configuration, qui est alors assigné à un périphérique. Avant de créer un nouveau profil, vérifiez les périphériques suivants :

```
~]$ nmcli dev status
DEVICE  TYPE      STATE         CONNECTION
ens3    ethernet  disconnected  --
```

```
ens9    ethernet  disconnected  --
lo      loopback  unmanaged     --
```

Ajouter une connexion Ethernet dynamique

Pour ajouter un profil de configuration Ethernet à une configuration **IP** dynamique, ce qui permet à **DHCP** d'assigner la configuration de réseau, on peut utiliser une commande du format suivant :

```
nmcli connection add type ethernet con-name connection-name ifname
interface-name
```

Ainsi, pour créer un profil de connexion dynamique nommé *my-office*, exécutez la commande suivante :

```
~]$ nmcli con add type ethernet con-name my-office ifname ens3
Connection 'my-office' (fb157a65-ad32-47ed-858c-102a48e064a2) successfully
added.
```

NetworkManager définira son paramètre interne **connection.autoconnect** à **yes**. Le **NetworkManager** inscrira également des paramètres de config dans **/etc/sysconfig/network-scripts/ifcfg-my-office** avec la directive ONBOOT définie sur **yes**.

Notez que les changements manuels apportés au fichier ifcfg ne seront pas remarqués par le **NetworkManager** tant que l'interface n'est pas appelée à nouveau. Voir Section 1.9, « Configuration de réseau par les fichiers sysconfig » pour obtenir plus d'informations sur la façon d'utiliser les fichiers de configuration.

Pour afficher la connexion Ethernet, exécuter la commande suivante :

```
~]$ nmcli con up my-office
Connection successfully activated (D-Bus active path:
/org/freedesktop/NetworkManager/ActiveConnection/5)
```

Pour vérifier le statut des périphériques et des connexions :

```
~]$ nmcli device status
DEVICE  TYPE      STATE         CONNECTION
ens3    ethernet  connected     my-office
ens9    ethernet  disconnected  --
lo      loopback  unmanaged     --
```

Pour modifier le nom d'un hôte envoyé par un hôte à un serveur **DHCP**, modifier la propriété **dhcp-hostname** comme suit :

```
~]$ nmcli con modify my-office my-office ipv4.dhcp-hostname host-name
ipv6.dhcp-hostname host-name
```

Pour modifier l'ID client **IPv4** envoyé par un hôte à un serveur **DHCP**, modifier la propriété **dhcp-client-id** comme suit :

```
~]$ nmcli con modify my-office my-office ipv4.dhcp-client-id client-ID-
string
```

Il n'y a pas de propriété **dhcp-client-id** pour **IPv6**, le **dhclient** crée un identifiant pour **IPv6**. Voir la page man **dhclient(8)** pour plus d'informations.

Pour ignorer les serveurs **DNS** envoyés à un hôte par un serveur **DHCP**, modifier la propriété **ignore-auto-dns** comme suit :

```
~]$ nmcli con modify my-office my-office ipv4.ignore-auto-dns yes
ipv6.ignore-auto-dns yes
```

Voir la page man **nm-settings(5)** pour obtenir plus d'informations sur les propriétés et leurs paramètres de configurations.

Exemple 2.1. Configurer une connexion Ethernet dynamique par l'éditeur interactif

Pour configurer une connexion Ethernet dynamique par l'éditeur interactif, exécuter les commandes suivantes :

```
~]$ nmcli con edit type ethernet con-name ens3

===| nmcli interactive connection editor |===

Ajouter une nouvelle connexion  '802-3-ethernet'

Tapez 'help' ou '?' pour interroger les commandes disponibles.
Tapez 'describe [<setting>.<prop>]' pour obtenir une description de
propriété détaillée.

Vous pouvez modifier les paramètres suivants : connection, 802-3-
ethernet (ethernet), 802-1x, ipv4, ipv6, dcb
nmcli> describe ipv4.method

=== [method] ===
[NM property description]
IPv4 configuration method.  Si 'auto' est spécifié, alors la méthode
automatique  (DHCP, PPP, etc) qui convient sera utilisée par
l'interface, et la plupart des autres propriétés devront demeurer non
définies. Si 'link-local' est spécifié, alors une adresse de  lien-local
dans l'intervalle de valeurs 169.254/16 sera assignée à l'interface.  Si
'manual' est spécifié, l'adressage  IP statique est utilisé, et une
adresse IP au moins devra être donnée dans la propriété 'addresses' . Si
'shared' est spécifié (indiquant ainsi que cette connexion donnera un
accès réseau à d'autres machines) alors, l'interface reçoit une adresse
dans la plage de valeurs 10.42.x.1/24 et un DHCP, le serveur de
transfert DNS est démarré, et l'interface est en NAT-ed dans la
connexion réseau courante par défaut. 'disabled' signifie qu' IPv4 ne
sera pas utilisé pour cette connnexion. Cette propriété devra être
définie.

nmcli> set ipv4.method auto
nmcli> save
Sauvegarde de la connexion avec 'autoconnect=yes'. Peut résulter en
activation immédiate de la connexion.
Souhaitez-vous toujours sauvegarder ? [yes] yes
Connexion 'ens3' (090b61f7-540f-4dd6-bf1f-a905831fc287) sauvegardée.
nmcli> quit
~]$
```

> L'action par défaut est de conserver le profil de connexion persistant. Si nécessaire, le profil peut être contenu en mémoire uniquement, jusqu'au prochain démarrage, par la commande **save temporary**.

Ajouter une connexion Ethernet statique

Pour ajouter une connexion Ethernet ayant une configuration **IPv4** statique, utiliser une commande sous le format suivant :

```
nmcli connection add type ethernet con-name connection-name ifname
interface-name ip4 address gw4 address
```

IPv6. L'adresse et la passerelle peuvent être ajoutées en utilisant les options **ip6** et **gw6**.

Par exemple, voici une commande pour créer une connexion Ethernet statique avec l'adresse **IPv4** et une passerelle :

```
~]$ nmcli con add type ethernet con-name test-lab ifname ens9 ip4
10.10.10.10/24 \
gw4 10.10.10.254
```

En option, spécifier en même temps l'adresse **IPv6** et la passerelle pour le périphérique comme suit :

```
~]$ nmcli con add type ethernet con-name test-lab ifname ens9 ip4
10.10.10.10/24 \
gw4 10.10.10.254 ip6 abbe::cafe gw6 2001:db8::1
Connection 'test-lab' (05abfd5e-324e-4461-844e-8501ba704773) successfully
added.
```

NetworkManager définira son paramètre interne **ipv4.method** à **manual** et **connection.autoconnect** à **yes**. Le **NetworkManager** inscrira également des paramètres de config dans **/etc/sysconfig/network-scripts/ifcfg-my-office** avec la directive BOOTPROTO définie sur **none** et ONBOOT définie sur **yes**..

Notez que les changements manuels apportés au fichier ifcfg ne seront pas remarqués par le **NetworkManager** tant que l'interface n'est pas appelée à nouveau. Voir Section 1.9, « Configuration de réseau par les fichiers sysconfig » pour obtenir plus d'informations sur la façon d'utiliser les fichiers de configuration.

Pour définir deux adresses de serveur **IPv4 DNS** :

```
~]$ nmcli con mod test-lab ipv4.dns "8.8.8.8 8.8.4.4"
```

Notez que cela remplacera tous les serveurs **DNS** déjà définis. Pour définir deux adresses de serveur **IPv6 DNS** :

```
~]$ nmcli con mod test-lab ipv6.dns "2001:4860:4860::8888
2001:4860:4860::8844"
```

Notez que cela remplacera tous les serveurs **DNS** déjà définis. Pour ajouter deux serveurs **DNS** à un serveur déjà défini, utiliser le préfixe **+** comme suit :

```
~]$ nmcli con mod test-lab +ipv4.dns "8.8.8.8 8.8.4.4"
```

```
~]$ nmcli con mod test-lab +ipv6.dns "2001:4860:4860::8888
2001:4860:4860::8844"
```

Pour afficher la nouvelle connexion Ethernet, exécuter la commande suivante :

```
~]$ nmcli con up test-lab ifname ens9
Connection successfully activated (D-Bus active path:
/org/freedesktop/NetworkManager/ActiveConnection/6)
```

Pour vérifier le statut des périphériques et des connexions :

```
~]$ nmcli device status
DEVICE  TYPE      STATE      CONNECTION
ens3    ethernet  connected  my-office
ens9    ethernet  connected  test-lab
lo      loopback  unmanaged  --
```

Pour vérifier les informations détaillées sur la connexion configurée, veuillez exécuter la commande suivante :

```
~]$ nmcli -p con show test-lab
================================================================================
=====
                      Connection profile details (test-lab)
================================================================================
=====
connection.id:                          test-lab
connection.uuid:                        05abfd5e-324e-4461-844e-
8501ba704773
connection.interface-name:              ens9
connection.type:                        802-3-ethernet
connection.autoconnect:                 yes
connection.timestamp:                   1410428968
connection.read-only:                   no
connection.permissions:
connection.zone:                        --
connection.master:                      --
connection.slave-type:                  --
connection.secondaries:
connection.gateway-ping-timeout:        0[output truncated]
```

L'option **-p, --pretty** ajoute un titre de bannière et une séparation de section à la sortie.

Exemple 2.2. Configurer une connexion Ethernet statique par l'éditeur interactif

Pour configurer une connexion Ethernet statique par l'éditeur interactif, exécuter la commande suivante :

```
~]$ nmcli con edit type ethernet con-name ens3

===| nmcli interactive connection editor |===

Ajouter une nouvelle connexion  '802-3-ethernet'
```

```
Tapez 'help' ou '?' pour interroger les commandes disponibles.
Tapez 'describe [<setting>.<prop>]' pour obtenir une description de
propriété détaillée.

Vous pouvez modifier les paramètres suivants : connection, 802-3-
ethernet (ethernet), 802-1x, ipv4, ipv6, dcb
nmcli> set ipv4.routes 192.168.122.88/24
Souhaitez-vous également définir 'ipv4.method' à 'manual'? [yes]: yes
nmcli>
nmcli> save temporary
Sauvegarde de la connexion avec 'autoconnect=yes'. Peut résulter en
activation immédiate de la connexion.
Souhaitez-vous toujours sauvegarder ? [yes] no
nmcli> save
Souhaitez-vous toujours sauvegarder ? [yes] yes
Connexion 'ens3' (704a5666-8cbd-4d89-b5f9-fa65a3dbc916) sauvegardée.
nmcli> quit
~]$
```

L'action par défaut est de conserver le profil de connexion persistant. Si nécessaire, le profil peut être contenu en mémoire uniquement, jusqu'au prochain démarrage, par la commande **save temporary**.

Verrouiller un profil sur un périphérique particulier

Pour verrouiller un profil à une interface spécifique, les commandes utilisées dans les exemples ci-dessus incluent le nom de l'interface. Par exemple :

```
nmcli connection add type ethernet con-name connection-name ifname
interface-name
```

. Pour rendre un profil utilisable par toutes les interfaces Ethernet compatibles, exécuter une commande comme :

```
nmcli connection add type ethernet con-name connection-name ifname "*"
```

. Notez que vous devez utiliser l'argument **ifname** même si vous ne souhaitez pas définir une interface spécifique. Utilisez le caractère générique ***** pour spécifier que le profil puisse être utilisé avec n'importe quel périphérique compatible.

Pour verrouiller un profil à une adresse MAC particulière, utilisez une commande sous le format suivant :

```
nmcli connection add type ethernet con-name "connection-name" ifname "*"
mac 00:00:5E:00:53:00
```

Ajout d'une connexion Wi-Fi

Pour afficher les points d'accès Wi-Fi, exécutez une commande comme suit :

```
~]$ nmcli dev wifi list
  SSID            MODE  CHAN  RATE     SIGNAL  BARS  SECURITY
  FedoraTest      Infra  11    54 MB/s  98      ▂▄▆   WPA1
  Red Hat Guest   Infra  6     54 MB/s  97      ▂▄▆   WPA2
  Red Hat         Infra  6     54 MB/s  77      ▂▄▆_  WPA2 802.1X
* Red Hat         Infra  40    54 MB/s  66      ▂▄▆_  WPA2 802.1X
  VoIP            Infra  1     54 MB/s  32      ▂▄__  WEP
```

```
  MyCafe         Infra  11    54 MB/s  39         ▂▄▆_  WPA2
```

Pour créer un profil de connexion Wi-Fi avec une configuration **IP** statique, tout en autorisant l'attribution d'adresses **DNS** automatiques, exécutez une commande comme suit :

```
~]$ nmcli con add con-name MyCafe ifname wlan0 type wifi ssid MyCafe \
ip4 192.168.100.101/24 gw4 192.168.100.1
```

Pour définir un mot de passe WPA2, comme par exemple « caffeine », exécutez les commandes suivantes :

```
~]$ nmcli con modify MyCafe wifi-sec.key-mgmt wpa-psk
~]$ nmcli con modify MyCafe wifi-sec.psk caffeine
```

Voir le guide *Red Hat Enterprise Linux 7 Security Guide* pour obtenir des informations sur la sécurité des mots de passe.

Pour modifier l'état Wi-Fi, exécutez une commande sous le format suivant :

```
~]$ nmcli radio wifi [on | off ]
```

Changer une propriété spécifique

Pour vérifier une propriété particulière, comme **mtu**, exécutez une commande comme suit :

```
~]$ nmcli connection show id 'MyCafe' | grep mtu
802-11-wireless.mtu:                    auto
```

Pour modifier la propriété d'un paramètre de configuration, veuillez exécuter la commande suivante :

```
~]$ nmcli connection modify id 'MyCafe' 802-11-wireless.mtu 1350
```

Pour vérifier la modification, exécutez une commande comme suit :

```
~]$ nmcli connection show id 'MyCafe' | grep mtu
802-11-wireless.mtu:                    1350
```

Notez que le **NetworkManager** fait référence à des paramètres de configuration comme **802-3-ethernet** et **802-11-wireless** et **mtu** comme propriétés de configuration. Voir la page man **nm-settings(5)** pour plus d'informations sur les propriétés et leurs configurations.

2.3.3. Configuration des routages statiques par nmcli

Pour configurer des routages statiques avec l'outil **nmcli**, on doit utiliser le mode d'édition interactive ou l'outil en lignes de commandes.

Exemple 2.3. Configuration des routages statiques par nmcli

Pour configurer un routage statique pour une connexion Ethernet existante par l'outil en lignes de commandes, exécuter les commandes suivantes :

```
~]# nmcli connection modify eth0 +ipv4.routes "192.168.122.0/24
10.10.10.1"
```

Cela dirigera le trafic du sous-réseau **192.168.122.0/24** vers la passerelle **10.10.10.1**

Exemple 2.4. Configuration des routages statiques par l'éditeur nmcli

Pour configurer un routage statique pour une connexion Ethernet par l'éditeur interactif, exécuter les commandes suivantes :

```
~]$ nmcli con edit type ethernet con-name ens3

===| nmcli interactive connection editor |===

Ajouter une nouvelle connexion  '802-3-ethernet'

Tapez 'help' ou '?' pour interroger les commandes disponibles.
Tapez 'describe [<setting>.<prop>]' pour obtenir une description de
propriété détaillée.

Vous pouvez modifier les paramètres suivants : connection, 802-3-
ethernet (ethernet), 802-1x, ipv4, ipv6, dcb
nmcli> set ipv4.routes 192.168.122.0/24 10.10.10.1
nmcli>
nmcli> save persistent
Sauvegarde de la connexion avec 'autoconnect=yes'. Peut résulter en
activation immédiate de la connexion.
Souhaitez-vous toujours sauvegarder ? [yes] yes
Connexion 'ens3' (704a5666-8cbd-4d89-b5f9-fa65a3dbc916) sauvegardée.
nmcli> quit
~]$
```

2.4. UTILISATION DE L'INTERFACE EN LIGNE DE COMMANDES (CLI)

2.4.1. Configurer une interface de réseau en utilisant les fichiers ifcfg

Les fichiers de configuration d'interface contrôlent l'interface du logiciel pour les périphériques de réseaux individuels. Quand le système démarre, il utilise ces fichiers pour déterminer quelles interfaces mettre en place et comment les configurer. Ces fichiers sont généralement nommés **ifcfg-*name***, où le suffixe de *name* désigne le nom du périphérique qui contrôle le fichier de configuration. Par convention, le suffixe **ifcfg** du fichier est identique à la chaîne donnée par la directive **DEVICE** dans le fichier de configuration lui-même.

Paramètres de réseaux statiques

Pour configurer une interface avec des paramètres de réseaux statiques à l'aide des fichiers **ifcfg**, pour une interface nommée eth0, créez un fichier ayant pour nom **ifcfg-eth0** dans le répertoire **/etc/sysconfig/network-scripts /** comme suit :

```
DEVICE=eth0
BOOTPROTO=none
ONBOOT=yes
PREFIX=24
IPADDR=10.0.1.27
```

Spécifier le matériel ou l'adresse MAC en utilisant la directive **HWADDR**. Notez que cela peut influencer la procédure d'affectation de noms, comme expliqué dans Chapitre 8, *Nommage de périphériques réseaux consistante*. Vous n'avez pas besoin de spécifier le réseau ou l'adresse de diffusion; c'est calculée automatiquement par **ipcalc**.

Configuration de réseaux dynamiques

Pour configurer une interface avec des paramètres de réseaux dynamiques à l'aide des fichiers **ifcfg**, pour une interface nommée em1, créez un fichier ayant pour nom **ifcfg-em1** dans le répertoire **/etc/sysconfig/network-scripts /** comme suit :

```
DEVICE=em1
BOOTPROTO=dhcp
ONBOOT=yes
```

Spécifier le matériel ou l'adresse MAC en utilisant la directive **HWADDR**. Notez que cela peut influencer la procédure d'affectation de noms, comme expliqué dans Chapitre 8, *Nommage de périphériques réseaux consistante*.

Pour configurer une interface pour qu'elle envoie un nom d'hôte différent au serveur **DHCP**, ajouter la ligne suivante au fichier **ifcfg**.

```
DHCP_HOSTNAME=hostname
```

Pour configurer une interface permettant d'ignorer les routages envoyés par un serveur **DHCP**, ajoutez la ligne suivante au fichier **ifcfg**.

```
PEERDNS=no
```

. Cela empêchera le service réseau de mettre à jour **/etc/resolv.conf** avec les serveurs **DNS** reçus d'un serveur **DHCP**.

Pour configurer une interface qui puisse utiliser des serveurs **DNS** particuliers, définir **PEERDNS=no** comme décrit ci-dessus et ajouter les lignes ci-dessous dans votre fichier **ifcfg** :

```
DNS1=ip-address
DNS2=ip-address
```

avec *ip-address* comme adresse du serveur **DNS**. Cela amenera le service réseau à mettre à jour **/etc/resolv.conf** avec les serveurs **DNS** spécifiés.

NetworkManager appelera le client **DHCP** par défaut, **dhclient**, quand un profil a été défini pour pouvoir recevoir des adresses automatiquement, ou quand un fichier de configuration d'interface a BOOTPROTO défini à **dhcp**. Quand **DHCP** est exigé, une instance du **dhclient** sera démarrée pour chaque protocole Internet, **IPv4** ou **IPv6**, sur une interface. Quand le **NetworkManager** n'est pas en cours d'exécution, ou n'exécute pas une interface, le service de réseautage d'origine appelera des instances du **dhclient** selon les besoins.

Configuration d'un client DHCP

2.4.2. Configurer une interface de réseau en utilisant les commandes ip

L'utilitaire **ip** peut être utilisé pour assigner des adresses **IP** à une interface. La commande prend la forme suivante :

-

```
ip addr [ add | del ] address dev ifname
```

Assigner une adresse statique par les commande ip

Pour assigner une adresse **IP** à une interface, exécuter la commande en tant qu'utilisateur **root** comme suit :

```
~]# ip address add 10.0.0.3/24 dev eth0
The address assignment of a specific device can be viewed as follows:
~]# ip addr show dev eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state
UP qlen 1000
    link/ether f0:de:f1:7b:6e:5f brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.3/24 brd 10.0.0.255 scope global global eth0
       valid_lft 58682sec preferred_lft 58682sec
    inet6 fe80::f2de:f1ff:fe7b:6e5f/64 scope link
       valid_lft forever preferred_lft forever
```

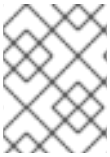
Vous trouverez d'autres exemples dans la page man **ip-address(8)**.

Configurer plusieurs adresses par les commandes ip

Comme l'utilitaire **ip** supporte l'attribution d'adresses à la même interface, il n'est plus très utile d'utiliser la méthode alias d'interface de liaison de plusieurs adresses à la même interface. La commande **ip** d'assignation d'adresse peut être répétée plusieurs fois pour pouvoir assigner plusieurs adresses. Par exemple :

```
~]# ip address add 192.168.2.223/24 dev eth1
~]# ip address add 192.168.4.223/24 dev eth1
~]# ip addr
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state
UP qlen 1000
    link/ether 52:54:00:fb:77:9e brd ff:ff:ff:ff:ff:ff
    inet 192.168.2.223/24 scope global eth1
    inet 192.168.4.223/24 scope global eth1
```

Les commandes de l'utilitaire **ip** sont documentées dans la page man **ip(8)**.

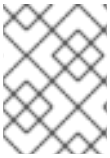


NOTE

Les commandes **ip** données en ligne de commandess ne seront pas persistantes après le redémarrage du système.

2.4.3. Routages statiques et Passerelle par défaut

Les routages statiques sont destinés au trafic qui ne doit pas, ou ne devrait pas, passer par la passerelle par défaut. Le routage est souvent géré par les périphériques qui se trouvent sur le réseau dédié au routage (bien qu'un périphérique peut être configuré pour effectuer un routage). Par conséquent, souvent, il n'est pas nécessaire de configurer des routages statiques sur les serveurs ou les clients Red Hat Enterprise Linux. Les exceptions incluent le trafic qui doit traverser par un tunnel VPN crypté ou le trafic qui doit suivre un routage spécifique pour des raisons de coût ou de sécurité. La passerelle par défaut est pour tout trafic non destiné au réseau local, et pour lequel aucun routage particulier n'est spécifié dans la table de routage. La passerelle par défaut est traditionnellement un routeur de réseau dédié.



NOTE

Afin de gagner en expertise, vous serez sans doute intéressé par le cours de formation Red Hat System Administration I (RH124).

Configuration des routages statiques par ligne de commandes

Si des routages statiques sont requis, ils peuvent être ajoutés à la table de routage par le biais de la commande **`ip route add`** ou enlevés par la commande **`ip route del`**. Les commandes **`ip route`** les plus fréquemment utilisées prennent la forme suivante :

```
ip route [add | del | change | append | replace ] destination-address
```

. Voir la page man **`ip-route(8)`** pour plus de détails sur les options et formats.

Utiliser la commande **`ip route`** sans options pour afficher la table de routage **`IP`**. Exemple :

```
~]$ ip route
default via 192.168.122.1 dev ens9  proto static  metric 1024
192.168.122.0/24 dev ens9  proto kernel  scope link  src 192.168.122.107
192.168.122.0/24 dev eth0  proto kernel  scope link  src 192.168.122.126
```

Pour ajouter un routage statique à une adresse hôte, ou autrement dit à une adresse **`IP`** unique, exécuter une commande en tant qu'utilisateur **`root`** :

```
ip route add 192.0.2.1 via 10.0.0.1 [dev ifname]
```

Quand *192.0.2.1* est l'adresse **`IP`** de l'hôte en notation décimale avec des points, *10.0.0.1* représente l'adresse du prochain tronçon et *ifname* l'interface de sortie qui mène au prochain tronçon.

Pour ajouter un routage statique à un réseau, ou autrement dit à une adresse **`IP`** représentant un groupe d'adresses **`IP`** , exécuter la commande suivante en tant qu'utilisateur **`root`** :

```
ip route add 192.0.2.0/24 via 10.0.0.1 [dev ifname]
```

avec *192.0.2.0* comme adresse **`IP`** du réseau de destination en notation décimale en points et avec */24* pour préfixe de réseau. Le préfixe de réseau correspond au nombre d'octets activés dans le msque du sous-réseau. Le format de l'adresse réseau / longueur du préfixe réseau ou *classless inter-domain routing* (CIDR).

La configuration du routage statique peut être stockée par interface dans un fichier **`/etc/sysconfig/network-scripts/route-`*`interface`***. Ainsi, les routages statiques de l'interface eth0 devraient être stockés dans le fichier **`fichier/etc/sysconfig/network-scripts/itinéraire-eth0`**. Le fichier **`route-`*`interface`*** a deux formats : arguments de commande **ip** et directives de réseau/masque réseau. Elles sont décrites ci-dessous.

Voir la page man **`ip-route(8)`** pour obtenir plus d'informations sur la commande **`ip route`**.

Configuration de la passerelle par défaut

La passerelle par défaut est déterminée par les scripts de réseau qui analysent le fichier **`/etc/sysconfig/network`** pour commencer, puis les fichiers **`ifcfg`** de l'interface de réseau pour les interfaces qui sont « up » (actives). Les fichiers **`ifcfg`** sont lus en ordre numérique croissant, et la dernière directive GATEWAY à lire est utilisée pour composer un routage par défaut dans la table de routage.

Le routage par défaut peut ainsi être indiqué par la directive GATEWAY et peut être spécifié soit globalement, soit dans les fichiers de configuration spécifiques à l'interface. Cependant, dans Red Hat Enterprise Linux l'utilisation du fichier global **/etc/sysconfig/network** est déprécié, et il faut maintenant indiquer la passerelle par l'intermédiaire des fichiers de configuration uniquement.

Dans les environnements de réseaux dynamiques, quand les hôtes mobiles sont gérés par un **NetworkManager**, l'information de passerelle a de grandes chances d'être spécifique à l'interface, et il vaut mieux qu'elle soit donnée par **DHCP**. Dans certains cas, quand il vaut mieux influencer la sélection du **NetworkManager** pour l'interface de sortie à utiliser pour atteindre une passerelle. Utiliser la commande **DEFROUTE=no** dans les fichiers **ifcfg** pour ces interfaces qui ne mènent pas à la passerelle par défaut.

2.4.4. Configuration des routages statiques par les fichiers ifcfg

Les configurations de routages statiques par les commandes **ip** par ligne de commandes seront perdus si le système est stoppé ou après un redémarrage. Pour configurer des routages statiques à être persistants après le redémarrage d'un système, ils devront figurer dans les fichiers de configurationpar-interface qui se trouvent à l'adresse suivante **/etc/sysconfig/network-scripts /**. Le nom du fichier doit être du format **route-*ifname***. Il existe deux types de commandes à utiliser dans les fichiers de configuration : les commandes **ip** comme expliqué dans Section 2.4.4.1, « Configuration des routages statiques en utilisant le format d'arguments de commandes IP » et le format *Réseau/Masque réseau* comme expliqué dans Section 2.4.4.2, « Format de directives de Réseau/Masque de réseau ».

2.4.4.1. Configuration des routages statiques en utilisant le format d'arguments de commandes IP

Si nécessaire, dans un fichier de configuration par interface, comme par exemple **/etc/sysconfig/network-scripts/route-eth0**, définir un itinéraire (routage) à une passerelle par défaut sur la première ligne. C'est seulement nécessaire si la passerelle n'est pas définie par **DHCP** et qu'elle n'est pas définie au niveau global dans le fichier **/etc/sysconfig/network** :

```
default via 192.168.1.1 dev interface
```

192.168.1.1 est l'adresse **IP** de la passerelle par défaut. L' *interface* est l'interface qui est connectée à, ou qui peut atteindre, la passerelle par défaut. L'option **dev** peut être omise, elle est facultative. Notez que ce paramètre prévaut sur un paramètre du fichier **fichier/etc/sysconfig/network**.

Si on a besoin d'un réseau distant, on peut spécifier un routage statique comme suit. Chaque ligne de routage est analysée individuellement.

```
10.10.10.0/24 via 192.168.1.1 [dev interface]
```

10.10.10.0/24 est l'adresse réseau et la longueur du préfixe du réseau de destination ou du réseau distant. L' adresse *192.168.1.1* est l'adresse **IP** menant au réseau distant. C'est, de préférence l'*adresse du tronçon suivant* (next hop address), mais l'adresse de l'interface de sortie fonctionnera. L'expression « next hop » désigne l'extrémité (la fin) d'un lien, par exemple une passerelle ou un routeur. L'option **dev** peut être utilisée pour spécifier l'interface de sortie *interface*, mais elle n'est pas nécessaire. Ajouter autant de routages statiques qu'il vous faut.

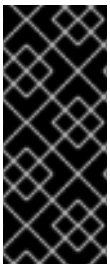
Voici un exemple de fichier **route-*interface*** qui utilise le format d'arguments de commandes **ip**. La passerelle par défaut est **192.168.0.1**, l'interface est eth0 et une ligne allouée ou une connexion WAN est disponible à l'adresse suivante **192.168.0.10**. Les deux routages statiques servent à joindre le réseau **10.10.10.0/24** et l'hôte **172.16.1.0/24** :

-

```
default via 192.168.0.1 dev eth0
10.10.10.0/24 via 192.168.0.10 dev eth0
172.16.1.10/32 via 192.168.0.10 dev eth0
```

Dans l'exemple ci-dessus, les paquets allant vers le réseau **192.168.0.0/24** seront dirigés en dehors de l'interface attachée à ce réseau. Les paquets allant vers le réseau **10.10.10.0/24** et vers l'hôte **172.16.1.10/32** seront dirigés vers **192.168.0.10**. Les paquets allant vers des paquets inconnus, distants utiliseront la passerelle par défaut, donc les routages statiques ne seront configurés uniquement que pour les réseaux ou hôtes distants si le routage par défaut ne convient pas. Dans ce contexte « distant » signifie tout réseau ou hôte qui n'est pas attaché directement au système.

Spécifier une interface de sortie est facultatif. Cela peut être utile si vous voulez forcer le trafic sur une interface spécifique. Par exemple, dans le cas d'un VPN, vous pouvez forcer le trafic vers un réseau distant pour passer à travers une interface tun0, même lorsque l'interface est dans un sous-réseau différent du réseau de destination.



IMPORTANT

Si la passerelle par défaut est déjà attribuée par le protocole **DHCP** et si la même passerelle avec le même métrique sont spécifiés dans un fichier de configuration, une erreur lors du démarrage ou lors de l'apparition de l'interface, se produira. Le message d'erreur suivant risque d'apparaître : "RTNETLINK answers: File exists". This error may be ignored.

2.4.4.2. Format de directives de Réseau/Masque de réseau

Vous pouvez également utiliser le format des directives de réseau/masque de réseau pour les fichiers **route-*interface***. Ce qui suit est un modèle pour le format réseau/masque de réseau, avec des instructions qui suivent :

```
ADDRESS0=10.10.10.0
NETMASK0=255.255.255.0
GATEWAY0=192.168.1.1
```

- **ADDRESS0=*10.10.10.0*** correspond à l'adresse de réseau du réseau distant ou de l'hôte à atteindre.
- **NETMASK0=*255.255.255.0*** correspond au masque de réseau de l'adresse de réseau définie dans **ADDRESS0=*10.10.10.0***.
- **GATEWAY0=*192.168.1.1*** est la passerelle par défaut, ou une adresse **IP** qui peut être utilisée pour atteindre **ADDRESS0=*10.10.10.0***

Voici un exemple de fichier **route-*interface*** qui utilise le format de directives de réseau/masque réseau. La passerelle par défaut est **192.168.0.1** mais une ligne allouée ou une connexion WAN est disponible à l'adresse suivante **192.168.0.10**. Les deux routages statiques servent à joindre les réseaux **10.10.10.0/24** et **172.16.1.0/24** :

```
ADDRESS0=10.10.10.0
NETMASK0=255.255.255.0
GATEWAY0=192.168.0.10
ADDRESS1=172.16.1.10
NETMASK1=255.255.255.0
GATEWAY1=192.168.0.10
```

Les routages statiques suivants sont numérotés séquentiellement, et ne doivent pas sauter une valeur. Par exemple, **ADDRESS0**, **ADDRESS1**, **ADDRESS2**, etc.

2.4.5. Configurer un VPN

IPsec, fourni par **Libreswan**, est la méthode préférée de création de VPN dans Red Hat Enterprise Linux 7. Configurer un VPM IPsec en ligne de commandess est documenté dans le guide *Red Hat Enterprise Linux 7 Security Guide*.

2.5. UTILISER LE NETWORKMANAGER AVEC L'INTERFACE GRAPHIQUE GNOME

Dans Red Hat Enterprise Linux 7, le **NetworkManager** n'a pas sa propre interface graphique (GUI). L'icône de connexion au réseau qui se trouve en haut et à droite du bureau est fourni comme faisant partie du GNOME Shell et l'outil de configuration des paramètres de **Réseau** est fourni comme faisant partie du GUI **control-center** de GNOME. L'ancien GUI **nm-connection-editor** est toujours disponible pour certaines tâches.

2.5.1. Se connecter à réseau par un GUI

Il existe deux manières d'accéder à la fenêtre de configuration du **Réseau** de l'application **control-center** :

- Appuyer sur la clé **Super** pour accéder au menu Activités, saisir **control network**, comme on le voit dans Figure 2.2, « L'utilitaire de réseau sélectionné dans GNOME » puis, appuyer sur la touche **Entrée**. L'outil de configuration **Réseau** apparaîtra. Continuer avec Section 2.5.2, « Configurer les nouvelles connexions et modifier les connexions existantes ».

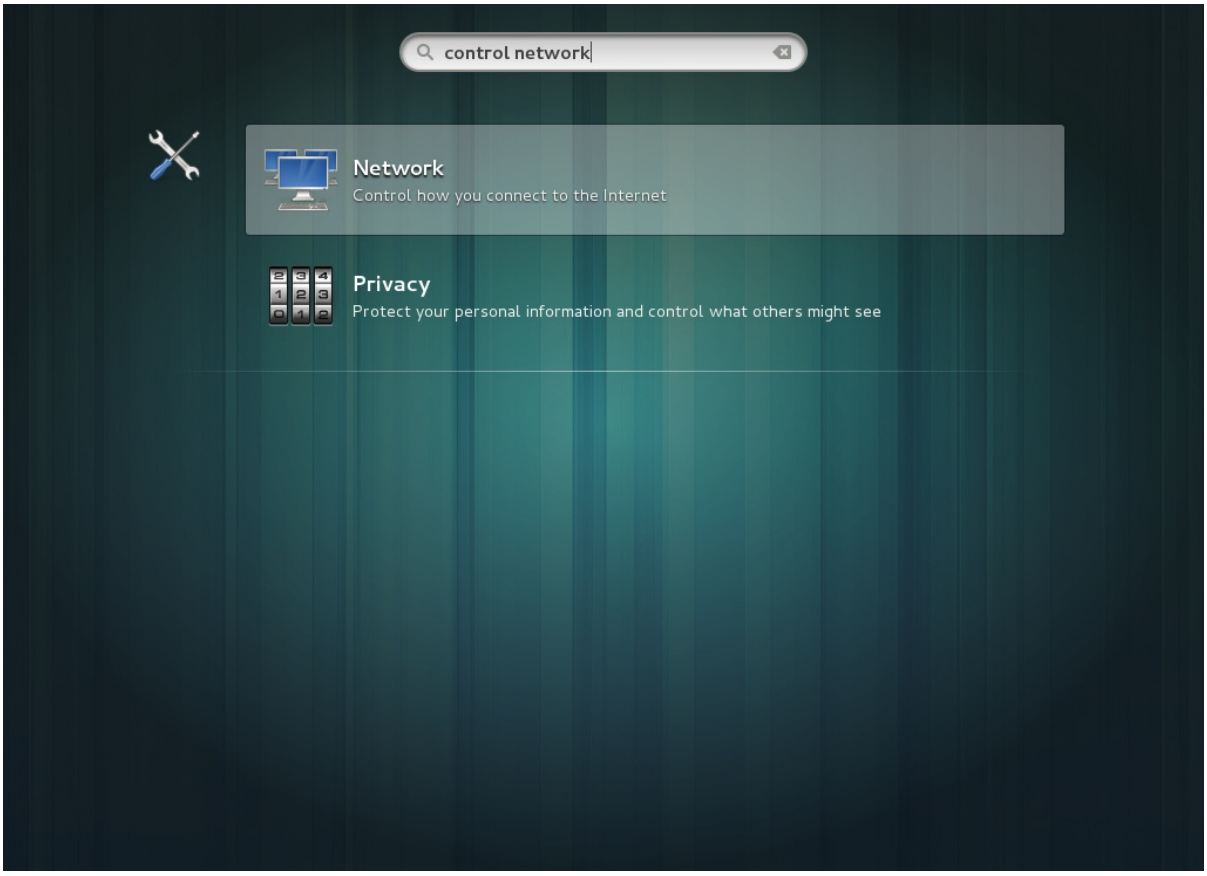


Figure 2.2. L'utilitaire de réseau sélectionné dans GNOME

- Cliquez sur l'icône de connexion du réseau du GNOME Shell qui se trouve dans le coin en haut et à droite de l'écran pour ouvrir son menu.

Quand vous cliquez sur l'icône de connexion du réseau du GNOME Shell, vous apercevrez :

- une liste des réseaux catégorisés auxquels vous êtes actuellement connectés (comme **Wired** ou **Wi-Fi**);
- une liste de tous les **Réseaux disponibles** que le **NetworkManager** aura détectés.
- Configurer un réseau privé virtuel (VPN)
- une option pour sélectionner l'option de menu **Configuration Réseau**.

Si vous êtes connectés à un réseau, c'est indiqué par le bouton symbolique **ON**. En cliquant n'importe où au niveau du bouton permutera l'état du bouton. Si vous passez de ON à OFF, vous vous disconnecterez de cette connexion réseau.

Cliquer sur **Configuration Réseau**. L'outil de configuration de **Réseau** apparaîtra, Continuez avec Section 2.5.2, « Configurer les nouvelles connexions et modifier les connexions existantes ».

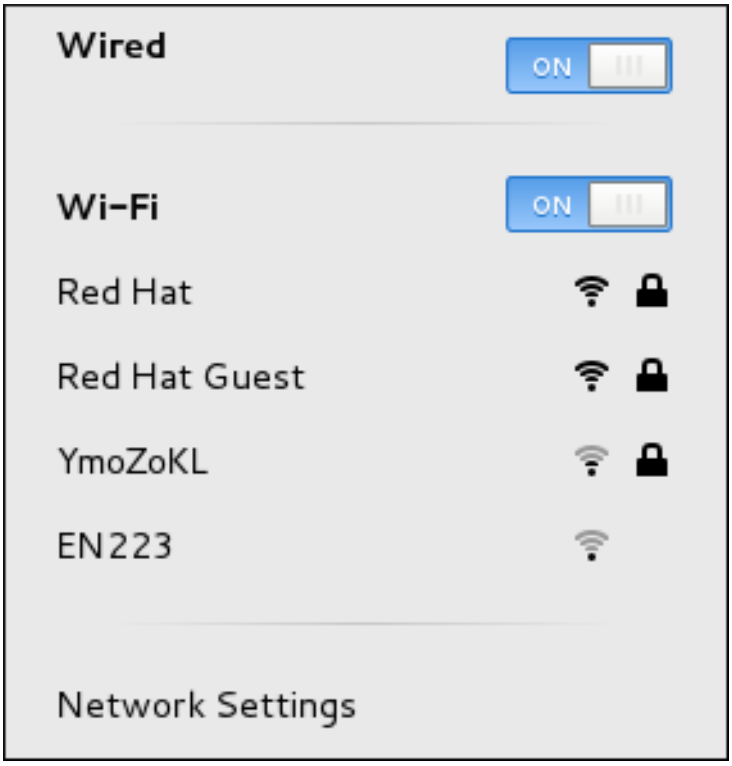


Figure 2.3. Le menu de réseau GNOME, qui montre toutes les réseaux disponibles ou auxquels on est connectés

2.5.2. Configurer les nouvelles connexions et modifier les connexions existantes

La fenêtre de configuration de **Réseau** vous montre le statut de connexion, son type, son interface, son adresse **IP**, ses infos de routage, etc..

Figure 2.4. Configurer les réseaux par la fenêtre de configuration de réseau

La fenêtre de configuration du **Réseau** a un menu sur le côté gauche, montrant les interfaces ou les périphériques de réseau disponibles. Ceci inclut les interfaces logicielles comme les VLAN, ponts, bonds et équipes. Sur la droite, les *profils de connexion* sont indiqués pour l'interface ou le périphérique réseau sélectionné. Un profil est une collection nommée de paramètres de configuration qui peuvent s'appliquer à une interface. Vous verrez dessous un bouton avec le signe plus et le signe moins pour

l'ajout et la suppression de nouvelles connexions réseau, et à droite un icône de roue dentée pour éditer les détails du périphérique réseau sélectionné ou de connexion VPN. Pour ajouter une nouvelle connexion, cliquez sur le signe plus pour ouvrir la fenêtre **Ajouter une connexion réseau** et continuer avec Section 2.5.2.1, « Configuration d'une nouvelle connexion ».

Modifier une connexion existante

En cliquant sur l'icône de roue dentée d'un profil de connexion existant de la fenêtre de configuration du **Réseau** ouvre la fenêtre de détails de **Réseau**, où vous pouvez effectuer la plupart des tâches de configuration réseau, telles que l'adressage **IP**, **DNS** ou la configuration de routage.

Figure 2.5. Configurer les réseaux par la fenêtre des détails de connexion réseau

2.5.2.1. Configuration d'une nouvelle connexion

Dans la fenêtre **Réseau**, cliquer sur le signe plus qui se situe sous le menu pour ouvrir la fenêtre **Ajouter une connexion réseau**. Cela affichera une liste de types de connexions qui peuvent être ajoutés.

Puis, pour configurer :

- **Connexions VPN**, cliquer sur l'entrée **VPN** et continuez avec Section 2.5.7, « Établir une connexion VPN »;
- **Connections Bond**, cliquer sur l'entrée **Bond**et continuez avec Section 4.6.1, « Établir une connexion de liaison »;
- **Connexions Bridge**, cliquez sur l'entrée **Bridge** et continuez avec Section 6.4.1, « Établir une connexion de pontage »;
- **Connexions VLAN**, cliquez sur l'entrée **VLAN** et continuez avec Section 7.5.1, « Établir une connexion VLAN »; ou,
- **Connexions Équipe**, cliquez sur l'entrée **Équipe** et continuez avec Section 5.13, « Créer un Network Team par l'interface graphique (GUI) ».

2.5.3. Se connecter à un réseau automatiquement

Pour chaque type de connexion que vous ajoutez ou configurez, vous pouvez décider si le **NetworkManager** doit tenter de se connecter à ce réseau automatiquement ou non.

Procédure 2.1. Configurer le NetworkManager pour qu'il se connecte à un réseau automatiquement quand il est détecté

1. Appuyer sur la clé **Super** pour accéder au menu Activités, saisir **control network**, et appuyez sur la touche **Entrée**. L'outil de configuration du **Réseau** apparaîtra.

2. Sélectionner l'interface de réseau à partir du menu sur la gauche.

3. Cliquez sur l'icône de roue dentée d'un profil de connexion dans le menu de droite. Si vous avez un seul profil associé à l'interface sélectionnée, l'icône de roue dentée sera dans le coin en bas à droite. La fenêtre contenant les détails de **Réseau** apparaîtra.

4. Sélectionnez l'entrée de menu **Identité** sur la gauche. La fenêtre **Réseau** passera à la vue Identité.

5. Sélectionnez **Connection automatique** pour que le **NetworkManager** s'auto-connecte quand **NetworkManager** détecte que la connexion est disponible. Décochez la case si vous ne souhaitez pas que le **NetworkManager** se connecte automatiquement. Si la case reste vide, vous devrez sélectionner cette connexion manuellement dans le menu d'icône de connexion de réseau pour que la connnexion ait lieu.

2.5.4. Profils de connexions privées ou sur tout le système

Le **NetworkManager** stocke tous les *profils de connexion*. Un profil est une collection de paramètres avec un nom, qui peut être appliquée à une interface. Le **NetworkManager** stocke ces profils de connexion pour un usage systémique (*connexions système*), ainsi que de tous les profils de *connexion utilisateur*. L'accès aux profils de connexion est contrôlé par les autorisations stockées par le **NetworkManager**. Voir la page man **nm-settings(5)** pour plus d'informations sur la propriété **permissions** du paramètre **connexion**. Les autorisations correspondent à la directive **USERS** qui se trouve dans les fichiers **ifcfg**. Si la directive **USERS** n'est pas présente, le profil de réseau sera disponible à tous les utilisateurs. À titre d'exemple, la commande suivante se trouvant dans un fichier **ifcfg** rendra la connexion disponible uniquement aux utilisateurs listés :

```
USERS = "joe bob alice"
```

. Cela peut également être défini à l'aide d'outils d'interface utilisateur graphique. Dans **nm-connection-editor**, il y a la case à cocher **Tous les utilisateurs peuvent se connecter à ce réseau** qui se trouve dans l'onglet **Général**, et dans la fenêtre Identité de la configuration de réseau **- control-center** de GNOME, il y a la case à cocher **Rendre disponible aux autres utilisateurs**.

La stratégie par défaut du **NetworkManager** est de permettre à tous les utilisateurs de créer et de modifier des connexions dans l'ensemble du système. Les profils qui doivent être disponibles au moment du démarrage ne peuvent être privés, parce qu'ils ne seront pas visibles tant que l'utilisateur ne se connecte pas. Par exemple, si l'utilisateur **user** crée un profil de connexion **user-em2** avec la case à cocher **Se connecter automatiquement** sélectionnée, mais avec la case **Rendre disponible aux autres utilisateurs** non sélectionnée, la connexion ne sera pas disponible au moment du démarrage.

Pour limiter les connexions et le réseautage, il y a deux options qui peuvent être utilisées soit seules, soit en conjonction.

- Décochez la case **Rendre disponible à tous les utilisateurs**, ce qui rendra la connexion modifiable et utilisable par l'utilisateur ayant effectué les changements uniquement.

- Utiliser le framework **polkit** pour limiter les permissions les opérations de réseau en général sur la base utilisateur.

Si vous utilisez ces deux options en même temps, vous obtiendrez un niveau de sécurité et de contrôle plus précis de réseautage. Voir la page man **polkit(8)** pour obtenir plus d'informations sur **polkit**.

Notez que les connexions VPN sont **toujours** créées comme étant privées-par-utilisateur, puisqu'elles sont sensées être plus privées qu'une connexion Wi-Fi ou Ethernet.

Procédure 2.2. Comment modifier une connexion pour qu'elle devienne spécifique utilisateur, plutôt que de s'appliquer à tout le système, et vice versa

Selon la stratégie suystème, vous aurez sans doute besoin de privilèges root sur le système pour pouvoir effectuer les changements pour qu'une connexion soit 'spécifique utilisateur' ou 'système'.

1. Appuyer sur la clé **Super** pour accéder au menu Activités, saisir **control network**, et appuyez sur la touche **Entrée**. L'outil de configuration du **Réseau** apparaîtra.

2. Sélectionner l'interface de réseau à partir du menu sur la gauche.

3. Cliquez sur l'icône de roue dentée d'un profil de connexion dans le menu de droite. Si vous avez un seul profil associé à l'interface sélectionnée, l'icône de roue dentée sera dans le coin en bas à droite. La fenêtre contenant les détails de **Réseau** apparaîtra.

4. Sélectionnez l'entrée de menu **Identité** sur la gauche. La fenêtre **Réseau** passera à la vue Identité.

5. Sélectionnez la case **Rendre disponible à tous les utilisateurs** pour que le **NetworkManager** puisse rendre la connexion disponible sur tout le système.

   À l'inverse, décochez la case **Rendre disponible à tous les utilisateurs** pour rendre la connexion spécifique à l'utilisateur.

2.5.5. Configuration d'une connexion filaire (Ethernet)

Pour configurer une connexion réseau filaire, appuyez sur la touche **Super** pour saisir Activités, puis saisir **control network** et appuyez sur la touche **Enter**. L'outil de configuration de **Réseau** apparaîtra.

Séelctionnez l'interface de réseau **Wired** dans le menu sur la gauche, si ce n'est pas déjà surligné.

Le système crée et configure un *profil de connexion* unique appelé **Wired** par défaut. Un profil est une collection de paramètres avec un nom, pouvant être appliqué à une interface. Plusieurs profils peuvent être créés pour une interface donnée pouvant être appliqués selon les besoins. Le profil par défaut ne peut pas être supprimé, mais ses paramètres peuvent être modifiés. Vous pouvez modifier le profil **Wired** par défaut en cliquant sur l'icône de roue dentée. Vous pouvez créer un nouveau profil de connexion filaire (wired) en cliquant sur le bouton **Ajouter Profil**. Les profils de connexion associées à une interface sélectionnée seront affichés dans le menu de droite.

Lorsque vous ajoutez une nouvelle connexion en cliquant sur le bouton **Add Profile**, le **NetworkManager** crée un nouveau fichier de configuration pour la connexion et ouvre la même boîte de dialogue que celle qui est utilisée pour modifier une connexion existante. La différence entre ces boîtes

de dialogue, c'est qu'un profil de connexion existant a une entrée de menu **Détails** et **Réinitialiser**. En effet, vous êtes toujours en train de modifier un profil de connexion ; la différence réside seulement dans la question de savoir si cette connexion existait avant ou vient d'être créée par le **NetworkManager** lorsque vous avez cliqué sur **Ajouter Profil**.

2.5.5.1. Configurer le Nom de connexion, le Comportement Auto-Connect, et la Disponiblité

De nombreuses configurations de la boîte de dialogue **Modifier** sont communes à tous les types de connexion. Voir **Identité** (ou l'onglet **Général** si vous utilisez **nm-connection-editor**) :

- **Nom** — saisir un nom descriptif de connexion réseau. Ce nom pourra être utilisé pour noter cette connexion dans le menu de la fenêtre **Réseau**.
- **Adresse MAC** — sélectionner l'adresse MAC de l'interface à laquelle ce profil doit s'appliquer.
- **Adresse clonée** — si nécessaire, saisir une autre adresse MAC à utiliser.
- **MTU** — si nécessaire, saisir une *maximum transmission unit* (MTU) à utiliser. La valeur MTU représente la taille (en octets) du plus grand paquet que le link-layer puisse transmettre. La valeur par défaut est **1500** et n'a généralement pas besoin d'être modifiée.
- **Zone de parefeu** — si nécessaire, sélectionnez une autre zone de parefeu à appliquer. Voir le guide *Red Hat Enterprise Linux 7 Security Guide* pour obtenir plus d'informations sur les zones de parefeux.
- **Se connecter automatiquement** — sélectionnez cette case si vous souhaitez que le **NetworkManager** s'auto-connecte à cette connexion quand elle sera disponible. Voir Section 2.5.3, « Se connecter à un réseau automatiquement » pour plus d'informations.
- **Rendre disponible à tous les utilisateurs** — sélectionnez cette case pour créer une connexion disponible à tous les utilisateurs sur le système. Changer ce paramètre peut nécessiter des privilèges d'utilisateur root. Consulter Section 2.5.4, « Profils de connexions privées ou sur tout le système » pour obtenir plus d'informations.
- **Se connecter automatiquement au VPN quand on utilise cette connexion** — sélectionnez cette case si vous souhaitez que le **NetworkManager** s'auto-connecte à cette connexion de VPN quand ce profil de connexion est connecté. Sélectionner le VPN à partir du menu déroulant.

Sauvegarder votre nouvelle connexion (ou votre connexion modifiée) et faire des configurations suplémentaires
Une fois vous aurez terminé de modifier votre connexion filaire (wired) au réseau local virtuel, cliquez sur le bouton **Apply** pour enregistrer votre configuration personnalisée. Si le profil était en cours d'utilisation lors de la modification, alimentez le cycle de connexion pour que le **NetworkManager** applique les modifications. Si le profil est désactivé (OFF), réglez-le sur ON ou sélectionnez-le dans le menu de l'icône de connexion réseau. Voir Section 2.5.1, « Se connecter à réseau par un GUI » pour plus d'informations sur l'utilisation de votre connexion nouvelle ou modifiée.

Vous pouvez configurer davantage une connexion existante en la sélectionnant dans la fenêtre **Réseau** et en cliquant sur l'icône de roue dentée pour retourner à la boîte de dialogue d'édition.

Puis, pour configurer :

- **port-based Network Access Control (PNAC)**, cliquez sur l'onglet **802.1X Security** et continuez avec Section 2.5.10.1, « Configuration de la sécurité 802.1X »;
- Paramètres de configuration **IPv4** pour la connection, cliquer sur l'onglet **IPv4 Settings** et continuer avec Section 2.5.10.4, « Configuration des paramètres IPv4 »; ou,
- Paramètres de configuration **IPv6** pour la connexion, cliquer sur l'onglet **IPv6 Settings** et continuez avec Section 2.5.10.5, « Configurer les paramètres IPv6 ».

2.5.6. Configurer une connexion Wi-Fi

Cette section explique comment utiliser le **NetworkManager** pour configurer une connexion Wi-Fi (également connue sous le nom wireless ou 802.11*a/b/g/n*) vers un Point d'accès.

Pour configurer une connexion à haut débit mobile (comme 3G), voir Section 2.5.8, « Établir une connexion à haut débit mobile ».

Comment se connecter rapidement à un point d'accès disponible

Pour vous connecter à un point d'accès disponible, la méthode la plus simple consiste à cliquer sur l'icône de connexion réseau pour activer le menu de l'icône de connexion réseau. Recherchez *Service Set Identifier* (SSID) du point d'accès dans la liste des réseaux **Wi-Fi** et cliquez dessus. Un symbole sous forme de cadenas indique le point d'accès qui requiert une authentification. Si le point d'accès est sécurisé, une boîte de dialogue vous invite à fournir une clé d'authentification ou un mot de passe.

Le **NetworkManager** essaie de détecter automatiquement le type de sécurité utilisé par le point d'accès. S'il y a plusieurs possibilités, le **NetworkManager** devine le type de sécurité et le présente dans le menu déroulant **sécurité Wi-Fi**. Pour la sécurité WPA-PSK (WPA avec une phrase de passe), aucun autre choix n'est nécessaire. Pour WPA Enterprise (802. 1 X), vous devez sélectionner la sécurité spécifiquement, car elle ne peut pas être détectée automatiquement. Si vous n'êtes pas sûr, essayez de vous connecter à chaque type l'un après l'autre. Enfin, entrez la clé ou la phrase de passe dans le champ **Mot de passe**. Certains types de mot de passe, comme la clé 40-bit WEP ou la clé WPA de 128-bits, ne sont pas valides s'ils ne sont pas de la longueur requise. Le bouton **Connexion** restera inactif jusqu'à ce que vous entriez une clé de la longueur requise pour le type de sécurité sélectionné. Pour en savoir plus sur la sécurité sans fil, voir Section 2.5.10.2, « Configurer la sécurité Wi-Fi ».

Si le **NetworkManager** se connecte à un point d'accès, l'icône de connexion au réseau se transformera en indicateur graphique correspondant à la force du signal de connexion sans fil.

Vous pouvez également modifier les paramètres d'une de ces connexions de point d'accès créées automatiquement, comme si vous l'aviez ajoutée vous-même. La page **Wi-Fi** de la fenêtre **Réseau** dispose d'un bouton **Historique**. En cliquant dessus, vous apercevrez une liste de toutes les connexions auxquelles vous avez essayé de vous connecter. Voir Section 2.5.6.2, « Modifier une connexion ou en créer une toute nouvelle »

2.5.6.1. Se connecter à un réseau Wi-Fi masqué

Tous les ponts d'accès ont un *Service Set Identifier* (SSID) pour les identifier, mais un point d'accès peut être configuré de façon à ne pas transmettre son SSID, dans lequel cas, il sera *masqué*, et il n'apparaîtra pas dans la liste du **NetworkManager** de réseaux **Disponible**. Vous pourrez toujours vous connecter à un point d'accès qui cache son SSID si vous connaissez son SSID, sa méthode d'authentification, et ses secrets.

Pour vous connecter à un réseau sans fil masqué, appuyez sur la touche **Super** pour afficher la Vue d'ensemble des activités, tapez **réseau de contrôle** et appuyez sur **entrée**. La fenêtre **réseau** apparaîtra. Sélectionnez **Wi-Fi** dans le menu, puis sélectionnez **Se connecter au réseau caché**

de causer une boîte de dialogue apparaît. Si vous êtiez connecté à un réseau caché, utilisez la **connexion** déroulante pour le sélectionner et cliquez sur **Se connecter**. Sinon, laissez la **Connexion** du menu déroulant à **Nouvelle**, entrez le SSID du réseau masqué, sélectionnez sa méthode de **sécurité Wi-Fi**, saisir les secrets d'authentification corrects et cliquez sur **Se connecter**.

Pour obtenir davantage d'informations sur la configuration de la sécurité sans fil, voir Section 2.5.10.2, « Configurer la sécurité Wi-Fi ».

2.5.6.2. Modifier une connexion ou en créer une toute nouvelle

Vous pouvez modifier une connexion existante que vous avez essayée ou à laquelle vous avez réussi à vous connecter à dans le passé, en ouvrant la page **Wi-Fi** de la boîte de dialogue **Réseau**, et en cliquant sur l'icône de roue dentée à droite du nom de connexion Wi-Fi. Si le réseau n'est pas actuellement dans la plage donnée, cliquez sur **Historique** pour afficher les dernières connexions tentées. Lorsque vous cliquerez sur l'icône de roue dentée, la boîte de dialogue Édition de connexion apparaîtra. La fenêtre **Détails** montre les détails des informations de connexion.

Pour configurer une nouvelle connexion dont le SSID est à portée, tentez, tout d'abord, de vous connecter en ouvrant la fenêtre **Réseau**, en sélectionnant l'entrée de menu **Wi-Fi**, et en cliquant sur le nom de la connexion (par défaut, le même que le SSID). Si le SSID n'est pas dans la plage proposée, voir Section 2.5.6.1, « Se connecter à un réseau Wi-Fi masqué ». Si le SSID est dans la plage proposée, la procédure est la suivante :

1. Appuyer sur la clé **Super** pour accéder au menu Activités, saisir **control network**, et appuyez sur la touche **Entrée**. L'outil de configuration du **Réseau** apparaîtra.

2. Sélectionner l'interface **Wi-Fi** dans le menu de gauche.

3. Cliquez sur le profil de connexion Wi-Fi auquel vous souhaitez vous connecter dans le menu de droite. Un symbole sous forme de cadenas indiquera si une clé ou un mot de passe sont exigés.

4. Si on vous le demande, saisir les détails d'authentification.

Configurer le SSID, le Comportement Auto-Connect, et la Disponiblité
Pour modifier les paramètres de configuration, sélectionner **Wi-Fi** dans la page **Network**, puis sélectionner l'icône de roue dentée à droite du nom de connexion Wi-Fi. Sélectionner **Identity**. Vous verrez les paramètres suivants :

SSID

Le *Service Set Identifier* (SSID) du point d'accès, en anglais Access Point (AP).

BSSID

Le premier *Basic Service Set Identifier* (BSSID) correspond à l'adresse MAC ou *adresse de matériel* du point d'accés sans fil particulier auquel vous êtes connecté quand vous êtes en mode **Infrastructure**. Ce champ reste vide par défaut, et vous pouvez vous connecter à un point d'accès sans fil par **SSID** sans avoir besoin d'indiquer son **BSSID**. Si le BSSID est indiqué, cela forcera le système à associer vers un point d'accès spécifique uniquement.

Pour les réseaux ad-hoc, le **BSSID** est généré au hasard par le sous-système **mac80211** quand le réseau ad-hoc est créé. Il n'est pas affiché dans le **NetworkManager**

Adresse MAC

Sélectionner l'adresse MAC ou l'*adresse de matériel* de l'interface Wi-Fi à utiliser.

Un système peut avoir une ou plusieurs cartes réseau sans fil connectées. Le champ **adresse MAC** permet donc d'associer une carte sans fil spécifique à une connexion spécifique (ou à des connexions).

Adressse clonée

Adresse MAC clonée à utiliser à la place de la véritable adresse de matériel. Laisser vide à moins qu'on vous indique autrement.

Les paramètres de configuration suivants sont communs à tous les profils de connexion :

- **Se connecter automatiquement** — sélectionnez cette case si vous souhaitez que le **NetworkManager** s'auto-connecte à cette connexion quand elle sera disponible. Voir Section 2.5.3, « Se connecter à un réseau automatiquement » pour plus d'informations.
- **Rendre disponible à tous les utilisateurs** — sélectionnez cette case pour créer une connexion disponible à tous les utilisateurs sur le système. Changer ce paramètre peut nécessiter des privilèges d'utilisateur root. Consulter Section 2.5.4, « Profils de connexions privées ou sur tout le système » pour obtenir plus d'informations.

Sauvegarder votre nouvelle connexion (ou votre connexion modifiée) et faire des configurations suplémentaires
Une fois vous aurez terminé de modifier votre connexion sans fil, cliquez sur le bouton **Apply** pour enregistrer votre configuration. Si vous avez une configuration correcte, vous pouvez vous connecter à votre connexion modifiée en la sélectionnant dans le menu d'icones de connexion de réseau. Voir Section 2.5.1, « Se connecter à réseau par un GUI » pour plus d'informations sur l'utilisation de votre connexion nouvelle ou modifiée.

Vous pouvez configurer davantage une connexion existante en la sélectionnant dans la fenêtre **Réseau** et en cliquant sur l'icône de roue dentée pour révéler les informations de connexion.

Puis, pour configurer :

- **Authentification de sécurité** pour la connexion sans fil, cliquer sur **Security** et allez dans Section 2.5.10.2, « Configurer la sécurité Wi-Fi » ;
- Configurations **IPv4** de la connexion, cliquer sur **IPv4** et allez dans Section 2.5.10.4, « Configuration des paramètres IPv4 » ; ou bien,
- Configuration **IPv6** de la connexion, cliquer sur **IPv6**et continuez avec Section 2.5.10.5, « Configurer les paramètres IPv6 ».

2.5.7. Établir une connexion VPN

IPsec, fourni par **Libreswan**, est la méthode préférée de création de VPN dans Red Hat Enterprise Linux 7. L'outil d'interface utilisateur GNOME décrit ci-dessous exige le paquet NetworkManager-libreswan-gnome. Si besoin est, pour vous assurer que le paquet est bien installé, exécutez la commande en tant qu'utilisateur **root** :

```
~]# yum install NetworkManager-libreswan-gnome
```

Voir *Red Hat Enterprise Linux 7 System Administrator's Guide* pour obtenir plus d'informations sur la façon d'installer les nouveaux packages dans Red Hat Enterprise Linux 7.

Établir un réseau privé virtuel (VPN) qui vous permettra de communiquer entre votre réseau local (LAN)

et un autre réseau LAN à distance. Cela se fait en mettant en place un tunnel à travers un réseau intermédiaire, tel qu'Internet. Le tunnel VPN qui est mis en place en général utilise l'authentification et le chiffrement. Après avoir avec succès établi une connexion VPN à l'aide d'un tunnel sécurisé, une passerelle ou un routeur VPN exécute les actions suivantes sur les paquets que vous transmettrez :

1. il ajoute un *En-tête d'authentification* pour le routage et l'authentification :
2. il codifie les données du paquet; et,
3. il enferme les données dans des paquets suivant le protocole ESP (de l'anglais Encapsulating Security Payload) pour le décodage et la gestion des directives.

Le routeur du VPN récepteur supprime les informations d'en-tête, déchiffre les données et les achemine vers leur destination prévue (un poste de travail ou autre nœud sur un réseau). En utilisant une connexion de réseau à réseau, le nœud récepteur du réseau local reçoit les paquets déjà décodées et prêts à être traités. Le processus de décodage et de chiffrement dans une connexion VPN de réseau à réseau est donc transparent pour les clients.

Comme ils utilisent plusieurs couches d'authentification et de décodage, les VPN représentent un moyen sécurisé et efficace pour connecter plusieurs noeuds distants entre eux en intranet unifié.

Procédure 2.3. Ajout d'une nouvelle connexion VPN

Vous pouvez configurer une connexion VLAN en ouvrant la fenêtre **Réseau**, et en sélectionnant le signe plus sous le menu.

1. Appuyer sur la clé **Super** pour accéder au menu Activités, saisir **control network**, et appuyez sur la touche **Entrée**. L'outil de configuration du **Réseau** apparaîtra.
2. Sélectionner le symbole plus sous le menu. La fenêtre **Ajouter une connexion réseau** apparaitra.
3. Sélectionner l'entrée **VPN** dans le menu. La vue vous donne maintenant la possibilité de configurer un VPN manuellement, ou d'importer un fichier de configuration de VPN.

   Le plug-in de VPN du **NetworkManager** du type de VPN que vous souhaitez configurer doit être installé. Voir Section 2.5.7, « Établir une connexion VPN ».
4. Cliquer sur le bouton **Ajouter** pour ouvrir l'assistant **Sélectionner un type de connexion de VPN**.
5. Sélectionner le protocole de VPN de la passerelle à laquelle vous vous connectez dans le menu. Les protocoles de VPN disponibles du menu correspondent aux plug-ins de VPN du **NetworkManager** installés. Voir Section 2.5.7, « Établir une connexion VPN ».
6. La fenêtre **Ajouter une connexion réseau** change pour afficher les paramètres personnalisés pour le type de connexion de VPN que vous avez sélectionné dans l'étape précédente.

Procédure 2.4. Modifier une connexion VPN existante

Vous pouvez configurer une connexion VPN existante en ouvrant la fenêtre **Réseau**, et en sélectionnant le nom de la connexion dans la liste. Puis, cliquer sur le bouton **Modifier**.

1. Appuyer sur la clé **Super** pour accéder au menu Activités, saisir **control network**, et appuyez sur la touche **Entrée**. L'outil de configuration du **Réseau** apparaîtra.

2. Sélectionner la connexion **VPN** que vous souhaitez modifier dans le menu de gauche.

3. Cliquer sur le bouton **Configurer**.

Configurer le Nom de connexion, le Comportement Auto-Connect, et la Disponiblité

Il existe cinq configurations de la boîte de dialogue **Modifier** qui sont communes à tous les types de connexion. Voir l'onglet **Général** :

- **Nom de connexion** — saisir un nom descriptif pour votre connexion de réseau. Ce nom sera utilisé pour lister cette connexion dans le menu de la fenêtre **Réseau**.

- **Se connecter automatiquement à ce réseau quand il est disponible** — sélectionnez cette case si vous souhaitez que le **NetworkManager** s'auto-connecte à cette connexion quand elle est disponible. Voir Section 2.5.3, « Se connecter à un réseau automatiquement » pour plus d'informations.

- **Rendre le réseau disponible à tous les utilisateurs** — sélectionnez cette case pour créer une connexion disponible à tous les utilisateurs sur le système. Changer ce paramètre peut nécessiter des privilèges d'utilisateur root. Consulter Section 2.5.4, « Profils de connexions privées ou sur tout le système » pour obtenir plus d'informations.

- **Se connecter automatiquement au VPN quand on utilise cette connexion** — sélectionnez cette case si vous souhaitez que le **NetworkManager** s'auto-connecte à une connexion de VPN quand elle est disponible. Sélectionner le VPN à partir du menu déroulant.

- **Zone de parefeu** — sélectionnez une zone de parefeu dans le menu déroulant. Voir le guide *Red Hat Enterprise Linux 7 Security Guide* pour obtenir plus d'informations sur les zones de parefeux.

Configurer l'onglet VPN

Passerelle

Le nom ou l'adresse **IP** d'une passerelle de VPN distante.

Nom de groupe

Le nom d'un groupe de VPN configuré sur la passerelle distante.

Mot de passe utilisateur

Si besoin est, saisir le mot de passe utilisé pour s'authentifier auprès du VPN.

Mot de passe de groupe

Si besoin est, saisir le mot de passe utilisé pour s'authentifier auprès du VPN.

Nom d'utilisateur

Si besoin est, saisir le nom d'utilisateur qui a servi à s'authentifier auprès du VPN.

Phase1 Algorithmes

Si besoin est, saisir les algorithmes à utiliser pour s'authentifier et mettre en place un canal chiffré.

Phase2 Algorithmes

Si besoin est, saisir les algorithmes à utiliser pour les négociations IPsec.

Domaine

Si besoin est, saisir le Nom de domaine.

Sauvegarder votre nouvelle connexion (ou votre connexion modifiée) et faire des configurations suplémentaires

Une fois vous aurez terminé de modifier votre connexion VPN au réseau local virtuel, cliquez sur le bouton **Save** pour enregistrer votre configuration personnalisée. Si le profil est utilisé alors qu'il est en cours de modification, alimentez le cycle de connexion pour que le **NetworkManager** applique les modifications. Si le profil est désactivé (OFF), réglez-le sur ON ou sélectionnez-le dans le menu de l'icône de connexion réseau. Voir Section 2.5.1, « Se connecter à réseau par un GUI » pour plus d'informations sur l'utilisation de votre connexion nouvelle ou modifiée.

Vous pouvez configurer davantage une connexion existante en la sélectionnant dans la fenêtre **Réseau** et en cliquant sur **Configurer** pour revenir à la boîte de dialogue **Modifier**.

Puis, pour configurer :

- Paramètres de configuration **IPv4** pour la connexion, cliquer sur l'onglet **IPv4 Settings** et continuer avec Section 2.5.10.4, « Configuration des paramètres IPv4 ».

2.5.8. Établir une connexion à haut débit mobile

Vous pouvez utiliser les possibilités de la connexion à haut débit du **NetworkManager** pour vous connecter aux services *2G* et *3G* suivants :

- 2G — *GPRS* (*General Packet Radio Service*), *EDGE* (*Enhanced Data Rates for GSM Evolution*), ou CDMA (Code Division Multiple Access).
- 3G — *UMTS* (*Universal Mobile Telecommunications System*), *HSPA* (*High Speed Packet Access*), ou EVDO (EVolution Data-Only).

Votre machine doit disposer d'un périphérique à haut débit mobile (modem), que le système aura découvert et reconnu, afin de créer la connexion. Ce dispositif peut être intégré à votre ordinateur (comme c'est le cas de nombreux ordinateurs portables ou miniportables), ou peut être fourni séparément avec le matériel ou en externe. Il s'agit de la carte de PC, le Modem USB ou une clé électronique, un téléphone mobile ou cellulaire capable d'agir en tant que modem.

Procédure 2.5. Ajouter une nouvelle connexion à haut débit mobile

Vous pouvez configurer une connexion à haut débit mobile en ouvrant l'outil **Connexions de réseau** et en sélectionnant l'onglet **Haut débit mobile**.

1. Appuyer sur la touche **Super** pour accéder au menu Activités, saisir **nm-connection-editor**, et appuyez sur la touche **Enter**. L'outil de configuration **Connexions de réseau** apparaîtra.
2. Cliquer sur le bouton **Ajouter** pour ouvrir l'assistant **Sélectionner un type de connexion**.
3. Sélectionner l'entrée **Haut débit mobile** dans le menu.
4. Cliquer sur **Create** pour ouvrir l'assistant **Installer une connexion à haut débit mobile**.

5. Sous **Créer une connexion pour ce périphérique à haut débit mobile**, choisissez le périphérique 2 G ou 3 G que vous souhaitez utiliser avec la connexion. Si le menu déroulant est inactif, cela signifie que le système est incapable de détecter un périphérique à haut débit mobile. Dans ce cas, cliquez sur **Annuler**, et assurez-vous que vous n'avez pas un périphérique à haut débit mobile qui soit attaché et reconnu par l'ordinateur, puis essayez à nouveau cette procédure. Cliquez sur le bouton **Continuer**.

6. Sélectionner le pays où se trouve votre fournisseur de services dans la liste et cliquez sur le bouton **Continuer**.

7. Sélectionner le fournisseur dans la liste et saisissez le manuellement. Cliquer sur le bouton **Continuer**.

8. Sélectionnez votre plan de paiement dans le menu déroulant, et confirmer que le point d'accès *Access Point Name* (APN) est bien correct. Cliquer sur le bouton **Continuer**.

9. Vérifier et confirmer les paramètres de configuration, puis cliquer sur le bouton **Appliquer**.

10. Modifier la configuration spéciale haut débit mobile en consultant Section 2.5.8.1, « Configuration de l'onglet Haut débit mobile ».

Procédure 2.6. Modifier une connexion à haut débit mobile existante

Suivre ces étapes pour modifier une connexion à haut débit mobile existante.

1. Appuyer sur la touche **Super** pour accéder au menu Activités, saisir **nm-connection-editor**, et appuyez sur la touche **Enter**. L'outil de configuration **Connexions de réseau** apparaîtra.

2. Sélectionnez l'onglet **Haut débit mobile**.

3. Sélectionner la connexion à modifier et cliquer sur le bouton **Modifier**.

4. Configurer le nom de la connexion, le comportement auto-connect, et les paramètres disponibles.

   Il existe cinq configurations de la boîte de dialogue **Modifier** qui sont communes à tous les types de connexion. Voir l'onglet **Général** :

   - **Nom de connexion** — saisir un nom descriptif pour votre connexion de réseau. Ce nom sera utilisé pour lister cette connexion dans le menu de la fenêtre **Réseau**.

   - **Se connecter automatiquement à ce réseau quand il est disponible** — sélectionnez cette case si vous souhaitez que le **NetworkManager** s'auto-connecte à cette connexion quand elle est disponible. Voir Section 2.5.3, « Se connecter à un réseau automatiquement » pour plus d'informations.

   - **Rendre le réseau disponible à tous les utilisateurs** — sélectionnez cette case pour créer une connexion disponible à tous les utilisateurs sur le système. Changer ce paramètre peut nécessiter des privilèges d'utilisateur root. Consulter Section 2.5.4, « Profils de connexions privées ou sur tout le système » pour obtenir plus d'informations.

   - **Se connecter automatiquement au VPN quand on utilise cette connexion** — sélectionnez cette case si vous souhaitez que le **NetworkManager** s'auto-connecte à une connexion de VPN quand elle est disponible. Sélectionner le VPN à partir du menu déroulant.

- **Zone de parefeu** — sélectionnez une zone de parefeu dans le menu déroulant. Voir le guide *Red Hat Enterprise Linux 7 Security Guide* pour obtenir plus d'informations sur les zones de parefeux.

5. Modifier la configuration spéciale haut débit mobile en consultant Section 2.5.8.1, « Configuration de l'onglet Haut débit mobile ».

Sauvegarder votre nouvelle connexion (ou votre connexion modifiée) et faire des configurations suplémentaires

Une fois vous aurez terminé de modifier votre connexion à haut débit mobile, cliquez sur le bouton **Apply** pour enregistrer votre configuration personnalisée. Si le profil était utilisé pendant la modification, alimentez le cycle de connexion pour que **NetworkManager** applique les modifications. Si le profil est désactivé (OFF), réglez-le sur ON ou sélectionnez-le dans le menu de l'icône de connexion réseau. Voir Section 2.5.1, « Se connecter à réseau par un GUI » pour plus d'informations sur l'utilisation de votre connexion nouvelle ou modifiée.

Vous pouvez configurer davantage une connexion existante en la sélectionnant dans la fenêtre **Connexions réseau** et en cliquant sur **Modifier** pour revenir à la boîte de dialogue **Modification**.

Puis, pour configurer :

- Configuration **Point-to-point** de la connexion, cliquez sur l'onglet **PPP Settings** et continuez avec Section 2.5.10.3, « Configurer les paramètres PPP (Point-to-Point) »;
- Paramètres de configuration **IPv4** pour la connection, cliquer sur l'onglet **IPv4 Settings** et continuer avec Section 2.5.10.4, « Configuration des paramètres IPv4 »; ou,
- Paramètres de configuration **IPv6** pour la connexion, cliquer sur l'onglet **IPv6 Settings** et continuez avec Section 2.5.10.5, « Configurer les paramètres IPv6 ».

2.5.8.1. Configuration de l'onglet Haut débit mobile

Si vous avez déjà ajouté une connexion à haut débit mobile à l'aide de l'assistant (voir Procédure 2.5, « Ajouter une nouvelle connexion à haut débit mobile » pour obtenir des instructions), vous pouvez modifier l'onglet **Haut débit mobile** pour désactiver l'itinérance si le réseau domestique n'est pas disponible, ou attribuer un ID de réseau ou inciter le **NetworkManager** à préférer une certaine technologie (par exemple 3 G ou 2 G) quand il utilise la connexion.

Numéro

Le numéro qui est composé pour établir une connexion PPP avec le réseau à haut débit mobile basé GSM. Ce champ peut être rempli automatiquement pendant l'installation initiale d'un périphérique à haut débit. Vous pouvez généralement laisser ce champ vide et entrer l'**APN** à la place.

Nom d'utilisateur

Saisir le nom d'utilisateur qui sert à s'authentifier sur le réseau. Certains fournisseurs ne donnent pas de nom d'utilisateur, et n'acceptent pas n'importe quel nom d'utilisateur quand vous souhaitez vous connecter au réseau.

Mot de passe

Saisir le mot de passe utilisé pour s'authentifier au réseau. Certains fournisseurs ne vous donneront pas de mot de passe, ni n'accepteront n'importe quel mot de passe.

APN

Saisir le point d'accès ou *Access Point Name* (APN) utilisé pour établir une connexion au réseau basé GSM. Saisir l'APN qui convient à une connexion est important car cela détermine souvent :

- Comment l'utilisateur est facturé pour son utilisation réseau: et/ou
- si l'utilisateur a accès à l'internet, un intranet, ou un sous-réseau.

`ID Réseau`

Saisir un **`ID Réseau`** amène le **NetworkManager** à forcer le périphérique à s'enregistrer uniquement auprès d'un réseau spécifique. Cela peut être utilisé pour s'assurer que la connexion n'est pas itinérante quand on ne peut pas contrôler l'itinérance directement.

`Type`

`Tous` — la valeur par défaut **`Tous`** laisse le modem choisir le réseau le plus rapide.

`3G (UMTS/HSPA)` — force la connexion à n'utiliser que les technologies de réseau 3G.

`2G (GPRS/EDGE)` — force la connexion à n'utiliser que les technologies de réseau 2G.

`Prefer 3G (UMTS/HSPA)` — première tentative de connexion à un réseau 3G comme HSPA ou UMTS, ou GPRS/EDGE en cas d'échec.

`Prefer 2G (GPRS/EDGE)` — première tentative de connexion à un réseau 2G comme GPRS ou EDGE, ou HSPA/UMTS en cas d'échec.

`Autoriser l'itinérance si le réseau domestique est indisponible`

Décochez cette case si vous souhaitez que le **NetworkManager** termine la connexion au lieu de transitionner d'un réseau domestique à un réseau itinérant, évitant ainsi les charges de réseau itinérant. Si la case est cochée, le **NetworkManager** essaiera de maintenir une bonne connexion en passant du réseau domestique au réseau itinérant, et vice versa.

`PIN`

Si le *SIM* (*Subscriber Identity Module*) de votre périphérique est vérrouillé par un *PIN* (*Personal Identification Number*), saisir le PIN pour que le **NetworkManager** puisse déverrouiller le périphérique. Le **NetworkManager** doit déverrouiller le SIM si le PIN est demandé, afin de pouvoir utiliser le périphérique dans des buts divers.

CDMA et EVDO ont plusieurs options. Ils ont les options **`APN`**, **`ID Réseau`**, ou **`Type`**.

2.5.9. Établir une connexion DSL

Cette section s'adresse aux installations qui ont une carte DSL intégrée dans l'hôte plutôt que des combinaisons de router/modem DSL externes, typique des installations SOHO ou de consommateurs privés.

Procédure 2.7. Ajouter une nouvelle connexion DSL

Vous pouvez configurer une nouvelle connexion DSL en ouvrant la fenêtre **`Connections réseau`**, en cliquant le bouton **`Ajouter`** et en sélectionnant **`DSL`** dans la section **`Matériel`** à partir de la liste de la nouvelle connexion.

1. Appuyer sur la touche **Super** pour accéder au menu Activités, saisir **nm-connection-editor**, et appuyez sur la touche **Enter**. L'outil de configuration **Connexions de réseau** apparaîtra.

2. Cliquer sur le bouton **Ajouter**.

3. La liste **Sélectionner un nouveau type de connexion** apparaîtra.

4. Sélectionner **DSL** et appuyer sur le bouton **Créer**.

5. La fenêtre **Modifier connexion DSL*1*** apparaîtra.

Procédure 2.8. Modifier une connexion DSL existante

Vous pouvez configurer une connexion DSL existante en ouvrant la fenêtre **Connexions réseau**, et en sélectionnant le nom de la connexion dans la liste. Puis, cliquer sur le bouton **Modifier**.

1. Appuyer sur la touche **Super** pour accéder au menu Activités, saisir **nm-connection-editor**, et appuyez sur la touche **Enter**. L'outil de configuration **Connexions de réseau** apparaîtra.

2. Sélectionner la connexion à modifier et cliquer sur le bouton **Modifier**.

Configurer le Nom de connexion, le Comportement Auto-Connect, et la Disponiblité

Il existe cinq configurations de la boîte de dialogue **Modifier** qui sont communes à tous les types de connexion. Voir l'onglet **Général** :

- **Nom de connexion** — saisir un nom descriptif pour votre connexion de réseau. Ce nom sera utilisé pour lister cette connexion dans le menu de la fenêtre **Réseau**.

- **Se connecter automatiquement à ce réseau quand il est disponible** — sélectionnez cette case si vous souhaitez que le **NetworkManager** s'auto-connecte à cette connexion quand elle est disponible. Voir Section 2.5.3, « Se connecter à un réseau automatiquement » pour plus d'informations.

- **Rendre le réseau disponible à tous les utilisateurs** — sélectionnez cette case pour créer une connexion disponible à tous les utilisateurs sur le système. Changer ce paramètre peut nécessiter des privilèges d'utilisateur root. Consulter Section 2.5.4, « Profils de connexions privées ou sur tout le système » pour obtenir plus d'informations.

- **Se connecter automatiquement au VPN quand on utilise cette connexion** — sélectionnez cette case si vous souhaitez que le **NetworkManager** s'auto-connecte à une connexion de VPN quand elle est disponible. Sélectionner le VPN à partir du menu déroulant.

- **Zone de parefeu** — sélectionnez une zone de parefeu dans le menu déroulant. Voir le guide *Red Hat Enterprise Linux 7 Security Guide* pour obtenir plus d'informations sur les zones de parefeux.

Configurer l'onglet DSL

Nom d'utilisateur

Saisir le nom d'utilisateur qui sert à s'authentifier auprès du fournisseur de service.

Service

Laissez ce champ vide à moins d'en être instruit autrement par votre fournisseur de service.

Mot de passe

Saisir le mot de passe fourni par le fournisseur de service.

Sauvegarder votre nouvelle connexion (ou votre connexion modifiée) et faire des configurations suplémentaires

Une fois vous aurez terminé de modifier votre connexion filaire DSL au réseau local virtuel, cliquez sur le bouton **Apply** pour enregistrer votre configuration personnalisée. Si le profil est en cours d'utilisation lors de la modification, alimentez le cycle de connexion pour que le **NetworkManager** applique les modifications. Si le profil est désactivé (OFF), réglez-le sur ON ou sélectionnez-le dans le menu de l'icône de connexion réseau. Voir Section 2.5.1, « Se connecter à réseau par un GUI » pour plus d'informations sur l'utilisation de votre connexion nouvelle ou modifiée.

Vous pouvez configurer davantage une connexion existante en la sélectionnant dans la fenêtre **Connexions réseau** et en cliquant sur **Modifier** pour revenir à la boîte de dialogue **Modification**.

Puis, pour configurer :

- Configuration **Adresse MAC et MTU**, cliquez sur l'onglet **Filaire** et continuez avec Section 2.5.5.1, « Configurer le Nom de connexion, le Comportement Auto-Connect, et la Disponiblité »;
- Configuration **Point-to-point** de la connexion, cliquez sur l'onglet **PPP Settings** et continuez avec Section 2.5.10.3, « Configurer les paramètres PPP (Point-to-Point) »;
- Paramètres de configuration **IPv4** pour la connexion, cliquer sur l'onglet **IPv4 Settings** et continuer avec Section 2.5.10.4, « Configuration des paramètres IPv4 ».

2.5.10. Configuration des paramètres de connexion

2.5.10.1. Configuration de la sécurité 802.1X

802. 1X security est le nom de la norme IEEE pour le *Contrôle d'accès réseau basé sur le port* (PNAC). S'appelle également *WPA Entreprise*. En termes simples, 802. 1X security est un moyen de contrôle d'accès à un *réseau logique* à partir d'un réseau physique. Tous les clients qui veulent se joindre au réseau logique doivent s'authentifier auprès du serveur (un routeur, par exemple) à l'aide de la méthode d'authentification 802.1X qui convient.

802.1X security est souvent associé à la sécurisation des réseaux sans fil (les WLAN), mais peut également être utilisé pour empêcher des intrus ayant un accès physique au réseau (LAN) d'accéder. Dans le passé, les serveurs **DHCP** étaient configurés de façon à ne pas allouer d'adresses **IP** à des utilisateurs non autorisés, mais pour diverses raisons, cette pratique est peu commode et peu sûre, et donc n'est plus recommandée. À la place, 802.1X security est utilisé pour garantir un réseau logiquement sécurisé grâce à une authentification basée sur les ports.

802.1X fournit une structure de contrôle d'accès à WLAN et à LAN et sert d'enveloppe de transport pour les types d'EAP (Extensible Authentification Protocol). Un type EAP est un protocole qui définit comment la sécurité se profile sur un réseau.

Vous pouvez configurer 802.1X security d'un type de connexion filaire ou sans fil, en ouvrant la fenêtre **Réseau** (voir Section 2.5.1, « Se connecter à réseau par un GUI ») et en suivant la procédure qui s'applique ci-dessous. Appuyez sur la touche **Super** pour saisir la Vue d'ensemble des activités, saisir

control network et appuyer sur la touche **Enter**. L'outil de configuraition du **Réseau** apparaîtra. Continuez avec Procédure 2.9, « Pour une connexion filaire » ou Procédure 2.10, « Pour une connexion sans fil »:

Procédure 2.9. Pour une connexion filaire

1. Sélectionner l'interface de réseau **Filaire** à partir du menu sur la gauche.
2. Vous pouvez soit cliquer sur **Ajouter profil** pour ajouter une nouvelle connexion de réseau sur laquelle vous souhaitez configurer la sécurité 802.1X, ou sélectionner un profil et cliquer sur l'icône de roue dentée.
3. Puis, sélectionner **Security** et mettre le bouton d'alimentation symbolique sur **ON** pour activer la configuration.
4. Continuez avec Section 2.5.10.1.1, « Configuration des paramètres TLS (Transport Layer Security) »

Procédure 2.10. Pour une connexion sans fil

1. Sélectionner une interface de réseau **Sans fil** dans le menu de gauche. Si besoin est, mettre le bouton d'alimentation symbolique sur **ON** et vérifiez que votre bouton de matériel est bien branché.
2. Sélectionner le nom de la connexion d'une nouvelle connexion ou cliquer sur l'icône de roue dentée d'un profil de connexion existant, pour lequel vous souhaitez configurer la sécurité 802. 1 X. Dans le cas d'une nouvelle connexion, compléter toutes les étapes d'authentification pour terminer la connexion, puis cliquer sur l'icône de roue dentée.
3. Sélectionner **Sécurité**.
4. Dans le menu déroulant, sélectionner une des méthodes de sécurité suivantes : **LEAP**, **Dynamic WEP (802.1X)**, ou **WPA & WPA2 Enterprise**.
5. Voir Section 2.5.10.1.1, « Configuration des paramètres TLS (Transport Layer Security) » pour obtenir des descriptions des types de *protocoles d'authentitification extensibles* (EAP de l'anglais Extensible Authentification Protocol) qui correspondent à votre sélection dans le menu déroulant **Sécurité**.

2.5.10.1.1. Configuration des paramètres TLS (Transport Layer Security)

Avec TLS, le client et le serveur s'authentifient mutuellement par le protocole TLS. Le serveur prouve qu'il détient un certificat digital, le client prouve sa propre identité en utilisant son certificat côté client, et des informations de clés sont échangées. Une fois que l'authentification est terminée, le tunnel TLS n'est plus utilisé. À la place, le client et le serveur utilisent les clés échangées pour chiffrer les données par AES, TKIP ou WEP.

Le fait que les certificats doivent être distribués à tous les clients qui veulent s'authentifier signifie que la méthode d'authentification EAP-TLS est robuste, mais également plus compliquée à mettre en place. Utiliser la sécurité TLS requiert un investissement d'infrastructure à clé publique (PKI) (Public Key Infrastructure) pour gérer les certificats. L'avantage d'utiliser la sécurité TLS est qu'un mot de passe compromis ne permet pas l'accès au réseau local (W) : un intrus doit également avoir accès à la clé privée du client qui s'authentifie.

Le **NetworkManager** ne détermine pas la version du TLS pris en charge. Le **NetworkManager** collecte les paramètres saisis par l'utilisateur, et les passe au démon, **wpa_supplicant**, qui gère la procédure. Il

utilise, à son tour, OpenSSL pour établir le tunnel TLS. OpenSSL lui-même négocie la version de protocole SSL/TLS. Il utilise la version la plus récente que les deux parties supportent.

Sélection d'une méthode d'authentification
Sélectionner l'une des méthodes d'authentification suivantes :

- Sélectionner **TLS** pour *Transport Layer Security* et continuer avec Section 2.5.10.1.2, « Configuration les paramètres TLS »;
- Sélectionnez **FAST** pour *Flexible Authentication via Secure Tunneling* et continuez avec Section 2.5.10.1.4, « Configurer les paramètres TLS Tunneled »;
- Sélectionner **Tunneled TLS** pour *Tunneled Transport Layer Security*, aussi connu comme TTLS, ou EAP-TTLS et continuer avec Section 2.5.10.1.4, « Configurer les paramètres TLS Tunneled » ;
- Sélectionner **Protected EAP (PEAP)** pour *Protected Extensible Authentication Protocol* et continuer avec Section 2.5.10.1.5, « Configurer un PEAP (Protected EAP) ».

2.5.10.1.2. Configuration les paramètres TLS

Identité

Fournit l'identité de ce serveur.

Certificat d'utilisateur

Naviguer sur un fichier certificat X.509 personnel encodé avec DER (*Distinguished Encoding Rules*) ou avec PEM (*Privacy Enhanced Mail*).

Certificat CA

Naviguer sur un fichier certificat *CA* X.509 personnel encodé avec DER (*Distinguished Encoding Rules*) ou avec PEM (*Privacy Enhanced Mail*).

Clé privée

Naviguer sur un fichier *clé privée* encodé avec DER(*Distinguished Encoding Rules*), PEM (*Privacy Enhanced Mail*), ou PKCS #12 (*Personal Information Exchange Syntax Standard*).

Mot de passe de la clé privée

Saisir le mot de passe de la clé privée spécifiée dans le champ **Clé privée**. Sélectionnez **Afficher le mot de passe** pour rendre le mot de passe visible lorsque vous le saisissez.

2.5.10.1.3. Configuration les paramètres FAST

Identité anonyme

Fournit l'identité de ce serveur.

Allocation PAC

Cochez la case pour activer, puis choisissez entre **Anonymous**, **Authenticated**, et **Both**.

Fichier PAC

Naviguez pour sélectionner un fichier PAC (*protected access credential*).

`Authentification interne`

GTC — Generic Token Card.

MSCHAPv2 — Microsoft Challenge Handshake Authentication Protocol version 2.

`Nom d'utilisateur`

Saisir le nom de l'utilisateur à utiliser pour le processus d'authentification.

`Mot de passe`

Saisir le mot de passe à utiliser pour le processus d'authentification.

2.5.10.1.4. Configurer les paramètres TLS Tunneled

`Identité anonyme`

Cette valeur est utilisée comme identité cryptée.

`Certificat CA`

Naviguez pour sélectionner un certificat CA.

`Authentification interne`

PAP — Password Authentication Protocol.

MSCHAP — Challenge Handshake Authentication Protocol.

MSCHAPv2 — Microsoft Challenge Handshake Authentication Protocol version 2.

CHAP — Challenge Handshake Authentication Protocol.

`Nom d'utilisateur`

Saisir le nom de l'utilisateur à utiliser pour le processus d'authentification.

`Mot de passe`

Saisir le mot de passe à utiliser pour le processus d'authentification.

2.5.10.1.5. Configurer un PEAP (Protected EAP)

`Identité anonyme`

Cette valeur est utilisée comme identité cryptée.

`Certificat CA`

Naviguez pour sélectionner un certificat CA.

`Version PEAP`

Version de Protected EAP à utiliser. Automatique, 0 ou 1.

`Authentification interne`

MSCHAPv2 — Microsoft Challenge Handshake Authentication Protocol version 2.

MD5 — Message Digest 5, une fonction de hachage cryptographique.

GTC — Generic Token Card.

`Nom d'utilisateur`

Saisir le nom de l'utilisateur à utiliser pour le processus d'authentification.

`Mot de passe`

Saisir le mot de passe à utiliser pour le processus d'authentification.

2.5.10.2. Configurer la sécurité Wi-Fi

`Securité`

Aucune — ne pas chiffrer la connexion Wi-Fi.

WEP 40/128-bit Key — Wired Equivalent Privacy (WEP), du standard IEEE 802.11. Utilise une clé unique PSK (de l'anglais Pre-Shared Key).

WEP 128-bit Passphrase — Un hachage MD5 de la phrase de passe qui sera utilisée pour former la clé WEP.

LEAP — Lightweight Extensible Authentication Protocol, Cisco Systems.

Dynamic WEP (802.1X) — les clés WEP sont échangées dynamiquement. À utiliser avec Section 2.5.10.1.1, « Configuration des paramètres TLS (Transport Layer Security) »

WPA & WPA2 Personal — Wi-Fi Protected Access (WPA), provient du projet de standard IEEE 802.11i. Vient en remplacement de WEP. Wi-Fi Protected Access II (WPA2), du standard 802.11i-2004. Le mode personnel utilise une clé WPA-PSK.

WPA & WPA2 Enterprise — WPA à utiliser avec un serveur d'authentification RADIUS pour donner un contrôle d'accès au réseau IEEE 802.1X. À utiliser avec Section 2.5.10.1.1, « Configuration des paramètres TLS (Transport Layer Security) »

Mot de passe

Saisir le mot de passe à utiliser pour le processus d'authentification.

2.5.10.3. Configurer les paramètres PPP (Point-to-Point)

`Méthodes de configuration`

`Cryptage de point à point de Microsoft (MPPE)`

Protocole Microsoft Point-To-Point Encryption (MPPE) (*RFC 3078*).

`Permet la compression de données BSD`

PPP BSD Compression Protocol (*RFC 1977*).

Permet la déflation de compression de données

PPP Deflate Protocol (*RFC 1979*).

Utiliser la compression d'en-tête TCP

Compression des en-têtes pour les liens en séries à petite vitesse (*RFC 1144*).

Envoyer PPP echo packets

Codes LCP Echo-Request et Echo-Reply pour les tests de bouclage (*RFC 1661*).

2.5.10.4. Configuration des paramètres IPv4

L'onglet **Paramètrage IPv4** vous permet de configurer la méthode utilisée pour se connecter à un réseau, d'entrer l'adresse **IP**, le routage et les informations **DNS**, selon les besoins. L'onglet **ParamètrageIPv4** est là pour que vous puissiez créer et modifier un des types de connexions suivantes : filaire, sans fil, à haut débit mobile, VPN ou DSL. Si vous avez besoin de configurer les adresses **IPv6**, voir Section 2.5.10.5, « Configurer les paramètres IPv6 ». Si vous devez configurer des routes statiques, cliquez sur le bouton **Routages** et passez à Section 2.5.10.6, « Routes de configuration ».

Si vous utilisez **DHCP** pour obtenir une adresse **IP** dynamique de la part d'un serveur **DHCP**, vous n'avez plus qu'à définir la **Méthode** à **Automatic (DHCP)**.

Définir la méthode

Méthodes IPv4 disponibles par type de connexion

Quand vous cliquez sur le menu déroulant **Méthode**, selon le type de connexion que vous configurez, vous pourrez sélectionner l'une des méthodes de connexion **IPv4** suivantes. Toutes les méthodes sont répertoriées ici par type de connexion, ou de types auquels elles sont associées :

Méthode

Automatic (DHCP) — sélectionner cette option si le réseau auquel vous vous connectez utilise un serveur **DHCP** pour assigner les adresses **IP**. Vous n'avez pas besoin de remplir le champ **ID client DHCP**.

Automatic (DHCP), adresses uniquement — sélectionner cette option si le réseau auquel vous vous connectez utilise un serveur **DHCP** pour assigner les adresses **IP**, mais que vous souhaitiez assigner vos serveurs **DNS** manuellement.

Link-Local Uniquement — sélectionner cette option si le réseau auquel vous vous connectez utilise un serveur **DHCP**, et que vous ne souhaitiez pas assigner les adresses **IP** manuellement. Des adresses aléatoires seront assignées selon *RFC 3927* avec le préfixe **169.254/16**.

Partagé avec d'autres ordinateurs— sélectionnez cette option si l'interface que vous configurez doit servir à partager une connexion Internet ou WAN. L'interface reçoit une adresse dans la plage **10.42.x.1/24** range, un serveur **DHCP** et un serveur **DNS** sont démarrés, et l'interface est connectée sur le système à une connexion de réseau par défaut par le *network address translation* (NAT).

Désactivé — **IPv4** est désactivé pour cette connexion.

Méthodes de connexion Filaire, Sans fil et DSL

Manuel — Sélectionnez cette option si vous souhaitez assigner des adresses **IP** manuellement.

Méthodes de connexions à haut débit mobiles

Automatic (PPP) — sélectionner cette option si le réseau auquel vous vous connectez assigne votre adresse **IP** et vos serveurs **DNS** automatiquement.

Adresses automatic (PPP) uniquement — sélectionner cette option si le réseau auquel vous vous connectez assigne votre adresse **IP** et vos serveurs automatiquement, mais que vous souhaitez spécifier les serveurs **DNS** manuellement.

Méthodes de connexion VPN

Automatic (VPN) — sélectionner cette option si le réseau auquel vous vous connectez assigne votre adresse **IP** et vos serveurs **DNS** automatiquement.

Adresses automatic (VPN) uniquement — sélectionner cette option si le réseau auquel vous vous connectez assigne votre adresse **IP** et vos serveurs automatiquement, mais que vous souhaitez spécifier les serveurs **DNS** manuellement.

Méthodes de connexion DSL

Automatic (PPPoE) — sélectionner cette option si le réseau auquel vous vous connectez assigne votre adresse **IP** et vos serveurs **DNS** automatiquement.

Adresses automatic (PPPoE) uniquement — sélectionner cette option si le réseau auquel vous vous connectez assigne votre adresse **IP** et vos serveurs automatiquement, mais que vous souhaitez spécifier les serveurs **DNS** manuellement.

Pour obtenir des informations sur la façon de configurer des routes statiques sur la connexion réseau, voir Section 2.5.10.6, « Routes de configuration ».

2.5.10.5. Configurer les paramètres IPv6

Méthode

Ignorer — sélectionnez cette option si vous souhaitez ignorer les paramètres **IPv6** de cette connexion.

Automatic —sélectionnez cette option pour utiliser *SLAAC* afin de créer une configuration automatique, stateless basée sur l'adresse de matériel et les annonces de routage (RA) (de l'anglais *router advertisements*).

Automatic , adresses uniquement — sélectionner cette option si le réseau auquel vous vous connectez utilise les annonces de routage (RA) (de l'anglais *router advertisements* pour créer une configuration automatique stateless, mais que vous souhaitez assigner vos serveurs **DNS** manuellement.

Automatic, DHCP uniquement — sélectionnez cette option pour ne pas utiliser RA, mais demandez des informations de **DHCPv6** directement pour créer une configuration stateful.

Manuel — Sélectionnez cette option si vous souhaitez assigner des adresses **IP** manuellement.

Link-Local Uniquement — sélectionner cette option si le réseau auquel vous vous connectez utilise un serveur **DHCP**, et que vous ne souhaitiez pas assigner les adresses **IP** manuellement. Des adresses aléatoires seront assignées selon *RFC 4862* avec le préfixe **FE80::0**.

Adresses

Serveurs DNS — saisir une liste de serveurs **DNS** séparée par des virgules.

Domaines de recherche — saisir une liste de contrôleurs de domaines séparée par des virgules.

Pour obtenir des informations sur la façon de configurer des routes statiques sur la connexion réseau, voir Section 2.5.10.6, « Routes de configuration ».

2.5.10.6. Routes de configuration

Une table de routage d'hôte se remplira automatiquement avec les routes vers les réseaux directement connectés. Les routes sont apprises en examinant les interfaces réseau lorsqu'elles sont « actives ». Cette section décrit la saisie de routes statiques allant vers des réseaux ou des hôtes qui peuvent être atteints en traversant un réseau intermédiaire ou une connexion, comme par exemple, un tunnel VPN ou ligne allouée. Pour atteindre un réseau distant ou un hôte, le système reçoit l'adresse d'une passerelle vers lequel le trafic doit être dirigé.

Quand une interface d'hôte est configurée via **DHCP**, une adresse de passerelle qui mène à un réseau en amont ou à Internet est généralement assignée. Cette passerelle est généralement considérée comme étant la passerelle par défaut, car c'est la porte d'entrée à utiliser si aucun autre itinéraire plus adequat n'est connu du système (et présent dans la table de routage). Les administrateurs réseau utilisent souvent l'adresse **IP** du premier ou du dernier hôte dans le réseau comme adresse de passerelle ; par exemple, **192.168.10.1** ou **192.168.10.254**. Ne doit pas être confondu avec l'adresse qui représente le réseau lui-même ; dans cet exemple, **192.168.10.0** ou adresse de diffusion du sous-réseau ; dans cet exemple **192.168.10.255**.

Configuration des routes statiques
Pour définir une route statique, ouvrir la fenêtre de configuration **IPv4** ou **IPv6** pour la connexion que vous souhaitez configurer. Voir Section 2.5.1, « Se connecter à réseau par un GUI » pour obtenir des instructions sur la façon de procéder.

Routes

Adresse — saisir l'adresse **IP** de réseau, sous-réseau ou hôte distant.

masque réseau — le masque réseau ou la longueur du préfixe de l'adresse **IP** saisie ci-dessus.

Gateway — l'adresse **IP** de la passerelle menant au réseau, sous-réseau ou hôte distant ci-dessus.

Metric — un coût de réseau, une valeur souhaitable à donner à ce routage. On favorisera les valeurs élevées.

Automatique

Quand « Automatic » est **ON** (activé), les routages de **RA** ou de **DHCP** seront utilisés, mais vous pouvez également ajouter des routages statiques. Quand « Automatic » est **OFF** (désactivé), seuls les routages statiques que vous définissez seront utilisés.

Utiliser cette connexion pour les ressources sur ce réseau uniquement

Sélectionner cette case pour éviter que la connexion ne devienne le routage par défaut. Les

exemples typiques sont les cas où une connexion correspond à un tunnel VPN ou à une ligne allouée à un siège d'entreprise et quand vous ne souhaitez pas que tout le trafic Internet passe sur cette connexion. En sélectionnant cette option, cela signifie que seul le trafic spécifiquement destiné aux routages appris automatiquement sur la connexion ou saisie ici manuellement sera routé via la connexion.

2.6. RESSOURCES SUPPLÉMENTAIRES

Les sources d'informations suivantes fournissent des ressources supplémentaires pertinentes à ce chapitre.

2.6.1. Documentation installée

- Page man **`ip(8)`** — décrit la syntaxe de l'utilitaire **ip**.
- Page man **`nmcli(1)`** — décrit l'outil de ligne de commandes du **NetworkManager**.
- Page man **`nmcli-examples(5)`** — donne des exemples des commandes **nmcli**.
- Page man **`nm-settings(5)`** — décrit les propriétés du **NetworkManager** et leurs paramètres de configuration.

2.6.2. Documentation en ligne

Red Hat Enterprise Linux 7 Security Guide

Décrit le VPN basé IPsec et sa configuration, ainsi que la façon d'utiliser les interrogations **`DNS`** authentifiées avec DNSSEC.

RFC 1518 **— Classless Inter-Domain Routing (CIDR)**

Décrit la stratégie d'agrégation et l'attribution d'adresses CIDR, ainsi que les sous-réseautage à longueurs variables.

RFC 1918 **— allocations d'adresses pour les Internets privés**

Décrit l'étendue des adresses **`IPv4`** réservées à usage privé.

RFC 3330 **— adresses IPv4 à usage spécial**

Décrit les blocs d'adresses **`IPv4`** qui ont été assignées par l'IANA (Internet Assigned Numbers Authority).

CHAPITRE 3. CONFIUGURATION DE NOMS D'HÔTES

3.1. COMPRENDRE LES NOMS D'HÔTES

Il y a trois classes de noms d'hôtes ou **hostname** : statique, pretty, et transitoire

Le nom d'hôte « statique » est un **hostname** traditionnel, qui peut être choisi par l'utilisateur, et qui se trouve dans le fichier **/etc/hostname**. Le nom d'hôte « transitoire » est un **hostname** dynamique maintenu par le noyau. Il est initialisé au nom d'hôte statique par défaut, dont la valeur correspond au « localhost ». Il peut être changé par **DHCP** ou **mDNS** en cours d'exécution. Le nom d'hôte « pretty » (**hostname**) se présente en forme libre UTF8 à l'utilisateur.

NOTE

Un nom d'hôte peut correspondre à une chaîne en forme libre allant jusqu'à 64 caractères. Red Hat recommande cependant que les noms statiques et transitoires correspondent au *nom de domaine complet* (FQDN) utilisé pour la machine pour le **DNS**, comme **host.exemple.com**. Il est également recommandé que les noms statiques et transitoires se composent uniquement de caractères minuscules ASCII de 7 octets, sans espace ou point, et qu'ils se limitent au format autorisé pour les étiquettes de nom de domaine **DNS**, même si ce n'est pas une exigence stricte. Les spécifications plus anciennes ne permettent pas le trait de soulignement ; leur utilisation n'est donc pas conseillée.

L'outil **hostnamectl** fera en sorte que les noms d'hôte statiques, transitoires ou pretty soient composés des éléments suivants **a-z**, **A-Z**, **0-9**, « **-** », « _ » et « **.** » uniquement, qu'ils ne commencent, ni ne se terminent par un point, et qu'ils ne soient pas composés de deux points à la suite l'un de l'autre. La taille limite est de 64 caractères.

3.1.1. Pratiques de nommage conseillées

ICANN (Internet Corporation for Assigned Names and Numbers) ajoute parfois des domaines de premier niveau précédemment non enregistrés (par exemple, **.yourcompany**) dans le registre public. Par conséquent, Red Hat recommande fortement que vous n'utilisiez pas un nom de domaine qui ne vous soit pas octroyé, même sur un réseau privé, car cela pourrait résulter en un nom de domaine qui se résolve différemment selon la configuration de réseau. Par conséquence, les ressources réseau peuvent devenir indisponibles. Utiliser des noms de domaine qui ne vous sont pas délégués complique aussi le déploiement et la maintenance de DNSSEC, car les colisions de noms de domaines nécessitent une configuration manuelle pour activer la validation de DNSSEC. Consultez la FAQ ICANN FAQ on domain name collision de l'ICANN sur les collisions de noms de domaine pour plus d'informations sur cette question.

3.2. CONFIGURER DES NOMS D'HÔTES PAR L'INTERFACE D'UTILISATEUR TEXTE, NMTUI

L'outil d'interface utilisateur de texte **nmtui** peut être utilisée pour configurer un nom d'hôte dans une fenêtre de terminal. Exécutez la commande suivante pour démarrer l'outil :

```
~]$ nmtui
```

L'interface utilisateur texte apparaîtra. Tout commande non valide affichera un message d'utilisation.

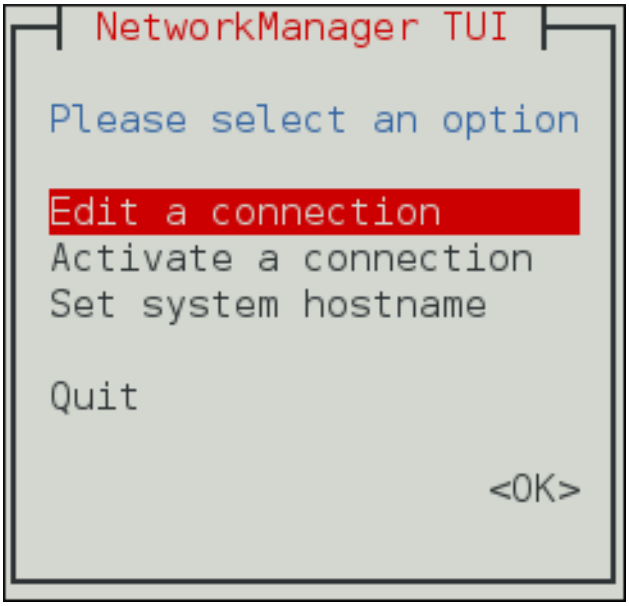


Figure 3.1. Le menu de démarrage de l'interface utilisateur texte du NetworkManager

Pour naviguer, utiliser les flèches ou appuyer sur **Tab** pour continuer et appuyer sur la combinaision de touches **Maj**+**Tab** pour revenir aux options. Appuyer sur la touche **Entrée** pour sélectionner une option. La barre **Espace** active/désactive le statut d'un case à cocher.

Voir Section 1.5, « Configuration réseau utilisant une interface utilisateur texte (nmtui) » pour obtenir des informations sur la façon d'installer **nmtui**.

L'outil d'interface utilisateur de texte du **NetworkManager**, **nmtui**, peut être utilisé pour interroger et définir le nom d'hôte statique dans le fichier **/etc/hostname**. Notez qu'au moment de la rédaction, changer le nom de cette façon ne sera pas remarqué par le **hostnamectl**.

Pour forcer le **hostnamectl** à remarquer le changement de nom d'hôte statique, démarrer à nouveau le **hostnamed** en tant qu'utilisateur **root**:

```
~]# systemctl restart systemd-hostnamed
```

3.3. CONFIGURER DES NOMS D'HÔTES PAR HOSTNAMECTL

L'outil **hostnamectl** est fourni pour administrer les trois classes de noms d'hôtes utilisées sur un système donné.

3.3.1. Voir tous les noms d'hôtes

Pour voir tous les noms d'hôtes actuels, saisir la commande suivante :

```
~]$ hostnamectl status
```

L'option **status** est impliquée par défaut si aucune option n'est donnée.

3.3.2. Voir tous les noms d'hôte

Pour voir tous les noms d'hôte dans un système, saisir la commande suivante en tant qu'utilisateur **root** :

```
~]# hostnamectl set-hostname name
```

Cela altèrera à la fois les noms pretty, statiques ou transitoires. Les noms statiques et transitoires seront sous une forme simplifiée des noms d'hôte « pretty ». Les espaces seront remplacés par des « **-** » et les caractères spéciaux seront supprimés.

3.3.3. Définir un nom d'hôte particulier

Pour saisir un nom d'hôte particulier, saisir la commande suivante en tant qu'utilisateur **root** avec l'option qui convient :

```
~]# hostnamectl set-hostname nom [option...]
```

Quand *option* correspond à un ou plusieurs : **--pretty**, **--static**, et **--transient**.

Si les options **--static** ou **--transient** sont utilisées ensemble avec l'option **--pretty**, les noms statiques et transitoires seront sous une forme simplifiée des noms d'hôte « pretty ». Les espaces seront remplacés par des « **-** » et les caractères spéciaux seront supprimés. Si l'option **--pretty** n'est pas donnée, aucune simplification n'aura lieu.

Quand on définit un nom d'hôte « pretty », ne pas oublier d'utiliser les caractères de citation qui conviennent si le nom d'hôte contient des espaces ou un caractère de citation unique. Par exemple :

```
~]# hostnamectl set-hostname "Stephen's notebook" --pretty
```

3.3.4. Supprimer un nom d'hôte particulier

Pour supprimer un nom d'hôte particulier et lui permettre de retrouver sa valeur par défaut, saisir la commande suivante en tant qu'utilisateur **root** avec l'option qui convient :

```
~]# hostnamectl set-hostname "" [option...]
```

Quand *""* correspond à une chaîne de cotation vide et que l'*option* correspond à un ou plusieurs : **--pretty**, **--static**, et **--transient**.

3.3.5. Changer des noms d'hôte à distance

Pour exécuter une commande **hostnamectl** sur un système distant, utiliser l'option **-H, --host**, comme suit :

```
~]# hostnamectl set-hostname -H [username]@hostname
```

Quand le *hostname* correspond à l'hôte distant que vous souhaitez configurer. Le *username* est en option. L'outil **hostnamectl** utilisera **SSH** pour se connecter au système distant.

3.4. CONFIGURER DES NOMS D'HÔTES PAR NMCLI

L'outil de **NetworkManager nmtui** peut être utilisé pour interroger et définir le nom d'hôte statique dans le fichier **/etc/hostname**. Notez qu'au moment de la rédaction, changer le nom de cette façon ne sera pas remarqué par le **hostnamectl**.

Pour interroger le nom d'hôte statique, exécuter la commande suivante :

```
~]$ nmcli general hostname
```

Pour définir le nom d'hôte statique à *my-server*, exécutez la commande suivante en tant qu'utilisateur **root** :

```
~]# nmcli general hostname my-server
```

Pour forcer le **hostnamectl** à remarquer le changement de nom d'hôte statique, démarrer à nouveau le **hostnamed** en tant qu'utilisateur **root**:

```
~]# systemctl restart systemd-hostnamed
```

3.5. RESSOURCES SUPPLÉMENTAIRES

Les sources d'informations suivantes fournissent des ressources supplémentaires à propos de **hostnamectl**.

3.5.1. Documentation installée

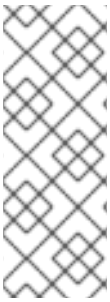
- Page man **hostnamectl(1)** — décrit **hostnamectl** y compris les commandes et les options de commandes.
- Page man **hostname(1)** — contient une explication sur les commandes **hostname** et **domainname**.
- Page man **hostname(5)** — contient une explication sur le fichier de nom d'hôte, son contenu et son utilisation.
- Page man **hostname(7)** — contient une explication de la résolution d'un nom d'hôte.
- Page man **machine-info(5)** — décrit le fichier d'information de la machine locale et les variables d'environnement qu'il contient.
- Page man **machine-id(5)** — décrit le fichier de configuration de l'ID de la machine locale.
- Page man **systemd-hostnamed.service(8)** — décrit le service système **systemd-hostnamed** utilisé par **hostnamectl**.

CHAPITRE 4. CONFIGURER NETWORK BONDING

Red Hat Enterprise Linux 7 permet aux administrateurs de relier plusieurs interfaces de réseaux entre elles en un seul canal unifié. Le canal de liaison permet à deux ou à plusieurs interfaces de réseaux d'agir un tant que canal unique, augmentant ainsi le débit et vérifiant la redondance.

AVERTISSEMENT

L'utilisation des connexions directes par cable sans commutateurs de réseau n'est pas prise en charge par le Network Bonding. Le mécanisme de basculement décrit ici ne fonctionnera pas comme prévu sans la présence de commutateurs de réseaux. Voir l'article de base de connaissance *Why is bonding in not supported with direct connection using crossover cables?* (Pourquoi Network Bonding ne prend il pas en charge les connexions directes avec des câbles croisés ?) pour obtenir plus d'informations.



NOTE

Les modes de sauvegarde active active-backup, balance-tlb et balance-alb ne nécessitent aucune configuration spécifique du commutateur. Quant aux autres modes de liaison, il faut la configuration du commutateur pour agréger les liens. Par exemple, un commutateur Cisco exige EtherChannel pour les Modes 0, 2 et 3, mais pour le Mode 4, il faut LACP et EtherChannel. Voir la documentation fournie avec votre commutateur et https://www.kernel.org/doc/Documentation/networking/bonding.txt

4.1. COMPRENDRE LES COMPORTEMENTS PAR DÉFAUT DES INTERFACES MAÎTRES ET ESCLAVES

À chaque fois que vous contrôlez les interfaces esclaves liées par le démon **`NetworkManager`**, surtout quand vous cherchez des fautes, gardez à l'esprit les faits suivants :

1. Démarrer l'interface maître ne démarre pas les interfaces esclaves automatiquement.
2. Démarrer l'interface esclave ne démarre pas toujours l'interface maître.
3. Stopper l'interface maître stoppe également les interfaces esclaves.
4. Une interface maître sans esclaves peut démarrer les connexions **`IP`** statiques.
5. Une interface maître sans esclaves attend les esclaves avant de démarrer les connexions **`DHCP`**.
6. Une interface maître avec une connexion **`DHCP`** en attente d'esclaves se termine quand un esclave accompagné d'un transporteur est ajouté.
7. Une interface maître avec une connexion **`DHCP`** en attente d'esclaves continue quand un esclave sans transporteur est ajouté.

4.2. CONFIGURER NETWORK BONDING PAR L'INTERFACE TEXTE UTILISATEUR, NMTUI

L'outil d'interface utilisateur de texte **nmtui** peut être utilisé pour configurer le Network Bonding dans une fenêtre de terminal. Exécutez la commande suivante pour démarrer l'outil :

```
~]$ nmtui
```

L'interface utilisateur texte apparaîtra. Tout commande non valide affichera un message d'utilisation.

Pour naviguer, utiliser les flèches ou appuyer sur **Tab** pour continuer et appuyer sur la combinaision de touches **Maj**+**Tab** pour revenir aux options. Appuyer sur la touche **Entrée** pour sélectionner une option. La barre **Espace** active/désactive le statut d'un case à cocher.

1. À partir du menu de démarrage, sélectionner **Modifier une connexion**. Sélectionner **Ajouter**, l'écran **Nouvelle connexion** apparaîtra.

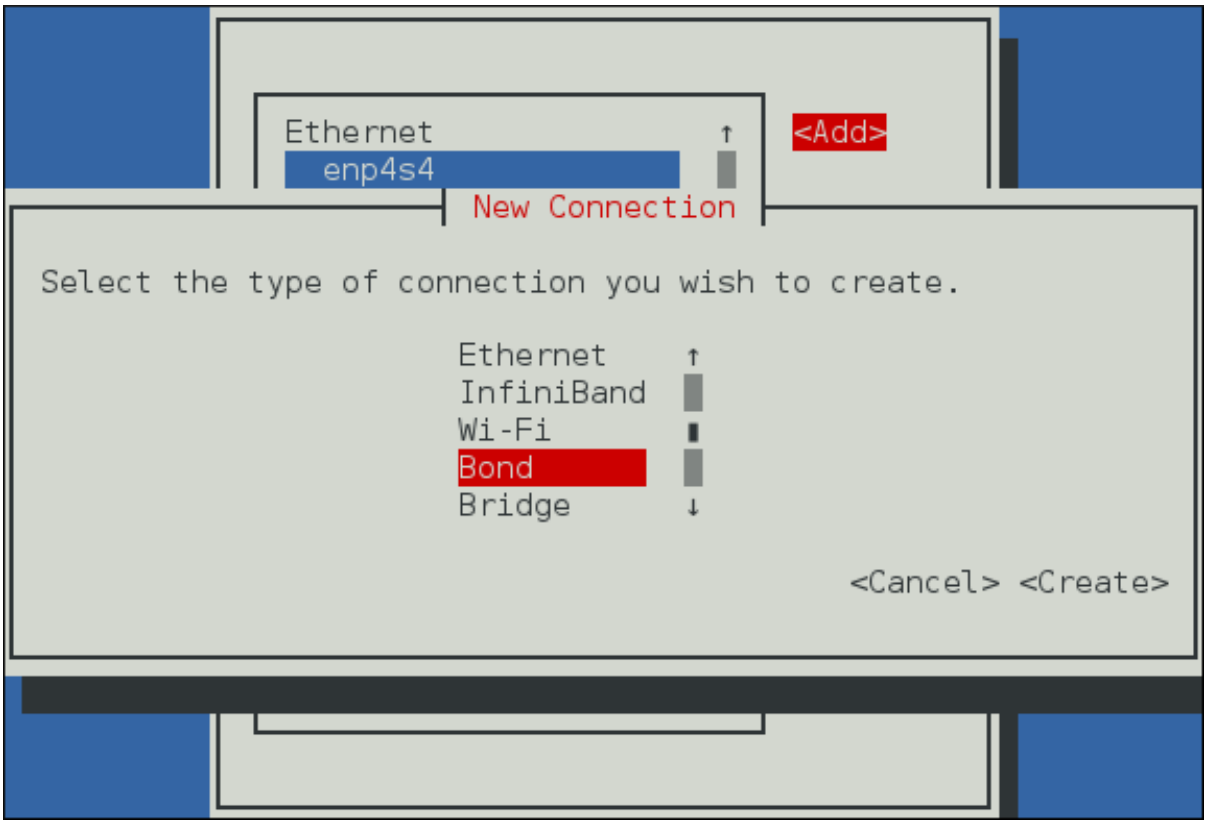


Figure 4.1. Pour que l'interface utilisateur texte du NetworkManager ajoute un menu de connexion de liaison

2. Sélectionner **Bond**, puis **Create** ; l'écran **Edit connection** de Network Bonding appraîtra.

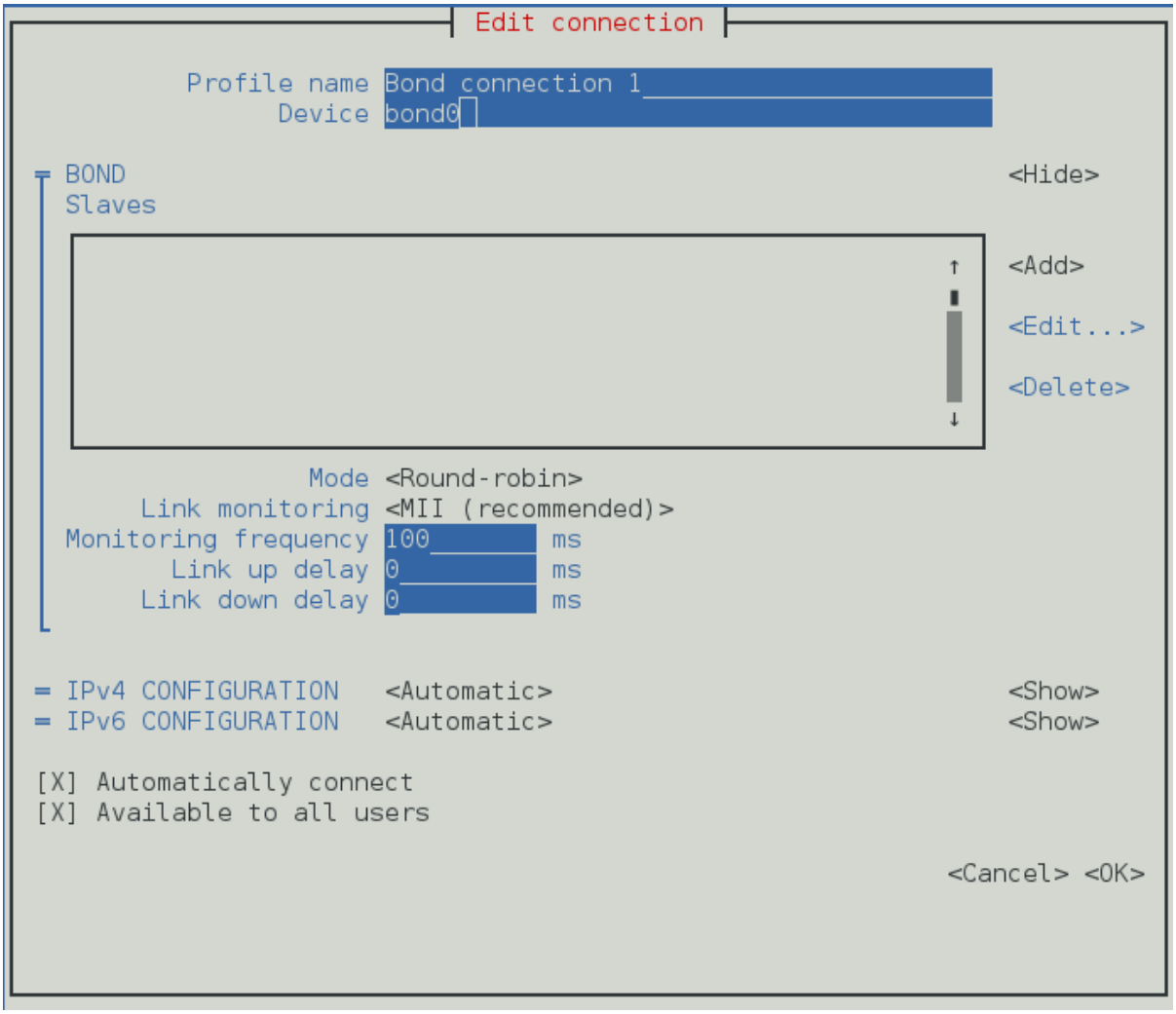


Figure 4.2. Pour que l'interface utilisateur texte du NetworkManager configure un menu de connexion de liaison

3. À ce moment là, les interfaces esclaves aront besoin d'être ajoutées à la liaison : pour les ajouter, sélectionner **Ajouter**, et l'écran **Nouvelle connexion** apparaîtra. Une fois que le type de connexion aura été sélectionné, appuyer sur le bouton **Créer**.

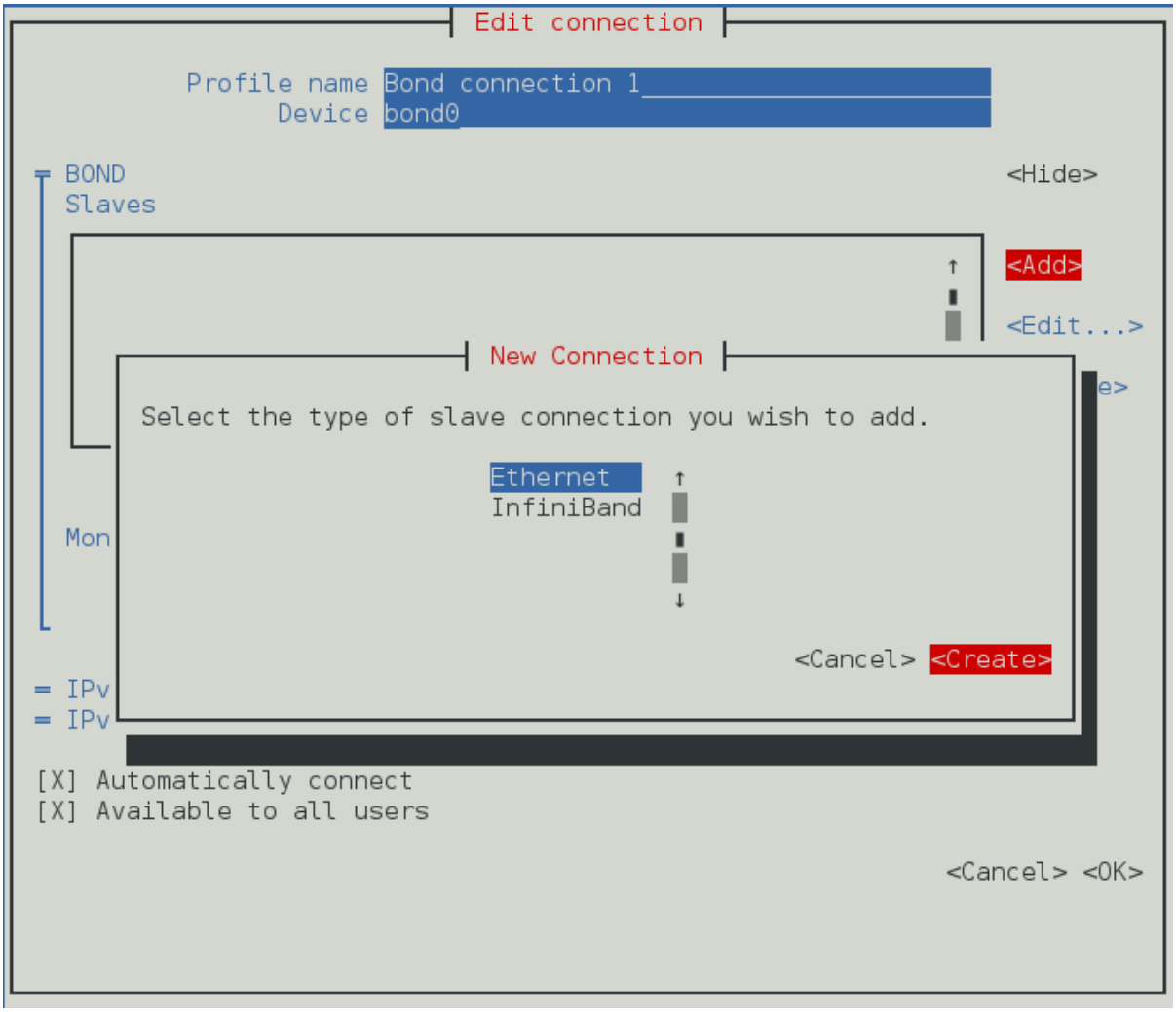


Figure 4.3. Pour que l'interface utilisateur texte du NetworkManager configure un nouveau menu de connexion de liaison esclave

4. Le bouton **Modifier la connexion** de l'esclave apparaît : saisir alors l'adresse MAC ou le nom de périphérique de l'esclave que vous aurez choisi dans la section **Périphériques**. Si besoin est, saisir une adresse MAC clonée à utiliser comme adresse MAC de liaison, en sélectionnant **Afficher** à droite de l'étiquette **Ethernet**. Sélectionnez le bouton **OK** pour sauvegarder l'esclave.

NOTE

Si le périphérique est spécifié sans adresse MAC, la section **Périphérique** sera remplie automatiquement une fois que la fenêtre **Modifier Connexion** sera chargée, mais uniquement s'il trouve le périphérique.

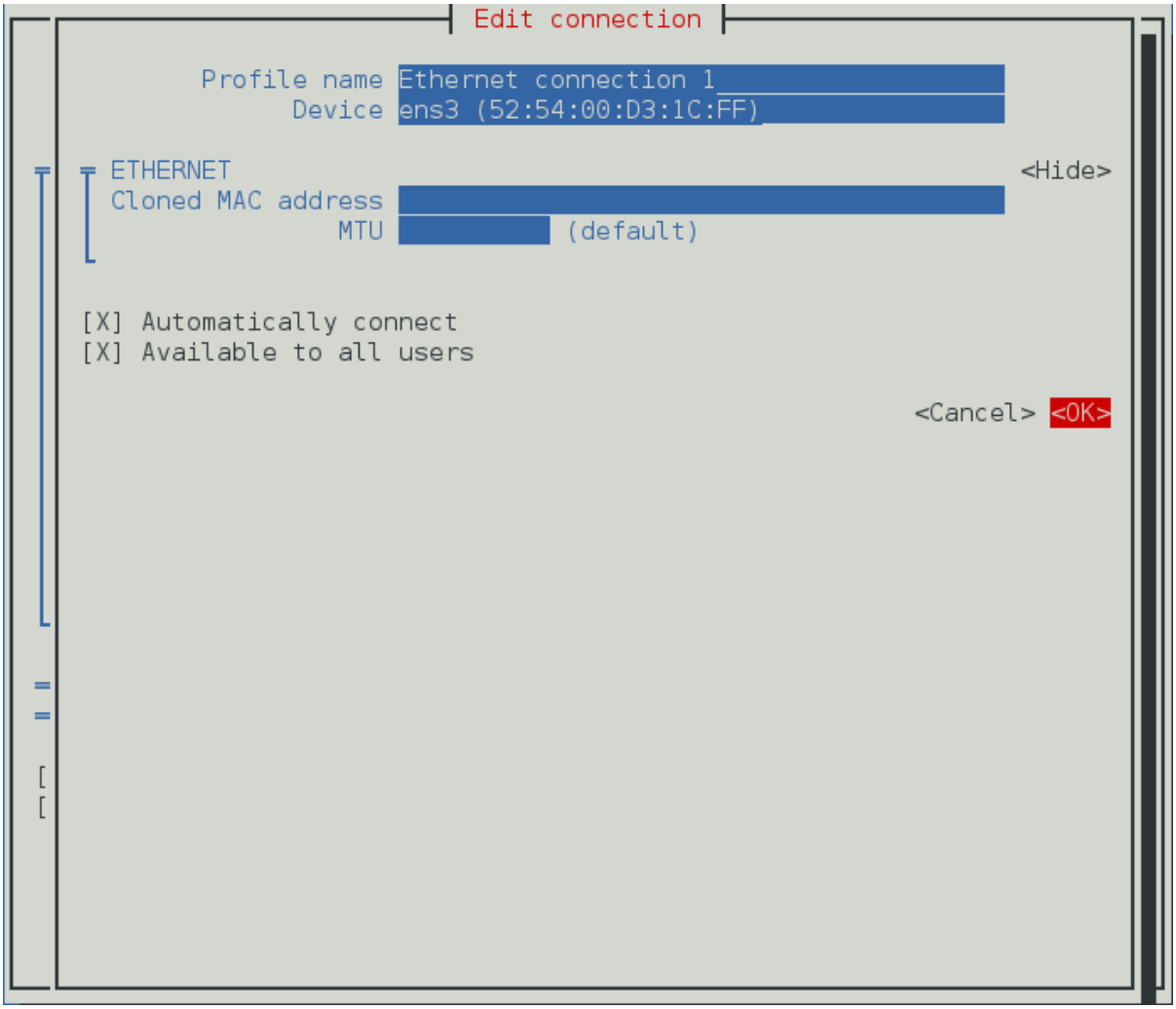


Figure 4.4. Pour que l'interface texte d'utilisateur du NetworkManager configure un menu de connexion de liaison esclave

5. Le nom de la liaison esclave apparaît dans la section **`Slaves`**. Répétez les étapes ci-dessus pour ajouter des connexions esclaves.

6. Vérifier et confirmer les paramètres de configuration, avant de cliquer sur le bouton **`OK`**.

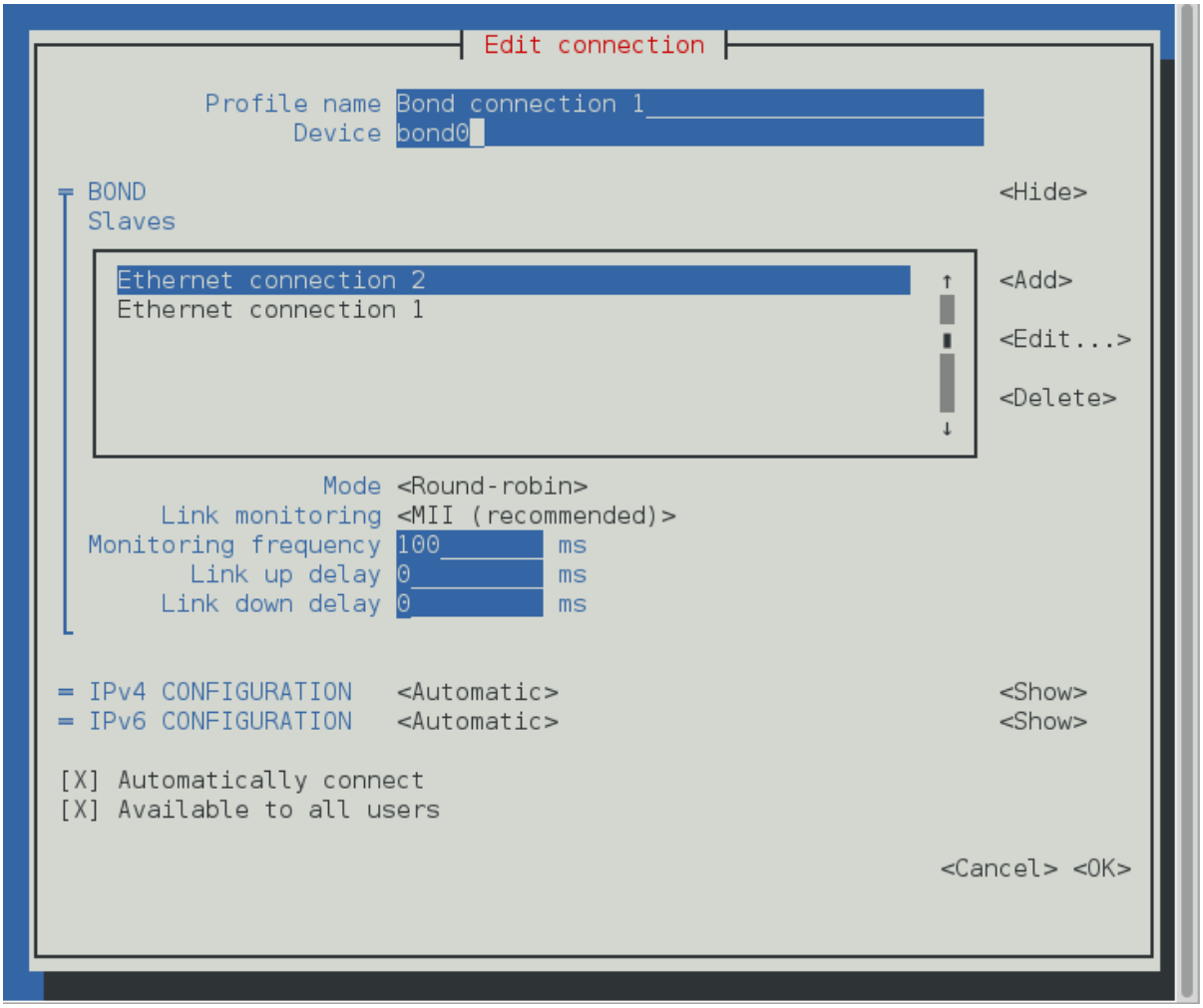


Figure 4.5. Pour que l'interface texte d'utilisateur du NetworkManager termine la liaison

Voir Section 4.6.1.1, « Configurer l'onglet Liaisons » pour obtenir des définitions sur les termes attenants aux liaisons.

Voir Section 1.5, « Configuration réseau utilisant une interface utilisateur texte (nmtui) » pour obtenir des informations sur la façon d'installer **nmtui**.

4.3. UTILISER L'OUTIL DE LIGNE DE COMMANDES DU NETWORKMANAGER, NMCLI

Pour créer une liaison, nommée *mybond0*, exécutez une commande comme suit :

```
~]$ nmcli con add type bond con-name mybond0 ifname mybond0 mode active-
backup
Connection 'mybond0' (9301ff97-abbc-4432-aad1-246d7faea7fb) successfully
added.
```

Pour ajouter une interface esclave, exécutez une commande du style :

```
~]$ nmcli con add type bond-slave ifname ens7 master mybond0
```

Pour ajouter des esclaves supplémentaires, répétez la commande précédente avec une nouvelle interface. Exemple :

```
~]$ nmcli con add type bond-slave ifname ens3 master mybond0
Connection 'bond-slave-ens3-1' (50c59350-1531-45f4-ba04-33431c16e386)
successfully added.
```

Notez que comme aucun **con-name** n'a été donné pour les eslaves, le nom a été dérivé du nom de l'interface ajouté au type. Au moment de la rédaction, **nmcli** ne prend en charge que les esclaves Ethernet.

Pour qu'une liaison apparaisse, les esclaves doivent tout d'abord apparaître comme suit :

```
~]$ nmcli con up bond-slave-ens7
Connection successfully activated (D-Bus active path:
/org/freedesktop/NetworkManager/ActiveConnection/14)
```

```
~]$ nmcli con up bond-slave-ens3
Connection successfully activated (D-Bus active path:
/org/freedesktop/NetworkManager/ActiveConnection/15)
```

Faîtes surgir la liaison ainsi :

```
~]$ nmcli con up bond-mybond0
Connection successfully activated (D-Bus active path:
/org/freedesktop/NetworkManager/ActiveConnection/16)
```

Voir Section 2.3, « Utiliser l'outil de ligne de commandes du NetworkManager, nmcli » pour une introduction à **nmcli**

4.4. UTILISATION DE L'INTERFACE EN LIGNE DE COMMANDES (CLI)

Une liaison qui utilise le module de noyau de liaison (**bonding**) et une interface de réseau spéciale appelée *interface de canal de liaison* sont créées.

4.4.1. Vérifier si le module de noyau de liaison est installé

Dans Red Hat Enterprise Linux 7, le module de liaison est téléchargé par défaut. Si nécessaire, vous pouvez télécharger le module en exécutant la commande suivante en tant qu'utilisateur **root** :

```
~]# modprobe --first-time bonding
```

Cette activation ne pourra pas persister à travers les redémarrages de système. Voir *MAJOROSVER; System Administrator's Guide* (Guide d'administrateur de systèmes) pour une explication sur le chargement de modules persistants. Notez que s'il y a un fichier de configuration correct qui utilise la directive **BONDING_OPTS**, le module Bonding sera téléchargé selon les besoins et n'aura donc pas besoin d'être téléchargé séparemment.

Pour afficher des informations sur le module, excuter la commande suivante :

```
 ~]$ modinfo bonding
```

Consulter la page man **modprobe(8)** pour plus d'options de commandes.

4.4.2. Créer une interface de canal de liaison

Pour créer une interface de canal de liaison, créer un fichier dans le répertoire **/etc/sysconfig/network-scripts/** nommé **ifcfg-bond*N***, et remplacer *N* par le numéro de l'interface, par exemple **0**.

Le contenu du fichier peut être basé sur un fichier de confiuguration ou n'importe quel type d'interface en cours de processus, comme l'interface Ethernet. Les différences principales étant que la directive **DEVICE**est **bond*N***, que l'on remplace *N* par le numéro d'interface, et **TYPE=Bond**. De plus, définir **BONDING_MASTER=yes**.

Exemple 4.1. Exemple de fichier de configuration d'interface ifcfg-bond0

Exemple d'interface de canal de liaison

```
DEVICE=bond0
NAME=bond0
TYPE=Bond
BONDING_MASTER=yes
IPADDR=192.168.1.1
PREFIX=24
ONBOOT=yes
BOOTPROTO=none
BONDING_OPTS="bonding parameters separated by spaces"
```

La directive NAME est utile pour nommer le profil de connexion dans le **NetworkManager**. ONBOOT indique si le profil doit être démarré à l'amorçage (ou plus généralement, quand le périphérique est auto-connecté).

IMPORTANT

Les paramètres du module de noyau de liaison doit être spécifié dans une liste séparée par des virgules dans la directive **BONDING_OPTS="*bonding parameters*"** du fichier de l'interface **ifcfg-bond*N***. Ne *pas* spécifier d'options pour le périphérique de liaison dans **/etc/modprobe.d/*bonding*.conf**, ou dans le fichier obsolète **/etc/modprobe.conf**.

Le paramètre **max_bonds** n'est pas spécifique à l'interface et ne doit pas être défini quand on utilise les fichiers **ifcfg-bond*N*** avec la directive **BONDING_OPTS** car cette directive amènera les scripts de réseau à créer des interfaces de liaison selon les besoins.

Pour obtenir plus d'informations et des conseils sur la façon de configurer le module de liaison, et pour voir la liste des paramètres de liaison, consulter Section 4.5, « Utiliser une liaison de canal ».

4.4.3. Création d'interfaces SLAVE

L'interface de canal de liaison est le « master » et les interfaces de liaison sont appelées les « esclaves ». Une fois que l'interface de liaison de canaux est créée, les interfaces réseau devant être reliées ensemble doivent être configurées en ajoutant les directives **MASTER** et **SLAVE** aux fichiers de configuration des esclaves. Les fichiers de configuration pour chaque interface esclave peuvent être presque identiques.

Exemple 4.2. Exemple de fichier de configuration d'interface esclave

Ainsi, si deux interfaces Ethernet sont reliées par canaux, **eth0** and **eth1**, elles peuvent toutes les deux ressembler à ce qui suit :

```
DEVICE=ethN
NAME=bond0-slave
TYPE=Ethernet
BOOTPROTO=none
ONBOOT=yes
MASTER=bond0
SLAVE=yes
```

Dans cet exemple, remplacez *N* par la valeur numérique de l'interface. Notez que s'il y a plus d'un seul profil ou fichier de configuration ayant **ONBOOT=yes** pour interface, ils risquent de rentrer en compétition et un profil **TYPE=Ethernet** risque d'être activé à la place d'une liaison esclave.

4.4.4. Activer une liaison de canaux

Pour activer une liaison, faîtes apparaître les esclaves. Puis, en tant qu'utilisateur **root**, exécutez la commande suivante :

```
~]# ifup ifcfg-eth0
Connection successfully activated (D-Bus active path:
/org/freedesktop/NetworkManager/ActiveConnection/7)
```

```
~]# ifup ifcfg-eth1
Connection successfully activated (D-Bus active path:
/org/freedesktop/NetworkManager/ActiveConnection/8)
```

Notez que si vous modifiez des fichiers d'interface pour des interfaces qui sont actuellement en ligne (« up »), commencez par les définir ainsi :

```
ifdown ethN
```

. Puis, une fois terminé, faites apparaître les esclaves, ce qui fera apparaître la liaison (sauf si « down »).

Pour rendre le **NetworkManager** au fait de ces changements, exécutez une commande pour chaque interface modifée, en tant qu'utilisateur **root**:

```
~]# nmcli con load /etc/sysconfig/network-scripts/ifcfg-device
```

Sinon, pour télécharger à nouveau toutes les interfaces, exécutez :

```
~]# nmcli con reload
```

Le comportement par défaut est pour **NetworkManager** de ne pas être au courant des changements et de continuer à utiliser les anciennes données de configuration. C'est défini par l'option **monitor-connection-files** dans le fichier **NetworkManager.conf**. Voir la page de manuel **NetworkManager.conf(5)** pour plus d'informations.

Pour voir le statut de l'interface de liaison, exécutez la commande suivante :

```
~]# ip link show
```

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode
DEFAULT
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
master bond0 state UP mode DEFAULT qlen 1000
    link/ether 52:54:00:e9:ce:d2 brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
master bond0 state UP mode DEFAULT qlen 1000
    link/ether 52:54:00:38:a6:4c brd ff:ff:ff:ff:ff:ff
4: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 1500 qdisc noqueue
state UP mode DEFAULT
    link/ether 52:54:00:38:a6:4c brd ff:ff:ff:ff:ff:ff
```

4.4.5. Création de plusieurs liaisons

Dans Red Hat Enterprise Linux 7, pour chaque liaison, une interface de canal de liaison est créée, comprenant la directive **BONDING_OPTS**. Cette méthode de configuration est utilisée pour que plusieurs périphériques de liaison puissent avoir des configurations différentes. Pour créer plusieurs interfaces de canaux de liaison, procédez ainsi :

- Créez plusieurs fichiers **ifcfg-bond*N*** avec la directive **BONDING_OPTS** car cette directive amènera les scripts de réseau à créer des interfaces de liaison selon les besoins.
- Créez ou modifiez des fichiers de configuration d'interface existants à relier, et inclure la directive **SLAVE**.
- Assignez les interfaces à relier et les interfaces esclaves aux canaux de liaison par la directive **MASTER**.

Exemple 4.3. Exemples de fichiers de configuration d'interface ifcfg-bondN

Ce qui suit est un exemple de fichier de configuration d'interface de canal de liaison :

```
DEVICE=bondN
NAME=bondN
TYPE=Bond
BONDING_MASTER=yes
IPADDR=192.168.1.1
PREFIX=24
ONBOOT=yes
BOOTPROTO=none
BONDING_OPTS="bonding parameters separated by spaces"
```

Dans cet exemple, remplacer *N* par le numéro de l'interface de liaison. Par exemple, pour créer deux liaisons, créez deux fichers de configuration, **ifcfg-bond0** et **ifcfg-bond1**, avec les adresses **IP** qui conviennent.

Créer les interfaces à relier en suivant les explications suivantes Exemple 4.2, « Exemple de fichier de configuration d'interface esclave » et les assigner aux interfaces de liaison, selon les besoins, en utilisant la directive **MASTER=bond*N***. Ainsi, en continuant avec l'exemple ci-dessus, si on a besoin de deux interfaces par liaison, alors deux liaisons créeront quatre fichiers de configuration d'interface. Assigner les deux premières à **MASTER=bond*0***, et les deux suivanes à **MASTER=bond*1***.

4.5. UTILISER UNE LIAISON DE CANAL

Pour améliorer les performances, ajuster les options de modules disponibles pour déterminer quelle combinaison fonctionne le mieux. Prêtez une attention particulière à **miimon** ou à **arp_interval**, et au paramètre **arp_ip_target**. Voir Section 4.5.1, « Relier des propriétés utilisateur » pour obtenir une liste des options disponibles et comment déterminer rapidement les meilleurs pour votre interface liée.

4.5.1. Relier des propriétés utilisateur

C'est une bonne idée de tester quels paramètres de module de liaison de canaux fonctionnent le mieux pour vos interfaces reliées avant de les ajouter à la directive **BONDING_OPTS="*bonding parameters*"** de votre fichier de configuration d'interface de liaison (par exemple **ifcfg-bond0**). Les paramètres d'interfaces reliés peuvent être configurés sans décharger (ou recharger) le module de liaison en manipulant des fichiers dans le système de fichiers **sysfs**.

sysfs est un système de fichiers virtuels qui représente des objets de noyau en tant que répertoires, fichiers, et liens symboliques. **sysfs** peut être utilisé pour chercher des informations sur les objets de noyau, et peut aussi manipuler ces objets par l'utilisation de commandes système de fichiers normaux. Le système de fichiers virtuel **sysfs** est monté sous le répertoire **/sys/**. Toutes les interfaces de liaison peuvent être configurées dynamiquement par interaction et en manipulant des fichiers dans le répertoire **/sys/class/net/**.

Afin de déterminer les meilleurs paramètres pour votre interface de liaison, créez un fichier d'interface de liaison de canaux comme **ifcfg-bond0** en suivant les instructions de Section 4.4.2, « Créer une interface de canal de liaison ». Insérer les directives **SLAVE=yes** et **MASTER=bond0** dans les fichiers de configuration pour chaque interface reliée à **bond0**. Une fois terminé, vous pouvez continuer à tester les paramètres.

Commencez par appeler la liaison que vous venez de créer en exécutant **ifup bond*N*** en tant qu'utilisateur **root**:

```
~]# ifup bond0
```

Si vous avez créé le fichier d'interface de liaison **ifcfg-bond0**, vous pourrez apercevoir **bond0** dans la sortie de la commande **ip link show** en tant qu'utilisateur **root** :

```
~]# ip link show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode
DEFAULT
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
master bond0 state UP mode DEFAULT qlen 1000
    link/ether 52:54:00:e9:ce:d2 brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
master bond0 state UP mode DEFAULT qlen 1000
    link/ether 52:54:00:38:a6:4c brd ff:ff:ff:ff:ff:ff
4: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 1500 qdisc noqueue
state UP mode DEFAULT
    link/ether 52:54:00:38:a6:4c brd ff:ff:ff:ff:ff:ff
```

Pour apercevoir toutes les liaisons existantes, même si elles ne sont pas actives, exécutez :

```
~]$ cat /sys/class/net/bonding_masters
bond0
```

Vous pouvez confiugurer chaque liaison individuellement en manipulant des fichiers qui se trouvent dans **/sys/class/net/bond*N*/bonding/**. Commencez par désactiver la liaison que vous configurez :

```
~]# ifdown bond0
```

Par exemple, pour activer le monitoring MII sur le bond0 avec une seconde d'intervalle, exécutez en tant qu'utilisateur **root** :

```
~]# echo 1000 > /sys/class/net/bond0/bonding/miimon
```

Pour configurer le bond0 en mode ***balance-alb***, exécutez soit :

```
~]# echo 6 > /sys/class/net/bond0/bonding/mode
```

... ou, utiliser le nom du mode :

```
~]# echo balance-alb > /sys/class/net/bond0/bonding/mode
```

Après avoir configuré les options pour la liaison en question, vous pouvez l'activer et la tester en exécutant **ifup bond*N***. Si vous décidez de changer les options, désactivez l'interface, modifiez ses paramètres à l'aide de **sysfs**, réactivez-la et testez-la à nouveau.

Une fois que vous aurez déterminé le meilleur jeu de paramètres pour votre liaison, ajouter ces paramètres sous forme de liste séparée par des espaces à la directive ***BONDING_OPTS*** **=** du fichier **/etc/sysconfig/network-scripts/ifcfg-bond *N*** pour l'interface de liaison que vous configurez. Chaque fois que ce lien est activé (par exemple, par le système au cours de la séquence de démarrage quand la directive ***ONBOOT=yes*** est définie), les options de liaison spécifiées dans ***BONDING_OPTS*** prendront effet pour cette liaison.

La liste suivante fournit les noms de nombreux paramètres de liaison de canaux connus, ainsi qu'une description de ce qu'ils font. Pour plus d'informations, voir la brève description de chaque **parm** en sortie de **modinfo collage**, ou pour plus d'informations, voir https://www.kernel.org/doc/Documentation/networking/bonding.txt.

Liaisons de paramètres d'interface

ad_select=*value*

Indique la logique de sélection d'aggrégation 802.3ad à utiliser. Voici les valeurs possibles :

- **stable** ou **0** — valeur par défaut. L'agrégateur actif est choisi par la plus grande bande passante globale. Une nouvelle sélection de l'agrégateur actif a lieu uniquement lorsque tous les esclaves de l'agrégateur actif sont désactivés ou si l'agrégateur actif n'a aucun esclave.
- **bandwidth** ou **1** — l'agrégateur est choisi par la plus grande bande passante globale. Une resélection a lieu à nouveau si :
  - Un esclave est ajouté ou supprimé d'une liaison ;
  - Un état de lien d'esclave change ;
  - Un état d'association 802.3ad d'esclave change ;

- L'état administratif de la liaison change à actif.

- **count** ou **2** — l'agrégateur est choisi par le plus grand nombre d'esclaves. La resélection a lieu comme expliqué dans la configuration de **bandwidth** ci-dessus.

Les politiques de sélection de **bandwidth** et de **count** permettent le basculement des agrégations 802.3ad en cas de panne partielle de l'agrégateur actif. Cela permet de maintenir l'agrégateur en disponibilité maximale, soit en bande passante ou en nombre d'esclaves, actif en permanence.

arp_interval=*time_in_milliseconds*

Indique la fréquence du contrôle **ARP**, en millisecondes.

IMPORTANT

Il est essentiel que les paramètres **arp_interval** et **arp_ip_target** soient spécifiés, ou bien, que le paramètre **miimon** soit spécifié. Si tel n'est pas le cas, on risque la dégradation de la performance réseau si un lien échoue.

Si vous utilisez cette confiuguration en **mode=0** ou en **mode=2** (les deux modes d'équilibrage des charges), le commutateur de réseau doit être configuré pour distribuer des paquets équitablement entre les cartes réseau. Pour plus d'informations sur la façon de procéder, voir https://www.kernel.org/doc/Documentation/networking/bonding.txt.

La valeur est définie à **0** par défaut, ce qui a pour effet de le désactiver.

arp_ip_target=*ip_address*[,*ip_address_2*,…*ip_address_16*]

Indique l'adresse **IP** cible des requêtes **ARP** quand le paramètre **arp_interval** est activé. On peut spécifier jusqu'à 16 adresses **IP** dans une liste séparée par des virgules.

arp_validate=*value*

Validater la source/distribution des sondes **ARP** ; la valeur par défaut est **none**. Autres valeurs possibles **active**, **backup**, et **all**.

downdelay=*time_in_milliseconds*

Indique (en millisecondes) la durée à attendre suite à un échec de lien avant de désactiver ce lien. La valeur doit correspondre à un multiple de la valeur spécifiée dans le paramètre **miimon**. La valeur est définie à **0** par défaut, pour la désactivation.

fail_over_mac=*value*

Spécifie si le mode active-backup doit définir tous les esclaves à la même adresse MAC au moment de la mise en esclavage (le comportement traditionnel), ou, lorsque activé, effectuer un traitement spécial de l'adresse MAC de la liaison conformément à la stratégie sélectionnée. Les valeurs possibles sont :

- **none** ou **0** — Valeur par défaut. Cette configuration désactive **fail_over_mac**, et cause le processus de liaison à définir tous les esclaves d'une liaison d'active-backup à la même adresse au moment de la mise en esclavage.

- **active** or **1** — les politiques « active » **fail_over_mac** indiquent que l'adresse MAC de la liaison doivent toujours correspondre à l'adresse MAC de l'esclave actif actuellement. L'adresse MAC des esclaves ne change pas, mais l'adresse MAC de la liaison change en

cas d'échec.

Cette politique s'avère utile pour les périphériques qui ne peuvent jamais changer d'adresse MAC, ou pour les périphériques qui refusent les émissions entrantes avec leur propre source MAC (qui interfère avec le moniteur de l'ARP). L'inconvénient de cette politique est que chaque périphérique sur le réseau doit être mis à jour via ARP spontané, contrairement à la méthode normale de commutateurs qui surveillent le trafic entrant pour mettre à jour leurs tables ARP. Si l'ARP spontané est perdu, la communication peut être perturbée.

Lorsque cette stratégie est utilisée en conjonction au moniteur MII, les périhériques, qui affirment un lien avant de réellement transmettre ou recevoir, sont particulièrement sensibles à la perte de l'ARP spontané, et définir un délai en amont qui convient peut d'avérer utile.

- **`follow`** ou **`2`** — les causes de la politique « follow » **`fail_over_mac`** amènent l'adresse MAC de la liaison à être sélectionnée normalement (l'adresse MAC du premier esclave ajouté à la liaison). Cependant, le deuxième et des esclaves suivants ne sont pas définis à cette adresse MAC quand ils sont en rôle de sauvegarde ; un esclave est programmé avec l'adresse MAC de la liaison au moment du basculement (et l'esclave anciennement actif reçoit l'adresse MAC de l'esclave nouvellement actif).

  Cette politique est utile pour les périphériques multiports, qui sont soit perturbés ou victimes d'une pénalité de performance quand des ports mulitples sont programmés sur une même adresse MAC.

lacp_rate=*value*

Indique le taux auquel les partenaires de liens doivent transmettre les paquets LACPDU en mode 802.3ad. Les valeurs possibles sont les suivantes :

- **`slow`** or **`0`** — valeur par défaut. Indique que les partenaires doivent transmettre les LACPDU toutes les 30 secondes.
- **`fast`** or **`1`** — indique que les partenaires doivent transmettre les LACPDU après chaque seconde qui s'écoule

`miimon=`*`time_in_milliseconds`*

Indique (en millisecondes) la fréquence de la surveillance de liens MII. Cela est utile dans les situations de haut débit car MII est utilisé pour vérifier si la carte réseau est active. Pour vérifier que le pilote d'une carte réseau particulière prenne bien en charge l'outil MII, saisir la commande suivante en tant qu'utilisateur root :

```
~]# ethtool interface_name | grep "Link detected:"
```

Avec cette commande, remplacez *interface_name*> par le nom de l'interface du périphérique, comme par exemple **`eth0`**, mais pas l'interface de liaison. Si MII est pris en charge, la commande renverra :

```
Liens détectés : oui
```

Si vous utilisez une interface liée en haut débit, le module de chaque carte réseau (NIC) doit prendre l'outil MII en charge. Définir la valeur à **`0`** (la valeur par défaut), stoppe cette fonctionnalité. Quand on configure ce paramètre, il est bon de commencer par la valeur **`100`**.

IMPORTANT

Il est essentiel que les paramètres **arp_interval** et **arp_ip_target** soient spécifiés, ou bien, que le paramètre **miimon** soit spécifié. Si tel n'est pas le cas, on risque la dégradation de la performance réseau si un lien échoue.

mode=*value*

Vous permet de spécifier la politique de mise en liaison. La *value* peut correspondre à une des possibilités suivantes :

- **balance-rr** ou **0** — définit une politique round-robin de répartition des charges et de diffusion de tolérance d'erreurs. Toutes les transmissions sont reçues et diffusées sur chaque interface esclave liée, en commençant par la première disponible
- **active-backup** or **1** — définit une politique active-backup de tolérance d'erreurs. Toutes les transmissions sont reçues et diffusées en commençant par la première interface disponible. Une autre interface esclave liée n'est utilisée que si l'interface esclave liée échoue.
- **balance-xor** ou **2** — les transmissions sont basées sur la politique de hachage sélectionnée. La valeur par défaut consiste à tirer un hachage par l'OUX (XOR) des adresses MAC source et de destination, multiplié par le modulo du nombre d'interfaces d'esclave. Dans ce mode, le trafic destiné à des homologues spécifiques sera toujours envoyé sur la même interface. Comme la destination est déterminée par l'adresse MAC, cette méthode fonctionne mieux pour le trafic en direction d'homologues sur le même lien ou réseau local. Si le trafic doit passer par un routeur unique, alors ce mode d'équilibrage de trafic sera sous-optimal.
- **broadcast** ou **3** — définit une politique de diffusion de tolérance d'erreurs. Toutes les transmissions sont diffusées sur toutes les interfaces esclaves.
- **802.3ad** ou **4** — définit une politique d'agrégation de liens dynamiques IEEE 802.3ad. Crée des groupes d'agrégation qui partagenent les mêmes configurations en matière de vitesse et de duplex. Transmet ou reçoit sur tous les esclaves dans l'agrégateur actif. Requiert un commutateur conforme à 802.3ad.
- **balance-tlb** ou **5** — définit une politique Transmit-Load-Balancing (TLB) de tolérance aux pannes et d'équilibrage des charges. Le trafic sortant est distribué selon la charge en cours sur chaque interface esclave. Le trafic entrant est reçu par l'esclave actuel. Si l'esclave récepteur échoue, un autre esclave reprend l'adresse MAC de l'esclave qui a échoué. Ce mode convient uniquement aux adresses locales connues du module de liaison du noyau et ne peut donc être utilisé pour un pont avec des machines virtuelles.
- **balance-alb** ou **6** — définit une politique Adaptive-Load-Balancing (ALB) de tolérance aux pannes et d'équilibrage des charges. Inclut la transmission et la réception de l'équilibrage des charges pour le trafic **IPv4**. La réception de l'équilibrage des charges est effectuée par négociation **ARP** Ce mode convient uniquement aux adresses locales connues du module de liaison du noyau et ne peut donc être utilisé pour un pont avec des machines virtuelles.

primary=*interface_name*

Indique le nom de l'interface, par exemple **eth0**, du périphérique principal. Le périphérique **primaire** correspondra à la première parmi les interfaces de liaison à être utilisée et ne sera pas abandonnée, sauf en cas d'échec. Ce paramètre est particulièrement utile lorsqu'une carte réseau de l'interface de liaison est plus rapide et, par conséquent, capable de supporter une charge plus élevée.

Ce paramètre de configuration n'est valide que quand l'interface de liaison est en mode **active-backup**. Voir https://www.kernel.org/doc/Documentation/networking/bonding.txt pour plus d'informations.

primary_reselect=*value*

Spécifie la politique de resélection de l'esclave primaire. Cela affecte la façon dont l'esclave primaire est choisie pour devenir l'esclave actif en cas de défaillance de l'esclave actif ou de recouvrement de l'esclave primaire. Ce paramètre est conçu pour empêcher la volte-face entre l'esclave primaire et d'autres esclaves. Les valeurs possibles sont :

- **always** ou **0** (valeur par défaut) - l'esclave primaire devient l'esclave actif quand il revient.
- **better** ou **1** — l'esclave primaire devient l'esclave actif quand l'esclave est «up» à nouveau, si la vitesse et le duplex de l'esclave primaire sont supérieurs à la vitesse et au duplex de l'esclave actif actuel.
- **failure** ou **2** — l'esclave primaire devient actif si l'esclave actuel actif échoue et si l'esclave primaire est «up».

Le paramètre **primary_reselect** sera ignoré dans les deux cas suivants :

- Si aucun esclave n'est actif, le premier esclave rétabli devient un esclave actif.
- Après avoir été rendu esclave au début, l'esclave primaire est toujours l'esclave actif.

Modifier la politique **primary_reselect** via **sysfs** va entraîner une sélection immédiate du meilleur esclave actif selon la nouvelle politique. Cela risque ou non de résulter par un changement de l'esclave actif, selon les circonstances.

resend_igmp=*range*

Spécifie le nombre de rapports d'adhésion à IGMP à produire suite à un basculement. Un rapport d'adhésion est délivré immédiatement après le basculement ; les paquets suivants sont envoyés dans des intervalles de 200ms chacun.

Les valeurs varient entre **0** to **255**; la valeur par défaut est **1**. La valeur**0** empêche la parution du rapport d'adhésion à IMGP suite à un basculement.

Cette option est utile pour les modes de liaison **balance-rr** (mode 0), **active-backup** (mode 1), **balance-tlb** (mode 5) et **balance-alb** (mode 6), pour lesquels un basculement peut faire passer le trafic IGMP d'un esclave à une autre. C'est pourquoi un nouveau rapport IGMP doit être émis pour que le commutateur transmette le trafic IGMP entrant sur l'esclave nouvellement sélectionné.

updelay=*time_in_milliseconds*

Indique (en millisecondes) la durée à attendre pour activer un lien. La valeur doit correspondre à un multiple de la valeur spécifiée dans le paramètre **miimon**. La valeur est définie à **0** par défaut, pour la désactivation.

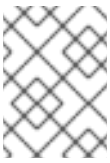
use_carrier=*number*

Spécifie si oui ou non le paramètre **miimon** doit utiliser MII/ETHTOOL ioctls ou **netif_carrier_ok()** pour déterminer le lien de l'état. La fonction **netif_carrier_ok()** dépend du pilote du périphérique pour maintenir son état **netif_carrier_*on/off***; la plupart des pilotes de périphériques supportent cette fonction.

Les outils MII/ETHTOOL ioctls utilisent une séquence d'appels dépréciés dans le noyau. Cependant, c'est toujours configurable si le pilote du périphérique ne supporte pas **netif_carrier_*on/off***.

Les valeurs acceptées sont les suivantes :

- **1** — configuration par défaut. Permet à l'utilisation de **netif_carrier_ok()**.
- **0** — permet l'utilisation de MII/ETHTOOL ioctls.



NOTE

Si l'interface de liaison insiste pour que le lien soit « up », il est possible que votre pilote de périphérique de réseau ne supporte pas **netif_carrier_*on/off***.

xmit_hash_policy=*value*

Sélectionne la politique de hachage de transmission utilisée pour la sélection des esclaves en modes **balance-xor** ou **802.3ad**. Les valeurs acceptées sont les suivantes :

- **0** or **layer2** — valeur par défaut. Ce paramètre utilise l'OUX (XOR) des adresses MAC de matériel pour générer le hachage. La formule utilisée est la suivante :

```
(source_MAC_address XOR destination_MAC) MODULO slave_count
```

Cet algorithme mettra les trafics dans un réseau homologue particulier sur le même esclave, et est conforme à 802.3ad.

- **1** ou **layer3+4** — utilise les informations de protocole des couches supérieures (si disponible) pour générer le hachage. Cela permet au trafic en direction d'un réseau homologue de s'étaler sur plusieurs esclaves, quoi qu'une simple connexion puisse s'étaler sur plusieurs esclaves.

La formule utilisée pour les paquets TCP et UDP non fragmentés est la suivante :

```
((source_port XOR dest_port) XOR
  ((source_IP XOR dest_IP) AND 0xffff)
    MODULO slave_count
```

Pour des paquets frragmentés TCP ou UDP, et pour tout autre protocole de trafic **IP**, les informations de port source ou de destination sont omises. Pour le trafic non-**IP**, la formule est la même que pour la politique de hachage de transfert **layer2**.

Cette politique à pour but d'imiter le comportement de certains commutateurs ; les commutateurs Cisco avec les PFC2, et certains produits Foundry et IBM.

L'algorithme utilisé pour cette politique n'est pas comforme à 802.3ad.

- **2** ou **layer2+3** — utilise une combinaison d'informations layer2 et layer3 pour générer le hachage.

Utilise l'OUX (XOR) des adresses MAC de matériel et les adresses **IP** pour générer le hachage. La formule est la suivante :

```
(((source_IP XOR dest_IP) AND 0xffff) XOR
  ( source_MAC XOR destination_MAC ))
    MODULO slave_count
```

Cet algorithme mettra les trafics dans un réseau homologue particulier sur le même esclave. Pour le trafic **IP**, la formule est la même que pour la politique de hachage de transmission de layer2.

Cette politique a pour but de fournir une distribution de trafic plus équilibrée qu'avec le layer 2 uniquement, surtout dans les environnements où le périphérique de passerelle du layer3 est requis pour atteindre la plupart des destinations.

L'algorithme de hachage est déterminé.

4.6. CRÉER UNE CONNEXION DE LIAISON PAR L'INTERFACE GRAPHIQUE (GUI).

Vous pouvez utiliser l'utilitaire **control-centrer** de GNOME pour demander au **NetworkManager** de créer une liaison à partir de deux ou plusieurs connexions filaires ou InfiniBand. Il n'est pas nécessaire de créer les connexions à relier entre elles pour commencer. Elles peuvent être configurées au moment de la configuration de la liaison. Vous devez avoir les adresses MAC des interfaces disponibles pour compléter le processus de configuration.

4.6.1. Établir une connexion de liaison

Procédure 4.1. Ajouter une nouvelle connexion de liaison

Suivre les étapes suivantes pour créer une nouvelle connexion de liaison.

1. Appuyer sur la touche **Super** pour accéder au menu Activités, saisir **control network**, et appuyez sur la touche **Enter**. L'outil de configuration du **Réseau** apparaîtra. Cette étape est totalement expliquée dans Section 2.5, « Utiliser le NetworkManager avec l'interface graphique GNOME ».

2. Cliquer sur le signe plus pour ouvrir la liste de sélection. Sélectionner **Bond**. La fenêtre **Modifier connexion de liaison*1*** apparaîtra.

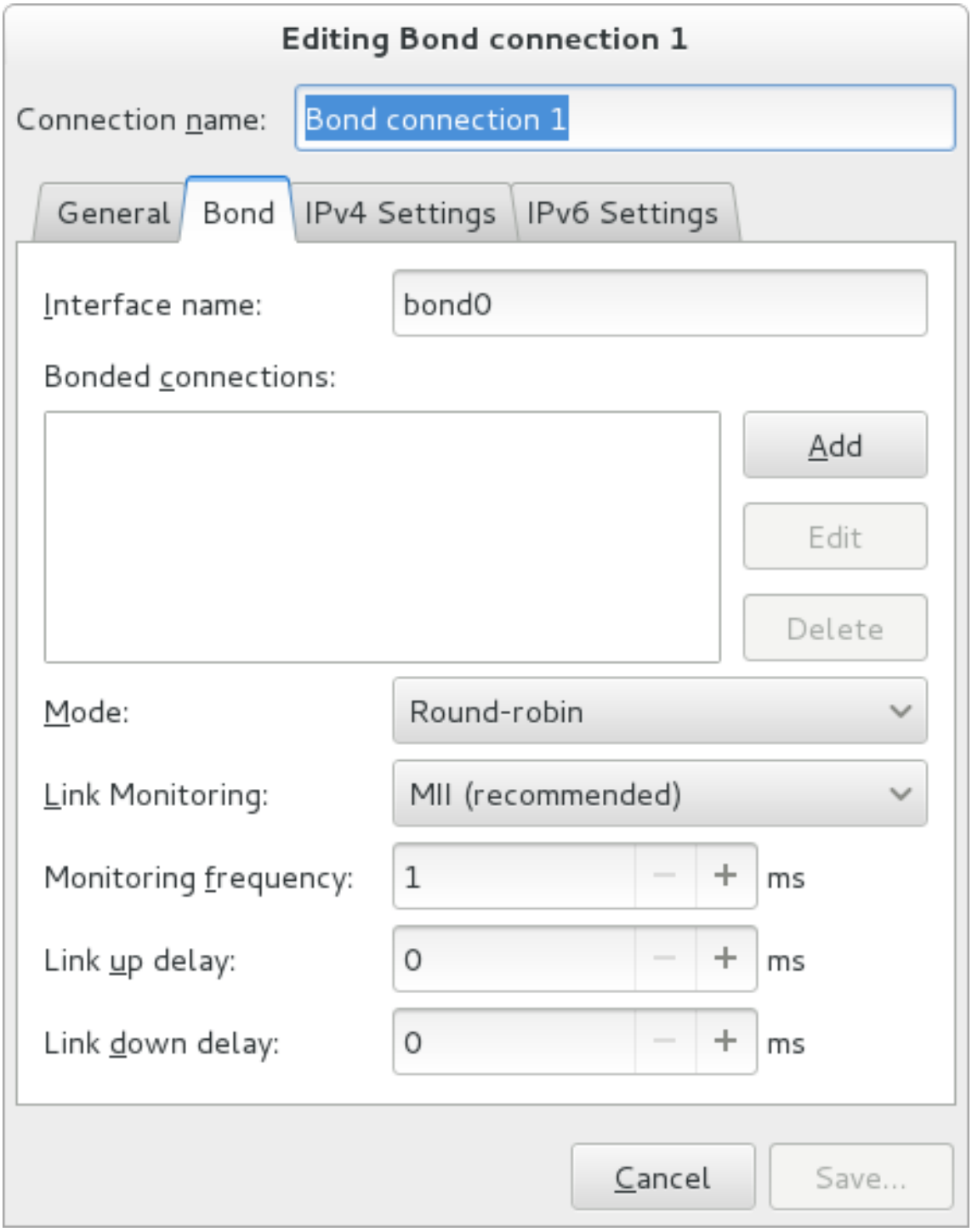


Figure 4.6. Pour que l'interface utilisateur graphique du NetworkManager ajoute un menu de liaisons (Bond).

3. Dans l'onglet **Bond** (ou Liaisons), cliquez sur **Ajouter** et sélectionnez le type d'interface que vous souhaitez utiliser avec la connexion de liaison. Cliquez sur le bouton **Créer**. Notez que la boîte de dialogue pour sélectionner le type d'esclave n'apparaît uniquement que lorsque vous créez le premier esclave. Après cela, ce même type sera utilisé automatiquement pour toutes les autres esclaves.

4. La fenêtre de **Editing bond0 esclave 1** s'affiche. Utilisez la liste déroulante **adresses MAC de périphériques** pour sélectionner l'adresse MAC de l'interface à relier. L'adresse MAC du premier esclave servira comme adresse MAC de l'interface de liaison. Si nécessaire, entrez une adresse MAC de clone à utiliser comme adresse MAC de liaison. Cliquez sur le bouton **Enregistrer**.

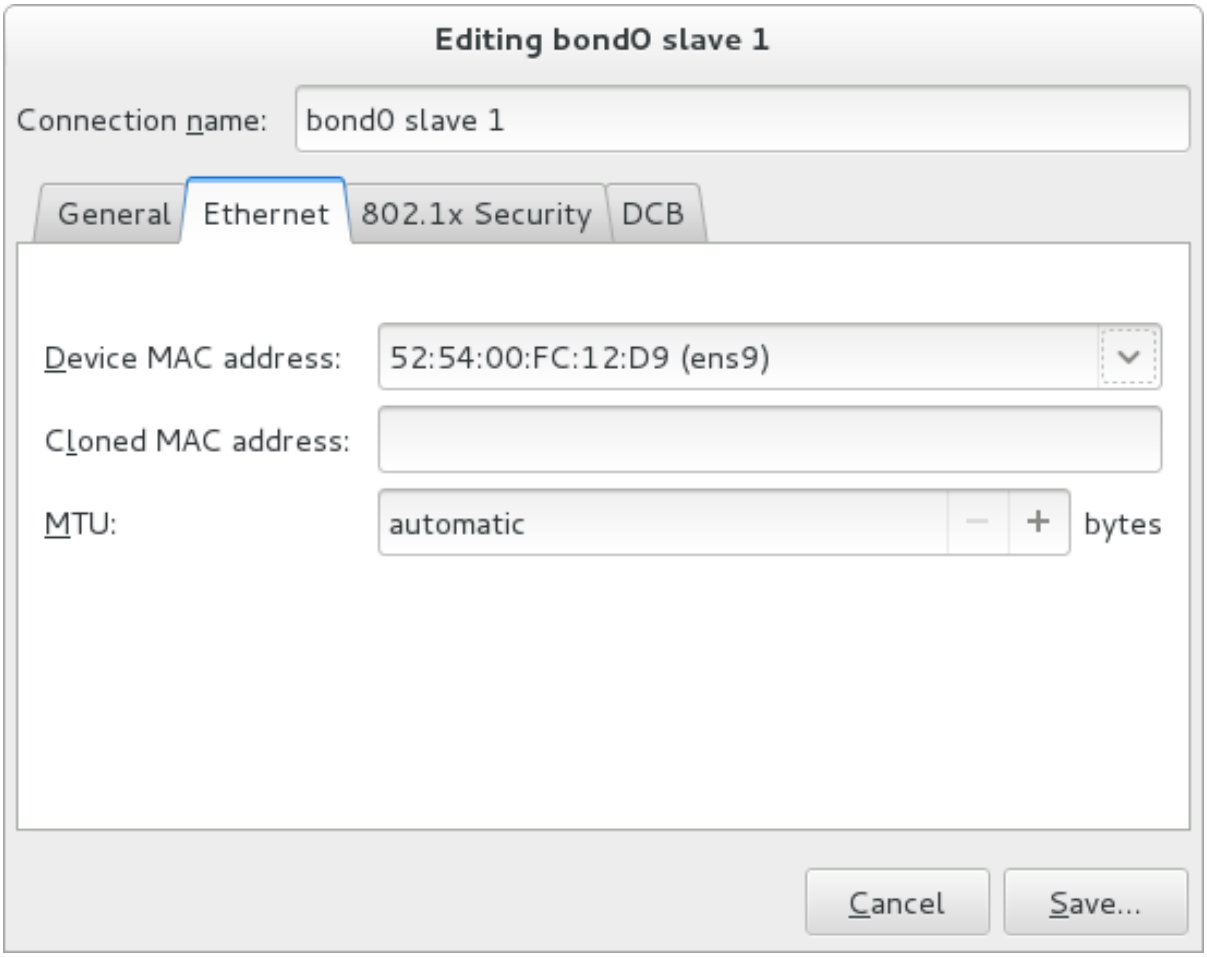


Figure 4.7. Pour que l'interface utilisateur graphique du NetworkManager ajoute un menu de connexion de liaison

5. Le nom de l'esclave lié apparaît dans la fenêtre **Connexions liées**. Cliquez sur le bouton **Ajouter** pour ajouter des connexions esclaves supplémentaires.

6. Vérifier et confirmer les paramètres de configuration, puis cliquer sur le bouton **Sauvegarder**.

7. Modifier la configuration spéciale liaisons en consultant Section 4.6.1.1, « Configurer l'onglet Liaisons ».

Procédure 4.2. Modifier une connexion de liaison existante

Suivre ces étapes pour modifier une connexion de liaison existante.

1. Appuyer sur la touche **Super** pour accéder au menu Activités, saisir **control network**, et appuyez sur la touche **Entrée**. L'outil de configuration du **Réseau** apparaîtra.

2. Sélectionner la connexion à modifier et cliquer sur le bouton **Options**.

3. Sélectionner l'onglet **Général**.

4. Configurer le nom de la connexion, le comportement auto-connect, et les paramètres disponibles.

   Il existe cinq configurations de la boîte de dialogue **Modifier** qui sont communes à tous les types de connexion. Voir l'onglet **Général** :

- **Nom de connexion** — saisir un nom descriptif pour votre connexion de réseau. Ce nom sera utilisé pour lister cette connexion dans le menu de la fenêtre **Réseau**.

- **Se connecter automatiquement à ce réseau quand il est disponible** — sélectionnez cette case si vous souhaitez que le **NetworkManager** s'auto-connecte à cette connexion quand elle sera disponible. Voir Section 2.5.3, « Se connecter à un réseau automatiquement » pour plus d'informations.

- **Rendre le réseau disponible à tous les utilisateurs** — sélectionnez cette case pour créer une connexion disponible à tous les utilisateurs sur le système. Changer ce paramètre peut nécessiter des privilèges d'utilisateur root. Consulter Section 2.5.4, « Profils de connexions privées ou sur tout le système » pour obtenir plus d'informations.

- **Se connecter automatiquement au VPN quand on utilise cette connexion** — sélectionnez cette case si vous souhaitez que le **NetworkManager** s'auto-connecte à une connexion de VPN quand il est disponible. Sélectionner le VPN à partir du menu déroulant.

- **Zone de parefeu** — sélectionnez une zone de parefeu dans le menu déroulant. Voir le guide *Red Hat Enterprise Linux 7 Security Guide* pour obtenir plus d'informations sur les Zones de parefeux.

5. Modifier la configuration spéciale liaisons en consultant Section 4.6.1.1, « Configurer l'onglet Liaisons ».

Sauvegarder votre nouvelle connexion (ou votre connexion modifiée) et faire des configurations supplémentaires

Une fois vous aurez terminé de modifier votre Bond Connection au réseau local virtuel, cliquez sur le bouton **Enregistrer** pour enregistrer votre configuration personnalisée. Si le profil est en cours d'utilisation alors qu'il est modifé, alimentez le cycle de connexion pour que le **NetworkManager** applique les modifications. Si le profil est désactivé (OFF), réglez-le sur ON ou sélectionnez-le dans le menu de l'icône de connexion réseau. Voir Section 2.5.1, « Se connecter à réseau par un GUI » pour plus d'informations sur l'utilisation de votre connexion nouvelle ou modifiée.

Vous pouvez configurer davantage une connexion existante en la sélectionnant dans la fenêtre **Réseau** et en cliquant sur **Options** pour revenir à la boîte de dialogue **Modifier**.

Puis, pour configurer :

- Paramètres de configuration **IPv4** pour la connection, cliquer sur l'onglet **IPv4 Settings** et continuer avec Section 2.5.10.4, « Configuration des paramètres IPv4 »; ou,

- Paramètres de configuration **IPv6** pour la connexion, cliquer sur l'onglet **IPv6 Settings**et continuez avec Section 2.5.10.5, « Configurer les paramètres IPv6 ».

Après la sauvegarde, la liaison apparaîtra dans l'outil de configuration du réseau, avec chaque esclave affiché.

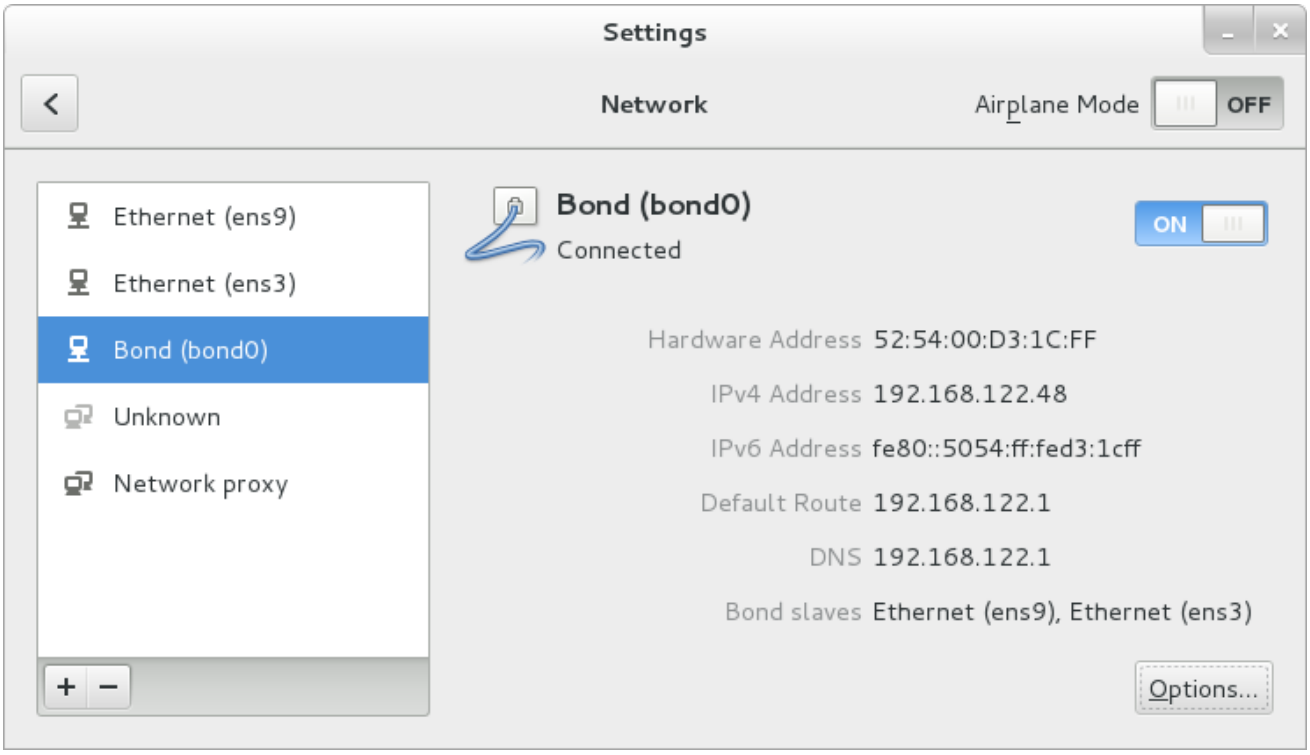


Figure 4.8. Interface utilisateur graphique du NetworkManager avec Liaison

4.6.1.1. Configurer l'onglet Liaisons

Si vous avez déjà ajouté une nouvelle connexion de liaison (voir Procédure 4.1, « Ajouter une nouvelle connexion de liaison » pour obtenir des instructions), vous pouvez modifier l'onglet **Bond** afin de définir le mode de partage de la charge et le type de contrôle de lien à utiliser pour détecter les échecs d'une connexion d'esclave.

`Mode`

Le mode utilisé pour partager le trafic sur les connexions esclaves qui constituent la liaison. La valeur par défaut correspond à **`Round-robin`**. D'autres modes de partage de la charge, comme **`802.3ad`**, peuvent être sélectionnés par l'intermédiaire de la liste du menu déroulant.

`Link Monitoring`

La méthode de surveillance de la capacité des esclaves à supporter le trafic réseau.

Les modes de partage de la charge suivants peuvent être sélectionnés à partir du menu déroulant **`Mode`** :

`Round-robin`

Définit une politique round-robin de répartition des charges et de diffusion de tolérance d'erreurs. Toutes les transmissions sont reçues et diffusées sur chaque interface esclave liée, en commençant par la première disponible. Ce mode risque de ne pas fonctionner pour un pontage de machines virtuelles sans configuration de commutateur supplémentaire.

`Active backup`

Définit une politique active-backup de tolérance d'erreurs. Toutes les transmissions sont reçues et diffusées en commençant par la première interface disponible. Une autre interface esclave liée n'est utilisée que si l'interface esclave liée échoue. Notez qu'il s'agit du seul mode disponible pour les liaisons de périphériques InfiniBand.

XOR

Les transmissions sont basées sur la politique de hachage sélectionnée. La valeur par défaut consiste à tirer un hachage par l'OUX (XOR) des adresses MAC de source et de destination, multiplié par le modulo du nombre d'interfaces d'esclave. Dans ce mode, le trafic destiné à des homologues spécifiques sera toujours envoyé sur la même interface. Comme la destination est déterminée par l'adresse MAC, cette méthode fonctionne mieux pour le trafic en direction d'homologues sur le même lien ou réseau local. Si le trafic doit passer par un routeur unique, alors ce mode d'équilibrage de trafic sera sous-optimal.

Broadcast

Définit une politique de diffusion de tolérance d'erreurs. Toutes les transmissions sont reçues et diffusées sur toutes les interfaces esclaves. Ce mode risque de ne pas fonctionner pour un pontage de machines virtuelles sans configuration de commutateur supplémentaire.

802.3ad

Définit une politique d'aggrégation de lien dynamique **802.3ad** IEEE. Crée des groupes d'agrégation qui partagent les mêmes configurations en matière de vitesse et de duplex. Transmet ou reçoit sur tous les esclaves dans l'agrégateru actif. Requiert un commutateur conforme à **802.3ad**.

Adaptive transmit load balancing (TLB)

Définit une politique Transmit-Load-Balancing (TLB) de tolérance aux pannes et d'équilibrage des charges. Le trafic sortant est distribué selon la charge en cours sur chaque interface esclave. Le trafic entrant est reçu par l'esclave actuel. Si l'esclave récepteur échoue, un autre esclave reprend l'adresse MAC de l'esclave qui a échoué. Ce mode convient uniquement aux adresses locales connues du module de liaison du noyau et ne peut donc être utilisé pour un pontage avec des machines virtuelles.

Adaptive load balancing (ALB)

Définit une politique Adaptive-Load-Balancing (ALB) de tolérance aux pannes et d'équilibrage des charges. Inclut la transmission et la réception de l'équilibrage des charges pour le trafic **IPv4**. La réception de l'équilibrage des charges est fait par négociation **ARP** Ce mode convient uniquement aux adresses locales connues du module de liaison du noyau et ne peut donc être utilisé pour un pontage avec des machines virtuelles.

Les types suivants de contrôle de liens peuvent être sélectionnés à partir du menu **Link Monitoring**. Il est judicieux de tester quels paramètres de module de liaisons de canaux fonctionne le mieux pour les interfaces liées.

MII (Media Independent Interface)

L'état de l'onde porteuse de l'interface est surveillé. Ceci est possible en interrogeant le pilote, en interrogeant les registres MII directement, ou en utilisant **ethtool** pour interroger le périphérique. Trois options sont possibles :

Monitoring Frequency

Durée, en millisecondes, entre les interrogations du piltote et les enregistrements MII.

Link up delay

La durée, en millisecondes, à attendre avant d'utiliser un lien qui a été rapporté comme étant actif. Ce délai peut être utilisé si certaines requêtes **ARP** spontanées sont perdues dans la période qui suit immédiatement le lien rapporté comme étant « up » (actif). Cela peut se produire pendant

l'initialisation du commutateur, par exemple.

`Link down delay`

La durée, en millisecondes, à attendre avant de changer à un autre lien quand un lien qui était actif est maintenant « down » (inactif). Ce délai peut être utilisé si un commutateur attaché prend un bon moment à changer en mode de sauvegarde.

ARP

Le protocole de résolution d'adresse (**ARP**) est utilisé pour sonder un ou plusieurs homologues, afin de déterminer quelles sont les connexions de couche de liaison qui fonctionnent. Cela dépend du pilote de périphérique fournissant l'heure de début d'émission et de dernier renvoi.

Il y a deux options disponibles :

`Monitoring Frequency`

Durée, en millisecondes, entre les envois de requêtes **ARP**.

`cibles ARP`

Liste d'adresses **`IP`** séparées par des virgules pour envoyer des requêtes **ARP**.

4.7. RESSOURCES SUPPLÉMENTAIRES

Les sources d'informations suivantes fournissent des ressources supplémentaires à propos des liaisons de réseaux.

4.7.1. Documentation installée

- Page man **`nmcli(1)`** — décrit l'outil de ligne de commande du **NetworkManager**.
- Page man **`nmcli-examples(5)`** — donne des exemples des commandes **nmcli**.
- Page man **`nm-settings(5)`** — décrit les configurations et les paramètres des connexions du **NetworkManager**.

4.7.2. Documentation en ligne

Red Hat Enterprise Linux 7 System Administrator's Guide

Explique comment utiliser les capacités du module du noyau.

https://access.redhat.com/site/node/28421/Configuring_VLAN_devices_over_a_bonded_interface

Un article de la base de connaissances de Red Hat sur la configuration de périphériques VLAN via interface liée.

CHAPITRE 5. CONFIGURATION DE NETWORK TEAMING

5.1. QU'EST-CE QUE NETWORK TEAMNING ?

La combinaison ou le regroupement de liens de réseau dans le but de fournir un lien logique avec un débit plus élevé, ou pour fournir une redondance, est connu sous plusieurs noms : « canal de liaison », « liaison Ethernet », « trunking de port », « regroupement de canaux ou canal teaming », « regroupement de cartes réseau », « agrégation de liens » et ainsi de suite. Ce concept qui avaient été initialement mis en place dans le noyau Linux correspond largement au « Collage ». Le terme regroupement de réseaux a été choisi pour désigner cette nouvelle implémentation du concept. Le pilote de liaison existant n'est pas affecté, le regroupement de réseaux est offert comme solution de rechange et ne remplace pas la liaison dans Red Hat Enterprise Linux 7.

Network Teaming, ou Team, est conçu pour mettre en oeuvre le concept d'une manière unique, en fournissant un petit pilote de noyau pour le traitement rapide des flux de paquets et de diverses applications de l'espace utilisateur pour pouvoir faire tout le reste en espace utilisateur. Le pilote a une *Interface de programmation d'application* (API), dénommée « Team Netlink API », qui actionne les communications Netlink. Les applications d'espace utilisateur peuvent utiliser cette API pour communiquer avec le pilote. Une bibliothèque, nommée « lib », est fournie pour regrouper les communications Team Netlink et les messages RT Netlink dans l'espace utilisateur. Un démon d'application, **`teamd`**, qui utilise Libteam lib est également fourni. Une seule instance de **`teamd`** peut contrôler une instance du pilote de Team. Le démon implémente la logique d'active-backup et l'équilibrage de la charge, comme round-robin, à l'aide de code supplémentaire « runners ». En séparant le code de cette manière, la mise en place de Network Teaming présente une solution facilement extensible et évolutive pour l'équilibrage des charge et les besoins de redondance. Par exemple, des runners personnalisés peuvent être écrits assez facilement pour mettre en œuvre la nouvelle logique via **`teamd`**, et même quand **`teamd`** est en option, les utilisateurs peuvent écrire leur propre application pour utiliser **libteam**.

Un outil qui sert à contrôler une instance d'exécution de **`teamd`** en utilisant D-bus est fourni par **teamdctl**. Il fournit un wrapper de D-Bus autour de l'API D-Bus **`teamd`**. Par défaut, **`teamd`** écoute et communique à l'aide de sockets de domaine Unix mais continue de surveiller le D-Bus. Il s'agit de veiller à ce que **`teamd`** puisse être utilisé dans des environnements où le D-Bus n'est pas présent ou n'a pas encore été chargé. Par exemple, lors de l'amorçage sur des liaisons **`teamd`**, D-Bus n'est pas encore téléchargé. L'outil **teamdctl** peut être utilisé pendant le temps d'exécution pour lire la configuration, l'état de link-watchers, pour extraire et modifier l'état des ports, pour ajouter et supprimer des ports, et pour changer des ports d'actifs en état de sauvegarde.

Team API Netlink communique avec les applications d'espace utilisateur à l'aide de messages Netlink. La bibliothèque d'espace-utilisateur **libteam** n'interagit pas directement avec l'API, mais utilise **libnl** ou **teamnl** pour interagir avec l'API du pilote.

Pour résumer, les instances du pilote de Team, en cours d'exécution dans le noyau, ne peuvent pas être configurées ou contrôlées directement. Toute la configuration se fait à l'aide des applications spatiales utilisateur, comme l'application **teamd**. L'application redirige ensuite la partie pilote du noyau en suivant par la suite.

5.2. COMPRENDRE LES COMPORTEMENTS PAR DÉFAUT DES INTERFACES MAÎTRES ET ESCLAVES

À chaque fois que vous contrôlez les interfaces de port regroupées (teamed) par le démon **`NetworkManager`**, et surtout quand vous cherchez des fautes, gardez à l'esprit les faits suivants :

1. Démarrer l'interface maître ne démarre pas les interfaces de port automatiquement.

2. Démarrer une interface de port ne démarre pas toujours l'interface maître.
3. Stopper l'interface maître stoppe également les interfaces de port.
4. Une interface maître sans port peut démarrer les connexions **IP** statiques.
5. Une interface maître sans port attend les ports avant de démarrer les connexions **DHCP**.
6. Une interface maître avec une connexion **DHCP** en attente de ports se termine quand un port accompagné d'un transporteur est ajouté.
7. Une interface maître avec une connexion **DHCP** en attente de ports continue quand un port sans transporteur est ajouté.

AVERTISSEMENT

L'utilisation des connexions directes par câble sans commutateurs de réseau n'est pas pris en charge pour le teaming. Le mécanisme de basculement décrit ici ne fonctionnera pas comme prévu sans la présence de commutateurs de réseaux. Voir l'article de base de connaissance *Why is bonding in not supported with direct connection using crossover cables?* (Pourquoi Network Bonding ne prend il pas en charge les connexions directes avec des câbles croisés ?) pour obtenir plus d'informations.

5.3. COMPARAISON ENTRE LE REGROUPEMENT ET LA LIAISON DE RÉSEAUX (TEAMING VERSUS BONDING)

Tableau 5.1. Comparaison des fonctionnalités de Bonding et Teaming

Fonctionnalité	Bonding	Team
politique de tx de diffusion	Oui	Oui
politique de tx round-robin	Oui	Oui
politique Tx active-backup	Oui	Oui
support LACP (802.3ad)	Oui (passif uniquement)	Oui
politique Tx basée hachage	Oui	Oui
L'utilisateur peut définr une fonction de hachage	Non	Oui
Support Tx Load-Balancing (TLB)	Oui	Oui

Fonctionnalité	Bonding	Team
sélection de port de hachage LACP	Oui	Oui
équilibrage des charges pour support LACP	Non	Oui
monitoring du lien Ethtool	Oui	Oui
monitoring du lien ARP	Oui	Oui
monitoring du lien NS/NA (IPv6)	Non	Oui
délais port actifs/inactifs	Oui	Oui
priorités de ports et stickiness (amélioration d'option « primaire »)	Non	Oui
installation du monitoring de liens séparés par-port	Non	Oui
installation du monitoring de liens multiples	Limitée	Oui
chemin Tx/Rx sans verrou	Non (rwlock)	Oui (RCU)
Support VLAN	Oui	Oui
contrôle de runtime d'espace utilisateur	Limitée	Plein
Logique en espace utilisateur	Non	Oui
Extensibilité	Difficile	Facile
Design modulaire	Non	Oui
Dégradation de la performance	Basse	Très basse
interface D-Bus	Non	Oui
empilage de plusieurs périphériques	Oui	Oui
config zéro avec LLDP	Non	(en cours de planification)

Fonctionnalité	Bonding	Team
Support de NetworkManager	Oui	Oui

5.4. COMPRENDRE LE DÉMON DE NETWORK TEAMING ET LES "RUNNERS"

Le démon de Team, **`teamd`**, utilise **libteam** pour contrôler une instance du pilote de team. Cette instance du pilote team ajoute des instances d'un pilote de périphérique de matériel pour former un « groupement » (team) de liaisons réseau. Le pilote de team présente une interface réseau, team0 par exemple, pour les autres parties du noyau. Les interfaces créées par les instances du pilote de team reçoivent des noms comme team0, team1 et ainsi de suite, dans la documentation. C'est pour la facilité de compréhension et d'autres noms peuvent être utilisés. La logique commune à toutes ces méthodes de regroupement (teaming) est implémentée par **`teamd`** ; ces fonctions qui sont uniques aux différentes méthodes de sauvegarde et de répartition de charges, comme round-robin, sont activées par des entités de code dénommées « runners ». Comme les mots « module » et « mode » ont déjà des significations spécifiques en ce qui concerne le noyau, le mot « runner » a été choisi pour faire référence à ces unités de code. L'utilisateur spécifie le runner dans le fichier de configuration au format JSON, et le code est ensuite compilé en une instance de **`teamd`** lorsque l'instance est créée. Un runner n'est pas un plug-in car le code d'un runner est compilé en une instance de **`teamd`** alors qu'il est créé. Le code pourrait être créé comme un plug-in pour **`teamd`** si le besoin se présentait.

Le runners suivants sont disponibles au moment de la rédaction.

- broadcast (les données sont diffusées sur tous les ports)
- round-robin (les données sont transmises à travers tous les ports chacun son tour)
- active-backup (un port ou un lien est utilisé tandis que les autres sont conservés en sauvegarde)
- loadbalance (avec équilibrage des charges tx et sélecteurs de port Tx basé BPF)
- lacp (implémentre le protocole 802.3ad Link Aggregation Control Protocol)

De plus, les link-watchers suivants sont disponibles :

- **ethtool** (lib de Libteam utilise **ethtool** pour surveiller les changements d'état de liens). C'est la valeur par défaut si aucun autre link-watcher n'est spécifié dans le fichier de configuration
- **arp_ping** (L'utilitaire **arp_ping** est utilisé pour surveiller la présence de l'adresse de matériel distant par les paquets ARP)
- **nsna_ping** (de l'anglais Neighbor Advertisements and Neighbor Solicitation d'**`IPv6`** Les protocoles Neighbor Discovery sont utilisés pour surveiller la présence d'une interface voisine)

Il n'y a aucune restriction niveau code pour empêcher un link-watcher particulier d'être utilisé avec un runner particulier, mais quand on utilise le runner **lacp**, **ethtool** est le seul link-watcher recommandé.

5.5. INSTALLATION DU DÉMON DE NETWORK TEAMING

Le démon de regroupement (team) de réseautage , **`teamd`**, n'est pas intallé par défaut. Pour installer le démon **`teamd`**, exécutez la commande suivante en tant qu'utilisateur **`root`** :

```
~]# yum install teamd
```

5.6. CONVERTIR UNE LIAISON (BOND) EN GROUPEMENT (TEAM)

Il est possible de convertir les fichiers de configuration de liaisons existants en fichiers de configuration de groupement par l'outil **bond2team**. Cet outil peut convertir les fichiers de configuration de liaison au format **ifcfg** en fichiers de configuration au format **ifcfg** ou JSON. Notez que les règles de pare-feu, alias interfaces, ou tout ce qui peut être lié au nom de l'interface d'origine peut être rompu après le renommage car l'outil ne changera que le fichier **ifcfg**, et rien d'autre.

Pour voir quelques exemples de format de commande, exécutez la commande suivante :

```
~]$ bond2team --examples
```

De nouveaux fichiers seront créés dans un répertoire dont le nom commence par **/tmp/bond2team.XXXXXX /**, où XXXXXX correspond à une chaîne aléatoire. Après avoir créé les nouveaux fichiers de configuration, déplacer les anciens fichiers de liaison dans un dossier de sauvegarde, puis déplacez les nouveaux fichiers dans le répertoire **/etc/sysconfig/network-scripts /**.

Exemple 5.1. Convertir une Liaison (Bond) en Groupement (Team)

Pour convertir une configuration **bond0** en **ifcfg** (team), exécutez la commande suivante, en tant qu'utilisateur **root** :

```
~]# /usr/bin/bond2team --master bond0
```

Notez qu'on retient le nom **bond0**. Pour utiliser un nouveau nom pour sauvegarder la configuration, utiliser l'option **--rename** comme suit :

```
~]# /usr/bin/bond2team --master bond0 --rename team0
```

ajouter l'option **--json** pour produire des fichiers au format JSON à la place de fichiers **ifcfg**. Voir la page man **teamd.conf(5)** pour obtenir des exemples de format JSON.

Exemple 5.2. Convertir une liaison (Bond) en groupement (Team) et Spécifier le nom de chemin d'accès

Pour convertir une configuration **bond0** en **ifcfg** (team), et pour spécifier manuellement le chemin d'accès au fichier **ifcfg**, exécutez la commande suivante, en tant qu'utilisateur **root** :

```
~]# /usr/bin/bond2team --master bond0 --configdir /path/to/ifcfg-file
```

ajouter l'option **--json** pour produire des fichiers au format JSON à la place de fichiers **ifcfg**.

Exemple 5.3. Créer une configuration Team en utilisant Bond2team

Il est également possible de créer une configuration team en fournissant à l'outil **bond2team** une liste de paramètres de liaison. Exemple :

```
~]# /usr/bin/bond2team --bonding_opts "mode=1 miimon=500"
```

On peut également fournir des ports en ligne de commandes, comme suite :

```
~]# /usr/bin/bond2team --bonding_opts "mode=1 miimon=500 primary=eth1 \
  primary_reselect-0" --port eth1 --port eth2 --port eth3 --port eth4
```

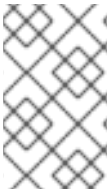
Voir la page man **bond2team(1)** pour plus de détails. Pour obtenir une explication sur les paramètres de liaison, voir Section 4.5, « Utiliser une liaison de canal »

5.7. SÉLECTIONNER LES INTERFACES À UTILISER COMME PORTS POUR UN NETWORK TEAM

Pour voir les connexions disponibles, exécutez la commande suivante :

```
~]$ ip link show
1: lo:  <LOOPBACK,UP,LOWER_UP > mtu 65536 qdisc noqueue state UNKNOWN mode
DEFAULT
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: em1:  <BROADCAST,MULTICAST,UP,LOWER_UP > mtu 1500 qdisc pfifo_fast
state UP mode DEFAULT qlen 1000
    link/ether 52:54:00:6a:02:8a brd ff:ff:ff:ff:ff:ff
3: em2:  <BROADCAST,MULTICAST,UP,LOWER_UP > mtu 1500 qdisc pfifo_fast
state UP mode DEFAULT qlen 1000
link/ether 52:54:00:9b:6d:2a brd ff:ff:ff:ff:ff:ff
```

En partant des interfaces disponibles, déterminez celles qui peuvent être ajoutées à votre groupement de réseaux, puis continuez avec Section 5.8, « Sélection des méthodes de configuration de Network Team »



NOTE

Les développeurs de Team préfèrent le terme « port » plutôt qu'« esclave », mais le **NetworkManager** utilise le terme « team-slave » pour faire référence aux interfaces qui forment un groupement.

5.8. SÉLECTION DES MÉTHODES DE CONFIGURATION DE NETWORK TEAM

Pour configurer un groupement de réseaux avec l'outil d'interface d'utilisateur texte du NetworkManager, **nmtui**, consultez Section 5.9, « Configurer Network Team par l'interface texte utilisateur, nmtui »

Pour créer un groupement de réseaux avec l'interface en ligne de commandes , **nmcli**, consultez Section 5.10.1, « Configurez un Network Team par le nmcli ».

Pour créer un groupement de réseaux par le démon de Team , **teamd**, consultez Section 5.10.2, « Créer un Network Team avec teamd ».

Pour créer un groupement de réseaux par les fichiers de configuration, consultez Section 5.10.3, « Créer un Network Team à l'aide des fichiers ifcfg ».

Pour configurer un groupement de réseaux par l'interface utilisateur graphique, consultez Section 5.13, « Créer un Network Team par l'interface graphique (GUI) »

5.9. CONFIGURER NETWORK TEAM PAR L'INTERFACE TEXTE UTILISATEUR, NMTUI

L'outil d'interface utilisateur de texte **nmtui** peut être utilisé pour configurer le groupement dans une fenêtre de terminal. Exécutez la commande suivante pour démarrer l'outil :

```
~]$ nmtui
```

L'interface utilisateur texte apparaîtra. Tout commande non valide affichera un message d'utilisation.

Pour naviguer, utiliser les flèches ou appuyer sur **Tab** pour continuer et appuyer sur la combinaision de touches **Maj**+**Tab** pour revenir aux options. Appuyer sur la touche **Entrée** pour sélectionner une option. La barre **Espace** active/désactive le statut d'une case à cocher.

1. À partir du menu de démarrage, sélectionner **Modifier une connexion**. Sélectionner **Ajouter**, l'écran **Nouvelle connexion** apparaîtra.

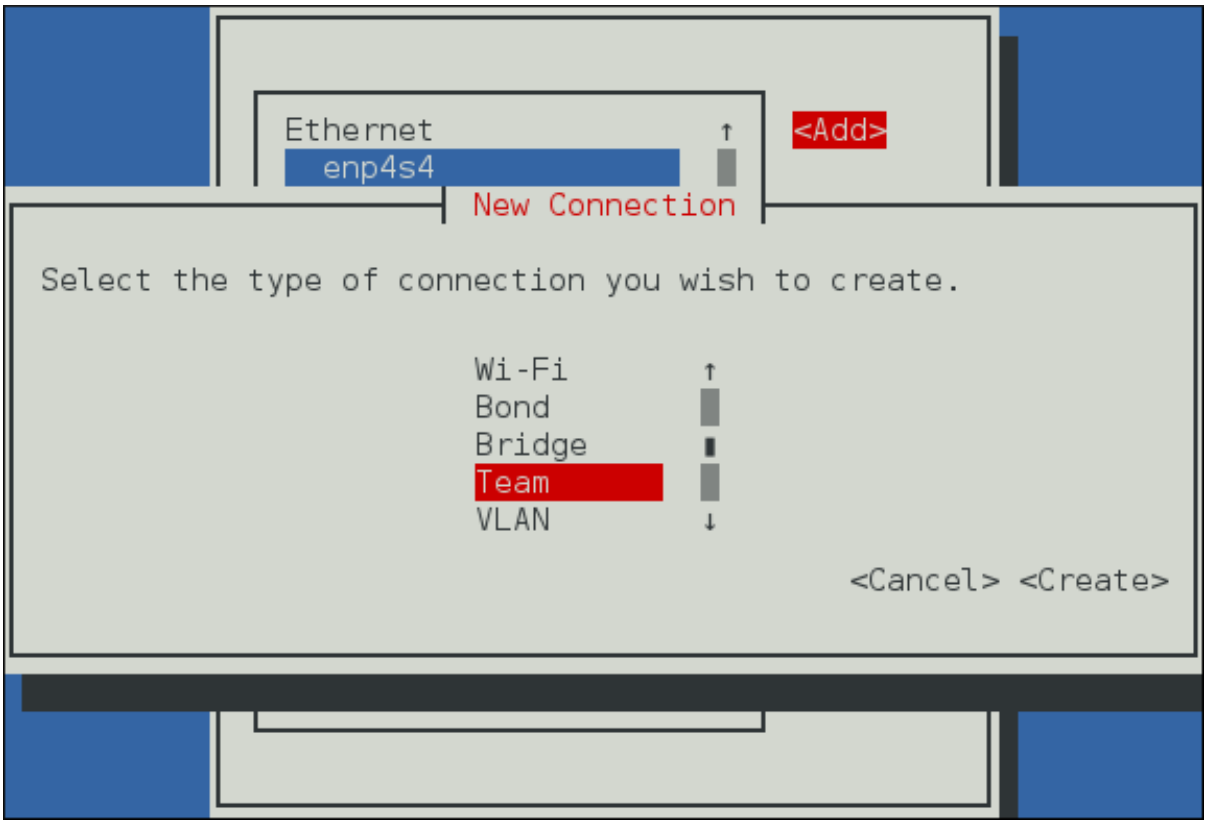


Figure 5.1. Pour que l'interface utilisateur texte du NetworkManager ajoute un menu de connexion de Team

2. Sélectionner **Team**, l'écran **Modifier Connexion** apparaîtra.

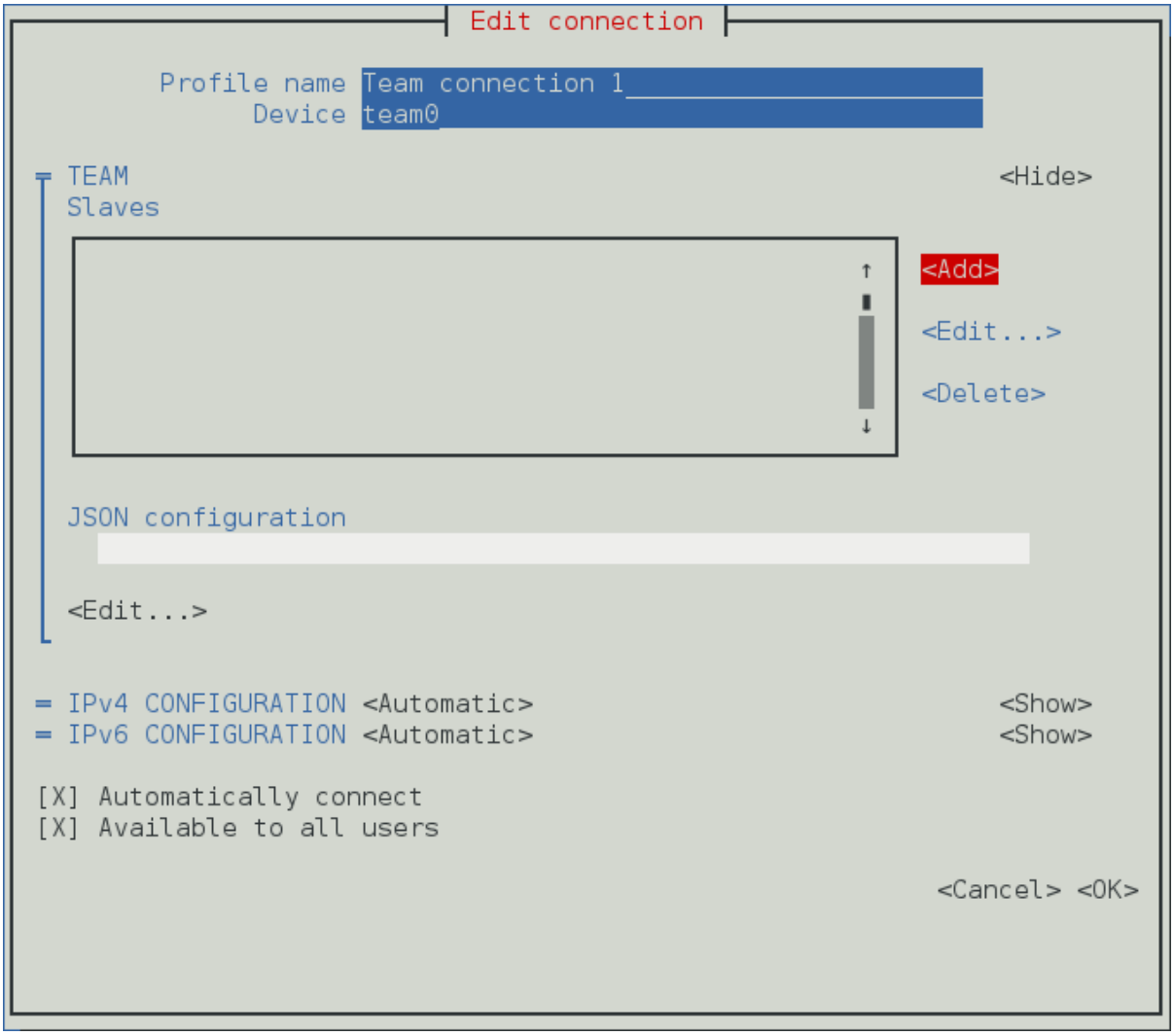


Figure 5.2. Pour que l'interface utilisateur texte du NetworkManager configure un menu de connexion Team

3. Pour ajouter des interfaces de ports au groupement, sélectionner **`Ajouter`**, et l'écran **`Nouvelle connexion`** apparaîtra. Une fois que le type de connexion aura été sélectionné, appuyer sur le bouton **`Créer`** pour que l'écran **`Modifier Connexion`** de Team puisse apparaître.

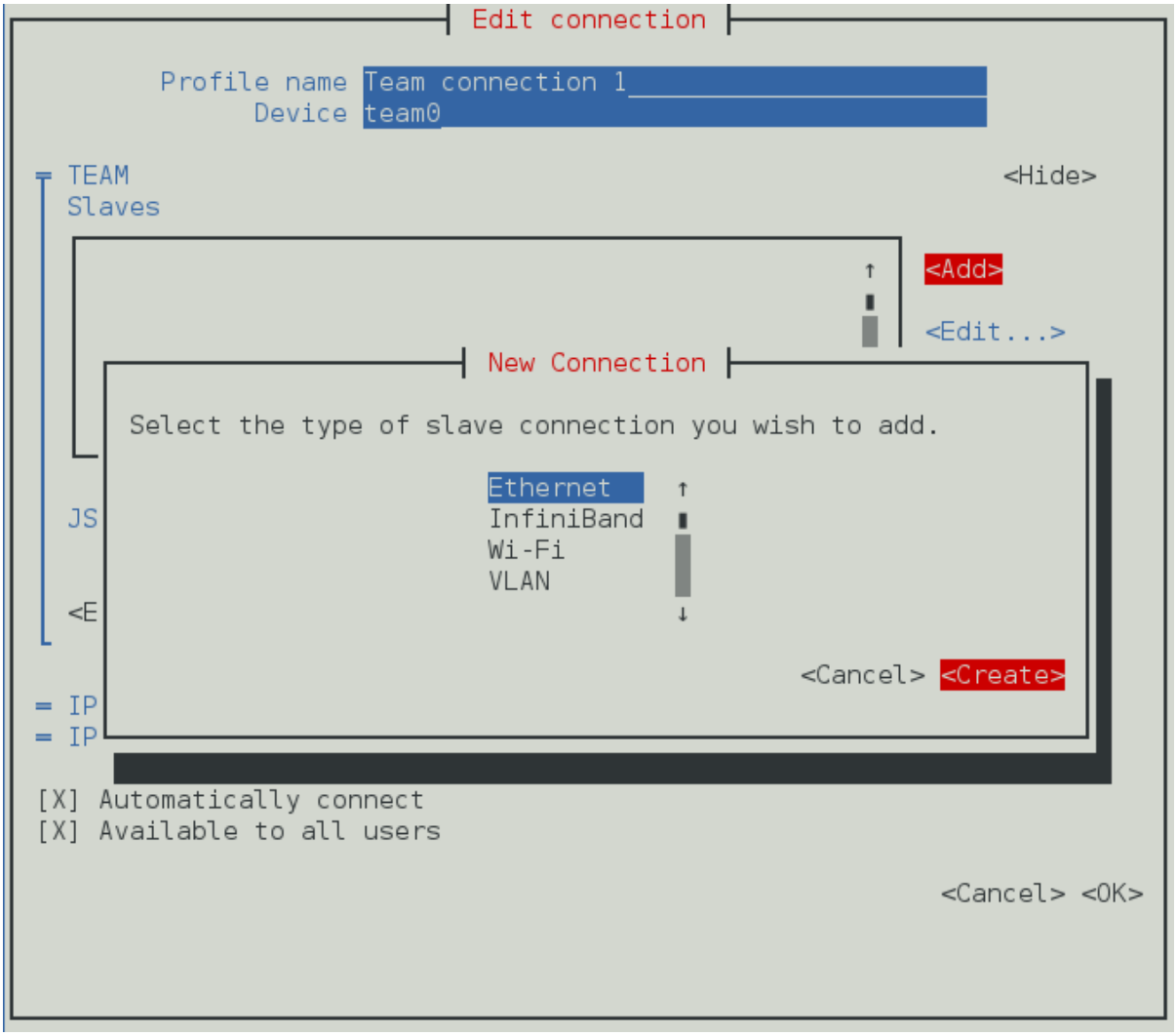


Figure 5.3. Pour que l'interface utilisateur texte du NetworkManager configure un nouveau menu de connexion d'interface de ports Team

4. Saisir l'adresse MAC ou le nom de périphérique de l'esclave que vous aurez choisi dans la section **Périphériques**. Si besoin est, saisir une adresse MAC clonée à utiliser comme adresse MAC de groupement, en sélectionnant **Afficher** à droite de l'étiquette **Ethernet**. Sélectionnez le bouton **OK**.

NOTE

Si le périphérique est spécifié sans adresse MAC, la section **Périphérique** sera remplie automatiquement une fois que la fenêtre **Modifier Connexion** est chargée, mais uniquement s'il trouve le périphérique.

Figure 5.4. Pour que l'interface utilisateur texte du NetworkManager configure un menu de connexion d'interface de ports Team

5. Le nom de l'esclave groupé (teamed) apparaît dans la section **`Esclaves`**. Répétez les étapes ci-dessus pour ajouter des connexions esclaves.

6. Si la configuration de ports doit être appliquée, sélectionner le bouton **`Edit`** qui se situe sous la section **`configuration JSON`**. Cela lancera une console **vim** où des changements risquent d'être appliqués. Une fois terminé, écrire les changements de **vim** et confirmez que la chaîne JSON qui s'affiche dans la **`configuration JSON`** correspond bien à ce qui avait été envisagé.

7. Vérifier et confirmer les paramètres de configuration, avant de cliquer sur le bouton **`OK`**.

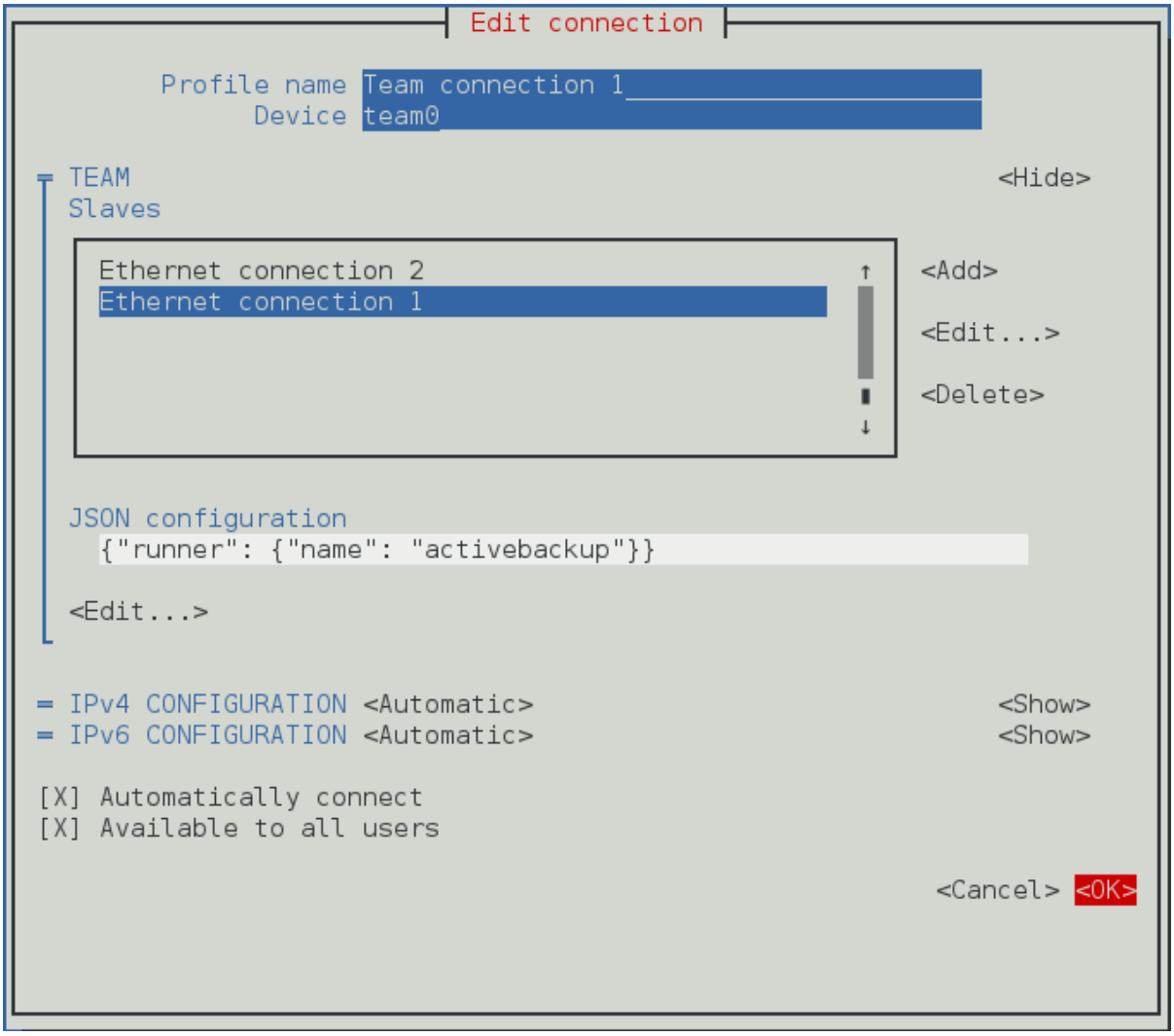


Figure 5.5. Pour que l'interface utilisateur texte du NetworkManager configure un menu de connexion Team

Voir Section 5.12, « Configurez les runners de teamd » pour obtenir des exemples de chaînes JSON. Noter que seules les sections pertinentes d'exemples de chaînes doivent être utilisées pour une configuration de ports ou de groupement (team) par **nmtui**. Ne spécifiez pas le « Périphérique » dans la chaîne JSON. Ainsi, seulement la chaîne JSON doit être utilisée après « périphérique » mais avant « port » dans le champ de configuration de JSON Team. Toutes les chaînes JSON allant vers un port s'ajoutent seulement dans le champ configuration de port.

Voir Section 1.5, « Configuration réseau utilisant une interface utilisateur texte (nmtui) » pour obtenir des informations sur la façon d'installer **nmtui**.

5.10. CONFIGUREZ UN NETWORK TEAM EN LIGNE DE COMMANDES

5.10.1. Configurez un Network Team par le nmcli

Pour voir les périphériques disponibles sur le système, exécutez la commande suivante :

```
~]$ nmcli connection show
NAME  UUID                                  TYPE            DEVICE
eth1  0e8185a1-f0fd-4802-99fb-bedbb31c689b  802-3-ethernet  --
eth0  dfe1f57b-419d-4d1c-aaf5-245deab82487  802-3-ethernet  --
```

Pour créer une nouvelle interface de regroupement, du nom *team-ServerA*, exécutez la commande comme suit :

```
~]$ nmcli connection add type team ifname team-ServerA
Connection 'team-ServerA' (b954c62f-5fdd-4339-97b0-40efac734c50)
successfully added.
```

Le **NetworkManager** fixera ses paramètres internes **connection.autoconnect** à **oui** et comme aucune adresse **IP** n'a été donnée, **ipv4.method** sera défini sur **auto**. Le **NetworkManager** écrira aussi un fichier de configuration **/etc/sysconfig/network-scripts/ifcfg-team-ServerA** où le ONBOOT correspondant sera définie sur **oui** et BOOTPROTO sera défini sur **dhcp**.

Notez que les changements manuels apportés au fichier ifcfg ne seront pas remarqués par le **NetworkManager** tant que l'interface n'est pas appelée à nouveau. Voir Section 1.9, « Configuration de réseau par les fichiers sysconfig » pour obtenir plus d'informations sur la façon d'utiliser les fichiers de configuration.

Pour afficher les autres valeurs assignées, veuillez exécuter une commande comme suit :

```
~]$ nmcli con show team-ServerA
connection.id:                          team-ServerA
connection.uuid:                        b954c62f-5fdd-4339-97b0-
40efac734c50
connection.interface-name:              ServerA
connection.type:                        team
connection.autoconnect:                 yes…
ipv4.method:                            auto[output truncated]
```

Comme aucune configuration JSO n'a a été spécifiée, les valeurs par défaut s'appliquent. Voir la page man **teamd.conf(5)** pour plus d'informations sur les paramètres JSON Team et leurs valeurs par défaut. Notez que le nom est dérivé du nom de l'interface en ajoutant le type. Vous pouvez également spécifier un nom avec l'option de **con-name**, comme suit :

```
~]$ nmcli connection add type team con-name Team0 ifname ServerB
Connection 'Team0' (5f7160a1-09f6-4204-8ff0-6d96a91218a7) successfully
added.
```

Pour afficher les interfaces qui viennent d'être configurées, veuillez exécuter une commande comme suit :

```
~]$ nmcli con show
NAME                UUID                                  TYPE
DEVICE
team-ServerA        b954c62f-5fdd-4339-97b0-40efac734c50  team
ServerA
eth1                0e8185a1-f0fd-4802-99fb-bedbb31c689b  802-3-ethernet
--
eth0                dfe1f57b-419d-4d1c-aaf5-245deab82487  802-3-ethernet
--
Team0               5f7160a1-09f6-4204-8ff0-6d96a91218a7  team
ServerB
```

Pour modifier le nom assigné à un groupement (team), exécutez une commande sur le format suivant :

```
nmcli con mod old-team-name connection.id new-team-name
```

Pour charger un fichier de configuration d'un groupement déjà existant, exécutez une commande du format suivant :

```
nmcli connexion modifyteam-nameteam.configJSON-config
```

. Vous pouvez spécifier la configuration team comme une chaîne JSON ou fournir un nom de fichier contenant la configuration. Le nom de fichier peut inclure le chemin d'accès. Dans les deux cas, ce qui est stocké dans la propriété ***team.config*** est la chaîne JSON. Dans le cas d'une chaîne JSON, entourez la chaîne de guillemets et collez la chaîne entière à la ligne de commande.

Pour modifier la propriété **team.config**, saisir une commande du format suivant :

```
nmcli con show team-name | grep team.config
```

Pour ajouter une interface *eth0* à **Team0**, du nom *Team0-port1*, exécutez la commande suivante :

```
~]$ nmcli con add type team-slave con-name Team0-port1 ifname eth0 master
Team0
Connection 'Team0-port1' (ccd87704-c866-459e-8fe7-01b06cf1cffc)
successfully added.
```

De même, pour ajouter une autre interface, *eth1*, avec le nom *Team0-port2*, exécutez la commande suivante :

```
~]$ nmcli con add type team-slave con-name Team0-port2 ifname eth1 master
Team0
Connection 'Team0-port2' (a89ccff8-8202-411e-8ca6-2953b7db52dd)
successfully added.
```

Au moment de la rédaction, **nmcli** ne prend en charge que les ports Ethernet.

Pour qu'un regroupement (team) apparaisse, les ports doivent tout d'abord apparaître comme suit :

```
~]$ nmcli connection up Team0-port1
Connection successfully activated (D-Bus active path:
/org/freedesktop/NetworkManager/ActiveConnection/2)
```

```
~]$ nmcli connection up Team0-port2
Connection successfully activated (D-Bus active path:
/org/freedesktop/NetworkManager/ActiveConnection/3)
```

Vous pouvez vérifier que l'interface du regroupement apparaisse suite à l'activation des ports, comme suit :

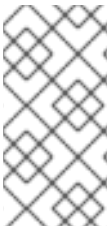
```
~]$ ip link
3:  Team0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state
UP mode DEFAULT
    link/ether 52:54:00:76:6f:f0 brd ff:ff:ff:ff:ff:f
```

Sinon, exécutez une commande pour faire apparaître le regroupement, comme suit :

```
~]$ nmcli connection up Team0
Connection successfully activated (D-Bus active path:
/org/freedesktop/NetworkManager/ActiveConnection/4)
```

Voir Section 2.3, « Utiliser l'outil de ligne de commandes du NetworkManager, nmcli » pour une introduction à **nmcli**

5.10.2. Créer un Network Team avec teamd



NOTE

Les configurations créées avec teamd ne sont pas persistantes, et donc, il faudrait créer un regroupement à partir des étapes décrites dans Section 5.10.1, « Configurez un Network Team par le nmcli » ou Section 5.10.3, « Créer un Network Team à l'aide des fichiers ifcfg ».

Pour créer un Network Team, il faut un fichier de configuration au format JSON pour l'interface virtuelle, qui servira d'interface au regroupement de ports ou de liens. Un moyen rapide consiste à copier les exemple de fichiers de configuration, puis de les modifiez à l'aide d'un éditeur, avec les privilèges d'utilisateur **root**. Pour afficher la liste des exemples de configurations disponibles, saisir la commande suivante :

```
~]$ ls /usr/share/doc/teamd-*/example_configs/
activebackup_arp_ping_1.conf  activebackup_multi_lw_1.conf
loadbalance_2.conf
activebackup_arp_ping_2.conf  activebackup_nsna_ping_1.conf
loadbalance_3.conf
activebackup_ethtool_1.conf   broadcast.conf                random.conf
activebackup_ethtool_2.conf   lacp_1.conf
roundrobin_2.conf
activebackup_ethtool_3.conf   loadbalance_1.conf
roundrobin.conf
```

Pour voir un des fichiers inclus, tel que **activebackup_ethtool_1.conf**, saisir la commande suivante :

```
~]$ cat /usr/share/doc/teamd-*/example_configs/activebackup_ethtool_1.conf
{
 "device": "team0",
 "runner": {"name": "activebackup"},
 "link_watch": {"name": "ethtool"},
 "ports": {
  "eth1": {
   "prio": -10,
   "sticky": true
  },
  "eth2": {
   "prio": 100
  }
 }
}
```

Créer un répertoire de travail de configurations pour stocker les fichiers de configuration de **teamd**. Par exemple, en tant qu'utilisateur ordinaire, saisir une commande comme suit :

```
~]$ mkdir ~/teamd_working_configs
```

Copiez le fichier que vous avez choisi dans votre répertoire de travail, et éditez-le si nécessaire. Par exemple, vous pouvez utiliser une commande du format suivant :

```
~]$ cp /usr/share/doc/teamd-*/example_configs/activebackup_ethtool_1.conf
\ ~/teamd_working_configs/activebackup_ethtool_1.conf
```

Pour modifier le fichier afin qu'il convienne à votre environnement, par exemple, pour changer les interfaces à utiliser en tant que ports pour le regroupement de réseaux, ouvrir le fichier à modifier comme suit :

```
~]$ vi ~/teamd_working_configs/activebackup_ethtool_1.conf
```

Éffectuer les changements utiles, et sauvegarder le fichier. Voir la page man **vi(1)** pour obtenir des renseignements sur la façon d'utiliser l'éditeur **vi** ou sur la façon d'utiliser votre éditeur préféré.

Noter qu'il est essentiel que les interfaces qui doivent être utilisées en tant que ports dans le regroupement ne soient pas actives, c'est à dire qu'elles doivent être « down », quand on les ajoute à un périphérique de regroupement. Pour vérifier leur statut, exécutez la commande suivante :

```
~]$ ip link show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode
DEFAULT
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: em1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state
UP mode DEFAULT qlen 1000
    link/ether 52:54:00:d5:f7:d4 brd ff:ff:ff:ff:ff:ff
3: em2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state
UP mode DEFAULT qlen 1000
  link/ether 52:54:00:d8:04:70 brd ff:ff:ff:ff:ff:ff
```

Dans cet exemple, on peut voir que les deux interfaces que nous souhaitons utiliser sont « UP ».

Pour désactiver une interface, exécutez une commande en tant qu'utilisateur **root** sous le format suivant :

```
~]# ip link set down em1
```

Répéter pour chaque interface requise.

Pour créer une interface de regroupement (team) basée sur le fichier de configuration, en tant qu'utilisateur **root**, modifiez le répertoire de travail de configurations (*teamd_working_configs* dans cet exemple) :

```
~]# cd /home/userteamd_working_configs
```

Puis, exécutez une commande au format suivant :

```
~]# teamd -g -f activebackup_ethtool_1.conf -d
Using team device "team0".
```

```
Using PID file "/var/run/teamd/team0.pid"
Using config file
"/home/user/teamd_working_configs/activebackup_ethtool_1.conf"
```

L'option **-g** est pour les messages de débogage, **-f** pour spécifier le fichier de configuration à télécharger, et **-d** pour exécuter le processus en tant que démon après le démarrage. Voir la page man **teamd(8)** pour les autres options.

Pour vérifier le statut du regroupement, exécutez la commande suivante en tant qu'utilisateur **root** :

```
~]# teamdctl team0 state
setup:
  runner: activebackup
ports:
  em1
    link watches:
      link summary: up
      instance[link_watch_0]:
        name: ethtool
        link: up
  em2
    link watches:
      link summary: up
      instance[link_watch_0]:
        name: ethtool
        link: up
runner:
  active port: em1
```

Pour donner une adresse à l'interface d'un regroupement de réseaux, team0, exécutez une commande en tant qu'utilisateur **root** sous le format suivant :

```
~]# ip addr add 192.168.23.2/24 dev team0
```

Pour vérifier l'adresse IP d'une interface de regroupement, exécutez une commande comme suit :

```
~]$ ip addr show team0
4: team0:  <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state
UP
    link/ether 16:38:57:60:20:6f brd ff:ff:ff:ff:ff:ff
    inet 192.168.23.2/24 scope global team0
       valid_lft forever preferred_lft forever
    inet6 2620:52:0:221d:1438:57ff:fe60:206f/64 scope global dynamic
       valid_lft 2591880sec preferred_lft 604680sec
    inet6 fe80::1438:57ff:fe60:206f/64 scope link
       valid_lft forever preferred_lft forever
```

Pour activer une interface de regroupement, ou l'activer (« up »), exécutez une commande en tant qu'utilisateur **root** sous le format suivant :

```
~]# ip link set dev team0 up
```

Pour désactiver une interface de regroupement, de façon temporaire, ou l'activer (« down »), exécutez une commande en tant qu'utilisateur **root** sous le format suivant :

```
~]# ip link set dev team0 down
```

Pour terminer ou annihiler (kill) une instance du démon du regroupement, exécutez une commande en tant qu'utilisateur **root** sous le format suivant :

```
~]# teamd -t team0 -k
```

L'option **-k** est là pour spécifier que l'instance du démon associé au périphérique team0 doit être annihilé (commande - kill). Voir la page **teamd(8)** pour plus d'options.

Pour obtenir de l'aide sur les options en ligne de commandes de **teamd**, exécutez la commande suivante :

```
~]$ teamd -h
```

De plus, voir la page man **teamd(8)**.

5.10.3. Créer un Network Team à l'aide des fichiers ifcfg

Pour créer un Network Team à l'aide des fichiers **ifcfg**, créer un fichier dans le répertoire **/etc/sysconfig/network-scripts/**, comme suit :

```
DEVICE=team0
DEVICETYPE=Team
ONBOOT=yes
BOOTPROTO=none
IPADDR=192.168.11.1
PREFIX=24
TEAM_CONFIG='{"runner": {"name": "activebackup"}, "link_watch": {"name":
"ethtool"}}'
```

Cela aura pour effet de créer l'interface du regroupement (team), c'est à dire, le master.

Pour créer un port qui puisse être un membre de team0, créer un ou plusieurs fichiers dans le répertoire**/etc/sysconfig/network-scripts/**, comme suit :

```
DEVICE=eth1
HWADDR=D4:85:64:01:46:9E
DEVICETYPE=TeamPort
ONBOOT=yes
TEAM_MASTER=team0
TEAM_PORT_CONFIG='{"prio": 100}'
```

Ajouter des interfaces de ports supplémentaires sur le modèle des interfcaces ci-dessus, en changeant les champs DEVICE et HWADDR pour qu'ils correspondent aux ports (les périphériques réseaux) ajoutés. Si la priorité de port n'est pas spécifiée par **prio**, sa valeur par défaut sera **0**; elle accepte des valeurs positives ou négatives allant de **-32,767** à **+32,767**.

Spécifier le matériel ou l'adresse MAC en utilisant la directive **HWADDR** peut influencer le procédure d'affectation de noms. Cela est expliqué dans Chapitre 8, *Nommage de périphériques réseaux consistante*.

Pour activer le regroupement, exécutez la commande suivant en tant qu'utililisateur **root** :

-

```
~]# ifup team0
```

Pour voir le regroupement de réseaux, exécutez la commande suivante :

```
~]$ ip link show
```

5.10.4. Comment ajouter un port à un Network Team à l'aide d'iputils

Pour ajouter un port em1 à un Network Team team0, utilisez **ip**, en exécutant la commande suivante en tant qu'utilisateur **root** :

```
~]# ip link set dev em1 down
~]# ip link set dev em1 master team0
```

Ajouter des ports supplémentaires selon les besoins. Le pilote Team activera les ports automatiquement.

5.10.5. Lister les ports d'un regroupement à l'aide de teamnl

Pour afficher ou lister les ports d'un regroupement de ports, avec l'utilitaire **teamnl**, exécutez la commande suivante, en tant qu'utilisateur **root** :

```
~]# teamnl team0 ports
em2: up 100 fullduplex
em1: up 100 fullduplex
```

5.10.6. Options de configuration d'un regroupement avec teamnl

Pour afficher ou lister les options actuellement disponibles, avec l'utilitaire **teamnl**, exécutez la commande suivante, en tant qu'utilisateur **root** :

```
~]# teamnl team0 options
```

Pour configurer un regroupement en mode backup, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# teamnl team0 setoption mode activebackup
```

5.10.7. Comment ajouter une adresse à un Network Team à l'aide d'iputils

Pour ajouter une adresse à un regroupement team0, utilisez **ip**, en exécutant la commande suivante en tant qu'utilisateur **root** :

```
~]# ip addr add 192.168.252.2/24 dev team0
```

5.10.8. Comment activer une interface dans un Network Team à l'aide d'iputils

Pour activer (« faire apparaître ») une interface dans un regroupement team0, utilisez **ip**, en exécutant la commande suivante en tant qu'utilisateur **root** :

```
~]# ip link set team0 up
```

5.10.9. Afficher les Options de port actif d'un regroupement avec teamnl

Pour afficher ou lister l'option **activeport** dans un regroupement de réseaux, avec l'utilitaire **teamnl**, exécutez la commande suivante, en tant qu'utilisateur **root** :

```
~]# teamnl team0 getoption activeport
0
```

5.10.10. Définir les options de port actives d'un regroupement avec teamnl

Pour définir l'option **activeport** dans un regroupement de réseaux, avec l'utilitaire **teamnl**, exécutez la commande suivante, en tant qu'utilisateur **root** :

```
~]# teamnl team0 setoption activeport 5
```

Pour vérifier le changement dans les options de port du regroupement, exécutez la commande suivant en tant qu'utilisateur **root** :

```
~]# teamnl team0 getoption activeport
5
```

5.11. CONTRÔLE DE TEAMD AVEC TEAMDCTL

Pour vérifier une instance en cours de **teamd** pour chercher des statistiques ou des renseignements de configuration, ou encore, pour apporter une modification, l'outil de contrôle **teamdctl** est utilisé.

Pour afficher l'état actuel d'un regroupement team0, saisir la commande suivante en tant qu'utilisateur **root** :

```
~]# teamdctl team0 state view
```

Pour des sorties plus détaillées :

```
~]# teamdctl team0 state view -v
```

Pour un vidage d'état complet en format JSON (utile pour le traitement machine) de team0, utiliser la commande suivante :

```
~]# teamdctl team0 state dump
```

Pour un vidage de la configuration en format JSON de team0, saisir la commande suivante :

```
~]# teamdctl team0 config dump
```

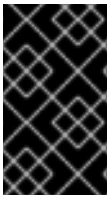
Pour apercevoir la configuration d'un port em1, faisant partie d'un regroupement team0, saisir la commande suivante :

```
~]# teamdctl team0 port config dump em1
```

5.11.1. Ajouter un port à un regroupement de réseaux

Pour ajouter un port em1 à un Network Team team0, utilisez la commande suivante en tant qu'utilisateur **root** :

```
~]# teamdctl team0 port add em1
```



IMPORTANT

Si vous utilisez **teamdctl** directement pour rendre un port dépendant, le port dépendant doit être défini sur *down*. Sinon, la commande **teamdctl team0 port add em1** échouera.

5.11.2. Supprimer un port d'un regroupement de réseaux

Pour supprimer un port em1 à un Network Team team0, utilisez la commande suivante en tant qu'utilisateur **root** :

```
~]# teamdctl team0 port remove em1
```

5.11.3. Appliquer une configuration à un port dans un Network Team

Pour appliquer une configuration de format JSON à un port em1 dans un regroupement de réseaux team0, exécutez une commande en tant qu'utilisateur **root** dans le format suivant :

```
~]# teamdctl team0 port config update em1 JSON-config-string
```

Quand *JSON-config-string* est la configuration sous forme de chaîne de texte en format JSON. Ceci mettra à jour la configuration du port à l'aide de la chaîne de format JSON fournie. Un exemple d'une chaîne JSON valide pour configurer un port ressemble à ce qui suit :

```
{
  "prio": -10,
  "sticky": true
}
```

Utilisez des guillemets simples autour d'une chaîne de configuration JSON et omettre les sauts de ligne.

Notez que l'ancienne configuration sera remplacée et que toutes les options omises seront réinitialisées aux valeurs par défaut. Voir la page de manuel **teamdctl(8)** pour obtenir plusieurs exemples de commandes d'outils de contrôle de démons de team.

5.11.4. Afficher la configuration à un port dans un Network Team

Pour copier la configuration d'un port em1 à un Network Team team0, utilisez la commande suivante en tant qu'utilisateur **root** :

```
~]# teamdctl team0 port config dump em1
```

Pour vider la configuration en format JSON du port dans une sortie standard.

5.12. CONFIGUREZ LES RUNNERS DE TEAMD

Les runners sont des unités de code qui sont ajoutées dans le démon de Team, quand une instance du démon est créée. Pour obtenir une introduction aux runners de **teamd**, voir Section 5.4, « Comprendre le démon de Network Teaming et les "Runners" ».

5.12.1. Configuration du runner de diffusion

Pour configurer le runner de diffusion, à l'aide d'un éditeur, en tant qu'utilisateur **root**, ajouter ce qui suit dans le fichier de configuration au format JSON du regroupement :

```
{
 "device": "team0",
 "runner": {"name": "broadcast"},
 "ports": {"em1": {}, "em2": {}}
}
```

Veuillez consulter la page man**teamd.conf(5)** pour plus d'informations.

5.12.2. Configuration du runner dynamique

Le runner dynamique se comporte de la même façon que le runner roundrobin.

Pour configurer le runner dynamique, à l'aide d'un éditeur, en tant qu'utilisateur **root**, ajouter ce qui suit dans le fichier de configuration au format JSON du regroupement :

```
{
 "device": "team0",
 "runner": {"name": "random"},
 "ports": {"em1": {}, "em2": {}}
}
```

Veuillez consulter la page man**teamd.conf(5)** pour plus d'informations.

5.12.3. Configuration du runner roundrobin

Pour configurer le runner roundrobin, à l'aide d'un éditeur, en tant qu'utilisateur **root**, ajouter ce qui suit dans le fichier de configuration au format JSON du regroupement :

```
{
 "device": "team0",
 "runner": {"name": "roundrobin"},
 "ports": {"em1": {}, "em2": {}}
}
```

Configuration de base de roundrobin

Veuillez consulter la page man**teamd.conf(5)** pour plus d'informations.

5.12.4. Configuration du runner activebackup

Le runner activebackup peut utiliser tous les link-watchers afin de déterminer l'état des liens dans un regroupement. L'un des exemples suivants peut être ajouté au fichier de configuration au format JSON du regroupement :

```
{
   "device": "team0",
   "runner": {
      "name": "activebackup"
   },
   "link_watch": {
      "name": "ethtool"
   },
   "ports": {
      "em1": {
         "prio": -10,
         "sticky": true
      },
      "em2": {
         "prio": 100
      }
   }
}
```

L'exemple de configuration utilise le runner active-backup avec **ethtool** comme link watcher. Le port em2 a une plus grande priorité. Le drapeau em1 (sticky) est rendu actif, et le demeurre tant que le lien est actif lui-même.

```
{
   "device": "team0",
   "runner": {
      "name": "activebackup"
   },
   "link_watch": {
      "name": "ethtool"
   },
   "ports": {
      "em1": {
         "prio": -10,
         "sticky": true,
         "queue_id": 4
      },
      "em2": {
         "prio": 100
      }
   }
}
```

L'exemple de configuration ajoute un ID de file d'attente correspondant à **4**. Il utilise le runner active-backup avec **ethtool** comme link watcher. Le port em2 a une plus grande priorité. Le drapeau em1 (sticky) est rendu actif, et le demeurre tant que le lien est actif.

Pour configurer le runner activebackup, à l'aide d'**ethtool** comme link-watcher et y appliquer un délai, en utilisant un éditeur en tant qu'utilisateur **root**, ajouter ce qui suit dans le fichier de configuration au format JSON du regroupement :

```
{
   "device": "team0",
   "runner": {
      "name": "activebackup"
```

```
    },
    "link_watch": {
        "name": "ethtool",
        "delay_up": 2500,
        "delay_down": 1000
    },
    "ports": {
        "em1": {
            "prio": -10,
            "sticky": true
        },
        "em2": {
            "prio": 100
        }
    }
}
```

L'exemple de configuration utilise le runner active-backup avec **ethtool** comme link watcher. Le port em2 a une plus grande priorité. Le drapeau sticky veille à ce que si em1 est rendu actif, il le demeure quand le lien est actif. Les changements de liens ne sont pas propagés par le runner immédiatement, mais des délais s'appliquent.

Veuillez consulter la page man **`teamd.conf(5)`** pour plus d'informations.

5.12.5. Configuration du runner d'équilibrage des charges

Ce runner peut être utilisé pour deux types d'équilibrage de charge, en mode actif et passif. En mode actif, un rééquilibrage constant du trafic s'effectue en utilisant des statistiques de trafic récents pour répartir le trafic aussi uniformément que possible. En mode statique, les flux de trafic sont répartis au hasard sur les liens disponibles. Cela favorise la vitesse d'exécution en raison de la réduction de la charge de traitement. C'est dans les applications de trafic de haut volume que c'est souvent apprécié, car le trafic est généralement constitué de plusieurs flux de données distribués au hasard entre les liens disponibles, et de cette manière, le partage des charges s'effectue sans intervention de la part de **`teamd`**.

Pour configurer le runner d'équilibrage des charges en transmission passive (Tx), à l'aide d'un éditeur, en tant qu'utilisateur **`root`**, ajouter ce qui suit dans le fichier de configuration au format JSON du regroupement :

```
{
 "device": "team0",
 "runner": {
   "name": "loadbalance",
   "tx_hash": ["eth", "ipv4", "ipv6"]
 },
 "ports": {"em1": {}, "em2": {}}
}
```

Configuration de l'équilibrage des charges de transmission passive (Tx) basé hachage.

Pour configurer le runner d'équilibrage des charges en transmission active (Tx), à l'aide d'un éditeur, en tant qu'utilisateur **`root`**, ajouter ce qui suit dans le fichier de configuration au format JSON du regroupement :

```
{
```

```
    "device": "team0",
    "runner": {
      "name": "loadbalance",
      "tx_hash": ["eth", "ipv4", "ipv6"],
      "tx_balancer": {
        "name": "basic"
      }
    },
    "ports": {"em1": {}, "em2": {}}
}
```

Configuration de l'équilibrage des charges de transmission active (Tx) à l'aide d'un équilibreur des charges de base.

Veuillez consulter la page man **`teamd.conf(5)`** pour plus d'informations.

5.12.6. Configuration du runner LACP (802.3ad)

Pour configurer le runner LACP, à l'aide d'**ethtool** comme link-watcher, en utilisant un éditeur en tant qu'utilisateur **`root`**, ajouter ce qui suit dans le fichier de configuration au format JSON du regroupement :

```
{
    "device": "team0",
    "runner": {
        "name": "lacp",
        "active": true,
        "fast_rate": true,
        "tx_hash": ["eth", "ipv4", "ipv6"]
    },
      "link_watch": {"name": "ethtool"},
      "ports": {"em1": {}, "em2": {}}
}
```

La configuration de la connexion à un homologue pouvant utiliser le protocole *link aggregation control protocol* (LACP). Le runner LACP doit utiliser **ethtool** pour surveiller l'état d'un lien. Il ne sert à rien d'utiliser n'importe quelle autre méthode de surveillance des liens, mise à part **ethtool** parce que, par exemple, dans le cas de **arp_ping**, le lien n'apparaissait pas ; la raison étant que le lien doit être tout d'abord établi, et seulement après, les paquets, ARP inclus, peuvent passer. L'utilisation de **ethtool** empêche cela, car il surveille chaque couche de liens individuellement.

L'équilibrage des charges actif est possible dans ce runner de la même façon qu'il est possible dans le runner d'équilibrage des charges. Pour activer l'équilibrage des charges de transmission active (Tx), ajouter la section suivante :

```
"tx_balancer": {
        "name": "basic"
}
```

Veuillez consulter la page man **`teamd.conf(5)`** pour plus d'informations.

5.12.7. Configurer le monitoring des états de liens

Les méthodes suivantes de surveillance des états de liens sont disponibles. Pour implémenter une des méthodes, ajouter la chaîne de format JSON au fichier de configuration de format JSON de Team, à l'aide de l'éditeur, en tant qu'utilisateur **`root`**.

5.12.7.1. Configurer Ethtool pour le monitoring des états de liens

Pour ajouter ou modifier un délai existant, en millisecondes, entre le lien à venir et le runner qui est notifié à ce sujet, ajoutez ou modifiez un article comme suit :

```
"link_watch": {
       "name": "ethtool",
       "delay_up": 2500
}
```

Pour ajouter ou modifier un délai existant, en millisecondes, entre le lien en voie de désactivation et le runner qui est notifié à ce sujet, ajoutez ou modifiez un article comme suit :

```
"link_watch": {
       "name": "ethtool",
       "delay_down": 1000
}
```

5.12.7.2. Configurer ARP Ping pour le monitoring d'états de liens

Le démon team **teamd** envoie une demande ARP REQUEST à une adresse tout en fin de lien afin de déterminer si le lien est actif. La méthode utilisée est la même que celle de l'utilitaire **arping**, sauf qu'il n'est pas utilisé.

Préparer un fichier contenant la nouvelle configuration en format JSON sur le modèle suivant :

```
{
       "device": "team0",
       "runner": {"name": "activebackup"},
       "link_watch":{
           "name": "arp_ping",
           "interval": 100,
           "missed_max": 30,
           "source_host": "192.168.23.2",
           "target_host": "192.168.23.1"
       },
         "ports": {
             "em1": {
               "prio": -10,
               "sticky": true
             },
             "em2": {
               "prio": 100
             }
         }
}
```

Cette configuration utilise **arp_ping** comme link watcher. L'option **missed_max** correspond à la valeur limite du nombre maximal de réponses manquées (réponses ARP, par exemple). Doit être combiné avec l'option **interval** afin de déterminer la durée totale avant qu'un lien puisse être signalé comme étant inactif.

Pour télécharger une nouvelle configuration d'un port em2 de regroupement, à partir d'un fichier contenant une configuration JSON, utilisez la commande suivante en tant qu'utilisateur **root** :

```
~]# port config update em2 JSON-config-file
```

Notez que l'ancienne configuration sera remplacée et que toutes les options omises seront réinitialisées aux valeurs par défaut. Voir la page de manuel **teamdctl(8)** pour obtenir plusieurs exemples de commandes d'outils de contrôle de démons de team.

5.12.7.3. Configurer IPv6 NA/NS pour le monitoring des états de liens

```
{
  "device": "team0",
    "runner": {"name": "activebackup"},
    "link_watch": {
    "name": "nsna_ping",
    "interval": 200,
    "missed_max": 15,
    "target_host": "fe80::210:18ff:feaa:bbcc"
    },
      "ports": {
        "em1": {
          "prio": -10,
          "sticky": true
        },
        "em2": {
          "prio": 100
        }
      }
}
```

Pour configurer l'intervalle entre les paquets NS/NA, ajouter ou modifier une section, comme suit :

```
"link_watch": {
    "name": "nsna_ping",
    "interval": 200
}
```

La valeur correspond à un nombre positif en millisecondes. Elle doit être choisie en conjonction à l'option **missed_max** pour déterminer la durée totale avant qu'un lien soit signalé comme étant inactif.

Pour configurer le nombre maximum de paquets de réponses NS/NA pour permettre de rapporter que le lien est inactif, ajouter ou modifier une section, comme suit :

```
"link_watch": {
    "name": "nsna_ping",
    "missed_max": 15
}
```

Nombre maximal de paquets de réponse NS/NA manquées. Si ce nombre est dépassé, le lien est signalé comme étant inactif. L'option **missed_max** correspond à la valeur limite du nombre maximal de réponses manquées (réponses ARP, par exemple). Doit être combiné avec l'option **interval** afin de déterminer la durée totale avant qu'un lien puisse être signalé comme étant inactif.

Pour configurer le nom d'hôte qui est résolu en adresse cible **IPv6** de paquets NS/NA, ajouter ou modifier une section, comme suit :

```
"link_watch": {
    "name": "nsna_ping",
    "target_host": "MyStorage"
}
```

L'option « target_host » contient le nom d'hôte à convertir en adresse **IPv6** qui sera utilisée comme adresse cible pour les paquets NS/NA. Une adresse **IPv6** pourra être utilisée à la place du nom d'hôte.

Veuillez consulter la page man **teamd.conf(5)** pour plus d'informations.

5.12.8. Configurer Sélection de port de substitution

Le port physique qui transmet une image est normalement sélectionné par la partie du noyau du pilote de l'agrégat (team) et n'est pas pertinent à l'utilisateur ou à l'administrateur système. Le port de sortie est sélectionné à l'aide des politiques du mode d'agrégat sélectionné (**teamd** runner), À l'occasion, toutefois, il est utile de diriger certaines classes de trafic sortant vers certaines interfaces physiques pour appliquer des politiques un peu plus complexes. Par défaut, le pilote team est sensible aux files d'attente multiples et 16 files d'attente sont créées lors de l'initialisation du pilote. Si on souhaite davantage ou moins de files d'attente, l'attribut **tx_queues** peut être utilisé pour modifier cette valeur lors de la création d'instance de pilote team.

L'ID de file d'attente d'un port peut être défini par l'option de configuration du port **queue_id** comme suit :

```
{
  "queue_id": 3
}
```

Ces ID de files d'attente peuvent être utilisées en conjonction à l'utilitaire **tc** afin de configurer une discipline de file d'attente pour files multiples, et des filtres destinés à certains trafics devant être transmis à des périphériques de port particuliers. Ainsi, si vous souhaitez utiliser la configuration ci-dessus et que vous souhaitiez forcer tout le trafic lié à **192.168.1.100**, pour utiliser eth1 dans l'équipe (team) faisant office de périphérique de sortie, exécutez la commande en tant qu'utilisateur **root** sous le format suivant :

```
~]# tc qdisc add dev team0 handle 1 root multiq
~]# tc filter add dev team0 protocol ip parent 1: prio 1 u32 match ip dst \
  192.168.1.100 action skbedit queue_mapping 3
```

Ce mécanisme de remplacement de logique de runner afin de relier le trafic à un port particulier peut être utilisé avec n'importe quel runner.

5.12.9. Configurer Sélectionneurs de ports Tx basés BPF

L'équilibrage des charges et les runners LACP utilisent les hachages de paquets pour trier les flux de trafic réseau. Le mécanisme de calcul de hachage est basé sur le code *Berkeley Packet Filter* (BPF). Le code BPF est utilisé pour générer un hachage au lieu de prendre une décision politique au sujet des paquets sortants. La longueur de hachage est de 8 bits, ce qui donne 256 variantes. Cela signifie que différents *tampons socket* (SKB) peuvent avoir le même hachage et donc faire passer le trafic sur un même lien. L'utilisation d'un hachage court est un moyen rapide de trier le trafic en flux divers à but d'équilibrage des charges sur des liens multiples. En mode statique, le hachage sert uniquement à décider à partir de quel port le trafic doit être envoyé. En mode actif, le runner redéfinira continuellement les hachages de ports différents pour tenter de parvenir à un équilibre parfait.

Les types de chaînes ou de fragments suivants peuvent être utilisés pour le calcul d'un hachage Tx de paquet :

- **eth** — Utilise les adresses MAC source et destination
- **vlan** — Utilise ID VLAN.
- **ipv4** — Utilise les adresses de source et de destination **IPv4**.
- **ipv6** — Utilise les adresses de source et de destination **IPv6**.
- **ip** — Utilise les adresses de source et de destination **IPv4** et **IPv6**.
- **l3** — Utilise les adresses de source et de destination **IPv4** et **IPv6**.
- **tcp** — Utilise les ports de source et de destination **TCP**.
- **udp** — Utilise les ports de source et de destination **UDP**.
- **sctp** — Utilise les ports de source et de destination **SCTP**.
- **l4** — Utilise les ports de source et de destination **TCP** et **UDP** et **SCTP**.

Ces chaînes peuvent être utilisées en ajoutant une ligne au format suivant au runner d'équilibrage des charges : exemple

```
"tx_hash": ["eth", "ipv4", "ipv6"]
```

See Section 5.12.5, « Configuration du runner d'équilibrage des charges ».

5.13. CRÉER UN NETWORK TEAM PAR L'INTERFACE GRAPHIQUE (GUI)

5.13.1. Établir une connexion Team

Vous pouvez utiliser l'utilitaire **control-centrer** de GNOME pour demander au **NetworkManager** de créer un agrégat (team) à partir de deux ou plusieurs connexions filaires ou InfiniBand. Il n'est pas nécessaire de créer les connexions à regrouper entre elles pour commencer. Elles peuvent être configurées au moment de la configuration de l'agrégat (team). Vous devez avoir les adresses MAC des interfaces disponibles pour compléter le processus de configuration.

Procédure 5.1. Ajout d'une nouvelle connexion Team

Suivre les étapes suivantes pour ajouter une nouvelle connexion Team.

1. Appuyer sur la touche **Super** pour accéder au menu Activités, saisir **control network**, et appuyez sur la touche **Enter**. L'outil de configuration du **Réseau** apparaîtra. Cette étape est totalement expliquée dans Section 2.5, « Utiliser le NetworkManager avec l'interface graphique GNOME ».
2. Cliquer sur le signe plus pour ouvrir la liste de sélection. Sélectionner **Team**. La fenêtre **Modifier Connexion Team1** apparaîtra.

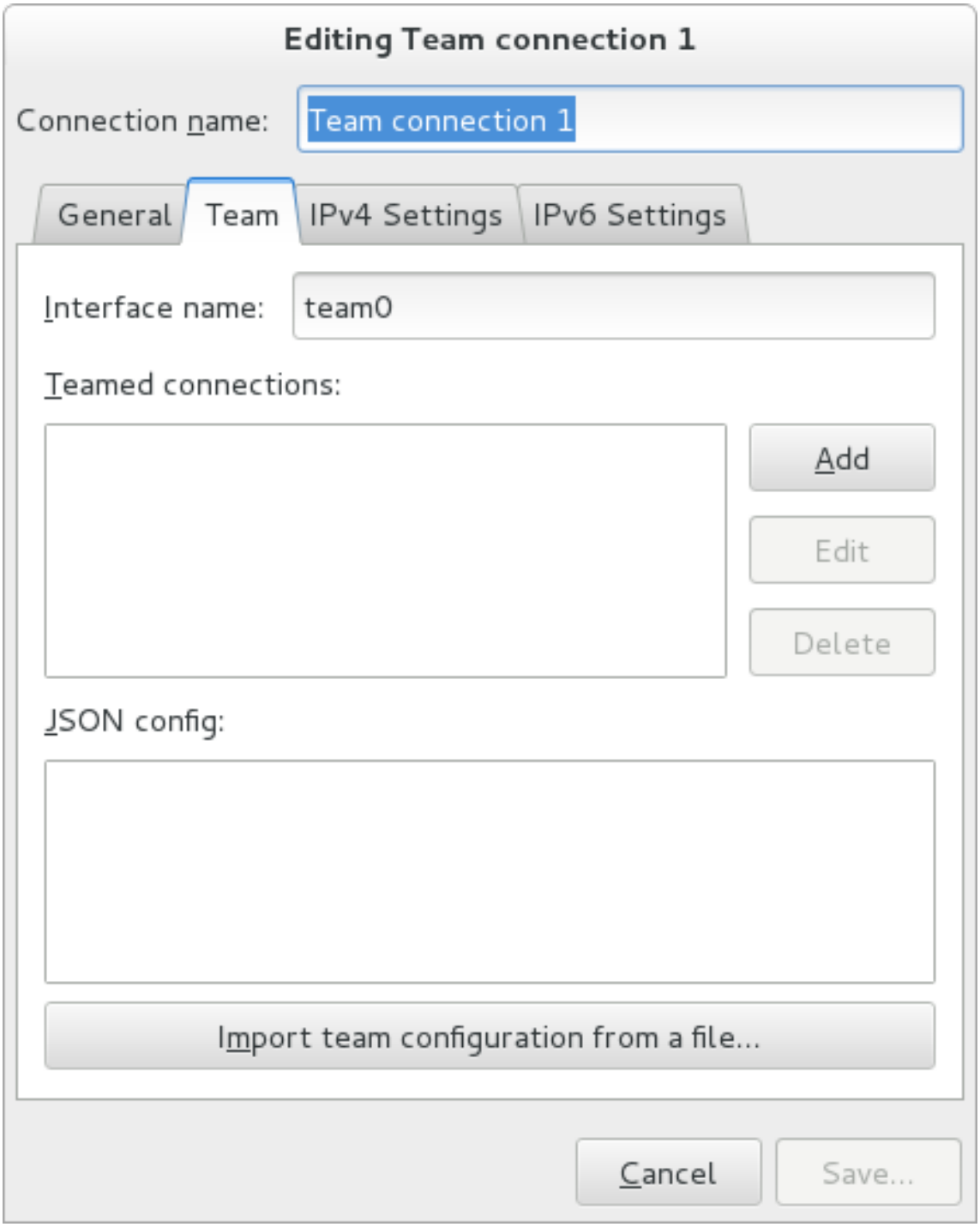


Figure 5.6. Pour que l'interface utilisateur graphique du NetworkManager ajoute un menu

3. Dans l'onglet **`Team`** (ou Liaisons), cliquez sur **`Ajouter`** et sélectionnez le type d'interface que vous souhaitez utiliser avec la connexion team. Cliquez sur le bouton **`Créer`**. Notez que la boîte de dialogue pour sélectionner le type de port n'apparaît uniquement que lorsque vous créez le premier port. Après cela, ce même type sera utilisé automatiquement pour toutes les autres ports.

4. La fenêtre **`Editing team0 port 1`** apparaît. Remplir l'adresse MAC dans la première interface à ajouter à l'agrégat (team).

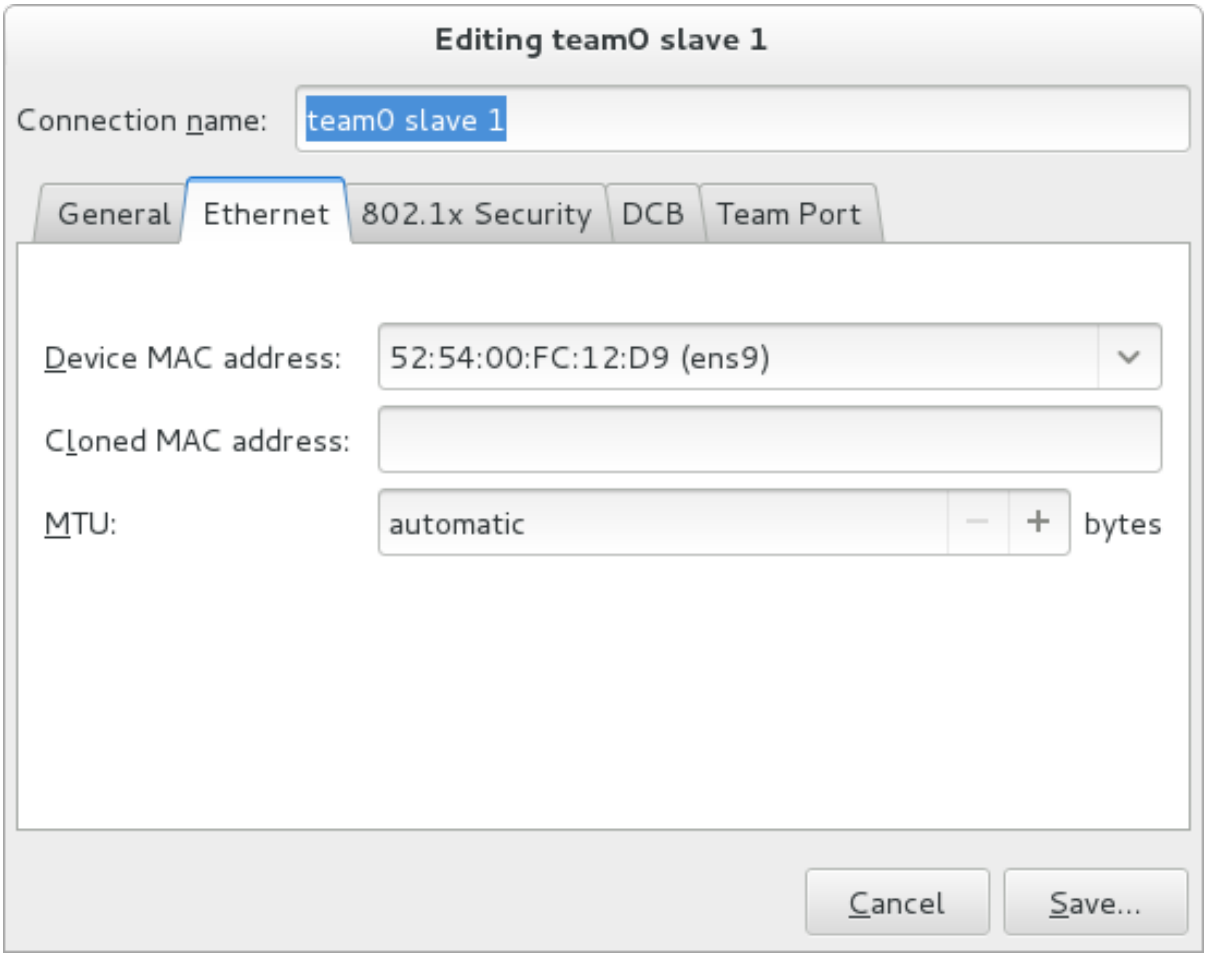


Figure 5.7. Pour que l'interface utilisateur graphique du NetworkManager ajoute une connexion esclave

5. Si on doit appliquer des configurations de ports personnalisées, cliquer sur l'onglet **`Team Port`** et saisir la chaîne de configuration JSON ou l'importer d'un fichier.

6. Cliquez sur le bouton **`Enregistrer`**.

7. Le nom de l'esclave teamed apparaît dans la fenêtre **`Teamed connexions`**. Cliquez sur le bouton **`Ajouter`** pour ajouter des connexions de ports supplémentaires.

8. Vérifier et confirmer les paramètres de configuration, puis cliquer sur le bouton **`Sauvegarder`**.

9. Modifier la configuration spéciale team en consultant Section 5.13.1.1, « Configurer l'onglet Team ».

Procédure 5.2. Modifier une connexion team existante

Suivre les étapes ci-dessous pour modifier une connexion team existante.

1. Appuyer sur la touche **`Super`** pour accéder au menu Activités, saisir **`control network`**, et appuyez sur la touche **`Entrée`**. L'outil de configuration du **Réseau** apparaîtra.

2. Sélectionner la connexion à modifier et cliquer sur le bouton **`Options`**.

3. Sélectionner l'onglet **`Général`**.

4. Configurer le nom de la connexion, le comportement auto-connect, et les paramètres disponibles.

   Il existe cinq configurations de la boîte de dialogue **Modifier** qui sont communes à tous les types de connexion. Voir l'onglet **Général** :

   - **Nom de connexion** — saisir un nom descriptif pour votre connexion de réseau. Ce nom sera utilisé pour lister cette connexion dans le menu de la fenêtre **Réseau**.
   - **Se connecter automatiquement à ce réseau quand il est disponible** — sélectionnez cette case si vous souhaitez que le **NetworkManager** s'auto-connecte à cette connexion quand elle sera disponible. Voir Section 2.5.3, « Se connecter à un réseau automatiquement » pour plus d'informations.
   - **Rendre le réseau disponible à tous les utilisateurs** — sélectionnez cette case pour créer une connexion disponible à tous les utilisateurs sur le système. Changer ce paramètre peut nécessiter des privilèges d'utilisateur root. Consulter Section 2.5.4, « Profils de connexions privées ou sur tout le système » pour obtenir plus d'informations.
   - **Se connecter automatiquement au VPN quand on utilise cette connexion** — sélectionnez cette case si vous souhaitez que le **NetworkManager** s'auto-connecte à une connexion de VPN quand il est disponible. Sélectionner le VPN à partir du menu déroulant.
   - **Zone de parefeu** — sélectionnez une zone de parefeu dans le menu déroulant. Voir le guide *Red Hat Enterprise Linux 7 Security Guide* pour obtenir plus d'informations sur les zones de pare-feu.

5. Modifier la configuration spéciale team en consultant Section 5.13.1.1, « Configurer l'onglet Team ».

Sauvegarder votre nouvelle connexion (ou votre connexion modifiée) et faire des configurations supplémentaires

Une fois vous aurez terminé de modifier votre connection team au réseau local virtuel, cliquez sur le bouton **Enregistrer** pour enregistrer votre configuration personnalisée. Si le profil est en cours d'utilisation lors de la modification, alimentez le cycle de connexion pour que le **NetworkManager** applique les modifications. Si le profil est désactivé (OFF), réglez-le sur ON ou sélectionnez-le dans le menu de l'icône de connexion réseau. Voir Section 2.5.1, « Se connecter à réseau par un GUI » pour plus d'informations sur l'utilisation de votre connexion nouvelle ou modifiée.

Vous pouvez configurer davantage une connexion existante en la sélectionnant dans la fenêtre **Réseau** et en cliquant sur **Options** pour revenir à la boîte de dialogue **Modifier**.

Puis, pour configurer :

- Paramètres de configuration **IPv4** pour la connection, cliquer sur l'onglet **IPv4 Settings** et continuer avec Section 2.5.10.4, « Configuration des paramètres IPv4 »; ou,
- Paramètres de configuration pour la connexion, cliquer sur l'onglet **IPv6 Settings**et continuez avec Section 2.5.10.5, « Configurer les paramètres IPv6 ».

Après la sauvegarde, Team apparaîtra dans l'outil de configuration du réseau.

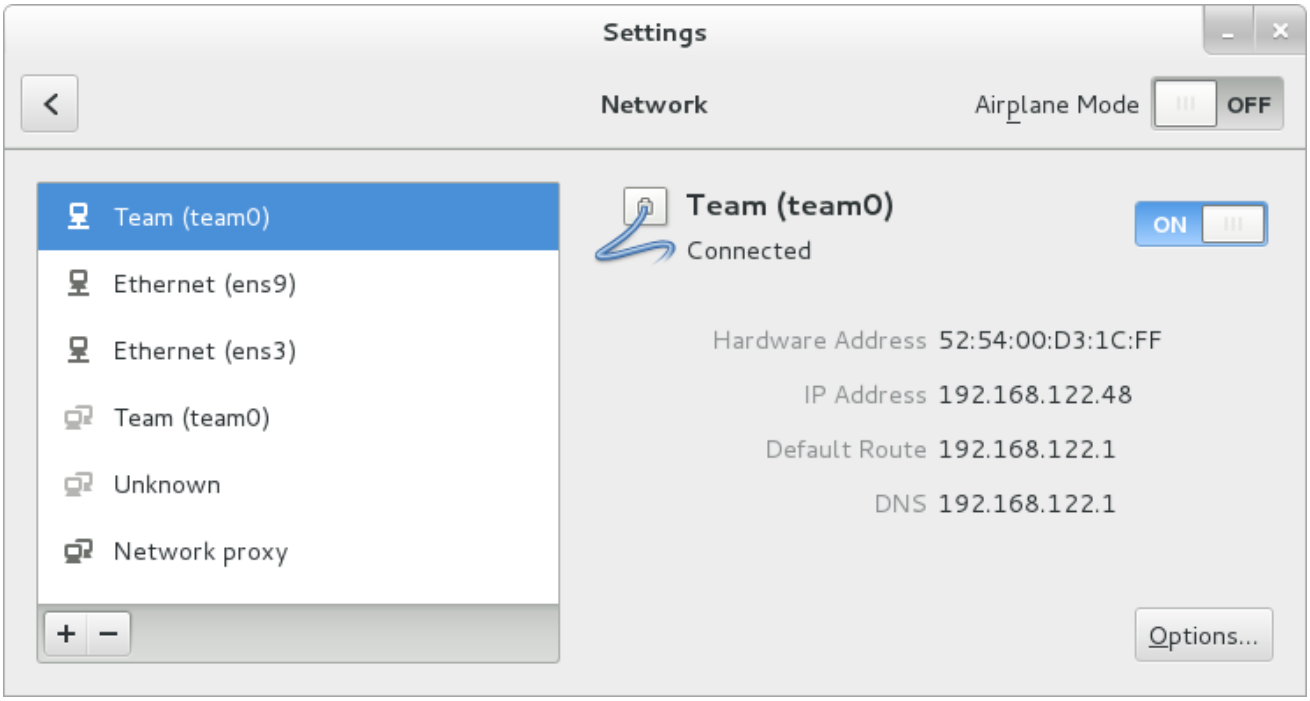


Figure 5.8. Interface utilisateur graphique du NetworkManager avec Team

5.13.1.1. Configurer l'onglet Team

Si vous avez déjà ajouté une nouvelle connexion team, vous pourrez entrer une chaîne de configuration JSON personnalisée dans la zone de texte ou importer un fichier de configuration. Cliquez sur **`Enregistrer`** pour appliquer la configuration de JSON à l'interface team.

Pour obtenir des exemples de chaînes JSON, voir Section 5.12, « Configurez les runners de teamd »

Voir Procédure 5.1, « Ajout d'une nouvelle connexion Team » pour obtenir des instructions sur la façon d'ajouter un nouvel agrégat (team).

5.14. RESSOURCES SUPPLÉMENTAIRES

Les sources d'informations suivantes fournissent des ressources supplémentaires à propos des associations de réseaux.

5.14.1. Documentation installée

- Page man **`teamd(8)`** — Décrit le service **`teamd`**.
- Page man **`teamdctl(8)`** — Décrit l'outil de contrôle **`teamd`**.
- Page man **`teamd.conf(5)`** — décrit le fichier de configuration du **`teamd`**.
- Page man **`teamnl(8)`** — Décrit la bibliothèque Netlink du **`teamd`**.
- Page man **`bond2team(1)`** — décrit un outil pour convertir les options de liaison à team.

5.14.2. Documentation en ligne

http://www.w3schools.com/json/json_syntax.asp

Une explication de la syntaxe de JSON.

CHAPITRE 6. CONFIGUREZ LES PONTAGES DE RÉSEAU

Un pont réseau est un périphérique de couche de liaison qui réachemine le trafic entre des réseaux sur la base des adresses MAC. Il prend des décisions de réacheminement basées sur une table d'adresses MAC qu'il crée en écoutant le trafic réseau, et ainsi, apprend quels hôtes sont connectés à chaque réseau. Un pont de logiciel peut être utilisé dans un hôte Linux afin d'émuler un pont de matériel, par exemple pour que des applications de virtualisation puissent partager une carte réseau (NIC) avec un ou plusieurs cartes virtuelles.

Notez qu'un pont ne peut pas être établi sur des réseaux Wi-Fi qui opèrent en modes *Ad-Hoc* ou *Infrastructure*. Cela est dû au standard IEEE 802.11 qui spécifie l'utilisation de structures 3-adresses en Wi-Fi pour une utilisation optimum des heures de diffusion.

6.1. CONFIGURER LE PONTAGE PAR L'INTERFACE TEXTE UTILISATEUR, NMTUI

L'outil d'interface utilisateur de texte **nmtui** peut être utilisé pour configurer le pontage dans une fenêtre de terminal. Exécutez la commande suivante pour démarrer l'outil :

```
~]$ nmtui
```

L'interface utilisateur texte apparaîtra. Tout commande non valide affichera un message d'utilisation.

Pour naviguer, utiliser les flèches ou appuyer sur **Tab** pour continuer et appuyer sur la combinaision de touches **Maj**+**Tab** pour revenir aux options. Appuyer sur la touche **Entrée** pour sélectionner une option. La barre **Espace** active/désactive le statut d'une case à cocher.

1. À partir du menu de démarrage, sélectionner **Modifier une connexion**. Sélectionner **Ajouter**, l'écran **Nouvelle connexion** apparaîtra.

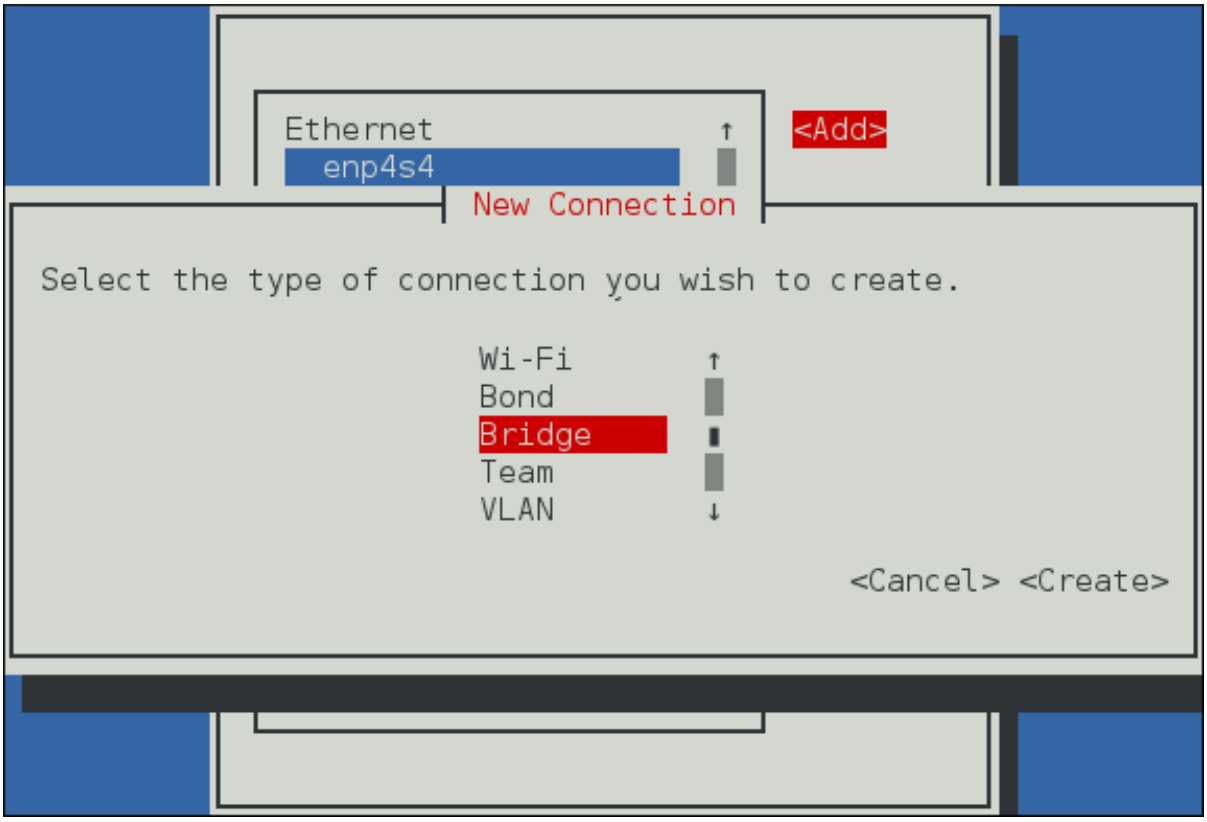


Figure 6.1. Pour que l'interface utilisateur texte du NetworkManager ajoute un menu de connexion de pont.

2. Sélectionner **Pont**, l'écran **Modifier Connexion** apparaîtra.

3. Pour ajouter des interfaces esclaves au pont, sélectionner **Ajouter**, et l'écran **Nouvelle connexion** apparaîtra. Une fois que le type de connexion aura été sélectionné, appuyer sur le bouton **Créer** pour que l'écran **Modifier Connexion** puisse apparaître.

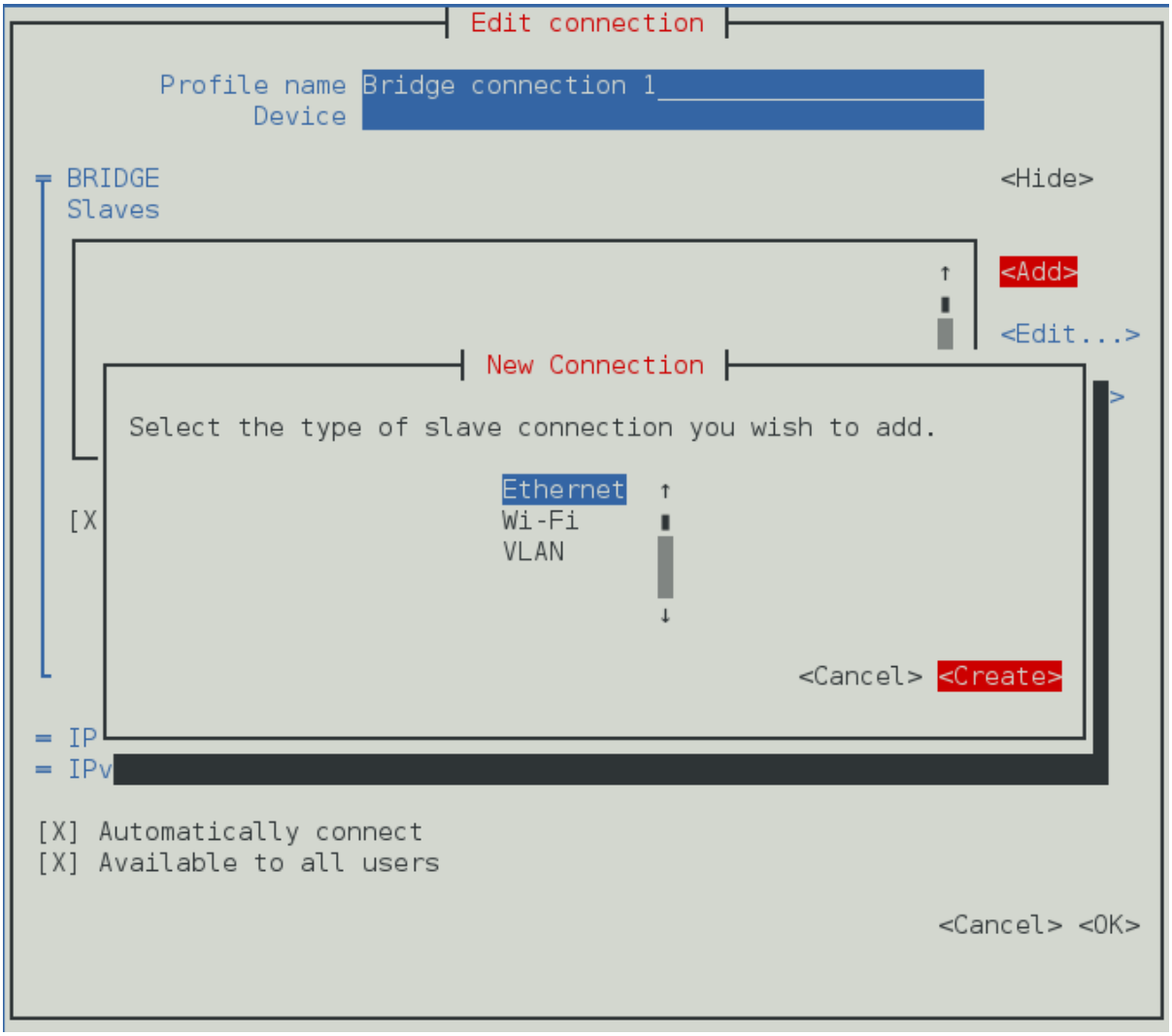
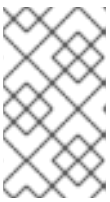


Figure 6.2. Pour que l'interface utilisateur texte du NetworkManager ajoute un nouveau menu de connexion de pont esclave

4. Saisir l'adresse MAC ou le nom de périphérique de l'esclave que vous aurez choisi dans la section **Périphériques**. Si besoin est, saisir une adresse MAC clonée à utiliser comme adresse MAC de pontage, en sélectionnant **Afficher** à droite de l'étiquette **Ethernet**. Sélectionnez le bouton **OK**.



NOTE

Si le périphérique est spécifié sans adresse MAC, la section **Périphérique** sera remplie automatiquement une fois que la fenêtre **Modifier Connexion** est chargée, mais uniquement s'il trouve le périphérique.

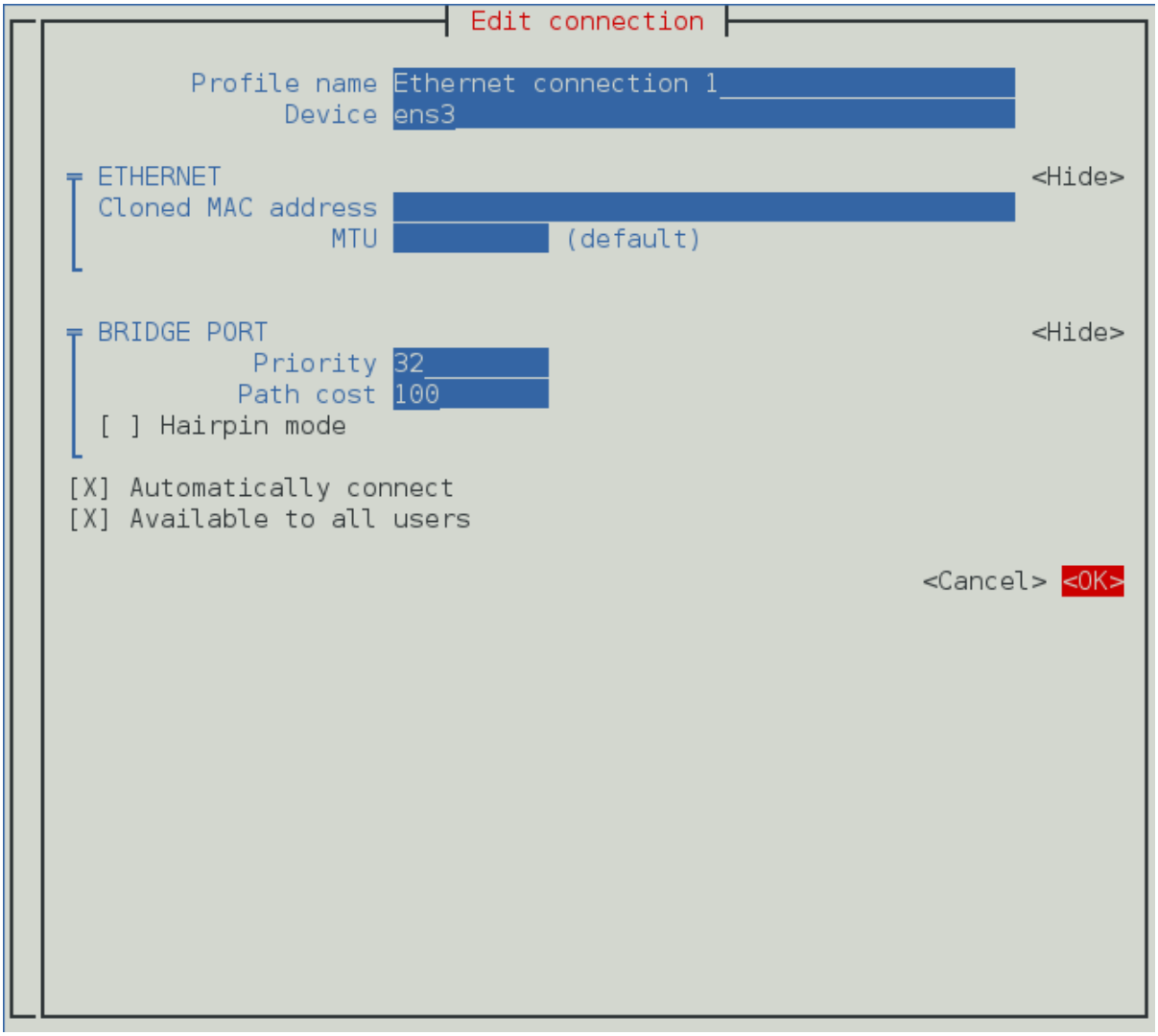


Figure 6.3. Pour que l'interface texte d'utilisateur du NetworkManager configure un menu de connexion de pont esclave

5. Le nom du pont esclave apparaît dans la section **`Esclaves`**. Répétez les étapes ci-dessus pour ajouter des connexions esclaves.

6. Vérifier et confirmer les paramètres de configuration, avant de cliquer sur le bouton **`OK`**.

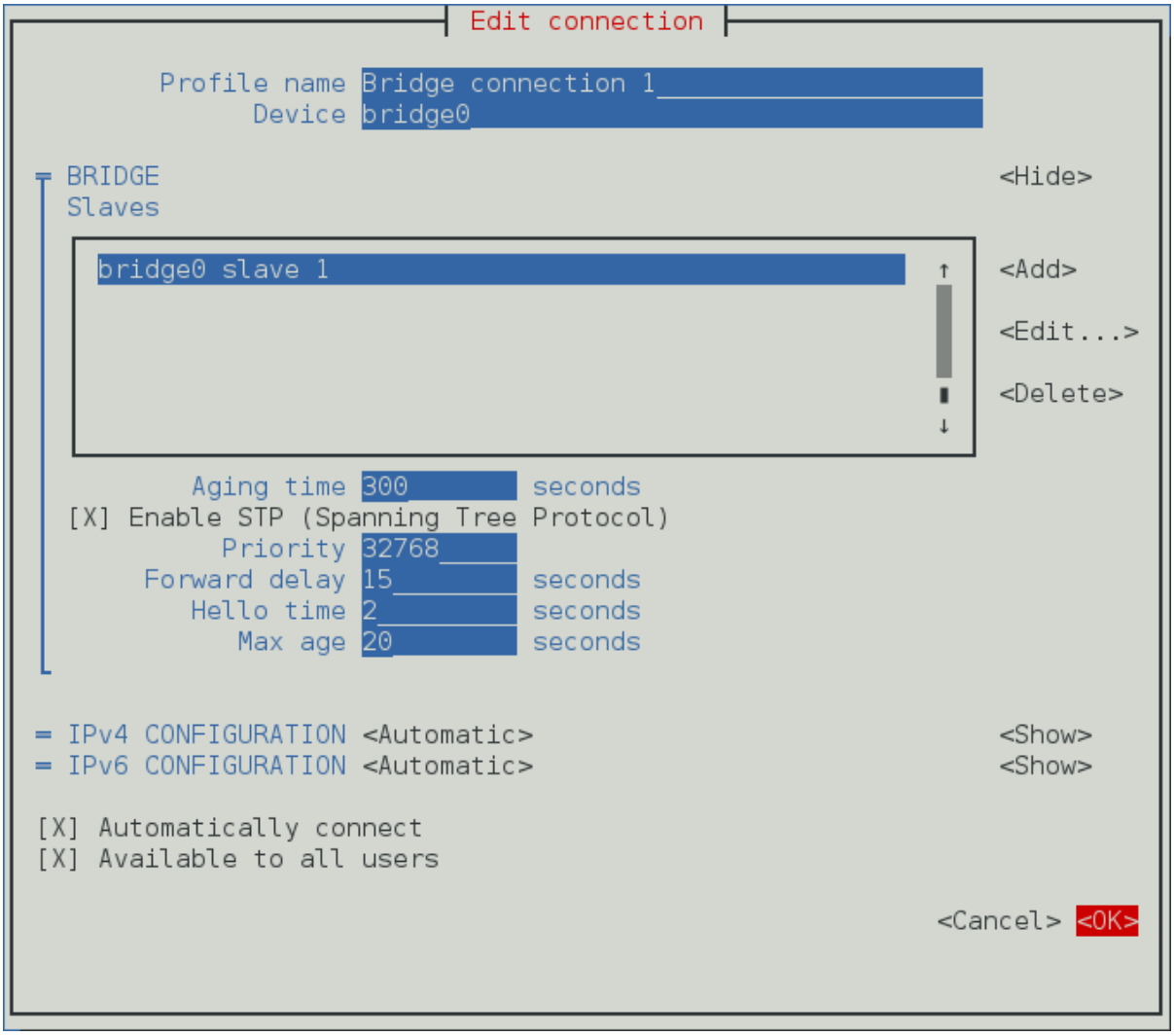


Figure 6.4. Pour que l'interface texte d'utilisateur du NetworkManager configure un menu de pontage

Voir Section 6.4.1.1, « Configurer l'onglet Pont » pour obtenir des définitions des termes de pontage.

Voir Section 1.5, « Configuration réseau utilisant une interface utilisateur texte (nmtui) » pour obtenir des informations sur la façon d'installer **nmtui**.

6.2. UTILISER L'OUTIL DE LIGNE DE COMMANDES DU NETWORKMANAGER, NMCLI

Pour créer un pont, nommé bridge-br0, exécutez une commande comme suit, en tant qu'utilisateur **root** :

```
~]# nmcli con add type bridge ifname br0
Connection 'bridge-br0' (6ad5bba6-98a0-4f20-839d-c997ba7668ad)
successfully added.
```

Si aucun nom d'interface n'est spécifié, le nom aura comme valeur par défaut bridge, bridge-1, bridge-2, etc.

Pour voir les connexions, exécutez la commande suivante :

```
~]$ nmcli con show
NAME        UUID                                  TYPE
```

```
DEVICE
bridge-br0  79cf6a3e-0310-4a78-b759-bda1cc3eef8d  bridge          br0
eth0        4d5c449a-a6c5-451c-8206-3c9a4ec88bca  802-3-ethernet  eth0
```

Le protocole *Spanning Tree Protocol* (STP) est activé par défaut. Les valeurs utilisées viennent du standard IEEE 802.1D-1998. Pour désactiver **STP** pour ce pont, exécutez la commande suivante en tant qu'utilisateur **root** :

```
~]# nmcli con modify bridge-br0 bridge.stp no
```

Pour réactiver **802.1D STP** pour ce pont, exécutez une commande comme suit, en tant qu'utilisateur **root** :

```
~]# nmcli con modify bridge-br0 bridge.stp yes
```

La priorité de pont par défaut du protocole **802.1 D STP** est **32768**. Le nombre inférieur est préféré dans la sélection de pont racine. Par exemple, un pont avec une priorité **28672** serait sélectionné comme étant pont racine de préférence par rapport à un pont avec une priorité **32768** (par défaut). Pour créer un pont sans valeur par défaut, excécutez la commande suivante :

```
~]$ nmcli con add type bridge ifname br5 stp yes priority 28672
Connection 'bridge-br5' (86b83ad3-b466-4795-aeb6-4a66eb1856c7)
successfully added.
```

Les valeurs autorisées sont dans la plage **0** à **65535**.

Pour modifier la priorité de pontage d'un pont existant à une valeur qui n'est pas une valeur par défaut, exécutez une commande dans le format suivant :

```
~]$ nmcli connection modify bridge-br5 bridge.priority 36864
```

Les valeurs autorisées sont dans la plage **0** à **65535**.

Pour voir les paramètres de configuration du pont, exécutez la commande suivante :

```
~]$ nmcli -f bridge con show bridge-br0
```

D'autres options de **802.1D STP** se trouvent dans la section de pontage de la page man **nmcli(1)**.

Pour ajouter, ou pour mettre une interface en esclavage, comme par exemple eth1,au pont bridge-br0, exécutez une commande qui ressemble à ceci :

```
~]$ nmcli con add type bridge-slave ifname eth1 master bridge-br0
Connection 'bridge-slave-eth1' (70ffae80-7428-4d9c-8cbd-2e35de72476e)
successfully added.
```

Au moment de la rédaction, **nmcli** ne prend en charge que les connexions Ethernet esclaves.

Pour modifier une valeur par le mode interactif, exécutez la commande suivante :

```
~]$ nmcli connection edit bridge-br0
```

Vous serez placé par invitation **nmcli**.

```
nmcli> set bridge.priority 4096
nmcli> save
Connection 'bridge-br0' (79cf6a3e-0310-4a78-b759-bda1cc3eef8d)
successfully saved.
nmcli> quit
```

Voir Section 2.3, « Utiliser l'outil de ligne de commandes du NetworkManager, nmcli » pour une introduction à **nmcli**.

6.3. UTILISATION DE L'INTERFACE EN LIGNE DE COMMANDES (CLI)

6.3.1. Vérifier si le module Bridging Kernel est installé

Dans Red Hat Enterprise Linux 7, le module de pontage est téléchargé par défaut. Si nécessaire, vous pouvez vérifier que le module soit bien chargé en exécutant la commande suivante en tant qu'utilisateur **root** :

```
~]# modprobe --first-time bridge
modprobe: ERROR: could not insert 'bridge': Module already in kernel
```

Pour afficher des informations sur le module, excuter la commande suivante :

```
~]$ modinfo bridge
```

Consulter la page man **modprobe(8)** pour plus d'options de commandes.

6.3.2. Créer un pontage de réseau

Pour créer un pontage de réseau, créer un fichier dans le répertoire **/etc/sysconfig/network-scripts/** nommé **ifcfg-br*N***, et remplacer *N* par le numéro de l'interface, par exemple **0**.

Le contenu du fichier est similaire à n'importe quel type d'interface en cours de processus de pontage, comme une interface Ethernet. Les différences principales sont illustrées dans l'exemple suivant :

- La directive **DEVICE** reçoit une nom d'interface comme argument ayant un format **br*N***, avec *N* qui est remplacé par le numéro de l'interface.
- La directive **TYPE** reçoit l'argument **Pont**. Cette directive détermine le type de périphérique et l'argument est sensible à la casse.
- Le fichier de configuration de l'interface de pontage reçoit une adresse **IP** tandis que le fichier de confiuguration de l'interface physique doit avoir une adresse MAC uniquement (voir ci-dessous).
- Une directive supplémentaire, **DELAY=0**, est ajoutée pour empêcher le pont d'attendre alors qu'il surveille le trafic, apprend où se trouvent les hôtes, et crée une table des adresses MAC sur lesquelles fonder ses décisions de filtrage. Le délai par défaut de 15 secondes n'est pas nécessaire si aucune boucle de routage n'est possible.

Exemple 6.1. Exemple de fichier de configuration d'interface ifcfg-br0

Ce qui suit est un exemple de fichier de configuration d'interface de pontage qui utilise une adresse **IP** statique :

```
DEVICE=br0
TYPE=Bridge
IPADDR=192.168.1.1
PREFIX=24
BOOTPROTO=none
ONBOOT=yes
DELAY=0
```

Pour compléter la création du pont, on crée une autre interface, ou on modifie une interface existante, et on la pointe vers l'interface de pontage.

Exemple 6.2. Exemple de fichier de configuration d'interface ifcfg-ethX

L'exemple suivant est un exemple de fichier de configuration d'interface Ethernet pointant vers une interface de pontage. Configurer l'interface parente dans **`/etc/sysconfig/network-scripts/ifcfg-ethX`**, avec *X* comme nombre unique correspondant à une interface particulière, comme suit :

```
DEVICE=ethX
TYPE=Ethernet
HWADDR=AA:BB:CC:DD:EE:FF
BOOTPROTO=none
ONBOOT=yes
BRIDGE=br0
```

Vous pouvez spécifier un nom en utilisant la directive NAME. Si aucun nom n'est spécifié, le plug-in du **NetworkManager**, **`ifcfg-rh`**, créera un nom pour le profil de connexion sous la forme d'une « Interface Type ». Dans cet exemple, cela signifie que le pont s'appellera **`Bridge br0`**. Alternativement, si **`NAME=bridge-br0`** est ajouté au fichier **`ifcfg-br0`**, le profil de connexion sera nommé **`bridge-br0`**.

NOTE

Pour la directive **`DEVICE`**, presque n'importe quel nom d'interface pourrait servir comme il ne détermine pas le type de périphérique. **`TYPE=Ethernet`** n'est pas strictement nécessaire. Si la directive **`TYPE`** n'est pas définie, le périphérique est considéré comme un périphérique Ethernet (à moins que son nom corresponde explicitement à un fichier de configuration d'interface différente).

Les directives sont sensibles à la casse.

Spécifier le matériel ou l'adresse MAC en utilisant la directive **`HWADDR`** peut influencer le procédure d'affectation de noms, comme expliqué dans Chapitre 8, *Nommage de périphériques réseaux consistante*.

AVERTISSEMENT

Si vous configurez le pontage d'un hôte distant et que vous êtes connecté à cet hôte sur la carte réseau physique que vous configurez, veuillez tenir compte des conséquences de perte de connectivité avant de continuer. Vous perdrez en connectivité lorsque vous redémarrez le service et vous risquez de ne pas être en mesure de regagner en connectivité si des erreurs ont été faites. Un accès par console ou out-of-band est conseillé.

Pour appeler les interfaces nouvelles ou récemment configurées, exécutez une commande du format suivant en tant qu'utilisateur **root** :

```
ifup device
```

Cette commande détectera si le **NetworkManager** est en cours d'exécution et appelera **nmcli con load *UUID***, puis **nmcli con up *UUID***.

Sinon, pour télécharger à nouveau toutes les interfaces, exécutez la commande en tant qu'utilisateur **root** :

```
~]# systemctl restart network
```

Cette commande stopppera le service réseau, démarrera le service réseau, et invoquera **ifup** pour tous les fichiers ifcfg ayant **ONBOOT=yes**.

NOTE

Le comportement par défaut est pour **NetworkManager** de ne pas être au courant des changements effectués sur les fichiers ifcfg et de continuer à utiliser les anciennes données de configuration jusqu'à ce que l'interface soit activée à nouveau. C'est défini par l'option **monitor-connection-files** dans le fichier **NetworkManager.conf**. Voir la page de manuel **NetworkManager.conf(5)** pour plus d'informations.

6.3.3. Pont de réseau avec liaison

En tant qu'application courante d'environnement de virtualisation, voici un exemple de pont de réseau formé par deux ou plusieurs interfaces Ethernet liées. Si vous n'êtes pas très familier avec les fichiers de configuration d'interfaces liées, voir Section 4.4.2, « Créer une interface de canal de liaison »

Créez ou modifiez deux ou plusieurs fichiers de configuration d'interface Ethernet, devant être reliés, comme suit :

```
DEVICE=ethX
TYPE=Ethernet
SLAVE=yes
MASTER=bond0
BOOTPROTO=none
HWADDR=AA:BB:CC:DD:EE:FF
```

NOTE

Utiliser **eth*X*** comme nom d'interface est pratique courante, mais presque n'importe quel nom peut être utilisé.

Créer ou modifier un fichier de configuration d'interface, **/etc/sysconfig/network-scripts/ifcfg-bond0**, comme suit :

```
DEVICE=bond0
ONBOOT=yes
BONDING_OPTS='mode=1 miimon=100'
BRIDGE=brbond0
```

Pour obtenir plus d'informations et des conseils sur la façon de configurer le module de liaison, et pour voir la liste des paramètres de liaison, consulter Section 4.5, « Utiliser une liaison de canal ».

Créer ou modifier un fichier de configuration d'interface, **/etc/sysconfig/network-scripts/ifcfg-brbond0**, comme suit :

```
DEVICE=brbond0
ONBOOT=yes
TYPE=Bridge
IPADDR=192.168.1.1
PREFIX=24
```

Nous avons maintenant deux ou plusieurs fichiers de configuration d'interface avec la directive **MASTER=bond0**. Ils pointent vers le fichier de configuration nommé **/etc/sysconfig/network-scripts/ifcfg-bond0**, qui contient la directive **DEVICE=bond0**. Ce **ifcfg-bond0** à son tour pointe vers le fichier de **/etc/sysconfig/network-scripts/ifcfg-brbond0**, qui contient l'adresse **IP** et agit comme une interface pour les réseaux virtuels à l'intérieur de l'hôte.

Pour appeler les interfaces nouvelles ou récemment configurées, exécutez une commande du format suivant en tant qu'utilisateur **root** :

```
ifup device
```

Cette commande détectera si le **NetworkManager** est en cours d'exécution et appelera **nmcli con load *UUID***, puis **nmcli con up *UUID***.

Sinon, pour télécharger à nouveau toutes les interfaces, exécutez la commande en tant qu'utilisateur **root** :

```
~]# systemctl restart network
```

Cette commande stopppera le service réseau, démarrera le service réseau, et invoquera **ifup** pour tous les fichiers ifcfg ayant **ONBOOT=yes**.

NOTE

Le comportement par défaut est pour **NetworkManager** de ne pas être au courant des changements effectués sur les fichiers ifcfg et de continuer à utiliser les anciennes données de configuration jusqu'à ce que l'interface soit activée à nouveau. C'est défini par l'option **`monitor-connection-files`** dans le fichier **`NetworkManager.conf`**. Voir la page de manuel **`NetworkManager.conf(5)`** pour plus d'informations.

6.4. CONFIGURER LE PONTAGE DE RÉSEAU PAR L'INTERFACE EN LIGNE DE COMMANDES (GUI)

Lors du démarrage d'une interface de pontage, le **NetworkManager** attend qu'un port au moins entre dans l'état « réacheminement » (forwarding) avant d'entreprendre une configuration d'**`IP`** dépendant du réseau, telle que la configuration automatique **`DHCP`** ou **`IPv6`**. Les adressages **`IP`** statiques sont autorisés avant que les esclaves ou les ports soient connectés ou commencent à réacheminer des paquets.

6.4.1. Établir une connexion de pontage

Procédure 6.1. Ajouter une nouvelle connexion de pontage

Suivre les étapes suivantes pour créer une nouvelle connexion de pontage.

1. Pour utiliser l'outil de configuration graphique de **Réseau**, appuyer sur la touche **`Super`** pour accéder au menu Activités, saisir **`control network`**, et appuyez sur la touche **`Enter`**. L'outil de configuration du **Réseau** apparaîtra. Cette étape est totalement expliquée dans Section 2.5, « Utiliser le NetworkManager avec l'interface graphique GNOME ».

2. Sélectionner le symbole plus sous le menu. La fenêtre **`Ajouter une connexion réseau`** apparaitra.

3. Sélectionner l'entrée de menu **`Pont`**. La fenêtre **`Modifier Connexion Pont`***`1`* apparaîtra.

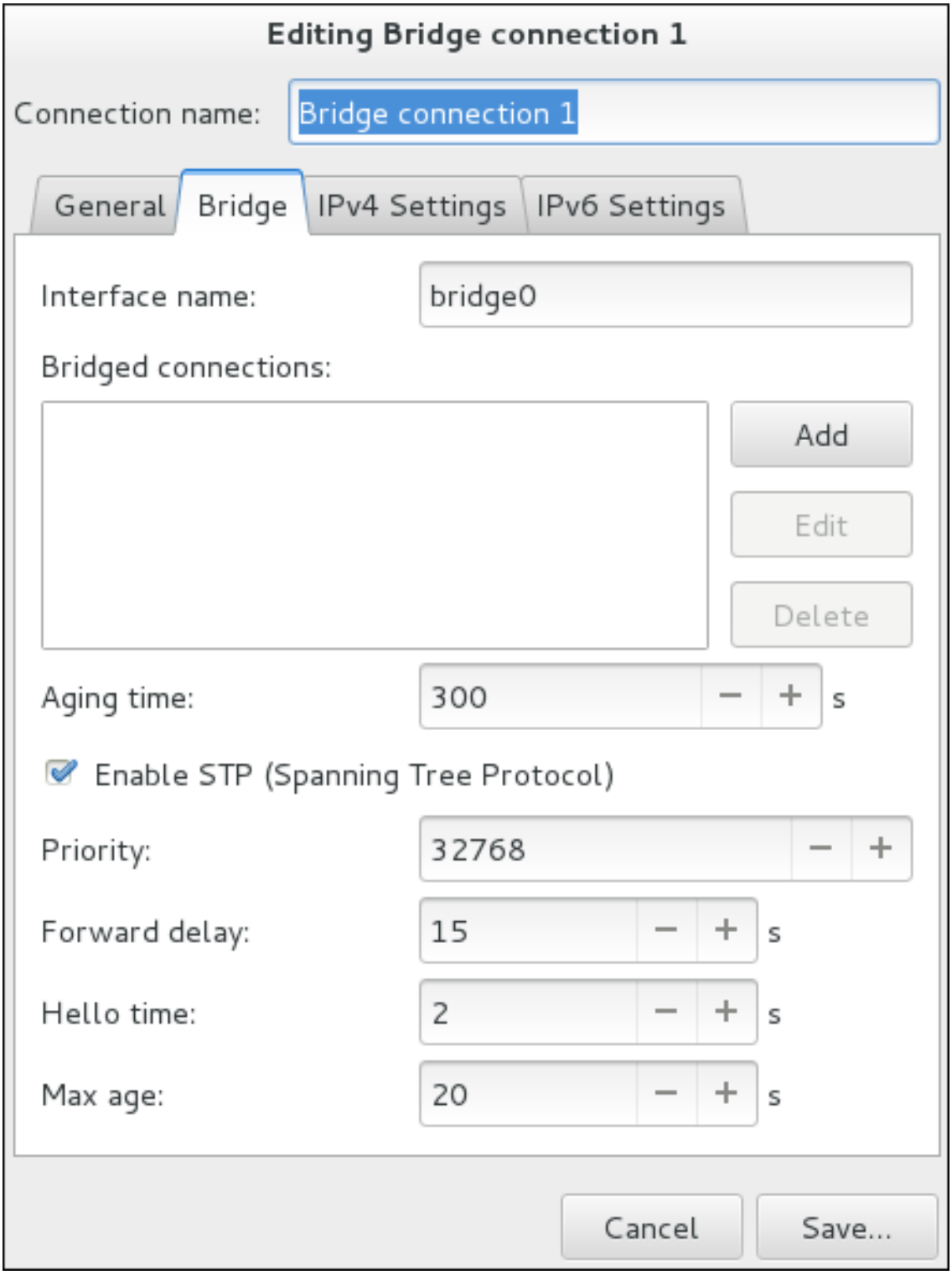


Figure 6.5. Modifier Connexion Pont 1

4. Ajouter les périphériques esclaves en vous basant sur la procédure Procédure 6.3, « Ajouter une interface esclave à un pont » ci-dessous.

Procédure 6.2. Modifier une connexion de pont existante

Vous pouvez configurer une connexion de pont existante en ouvrant la fenêtre **Réseau**, et en sélectionnant le nom de la connexion dans la liste. Puis, cliquer sur le bouton **Edit**.

1. Appuyer sur la touche **Super** pour accéder au menu Activités, saisir **control network**, et appuyez sur la touche **Entrée**. L'outil de configuration du **Réseau** apparaîtra.

2. Sélectionner la connexion **Pont** que vous souhaitez modifier dans le menu de gauche.

3. Cliquer sur le bouton **Options**.

Configurer le Nom de connexion, le Comportement Auto-Connect, et la Disponiblité
Il existe cinq configurations de la boîte de dialogue **Modifier** qui sont communes à tous les types de connexion. Voir l'onglet **Général** :

- **Nom de connexion** — saisir un nom descriptif pour votre connexion de réseau. Ce nom sera utilisé pour lister cette connexion dans le menu de la fenêtre **Réseau**.

- **Se connecter automatiquement à ce réseau quand il est disponible** — sélectionnez cette case si vous souhaitez que le **NetworkManager** s'auto-connecte à cette connexion quand elle sera disponible. Voir Section 2.5.3, « Se connecter à un réseau automatiquement » pour plus d'informations.

- **Rendre le réseau disponible à tous les utilisateurs** — sélectionnez cette case pour créer une connexion disponible à tous les utilisateurs sur le système. Changer ce paramètre peut nécessiter des privilèges d'utilisateur root. Consulter Section 2.5.4, « Profils de connexions privées ou sur tout le système » pour obtenir plus d'informations.

- **Se connecter automatiquement au VPN quand on utilise cette connexion** — sélectionnez cette case si vous souhaitez que le **NetworkManager** s'auto-connecte à une connexion de VPN quand il est disponible. Sélectionner le VPN à partir du menu déroulant.

- **Zone de parefeu** — sélectionnez une zone de parefeu dans le menu déroulant. Voir le guide *Red Hat Enterprise Linux 7 Security Guide* pour obtenir plus d'informations sur les Zones de parefeux.

6.4.1.1. Configurer l'onglet Pont

Nom d'interface

Le nom de l'interface qui va vers le pont

Connexions avec pont

Une ou plusieurs interfaces esclaves.

Durée de vieillissement autorisée

La durée, en secondes, pendant laquelle une adresse MAC est conservée dans la base de données des adresses MAC de transfert.

Activer STP (Spanning Tree Protocol)

Si requis, cochez la case pour activer **STP**.

Priorité

La priorité du pont ; le pont avec la priorité la moins élevée sera considéré comme le pont racine.

Durée de réacheminement

La durée, en secondes, passée à la fois dans les états d'Écoute (Listening) et d'Apprentissage (Learning), avant d'entrer dans l'état de réacheminement (Forwarding). La valeur par défaut est de 15 secondes.

`Hello time`

La durée, en secondes, entre les envois d'information de configuration en BPDU (Bridge Protocol Data Units).

`Age Max`

La durée maximum, en secondes, pendant laquelle stocker les informations de configuration des BPDU. Cette valeur doit correspondre à deux fois la valeur du Hello Time plus 1, mais doit être inférieure à deux fois la valeur de la Durée de réacheminement moins 1.

Procédure 6.3. Ajouter une interface esclave à un pont

1. Pour ajouter un port à un pont, sélectionner l'onglet **`Pont`** dans la fenêtre **`Modifier Connexion Pont1`**. Si nécessaire, ouvrir cette fenêtre en suivant la procédure suivante Procédure 6.2, « Modifier une connexion de pont existante ».

2. Cliquer sur **`Ajouter`**. Le menu **`Sélectionner un type de connexion`** apparaîtra.

3. Sélectionner le type de connexion à créer dans la liste. Cliquer sur **`Créer`**. Une fenêtre correspondant au type de connexion sélectionné apparaîtra.

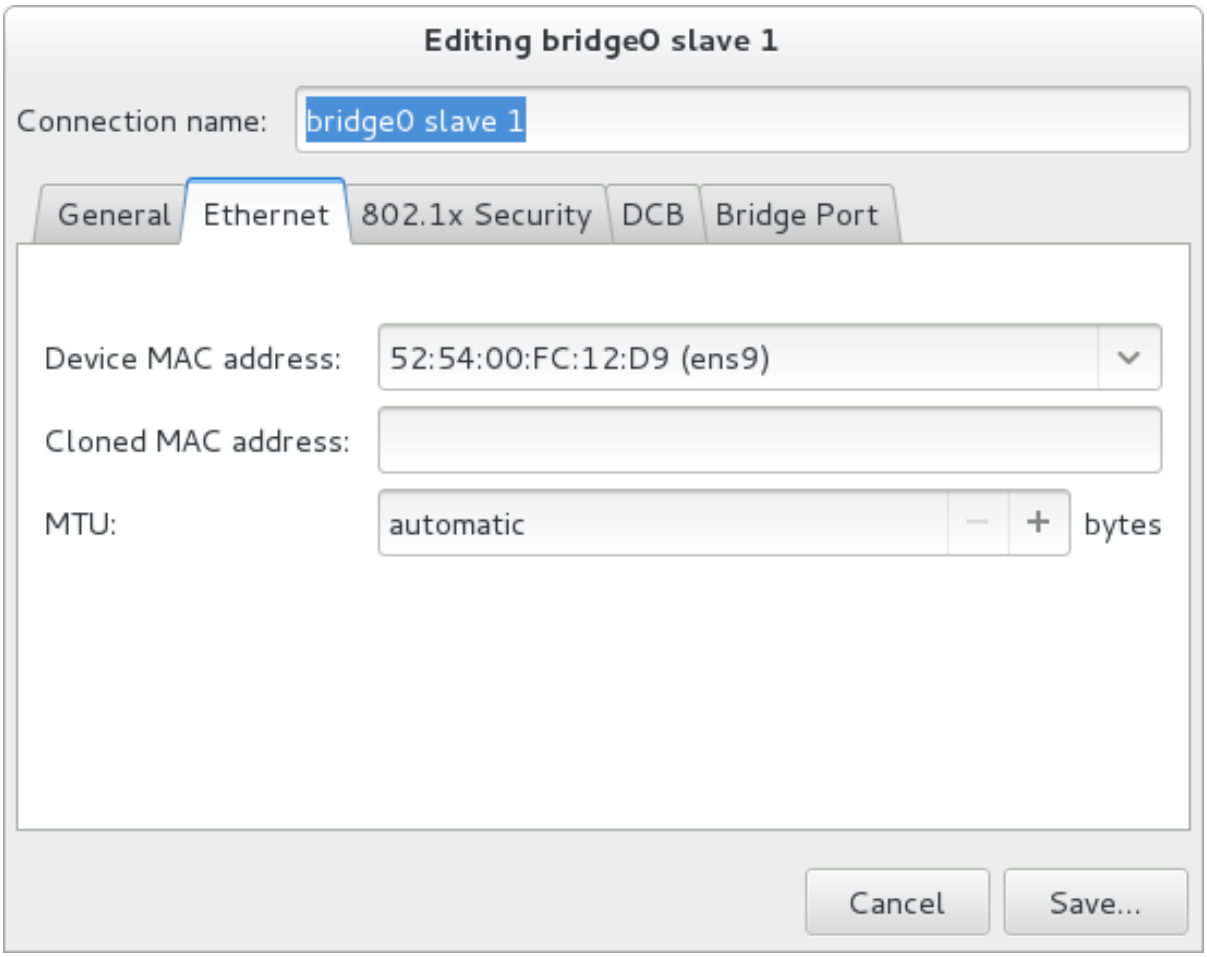


Figure 6.6. Pour que l'interface utilisateur graphique du NetworkManager ajoute un menu de connexion de pont.

4. Sélectionner l'onglet **`Port de pont`**. Configurer **`Priorité`** et **`Coût du chemin`** selon les besoins. Notez que la priorité STP d'un port de pont est limitée par le noyau Linux. Malgré que le standard permette une plage de valeurs allant de **`0`** à **`255`**, Linux ne permet que de **`0`** à **`63`**. La

valeur par défaut est **32** dans ce cas.

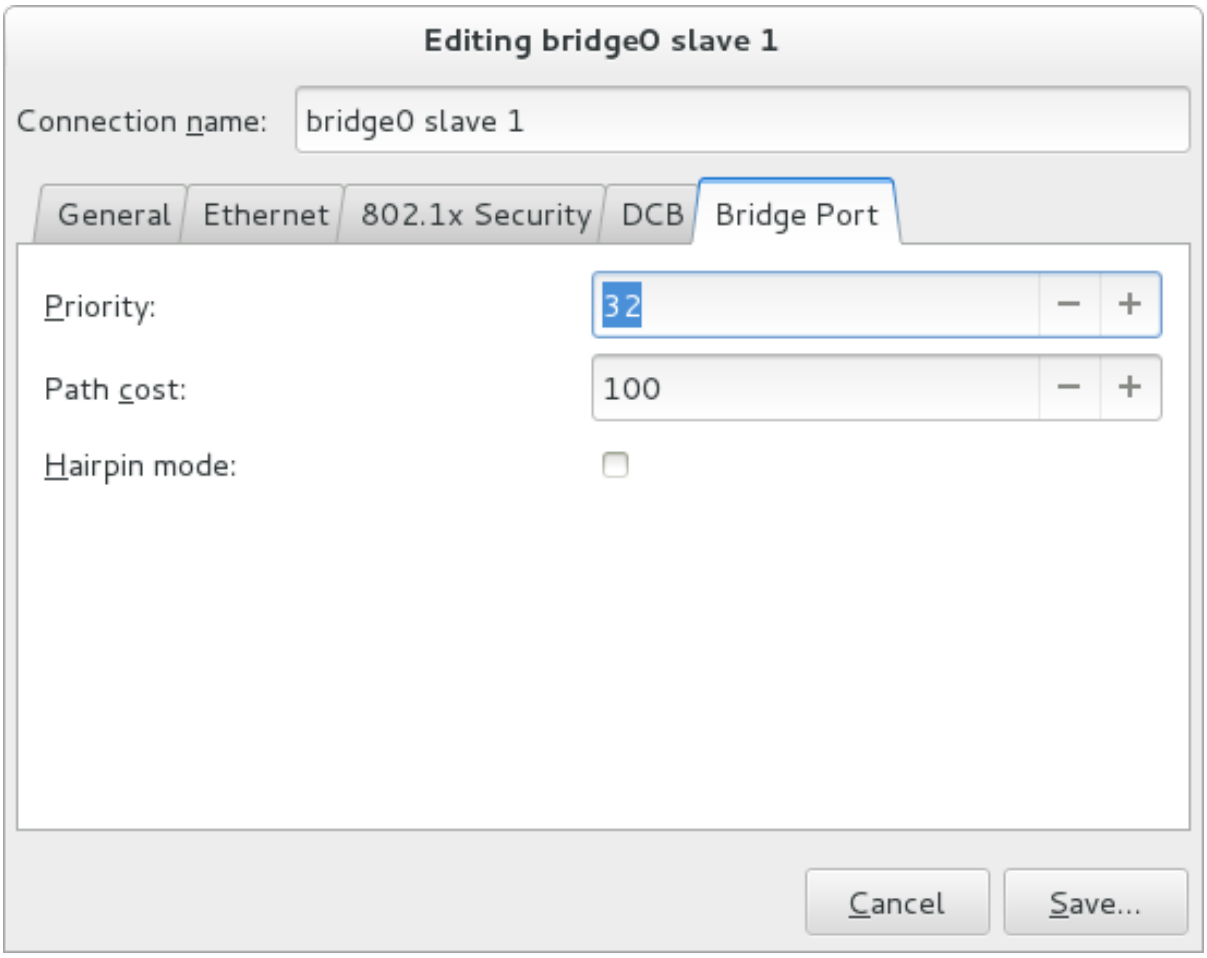


Figure 6.7. L'onglet de port de pont de l'Interface utilisateur graphique du NetworkManager

5. Si besoin est, sélectionner la case **`Mode Hairpin`** pour activer le réacheminement d'images à être traitées en externe. Connu également sous le nom de mode VEPA, de l'anglais *virtual Ethernet port aggregator* (VEPA).

Puis, pour configurer :

- Une esclave Ethernet, cliquer sur l'onglet **`Ethernet`**, puis sur Section 2.5.5.1, « Configurer le Nom de connexion, le Comportement Auto-Connect, et la Disponiblité », ou;
- Une liaison esclave, cliquer sur l'onglet **`Liaison`**, puis Section 4.6.1.1, « Configurer l'onglet Liaisons », ou ;
- Une liaison esclave, cliquer sur l'onglet **`Liaison`**, puis Section 5.13.1.1, « Configurer l'onglet Team », ou ;
- Un VLAN esclave, cliquer sur l'onglet **`VLAN`**, puis Section 7.5.1.1, « Configurer l'onglet VLAN », ou ;

Sauvegarder votre nouvelle connexion (ou votre connexion modifiée) et faire des configurations supplémentaires
Une fois vous aurez terminé de modifier votre connexion de pontage au réseau local virtuel, cliquez sur le bouton **Save** pour enregistrer votre configuration personnalisée. Si le profil est en cours d'utilisation lors de la modification, alimentez le cycle de connexion pour que **NetworkManager** applique les

modifications. Si le profil est désactivé (OFF), réglez-le sur ON ou sélectionnez-le dans le menu de l'icône de connexion réseau. Voir Section 2.5.1, « Se connecter à réseau par un GUI » pour plus d'informations sur l'utilisation de votre connexion nouvelle ou modifiée.

Vous pouvez configurer davantage une connexion existante en la sélectionnant dans la fenêtre **Réseau** et en cliquant sur **Options** pour revenir à la boîte de dialogue **Modifier**.

Puis, pour configurer :

- Paramètres de configuration **IPv4** pour la connection, cliquer sur l'onglet **IPv4 Settings** et continuer avec Section 2.5.10.4, « Configuration des paramètres IPv4 », ou ;
- Paramètres de configuration **IPv6** pour la connexion, cliquer sur l'onglet **IPv6 Settings**et continuez avec Section 2.5.10.5, « Configurer les paramètres IPv6 ».

Après la sauvegarde, le pont apparaîtra dans l'outil de configuration du réseau, avec chaque esclave affiché.

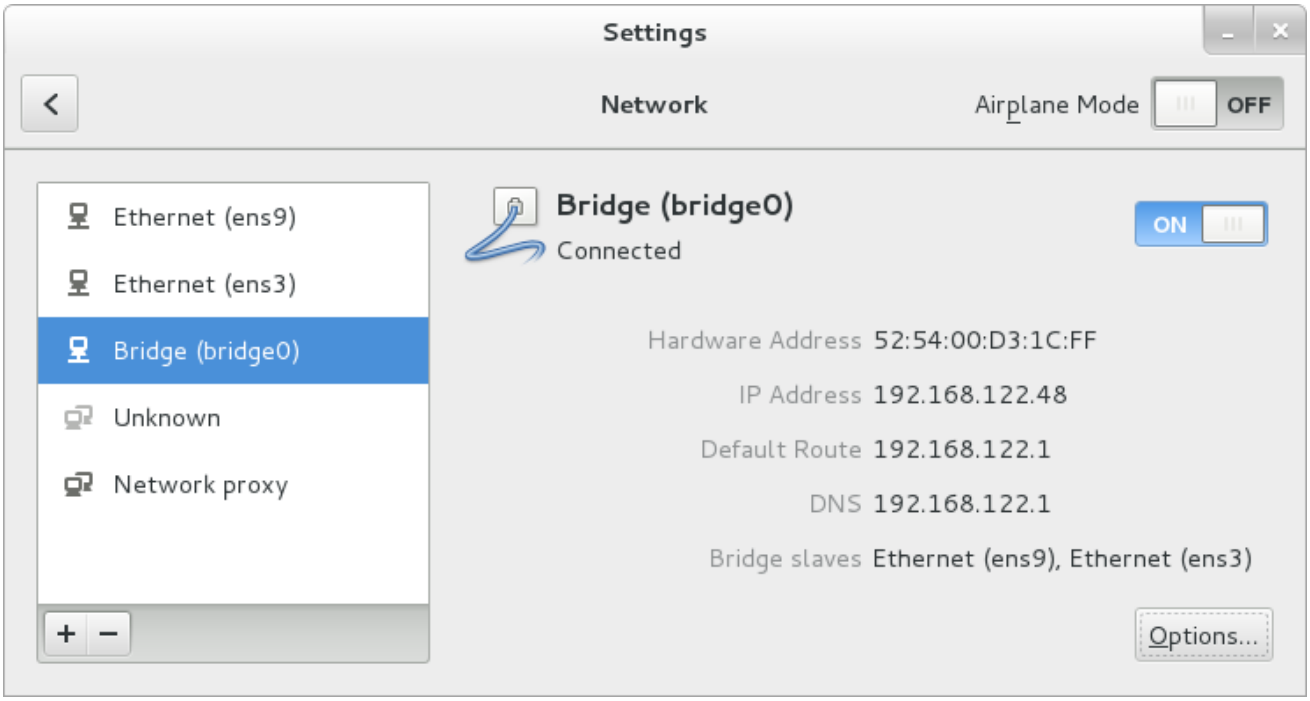


Figure 6.8. Interface utilisateur graphique du NetworkManager avec Pontage

6.5. RESSOURCES SUPPLÉMENTAIRES

Les sources d'informations suivantes fournissent des ressources supplémentaires à propos des pontages de réseaux.

6.5.1. Documentation installée

- Page man **nmcli(1)** — décrit l'outil de ligne de commandes du **NetworkManager**.
- Page man **nmcli-examples(5)** — donne des exemples des commandes **nmcli**.
- Page man **nm-settings(5)** — décrit les configurations et les paramètres des connexions du **NetworkManager**.

CHAPITRE 7. CONFIGURATION DU BALISAGE D'UN RÉSEAU VIRTUEL VLAN 802.1Q

Pour créer un réseau local virtuel, on crée une interface sur une autre interface dénommée l'*interface parente*. L'interface VLAN va baliser les paquets avec l'ID VLAN puisqu'ils passent par le biais de l'interface, et les paquets retour seront dé-balisés. L'interface VLAN peut être configurée de la même façon que n'importe quelle autre interface. L'interface parente ne doit pas être une interface Ethernet. Une interface de balisage VLAN 802. 1 q peut être créée sur un pontage, une liaison ou des interfaces d'équipe, mais il faut noter certans points :

- Pour les réseaux VLAN sur liaisons, il est important que la liaison ait des esclaves et qu'ils soient « activés » avant d'activer l'interface VLAN. Au moment où l'on écrit ce document, ajouter une interface VLAN à une liaison sans esclave ne marche pas.

- Impossible de configurer un esclave VLAN sur une liaison avec l'option **`fail_over_mac=follow`**, parce que le périphérique virtuel VLAN ne peut pas changer son adresse MAC pour qu'elle correspondre à la nouvelle adresse MAC. Dans un tel cas, le trafic est toujours envoyé avec l'adresse MAC source maintenant incorrect.

- L'envoi de paquets VLAN balisés par un commutateur réseau nécessite la configuration du commutateur. Reportez-vous à la documentation à propos du commutateur. Par exemple, pour les commutateurs Cisco, le port doit être assigné à un VLAN ou configuré pour être un port trunk pour accepter les paquets balisés pour plusieurs réseaux locaux virtuels. Les paquets non balisés peuvent également être traités par un port trunk et considérés comme appartenant au *VLAN natif*, mais c'est un risque de sécurité et cela a pu être désactivé, ou bien ne pas être activé par défaut, selon la marque du commutateur.

- Certaines anciennes cartes d'interface réseau, les interfaces loopback, les cartes Wimax et certains périphériques InfiniBand, sont censés être *VLAN challenged*, ce qui signifie qu'ils ne supportent pas les VLAN. C'est généralement parce que ces périphériques ne peuvent pas comprendre les en-têtes VLAN et la grande taille de l'unité MTU associée aux paquets balisés.

7.1. SÉLECTION DES MÉTHODES DE CONFIGURATION D'INTERFACE

- **Pour configurer une interface VLAN avec l'outil d'interface d'utilisateur texte du NetworkManager**, **nmtui**, consulter Section 7.2, « Configurer le balisage 802.1Q VLAN à l'aide de l'interface utilisateur texte, nmtui »

- **Pour configurer une interface VLAN avec l'outil en ligne de commandes du NetworkManager**, **nmtui**, consulter Section 7.3, « Pour configurer le balisage 802.1Q VLAN à l'aide de l'outil de ligne de commandes, nmcli »

- **Pour configurer une interface de réseau manuellement**, voir Section 7.4, « Pour configurer le balisage du réseau VLAN 802.1Q à l'aide de l'outil de ligne de commandes ».

- **Pour configurer un réseau à l'aide d'outils d'interface utilisateur graphiques** , voir Section 7.5, « Pour configurer le balisage du réseau VLAN 802.1Q à l'aide de l'interface graphique »

7.2. CONFIGURER LE BALISAGE 802.1Q VLAN À L'AIDE DE L'INTERFACE UTILISATEUR TEXTE, NMTUI

L'outil d'interface utilisateur de texte **nmtui** peut être utilisée pour configurer les 802.1Q VLAN dans une fenêtre de terminal. Veuillez exécuter la commande suivante pour démarrer l'outil :

```
~]$ nmtui
```

L'interface utilisateur texte apparaîtra. Tout commande non valide affichera un message d'utilisation.

Pour naviguer, utiliser les flèches ou appuyer sur **Tab** pour continuer et appuyer sur la combinaision de touches **Maj**+**Tab** pour revenir aux options. Appuyer sur la touche **Entrée** pour sélectionner une option. La barre **Espace** active/désactive le statut d'un case à cocher.

À partir du menu de démarrage, sélectionner **Modifier une connexion**. Sélectionner **Ajouter**, l'écran **Nouvelle connexion** apparaîtra.

Figure 7.1. Pour que l'interface utilisateur texte du NetworkManager ajoute un menu de connexion VLAN

Sélectionner **VLAN**, l'écran **Modifier connexion** apparaîtra. Suivre les invites de l'écran pour terminer la configuration.

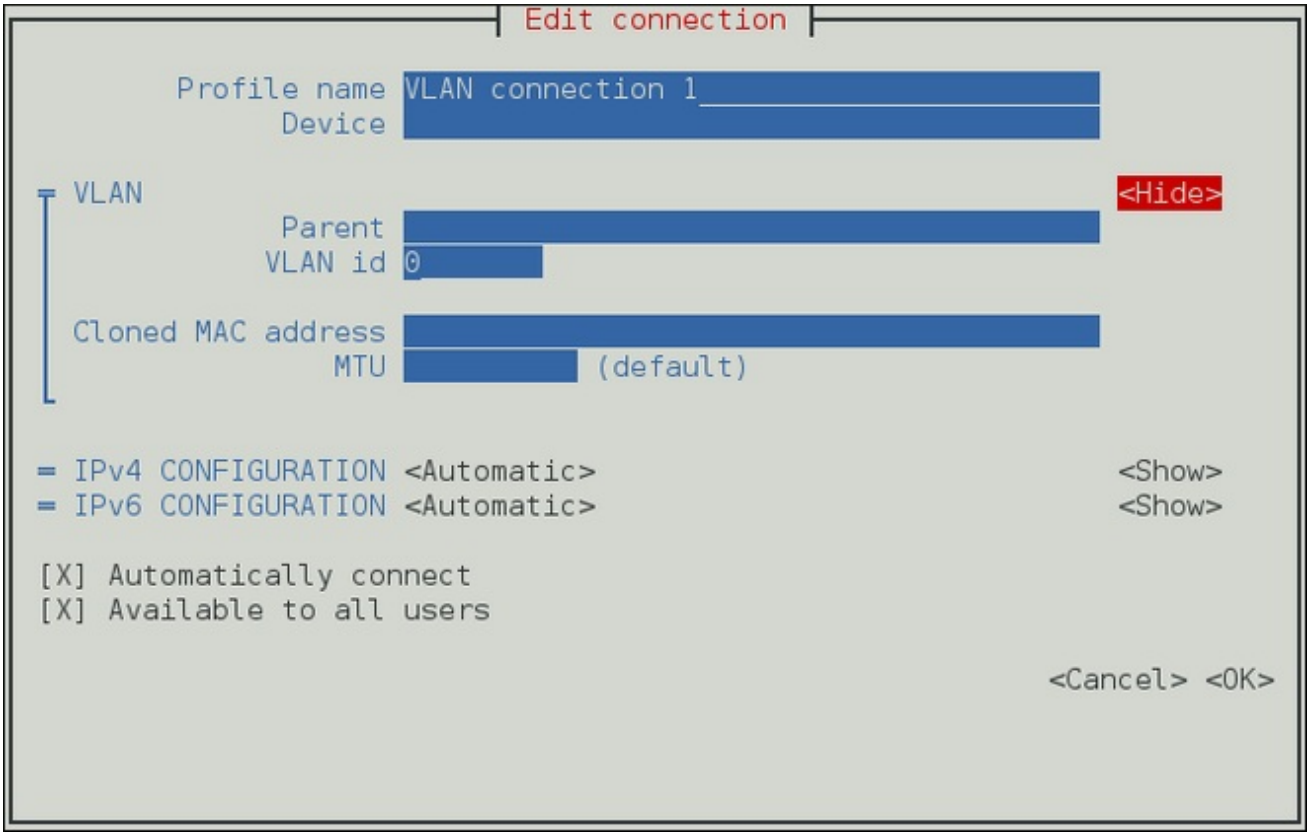


Figure 7.2. Pour que l'interface utilisateur texte du NetworkManager configure un menu de connexion VLAN

Voir Section 7.5.1.1, « Configurer l'onglet VLAN » pour obtenir des définitions des termes VLAN.

Voir Section 1.5, « Configuration réseau utilisant une interface utilisateur texte (nmtui) » pour obtenir des informations sur la façon d'installer **nmtui**.

7.3. POUR CONFIGURER LE BALISAGE 802.1Q VLAN À L'AIDE DE L'OUTIL DE LIGNE DE COMMANDES, NMCLI

Pour afficher les interfaces disponibles sur le système, veuillez exécuter une commande comme suit :

```
~]$ nmcli con show
NAME         UUID                                   TYPE
DEVICE
System eth1  9c92fad9-6ecb-3e6c-eb4d-8a47c6f50c04  802-3-ethernet  eth1
System eth0  5fb06bd0-0bb0-7ffb-45f1-d6edd65f3e03  802-3-ethernet  eth0
```

Notez que le champ de nom NAME dans la sortie indique toujours l'ID de connexion. Il ne correspond pas au nom de l'interface même s'il en a le même aspect. L'ID peut être utilisé avec les commandes **`nmcli connection`** pour identifier une connexion. Utilisez le nom du périphérique avec d'autres applications comme **`firewalld`**.

Pour créer une interface 802. 1 q VLAN sur interface Ethernet *eth0*, avec l'interface VLAN *VLAN10* et l'ID **`10`**, exécutez une commande comme suit :

```
~]$ nmcli con add type vlan ifname VLAN10 dev eth0 id 10
Connection 'vlan-VLAN10' (37750b4a-8ef5-40e6-be9b-4fb21a4b6d17)
successfully added.
```

Notez que comme aucun **con-name** n'a été donné à l'interface VLAN, le nom est dérivé d'un nom d'interface composé à l'aide du type. Autre méthode: spécifier un nom par l'option **con-name** comme suit :

```
~]$ nmcli con add type vlan con-name VLAN12 dev eth0 id 12
Connection 'VLAN12' (b796c16a-9f5f-441c-835c-f594d40e6533) successfully
added.
```

Assigner des adresses aux interfaces VLAN

Vous pouvez utiliser les mêmes commandes **nmcli** pour assigner des adresses statiques ou dynamiques comme pour toute autre interface.

Par exemple, voici une commande pour créer une interface VLAN avec une adresse **IPv4** statique et une passerelle :

```
~]$ nmcli con add type vlan con-name VLAN20 dev eth0 id 20 ip4
10.10.10.10/24 \
gw4 10.10.10.254
```

Pour créer une interface VLAN avec une adresse assignée de façon dynamique, veuillez exécuter la commande suivante :

```
~]$ nmcli con add type vlan con-name VLAN30 dev eth0 id 30
```

Voir Section 2.3.2, « Se connecter à réseau par nmcli » pour obtenir des exemples sur la façon d'utiliser les commandes **nmcli** de configuration d'interfaces.

Pour réviser les interfaces VLAN créées, veuillez exécuter une commande du style :

```
~]$ nmcli con show
NAME         UUID                                   TYPE
DEVICE
VLAN12       4129a37d-4feb-4be5-ac17-14a193821755   vlan
eth0.12
System eth1  9c92fad9-6ecb-3e6c-eb4d-8a47c6f50c04   802-3-ethernet  eth1
System eth0  5fb06bd0-0bb0-7ffb-45f1-d6edd65f3e03   802-3-ethernet  eth0
vlan-VLAN10  1be91581-11c2-461a-b40d-893d42fed4f4   vlan            VLAN10
```

Pour vérifier les informations détaillées sur la connexion configurée, veuillez exécuter la commande suivante :

```
~]$ nmcli -p con show VLAN12
===============================================================================
=====
                     Connection profile details (VLAN12)
===============================================================================
=====
connection.id:                          VLAN12
connection.uuid:                        4129a37d-4feb-4be5-ac17-
14a193821755
connection.interface-name:              --
connection.type:                        vlan
connection.autoconnect:                 yes…
-------------------------------------------------------------------------------
```

```
---------
802-3-ethernet.port:                    --
802-3-ethernet.speed:                   0
802-3-ethernet.duplex:                  --
802-3-ethernet.auto-negotiate:          yes
802-3-ethernet.mac-address:             --
802-3-ethernet.cloned-mac-address:      --
802-3-ethernet.mac-address-blacklist:
802-3-ethernet.mtu:                     auto…
vlan.interface-name:                    --
vlan.parent:                            eth0
vlan.id:                                12
vlan.flags:                             0 (NONE)
vlan.ingress-priority-map:
vlan.egress-priority-map:
-------------------------------------------------------------------------
---------
=========================================================================
=====
      Activate connection details (4129a37d-4feb-4be5-ac17-14a193821755)
=========================================================================
=====
GENERAL.NAME:                           VLAN12
GENERAL.UUID:                           4129a37d-4feb-4be5-ac17-
14a193821755
GENERAL.DEVICES:                        eth0.12
GENERAL.STATE:                          activating[output truncated]
```

Vous trouverez d'autres options pour la commande VLAN dans la section réseau local virtuel de la page man **nmcli(1)**. Dans les pages man, le périphérique sur lequel le réseau local virtuel est créé est dénommé le périphérique parent. Dans l'exemple ci-dessus, le périphérique a été spécifié par son nom d'interface, **eth0**, il peut également être spécifié par la connexion UUID ou l'adresse MAC.

Pour créer un profil de connexion 802.1Q VLAN avec un mappage prioritaire ingress sur interface Ethernet *eth1*, ayant pour nom VLAN1 et l'ID **13**, émettez une commande comme suit :

```
~]$ nmcli con add type vlan con-name VLAN1 dev eth2 id 13 ingress
"2:3,3:5"
```

Pour afficher tous les paramètres associés au VLAN créé ci-dessus, veuillez exécuter une commande comme suit :

```
~]$ nmcli connection show vlan-VLAN10
```

Pour apporter des modifications à l'unité MTU, veuillez exécuter la commande suivante :

```
~]$ nmcli connection modify vlan-VLAN10 802.mtu 1496
```

La configuration MTU détermine la taille maximale du paquet de couche de réseau. La taille maximale de la charge utile que le support de couche liaison peut porter à son tour limite la couche réseau MTU. Pour les trames Ethernet standard, cela signifie un MTU de 1500 octets. Il ne devrait pas être nécessaire de changer la valeur MTU lorsque vous configurez un réseau local virtuel quand l'en-tête de couche de liaison augmente en taille de 4 octets pour accueillir la balise 802.1Q.

Au moment de la rédaction, **vlan.interface-** et **connection.interface-name** doivent être les

mêmes (s'ils sont définis). Ils doivent donc être changés en même temps à l'aide de **nmcli**, en mode interactif. Pour modifier un nom de connexion de réseau local virtuel, exécuter les commandes suivantes :

```
~]$ nmcli con edit vlan-VLAN10
nmcli> set vlan.interface-name superVLAN
nmcli> set connection.interface-name superVLAN
nmcli> save
nmcli> quit
```

L'utilitaire **nmcli** peut être utilisé pour définir et effacer les marqueurs **ioctl** qui influencent la façon dont le code 802.1Q fonctionne. Les marqueurs VLAN sont pris en charge par le **NetworkManager**:

- 0x01 - réordonnancement des en-têtes de packets sortants
- 0x02 - utilisation du protocole GVRP
- 0x04 - liaison lâche de l'interface et de son master

L'état du VLAN est synchronisé à l'état de l'interface parente ou maître (l'interface ou le périphérique sur lequel le réseau local virtuel est créé). Si l'interface parente est définie à l'état admin « down », alors tous les VLAN associés sont définis à «down» et tous les itinéraires seront vidés de la table de routage. Le marqueur **0 x 04** active le mode *loose binding*, dans lequel seul l'état opérationnel est passé du parent aux VLAN associé, mais l'état du périphérique VLAN demeure inchangé.

Pour définir un marqueur VLAN

```
~]$ nmcli connection modify vlan-VLAN10 vlan.flags 1
```

Voir Section 2.3, « Utiliser l'outil de ligne de commandes du NetworkManager, nmcli » pour une introduction à **nmcli**.

7.4. POUR CONFIGURER LE BALISAGE DU RÉSEAU VLAN 802.1Q À L'AIDE DE L'OUTIL DE LIGNE DE COMMANDES

Dans Red Hat Enterprise Linux 7, le module **8021q** est téléchargé par défaut. Si nécessaire, vous pouvez vérifier que le module est bien chargé en exécutant la commande suivante en tant qu'utilisateur **root** :

```
~]# modprobe --first-time 8021q
modprobe: ERROR: could not insert '8021q': Module already in kernel
```

Pour afficher des informations sur le module, exécutez la commande suivante :

```
~]$ modinfo 8021q
```

Consulter la page man **modprobe(8)** pour plus d'options de commandes.

7.4.1. Pour configurer le balisage du réseau VLAN 802.1Q à l'aide des fichiers ifcfg

1. Configurer l'interface parente dans **/etc/sysconfig/network-scripts/ifcfg-ethX**, avec *X* comme nombre unique correspondant à une interface particulière, comme suit :

```
DEVICE=ethX
TYPE=Ethernet
BOOTPROTO=none
ONBOOT=yes
```

2. Configurer l'interface de VLAN dans le répertoire **/etc/sysconfig/network-scripts/**. Le nom du fichier de configuration doit se trouver dans l'interface parente avec un signe **.** en plus du numéro d'ID du VLAN. Ainsi, si l'ID du VLAN est de 192, et que l'interface parente correspond à *eth0*, alors le nom du fichier de configuration doit être **ifcfg-eth0.192** :

```
DEVICE=ethX.192
BOOTPROTO=none
ONBOOT=yes
IPADDR=192.168.1.1
PREFIX=24
NETWORK=192.168.1.0
VLAN=yes
```

   Si vous avez besoin de configurer un second VLAN, avec par exemple, ID VLAN 193, sur la même interface, *eth0*, ajouter une nouveau fichier ayant pour nom **eth0.193** avec les détails de configuration du VLAN.

3. Démarrer à nouveau le service de réseautage pour que les changements puissent prendre effet. Exécuter la comande suivante en tant qu'utilisateur **root** :

```
~]# systemctl restart network
```

7.4.2. Pour configurer le balisage du réseau VLAN 802.1Q à l'aide des commandes ip

Pour créer une interface de réseau VLAN 802. 1 Q sur interface Ethernet *eth0*, avec l'interface *VLAN8* et l'ID **8**, émettez une commande comme suit, en tant qu'utilisteur **root** :

```
~]# ip link add link eth0 name eth0.8 type vlan id 8
```

Pour voir le VLAN, exécutez la commande suivante :

```
~]$ ip -d link show eth0.8
4: eth0.8@eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue
state UP mode DEFAULT
     link/ether 52:54:00:ce:5f:6c brd ff:ff:ff:ff:ff:ff promiscuity 0
     vlan protocol 802.1Q id 8 <REORDER_HDR>
```

Notez que l'utilitaire **ip** interprète l'ID du VLAN sous forme hexadécimale, si précédé par **0x**, et comme une valeur octale, s'il y a un **0** pour commencer. Cela signifie qu'afin d'attribuer un ID de VLAN avec une valeur décimale de **22**, vous ne devez pas ajouter de zéros.

Pour supprimer le VLAN, exécutez la commande suivante en tant qu'utilisateur **root** :

```
~]# ip link delete eth0.8
```

NOTE

Les interfaced VLAN créées en utilisant les commandes **ip** à l'invite de commande seront perdues si le système est fermé ou démarré à nouveau. Pour configurer les interfaces VLAN pour qu'elles soient persistantes après un redémarrage du système, utiliser les fichiers **ifcfg**. Voir Section 7.4.1, « Pour configurer le balisage du réseau VLAN 802.1Q à l'aide des fichiers ifcfg »

7.5. POUR CONFIGURER LE BALISAGE DU RÉSEAU VLAN 802.1Q À L'AIDE DE L'INTERFACE GRAPHIQUE

7.5.1. Établir une connexion VLAN

Vous pouvez utiliser **control-center** de GNOME pour demander à **NetworkManager** de créer un réseau local virtuel à l'aide d'une interface existante comme interface parente. Au moment de la rédaction, vous ne pouvez créer des VLAN que sur les périphériques Ethernet. Notez que les périphériques VLAN ne sont créés automatiquement que si l'interface parente est définie pour se connecter automatiquement.

Procédure 7.1. Ajouter une nouvelle connexion VLAN

Vous pouvez configurer une connexion VLAN en ouvrant la fenêtre **Réseau**, en cliquant le signe plus, et en sélectionnant le **VLAN** de la liste.

1. Appuyer sur la clé **Super** pour accéder au menu Activités, saisir **control network**, et appuyez sur la touche **Enter**. L'outil de configuration du **Réseau** apparaîtra.
2. Cliquer sur le signe plus pour ouvrir la liste de sélection. Sélectionner **VLAN**. La fenêtre **Modifier Connexion VLAN1** apparaîtra.
3. Dans l'onglet **VLAN**, sélectionner l'interface parente de la liste déroulante que vous souhaitez utiliser pour la connexion VLAN.
4. Saisir l'ID VLAN
5. Saisir un nom d'interface VLAN. C'est le nom de l'interface VLAN qui sera créée. Par exemple, **eth0.1** ou **vlan2**. (Normalement, c'est soit le nom de l'interface parente plus un point « **.** » ou d'ID du VLAN, ou « **vlan** » plus l'ID du VLAN.)
6. Vérifier et confirmer les paramètres de configuration, puis cliquer sur le bouton **Sauvegarder**.
7. Pour modifier les configurations VLAN, consulter Section 7.5.1.1, « Configurer l'onglet VLAN ».

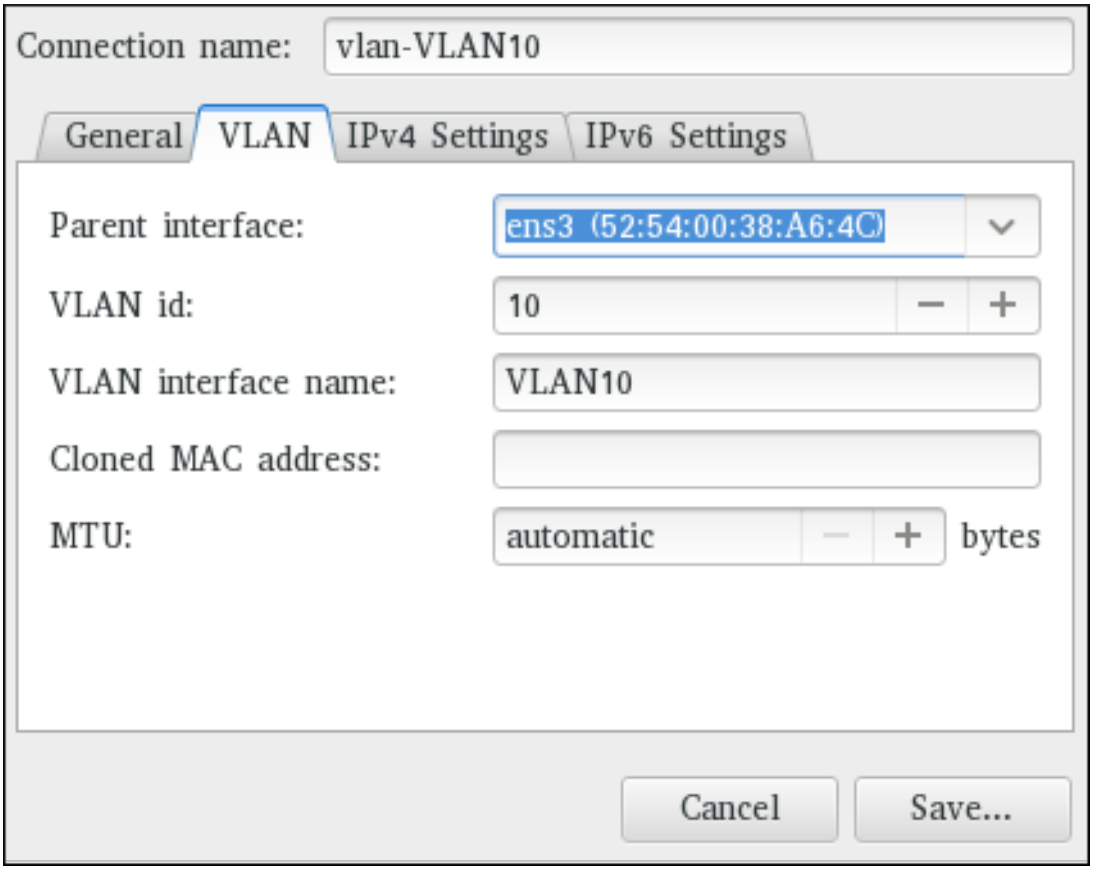


Figure 7.3. Ajouter une nouvelle connexion VLAN

Procédure 7.2. Modifier une connexion VLAN existante

Suivre ces étapes pour modifier une connexion VLAN existante.

1. Appuyer sur la clé **Super** pour accéder au menu Activités, saisir **control network**, et appuyez sur la touche **Enter**. L'outil de configuration du **Réseau** apparaîtra.

2. Sélectionner la connexion à modifier et cliquer sur le bouton **Options**.

3. Sélectionner l'onglet **Général**.

4. Configurer le nom de la connexion, le comportement auto-connect, et les paramètres disponibles.

   Certains paramètres de la boîte de dialogue **Modifier** sont communs à tous les types de connexion :

   - **Nom de connexion** — saisir un nom descriptif pour votre connexion de réseau. Ce nom sera utilisé pour lister cette connexion dans la section **VLAN** de la fenêtre **Réseau**.

   - **Se connecter automatiquement à ce réseau quand il est disponible** — sélectionnez cette case si vous souhaitez que le **NetworkManager** s'auto-connecte à cette connexion quand elle est disponible. Voir Section 2.5.3, « Se connecter à un réseau automatiquement » pour plus d'informations.

   - **Disponible à tous les utilisateurs** — sélectionnez cette case pour créer une connexion disponible à tous les utilisateurs sur le système. Changer ce paramètre peut nécessiter des privilèges d'utilisateur root. Consulter Section 2.5.4, « Profils de connexions privées ou sur tout le système » pour obtenir plus d'informations.

5. Pour modifier les configurations VLAN, consulter Section 7.5.1.1, « Configurer l'onglet VLAN ».

Sauvegarder votre nouvelle connexion (ou votre connexion modifiée) et faire des configurations supplémentaires

Une fois vous aurez terminé de modifier votre connexion VLAN au réseau local virtuel, cliquez sur le bouton **Save** pour enregistrer votre configuration personnalisée. Si le profil est en cours d'utilisation lors de la modification, alimentez le cycle de connexion pour que le **NetworkManager** applique les modifications. Si le profil est désactivé (OFF), réglez-le sur ON ou sélectionnez-le dans le menu de l'icône de connexion réseau. Voir Section 2.5.1, « Se connecter à réseau par un GUI » pour plus d'informations sur l'utilisation de votre connexion nouvelle ou modifiée.

Vous pouvez configurer davantage une connexion existante en la sélectionnant dans la fenêtre **Réseau** et en cliquant sur **Options** pour revenir à la boîte de dialogue **Modifier**.

Puis, pour configurer :

- Les configurations IPv4 pour la connection, cliquer sur l'onglet **IPv4 Settings** et continuer avec Section 2.5.10.4, « Configuration des paramètres IPv4 ».

7.5.1.1. Configurer l'onglet VLAN

Si vous avez déjà ajouté une nouvelle connexion VLAN (voir Procédure 7.1, « Ajouter une nouvelle connexion VLAN » pour obtenir des instructions), vous pouvez modifier l'onglet **VLAN** afin de définir l'interface parente et l'ID du VLAN.

Interface parente

Une interface déjà configurée peut être sélectionnée dans le menu déroulant.

ID VLAN

Le numéro d'identification à utiliser pour baliser le trafic réseau VLAN.

Nom Interface VLAN

Le nom de l'interface VLAN qui sera créée. Par exemple, **eth0.1** ou **vlan2**.

Adresse MAC clonée

En option, définit une adresse MAC différente à utiliser pour identifier l'interface VLAN. Peut être utilisé pour changer l'adresse MAC source pour des paquets envoyés sur ce VLAN.

MTU

Définit une taille de MTU (de l'anglais Maximum Transmission Unit) à utiliser pour les paquets à envoyer sur la connexion VLAN.

7.6. RESSOURCES SUPPLÉMENTAIRES

Les sources d'informations suivantes fournissent des ressources supplémentaires à propos des associations de réseaux.

7.6.1. Documentation installée

- Page man **`ip-link(8)`** — décrit les commandes de configuration de périphériques de réseau de l'utilitaire **ip**.
- Page man **`nmcli(1)`** — décrit l'outil de ligne de commandes du **NetworkManager**.
- Page man **`nmcli-examples(5)`** — donne des exemples de commandes **nmcli**.
- Page man **`nm-settings(5)`** — décrit les configurations et les paramètres des connexions du **NetworkManager**.

CHAPITRE 8. NOMMAGE DE PÉRIPHÉRIQUES RÉSEAUX CONSISTANTE

Red Hat Enterprise Linux 7 fournit des méthodes pour donner des noms de périphériques réseau de manière consistante et prévisible à des interfaces de réseau. Ces fonctionnalités changent le nom des interfaces de réseau sur un système afin de faciliter la localisation et la différentiation des interfaces.

Traditionnellement, les interfaces de réseau sous Linux sont énumérées **eth[0123…]**, mais ces noms ne correspondent pas forcément à des étiquettes sur le châssis. Les plateformes des serveurs modernes avec de multiples adaptateurs réseau peuvent rencontrer des noms d'interfaces qui ne sont pas déterminants et contre-intuitifs. Ceci affecte les adaptateurs réseau intégrés à la carte mère (*Lan-on-Motherboard*, or *LOM*) et les adaptateurs add-in (uniques et multi-ports).

Dans Red Hat Enterprise Linux 7, **udev** prend en charge un certain nombre de schémas d'affectation de noms. Le comportement par défaut est d'assigner des noms fixes basés sur le microprogramme, la topologie et les informations sur l'emplacement. Ceci a pour avantage d'offrir des noms complètement automatiques et prévisibles, qui resteront fixes, même lorsque du matériel est ajouté ou supprimé (il ne se produit pas de ré-énumération) et le matériel endommagé peut être remplacé de façon transparente. L'inconvénient de ce comportement est que les noms sont parfois plus difficiles à lire que les noms traditionnellement utilisés au préalable comme eth0 ou wlan0. Exemple : enp5s0.

8.1. SCHÉMA DE DÉNOMINATION

Par défaut, **systemd** nomme les interfaces en suivant la politique de schéma de dénomination suivante :

- **Scheme 1:** Les noms comportant les numéros d'index fournis par le microprogramme ou le BIOS, par exemple **eno1**, sont appliqués si ces informations en provenance du microprogramme ou du BIOS sont applicables et disponibles, sinon le schéma 2 est utilisé comme schéma de secours.
- **Scheme 2:** Les noms comportant les numéros d'indexes de slots de connexion à chaud de PCI Express fournis par le microprogramme ou le BIOS, par exemple **ens1**, sont appliqués si ces informations en provenance du microprogramme ou du BIOS sont applicables et disponibles, sinon le schéma 2 est utilisé comme schéma de secours.
- **Scheme 3:** Les noms comportant un emplacement physique du connecteur de matériel (exemple : **enp2s0**), sont applicables si possible, sinon le schéma 5 est utilisé comme schéma de secours dans tous les autres cas.
- **Scheme 4:** Les noms comportant des adresses MAC d'interfaces (exemple : **enx78e7d1ea46da**), ne sont pas utilisés par défaut, mais sont à la disposition de l'utilisateur s'il choisit cette possibilité.
- **Scheme 5:** Le schéma de dénomination de noyau traditionnel et imprédictible et est utilisé si toutes les autres méthodes échouent (exemple: **eth0**).

Cette politique, ou procédure décrite ci-dessus, est utilisée par défaut. Si le système a **biosdevname** activé, il sera utilisé par défaut. Notez que pour activer **biosdevname**, vous devrez passer le paramètre **biosdevname=1** en ligne de commandes, sauf en cas de système Dell, quand **biosdevname** est utilisé par défaut s'il est installé. Si l'utilisateur a jouté des règles **udev** qui changent le nom des périphériques du noyau, ces règles auront prévalence.

8.2. COMPRENDRE LA PROCÉDURE D'AFFECTATION DE NOMS AUX PÉRIPHÉRIQUES

La procédure d'affectation de noms à des périphérique est détaillée ci-dessous :

1. Une règle qui se trouve dans **`/usr/lib/udev/rules.d/60-net.rules`** instruit l'assistant d'**udev**, /**lib**/**udev**/**rename_device**, de chercher dans tous les fichiers **`/etc/sysconfig/network-scripts/ifcfg-`*`suffix`***. S'il trouve un fichier **`ifcfg`** avec une entrée **`HWADDR`** qui correspond à l'adresse d'une interface, il renomme l'interface avec le nom donné dans le fichier **`ifcfg`** par la directive du **`DEVICE`**.

2. Une règle qui se trouve dans **`/usr/lib/udev/rules.d/71-biosdevname.rules`** instruit **biosdevname** de renommer l'interface suivant la politique de nommage, dans la mesure où elle n'a pas été renommée au cours d'une étape précédente, que **biosdevname** est installé, et que **`biosdevname=0`** n'a pas reçu de commande de noyau en ligne de commande boot.

3. Une règle qui se trouve dans **`/lib/udev/rules.d/75-net-description.rules`** instruit **udev** de remplir les valeurs des propriétés du périphérique **udev** ID_NET_NAME_ONBOARD, ID_NET_NAME_SLOT, ID_NET_NAME_PATH, ID_NET_NAME_MAC en examinant le périphérique de l'interface réseau. Notez que certaines propriétés de périphérique risquent de ne pas être définies.

4. Une règle qui se trouve dans **`/usr/lib/udev/rules.d/80-net-name-slot.rules`** instruit **udev** de renommer l'interface, si elle n'a pas été renommée dans les étapes 1 ou 2, et que le paramètre de noyau **`net.ifnames=0`** n'a pas été donné, en suivant les priorités suivantes : ID_NET_NAME_ONBOARD, ID_NET_NAME_SLOT, ID_NET_NAME_PATH. Il se retrouve comme suivant dans la liste, si l'une n'est pas définie, sinon, l'interface ne sera pas renommée.

Les étapes 3 et 4 sont implémentées dans les schéma de dénomination 1, 2, 3, et optionnellement 4, décrits dans Section 8.1, « Schéma de dénomination ». L'étape 2 est détaillée dans Section 8.6, « Nommage de périphériques réseaux consistante avec biosdevname ».

8.3. COMPRENDRE LE NOMMAGE DE PÉRIPHÉRIQUE D'INTERFACE DE RÉSEAU PRÉDICTIBLE

Les noms ont des préfixes composés de deux caractères pour le type d'interface :

1. **en** pour Ethernet,

2. **`wl`** pour wireless LAN (WLAN),

3. **`ww`** pour wireless wide area network (WWAN).

Les noms prennent les formes suivantes :

Tableau 8.1. Types de noms de périphériques

Format	Description
o<*index*>	numéro d'index de périphérique on-board

Format	Description
s<*slot*>[f<*function*>][d<*dev_id*>]	numéro d'index de slot d'enfichage à chaud
x<*MAC*>	adresse MAC
p<*bus*>s<*slot*>[f<*function*>][d<*dev_id*>]	Emplacement géographique PCI
p<*bus*>s<*slot*>[f<*function*>][u<*port*>][..][c<*config*>][i<*interface*>]	chaîne de numéros de port USB

- Tous les périphériques PCI multi-fonctions porteront le numéro de [f<*function*>] dans le nom de périphérique, y compris function 0 device.
- Pour les périphériques USB, une chaîne complète de numéros de ports de hubs est composée. Si le nom est supérieur à 15 caractères (le nombre max), le nom ne pourra pas être exporté.
- Les valeurs des descripteurs de configuration USB == 1 et les descripteurs d'interface USB == 0 sont supprimées (configuration == 1 et interface == 0 sont les valeurs pas défaut si une configuration USB uniquement ou une interface existent).

8.4. SCHÉMA DE DÉNOMINATION POUR LES PÉRIPHÉRIQUES RÉSEAU DE LINUX SUR SYSTÈME Z

Utilise l'ID de bus pour créer des noms de périphériques prévisibles pour les interfaces réseau dans Linux sur les instances de System z. L'ID de bus identifie un périphérique dans le sous-système de canal s390. Un ID de bus identifie le périphérique dans une instance de Linux. Pour un périphérique CCW, l'ID de bus est le numéro du périphérique avec **0.n**, où **n** correspond à l'ID définie par le sous-canal. Par exemple, **0.1.0ab1**.

Les interfaces réseaux de périphériques Ethernet sont nommées ainsi :

```
enccw0.0.1234
```

Les périphériques réseaux CTC de périphériques de type SLIP sont nommés ainsi :

```
slccw0.0.1234
```

Utiliser la commande **znetconf -c** ou bien la commande **lscss -a** pour afficher les périphériques réseau disponibles et leurs ID de bus.

Tableau 8.2. Noms de types de périphériques pour Linux sur System z

Format	Description
enccw*bus-ID*	Types de périphériques Ethernet
slccw*bus-ID*	Périphériques réseau CTC de périphériques de type SLIP

8.5. SCHÉMA DE DÉNOMINATION POUR LES INTERFACES VLAN

Traditionnellement, les noms d'interface VLAN du format : *interface-name*. *VLAN-ID* sont utilisés. L' **ID VLAN** varie entre **0** et **4096**, avec un maximum de quatre caractères et le nom de l'interface total est limité à 15 caractères. La longueur de nom d'interface maximale est définie par les en-têtes du noyau et est une limite qui affecte toutes les applications.

Dans Red Hat Enterprise Linux 7, quatre conventions de nommage de noms d'interface VLAN sont prises en charge :

VLAN plus ID VLAN

Le mot **vlan** plus ID VLAN. Par exemple: vlan0005

VLAN plus ID VLAN sans remplissage

Le mot **vlan** plus ID VLAN sans remplissage en ajoutant des zéros devant. Exemple: vlan5

Nom du périphérique plus ID VLAN

Le nom de l'interface parente plus l'ID VLAN. Exemple : eth0.0005

Le nom du périphérique plus l'ID VLAN sans remplissage

Le nom de l'interface parente plus ID VLAN sans remplissage en ajoutant des zéros devant. Exemple: eth0.5

8.6. NOMMAGE DE PÉRIPHÉRIQUES RÉSEAUX CONSISTANTE AVEC BIOSDEVNAME

Cette fonctionnalité, implémentée via l'assistant **biosdevname udev** changera le nom de toutes les interfaces de réseaux imbriquées, des interfaces de réseau de cartes PCI, des interfaces de réseaux de fonctions virtuelles par rapport aux noms d'interfaces existants **eth[0123…]** et en suivant la nouvelle convention d'affectation de noms expliquée ici Tableau 8.3, « La convention de nommage biosdevname ». Notez qu'à moins que le système soit un système Dell, ou que **biosdevname** soit activé explicitement comme décrit dans Section 8.6.2, « Activer et désactiver la fonctionnalité », la convention d'affectation de noms **systemd** prévaudra.

Tableau 8.3. La convention de nommage biosdevname

Périphérique	Ancien nom	Nouveau nom
Interface de réseau intégrée (LOM)	**eth[0123…]**	**em[1234…]**[a]
Interface de réseau de carte PCI	**eth[0123…]**	**p<*slot*>p<*ethernet port*>**[b]
Fonction virtuelle	**eth[0123…]**	**p<*slot*>p<*ethernet port*>_<*virtual interface*>**[c]

Périphérique	Ancien nom	Nouveau nom

[a] New enumeration starts at **1**.

[b] For example: **p3p4**

[c] For example: **p3p4_1**

8.6.1. Conditions préalables Système

Le programme **biosdevname** utilise les informations du BIOS du système, et plus particulièrement, les champs *type 9* (Slot Système) et *type 41* (Onboard Devices Extended Information) contenus dans le SMBIOS. Si le BIOS du système n'a pas un SMBIOS version 2.6, ou version ultérieure et ces données, la nouvelle convention de nommage ne sera pas utilisée. Le matériel ancien ne prend pas en charge cette fonction en raison du manque d'informations de champs et de version SMBIOS correcte dans les BIOS. Pour plus d'informations sur la version SMBIOS ou BIOS, contactez votre fournisseur de matériel.

Pour que cette fonctionnalité puisse prendre effet, le package biosdevname doit être installé. Pour l'installer, exécutez la commande suivante, en tant qu'utilisateur **root** :

```
~]# yum install biosdevname
```

8.6.2. Activer et désactiver la fonctionnalité

Pour désactiver cette fonctionnalité, passez l'option suivante en ligne de commande, avant et après l'installation :

```
biosdevname=0
```

Pour activer cette fonctionnalité, passez l'option suivante en ligne de commande, avant et après l'installation :

```
biosdevname=1
```

À moins que le système remplisse les conditions préalables minimum, cette option sera ignorée, et le système utilisera le schéma d'affectation de noms de **systemd** décrit en début de chapitre.

Si l'option d'installation **biosdevname** est spécifiée, elle doit demeurer une option boot pour la durée de vie du système.

8.7. NOTES POUR LES ADMINISTRATEURS

Plusieurs fichiers de personnalisation du système peuvent inclure des noms d'interfaces réseau, et donc, cela nécessitera des mises à jour si on doit passer de l'ancienne à la nouvelle convention dans le système. Si vous utilisez la nouvelle convention de nommage, vous devrez également mettre à jour les noms d'interfaces de réseau à certains endroits, comme dans les règles d'**iptables**, les scripts de modification **irqbalance** et autres fichiers de configuration semblables. Aussi, permettre cette modification d'installation nécessitera des changements aux fichiers **kickstart** existants qui utilisent des noms de périphériques via le paramètre **ksdevice** ; ces fichiers **kickstart** devront être mis à jour afin de pouvoir utiliser le nouveau nom de périphérique réseau ou l'adresse MAC du périphérique réseau.

NOTE

La longueur du nom d'interface maximum est définie par les en-têtes de noyau et représente une limite globale, affectant toutes les applications.

8.8. CONTRÔLE DE LA SÉLECTION DE NOMS DE PÉRIPHÉRIQUES RÉSEAU

Le nommage de périphériques peut être contrôlé de la manière suivante :

En identifiant le périphérique d'interface de réseau

Configurer l'adresse MAC dans un fichier **ifcfg** en utilisant la directive **HWADDR** lui permet d'être identifiée par **udev**. On prendra le nom de la chaîne donnée par la directive du **DEVICE**, qui, par convention, est le même que le suffixe **ifcfg**. Exemple, **ifcfg**-*eth0*.

En activant / désactivant biosdevname

Le nom fourni par **biosdevname** sera utilisé (si **biosdevname** peut en déterminer un).

En activant ou désactivant le schéma de dénomination de systemd-udev

Le nom fourni par **systemd-udev** sera utilisé (si **systemd-udev** peut en déterminer un).

8.9. DÉSACTIVER LE NOMMAGE DE PÉRIPHÉRIQUES RÉSEAUX CONSISTANT

Pour désactiver le système d'affectation de noms de périphériques réseau consistante, choisir parmi les méthodes suivantes :

- Désactiver l'affectation des noms fixes, de façon à ce que les noms de noyau imprévisibles soient utilisés à nouveau, en masquant le fichier de règles **udev** pour la politique par défaut. Ce « masque » peut être créé par la mise en place d'un lien symbolique **/dev/null**. En tant qu'utilisateur **root**, exécutez la commande suivante :

```
~]# ln -s /dev/null /etc/udev/rules.d/80-net-name-slot.rules
```

- Créer votre propre schéma manuel de nommage, en nommant, par exemple, vos interfaces « internet0 », « dmz0 » ou « lan0 ». Pour cela, créer votre propre fichier de règles **udev**, et définissez la propriété NAME pour les périphériques. Assurez-vous d'ordonnancer ce fichier avant le fichier de politique par défaut, en le nommant, par exemple : **/etc/udev/rules.d/70-my-net-names.rules**.

- Altérer le fichier de politique par défaut pour sélectionner un schéma de dénomination différent, comme par exemple, de nommer toutes les interfaces en suivant leurs adresses MAC par défaut. En tant qu'utilisateur **root**, copiez le fichier de politique par défaut comme suit :

```
~]# cp /usr/lib/udev/rules.d/80-net-name-slot.rules
/etc/udev/rules.d/80-net-name-slot.rules
```

  Modifiez le répertoire **/etc/udev/rules.d/** et changez les lignes qui conviennent.

- Ajouter 'net.ifnames=0' en tant que paramètre de noyau au menu GRUB_CMDLINE_LINUX variable of GRUB 2.

```
GRUB_CMDLINE_LINUX="net.ifnames=0"
```

NOTE

GRUB_CMDLINE_LINUX peut contenir une configuration supplémentaire selon la configuration système.

Pour mettre à jour toutes les entrées de menu de noyau de GRUB 2, saisir une commande en tant que superutilisateur **root** comme suit :

```
~]# grubby --update-kernel=ALL --args=net.ifnames=0
```

Pour obtenir plus d'informations surGRUB 2, voir le guide *Red Hat Enterprise Linux 7 System Administrator's Guide*.

8.10. RÉSOLUTION DE PROBLÈMES POUR LE NOMMAGE DE PÉRIPHÉRIQUES RÉSEAUX

Les noms d'interfaces prévisibles seront alloués à chaque interface, si possible, selon la procédure décrite dans Section 8.2, « Comprendre la procédure d'affectation de noms aux périphériques ». Pour voir la liste de noms possibles qu'**udev** va utiliser, exécutez une commande, comme suit, en tant qu'utilisateur **root** :

```
~]# udevadm info /sys/class/net/ifname | grep ID_NET_NAME
```

quand *ifname* est l'une des interfaces listées par la commande suivante :

```
~]$ ls /sys/class/net/
```

Une possibilité de nom sera appliquée par **udev** suivant les règles décrites dans Section 8.2, « Comprendre la procédure d'affectation de noms aux périphériques », dont voici un récapitulatif :

- **/usr/lib/udev/rules.d/60-net.rules** - d'initscripts,
- **/usr/lib/udev/rules.d/71-biosdevname.rules** - de **biosdevname**,
- **/usr/lib/udev/rules.d/80-net-name-slot.rules** - de **systemd**

Dans la liste de fichiers de règles ci-dessus, on peut constater que si l'interface d'affectation de noms se fait via initscripts ou **biosdevname**, elle prévaudra toujours sur la politique native d'**udev**. Cependant, si le renommage d'initscripts n'a pas lieu et que **biosdevname** est désactivé, alors, pour modifier les noms d'interfaces, copier **80-net-name-slot.rules** de **/usr** dans **/etc /** et modifier le fichier correctement. En d'autres termes, commentez ou organisez les schéma pour qu'ils puissent être utilisés dans un certain ordre.

Exemple 8.1. Certaines interfaces ont des noms d'espace-noms de noyaux (eth[0,1,2...]) alors que d'autres sont renommés par udev

Les schéma mixtes signifient probablement que pour certains matériels, il n'y a aucune information utilisable fournie par le noyau pour **udev**, donc il ne peut pas comprendre que les noms ou les informations fournies à **udev** ne sont pas appropriées, comme les ID de périphériques non uniques.

Cela est assez courant et la solution est d'utiliser un schéma de nommage personnalisé dans les fichiers ifcfg ou altérer quel schéma **udev** est utilisé en éditant 80-net-name-slot.rules.

Exemple 8.2. Dans /var/log/messages ou dans le journal de systemd journal, on voit le renommage apparaître à deux reprises pour chaque interface

Les systèmes avec le schéma de nommage encodé dans les fichiers ifcfg, mais qui n'ont pas une image **`initrd`** régénérée sont susceptibles de rencontrer ce problème. Le nom de l'interface est initialement assigné (via **biosdevname** ou **udev** ou par les paramètres dracut sur la ligne de commande du noyau) lors du démarrage boot précoce, quand on est encore dans **`initrd`**. Puis, après le passage au **`rootfs`** réel, le changement de noms a lieu une deuxième fois et un nouveau nom d'interface est déterminé par le binaire de **`usr/lib/udev/rename_device`** engendré par **udev** par les 60-net.rules. Vous pouvez ignorer de tels messages sans inquiétude.

Exemple 8.3. Utiliser un schéma de dénomination avec des noms ethX dans les fichiers ifcfg ne marche pas

L'utilisation de noms d'interfaces et d'espace-noms de noyaux n'est pas conseillé. Pour obtenir des noms d'interface prévisibles et stables, utiliser d'autres préfixes qu' "eth".

8.11. RESSOURCES SUPPLÉMENTAIRES

Les sources d'informations suivantes fournissent des ressources supplémentaires à propos des associations de réseaux.

8.11.1. Documentation installée

- Page man **`udev(7)`** — décrit le démon de gestion du périphérique dynamique de Linux **`udevd`**.
- Page man **`systemd(1)`** man page — décrit le système et le gestionnaire de service **`systemd`**.
- Page man **`biosdevname(1)`** — décrit un utilitaire pour obtenir le nom généré-BIOS d'un périphérique.

8.11.2. Documentation en ligne

- IBM Knowledge Center Publication SC34-2710-00 les *Pilotes de périphériques, Composants et Commandes dans Red Hat Enterprise Linux 7* inclut des informations sur les « Noms de périphériques réseau prévisibles » pour les attachements et les périphériques IBM System z.

PARTIE II. INFINIBAND ET RDMA NETWORKING

Cette partie traite de la façon de configurer RDMA, InfiniBand, et IP sur les connexions de réseau InfiniBand.

CHAPITRE 9. CONFIGURATION D'INFINIBAND ET DES RÉSEAUX RDMA

9.1. COMPRENDRE LES TECHNOLOGIES INFINIBAND ET RDMA

InfiniBand désigne deux choses distinctes. D'une part, il s'agit d'un protocole de couche de liaison physique pour réseaux InfiniBand. Deuxièmement, cela se réfère à une API de programmation de niveau supérieur appelée l'API InfiniBand Verbs. L'API InfiniBand Verbs est une implémentation d'une technologie *remote direct memory access* (RDMA).

Les communications RDMA se distinguent des communications **IP** normales car elles contournent l'intervention du noyau dans le processus de communication et réduisent ainsi considérablement la surcharge de processeur qu'il faut normalement pour traiter les communications réseau. Dans un transfert de données **IP** typique, une application X sur une machine A enverra certaines données à une application Y sur un ordinateur B. Dans le cadre du transfert, le noyau de l'ordinateur B doit tout d'abord recevoir les données, décoder les en-têtes de paquets, déterminer que les données appartiennent bien à l'application Y, réveiller l'application Y, attendre que l'application Y effectue une lecture syscall dans le noyau, puis doit copier manuellement les données de l'espace mémoire interne du noyau propre dans la mémoire tampon fournie par application Y. Ce processus signifie que la majorité du trafic réseau doit être copié dans la mémoire bus principale du système au moins deux fois (une fois, quand l'adaptateur hôte utilise le DMA pour mettre les données dans la mémoire tampon de la mémoire fournie par le noyau, et à nouveau, quand le noyau déplace les données vers la mémoire tampon de l'application), et cela signifie aussi que l'ordinateur doit exécuter un certain nombre de changements de contexte pour basculer du contexte noyau au contexte d'application Y. Ces deux choses imposent des charges de CPU extrêmement élevées sur le système lorsque le trafic réseau circule à une cadence très élevée.

Le protocole RDMA permet à l'adaptateur d'hôte de la machine de savoir quand un paquet arrive sur le réseau, quelle application doit recevoir ce paquet, et où il doit aller dans l'espace mémoire de l'application. Au lieu d'envoyer le paquet au noyau pour qu'il soit traité, et pour qu'il soit ensuite copié dans la mémoire de l'application de l'utilisateur, il place le contenu du paquet directement dans le tampon de l'application sans qu'il y ait besoin d'une autre intervention. Cela réduit considérablement la charge des communications réseau à haute vitesse. Cependant, cela ne peut être accompli en utilisant l'API de Sockets Berkeley standard sur laquelle la plupart des applications réseau **IP** se reposent, donc il doit fournir sa propre API, l'API InfiniBand Verbs, et les applications doivent être déplacées vers cette API, avant de pouvoir utiliser la technologie RDMA directement.

Red Hat Enterprise Linux 7 prend en charge le matériel InfiniBand et l'API InfiniBand Verbs. Aussi, il existe deux technologies supplémentaires qui sont prises en charge et permettant à l'API InfiniBand Verbs d'être utilisé sur du matériel non-InfiniBand. Il s'agit d'iWARP (Internet Wide Area RDMA Protocol) et RCI/IBoE (RDMA via Converged Ethernet, plus tard renommé InfiniBand over Ethernet). Ces deux technologies ont une couche de liaison de réseau **IP** normale comme leur technologie sous-jacente, et donc la majorité de leur configuration est effectivement couverte dans le chapitre Chapitre 2, *Configurer la Gestion des réseaux IP* du présent document. Pour l'essentiel, une fois que leurs fonctionnalités réseau **IP** sont correctement configurées, leurs fonctionnalités RDMA sont automatiques et apparaîtront aussi longtemps que les pilotes qu'il faut seront installés. Les pilotes du noyau sont toujours inclus dans chaque noyau que Red Hat fournit ; cependant, les pilotes de l'espace utilisateur doivent être installés manuellement si le groupe de packages InfiniBand n'a pas été sélectionné lors de l'installation de machine.

Voici les packages espace-utilisateur qu'il faut :

iWARP

matériel Chelsio — **libcxgb3** ou **libcxgb4** selon la version du matériel

RoCE/IBoE

`matériel Mellanox` — **libmlx4** ou **libmlx5**, selon la version du matériel. De plus, modifiez les fichiers **`/etc/rdma/mlx4.conf`** selon les besoins, afin de définir les types de ports correctement pour l'utilisation RoCE/IBoE. Modifiez les fichiers **`/etc/modprobe.d/mlx4.conf`** pour configurer quelle priorité de paquet correspond au service « no-drop » sur les commutateurs Ethernet sur lesquels les cartes sont connectées.

Avec ces paquets de pilotes installés (en plus les paquets RDMA normaux généralement installés avec les installations InfiniBand), un utilisateur doit être en mesure d'utiliser la plupart des applications RDMA normales pour tester et voir le protocole de communication RDMA sur leurs adaptateurs. Cependant, tous les programmes inclus dans Red Hat Enterprise Linux 7 ne prennent pas tous en charge les périphériques iWARP ou RoCE/IBoE. C'est parce que le protocole de mise en place de connexion sur iWARP en particulier, est différent que c'est sur les liens réels de couche liaison InfiniBand. Si le programme en question utilise la bibliothèque de gestion de connexion **librdmacm**, il prendra en charge les différences entre iWARP et InfiniBand silencieusement et le programme devrait fonctionner. Si l'application essaie de faire sa propre gestion des connexions, alors, il doit prendre en charge spécifiquement iWARP, sinon cela ne fonctionnera pas.

9.2. PAQUETS DE LOGICIELS RELATIFS À INFINIBAND ET À RDMA

Comme les applications RDMA sont tellement différentes des applications basées Berkeley Sockets et du réseautage **`IP`** normal, la plupart des applications qui sont utilisées sur un réseau **`IP`** ne peuvent pas être utilisées directement sur un réseau RDMA. Red Hat Enterprise Linux 7 est livré avec un certain nombre de paquets de logiciels pour l'administration de réseaux RDMA, des tests et débogages, des API de développement de logiciels de haut niveau, et d'analyse des performances.

Pour pouvoir utiliser ces réseaux, certains de ces paquets doivent être installés (cette liste n'est pas complète, mais elle couvre les paquets les plus importants relatifs à RDMA).

Paquets requis :

- **`rdma`** — chargé de l'initialisation du noyau de la pile RDMA.
- **`libibverbs`** — fourni l'API InfiniBand Verbs.
- **`opensm`** — gestionnaire de sous-réseau (requis sur une machine uniquement, et seulement s'il n'y a pas de gestionnaire de sous-réseau actif sur la structure).
- **`user space driver for installed hardware`** — un ou plusieurs parmi les paquets suivants : infinipath-psm, libcxgb3, libcxgb4, libehca, libipathverbs, libmthca, libmlx4, libmlx5, libnes, and libocrdma. Notez que libehca n'est disponible que pour les serveurs IBM Power Systems.

Packages recommandés :

- **`librdmacm, librdmacm-utils, et ibacm`** — Bibliothèque de gestion des connexions qui prend en considération les différences entre InfiniBand, iWARP, et RoCE et qui est en mesure d'ouvrir correctement les connexions sur l'ensemble de ces types de matériels, certains programmes de test simples pour vérifier le fonctionnement du réseau et un démon de cache qui s'intègre à la bibliothèque pour rendre la résolution d'hôte distant en larges clusters plus rapide.
- **`libibverbs-utils`** — Simples programmes basés Verbs pour chercher le matériel installé et vérifier les communications sur la structure.

- **infiniband-diags et ibutils** — Fournit un certain nombre d'outils de débogage utiles pour la gestion de la structure InfiniBand. Ils fournissent uniquement des fonctionnalités très limitées sur iWARP ou RoCE car la plupart des outils de travail opérent dans la couche de liaison InfiniBand, et non pas dans la couche d'API Verbs.

- **perftest et qperf** — Applications de test de performance pour divers types de communications RDMA.

Packages en option :

Ces paquets sont disponibles sur le canal Optional. Avant d'installer les paquets à partir du canal Optional, voir Scope of Coverage Details. Les informations sur les abonnements au canal Optional se trouvent dans la base de connaissances Red Hat How to access Optional and Supplementary channels.

- **dapl, dapl-devel, et dapl-utils** — Fournit une API différente que l'API Verbs pour RDMA. Il y a à la fois un composant de runtime et un composant de développement à ces packages.

- **openmpi, mvapich2, et mvapich2-psm** — Piles MPI qui ont la possibilité d'utiliser les communications RDMA. Les applications d'espace utilisateur qui écrivent dans ces piles ne savent pas forcéement que des communications RDMA ont lieu.

9.3. CONFIGURER LE SOUS-SYSTÈME RDMA DE BASE

9.3.1. L'installation du package RDMA

Le paquet rdma ne fait pas partie de l'ensemble de packages à installer par défaut. Si le groupe de packages InfiniBand n'a pas été sélectionné pendant l'installation, le paquet rdma (ainsi que quelques autres énumérés dans la section précédente) peuvent être installés après que l'installation initiale soit terminée. S'il n'a pas été installé au moment de l'installation de la machine et a été installé manuellement par la suite, alors il convient de reconstruire les images **initramfs** avec **dracut** afin qu'il fonctionne parfaitement, comme prévu. Exécutez les commandes suivantes en tant qu'utilisateur **root** :

```
~]# yum install rdma
dracut -f
```

Le démarrage du service **rdma** est automatique. Quand du matériel compatible RDMA, InfiniBand, iWARP ou RoCE/IBoE est détecté, **udev** charge **systemd** de démarrer le service **rdma**. Les utilisateurs n'ont pas besoin d'activer le service **rdma**, mais ils peuvent le forcer à tout moment s'ils le souhaitent. Pour ce faire, exécutez la commande suivante :

```
~]# systemctl enable rdma
```

9.3.2. Configuration du fichier rdma-conf

Le service **rdma** lit **/etc/rdma/rdma.conf** pour trouver quels protocoles RDMA niveau utilisateur ou niveau noyau, l'administrateur veut charger par défaut. Les utilisateurs doivent éditer ce fichier pour démarrer ou stopper les différents pilotes.

Voici un certain nombre de pilotes qui peuvent être activés ou désactivés :

- **IPoIB** — Il s'agit d'une couche d'émulation de réseaux **IP** qui permet aux applications **IP** d'exécuter sur les réseaux InfiniBand.

- **SRP** — Il s'agit du protocole de demandes SCSI Request Protocol. Ce protocole permet à une machine de monter un lecteur ou un groupe de lecteurs distants exportés sur la machine, via le protocole **SRP**, comme s'il s'agissait d'un disque dur local.

- **SRPT** — c'est le mode cible ou serveur du protocole **SRP**. Ceci charge le support noyau nécessaire pour pouvoir exporter un lecteur ou un groupe de lecteurs pour que d'autres machines puissent être montées, comme s'il était local sur leur machine. Vous devrez effectuer une configuration plus poussée en mode cible avant de pouvoir exporter un périphérique. Consultez la documentation dans les paquets targetd et targetcli pour de plus amples informations.

- **ISER** — Il s'agit d'un pilote de bas niveau de la couche générale iSCSI du noyau Linux qui fournit un transport sur les réseaux InfiniBand pour les périphériques iSCSI.

- **RDS** — Reliable Datagram Service du noyau Linux. Non actif dans les noyaux Red Hat Enterprise Linux 7 et ne peut donc pas être téléchargé.

9.3.3. Utilisation de 70-persistent-ipoib.rules

Le paquet de rdma fournit le fichier **/etc/udev.d/rules.d/70-persistent-ipoib.rules**. Ce fichier de règles **udev** est utilisé pour renommer des dispositifs IPoIB de leurs noms par défaut (par exemple, **ib0** et **ib1**) à des noms plus descriptifs. Les utilisateurs doivent modifier ce fichier pour changer comment nommer leurs périphériques. Tout d'abord, trouver l'adresse GUID pour le périphérique qui doit être renommé :

```
~]$ ip link show ib0
8: ib0: >BROADCAST,MULTICAST,UP,LOWER_UP< mtu 65520 qdisc pfifo_fast state
UP mode DEFAULT qlen 256
    link/infiniband
80:00:02:00:fe:80:00:00:00:00:00:00:f4:52:14:03:00:7b:cb:a1 brd
00:ff:ff:ff:ff:12:40:1b:ff:ff:00:00:00:00:00:00:ff:ff:ff:ff
```

Immédiatement après **lien/infiniband** se trouve l'adresse de matériel de 20 octets pour l'interface IPoIB. Les 8 octets finaux de l'adresse, marqué en gras ci-dessus, est tout ce qu'il faut pour créer un nouveau nom. Les utilisateurs peuvent créer un schéma de nommage qui leur convient. Par exemple, utiliser une convention d'affectation de noms *device_fabric* comme **mlx4_ib0**, si un périphérique **mlx4** est connecté à une fibre de sous-réseau **ib0**. La seule chose qui n'est pas recommandée est d'utiliser les noms standards, comme **ib0** ou **ib1**, car ceux-ci peuvent entrer en conflit avec les noms assignés automtiquement par le noyau. Ensuite, ajoutez une entrée dans le fichier de règles. Copiez l'exemple existant dans le fichier de règles, remplacez les 8 octets dans l'entrée **ATTR {address}** avec les 8 octets en surbrillance du périphérique qui doit être renommé, et entrez le nouveau nom à utiliser dans le champ **NAME**.

9.3.4. Relaxation des restrictions memlock pour les utilisateurs

Les communications RDMA exigent que la mémoire physique de l'ordinateur soit épinglée (ce qui signifie que le noyau ne doit pas être autorisé à échanger cette mémoire dans un fichier d'échange dans le cas où l'ordinateur, dans son ensemble, commence à être à court de mémoire disponible). Normalement, l'épinglage de la mémoire est une opération très privilégiée. Pour permettre aux utilisateurs autres que **root** d'exécuter des applications RDMA de grande envergure, il va probablement falloir augmenter la quantité de mémoire que les utilisateurs non - **root** sont autorisés à épingler dans le système. Cela se fait en ajoutant un fichier dans le répertoire **/etc/security/limits.d/** avec des contenus semblables à ceci :

```
~]$ more /etc/security/limits.d/rdma.conf
# configuration for rdma tuning
*       soft    memlock         unlimited
*       hard    memlock         unlimited
# rdma tuning end
```

9.3.5. Configurer des cartes Mellanox pour l'opération Ethernet

Certains matériels de Mellanox sont capables d'exécuter en mode Ethernet ou InfiniBand. Ces cartes ont généralement InfiniBand comme valeur par défaut. Les utilisateurs peuvent définir les cartes en mode Ethernet. Il y a actuellement un support pour définir le mode sur matériel ConnectX family uniquement (qui utilise le pilote **mlx4**). Pour définir le mode, les utilisateurs doivent suivre les instructions dans le fichier **/etc/rdma/mlx4.conf** pour trouver l'ID de périphérique PCI pour leur matériel. Ils doivent ensuite créer une ligne dans le fichier à l'aide de cet ID de périphérique et le type de port requis, puis reconstruire leur **initramfs** pour s'assurer que les paramètres de port mis à jour sont copiés dans **initramfs**.

Une fois que le type de port a été défini, si un ou deux ports sont définis sur Ethernet, les utilisateurs peuvent apercevoir ce message dans leurs journaux : **mlx4_core 0000:05:00.0: Requested port type for port 1 is not supported on this HCA**. C'est normal et n'affectera pas le fonctionnement. Le script chargé de définir le type de port n'a aucun moyen de savoir quand le pilote a fini de passer du port 2 au type demandé en interne, et le temps que le script émette une requête de changement au port 2 et que le changement soit terminé, les tentatives de régler le port 1 à un autre type sont rejetées. Le script tente à nouveau jusqu'à ce que la commande réussisse, ou jusqu'à ce qu'un délai d'attente soit écoulé, indiquant que le changement de port n'a pas réussi.

9.4. CONFIGURER LE GESTIONNAIRE DES SOUS-RÉSEAUX

9.4.1. Déterminer la nécessité

La plupart des commutateurs InfiniBand disposent d'un gestionnaire de sous-réseaux intégré. Toutefois, au cas où un gestionnaire de sous-réseau plus récemment mis à jour que le commutateur firmware soit requis, ou si un contrôle plus complet que le gestionnaire de commutateurs est nécessaire, Red Hat Enterprise Linux 7 inclut le gestionnaire de sous-réseaux **opensm**. Tous les réseaux InfiniBand **doivent** avoir un gestionnaire de sous-réseaux en cours d'exécution pour que le réseau fonctionne. Cela est vrai même si un simple réseau composé de deux machines sans interrupteur et des cartes sont branchées dos à dos, et un gestionnaire de sous-réseau est requis pour que le lien apparaisse sur les cartes. Il est possible d'en avoir plus d'un, auquel cas, l'un d'entre eux agira comme master, et tout autre sous-réseau de gestionnaire agira comme esclave, prêt à prendre la relève si le gestionnaire de sous-réseaux venait à échouer.

9.4.2. Configuration d'un fichier de configuration master opensm

Le programme **opensm** conserve son fichier de configuration maître dans **/etc/rdma/opensm.conf**. Les utilisateurs peuvent modifier ce fichier à tout moment et les modifications seront conservées lors de la mise à niveau. Vous trouverez une documentation complète sur les options dans le fichier lui-même. Toutefois, pour les deux modifications les plus courantes, c-a-d définir le GUID auquel se lier et la PRIORITÉ à encourir, il est fortement recommandé que le fichier **opensm.conf** ne soit pas édité, mais que le fichier **/etc/sysconfig/opensm** le soit à la place. S'il n'y a aucune modification dans le fichier de base **/etc/rdma/opensm.conf**, il sera mis à jour à chaque fois que le paquet opensm sera mis à jour. Comme de nouvelles options sont ajoutées régulièrement à ce fichier, cela facilite l'actualisation de la configuration actuelle. Si le fichier **opensm.conf** a été changé, alors, au moment de la mise à niveau, il faudra sans doute faire fusionner les nouvelles options dans le fichier édité.

9.4.3. Configurer les options opensm startup

Les options qui se trouvent dans le fichier **/etc/sysconfig/opensm** contrôlent comment le gestionnaire de sous-réseaux démarre, ainsi que le nombre d'exemplaires du gestionnaire de sous-réseaux démarrent. Par exemple, une carte de double port InfiniBand, avec chaque port branché sur des réseaux physiquement séparés, aura besoin d'une copie du gestionnaire de sous-réseaux en cours d'exécution sur chaque port. Le gestionnaire de sous-réseaux **opensm** va gérer uniquement un sous-réseau par instance d'application, et doit être exécuté à chaque fois pour chaque sous-réseau qui doit être géré. De plus, s'il y a plus d'un serveur **opensm**, définissez les priorités sur chaque serveur pour contrôler ceux qui doivent être des esclaves et ceux qui doivent être maîtres.

Le fichier **/etc/sysconfig/opensm** est utilisé pour fournir un moyen simple de définir la priorité du gestionnaire de sous-réseaux et pour contrôler à quel GUID le gestionnaire de sous-réseaux se lie. Il y a une explication approfondie des options dans le fichier **/etc/sysconfig/opensm** lui-même. Les utilisateurs ont uniquement besoin de lire et de suivre les instructions contenues dans le fichier pour permettre le basculement et l'opération Multi Fabric d'**opensm**.

9.4.4. Création d'une définition P_Key

Par défaut, **opensm.conf** recherche le fichier **/etc/rdma/partitions.conf** pour obtenir une liste de partitions à créer sur la structure. Toutes les structures doivent contenir le sous-réseau **0x7fff**, et tous les commutateurs et hôtes doivent appartenir à cette structure. Toute autre partition peut être créée en plus, et tous les hôtes et interrupteurs n'ont pas besoin d'être membres de ces partitions supplémentaires. Cela permet à un administrateur de créer des sous-réseaux qui ressemblent aux structures VLAN Ethernet ou InfiniBand. Si une partition est définie par vitesse donnée, par exemple 40 Gbit/s, et qu'il y a un hôte sur le réseau qui est incapable d'aller à 40 Gbit/s, alors cet hôte sera incapable de joindre la partition, même s'il est autorisé à le faire car il ne pourra pas répondre aux exigences de vitesse. Il est donc recommandé que la vitesse d'une partition soit définie à la vitesse la plus lente correspondant à tout hôte ayant la permission de rejoindre la partition. S'il vous faut une partition plus rapide de certains sous-ensembles d'hôtes, alors, créez une partition différente avec la vitesse plus élevée pour cette partition.

Le fichier de partition suivant équivaudrait à une partition par défaut de **0x7fff** à une vitesse réduite de 10 Gops, et une partition **0x0002** avec une vitesse de 40 Gops :

```
~]$ more /etc/rdma/partitions.conf
# For reference:
# IPv4 IANA reserved multicast addresses:
#   http://www.iana.org/assignments/multicast-addresses/multicast-
addresses.txt
# IPv6 IANA reserved multicast addresses:
#   http://www.iana.org/assignments/ipv6-multicast-addresses/ipv6-
multicast-addresses.xml
#
# mtu =
#   1 = 256
#   2 = 512
#   3 = 1024
#   4 = 2048
#   5 = 4096
#
# rate =
#   2  =   2.5 GBit/s
#   3  =  10   GBit/s
#   4  =  30   GBit/s
#   5  =   5   GBit/s
```

```
#   6  =  20   GBit/s
#   7  =  40   GBit/s
#   8  =  60   GBit/s
#   9  =  80   GBit/s
#   10 = 120   GBit/s

Default=0x7fff, rate=3, mtu=4, scope=2, defmember=full:
        ALL, ALL_SWITCHES=full;
Default=0x7fff, ipoib, rate=3, mtu=4, scope=2:
        mgid=ff12:401b::ffff:ffff       # IPv4 Broadcast address
        mgid=ff12:401b::1               # IPv4 All Hosts group
        mgid=ff12:401b::2               # IPv4 All Routers group
        mgid=ff12:401b::16              # IPv4 IGMP group
        mgid=ff12:401b::fb              # IPv4 mDNS group
        mgid=ff12:401b::fc              # IPv4 Multicast Link Local Name
Resolution group
        mgid=ff12:401b::101             # IPv4 NTP group
        mgid=ff12:401b::202             # IPv4 Sun RPC
        mgid=ff12:601b::1               # IPv6 All Hosts group
        mgid=ff12:601b::2               # IPv6 All Routers group
        mgid=ff12:601b::16              # IPv6 MLDv2-capable Routers
group
        mgid=ff12:601b::fb              # IPv6 mDNS group
        mgid=ff12:601b::101             # IPv6 NTP group
        mgid=ff12:601b::202             # IPv6 Sun RPC group
        mgid=ff12:601b::1:3             # IPv6 Multicast Link Local Name
Resolution group
        ALL=full, ALL_SWITCHES=full;

ib0_2=0x0002, rate=7, mtu=4, scope=2, defmember=full:
        ALL, ALL_SWITCHES=full;
ib0_2=0x0002, ipoib, rate=7, mtu=4, scope=2:
        mgid=ff12:401b::ffff:ffff       # IPv4 Broadcast address
        mgid=ff12:401b::1               # IPv4 All Hosts group
        mgid=ff12:401b::2               # IPv4 All Routers group
        mgid=ff12:401b::16              # IPv4 IGMP group
        mgid=ff12:401b::fb              # IPv4 mDNS group
        mgid=ff12:401b::fc              # IPv4 Multicast Link Local Name
Resolution group
        mgid=ff12:401b::101             # IPv4 NTP group
        mgid=ff12:401b::202             # IPv4 Sun RPC
        mgid=ff12:601b::1               # IPv6 All Hosts group
        mgid=ff12:601b::2               # IPv6 All Routers group
        mgid=ff12:601b::16              # IPv6 MLDv2-capable Routers
group
        mgid=ff12:601b::fb              # IPv6 mDNS group
        mgid=ff12:601b::101             # IPv6 NTP group
        mgid=ff12:601b::202             # IPv6 Sun RPC group
        mgid=ff12:601b::1:3             # IPv6 Multicast Link Local Name
Resolution group
        ALL=full, ALL_SWITCHES=full;
```

9.4.5. Activation d'opensm

Les utilisateurs ont besoin d'activer le service **opensm** car il n'est pas actif par défaut une fois installé. Exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl enable opensm
```

9.5. TESTING D'ANCIENNES OPÉRATIONS RDMA INFINIBAND

NOTE

Cette section s'applique uniquement aux périphériques InfiniBand. Depuis que les périphériques iWARP et RCI/IBoE sont basés **IP**, les utilisateurs doivent procéder à la section sur les tests d'opérations, une fois qu'IPoIB aura été configuré et les périphériques auront des adresses **IP**.

Une fois que le service **rdma** est activé, que le service **opensm** (si nécessaire) est activé, et que la bibliothèque de l'espace utilisateur appropriée au matériel spécifique a été installée, les opération de **rdma** espace utilisateur devraient être rendues possibles. Les programmes de test simples, grâce au paquet libibverbs-utils, sont utiles pour vérifier que les opérations RDMA fonctionnent correctement. Le programme **ibv_devices** indiquera quels périphériques sont présents dans le système, et la commande **ibv_devinfo** donnera des informations détaillées sur chaque périphérique. Exemple :

```
~]$ ibv_devices
    device                 node GUID
    ------              ----------------
    mlx4_0              0002c903003178f0
    mlx4_1              f4521403007bcba0
~]$ ibv_devinfo -d mlx4_1
hca_id: mlx4_1
        transport:                      InfiniBand (0)
        fw_ver:                         2.30.8000
        node_guid:                      f452:1403:007b:cba0
        sys_image_guid:                 f452:1403:007b:cba3
        vendor_id:                      0x02c9
        vendor_part_id:                 4099
        hw_ver:                         0x0
        board_id:                       MT_1090120019
        phys_port_cnt:                  2
                port:   1
                        state:                  PORT_ACTIVE (4)
                        max_mtu:                4096 (5)
                        active_mtu:             2048 (4)
                        sm_lid:                 2
                        port_lid:               2
                        port_lmc:               0x01
                        link_layer:             InfiniBand

                port:   2
                        state:                  PORT_ACTIVE (4)
                        max_mtu:                4096 (5)
                        active_mtu:             4096 (5)
                        sm_lid:                 0
                        port_lid:               0
                        port_lmc:               0x00
```

```
                        link_layer:             Ethernet
~]$ ibstat mlx4_1
CA 'mlx4_1'
        CA type: MT4099
        Number of ports: 2
        Firmware version: 2.30.8000
        Hardware version: 0
        Node GUID: 0xf4521403007bcba0
        System image GUID: 0xf4521403007bcba3
        Port 1:
                State: Active
                Physical state: LinkUp
                Rate: 56
                Base lid: 2
                LMC: 1
                SM lid: 2
                Capability mask: 0x0251486a
                Port GUID: 0xf4521403007bcba1
                Link layer: InfiniBand
        Port 2:
                State: Active
                Physical state: LinkUp
                Rate: 40
                Base lid: 0
                LMC: 0
                SM lid: 0
                Capability mask: 0x04010000
                Port GUID: 0xf65214fffe7bcba2
                Link layer: Ethernet
```

Les sorties de commande de **ibv_devinfo** et de **ibstat** rendent des informations légèrement différentes (ainsi, le port MTU est présent dans **ibv_devinfo** mais pas dans la sortie de **ibstat**, et le port GUID est présent dans la sortie de **ibstat**, mais pas dans celle de **ibv_devinfo**) et certaines choses sont nommées différemment (par exemple, le *local identifier* (LID) de base dans la sortie **ibstat** correspond au **port_lid** de la sortie de **ibv_devinfo**)

Les programmes ping simples, comme **ibping** du paquet infiniband-diags, peuvent être utilisés pour tester la connectivité RDMA. Le programme **ibping** utilise un modèle client/serveur. Vous devez d'abord démarrer un serveur **ibping** sur une seule machine, puis exécuter **ibping** en tant que client sur une autre machine et lui demander de se connecter au serveur **ibping**. Comme nous sommes désireux de tester la fonctionnalité RDMA de base, nous avons besoin d'utiliser une méthode de résolution d'adresse spécifique RDMA, plutôt que d'adresses **IP** pour spécifier le serveur.

Sur le serveur, les utilisateurs peuvent utiliser les commandes **ibv_devinfo** et **ibstat** pour afficher le **port_lid** (lid de Base) et le Port GUID du port qu'ils veulent tester (en supposant que le port 1 de l'interface ci-dessus, le **port_lid** / **Base LID** corresponde à **2** et le Port GUID à **0xf4521403007bcba1**)). Puis, lancer **ibping** avec les options nécessaires pour lier spécifiquement la carte et le port à tester; et aussi, en spécifiant **ibping**, on doit exécuter en mode serveur. Vous pourrez apercevoir les options disponibles à **ibping** en ajoutant les options **-?** ou **--aider**, mais dans ce cas, il faudra soit l'option **-S** ou **--Server** et pour relier la carte au port, il faudra soit **-C** ou **--Ca** et **-P** ou **--Port**. Remarque : dans cette instance, le port ne désigne pas un numéro de port de réseau, mais désigne le numéro de port physique situé sur la carte lorsque vous utilisez une carte multi-ports. Pour tester la connectivité de la structure RDMA en utilisant, par exemple, le deuxième port d'une carte multiport, vous devrez instruire**ibping** de se lier au port **2** sur la carte. Lorsque vous utilisez une carte à port unique, ou si vous tester le premier port sur une carte, cette option n'est pas nécessaire. Exemple :

```
~]$ ibping -S -C mlx4_1 -P 1
```

Passez alors à la machine client et exécuter **ibping**. Prenez note de chaque port GUID du port auquel le programme **ibping** est lié, ou l' *identificateur local* (LID de l'anglais Local Indentifier) du port auquel le programme **ibping** est lié. Aussi, notez quelle carte et quel port de la machine client sont physiquement connectés au même réseau que la carte et le port qui était liés au serveur. Par exemple, si le second port de la première carte était lié au serveur, et que le port était connecté à une structure RDMA secondaire, alors, spécifier sur le client quelle carte ou port sont nécessaires pour être connecté également à cette structure secondaire. Une fois que ces choses sont connues, exécutez le programme **ibping** en tant que client et connectez-vous au serveur en utilisant soit le LID ou le GUID du port trouvé sur le serveur comme étant l'adresse à laquelle se connecter. Exemple :

```
~]$ ibping -c 10000 -f -C mlx4_0 -P 1 -L 2
--- rdma-host.example.com.(none) (Lid 2) ibping statistics ---
10000 packets transmitted, 10000 received, 0% packet loss, time 816 ms
rtt min/avg/max = 0.032/0.081/0.446 ms
```

ou

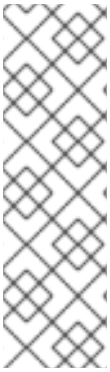
```
~]$ ibping -c 10000 -f -C mlx4_0 -P 1 -G 0xf4521403007bcba1 \
--- rdma-host.example.com.(none) (Lid 2) ibping statistics ---
10000 packets transmitted, 10000 received, 0% packet loss, time 769 ms
rtt min/avg/max = 0.027/0.076/0.278 ms
```

Cette sortie vérifie que les communications RDMA end-to-end fonctionnent pour les applications d'espace utilisateur.

L'erreur suivante peut être vue :

```
~]$ ibv_devinfo
libibverbs: Warning: no userspace device-specific driver found for
/sys/class/infiniband_verbs/uverbs0
No IB devices found
```

Cette erreur indique que la bibliothèque de l'espace utilisateur nécessaire n'est pas installée. L'administrateur doit installer une des bibliothèques répertoriées de l'espace utilisateur (selon leur matériel) dans la section Section 9.2, « Paquets de logiciels relatifs à InfiniBand et à RDMA ». En de rares occasions, cela peut se produire si un utilisateur installe le mauvais type d'arch pour le pilote ou pour **libibverbs**. Par exemple, si **libibverbs** est d'arch **x86_64** et que **libmlx4** est installé, mais est de type **i686**, alors cette erreur peut se produire.



NOTE

De nombreux exemples d'applications préfèrent utiliser des adresses ou des noms d'hôte au lieu de LID pour ouvrir une communication entre le serveur et le client. Pour ces applications, il faut définir les IPoIB avant de tenter de tester des communications RDMA end-to-end. L'application **ibping** est inhabituelle car elle accepte des LID simples comme forme d'adressage, et cela lui permet de constituer un test simple qui élimine les éventuels problèmes d'adressage d'IPoIB des scénarios de test et nous donne donc une vue plus isolée pour savoir si oui ou non, les communications RDMA simples fonctionnement.

9.6. CONFIGURATION D'IPOIB

9.6.1. Comprendre le rôle d'IPoIB

Comme mentionné dans Section 1.2, « Réseaux IP versus Réseaux non-IP », la plupart des réseaux sont des réseaux **IP**. InfiniBand n'en est pas un. Le rôle d'IPoIB est de fournir une couche d'émulation de réseau **IP** sur les réseaux RDMA InfiniBand. Cela permet à des applications existantes d'exécuter sur InfiniBand réseaux non modifiés. Toutefois, les performances de ces applications sont considérablement inférieures par rapport à la possibilité que l'application ait été écrite pour utiliser la communication RDMA nativement. Étant donné que la plupart des réseaux InfiniBand comprennent un ensemble d'applications qui doivent vraiment essayer d'obtenir toute la performance possible du réseau, et d'autres applications pour lesquelles un taux de rendement moindre est acceptable, si cela signifie que l'application ne doit pas être modifiée pour utiliser les communications RDMA. IPoIB est là pour permettre aux applications moins critiques d'exécuter sur le réseau telles qu'elles.

Comme les réseaux iWARP et RoCE/IBoE sont en fait des réseaux **IP** avec des couches RDMA au dessus de leur couche liaison **IP**, ils n'ont pas besoin d'IPoIB. De ce fait, le noyau refusera de créer des périphériques IPoIB sur les périphériques RDMA iWARP ou RoCE/IBoE.

9.6.2. Comprendre les modes de communication IPoIB

Les périphériques IPoIB peuvent être configurés pour exécuter en mode datagramme ou connecté. La différence est dans quel type de paire de file d'attentes la couche IPoIB tente d'ouvrir avec la machine à l'autre bout de la communication. Pour le mode datagramme, une paire de file d'attentes peu fiables, déconnectées, est ouverte. Pour le mode connecté, une paire de files d'attentes fiables, connectées est ouverte.

En mode datagramme, le type de paire de file d'attentes peu fiables, déconnectées n'autorise pas les paquets qui sont plus volumineux que le MTU de la couche-liaison d'InfiniBand. La couche IPoIB ajoute un en-tête IPoIB de 4 octets au dessus du paquet **IP** transmis. Ainsi, le MTU IPoIB doit être de 4 octets de moins que le MTU de couche liaison d'InfiniBand. Comme 2048 est un MTU de couche liaison d'InfiniBand commun, le MTU de périphérique commun d'IPoIB MTU en mode datagramme est de 2044

Lorsque vous utilisez le mode connecté, la paire de file d'attentes fiables et connectées autorise des messages qui sont plus volumineux que le MTU de couche liaison d'InfiniBand et l'adaptateur de l'hôte gère une segmentation de paquets et leur réassemblage à chaque extrémité. En conséquence, il n'y a aucune limite de taille imposée sur la taille des messages IPoIB, qui peuvent être envoyés par les adaptateurs InfiniBand en mode connecté. Cependant, il y a toujours la limitation qu'un paquet **IP** doit avoir un champ de taille de 16 bits seulement, et qu'il est donc limité à **65535** comme nombre d'octets maximum. Le MTU maximal autorisé est en fait plus petit que cela, parce que nous devons tenir compte des différents en-têtes TCP/IP qui doivent également correspondre à cette taille. En conséquence, le MTU IPoIB en mode connecté est plafonné à **65520** afin de s'assurer il y a suffisamment de place pour tous les en-têtes **TCP** nécessaires.

L'option mode connecté a généralement des performances supérieures, mais il consomme aussi plus de mémoire du noyau. Comme la plupart des systèmes s'intéressent davantage à la performances qu'à la consommation de mémoire, le mode connecté est le mode le plus couramment utilisé.

Toutefois, si un système est configuré en mode connecté, il doit encore envoyer le trafic multidiffusion en mode datagramme (la structure et les commutateurs InfiniBand ne peut pas faire passer le trafic multidiffusion en mode connecté). De plus, il reviendra en mode datagramme lorsqu'il communiquera avec tous les hôtes non configurés en mode connecté. Les administrateurs doivent être conscients que s'ils ont l'intention d'exécuter des programmes qui envoient des données multidiffusion, et que ces programmes tentent d'envoyer des données multidiffusion à hauteur de la valeur maximale MTU sur l'interface, il faudra configurer l'interface pour l'opération de datagramme ou trouver un moyen de configurer l'application multidiffusion pour plafonner la taille de leurs paquets envoyés à une taille qui s'adaptera à celle des paquets datagramme.

9.6.3. Comprendre les adresses de matériel IPoIB

Les périphériques IPoIB ont des adresses de 20 octets. L'utilitaire déprécié **ifconfig** n'est pas en mesure de lire l'ensemble des 20 octets et n'est pas assez fiable pour chercher les adresses correctes de matériel pour un périphérique IPoIB. Les utilitaires **ip** du package iproute fonctionnenet correctement.

Les 4 premiers octets de l'adresse matérielle d'IPoIB correspondent aux drapeaux et au numéro de paire de file d'attentes. Les prochains 8 octets représentent le préfixe de sous-réseau. Quand le périphérique IPoIB est tout d'abord créé, il aura le préfixe de sous-réseau par défaut de **`0xfe:80:00:00:00:00:00:00`**. Le périphérique utilise le préfixe de sous-réseau par défaut (0xfe80000000000000) jusqu'à ce qu'il entre en contact avec le gestionnaire de sous-réseau, et à moment là, il se réinitialisera le préfixe de sous-réseau pour correspondre à la valeur de configuration devisée par le gestionnaire de sous-réseau. Les 8 octets de la fin correspondent à l'adresse GUID du port InfiniBand auquel le périphérique IPoIB est attaché. Comme les 4 premiers octets et les 8 octets suivants peuvent changer de temps en temps, ils ne sont pas utilisés ou mis en correspondance lorsque vous spécifiez l'adresse matérielle d'une interface IPoIB. La section Section 9.3.3, « Utilisation de 70-persistent-ipoib.rules » explique comment dériver l'adresse en laissant les 12 premiers octets en dehors du champ **`ATTR {address}`** dans le fichier de règles **udev** pour que la correspondance au périphérique soit fiable. Lorsque vous configurez des interfaces IPoIB, le champ HWADDR du fichier de configuration peut contenir les 20 octets, mais seuls les 8 derniers octets sont effectivement utilisés pour la correspondance et pour trouver le matériel spécifié par un fichier de configuration. Toutefois, si le **`TYPE=InfiniBand`** n'est pas correctement épelé dans le fichier de configuration de périphérique, et que **ifup-ib** n'est pas le script réel utilisé pour faire apparaître l'interface IPoIB, alors, une erreur expliquant que le système est incapable de trouver le matériel spécifié par la configuration surgira. Pour les interfaces IPoIB, le champ **`TYPE=`** du fichier de configuration doit être **`InfiniBand`** ou **`infiniband`** (l'entrée est sensible à la casse, mais les scripts accepteront les deux façons d'épeler).

9.6.4. Comprendre les sous réseaux P_Key d'InfiniBand

La structure InfiniBand peut être logiquement segmentée en sous-réseaux virtuels avec différents sous-réseaux **`P_Key`**. Ceci est similaire à l'utilisation de réseaux locaux virtuels sur des interfaces Ethernet. Tous les commutateurs et les hôtes doivent être membres du sous-réseau **`P_Key`** qui correspond à la valeur par défaut, mais les administrateurs peuvent créer des sous-réseaux supplémentaires et limiter des membres de ces sous-réseaux à des sous-réseaux d'hôtes ou de commutateurs de la structure. Un sous-réseau de **`P_Key`** doit être défini par le gestionnaire de sous-réseaux avant qu'un hôte puisse l'utiliser. Voir la section Section 9.4.4, « Création d'une définition P_Key » pour plus d'informations sur la façon de définir un sous-réseau **`P_Key`** en utilisant le gestionnaire de sous-réseaux **opensm**. Pour les interfaces IPoIB, lorsqu'un sous-réseau **`P_Key`** a été créé, nous pouvons créer des fichiers de configuration IPoIB supplémentaires spécifiquement pour ces sous-réseaux **`P_Key`**. Tout comme les interfaces VLAN sur les périphériques Ethernet, chaque interface de IPoIB se comporte comme si elle était sur une structure complètement différentes des autres interfaces IPoIB partageant la même liaison, mais ayant des valeurs **`P_Key`** différentes.

Il y a des prescriptions particulières pour les noms des interfaces **`P_Key`** d'IPoIB. Toutes les **`P_Key`** d'IPiIB varient entre **`0x0000`** et **`0x7fff`** et le caractère étendu (high bit), **`0x8000`** indique que l'appartenance à un **`P_Key`** est membre à part entière et qu'il ne s'agit pas d'une adhésion partielle. Le pilote IPoIB du noyau Linux ne supporte que des membres à part entière dans les sous-réseaux **`P_Key`**, donc, pour n'importe quel sous-réseau sur lequel Linux peut établir une connexion, la valeur high bit de **`P_Key`** sera toujours définie. Cela signifie que si un ordinateur Linux rejoint**`P_Key 0x0002`**, son nombre **`P_Key`** une fois rejoint, sera **`0x8002`**, indiquant que nous sommes membres à part entière de **`P_Key 0x0002`**. Pour cette raison, lorsque vous créez une définition de **`P_Key`** dans un fichier **opensm `partitions.conf`** comme décrit dans la section Section 9.4.4, « Création d'une définition P_Key », il est faut spécifier une valeur **`P_Key`** sans **`0x8000`**, mais lors de la définition des interfaces **`P_Key`** d'IPoIB sur les clients Linux, ajoutez la valeur de **`0x8000`** à la valeur de base **`P_Key`**.

9.7. CONFIGURER INFINIBAND PAR L'INTERFACE TEXTE UTILISATEUR, NMTUI

L'outil d'interface utilisateur de texte **nmtui** peut être utilisé pour configurer InfiniBand dans une fenêtre de terminal. Exécutez la commande suivante pour démarrer l'outil :

```
~]$ nmtui
```

L'interface utilisateur texte apparaîtra. Tout commande non valide affichera un message d'utilisation.

Pour naviguer, utiliser les flèches ou appuyer sur **Tab** pour continuer et appuyer sur la combinaision de touches **Maj**+**Tab** pour revenir aux options. Appuyer sur la touche **Entrée** pour sélectionner une option. La barre **Espace** active/désactive le statut d'un case à cocher.

À partir du menu de démarrage, sélectionner **Modifier une connexion**. Sélectionner **Ajouter**, l'écran **Nouvelle connexion** apparaîtra.

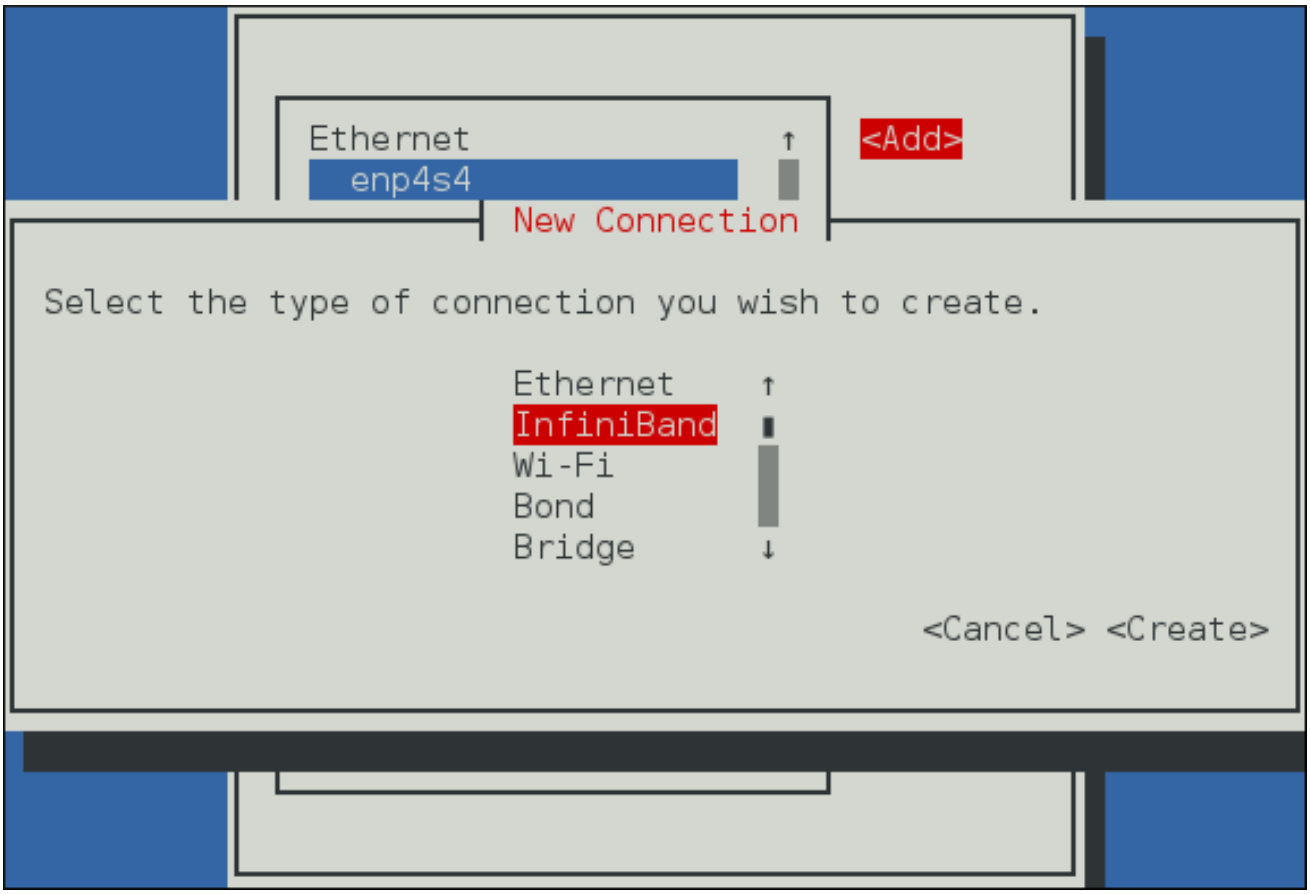


Figure 9.1. Pour que l'interface utilisateur texte du NetworkManager ajoute un menu de connexion InfiniBand

Sélectionner **InfiniBand**, l'écran **Modifier connexion** apparaîtra. Suivre les invites de l'écran pour terminer la configuration.

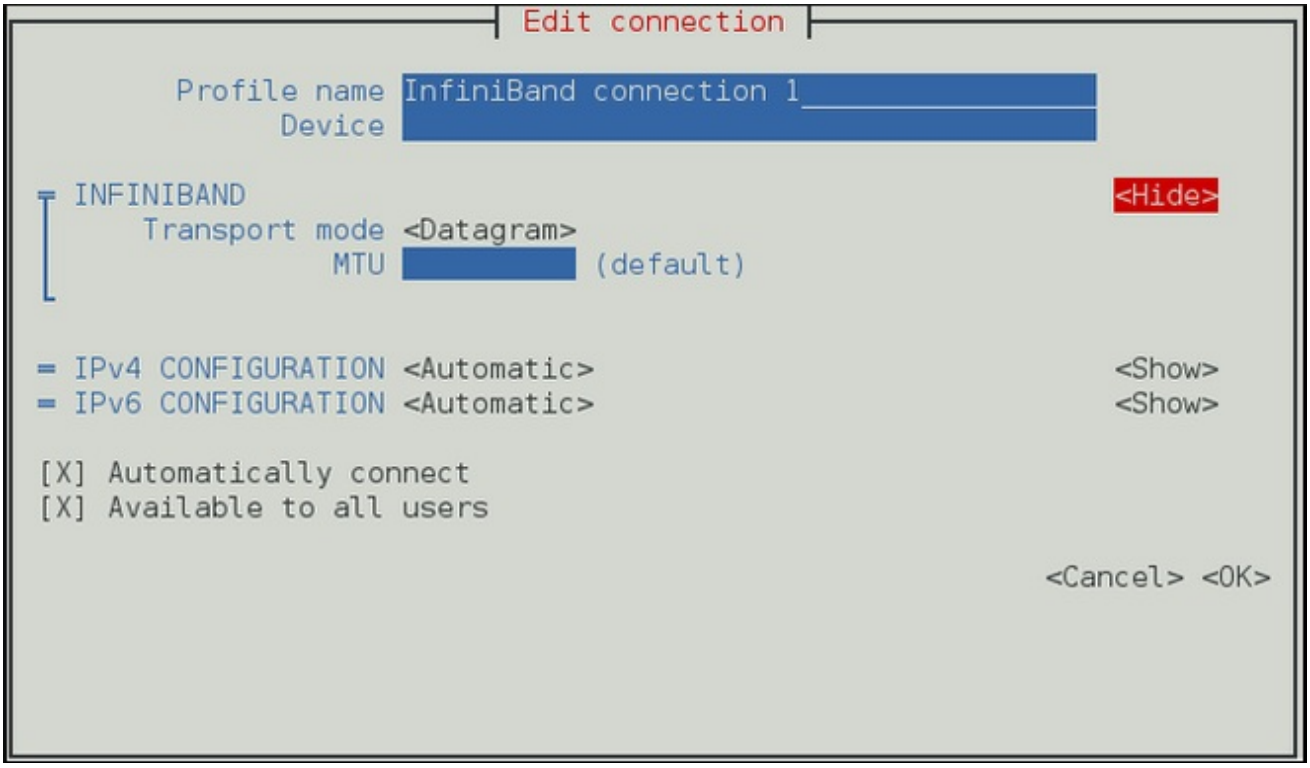


Figure 9.2. Pour que l'interface utilisateur texte du NetworkManager configure un menu de connexion InfiniBand

Voir Section 9.11.1, « Configurer l'onglet InfiniBand » pour obtenir des définitions des termes InfiniBand.

Voir Section 1.5, « Configuration réseau utilisant une interface utilisateur texte (nmtui) » pour obtenir des informations sur la façon d'installer **nmtui**.

9.8. CONFIGURER IPOIB PAR L'OUTILS DE LIGNE DE COMMANDES, NMCLI

Commencez par déterminer s'il est nécessaire de renommer le ou les périphériques IPoIB par défaut, et dans l'affirmative, suivez les instructions dans la section Section 9.3.3, « Utilisation de 70-persistent-ipoib.rules » pour renommer les périphériques en utilisant les règles de renommage de **udev**. Les utilisateurs peuvent forcer les interfaces IPoIB à être renommées sans effectuer de redémarrage, en enlevant le module du noyau **`ib_ipoib`** et en le rechargeant ensuite comme suit :

```
~]$ rmmod ib_ipoib
~]$ modprobe ib_ipoib
```

Une fois que les périphériques ont le nom voulu, utiliser l'outil **nmcli** pour créer le ou les interface(s) IPoIB comme suit :

```
~]$ nmcli con add type infiniband con-name mlx4_ib0 ifname mlx4_ib0
transport-mode connected mtu 65520
Connection 'mlx4_ib0' (8029a0d7-8b05-49ff-a826-2a6d722025cc) successfully
added.
~]$ nmcli con edit mlx4_ib0

===| nmcli interactive connection editor |===

Editing existing 'infiniband' connection: 'mlx4_ib0'
```

```
Type 'help' or '?' for available commands.
Type 'describe [>setting<.>prop<]' for detailed property description.

You may edit the following settings: connection, infiniband, ipv4, ipv6
nmcli> set infiniband.mac-address
80:00:02:00:fe:80:00:00:00:00:00:00:f4:52:14:03:00:7b:cb:a3
nmcli> save
Connection 'mlx4_ib3' (8029a0d7-8b05-49ff-a826-2a6d722025cc) successfully
updated.
nmcli> quit
```

À ce stade, une interface IPoIB nommée **mlx4_ib0** a été créée et configurée pour utiliser le mode connecté, avec le mode MTU connecté maximal, **DHCP** pour **IPv4** et **IPv6**. Si vous utilisez des interfaces IPoIB pour le trafic dans le cluster et une interface Ethernet pour les communications hors du cluster, vous devrez sans doute désactiver les itinéraires par défaut et n'importe quel serveur de nom par défaut sur les interfaces IPoIB. Cela peut être effectué ainsi :

```
~]$ nmcli con edit mlx4_ib0

===| nmcli interactive connection editor |===

Editing existing 'infiniband' connection: 'mlx4_ib0'

Type 'help' or '?' for available commands.
Type 'describe [>setting<.>prop<]' for detailed property description.

You may edit the following settings: connection, infiniband, ipv4, ipv6
nmcli> set ipv4.ignore-auto-dns yes
nmcli> set ipv4.ignore-auto-routes yes
nmcli> set ipv4.never-default true
nmcli> set ipv6.ignore-auto-dns yes
nmcli> set ipv6.ignore-auto-routes yes
nmcli> set ipv6.never-default true
nmcli> save
Connection 'mlx4_ib0' (8029a0d7-8b05-49ff-a826-2a6d722025cc) successfully
updated.
nmcli> quit
```

Si une interface **P_Key** est requise, créez-en une avec **nmcli**, comme suit :

```
~]$ nmcli con add type infiniband con-name mlx4_ib0.8002 ifname
mlx4_ib0.8002 parent mlx4_ib0 p-key 0x8002
Connection 'mlx4_ib0.8002' (4a9f5509-7bd9-4e89-87e9-77751a1c54b4)
successfully added.
~]$ nmcli con modify mlx4_ib0.8002 infiniband.mtu 65520
infiniband.transport-mode connected ipv4.ignore-auto-dns yes ipv4.ignore-
auto-routes yes ipv4.never-default true ipv6.ignore-auto-dns yes
ipv6.ignore-auto-routes yes ipv6.never-default true
```

9.9. CONFIGUREZ IPOIB PAR INTERFACE EN LIGNE DE COMMANDES

Commencez par déterminer s'il est nécessaire de renommer le ou les périphériques IPoIB par défaut, et dans l'affirmative, suivez les instructions dans la section Section 9.3.3, « Utilisation de 70-persistent-ipoib.rules » pour renommer les périphériques en utilisant les règles de renommage de **udev**. Les

utilisateurs peuvent forcer les interfaces IPoIB à être renommées sans effectuer de redémarrage, en enlevant le module du noyau **ib_ipoib** et en le rechargeant ensuite comme suit :

```
~]$ rmmod ib_ipoib
~]$ modprobe ib_ipoib
```

Une fois que les périphériques auront reçu le nom qui convient, les administrateurs peuvent créer des fichiers **ifcfg** avec leur éditeur préféré pour contrôler les périphériques. Un fichier de configuration IPoIB typique d'adressage static **IPv4** ressemblera à ce qui suit :

```
~]$ more ifcfg-mlx4_ib0
DEVICE=mlx4_ib0
TYPE=InfiniBand
ONBOOT=yes
HWADDR=80:00:00:4c:fe:80:00:00:00:00:00:00:f4:52:14:03:00:7b:cb:a1
BOOTPROTO=none
IPADDR=172.31.0.254
PREFIX=24
NETWORK=172.31.0.0
BROADCAST=172.31.0.255
IPV4_FAILURE_FATAL=yes
IPV6INIT=no
MTU=65520
CONNECTED_MODE=yes
NAME=mlx4_ib0
```

Le champ DEVICE doit correspondre au nom personnalisé créé dans n'importe quelle règle de renommage d'**udev**. L'entrée de nom NAME n'a pas besoin de correspondre au nom du périphérique. Si l'éditeur de connexion du GUI est démarré, le champ NAME est ce qui sert à présenter un nom pour cette connexion à l'utilisateur. Le champ TYPE doit être InfiniBand pour que les options InfiniBand soient traitées correctement. CONNECTED_MODE est **Oui** ou **non** : **oui** en mode connecté et **non** en mode datagramme pour les communications (voir section Section 9.6.2, « Comprendre les modes de communication IPoIB »).

Pour les interfaces **P_Key**, voici un fichier de configuration typique :

```
~]$ more ifcfg-mlx4_ib0.8002
DEVICE=mlx4_ib0.8002
PHYSDEV=mlx4_ib0
PKEY=yes
PKEY_ID=2
TYPE=InfiniBand
ONBOOT=yes
HWADDR=80:00:00:4c:fe:80:00:00:00:00:00:00:f4:52:14:03:00:7b:cb:a1
BOOTPROTO=none
IPADDR=172.31.2.254
PREFIX=24
NETWORK=172.31.2.0
BROADCAST=172.31.2.255
IPV4_FAILURE_FATAL=yes
IPV6INIT=no
MTU=65520
CONNECTED_MODE=yes
NAME=mlx4_ib0.8002
```

Pour tous les fichiers d'interface **P_Key**, la directive PHYSDEV est nécessaire et correspond au nom du périphérique parent. La directive PKEY doit être définie sur **oui**, et **PKEY_ID** doit correspondre au numéro de l'interface (avec ou sans **0x8000** ajouté). Le nom du périphérique, toutefois, doit correspondre à la représentation hexadécimal à quatre chiffres de **PKEY_ID** combinée à **0x8000** utilisant l'opérateur logique OR comme suit :

```
NAME=${PHYSDEV}.$((0x8000 | $PKEY_ID))
```

Par défaut, le **PKEY_ID** du fichier est traité comme un nombre décimal et converti en hexadécimal, puis est combiné par l'opérateur logique OR avec **0x8000** pour obtenir le nom du périphérique qui convient, mais les utilisateurs peuvent spécifier le **PKEY_ID** en hexadécimal, en ajoutant le préfixe standard **0x** au nombre.

9.10. TEST D'UN RÉSEAU RDMA UNE FOIS QU'IPOIB EST CONFIGURÉ.

Une fois qu'IPoIB est configuré, il est possible d'utiliser des adresses **IP** pour spécifier les périphériques RDMA. En raison de l'ubiquité de l'utilisation des adresses **IP** et des noms d'hôtes pour spécifier des machines, la plupart des applications de RDMA l'utilisent comme leur propre façon, ou dans certains cas, pour spécifier des machines distantes ou des périphériques locaux auxquels se connecter.

Pour tester la fonctionnalité de la couche IPoIB, il est possible d'utiliser n'importe quel outil de testing de réseau **IP** et de fournir l'adresse **IP** des périphériques IPoIB à tester. Ainsi, la commande ping entre les adresses **IP** des périphériques IPoIB doit fonctionner.

Il y a deux packages diffférents existants pour les tests de performance RDMA dans Red Hat Enterprise Linux, qperf et perftest. L'un d'entre eux peut être utilisé pour tester davantage la performance d'un réseau RDMA.

Toutefois, lorsque vous utilisez une des applications qui font partie du package perftest, ou en utilisant l'application **qperf**, il y une note spéciale sur la résolution d'adresse. Même si l'hôte distant est spécifié à l'aide d'un nom d'hôte ou d'une adresse **IP** du périphérique IPoIB, l'application de test est autorisée à se connecter via une interface RDMA différente. C'est parce que le processus de conversion du nom d'hôte ou d'adresse **IP** à une adresse RDMA permet à n'importe quelle paire d'adresse RDMA valides entre les deux machines d'être utilisées. S'il existe plusieurs moyens pour le client de se connecter au serveur, alors les programmes peuvent choisir d'utiliser un autre chemin, s'il y a un problème avec le chemin d'accès spécifié. Par exemple, s'il y a deux ports sur chaque machine connectée au même sous-réseau InfiniBand, et une adresse **IP** pour pour le second port sur chaque machine, il est probable que le logiciel trouve que le premier port sur chaque machine corresponde à une méthode de connexion valide et il les utilisera à la place. Dans ce cas, des options de ligne de commandess pour tous les programmes perftest peuvent être utilisées pour leur dire à quelle carte et à quel port se relier, comme cela avait été fait avec **ibping** dans Section 9.5, « Testing d'anciennes opérations RDMA InfiniBand », afin de veiller à ce que les tests s'effectuent sur les ports spécifiques devant être testés. Avec **qperf**, la méthode de liaison aux ports est légèrement différente. Le programme **qperf** fonctionne comme serveur sur une machine à l'écoute de tous les appareils (y compris les périphériques non-RDMA). Le client peut se connecter à **qperf** à l'aide de n'importe quel nom d'hôte ou adresse **IP** valides du serveur. **Qperf** tentera d'abord d'ouvrir une connexion de données et d'exécuter les tests demandés sur le nom d'hôte ou l'adresse **IP** donnée sur la ligne de commandes du client, mais si il n'y a un problème à utiliser cette adresse, **qperf** vous renverra à tenter d'exécuter le test sur un chemin d'accès valide entre le client et le serveur. Pour cette raison, pour forcer **qperf** à tester sur un lien spécifique, utilisez les options **-loc_id** et **-rem_id** pour le client **qperf** afin de forcer le test à exécuter sur une liaison spécifique.

9.11. CONFIGUREZ IPOIB PAR INTERFACE GRAPHIQUE (GUI)

Pour configurer une connexion InfiniBand par un outil graphique, utilisez l'outil **Connections Réseau**.

Procédure 9.1. Ajout d'une nouvelle connexion InfiniBand

1. Pour utiliser l'outil graphique de **Connexions Réseau**, appuyer sur la touche **Super** pour apercevoir l'ensemble des activités, puis saisir **Connections Réseau**, et appuyer sur la touche **Saisir**. L'outil **Connections Réseau** apparaîtra.

2. Cliquer sur le bouton **Ajouter** pour ouvrir la liste de sélection. Sélectionner **InfiniBand**, puis cliquer sur**Créer**. La fenêtre **Modifier la connexion InfiniBand1** apparaîtra.

3. Dans l'onglet **InfiniBand**, sélectionner le mode de transport dans la liste déroulante que vous souhaitez utiliser pour la connexion InfiniBand

4. Saisir l'adresse MAC InfiniBand

5. Vérifier et confirmer les paramètres de configuration, puis cliquer sur le bouton **Sauvegarder**.

6. Modifier la configuration spéciale IndiniBand en consultant Section 9.11.1, « Configurer l'onglet InfiniBand ».

Procédure 9.2. Modifier une connexion InfiniBand existante

Suivre ces étapes pour modifier une connexion InfiniBand existante.

1. Appuyer sur la touche **Super**pour apercevoir l'ensemble des activités, puis saisir **Connections Réseau**, et appuyer sur la touche **Saisir**. L'outil **Connections Réseau** apparaîtra.

2. Sélectionner la connexion à modifier et cliquer sur le bouton **Modifier**.

3. Sélectionner l'onglet **Général**.

4. Configurer le nom de la connexion, le comportement auto-connect, et les paramètres disponibles.

   Il existe cinq configurations de la boîte de dialogue **Modifier** qui sont communes à tous les types de connexion. Voir l'onglet **Général** :

   - **Nom de connexion** — saisir un nom descriptif pour votre connexion de réseau. Ce nom sera utilisé pour lister cette connexion dans le menu de la fenêtre **Réseau**.

   - **Se connecter automatiquement à ce réseau quand il est disponible** — sélectionnez cette case si vous souhaitez que le **NetworkManager** s'auto-connecte à cette connexion quand elle devient disponible. Voir Section 2.5.3, « Se connecter à un réseau automatiquement » pour plus d'informations.

   - **Rendre le réseau disponible à tous les utilisateurs** — sélectionnez cette case pour créer une connexion disponible à tous les utilisateurs sur le système. Changer ce paramètre peut nécessiter des privilèges d'utilisateur root. Consulter Section 2.5.4, « Profils de connexions privées ou sur tout le système » pour obtenir plus d'informations.

   - **Se connecter automatiquement au VPN quand on utilise cette connexion** — sélectionnez cette case si vous souhaitez que le **NetworkManager** s'auto-connecte à une connexion de VPN quand il est disponible. Sélectionner le VPN à partir du menu déroulant.

- **Zone de parefeu** — sélectionnez une zone de parefeu dans le menu déroulant. Voir le guide *Red Hat Enterprise Linux 7 Security Guide* pour obtenir plus d'informations sur les Zones de parefeux.

5. Modifier la configuration spéciale IndiniBand en consultant Section 9.11.1, « Configurer l'onglet InfiniBand ».

Sauvegarder votre nouvelle connexion (ou votre connexion modifiée) et faire des configurations supplémentaires

Une fois vous aurez terminé de modifier votre connexion InfiniBand au réseau local virtuel, cliquez sur le bouton **Save** pour enregistrer votre configuration personnalisée. Si le profil est en cours d'utilisation alors qu'il est modifié, alimentez le cycle de connexion pour que **NetworkManager** applique les modifications. Si le profil est désactivé (OFF), réglez-le sur ON ou sélectionnez-le dans le menu de l'icône de connexion réseau. Voir Section 2.5.1, « Se connecter à réseau par un GUI » pour plus d'informations sur l'utilisation de votre connexion nouvelle ou modifiée.

Vous pouvez configurer davantage une connexion existante en la sélectionnant dans la fenêtre **Connexions réseau** et en cliquant sur **Modifier** pour revenir à la boîte de dialogue **Modification**.

Puis, pour configurer :

- Pour les paramètres de configuration pour la connection, cliquer sur l'onglet **IPv4 Settings** et continuer avec Section 2.5.10.4, « Configuration des paramètres IPv4 »; ou,
- Pour les paramètres de configuration pour la connexion, cliquer sur l'onglet **IPv6 Settings**et continuez avec Section 2.5.10.5, « Configurer les paramètres IPv6 ».

9.11.1. Configurer l'onglet InfiniBand

Si vous avez déjà ajouté une nouvelle connexion InfiniBand (voir Procédure 9.1, « Ajout d'une nouvelle connexion InfiniBand » pour obtenir des instructions), vous pouvez modifier l'onglet **InfiniBand** afin de définir l'interface parente et l'ID du VLAN.

Mode de transport

Les modes Datagramme ou Connnecté peuvent être sélectionnés dans la liste de menu déroulant. Sélectionner le même mode que celui que le réseau IPoIB utilise.

Adresse MAC du périphérique

L'adresse MAC du périphérique compatible InfiniBand à utiliser pour le trafic de réseau InfiniBand. Ce champ d'adresse de matériel sera pré-rempli si vous avez du matériel InfiniBand installé.

MTU

Définit en option une taille de MTU (de l'anglais Maximum Transmission Unit) à utiliser pour les paquets à envoyer sur la connexion InfiniBand.

9.12. RESSOURCES SUPPLÉMENTAIRES

Les sources d'informations suivantes vous donneront d'autres ressources sur le réseautage RDMA et InfiniBand de Red Hat Enterprise Linux 7.

9.12.1. Documentation installée

- **`/usr/share/doc/initscripts-*version*/sysconfig.txt`** — décrit les fichiers de configuration et leurs directives.

9.12.2. Documentation en ligne

https://www.kernel.org/doc/Documentation/infiniband/ipoib.txt

Une description du pilote IPoIB. Inclut des références aux divers RFC.

PARTIE III. SERVEURS

Cette partie traite de la façon de configurer des serveurs servant normalement à la gestion des réseaux.

CHAPITRE 10. SERVEURS DHCP

DHCP (Dynamic Host Configuration Protocol) est un protocole de réseau qui assigne automatiquement les informations TCP/IP à des machines clientes. Chaque client **DHCP** se connecte au serveur **DHCP** situé localement, qui renvoie la configuration de réseau (y compris l'adresse **IP**, la passerelle, et les serveurs **DNS**) de ce client.

10.1. POURQUOI UTILISER DHCP ?

DHCP est utile pour la configuration automatique des interfaces de réseaux clientes. Lorsque vous configurez le système client, vous pouvez choisir **DHCP** au lieu de spécifier une adresse **IP**, un masque de sous-réseau, une passerelle ou des serveurs **DNS**. Le client récupère cette information dans le serveur **DHCP**. **DHCP** est également utile si vous souhaitez modifier les adresses **IP** d'un grand nombre de systèmes. Au lieu de reconfigurer tous les systèmes, vous pouvez simplement modifier un fichier de configuration sur le serveur pour un nouvel ensemble d'adresses **IP**. Si les serveurs **DNS** d'une organisation change, les changements se produiront sur le serveur **DHCP**, pas sur les clients **DHCP**. Lorsque vous redémarrerez le réseau ou les clients, les changements entreront en vigueur.

Si une organisation a un serveur **DHCP** fonctionnel correctement connecté à un réseau, les utilisateurs d'ordinateurs portables ou d'autres ordinateurs mobiles peuvent déplacer ces périphériques de bureau en bureau.

Notez que les administrateurs de serveurs **DNS** et **DHCP**, ainsi que toute application de provisioning, doivent s'accorder sur le format des noms d'hôtes utilisé dans une organisation. Veuillez consulter le Section 3.1.1, « Pratiques de nommage conseillées » pour obtenir davantage d'informations sur les formats de noms d'hôtes.

10.2. CONFIGURATION D'UN SERVEUR DHCP

Le package dhcp contient un serveur ISC (*Internet Systems Consortium*) **DHCP**. Installez le package en tant qu'utilisateur **root** :

```
~]# yum install dhcp
```

En installant le paquet dhcp, vous allez créer un fichier, **/etc/dhcp/dhcpd.conf**, qui correspond tout simplement à un fichier de configuration vide. En tant qu'utilisateur **root**, exécutez la commande suivante :

```
~]# cat /etc/dhcp/dhcpd.conf
#
# DHCP Server Configuration file.
#   see /usr/share/doc/dhcp*/dhcpd.conf.example
#   see dhcpd.conf(5) man page
#uet
```

On peut trouver l'exemple de configuration dans **/usr/share/doc/dhcp-*version*;/dhcpd.conf.example**. Vous devez utiliser ce fichier pour pouvoir configurer **/etc/dhcp/dhcpd.conf**, expliqué en détails ci-dessous.

DHCP utilise également le fichier **/var/lib/dhcpd/dhcpd.leases** pour stocker la base de données d'attribution client. Voir Section 10.2.2, « Base de données d'attribution » pour plus d'informations.

10.2.1. Fichier de configuration

La première étape pour configurer un serveur **DHCP** consiste à créer le fichier de configuration qui stocke les informations réseau des clients. Utiliser ce fichier pour déclarer des options dans les systèmes clients.

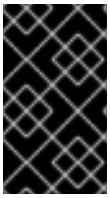
Le fichier de configuration peut contenir des tabulations ou lignes vierges complémentaires pour faciliter le formatage. Les mots-clés sont sensibles à la casse et les lignes commençant par un signe dièse (#) correspondent à des commentaires.

Il existe deux types de déclarations dans le fichier de configuration :

- Paramètres - expliquent comment effectuer une tâche, quand l'effectuer, ou quelles options de configuration donner au client.
- Déclarations - décrit la topologie du réseau, les clients, donne des adresses aux clients, ou applique un groupe de paramètres à un groupe de déclarations.

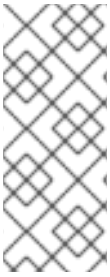
Les paramètres qui commencent avec le mot-clé option sont considérés comme des *options*. Ces options configurent les options **DHCP**, alors que les paramètres eux, configurent des valeurs qui ne sont pas facultatives ou contrôlent le comportement du serveur **DHCP**.

Les paramètres (y compris les options) déclarés avant une section entre accolades (**{ }**) sont considérés comme des paramètres globaux. Ceux-ci s'appliquent à toutes les sections se trouvant en dessous.



IMPORTANT

Si le fichier de configuration est modifié, les changements ne pourront pas prendre effet tant que le démon **DHCP** n'aura pas redémarré par la commande **`systemctl restart dhcpd`**.



NOTE

Au lieu de modifier un fichier de configuration **DHCP** et redémarrer le service à chaque fois, en utilisant la commande **`omshell`**, vous pouvez vous connecter, interroger ou modifier la configuration du serveur **DHCP** de façon interactive. Avec **`omshell`**, vous pourrez effectuer tous les changements quand le serveur est en cours d'exécution. Pour plus d'informations sur **`omshell`**, voir la page man **`omshell`**.

Dans Exemple 10.1, « Déclaration de sous-réseau », les options **`routers`**, **`subnet-mask`**, **`domain-search`**, **`domain-name-servers`**, et **`time-offset`** sont utilisées pour les énoncés d' **`hôte`** déclarés dessous.

Pour chaque sous-réseau servi, et pour chaque sous-réseau auquel le serveur **DHCP** se connecte, il doit y avoir une déclaration de **`sous-réseau`**, qui indique au démon **DHCP** comment reconnaître qu'une adresse est sur ce sous-réseau. Une déclaration de **`sous-réseau`** est exigée pour chaque sous-réseau, même si aucune adresse ne doit être allouée de façon dynamique sur ce sous-réseau.

Dans cet exemple, il y a des options globales pour chaque client **DHCP** dans le sous-réseau, et une **`plage`** est annoncée. Les clients reçoivent une adresse **`IP`** incluse dans l'**`intervalle`** indiqué.

Exemple 10.1. Déclaration de sous-réseau

```
subnet 192.168.1.0 netmask 255.255.255.0 {
        option routers                  192.168.1.254;
        option subnet-mask              255.255.255.0;
        option domain-search              "example.com";
        option domain-name-servers       192.168.1.1;
        option time-offset              -18000;     # Eastern Standard
Time
 range 192.168.1.10 192.168.1.100;
}
```

Pour configurer un serveur **DHCP** qui alloue une adresse **IP** à un système se trouvant dans un sous-réseau, modifier les valeurs de l'exemple de Exemple 10.2, « La paramètre « range » de plage ». Il déclare une durée d'allocation par défaut, une durée d'allocation maximale, et des valeurs de configuration de réseau pour les clients. Cet exemple alloue des adresses **IP** sur les **plages de valeurs 192.168.1.10** et **192.168.1.100** aux systèmes clients.

Exemple 10.2. La paramètre « range » de plage

```
default-lease-time 600;
max-lease-time 7200;
option subnet-mask 255.255.255.0;
option broadcast-address 192.168.1.255;
option routers 192.168.1.254;
option domain-name-servers 192.168.1.1, 192.168.1.2;
option domain-search "example.com";
subnet 192.168.1.0 netmask 255.255.255.0 {
   range 192.168.1.10 192.168.1.100;
}
```

Pour assigner une adresse **IP** à un client sur la base d'une adresse MAC de la carte d'interface de réseau, utiliser le paramètre **hardware ethernet** qui se trouve dans la déclaration de l' **hôte**. Comme le montre Exemple 10.3, « Adresse IP statique utilisant DHCP », la déclaration **host apex** spécifie que la carte d'interface de réseau ayant pour adresse MAC **00:A0:78:8E:9E:AA** reçoit toujours l'adresse **IP 192.168.1.4**.

Notez que vous pouvez également utiliser le paramètre **host-name** pour assigner un nom d'hôte au client.

Exemple 10.3. Adresse IP statique utilisant DHCP

```
host apex {
   option host-name "apex.example.com";
   hardware ethernet 00:A0:78:8E:9E:AA;
   fixed-address 192.168.1.4;
}
```

Red Hat Enterprise Linux 7 prend en charge l'assignation des adresses **IP** statiques aux interfaces InfiniBand IPoIB. Cependant, comme ces interfaces n'ont pas d'adresse Ethernet de matériel normale, on doit utiliser une méthode différente pour spécifier un identificateur unique pour l'interface IPoIB. On utilise normalement l'option **dhcp-client-identifier =** pour spécifier le champ **dhcp-client-**

identificateur de l'interface IPoIB. La construction d'hôte de serveur **DHCP** ne supporte pas plus d'une entrée matériel Ethernet et une entrée de **dhcp-client-identifier** par hôte. Toutefois, il peut y avoir plus d'une entrée d'adresse fixe et le serveur **DHCP** répondra automatiquement avec une adresse qui est appropriée au réseau sur lequel la requête **DHCP** a été reçue.

Exemple 10.4. Adresse IP statique utilisant DHCP sur interfaces multiples

Si une machine a une configuration complexe, par exemple deux interfaces InfiniBand et des interfaces **P_Key** sur chaque interface physique, ainsi qu'une connexion Ethernet, la construction statique **IP** peut être utilisée pour cette configuration :

```
Host apex.0 {
    option host-name “apex.example.com”;
    hardware ethernet 00:A0:78:8E:9E:AA;
    option dhcp-client-
identifier=ff:00:00:00:00:00:02:00:00:02:c9:00:00:02:c9:03:00:31:7b:11;
    fixed-address 172.31.0.50,172.31.2.50,172.31.1.50,172.31.3.50;
}

host apex.1 {
    option host-name “apex.example.com”;
    hardware ethernet 00:A0:78:8E:9E:AB;
    option dhcp-client-
identifier=ff:00:00:00:00:00:02:00:00:02:c9:00:00:02:c9:03:00:31:7b:12;
    fixed-address 172.31.0.50,172.31.2.50,172.31.1.50,172.31.3.50;
}
```

Afin de trouver le bon **dhcp-client-identifier** pour votre périphérique, vous pouvez normalement utiliser le préfixe **ff:00:00:00:00:00:02:00:00:02:c9:00** et y ajouter les 8 derniers octets de l'interface IPoIB (qui correspond également au GUID de 8 octets du port InfiniBand du port InfinisBand sur lequel se trouve l'interface IPoIB). Sur certains contrôleurs, ce préfixe n'est pas correct. Dans un tel cas, nous conseillons d'utiliser **tcpdump** sur le serveur **DHCP** pour capturer la demande IPoIB **DHCP** entrante et collecter le **dhcp-client-identifier** qui convient pour cette capture. Par exemple :

```
]$ tcpdump -vv -i mlx4_ib0
tcpdump: listening on mlx4_ib0, link-type LINUX_SLL (Linux cooked),
capture size 65535 bytes
23:42:44.131447 IP (tos 0x10, ttl 128, id 0, offset 0, flags [none],
proto UDP (17), length 328)
    0.0.0.0.bootpc > 255.255.255.255.bootps: [udp sum ok] BOOTP/DHCP,
Request, length 300, htype 32, hlen 0, xid 0x975cb024, Flags [Broadcast]
(0x8000)
          Vendor-rfc1048 Extensions
            Magic Cookie 0x63825363
            DHCP-Message Option 53, length 1: Discover
            Hostname Option 12, length 10: "rdma-qe-03"
            Parameter-Request Option 55, length 18:
              Subnet-Mask, BR, Time-Zone, Classless-Static-Route
              Domain-Name, Domain-Name-Server, Hostname, YD
              YS, NTP, MTU, Option 119
              Default-Gateway, Classless-Static-Route, Classless-
Static-Route-Microsoft, Static-Route
```

```
              Option 252, NTP
            Client-ID Option 61, length 20: hardware-type 255,
00:00:00:00:00:02:00:00:02:c9:00:00:02:c9:02:00:21:ac:c1
```

L'image ci-dessus montre le champ identifiant le client. Le type de matériel **255** correspond aux initiales **ff:** de l'ID, le reste de l'ID est ensuite cité exactement comme il doit apparaître dans le fichier de configuration de **DHCP**.

Tous les sous-réseaux qui partagent le même réseau physique doivent être déclarés dans une déclaration de réseau partagé (**shared-network**), comme indiqué dans Exemple 10.5, « Déclaration de réseau partagé ». Les paramètres qui se trouvent dans le **shared-network**, mais en dehors des déclarations de sous-réseau clos, sont considérés comme paramètres globaux. Le nom attribué au **shared-network** doit être un titre descriptif pour le réseau, comme l'utilisation du titre « -test-lab » qui décrit tous les sous-réseaux dans un environnement de test.

Exemple 10.5. Déclaration de réseau partagé

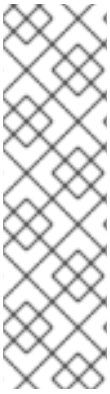
```
shared-network name {
    option domain-search           "test.redhat.com";
    option domain-name-servers     ns1.redhat.com, ns2.redhat.com;
    option routers                 192.168.0.254;
    #more parameters for EXAMPLE shared-network
    subnet 192.168.1.0 netmask 255.255.252.0 {
        #parameters for subnet
        range 192.168.1.1 192.168.1.254;
    }
    subnet 192.168.2.0 netmask 255.255.252.0 {
        #parameters for subnet
        range 192.168.2.1 192.168.2.254;
    }
}
```

Comme démontré dans Exemple 10.6, « Déclaration de groupe », la déclaration de **groupe** est utilisée pour appliquer les paramètres globaux à un groupe de déclarations. Ainsi, les réseaux partagés, les sous-réseaux, et les hôtes peuvent être groupés.

Exemple 10.6. Déclaration de groupe

```
group {
   option routers                192.168.1.254;
   option subnet-mask            255.255.255.0;
   option domain-search            "example.com";
   option domain-name-servers     192.168.1.1;
   option time-offset            -18000;     # Eastern Standard Time
   host apex {
      option host-name "apex.example.com";
      hardware ethernet 00:A0:78:8E:9E:AA;
      fixed-address 192.168.1.4;
   }
   host raleigh {
      option host-name "raleigh.example.com";
      hardware ethernet 00:A1:DD:74:C3:F2;
```

```
        fixed-address 192.168.1.6;
    }
}
```



NOTE

Vous pouvez utiliser le fichier de configuration donné dans l'exemple pour commencer, et y ajouter des options de configuration personnalisées. Pour copier ce fichier dans l'emplacement qui convient, utiliser la commande suivante en tant qu'utilisateur **root** :

```
~]# cp /usr/share/doc/dhcp-version_number/dhcpd.conf.example
/etc/dhcp/dhcpd.conf
```

... avec *version_number* comme numéro de version **DHCP**.

Pour obtenir une liste complète des énoncés d'options, et de ce qu'ils font, voir la page man **dhcp-options(5)**.

10.2.2. Base de données d'attribution

Sur le serveur **DHCP**, le fichier **/var/lib/dhcpd/dhcpd.leases** stocke la base de données de l'allocation du client **DHCP**. Ne pas modifiez ce fichier. Les informations d'allocation **DHCP** pour chaque adresse **IP** récemment attribuée sont automatiquement stockées dans la base de donnée d'allocation. L'information comprend la durée de l'allocation, à laquelle l'adresse **IP** a été assignée, les dates de début et de fin de l'allocation et l'adresse MAC de la carte d'interface réseau qui a été utilisée pour récupérer l'allocation.

Toutes les heures de la base de données d'attribution sont des heures "Coordinated Universal Time" (UTC) et non pas des heures locales.

La base de données de l'allocation est recréé de temps en temps pour qu'elle n'atteigne pas des proportions trop élevées. Tout d'abord, toutes la allocations connues sont enregistrés dans une base de données d'allocations temporaires. Le fichier **dhcpd.leases** est rebaptisé **dhcpd.leases~** et la base de données d'allocations temporaires est inscrite dans **dhcpd.leases**.

Le démon **DHCP** peut être terminé ou le système peut se bloquer après que la base de données d'allocations a été renommée dans le fichier de sauvegarde, mais avant que le nouveau fichier n'ait été écrit. Dans ce cas, le fichier **dhcpd.leases** n'existe pas, mais il est nécessaire de démarrer le service. Ne créez de nouveau fichier d'allocation. Si vous le faites, toutes les anciennes allocations seront perdues, ce qui provoquera de nombreux problèmes. La meilleure solution consiste à renommer le fichier de sauvegarde **dhcpd.leases~** à **dhcpd.leases**, et à démarrez le démon.

10.2.3. Lancement et interruption du serveur

IMPORTANT

Lorsque le serveur **DHCP** est démarré pour la première fois, il échoue à moins que le fichier **dhcpd.leases** existe. Vous pouvez utiliser la commande **touch /var/lib/dhcpd/dhcpd.leases** pour créer le fichier s'il n'existe pas. Si le même serveur exécute également BIND en tant que serveur **DNS**, cette étape n'est pas nécessaire, car démarrer le service **named** entraînera une recherche automatique du fichier **dhcpd.leases**.

Ne pas créer de nouveau fichier d'allocation sur un système qui était en cours d'exécution auparavant. Si vous le faîtes, toutes la anciennes allocations seront perdues, ce qui entraîne de nombreux problèmes. La solution est de renommer le fichier de sauvegarde **dhcpd.leases~** en **dhcpd.leases**, puis démarrer le démon.

Pour démarrer le service **DHCP**, utiliser la commande suivante :

```
systemctl start dhcpd.service
```

Pour arrêter le serveur **DHCP**, saisir :

```
systemctl stop dhcpd.service
```

Par défaut, le service **DHCP** ne démarre pas à l'amorçage. Pour obtenir des informations sur la façon de configurer le démon pour qu'il démarre automatiquement à l'amorçage, consulter le guide *Red Hat Enterprise Linux 7 System Administrator's Guide*.

Si plus d'une interface réseau est reliée au système, mais que le serveur **DHCP** doit seulement écouter les requêtes **DHCP** sur une des interfaces, configurez le serveur **DHCP** pour écouter uniquement sur ce périphérique. Le démon **DHCP** écoute uniquement sur les interfaces pour lesquelles il pourra trouver une déclaration de sous-réseau dans le fichier **/etc/dhcp/dhcpd.conf**.

Cela est utile si vous avez un ordinateur protégé par un pare-feu et doté de deux cartes réseau. L'une d'elles peut être configurée comme client **DHCP** pour récupérer une adresse **IP** d'internet. L'autre carte de réseau peut servir de serveur **DHCP** pour le réseau interne se trouvant derrière le pare-feu. En ne spécifiant que la carte réseau connectée au réseau interne, votre système sera plus sûr puisque les utilisateurs ne pourront pas se connecter au démon par le biais de l'internet.

Pour spécifier les options de ligne de commandes, copier et éditer le fichier **dhcpd.service** en tant qu'utilisateur **root**. Par exemple :

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

Modifier la ligne sous la section [Service]:

```
ExecStart=/usr/sbin/dhcpd -f -cf /etc/dhcp/dhcpd.conf -user dhcpd -group
dhcpd --no-pid your_interface_name(s)
```

. Puis, en tant qu'utilisateur **root**, démarrer le service à nouveau.

```
~]# systemctl --system daemon-reload
~]# systemctl restart dhcpd
```

Les options de ligne de commande peuvent se rajouter à **ExecStart=/usr/sbin/dhcpd** dans le fichier d'unité **/etc/systemd/system/dhcpd.service** sous la section [Service]. Elles incluent :

- **-p*portnum*** — spécifie le numéro de port UDP sur lequel **dhcpd** doit écouter. La valeur par défaut est le port 67. Le serveur **DHCP** transmet les réponses aux clients **DHCP** à un numéro de port d'un numéro supérieur au numéro de port UDP spécifié. Par exemple, si le port par défaut 67 est utilisé, le serveur écoute les requêtes sur le port 67, et répond au client sur le port 68. Si un port est spécifié ici et que l'agent de relais **DHCP** est utilisé, le même port que celui sur lequel l'agent de relais **DHCP** doit écouter, devra être spécifié. Voir Section 10.3, « Agent de relais DHCP » pour plus de détails.

- **-f** — exécute le démon en arrière plan. Utilisé uniquement pour le débogage.

- **-d** — logue le démon de serveur **DHCP** en descripteur d'erreur standard. Utilisé surtout pour le débogage. Si non spécifié, la journalisation apparaîtra sur le fichier **/var/log/messages**.

- **-cf *filename*** — indique l'emplacement du fichier de configuration. L'emplacement par défaut est **/etc/dhcp/dhcpd.conf**.

- **-lf *filename*** — indique l'emplacement du fichier de base de données d'allocations. Si un fichier de base de données d'allocation existe déjà, il est très important que le même fichier soit utilisé à chaque fois que le serveur **DHCP** démarre. Il est fortement conseillé d'utiliser cette option à but de débogage sur des machines qui ne soient pas des machines de production. L'emplacement par défaut est **/var/lib/dhcpd/dhcpd.leases**.

- **-q** — ne pas afficher tout le message copyright quand vous démarrez le démon.

10.3. AGENT DE RELAIS DHCP

L'agent de relais DHCP (**dhcrelay**) permet de relayer les requêtes **DHCP** et **BOOTP** d'un sous-réseau sans serveur **DHCP** vers un ou plusieurs serveurs **DHCP** sur d'autres sous-réseaux.

Lorsqu'un client **DHCP** demande des informations, l'agent de relais DHCP transfère la requête à la liste des serveurs **DHCP** spécifiés lors du démarrage de l'agent de relais DHCP. Lorsqu'un serveur **DHCP** renvoie une réponse, la réponse est diffusée sur le réseau ayant envoyé la requête d'origine.

L'agent de relais DHCP pour **IPv4**, **dhcrelay**, écoutes les demandes **DHCPv4** et **BOOTP** sur toutes les interfaces, sauf si les interfaces sont spécifiées dans **/etc/sysconfig/dhcrelay** avec la directive **INTERFACES**. Voir Section 10.3.1, « Configurer dhcrelay en tant qu'agent de relais DHCPv4 et BOOTP ». L'agent de relais DHCP pour **IPv6**, **dhcrelay6**, n'a pas ce comportement par défaut et vous devez spécifier les interfaces pour écouter les requêtes **DHCPv6**. Voir Section 10.3.2, « Configurer dhcrelay en tant qu'agent de relais DHCPv6 ».

dhcrelay peut être exécuté en tant qu'agent de relais **DHCPv4** et **BOOTP** (par défaut) ou en tant qu'agent de relais **DHCPv6** (avec l'argument **-6**). Pour voir le message d'utilisation, exécutez la commande **dhcrelay -h**.

10.3.1. Configurer dhcrelay en tant qu'agent de relais DHCPv4 et BOOTP

Pour exécuter **dhcrelay** en mode **DHCPv4** et **BOOTP**, spécifier les serveurs dans lesquels vous souhaitez envoyer les demandes. Copier et éditer le fichier **dhcrelay.service** en tant qu'utilisateur **root** :

```
~]# cp /lib/systemd/system/dhcrelay.service /etc/systemd/system/
~]# vi /etc/systemd/system/dhcrelay.service
```

-

Modifiez l'option **ExecStart** sous la section [Service] et ajouter une ou plusieurs adresses **IPv4** de serveur à la fin de la ligne, par exemple :

```
ExecStart=/usr/sbin/dhcrelay -d --no-pid 192.168.1.1
```

Si vous souhaitez également spécifier des interfaces où l'agent de relais DHCP écoute les requêtes **DHCP**, ajoutez-les à l'option **ExecStart** avec l'argument **-i** (sinon, il écoutera toutes les interfaces), par exemple :

```
ExecStart=/usr/sbin/dhcrelay -d --no-pid 192.168.1.1 -i em1
```

. Pour les autres options, consulter la page man **dhcrelay(8)**.

Pour activer les changements, en tant qu'utilisateur **root**, démarrez le service à nouveau :

```
~]# systemctl --system daemon-reload
~]# systemctl restart dhcrelay
```

10.3.2. Configurer dhcrelay en tant qu'agent de relais DHCPv6

Pour exécuter **dhcrelay** en mode **DHCPv6**, ajouter l'argument **-6** et indiquer « lower interface » (interface de niveau inférieur sur laquelle les requêtes vont être reçues des clients ou en provenance d'autres agents de relais) et « upper interface » (l'interface de niveau supérieur sur laquelle des requêtes des clients et d'autres agents de relais doivent être transférées). Copier **dhcrelay.service** dans **dhcrelay6.service** et modifiez-le en tant qu'utilisateur **root** :

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

Modifiez l'option **ExecStart** sous la section [Service] Ajouter l'argument **-6** et ajpouter les interfaces « lower interface » et « upper interface », par exemple :

```
ExecStart=/usr/sbin/dhcrelay -d --no-pid -6 -l em1 -u em2
```

. Pour obtenir des options supplémentaires, consulter la page man **dhcrelay(8)**.

Pour activer les changements, en tant qu'utilisateur **root**, démarrez le service à nouveau :

```
~]# systemctl --system daemon-reload
~]# systemctl restart dhcrelay6
```

10.4. CONFIGURATION D'UN SERVEUR DHCP MULTI-HÔTES

Un serveur **DHCP** multi-hôtes sert plusieurs réseaux, c'est-à-dire, plusieurs sous-réseaux. Les exemples dans les sections suivantes décrivent comment configurer un serveur **DHCP** pour qu'il puisse desservir plusieurs réseaux, sélectionner les interface réseau à écouter, et comment définir les paramètres réseau pour les systèmes qui se déplacent entre les réseaux.

Avant de procéder à tout changement, sauvegarder le fichier **/etc/dhcp/dhcpd.conf** existant.

Le démon **DHCP** n'écoutera que les interfaces pour lesquelles il pourra trouver une déclaration de sous-réseau dans le fichier **/etc/dhcp/dhcpd.conf**.

Voici un fichier de base **/etc/dhcp/dhcpd.conf**, pour un serveur qui a deux interfaces de réseau, eth0 dans un réseau **10.0.0.0/24** et eth1 dans un réseau **172.16.0.0/24**. Plusieurs déclarations de **sous-réseau** permettent de définir des paramètres différents pour plusieurs réseaux :

```
default-lease-time 600;
max-lease-time 7200;
subnet 10.0.0.0 netmask 255.255.255.0 {
 option subnet-mask 255.255.255.0;
 option routers 10.0.0.1;
 range 10.0.0.5 10.0.0.15;
}
subnet 172.16.0.0 netmask 255.255.255.0 {
 option subnet-mask 255.255.255.0;
 option routers 172.16.0.1;
 range 172.16.0.5 172.16.0.15;
}
```

sous-réseau *10.0.0.0* masque réseau *255.255.255.0*;

Une déclaration de **sous-réseau** est obligatoire pour tout réseau que votre serveur **DHCP** déssert. Plusieurs sous-réseaux nécessitent plusieurs déclarations de **sous-réseaux**. Si le serveur **DHCP** n'a pas d'interface réseau faisant partie de la plage d'une déclaration de **sous-réseau**, le serveur **DHCP** ne desservira pas de ce réseau.

S'il n'y a qu'une déclaration de **sous-réseau**aucune interface de réseau qui se situe dans la plage de ce sous-réseau, le démon **DHCP** échouera au démarrage, et une erreur sera journalisée dans le fichier **/var/log/messages** :

```
dhcpd: No subnet declaration for eth0 (0.0.0.0).
dhcpd: ** Ignoring requests on eth0.  If this is not what
dhcpd:    you want, please write a subnet declaration
dhcpd:    in your dhcpd.conf file for the network segment
dhcpd:    to which interface eth1 is attached. **
dhcpd:
dhcpd:
dhcpd: Not configured to listen on any interfaces!
```

option subnet-mask *255.255.255.0*;

L'option **option subnet-mask** définit un masque de sous-réseau, et remplace la valeur du **masque de réseau** dans la déclaration du **sous-réseau**. Dans les cas simples, les valeurs du sous-réseau et du masque de réseau correspondent aux mêmes valeurs.

option routers *10.0.0.1*;

L'option **option routers** définit la passerelle par défaut du sous-réseau. Cela est exigé pour que les systèmes puissent atteindre les réseaux internes sur un sous-réseau différent, ainsi que sur des réseaux externes.

range *10.0.0.5 10.0.0.15*;

L'option **range** (plage) se réfère au pool d'adresses **IP** disponibles. Une adresse faisant partie d'une plage de valeurs **IP** indiquée est assignée aux systèmes.

Pour plus d'informations, consulter la page man **dhcpd.conf(5)**.

AVERTISSEMENT

Pour éviter une mauvaise configuration quand le serveur DHCP donne des adresses IP d'une plage d'adresses IP à un autre segment de Ethernet physique, assurez-vous de ne pas confiner plusieurs sous-réseaux dans une déclaration de réseau partagé.

10.4.1. Configuration de l'hôte

Avant de procéder à tout changement, sauvegarder les fichiers **/etc/dhcp/dhcpd.conf** et **/etc/dhcp/dhcpd.conf** existants.

Configurer un système unique pour plusieurs réseaux

L'exemple **/etc/dhcp/dhcpd.conf** suivant crée deux sous-réseaux et configure une adresse **IP** pour le même système, selon le réseau auquel elle se connecte :

```
default-lease-time 600;
max-lease-time 7200;
subnet 10.0.0.0 netmask 255.255.255.0 {
 option subnet-mask 255.255.255.0;
 option routers 10.0.0.1;
 range 10.0.0.5 10.0.0.15;
}
subnet 172.16.0.0 netmask 255.255.255.0 {
 option subnet-mask 255.255.255.0;
 option routers 172.16.0.1;
 range 172.16.0.5 172.16.0.15;
}
host example0 {
 hardware ethernet 00:1A:6B:6A:2E:0B;
 fixed-address 10.0.0.20;
}
host example1 {
 hardware ethernet 00:1A:6B:6A:2E:0B;
 fixed-address 172.16.0.20;
}
```

hôte *example0*

La déclaration **host** définit des paramètres spécifiques pour un système unique, comme une adresse **IP**. Pour configurer des paramètres spécifiques sur plusieurs hôtes, utiliser des déclarations d'**hôte**.

La plupart des clients **DHCP** ignorent le nom dans les déclarations d'**hôte**, et à ce titre, ce nom peut être n'importe quoi, tant que c'est différent des autres déclarations d'**hôte**. Pour configurer le même système sur des réseaux multiples, utilisez un nom différent pour chaque déclaration d'**hôte**, sinon le démon **DHCP** ne démarrera pas. Les systèmes sont identifiés par l'option **hardware ethernet**, et non pas le nom dans la déclaration de **l'hôte**.

hardware ethernet *00:1A:6B:6A:2E:0B*;

L'option **hardware ethernet** identifie le système. Pour trouver cette adresse, exécuter la commande **ip link**.

fixed-address *10.0.0.20*;

L'option **fixed-address** assigne une adresse **IP** au système spécifié par l'option **hardware ethernet**. Cette adresse doit se trouver en dehors de la plage d'adressses **IP** spécifiée par l'option **range**.

Si les énoncés d'**option** ne se terminent pas par un point virgule, le démon **DHCP** ne démarrera pas, et une errreur ressemblant à ce qui suit sera journalisée dans **/var/log/messages** :

```
/etc/dhcp/dhcpd.conf line 20: semicolon expected.
dhcpd: }
dhcpd: ^
dhcpd: /etc/dhcp/dhcpd.conf line 38: unexpected end of file
dhcpd:
dhcpd: ^
dhcpd: Configuration file errors encountered -- exiting
```

Configurer des systèmes avec des interfaces de réseaux multiples

Les déclarations suivantes d'**hôte** configurent un système unique, qui possède plusieurs interfaces de réseau, pour que chaque interface reçoive la même adresse **IP**. Cette configuration ne fonctionnera pas si les deux interfaces réseau sont connectées au même réseau en même temps :

```
host interface0 {
 hardware ethernet 00:1a:6b:6a:2e:0b;
 fixed-address 10.0.0.18;
}
host interface1 {
 hardware ethernet 00:1A:6B:6A:27:3A;
 fixed-address 10.0.0.18;
}
```

Pour cet exemple, **interface0** est la première interface réseau, et **interface1** est la seconde interface. Les différentes options **hardware ethernet** identifient chaque interface.

Si un tel système se connecte à un autre réseau, ajouter plus de déclarations d'**hôte**, sans oublier :

- d'assigner une **fixed-address** valide au réseau auquel l'hôte se connecte.
- de rendre le nom de déclaration d'**host** unique.

Quand un nom non unique est donnée dans une déclaration d'**hôte**, le démon **DHCP** ne démarre pas, et l'erreur suivante est journalisée dans **/var/log/messages** :

```
dhcpd: /etc/dhcp/dhcpd.conf line 31: host interface0: already exists
dhcpd: }
dhcpd: ^
dhcpd: Configuration file errors encountered -- exiting
```

L'erreur vient du fait d'avoir plusieurs déclarations d'**host interface0** définies dans **/etc/dhcp/dhcpd.conf**.

10.5. DHCP POUR IPV6 (DHCPV6)

L'ISC **DHCP** inclut un support pour **IPv6** (**DHCPv6**) depuis la version 4.x avec un serveur **DHCPv6**, un client et une fonctionnalité d'agent de relais. Les agents supportent à la fois **IPv4** et **IPv6**. Cependant, les agents ne peuvent gérer qu'un seul protocole à la fois; et pour le support double, ils doivent être démarrés séparemment pour **IPv4** et **IPv6**. Par exemple, configurez **DHCPv4** et **DHCPv6** en éditant leurs fichiers de configuration respectifs **/etc/dhcp/dhcpd.conf** et **/etc/dhcp/dhcpd6.conf**, puis exécutez les commandes suivantes :

```
~]# systemctl start dhcpd
~]# systemctl start dhcpd6
```

Le fichier de configuration du serveur **DHCPv6** se trouve dans **/etc/dhcp/dhcpd6.conf**.

On peut trouver le fichier de configuration de l'exemple de serveur dans **/usr/share/doc/dhcp-*version*/dhcpd6.conf.example**.

Un simple fichier de configuration de serveur **DHCPv6** peut ressembler à ce qui suit :

```
subnet6 2001:db8:0:1::/64 {
        range6 2001:db8:0:1::129 2001:db8:0:1::254;
        option dhcp6.name-servers fec0:0:0:1::1;
        option dhcp6.domain-search "domain.example";
}
```

10.6. RESSOURCES SUPPLÉMENTAIRES

Les sources d'informations suivantes fournissent des ressources supplémentaires à propos de **DHCP**.

10.6.1. Documentation installée

- Page man **dhcpd(8)** — décrit comment le démon **DHCP** fonctionne.
- Page man **dhcpd.conf(5)** — décrit comment configurer le fichier de configuration **DHCP**; inclut certains exemples.
- Page man **dhcpd.leases(5)** man page — décrit une base de données d'attributions persistante.
- Page man **dhcp-options(5)** — explique la syntaxe pour déclarer les options **DHCP** dans **dhcpd.conf**; inclut des exemples.
- Page man **dhcrelay(8)** — explique l'agent de relais **DHCP** et ses options de configuration.
- **/usr/share/doc/dhcp-*version*/** — contient des fichiers d'exemples, des fichiers README, et des notes de version des versions en cours du service **DHCP**.

CHAPITRE 11. LES SERVEURS DNS

DNS (Domain Name System) est un système de base de données distribué utilisé pour associer les noms d'hôtes à leurs adresses **IP** respectives. Pour les utilisateurs, cela a l'avantage qu'ils peuvent faire référence à des machines du réseau par leur nom, ce qui est normalement plus facile à mémoriser que les adresses numériques de réseau. Pour les administrateurs de système, l'utilisation d'un serveur **DNS** (ou *nameserver*) permet de changer l'adresse **IP** pour un hôte sans affecter pour autant les recherches basées nom. L'utilisation des bases de données **DNS** sert non seulement à résoudre les adresses **IP** en noms de domaine, mais leur utilisation s'élargit de plus en plus au fur et à mesure que DNSSEC se déploie.

11.1. INTRODUCTION À DNS

DNS est généralement implémenté à l'aide d'un ou plusieurs serveurs centralisés qui font autorité pour certains domaines. Lorsqu'un hôte client demande des informations à un serveur de noms, il se connecte généralement au port 53. Le serveur de noms tente alors de résoudre le nom demandé. Si le serveur de noms est configuré en serveur de noms récursifs et qu'il n'a pas une réponse faisant autorité, ou n'a pas la réponse en cache suite à une requête antérieure, il interroge d'autres serveurs de noms, appelés *serveurs de noms racine*, pour déterminer quels serveurs de noms font autorité pour le nom en question, puis, il les interroge pour obtenir le nom demandé. Les noms de serveurs configurés comme étant purement autoritatifs, avec la récurrence désactivée, ne feront pas de recherches pour les compte des clients.

11.1.1. Zones de noms de serveurs

Dans un serveur **DNS**, toutes les informations sont stockées dans des éléments de base de données appelés *enregistrements de ressources* (RR). Les enregistrements de ressources sont définis dans RFC 1034. Les noms de domaine sont organisés en structure arborescente. Chaque niveau de la hiérarchie est divisé par un point (**.**). Par exemple : le domaine racine, précédé de **.**, correspond à la racine de l'arborescence **DNS**, qui est au niveau zéro. Le nom de domaine **com**, dénommé le *domaine de premier niveau* (TLD) est un enfant du domaine racine (**.**) et correspond donc au premier niveau de la hiérarchie. Le nom de domaine **example.com** est au deuxième niveau de la hiérarchie.

> **Exemple 11.1. Un simple enregistrement de ressource**
>
> Exemple d'enregistrement de ressource ou *enregistrement de ressource* (RR) simple :
>
> ```
> example.com.      86400    IN         A           192.0.2.1
> ```
>
> Le nom de domaine **example.com** est le *propriétaire* du RR (enregistrement de ressource). La valeur **86400** correspond à la *durée de vie* (ou TTL). Les lettres **IN** (remplaçant « système Internet »), indiquent la *classe* de l'enregistrement de ressource (RR). La lettre **A** indique le *type* de RR (dans cet exemple, une adresse d'hôte). L'adresse de l'hôte **192.0.2.1** représente les données contenues dans la dernière partie de cet enregistrement de ressource (RR). C'est un exemple d'enregistrement de ressource (RR) en une ligne. Un ensemble d'enregistrements de ressources avec les mêmes type, propriétaire et classe est appelé un *resource record set* (RRSet).

Les zones sont définies sur les serveurs de noms autoritatifs par l'utilisation des *fichiers de zone*, qui contiennent des définitions des enregistrements de ressources dans chaque zone. Les fichiers sont stockés sur des *serveurs de noms primaires* (également appelés *serveurs de noms maîtres*), où des modifications sont apportées aux fichiers et aux *serveurs de noms secondaires* (également appelés *noms de serveurs esclaves*), qui reçoivent des définitions de zones des serveurs de noms primaires. Les

serveurs de noms primaires et secondaires font autorités pour la zone et se ressemblent pour le client. Selon la configuration, un serveur de noms peut également servir de serveur principal ou secondaire pour plusieurs zones à la fois en même temps.

Notez que les administrateurs de serveurs **DNS** et **DHCP**, ainsi que toute application de provisioning, doivent s'accorder sur le format des noms d'hôtes utilisé dans une organisation. Veuillez consulter le Section 3.1.1, « Pratiques de nommage conseillées » pour obtenir davantage d'informations sur les formats de noms d'hôtes.

11.1.2. Types de noms de serveurs

Il existe deux types de déclarations dans le fichier de configuration :

authoritative

Les serveurs de noms autoritatifs répondent à des enregistrements de ressources qui font partie de leur zone uniquement. Cette catégorie inclut à la fois les serveurs primaires (master) et secondaires (esclave).

recursive

Les serveurs de noms récursifs offrent des services de résolution, mais ils ne font pas autorité pour toutes les zones. Des réponses pour toutes les résolutions sont mises en cache en mémoire pour une période déterminée, qui est spécifiée par l'enregistrement de ressource obtenue.

Bien qu'un serveur de noms puisse être à la fois faire valoir d'autorité et être récursif, il est recommandé ne pas de combiner les types de configuration. Pour être en mesure d'exécuter leur travail, les serveurs faisant autorités doivent être disponibles à tous les clients à tout moment. Mais comme la recherche récursive prend beaucoup plus de temps que les réponses qui font autorité, les serveurs récursifs doivent être accessibles à un nombre restreint de clients uniquement, sinon, ils auront tendance à lancer des attaques par déni de service (DDoS).

11.1.3. BIND en tant que serveur de noms

BIND représente un ensemble de programmes liés à DNS. Il comprend un serveur de noms intitulé **named**, un utilitaire d'administration intitulé **rndc**, et un outil de déboggage intitulé **dig**. Voir le guide *Red Hat Enterprise Linux 7 System Administrator's Guide* pour obtenir plus d'informations sur la façon d'exécuter un service dans Red Hat Enterprise Linux.

11.2. BIND

Cette section porte sur le serveur **BIND** (Berkeley Internet Name Domain), le serveur **DNS** inclus dans Red Hat Enterprise Linux. Elle est orientée sur la structure de ses fichiers de configuration, et décrit comment les administrer à la fois localement et à distance.

11.2.1. Zones vides

BIND configure un certain nombre de « zones vides » afin d'empêcher des serveurs récursifs d'envoyer des requêtes inutiles vers des serveurs Internet qui ne peuvent pas les gérer (créant ainsi des retards et des réponses SERVFAIL aux clients qui les interrogent). Ces zones vides veillent à ce que les réponses NXDOMAIN immédiates et faisant autorités soient retournées à la place. L' option de configuration **vide-zones-enable** détermine si oui ou non les zones vides sont créées, tandis que l'option **disable-vide-zone** peut servir à désactiver une ou plusieurs zones vides dans la liste des préfixes par défaut, qui est utilisée.

Le nombre d'espaces vides créés pour les préfixes *RFC 1918* a augmenté, et les utilisateurs de **BIND 9.9** et versions ultérieures, verront des espaces vides *RFC 1918* à la fois quand **empty-zones-enable** n'est pas spécifié (valeur par défaut **yes**), ou quand RFC 1918 est défini à **yes**.

11.2.2. Configuration du service «named»

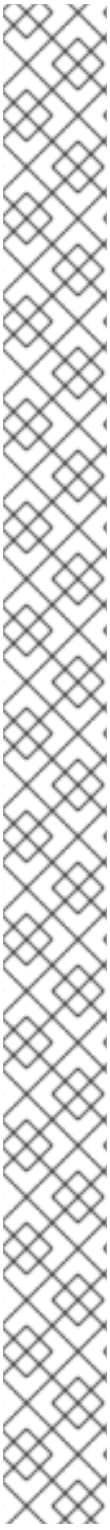
Quand le service **named** démarre, il lit la configuration à partir des fichiers décrits dans Tableau 11.1, « Les fichiers de configuration du service «named» ».

Tableau 11.1. Les fichiers de configuration du service «named»

Chemin	Description
/etc/named.conf	Fichier de configuration principal.
/etc/named/	Répertoire auxiliaire pour les fichiers de configuration inclus dans le fichier de configuration principal.

Le fichier de configuration consiste en un ensemble d'arguments comprenant des options imbriquées entourées par des crochets courbes (**{** et **}**). Veuillez noter que si vous modifiez le fichier, le service **named** ne démarrera pas. Un fichier **/etc/named.conf** se présente ainsi :

```
statement-1 ["statement-1-name"] [statement-1-class] {
  option-1;
  option-2;
  option-N;
};
statement-2 ["statement-2-name"] [statement-2-class] {
  option-1;
  option-2;
  option-N;
};
statement-N ["statement-N-name"] [statement-N-class] {
  option-1;
  option-2;
  option-N;
};
```



NOTE

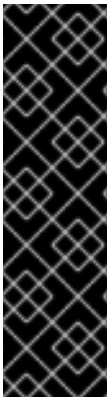
Si vous avez installé le paquet bind-chroot, le service de liaison exécutera dans l'environnement **chroot**. Dans ce cas, le script d'initialisation procédera au montage des fichiers de configuration ci-dessus à l'aide de la commande **mount--bind**, afin que vous puissiez contrôler la configuration en dehors de cet environnement. Il n'y a pas besoin de copier quoi que ce soit dans le répertoire **/var/named/chroot/** parce qu'elle est montée automatiquement. Cela simplifie la maintenance puisque vous n'avez pas besoin de prendre un soin particulier des fichiers de configuration **BIND** si la commande est exécutée dans un environnement **chroot**. Vous pouvez tout organiser comme vous le feriez avec **BIND** si vous n'étiez pas dans un environnement **chroot**.

Les répertoires suivants sont montés automatiquement sur **/var/named/chroot/** si les répertoires de point de montage correspondants qui se trouvent sous **/var/named/chroot/** sont vides :

- **/etc/named**
- **/etc/pki/dnssec-keys**
- **/run/named**
- **/var/named**
- **/usr/lib64/bind** ou **/usr/lib/bind** (suivant l'architecture).

Les fichiers suivants sont également montés si le fichier cible n'existe pas dans **/var/named/chroot/** :

- **/etc/named.conf**
- **/etc/rndc.conf**
- **/etc/rndc.key**
- **/etc/named.rfc1912.zones**
- **/etc/named.dnssec.keys**
- **/etc/named.iscdlv.key**
- **/etc/named.root.key**



IMPORTANT

Pour modifier des fichiers qui ont été montés dans un environnement **chroot**, vous devrez créer une copie de sauvegarde, puis modifier le fichier d'origine. Sinon, utiliser un éditeur avec le mode « edit-a-copy » désactivé. Ainsi, pour modifier le fichier de configuration de BIND, **/etc/named.conf**, avec Vim quand il exécute dans un envirionnement **chroot**, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# vim -c "set backupcopy=yes" /etc/named.conf
```

11.2.2.1. Installation de BIND dans un envirionnement chroot

Pour installer **BIND** pour qu'il exécute dans un environnement **chroot**, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install bind-chroot
```

Pour activer le service **named-chroot**, commencez par vérifier que le service **named** est en cours d'exécution en lançant la commande suivante :

```
~]$ systemctl status named
```

S'il est en cours d'exécution, il doit être désactivé.

Pour déactiver le service **chronyd**, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl stop named
```

```
~]# systemctl disable named
```

Puis, pour déactiver le service **named-chroot**, veuillez exécuter les commandes suivantes en tant qu'utilisateur **root** :

```
~]# systemctl enable named-chroot
```

```
~]# systemctl start named-chroot
```

Pour vérifier le statut du service **named-chroot**, veuillez exécuter les commandes suivantes en tant qu'utilisateur **root** :

```
~]# systemctl status named-chroot
```

11.2.2.2. Types d'arguments communs

Les types d'arguments suivants sont souvent utilisés dans **/etc/named.conf** :

acl

L'argument **acl** (Access Control List) vous permet de définir des groupes d'hôtes, de façon à ce qu'ils aient accès ou non au serveur de noms. Prend la forme suivante :

```
acl acl-name {
  match-element;
  ...
};
```

Le nom d'argument*acl-name* correspond au nom de la liste de contrôle d'accès, et l'option *match-element* correspond normalement à une adresse **IP** individuelle (comme **10.0.1.1**) ou à une notation de réseau *Classless Inter-Domain Routing* (CIDR) (par exemple, **10.0.1.0/24**). Pour obtenir une liste de mots clés déjà pré-définis, voir Tableau 11.2, « Listes de contrôle d'accès pré-définies ».

Tableau 11.2. Listes de contrôle d'accès pré-définies

Mot-clé	Description
any	Fait correspondre chaque adresse **IP**.
localhost	Fait correspondre chaque adresse **IP** utilisée par le système local.
localnets	Fait correspondre chaque adresse **IP** sur n'importe quel réseau connecté au système local.
none	Ne correspond à aucune adresse **IP**.

L'argument **acl** peut être particulièrement utile s'il est utilisé en conjonction à d'autres arguments tels que **options**. Exemple 11.2, « Utilisation d'acl en conjonction aux options » définit deux listes de contrôle d'accès, **black-hats** et **red-hats**, et ajoute **black-hats** à la liste noire tout en donnant l'accès normal **red-hats**.

Exemple 11.2. Utilisation d'acl en conjonction aux options

```
acl black-hats {
  10.0.2.0/24;
  192.168.0.0/24;
  1234:5678::9abc/24;
};
acl red-hats {
  10.0.1.0/24;
};
options {
  blackhole { black-hats; };
  allow-query { red-hats; };
  allow-query-cache { red-hats; };
};
```

include

L'argument **include** vous permet d'inclure des fichiers dans **/etc/named.conf**, de façon à ce que des données pouvant être sensibles puissent être mises dans un fichier séparé sans limitation de permissions. Prend la forme suivant :

```
include "file-name"
```

L'argument *file-name* est un nom d'argument qui se trouve dans un chemin d'accès complet de fichier.

Exemple 11.3. Inclure un fichier dans /etc/named.conf

```
inclure "/etc/named.rfc1912.zones";
```

options

L'argument **options** vous permet de définir des options de configuration du serveur global, ainsi que de définir les valeurs par défaut d'autres arguments. Il peut être utilisé pour spécifier l'emplacement du répertoire de travail de **named**, les types de requêtes autorisées et bien plus encore. Il prend la forme suivante :

```
options {
  option;
  ...
};
```

Pour obtenir une liste des possibilités d'*option* les plus communément utilisées, consulter le tableau ci-dessous Tableau 11.3, « Options de configuration souvent utilisées ».

Tableau 11.3. Options de configuration souvent utilisées

Option	Description
allow-query	Spécifie quels hôtes sont autorisés à interroger le serveur de noms pour les enregistrements de ressources de référence. Il accepte une liste de contrôle d'accès, une collection d'adresses **IP**, ou des réseaux dans la notation CIDR. Par défaut, tous les hôtes sont autorisés.
allow-query-cache	Indique quels hôtes sont autorisés à interroger le serveur de noms pour des données ne faisant pas autorité, comme les requêtes récursives. Par défaut, seuls les **localhost** et **localnets** sont autorisés.
blackhole	Indique les hôtes *non* autorisés à interroger le serveur de noms. Cette option doit être utilisée lorsqu'un hôte particulier ou qu'réseau inonde le serveur de demandes. L'option par défaut est **none**.
directory	Indique un répertoire de travail pour le service **named**. L'option par défaut est **/var/named/**.
disable-empty-zone	Utilisé pour désactiver une ou plusieurs des zones vides de la liste des préfixes par défaut qui pourraient être utilisés. Peut être spécifié dans les arguments d'options et également dans les arguments de vues. Il peut être utilisé à plusieurs reprises.
dnssec-enable	Indique si l'on doit retourner aux enregistrements de ressources liées à DNSSEC. L'option par défaut est **yes**.
dnssec-validation	Indique si l'on doit prouver que les enregistrements de ressources sont authentiques via DNSSEC. L'option par défaut est **yes**.
empty-zones-enable	Contrôle si les zones vides sont créées ou non. Ne peut être spécifié que dans les arguments d'options.
forwarders	Indique une liste d'adresses **IP** pour les serveurs de noms vers lesquels les requêtes doivent être redirigées pour qu'elles soient résolues.

Option	Description
forward	Indique le comportement de la directive **forwarders**. Accepte les options suivantes : • **first** — Le serveur va vérifier les serveurs de noms énumérés dans la directive **forwarders** avant de tenter de résoudre le nom par lui-même. • **only** — Quand on ne peut pas vérifier les serveurs de noms éumérés dans la directive **forwarders**, le serveur ne tentera pas de résoudre le nom par lui-même.
listen-on	Indique l'interface de réseau **IPv4** sur lequel écouter les requêtes. Sur un serveur **DNS**, qui agit aussi comme une passerelle, vous pouvez utiliser cette option pour répondre à des requêtes provenant d'un réseau uniquement. Toutes les interfaces **IPv4** sont utilisées par défaut.
listen-on-v6	Indique l'interface de réseau **IPv6** sur lequel écouter pour les requêtes. Sur un serveur **DNS** qui agit aussi en tant quepasserelle, vous pouvez utiliser cette option pour répondre à des requêtes provenant d'un seul réseau. Toutes les interfaces **IPv6** sont utilisées par défaut.
max-cache-size	Indique la quantité maximale de mémoire à utiliser pour les caches de serveurs. Lorsque la limite est atteinte, le serveur entraîne l'expiration prématurée des enregistrements, alors que la limite n'est pas dépassée. Sur un serveur avec plusieurs vues, la limite s'appliquera séparément dans le cache de chaque vue. L'option par défaut est **32M**.
notify	Indique si on doit notifier les serveurs de noms secondaires quand une zone est mise à jour. Accepte les options suivantes : • **yes** — le serveur notifiera tous les serveurs de noms secondaires. • **no** — le serveur ne notifiera *aucun* serveur de noms secondaire. • **master-only** — le serveur notifiera un serveur primaire pour cette zone uniquement. • **explicit** — le serveur ne notifiera que les serveurs secondaires spécifiés dans la liste **also-notify** dans un argument de zone.
pid-file	Indique l'emplacement du fichier d'ID de processus créé par le service **named**.
recursion	Indique comportement de serveur récurvif ou non. L'option par défaut est **yes**.
statistics-file	Indique un emplacement spécifique pour les fichiers de statistiques. Le fichier **/var/named/named.stats** est utilisé par défaut.

NOTE

Le répertoire utilisé par **named** pour les données de runtime a été déplacé de l'emplacement par défaut de BIND, **/var/run/nom /**, à un nouvel emplacement **/run/name/**. Ainsi, le fichier PID a été déplacé de l'emplacement par défaut **/var/run/named/named.pid** au nouvel emplacement **/run/named/named.pid**. De plus, le fichier de clés de session a été déplacé à **/run/named/session.key**. Ces emplacements devront être précisés par les arguments de la section options. Voir Exemple 11.4, « Utiliser l'argument « option » ».

IMPORTANT

Pour éviter les attaques DDoS (de l'anglais, distributed denial of service), nous vous conseillons d'utiliser l'option **allow-query-cache** pour limiter l'accès aux services **DNS** récursifs à quelques groupes de clients particuliers.

Voir le *BIND 9 Administrator Reference Manual* référencé dans Section 11.2.8.1, « Documentation installée », et la page man **named.conf** pour obtenir une liste complète des options.

Exemple 11.4. Utiliser l'argument « option »

```
options {
  allow-query       { localhost; };
  listen-on port    53 { 127.0.0.1; };
  listen-on-v6 port 53 { ::1; };
  max-cache-size    256M;
  directory         "/var/named";
  statistics-file   "/var/named/data/named_stats.txt";

  recursion         yes;
  dnssec-enable     yes;
  dnssec-validation yes;

  pid-file          "/run/named/named.pid";
  session-keyfile   "/run/named/session.key";
};
```

zone

L'argument **zone** vous permet de définir les caractéristiques d'une zone, comme l'emplacement de son fichier de configuration et les options spécifiques à la zone, et il peut être utilisé en remplacement aux arguments d'**options**. Il prend la forme suivante :

```
zone zone-name [zone-class] {
  option;
  ...
};
```

L'attribut *zone-name* correspond au nom de la zone, *zone-class* correspond à la classe optionnelle de zone, et *option* correspond à une option d'argument de **zone** comme expliqué dans Tableau 11.4, « Options couramment utilisées comme arguments de zone ».

L'attribut *zone-name* est particulièrement important, car il correspond à la valeur par défaut assignée à la directive **$ORIGIN** utilisée dans le fichier de zone correspondant qui se trouve dans le répertoire **/var/named/**. Le démon **named** ajoute le nom de la zone à n'importe quel nom de domaine incomplet répertorié dans le fichier de zone. Par exemple, si un argument de **zone** définit l'espace de noms pour **example.com**, utilisez **example.com** comme *zone-name* pour qu'il soit placé à la fin des noms d'hôte dans le fichier de zone **example.com**.

Pour obtenir plus d'informations sur les fichiers de zone, consulter Section 11.2.3, « Conflits de fichiers ».

Tableau 11.4. Options couramment utilisées comme arguments de zone

Option	Description
allow-query	Indique quels clients sont autorisés à demander des informations sur cette zone. Cette option prévaut sur l'option globale **allow-query**. Tous les demandes de requête sont autorisées par défaut.
allow-transfer	Indique les serveurs secondaires qui sont autorisés à demander un transfert de l'information de zone. Toutes les requêtes de transfert sont autorisées par défaut.
allow-update	Indique quels hôtes sont autorisés à mettre leurs informations à jour de façon dynamique dans leur zone. L'option par défaut est de refuser toutes les demandes de mise à jour dynamiques. Notez que vous devez être prudents lorsque vous autorisez les clients à mettre à jour des informations sur leur zone. Ne définissez pas les adresses **IP** dans cette option à moins que le serveur soit dans un réseau fiable. À la place, utiliser la clé TSIG décrite dans Section 11.2.6.3, « TSIG (Transaction SIGnatures) ».
file	Indique le nom du fichier qui se trouve dans le répertoire de travail **named** qui contient les données de configuration de la zone.
masters	Indique à partir de quelles adresses **IP** on peut demander des informations de zone autoritatives. Cette option n'est utilisée que si la zone est définie en tant que **type slave**.
notify	Indique si on doit notifier les serveurs de noms secondaires quand une zone est mise à jour. Accepte les options suivantes : • **yes** — le serveur va notifier tous les serveurs de noms secondaires. • **no** — le serveur ne notifiera *aucun* serveur de noms secondaires. • **master-only** — le serveur ne notifiera le serveur primaire que pour la zone. • **explicit** — le serveur ne notifiera que les serveurs secondaires spécifiés dans la liste **also-notify** dans l'argument de zone.

Option	Description
type	Indique le type de zone. Accepte les options suivantes : • **delegation-only** — active le statut de délégation de zones d'infrastructures comme COM, NET, ou ORG. Toute réponse reçue sans délégation implicite ou explicite sera traitée comme **NXDOMAIN**. Cette option ne s'applique qu'aux niveaux TLDs (Top-Level Domain) ou pour les fichiers de zone racine pour les implémentations en cache ou récursives. • **forward** — transfère toutes les demandes d'informations de cette zone vers d'autres serveurs de noms. • **hint** — un type particulier de zone utilisée pour pointer vers les serveurs de noms racine qui résolvent les requêtes lorsqu'une zone n'est pas connue par ailleurs. Aucune configuration au-delà de la valeur par défaut n'est nécessaire dans une zone **hint** (astuce). • **master** — indique le serveur de noms comme étant le serveur faisant autorité pour cette zone. Une zone doit être définie comme **master** si les fichiers de configuration de la zone se trouvent sur le système. • **slave** — désigne le serveur de noms comme serveur esclave de la zone. Le serveur maître est spécifié dans la directive **masters**.

La plupart des changements au fichier **/etc/named.conf** d'un serveur de noms primaire ou secondaire consistent à ajouter, modifier ou supprimer des arguments de **zone**, et seul un petit nombre d'options d'arguments de **zone** est normalement utile pour qu'un serveur de noms puisse fonctionner efficacement.

Dans Exemple 11.5, « Argument de zone pour un serveur de noms primaire », la zone est identifiée comme **example.com**, le type est défini sur le **master** et le service **named** est chargé de lire le fichier **/var/named/example.com.zone**. Il permet également à un serveur de noms secondaire (**192.168.0.2**) uniquement de transférer la zone.

Exemple 11.5. Argument de zone pour un serveur de noms primaire

```
zone "example.com" IN {
  type master;
  file "example.com.zone";
  allow-transfer { 192.168.0.2; };
};
```

L'argument de **zone** d'un serveur secondaire est légèrement différent. Le type est défini sur **l'esclave**, et la directive du **master** indique au service **named** l'adresse **IP** du serveur maître.

Dans Exemple 11.6, « Un argument de zone pour le serveur de noms secondaire », le service **named** est configuré pour interroger le serveur principal à l'adresse **IP 192.168.0.1** pour obtenir des informations sur la zone **example.com**. L'information reçue est alors enregistrée dans le fichier

/var/named/slaves/example.com.zone. Notez que vous devez mettre toutes les zones esclave dans le répertoire **/var/named/esclaves /**, sinon le service ne pourra pas transférer la zone.

Exemple 11.6. Un argument de zone pour le serveur de noms secondaire

```
zone "example.com" {
  type slave;
  file "slaves/example.com.zone";
  masters { 192.168.0.1; };
};
```

11.2.2.3. Autres types d'arguments

Les types d'arguments suivants sont moins souvent utilisés dans **/etc/named.conf** :

controls

L'argument **controls** vous permet de configurer les diverses exigences de sécurité avant d'utiliser la commande **rndc** qui sert à administrer le service **named**.

Reportez-vous à Section 11.2.4, « Comment se servir de l'utilitaire rndc » pour obtenir plus d'informations sur l'utilitaire **rndc** et la façon de l'utiliser.

key

L'argument **key** vous permet de définir une clé particulière par son nom. Les clés sont utilisées pour authentifier les différentes actions, telles que les mises à jour sécurisées ou l'utilisation de la commande **rndc**. Deux options sont utilisées avec l'argument **key** :

- **algorithm *algorithm-name*** — le type d'algorithme à utiliser (par exemple, **hmac-md5**).
- **secret "*key-value*"** — la clé cryptée.

Reportez-vous à Section 11.2.4, « Comment se servir de l'utilitaire rndc » pour obtenir plus d'informations sur l'utilitaire **rndc** et la façon de l'utiliser.

logging

L'argument **logging** vous permet d'utiliser plusieurs types de journaux, appelés *canaux*. En utilisant l'option **channel** dans l'argument, vous pouvez construire un type personnalisé du journal avec son propre nom de fichier (**file**), une limite de taille (**taille**), un numéro de version (**version**) et un niveau d'importance (**gravité**). Une fois qu'un canal (channel) personnalisé est défini, une option **catégorie** est utilisée pour catégoriser le canal et commencer la journalisation lorsque le service **named** démarre à nouveau.

Par défaut, le service **named** envoie des messages standards au démon **rsyslog**, qui les place dans **/var/log/messages**. Plusieurs canaux standards sont intégrés dans BIND selon les niveaux de gravité, comme **default_syslog** (qui gère les messages de journalisation d'information) et **default_debug** (qui gère spécifiquement les messages de débogage). Une catégorie par défaut, appelée **par défaut**, utilise les canaux intégrés afin d'effectuer une journalisation normale sans aucune configuration spéciale.

Personnaliser le processus de journalisation peut être un processus très détaillé, qui dépasse le cadre du présent chapitre. Pour obtenir des informations sur la création de journaux BIND personnalisés, consultez le guide *BIND 9 Administrator Reference Manual* référencé dans Section 11.2.8.1, « Documentation installée ».

server

L'argument **server** vous permet de spécifier des options qui affectent comment le service **named** doit répondre à des serveurs de noms éloignés, surtout au niveau notifications et transferts de zones.

L'option **transfer-format** contrôle le nombre d'enregistrements de ressources qui sont envoyées avec chaque message. Il peut être **one-answer** (enregistrement d'une seule ressource) ou **many-answers** (plusieurs enregistrements de ressources). Notez que bien l'option **many-answers** soit plus efficace, elle n'est pas supportée par les anciennes versions de BIND.

trusted-keys

L'argument **trusted-keys** vous permet d'indiquer des clés publiques assorties utilisées pour sécuriser le protocole **DNS** (DNSSEC). Voir Section 11.2.6.4, « DNSSEC (DNS Security Extensions) » pour obtenir plus d'informations à ce sujet.

view

L'argument **view** vous permet de créer des affichages spéciaux selon le réseau sur lequel l'hôte qui interroge le serveur se trouve. Cela permet à certains hôtes de recevoir une réponse concernant une zone tandis que les autres hôtes reçoivent des informations totalement différentes. Par ailleurs, certaines zones ne peuvent être rendues disponibles qu'aux hôtes de confiance, tandis que les hôtes non approuvés ne peuvent effectuer des requêtes que pour d'autres zones.

Des vues multiples peuvent être utilisées tant que leurs noms sont uniques. L'option **match-clients** (correspondance client) vous permet de spécifier les adresses **IP** qui s'appliquent à une vision particulière. Si l'argument **options** est utilisé dans une vue, il remplacera les options globales déjà configurées. Enfin, la plupart des arguments de **view** contiennent plusieurs arguments de **zone** qui s'appliquent à la liste de **match-clients**.

Notez que l'ordre dans lequel les arguments de **view** sont listés est important, car le premier argument qui correspond à une adresse **IP** de client particulier sera utilisée. Pour plus d'informations sur ce sujet, voir Section 11.2.6.1, « Vues multiples ».

11.2.2.4. Balises de commentaires

En plus des arguments, le fichier **/etc/named.conf** peut également contenir des commentaires. Les commentaires peuvent être ignorés par le service **named**, mais peuvent se révéler utiles quand on donne des informations supplémentaires à un utilisateur. Voici des balises de commentaires valides :

//

Tout texte se trouvant après **//** en fin de ligne est considéré comme un commentaire seulement. Exemple :

```
notify yes;  // notifie tous les serveurs de noms secondaires
```

#

Tout texte se trouvant après # en fin de ligne est considéré comme un commentaire seulement. Exemple :

```
notify yes;  # notifie tous les serveurs de noms secondaires
```

/* et */

Tout bloc de texte se trouvant entre **/*** et ***/** est considéré comme un commentaire. Exemple :

```
notify yes;  /* notifie tous les serveurs de noms secondaires */
```

11.2.3. Conflits de fichiers

Comme indiqué dans Section 11.1.1, « Zones de noms de serveurs », les fichiers de zone contiennent des informations sur un espace de noms. Ils sont stockés dans le répertoire de travail **named** situé dans **/var/named/** par défaut. Chaque fichier de zone est nommé selon l'option de **fichier** dans l'argument **zone**, habituellement d'une manière qui porte sur le domaine et qui identifie le fichier comme contenant des données de zone, comme **example.com.zone**.

Tableau 11.5. Les fichiers de zones de service « named »

Chemin	Description
/var/named/	Le répertoire de travail du service **named**. Le serveur de noms n'est *pas* autorisé à écrire dans ce répertoire.
/var/named/slaves/	Le répertoire de zones secondaires. Ce répertoire peut contenir des écritures du service **named**.
/var/named/dynamic/	Le répertoire pour les autres fichiers, comme les zones dynamiques **DNS** (DDNS) ou les clés DNSSEC gérées. Ce répertoire est accessible en écriture par le service **named**.
/var/named/data/	Le répertoire de statistiques variées et de fichiers de débogage. Ce répertoire peut contenir des écritures du service **named**.

Un fichier de zone se compose d'enregistrements de ressources et de directives. Les directives instruisent le serveur de noms à effectuer des tâches ou à appliquer des paramètres de configuration particuliers à la zone. Les enregistrements de ressources définissent les paramètres de la zone et attribuent des identités aux hôtes individuels. Bien que les directives soient facultatives, les enregistrements de ressources sont utiles pour procurer un service de nommage à une zone.

Toutes les direcives et les enregistrements de noms doivent être saisis sur des lignes individuelles.

11.2.3.1. Directives communes

Les directives commencent par le signe (**$**) et sont suivies par le nom de la directive, qui apparaît normalement en haut du fichier. Les directives suivantes sont couramment utilisées dans les fichiers de zone :

$INCLUDE

La directive **$INCLUDE** vous permet d'inclure un autre fichier là où il se trouve, de façon à ce que les autres paramètres de configuration de zone puissent être stockés dans un fichier de zone séparé.

Exemple 11.7. Utilisation de la directive $INCLUDE

```
$INCLUDE /var/named/penguin.example.com
```

$ORIGIN

La directive **$ORIGIN** vous permet d'ajouter le nom de domaine aux enregistrements non qualifiés, comme ceux qui ont un nom d'hôte uniquement. Notez que l'utilisation de cette directive n'est pas nécessaire si la zone est spécifiée dans **/etc/named.conf**, puisque la zone est utilisée par défaut.

Dans Exemple 11.8, « Utilisation de la directive $ORIGIN », tous les noms utilisés dans les enregistrements de ressources ne se terminant pas par un point final (le signe **.**) se terminent par **example.com**.

Exemple 11.8. Utilisation de la directive $ORIGIN

```
$ORIGIN example.com.
```

$TTL

La directive **$TTL** vous permet de définir la valeur par défaut de *Time to Live* (TTL) pour la zone, autrement dit, combien de temps est un enregistrement de zone est valide. Chaque enregistrement de ressource peut avoir sa propre valeur TTL, qui se substitue à la cette directive.

Augmenter cette valeur permet aux serveurs de noms distants de mettre les informations de zone en cache pour une plus longue période, réduisant le nombre de requêtes pour la zone et allongeant ainsi l'intervalle de temps requis pour propager les modifications d'enregistrements de ressources.

Exemple 11.9. Utilisation de la directive $TTL

```
$TTL 1D
```

11.2.3.2. Enregistrements de ressources communs

Les enregistrements de ressources suivants sont couramment utilisés dans les fichiers de zone :

A

L'enregistrement *Address* indique l'adresse **IP** à assigner à un nom. Prend la forme suivante :

```
hostname IN A IP-address
```

Si la valeur *hostname* est omise, l'enregistrement pointera vers le dernier *hostname* indiqué.

Dans Exemple 11.10, « Utiliser un enregistrement de ressource A », les requêtes de **server1.example.com** pointent vers **10.0.1.3** ou **10.0.1.5**.

Exemple 11.10. Utiliser un enregistrement de ressource A

```
server1  IN  A  10.0.1.3
         IN  A  10.0.1.5
```

CNAME

L'enregistrement *Canonical Name* fait correspondre un nom à un autre. Pour cette raison, ce type d'enregistrement est souvent appelé *alias record*. Il prend la forme suivante :

```
alias-name IN CNAME real-name
```

Les enregistrements **CNAME** sont plus couramment utilisés pour pointer vers des services qui utilisent un schéma d'affectation de noms, comme **www** pour les serveurs Web. Cependant, il existe plusieurs restrictions à leur utilisation :

- Les enregistrements CNAME ne doivent pas pointer vers d'autres enregistrements CNAME, ceci, afin d'éviter des boucles à l'infini.
- Les enregistrements CNAME ne doivent pas contenir d'autres types de ressources (comme A, NS, MX, etc.), excepté pour les enregistrements DNSSEC (RRSIG, NSEC, etc.) quand la zone est signée.
- Les autres enregistrements de ressources qui pointent vers le nom de domaine complet (FQDN) d'un hôte (NS, MX, PTR) ne doivent pas pointer vers un enregistrement CNAME.

Dans Exemple 11.11, « Utiliser l'enregistrement de ressource CNAME », l'enregistrement **A** relie un nom d'hôte à une adresse **IP**, tandis que l'enregistrement **CNAME** y fait pointer le nom d'hôte commun **www**.

Exemple 11.11. Utiliser l'enregistrement de ressource CNAME

```
server1  IN  A      10.0.1.5
www      IN  CNAME  server1
```

MX

Le premier enregistrement *Mail Exchange* indique où le courrier envoyé à un espace nom particulier contrôlé par cette zone doit aller. Il est sous la forme suivante :

```
IN MX preference-value email-server-name
```

L'*email-server-name* est un nom de domaine complet (FQDN). La *preference-value* permet un classement numérique des serveurs e-mail pour un espace de noms, donnant la préférence à certains systèmes de courrier électronique sur d'autres. L'enregistrement de ressource **MX** avec la

plus faible *preference-value* sera privilégiée au détriment des autres. Toutefois, plusieurs serveurs de messagerie peuvent posséder la même valeur pour distribuer le trafic e-mail équitablement entre eux.

Dans Exemple 11.12, « Utilisation de l'enregistrement de ressource MX », le premier serveur d'emails **mail.example.com** sera préféré au serveur d'emails **mail2.example.com** pour recevoir des emails destinés au domaine **example.com**.

Exemple 11.12. Utilisation de l'enregistrement de ressource MX

```
example.com.  IN  MX  10  mail.example.com.
              IN  MX  20  mail2.example.com.
```

NS

L'enregistrement *Nameserver* indique les serveurs de noms qui font autorité pour une zone particulière. Il est sous la forme suivante :

```
ssh username@penguin.example.net
```

Le *nameserver-name* doit correspondre à un nom complet (FQDN). Notez que quand deux noms de serveurs sont répertoriés comme étant autoritatifs pour un domaine, le fait qu'ils soient des serveurs de noms secondaires ou si l'un d'entre eux est un serveur primaire n'est pas important. Ils sont tous deux considérés comme faisant référence d'autorité.

Exemple 11.13. Utiliser l'enregistrement de ressource NS

```
IN  NS  dns1.example.com.
IN  NS  dns2.example.com.
```

PTR

L'enregistrement *Pointer* pointe vers une autre partie de l'espace nom. Prend la forme suivante :

```
last-IP-digit IN PTR FQDN-of-system
```

La directive *last-IP-digit* correspond au dernier numéro d'une adresse **IP**, et le *FQDN-of-system* est un nom complet (FQDN).

Les enregistrements **PTR** sont principalement utilisés pour la résolution de noms inversés, car ils pointent vers des adresses **IP** qui renvoient à un nom particulier. Voir Section 11.2.3.4.2, « Un fichier de zone de résolution de noms inversés » pour obtenir des enregistrements **PTR** en cours d'utilisation.

SOA

L'enregistrement *Start of Authority* donne des informations importantes en matière d'autorité sur un espace nom ou un serveur de noms. Situé juste après la directive, c'est le premier enregistrement de ressource d'un fichier de zone. Prend la forme suivante :

```
@  IN  SOA  primary-name-server hostmaster-email (
         serial-number
```

```
        time-to-refresh
        time-to-retry
        time-to-expire
        minimum-TTL  )
```

Voici les directives :

- Le symbole @ met la directive **$ORIGIN** (ou le nom de zone si la directive **$ORIGIN** n'est pas définie) comme espace nom défini par cet enregistrement **SOA** de ressource.
- La directive *primary-name-server* est le nom d'hôte du serveur de noms primaire qui fait autorité pour ce domaine.
- La directive *hostmaster-email* correspond à l'email de la personne à contacter au sujet de l'espace nom.
- La directive *serial-number* est une valeur numérique incrémentée à chaque fois que le fichier de zone est altéré pour indiquer qu'il est temps que le service **named** charge la zone à nouveau.
- La directive *time-to-refresh* est une valeur numérique que les serveurs de noms secondaires utilisent pour déterminer combien de temps il faut attendre avant de demander au serveur de noms primaire si des changements ont été apportés à la zone.
- La directive *time-to-retry* est une valeur numérique utilisée par les serveurs de noms secondaires pour déterminer la durée à attendre avant d'émettre une demande d'actualisation, si le serveur de noms primaire ne répond pas. Si le serveur principal n'a pas répondu à une demande de rafraîchissement dans un délai spécifié dans la directive *time-to-expire*, les serveurs secondaires bloque d'autorité pour les demandes concernant cet espace de noms.
- Dans BIND 4 et 8, la directive *minimum-TTL* est la durée pendant laquelle les autres serveurs cachent les informations de la zone. Dans BIND 9, elle définit combien de temps les réponses négatives sont mises en cache. La mise en cache des réponses négatives peut être définie à un maximum de 3 heures (**3H**).

Lors de la configuration de BIND, toutes les heures sont spécifiées en secondes. Toutefois, il est possible d'utiliser des abréviations lorsque l'on indique des unités de temps autres que des secondes, comme des minutes (**M**), des heures (**H**), des jours (**D**) ou des semaines (**W**). Tableau 11.6, « Secondes comparées à d'autres unités de temps » affiche une durée en secondes et un temps équivalent dans un autre format.

Tableau 11.6. Secondes comparées à d'autres unités de temps

Secondes	Autres unités de temps
60	**1M**
1800	**30M**
3600	**1H**
10800	**3H**

Secondes	Autres unités de temps
21600	**6H**
43200	**12H**
86400	**1D**
259200	**3D**
604800	**1W**
31536000	**365D**

Exemple 11.14. Utilisation d'un enregistrement de ressource SOA

```
@  IN  SOA  dns1.example.com.  hostmaster.example.com. (
        2001062501  ; serial
        21600       ; refresh after 6 hours
        3600        ; retry after 1 hour
        604800      ; expire after 1 week
        86400 )     ; minimum TTL of 1 day
```

11.2.3.3. Balises de commentaires

En plus des enregistrements de ressources et des directives, un fichier de zone peut également contenir des commentaires. Les commentaires sont ignorés par le service **named**, mais peuvent s'avérer utiles pour fournir des renseignements supplémentaires à l'utilisateur. N'importe quel texte après le point-virgule, en fin de ligne, est considéré comme un commentaire. Exemple :

```
    604800  ; expire après 1 semaine
```

11.2.3.4. Exemple d'utilisation

Les informations suivantes expliquent l'élément que chaque option configure :

11.2.3.4.1. Fichier de zone simple

Exemple 11.15, « Fichier de zone simple » démontre l'utilisation de directives standards et de valeurs **SOA**.

Exemple 11.15. Fichier de zone simple

```
$ORIGIN example.com.
$TTL 86400
@         IN  SOA  dns1.example.com.  hostmaster.example.com. (
              2001062501  ; serial
```

```
                21600       ; refresh after 6 hours
                3600        ; retry after 1 hour
                604800      ; expire after 1 week
                86400 )     ; minimum TTL of 1 day
;
;
          IN  NS     dns1.example.com.
          IN  NS     dns2.example.com.
dns1      IN  A      10.0.1.1
          IN  AAAA   aaaa:bbbb::1
dns2      IN  A      10.0.1.2
          IN  AAAA   aaaa:bbbb::2
;
;
@         IN  MX     10  mail.example.com.
          IN  MX     20  mail2.example.com.
mail      IN  A      10.0.1.5
          IN  AAAA   aaaa:bbbb::5
mail2     IN  A      10.0.1.6
          IN  AAAA   aaaa:bbbb::6
;
;
; This sample zone file illustrates sharing the same IP addresses
; for multiple services:
;
services  IN  A      10.0.1.10
          IN  AAAA   aaaa:bbbb::10
          IN  A      10.0.1.11
          IN  AAAA   aaaa:bbbb::11

ftp       IN  CNAME  services.example.com.
www       IN  CNAME  services.example.com.
;
;
```

Dans cet exemple, les serveurs de noms autoritatifs sont définis en tant que **dns1.example.com** et **dns2.example.com**, et sont liés aux adressess **10.0.1.1** et **10.0.1.2 IP** respectivement en utilisant l'enregistrement **A**.

Les serveurs d'email configurés avec les enregistrements **MX** pointent vers **mail** et **mail2** via les enregistrements**A**. Comme ces noms ne se terminent pas par un point final, le domaine **$ORIGIN** est placé à leur suite, ce qui les étend à **mail.example.com** et à **mail2.example.com**.

Les services disponibles en noms standards, comme **www.example.com** (WWW), pointent vers les serveurs qui conviennent, en utilisant l'enregistrement **CNAME**.

Ce fichier de zone peut être mis en service avec un argument **zone** dans **/etc/named.conf** sous la forme suivante :

```
zone "example.com" IN {
  type master;
  file "example.com.zone";
  allow-update { none; };
};
```

11.2.3.4.2. Un fichier de zone de résolution de noms inversés

Un fichier de zone de résolution de noms inversés est utilisé pour traduire une adresse **IP** d'espace nom particulier en nom complet (FDNQ). Ce fichier ressemble beaucoup à un fichier de zone standard, sauf que les enregistrements de ressources **PTR** sont utilisés pour faire correspondre les adresses **IP** à un nom de domaine complet comme le montre Exemple 11.16, « Un fichier de zone de résolution de noms inversés ».

Exemple 11.16. Un fichier de zone de résolution de noms inversés

```
$ORIGIN 1.0.10.in-addr.arpa.
$TTL 86400
@  IN  SOA  dns1.example.com.  hostmaster.example.com. (
        2001062501  ; serial
        21600       ; refresh after 6 hours
        3600        ; retry after 1 hour
        604800      ; expire after 1 week
        86400 )     ; minimum TTL of 1 day
;
@  IN  NS   dns1.example.com.
;
1  IN  PTR  dns1.example.com.
2  IN  PTR  dns2.example.com.
;
5  IN  PTR  server1.example.com.
6  IN  PTR  server2.example.com.
;
3  IN  PTR  ftp.example.com.
4  IN  PTR  ftp.example.com.
```

Dans cet exemple, les adresses **IP** qui vont de **10.0.1.1** à **10.0.1.6** pointent vers le nom de domaine complet correspondant.

Ce fichier de zone peut être mis en service avec un argument de **zone** dans le fichier **/etc/named.conf** sous la forme suivante :

```
zone "1.0.10.in-addr.arpa" IN {
  type master;
  file "example.com.rr.zone";
  allow-update { none; };
};
```

Il n'y a guère de différence entre cet exemple et un argument de **zone** standard, à l'exception du nom de zone. Notez qu'une zone de résolution de noms inversés nécessite que les trois premiers blocs de l'adresse **IP** soient inversés, suivis par **.in-addr.arpa**. Cela permet à l'unique bloc de numéros **IP** utilisé dans le fichier de zone de résolution de noms inversés d'être associé à la zone.

11.2.4. Comment se servir de l'utilitaire rndc

L'utilitaire **rndc** est un outil de ligne de commandes qui vous permet d'administrer le service **named**, à la fois localement et à partir d'une machine éloignée. Son usage est le suivant :

```
rndc [option...] command [command-option]
```

11.2.4.1. Configuration de l'utilitaire

Pour éviter l'accès non autorisé au service, **named** doit être configuré pour écouter le port sélectionné (**953** par défaut), et une clé identique doit être utilisée par le service et l'utilitaire **rndc** à la fois.

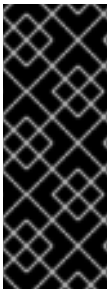
Tableau 11.7. Fichiers appropriés

Chemin	Description
/etc/named.conf	Le fichier de configuration par défaut du service **named**.
/etc/rndc.conf	Le fichier de configuration par défaut de l'utilitaire **rndc**.
/etc/rndc.key	L'emplacement de la clé par défaut

La configuration de **rndc** est située dans **/etc/rndc.conf**. Si le fichier n'existe pas, l'utilitaire utilisera la clé située dans **/etc/rndc.key**, qui a été générée automatiquement pendant l'installation par la commande **rndc-confgen-a**.

Le service **named** est configuré à l'aide de l'argument **controls** qui se trouve dans le fichier de configuration **/etc/named.conf** comme décrit dans Section 11.2.2.3, « Autres types d'arguments ». À moins que cet argument soit présent, seules les connexions de l'adresse de loopback (**127.0.0.1**) seront autorisées, et la clé qui se trouve dans **/etc/rndc.key** sera utilisée.

Pour plus d'informations à ce sujet, voir les pages man et le guide de référence *BIND 9 Administrator Reference Manual* qui se trouve dans Section 11.2.8, « Ressources supplémentaires ».



IMPORTANT

Pour empêcher les utilisateurs non privilégiés d'envoyer des commandes de contrôle au service, veillez à ce que seul l'utilisateur **root** soit autorisé à lire le fichier **/etc/rndc.key** :

```
~]# chmod o-rwx /etc/rndc.key
```

11.2.4.2. Vérifier le statut de service

Pour vérifier le statut actuel du service **named**, utiliser la commande suivante :

```
~]# rndc status
version: 9.7.0-P2-RedHat-9.7.0-5.P2.el6
CPUs found: 1
worker threads: 1
number of zones: 16
debug level: 0
xfers running: 0
xfers deferred: 0
soa queries in progress: 0
query logging is OFF
recursive clients: 0/0/1000
tcp clients: 0/100
server is up and running
```

-

11.2.4.3. Configuration en ligne de commandes

La mise à jour d'un paquetage est semblable à son installation. Entrez la commande suivante à l'invite du shell :

```
~]# rndc reload
server reload successful
```

Cette commande téléchargera à nouveau les zones tout en conservant toutes les réponses mises en cache, de façon à ce que vous puissiez effectuer des changements à des fichiers de zone sans perdre toutes les résolutions de noms stockées.

Pour télécharger à nouveau une zone particulière, indiquer son nom à la suite de la commande **reload**, comme par exemple :

```
~]# rndc reload localhost
zone reload up-to-date
```

Enfin, pour charger à nouveau le fichier de configuration et les zones nouvellement ajoutées, saisissez :

```
~]# rndc reconfig
```

NOTE

Si vous souhaitez modifier une zone qui utilise un **DNS** Dynamique (DDNS), veillez à exécuter la commande **freeze** pour commenncer :

```
~]# rndc freeze localhost
```

Quand vous aurez terminé, exécuter la commande **thaw** pour autoriser **DDNS** à nouveau, et charger la zone à nouveau.

```
~]# rndc thaw localhost
The zone reload and thaw was successful.
```

11.2.4.4. Mise à jour des clés de zone

Pour metter à jour les clés DNSSEC et signer la zone, utiliser la commande **sign**. Exemple :

```
~]# rndc sign localhost
```

Notez que pour signer une zone avec la commande ci-dessus, l'option **auto-dnssec** doit être définie sur **maintain** dans l'argument de la zone. Exemple :

```
zone "localhost" IN {
  type master;
  file "named.localhost";
  allow-update { none; };
  auto-dnssec maintain;
};
```

11.2.4.5. Activation de la validation DNSSEC

Pour activer la valisation DNSSEC, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# rndc validation on
```

De même, pour désactiver cette option, saisissez :

```
~]# rndc validation off
```

Voir l'argument **options** décrit dans Section 11.2.2.2, « Types d'arguments communs » pour obtenir des informations sur la façon de configurer cette option dans **/etc/named.conf**.

Le guide *Red Hat Enterprise Linux 7 Security Guide* a une section bien complète sur DNSSEC.

11.2.4.6. Activation de la journalisation des requêtes

Pour activer (ou désactiver si elle est déjà activée) la journalisation des requêtes, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# rndc querylog
```

Pour vérifier la configuration en cours, utilisez la commande **status** décrite dans Section 11.2.4.2, « Vérifier le statut de service ».

11.2.5. Utilisation de l'utilitaire dig

L'utilitaire **dig** est un outil de ligne de commandes qui vous permet de faire des consultations **DNS** et de déboguer une configuration de serveur de noms. Son usage est le suivant :

```
dig [@server] [option...] name type
```

Voir Section 11.2.3.2, « Enregistrements de ressources communs » pour obtenir une liste de valeurs communes à utiliser avec *type*.

11.2.5.1. Rechercher un serveur de noms

Pour chercher un serveur de noms particulier, utiliser la commande sous la forme suivante :

```
dig name NS
```

Dans Exemple 11.17, « Exemple de recherche de serveur de noms », l'utilitaire **dig** est utilisé pour afficher les serveurs de noms de **example.com**.

Exemple 11.17. Exemple de recherche de serveur de noms

```
~]$ dig example.com NS

; <<>> DiG 9.7.1-P2-RedHat-9.7.1-2.P2.fc13 <<>> example.com NS
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 57883
```

```
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;example.com.                   IN      NS

;; ANSWER SECTION:
example.com.            99374   IN      NS      a.iana-servers.net.
example.com.            99374   IN      NS      b.iana-servers.net.

;; Query time: 1 msec
;; SERVER: 10.34.255.7#53(10.34.255.7)
;; WHEN: Wed Aug 18 18:04:06 2010
;; MSG SIZE  rcvd: 77
```

11.2.5.2. Recherche d'une adresse IP

Pour chercher une adresse **IP** assignée à un domaine en particulier, utiliser la commande sous la forme suivante :

```
dig name A
```

Dans Exemple 11.18, « Exemple de recherche d'adresse IP », l'utilitaire **dig** est utilisé pour afficher l'adresse **IP** de **example.com**.

Exemple 11.18. Exemple de recherche d'adresse IP

```
~]$ dig example.com A

; <<>> DiG 9.7.1-P2-RedHat-9.7.1-2.P2.fc13 <<>> example.com A
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 4849
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 0

;; QUESTION SECTION:
;example.com.                   IN      A

;; ANSWER SECTION:
example.com.            155606  IN      A       192.0.32.10

;; AUTHORITY SECTION:
example.com.            99175   IN      NS      a.iana-servers.net.
example.com.            99175   IN      NS      b.iana-servers.net.

;; Query time: 1 msec
;; SERVER: 10.34.255.7#53(10.34.255.7)
;; WHEN: Wed Aug 18 18:07:25 2010
;; MSG SIZE  rcvd: 93
```

11.2.5.3. Recherche d'un nom d'hôte

Pour rechercher un nom d'hôte pour une adresse **IP** particulière, utiliser la commande sous la forme suivante :

```
dig -x address
```

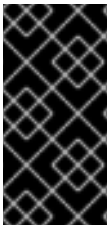
Dans Exemple 11.19, « Exemple de recherche de nom d'hôte », l'utilitaire **dig** est utilisé pour afficher le nom d'hôte assigné à **192.0.32.10**.

Exemple 11.19. Exemple de recherche de nom d'hôte

```
~]$ dig -x 192.0.32.10

; <<>> DiG 9.7.1-P2-RedHat-9.7.1-2.P2.fc13 <<>> -x 192.0.32.10
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 29683
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 5, ADDITIONAL: 6

;; QUESTION SECTION:
;10.32.0.192.in-addr.arpa.      IN      PTR

;; ANSWER SECTION:
10.32.0.192.in-addr.arpa. 21600 IN      PTR     www.example.com.

;; AUTHORITY SECTION:
32.0.192.in-addr.arpa.  21600   IN      NS      b.iana-servers.org.
32.0.192.in-addr.arpa.  21600   IN      NS      c.iana-servers.net.
32.0.192.in-addr.arpa.  21600   IN      NS      d.iana-servers.net.
32.0.192.in-addr.arpa.  21600   IN      NS      ns.icann.org.
32.0.192.in-addr.arpa.  21600   IN      NS      a.iana-servers.net.

;; ADDITIONAL SECTION:
a.iana-servers.net.     13688   IN      A       192.0.34.43
b.iana-servers.org.     5844    IN      A       193.0.0.236
b.iana-servers.org.     5844    IN      AAAA    2001:610:240:2::c100:ec
c.iana-servers.net.     12173   IN      A       139.91.1.10
c.iana-servers.net.     12173   IN      AAAA    2001:648:2c30::1:10
ns.icann.org.           12884   IN      A       192.0.34.126

;; Query time: 156 msec
;; SERVER: 10.34.255.7#53(10.34.255.7)
;; WHEN: Wed Aug 18 18:25:15 2010
;; MSG SIZE  rcvd: 310
```

11.2.6. Fonctionnalités avancées de BIND

La plupart des implémentations de BIND utilisent seulement le service **named** pour fournir des services de résolution de nom ou pour agir comme autorité pour un domaine particulier. Cependant, la version BIND 9 a un certain nombre de fonctionnalités avancées qui permettent un service **DNS** plus sûr et plus efficace.



IMPORTANT

Avant d'utiliser des fonctionnalités avancées comme DNSSEC, TSIG ou IXFR (transfert de zone incrémentiel), assurez-vous que la fonctionnalité en question est prise en charge par tous les serveurs de noms dans l'environnement réseau, surtout lorsque vous utilisez des versions plus anciennes de serveurs de liaison (BIND) ou des serveurs non-BIND.

Toutes les fonctionnalités mentionnées sont expliquées en détails dans le manuel *BIND 9 Administrator Reference Manual* qui se trouve dans Section 11.2.8.1, « Documentation installée ».

11.2.6.1. Vues multiples

Éventuellement, des informations différentes peuvent être présentées à un client selon le réseau de provenance de la demande. Ceci est principalement utilisé pour refuser l'accès à des données sensibles **DNS** de la part de clients se trouvant à l'extérieur du réseau local, tout en permettant aux requêtes des clients à l'intérieur du réseau local.

Pour configurer plusieurs affichages, ajoutez l'argument **view** dans le fichier de configuration **/etc/named.conf**. Utilisez l'option de **match-clients** pour faire correspondre les adresses **IP** ou des réseaux dans leur ensemble et leur donner des options spéciales et les données de zone.

11.2.6.2. IXFR (Incremental Zone Transfers)

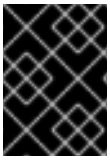
Incremental Zone Transfers (*IXFR*) permet à un serveur de noms secondaire de télécharger les portions mises à jour d'une zone modifiée sur un serveur de noms primaire. Par rapport à un processus de transfert standard, cela rend le processus de notification et de mise à jour bien plus efficace.

Veuillez noter qu'IXFR n'est disponible qu'en utilisant la mise à jour dynamique pour effectuer des changements dans les enregistrements de zones du master. Pour modifier des fichiers de zone manuellement, *Automatic Zone Transfer* (*AXFR*) est utilisé.

11.2.6.3. TSIG (Transaction SIGnatures)

Transaction SIGnatures (TSIG) veille à ce qu'une clé secrète partagée existe à la fois sur des serveurs de noms primaires et secondaires avant d'autoriser un transfert. Cela renforce la méthode axée sur l'adresse **IP** standard d'autorisation de transfert, dans la mesure où les attaquants auraient non seulement besoin d'avoir accès à l'adresse **IP** de la zone de transfert, mais il faudrait aussi qu'ils connaissent la clé secrète.

Depuis la version 9, BIND prend également *TKEY* en charge, une autre méthode de clés secrêtes partagées pour autoriser les transferts de zone.



IMPORTANT

Quand on communique sur réseau non sécurisé, ne pas se fier à l'authentification **IP** basée adresse uniquement.

11.2.6.4. DNSSEC (DNS Security Extensions)

Domain Name System Security Extensions (*DNSSEC*) fournit une authentification d'origine aux données **DNS**, le déni d'existence authentifié, et l'intégrité des données. Quand un domaine particulier est marqué comme sécurisé, la réponse **SERVFAIL** est retournée pour chaque enregistrement de ressource qui échoue au niveau validation.

Notez que pour déboguer un domaine de signature DNSSEC ou un résolveur DNSSEC-aware, vous pouvez utiliser l'utilitaire de **dig** comme décrit dans Section 11.2.5, « Utilisation de l'utilitaire dig ». Options utiles : **+ dnssec** (demande d'enregistrements de ressources liées à DNSSEC en définissant le DNSSEC OK bit), **+ cd** (indique au serveur de noms récursif de ne pas valider la réponse), et **+ bufsize = 512** (modifie la taille du paquet à 512 octets pour pouvoir passer à travers certains pare-feux).

11.2.6.5. IPv6 (Internet Protocol version 6)

Internet Protocol version 6 (*IPv6*) est pris en charge par l'utilisation des enregistrements de ressources **AAAA**, et la directive **listen-on-v6** décrite dans Tableau 11.3, « Options de configuration souvent utilisées ».

11.2.7. Erreurs communes à éviter

Voici une liste des recommandations pour éviter les erreurs que les utilisateurs font souvent quand ils configurent un serveur de noms :

Utiliser les points virgule et les crochets courbes correctement

Un point virgule ou un crochet courbe qui ne correspond pas, dans le fichier **/etc/named.conf**, peut empêcher le démarrage du service **named**.

Utiliser le point final (le signe .) correctement.

Dans les fichiers de zone, un point virgule en fin de nom de domaine indique un nom de domaine complet. Si vous l'omettez, le service **named** ajoutera le nom de la zone ou la valeur correspondant à **$ORIGIN** pour compléter le nom de domaine.

Incrémenter le numéro de série lorsque vous modifiez un fichier de zone

Si le numéro de série n'est pas incrémenté, le serveur de noms primaire aura la nouvelle information correcte, mais les noms de serveurs ne seront jamais notifiés du changement, et ne tenteront pas de réactualiser leurs données dans cette zone.

Fichier de configuration

Si un parefeu bloque des connexions d'un service **named** à d'autres serveurs de noms, la pratique conseillée est de modifier les paramètres de configuration du parefeu.

AVERTISSEMENT

L'utlisation d'un port source **UDP** pour les recherches **DNS** représente un danger potentiel de sécurité car cela permettrait à un attaquant de conduire des attaques d'empoisonnement de cache plus facilement. Pour éviter cela, par défaut, **DNS** envoie un port éphémère. Configurer votre parefeu pour autoriser les demandes aléatoires sortantes de ports source **UDP**. Une plage de **1024** à **65535** est utilisée par défaut.

11.2.8. Ressources supplémentaires

Les sources d'informations suivantes fournissent des ressources supplémentaires à propos de BIND.

11.2.8.1. Documentation installée

BIND comprend un grand nombre de documentations installées couvrant plusieurs sujets, chaque sujet étant placé dans son propre répertoire de sujet. Pour chaque élément ci-dessous, remplacer la *version* par la version du package bind installée sur le système :

`/usr/share/doc/bind-version/`

Le répertoire principal contenant la documentation la plus récente. Le répertoire contient le manuel *BIND 9 Administrator Reference Manual* au format HTML et PDF, qui détaille les ressources BIND nécessaires, comment configurer différents types de serveurs de noms, comment effectuer l'équilibrage des charges et autres sujets avancés.

`/usr/share/doc/bind-version/sample/etc/`

Le répertoire contient des exemples de fichiers de configuration **named**.

`rndc(8)`

La page man de l'utilitaire de contrôle de serveur de noms **rndc** contient une documentation sur son utilisation.

`named(8)`

La page man du serveur de noms de domaine Internet **named** qui contient une documentation sur un assortiement d'arguments pouvant être utilisés pour contrôler le démon du serveur de noms BIND.

`lwresd(8)`

Page man pour le démon du programme de résolution léger **lwresd** (de l'anglais lightweight resolver daemon), qui contient la documentation sur le démon et sur son utilisation.

`named.conf(5)`

Page man avec une liste complète d'options disponibles dans le fichier de configuration de **named**.

`rndc.conf(5)`

Page man comprenant une liste d'options disponibles pour le fichier de configuration de **rndc**.

11.2.8.2. Ressources en ligne

https://access.redhat.com/site/articles/770133

Un article de la base de connaissances de Red Hat sur l'exécution de BIND dans un environnement **chroot**, comprenant les différences par rapport à Red Hat Enterprise Linux 6.

https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Security_Guide/

Le guide *Red Hat Enterprise Linux 7 Security Guide* a une bonne section sur DNSSEC.

https://www.icann.org/namecollision

ICANN FAQ on domain name collision.

CHAPITRE 12. SQUID

Ce chapitre traite de **Squid**, un serveur de mise en cache de proxy de haute performance pour les clients web. Dans cette section, vous pouvez lire comment configurer **Squid**, comment authentifier, et bloquer l'accès avec **Squid**.

12.1. INTRODUCTION À SQUID

Squid est un serveur web proxy qui optimise le fonctionnement du site Web en mettant les pages en cache afin qu'elles soient chargées plus rapidement, ce qui améliore le temps de réponse pour les pages que les utilisateurs visitent le plus fréquemment. **Squid** fournit des services de proxy et de cache pour le protocole Hypertext Transport Protocol (HTTP), le protocole FTP (File Transfer) et d'autres protocoles populaires. **Squid** est surtout utilisé pour accélérer un serveur web en mettant en cache des demandes répétées, aidant ainsi la sécurité par filtrage du trafic ou pour limiter l'accès des utilisateurs à des pages spécifiques

Squid supporte les objets de données **FTP**, **gopher**, **ICAP**, **ICP**, **HTCP**, et **HTTP**.

Squid se compose :

- d'un programme de serveur principal **Squid**
- de programmes optionnels de traitement et d'authentification personnalisés
- d'outils de gestion et d'outils clients

12.2. INSTALLATION ET EXÉCUTION DE SQUID

Dans Red Hat Enterprise Linux, le package squid fournit le serveur proxy de mise en cache **Squid**. Exécuter la commande **rpm -q squid** pour voir si le package squid est installé. Sinon, exécuter la commande suivante en tant qu'utilisateur **root** pour l'installer :

```
~]# yum install squid
```

Exécuter la commande **systemctl start squid** en tant qu'utilisateur **root** pour démarrer **Squid** :

```
~]# systemctl start squid
```

Squid va maintenant commencer à écouter le port 3128 (port par défaut) sur toutes les interfaces de réseau de la machine.

Exécuter la commande **systemctl start squid** pour confirmer que **Squid** est en cours d'exécution. En voici un exemple :

```
~]# systemctl status squid
● squid.service - Squid caching proxy
   Loaded: loaded (/usr/lib/systemd/system/squid.service; disabled; vendor
preset: disabled)
   Active: active (running) since Wed 2016-04-06 13:15:05 CEST; 2min 17s
ago
[output truncated]
```

Exécuter la commande **ps -eZ | grep squid** pour afficher les processus de **Squid** :

```
~]# ps -eZ | grep squid
unconfined_u:system_r:squid_t:s0 2522 ?        00:00:00 squid
unconfined_u:system_r:squid_t:s0 2524 ?        00:00:00 squid
unconfined_u:system_r:squid_t:s0 2526 ?        00:00:00 ncsa_auth
unconfined_u:system_r:squid_t:s0 2527 ?        00:00:00 ncsa_auth
unconfined_u:system_r:squid_t:s0 2528 ?        00:00:00 ncsa_auth
unconfined_u:system_r:squid_t:s0 2529 ?        00:00:00 ncsa_auth
unconfined_u:system_r:squid_t:s0 2530 ?        00:00:00 ncsa_auth
unconfined_u:system_r:squid_t:s0 2531 ?        00:00:00 unlinkd
```

Si vous êtes interressés par les statistiques de **Squid** en environnement de ligne de commande, utiliser l'outil **squidclient** qui peut accéder au service **Squid** et extraire des statistiques. Ainsi, pour obtenir des statistiques sur la performane en général, exécuter la commande suivante sur le serveur **Squid** :

```
~]# squidclient -p squid-port mgr:info
```

Pour stopper **Squid**, exécutez cette commande :

```
~]# systemctl stop squid
```

Fichiers journaux Squid

Les fichiers journaux du serveur proxy de **Squid** se trouvent dans le répertoire **/var/log/squid/**. Le fichier journal qui stocke des informations sur les requêtes proxy se trouve dans le fichier **/var/log/squid/access.log**.

12.3. CONFIGURATION SQUID

Pour configurer **Squid**, ajuster les directives dans le fichier de configuration. **Squid** est normalement configuré selon les prérequis d'un réseau donné en ligne de commande et en modifiant le fichier de configuration **Squid** situé dans **/etc/squid/squid.conf**, qui contient le minimum de configurations conseillées.

12.3.1. Configuration de base et /etc/squid/squid.conf

Procédure 12.1. Configuration de base

1. Sauvegarde du fichier de configuration original.

   ```
   mv /etc/squid/squid.conf /etc/squid/squid.conf.org
   ```

2. Créer un nouveau fichier **/etc/squid/squid.conf** avec le contenu suivant. Modifier l'ACL (de l'anglais Access Control List) à la ligne *mynetwork* pour définir le réseau d'origine pour votre réseau local. Il s'agit du réseau où les systèmes client utilisent le serveur de **Squid** comme proxy.

   NOTE

   L'ordre des éléments qui se trouvent dans le fichier de configuration **/etc/squid/squid.conf** est important car **Squid** le lit à partir du début.

   ```
   acl mynetwork src xxx.xxx.xxx.0/24
   ```

```
http_access allow mynetwork

#defaults
acl localnet src 10.0.0.0/8
acl localnet src 172.16.0.0/12
acl localnet src 192.168.0.0/16
acl localnet src fc00::/7
acl localnet src fe80::/10
acl SSL_ports port 443
acl Safe_ports port 80
acl Safe_ports port 21
acl Safe_ports port 443
acl Safe_ports port 70
acl Safe_ports port 210
acl Safe_ports port 1025-65535
acl Safe_ports port 280
acl Safe_ports port 488
acl Safe_ports port 591
acl Safe_ports port 777
acl CONNECT method CONNECT
http_access allow manager localhost
http_access deny manager
http_access deny !Safe_ports
http_access deny CONNECT !SSL_ports
http_access allow localnet
http_access allow localhost
http_access deny all
http_port 3128
hierarchy_stoplist cgi-bin ?
coredump_dir /var/spool/squid
refresh_pattern ^ftp:       1440    20% 10080
refresh_pattern ^gopher:    1440    0%  1440
refresh_pattern -i (/cgi-bin/|\?) 0 0%  0
refresh_pattern .       0   20% 4320
```

3. Démarrer le service et l'activer au démarrage :

```
~]#  systemctl enable squid
~]#  systemctl start squid
```

4. Si le parefeu est activé, autoriser le port de **Squid**.

```
~]#  firewall-cmd --add-port=3128/tcp --permanent
```

5. Configurer votre navigateur web pour qu'il utilise le serveur proxy. Cela dépendra du navigateur et de la version utilisés. Ainsi, pour configurer **Firefox** version 46.0.0 :

Procédure 12.2. Configurer Firefox avec le serveur Proxy

1. Dans le menu Firefox situé en haut à droite, sélectionner **Avancé**, puis **Réseau** à partir des onglets, puis, sélectionner **Réseau** à partir des onglets situés en haut sur la barre de navigaiton.

2. Dans la section **Connexion**, ouvrir **Paramètres de configuration**.

3. Dans la nouvelle fenêtre qui vient de s'ouvrir, cochez **Configuration de serveur proxy manuelle** et saisir le serveur proxy auquel vous êtes connecté dans le champ **Proxy HTTP**. Si vous avez besoin de saisir un port particulier, le saisir dans le champ **Port**.

Pour plus d'informations sur **/etc/squid/squid.conf**, voir la page man **squid(8)**.

12.3.2. Configurer Squid en tant que serveur proxy HTTP

Procédure 12.3. Configurer Squid en tant que serveur proxy HTTP

1. Ajouter les lignes suivantes en haut du fichier **/etc/squid/squid.conf** pour remplacer l'exemple d'adresse IP :

```
cache_dir ufs /var/spool/squid 500 16 256
acl my_machine src 192.0.2.21 # Replace with your IP address
http_access allow my_machine
```

2. Créerdes répertoires cache par la commande suivante :

```
~]# systemctl restart squid
```

Squid commence maintenant à écouter le port 3128 (port par défaut) sur toutes les interfaces de réseau de la machine.

3. Configurer votre navigateur, par exemple **Firefox**, afin d'utiliser **Squid** en tant que serveur proxy HTTP avec l'hôte comme adresse IP de la machine et le port 3128: pour plus d'informations, voir Procédure 12.2, « Configurer Firefox avec le serveur Proxy »

12.3.2.1. Configuration du port HTTP

La directive **http_port** est utilisée pour spécifier le port où **Squid** va pouvoir écouter les connexions. Le comportement par défaut est d'écouter le port 3128 sur toutes les interfaces de la machine. Vous pouvez forcer **Squid** à écouter plusieurs interfaces sur différents ports, sur différentes interfaces.

Exemple 12.1. Spécifier le port HTTP

Ouvrir **/etc/squid/squid.conf** et modifier la ligne qui convient. Dans cet exemple, **Squid** est configuré pour écouter le port 8080.

```
# Squid normally listens to port 3128
http_port 8080
```

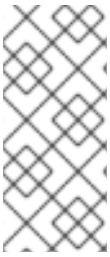
Le serveur **Squid** peut écouter différents ports en même temps.

Exemple 12.2. Spécifier deux ports ou davantage

Avec la configuration actuelle, **Squid** écoute à la fois le port 8080 et le port 9090:

```
http_port 8080 9090
```

-



NOTE

N'oubliez pas de démarrer à nouveau le serveur de **Squid** pour appliquer les nouvelles configurations :

```
~]# systemctl restart squid
```

Vous pouvez également spécifier la combinaison port et adresse IP dans **/etc/squid/squid.conf**. Normalement, cette approche est utilisée lorsque vous avez plusieurs interfaces sur la machine et que **Squid** écoute uniquement l'interface connectée à un réseau local (LAN).

Exemple 12.3. Configurer les adresses IP

Les commandes suivantes ordonnent à **Squid** d'écouter le port 3128 sur l'interface ayant pour adresse IP 192.0.2.25:

```
http_port 192.0.2.25:3128
```

De plus, vous pouvez spécifier **http_port** en utilisant une combinaison *nom d'hôte* et *port*. Le nom d'hôte sera traduit en adresse IP par **Squid**, qui écoutera alors sur le port 8080 sur cette adresse IP particuliére.

```
http_port myproxy.example.com:8080
```

Un autre aspect de la directive **http_port** est que cela peut prendre plusieurs valeurs sur des lignes distinctes. Les lignes suivantes déclencheront **Squid** pour écouter trois combinaisons d'adresses IP / port différentes. C'est généralement utile lorsque vous avez des clients dans différents réseaux locaux (LAN) configurés pour utiliser des ports différents pour le serveur proxy. Modifiez le fichier **/etc/squid/squid.conf** comme suit :

```
http_port 192.0.2.25:8080
http_port lan1.example.com:3128
http_port lan2.example.com:8081
```

12.3.2.2. Contrôle d'accès HTTP et ACL

Les listes des contrôle d'accès (ACL) sont les éléments de base du contrôle d'accès et sont normalement utilisées en combinaison à d'autres directives, comme **http_access**, pour contrôler l'accès à divers composants de **Squid** ou ressources web.

Exemple 12.4. Contruire un ACL pour un nom de domaine

Cet exemple vous montre comment éditer l'instruction (générale) suivante :

```
acl example_site dstdomain example.com
```

comme suit. Donnez un nom à votre ACN en remplaçant *example_site* par un nom. Le type utilisé ici est **dstdomain**, qui indique que la valeur (le site web) est un nom de domaine.

```
acl FB dstdomain facebook.com
```

Si vous devez construire un ACL pour un certain nombre de sites web, vous pouvez :

- Écrire les valeurs en une seule ligne :

```
acl example_sites dstdomain example.com example.net example.org
```

- Écrire les valeurs sur plusieurs lignes si les valeurs augmentent beaucoup :

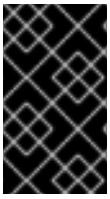
```
acl example_sites dstdomain example.com example.net
acl example_sites dstdomain example.org
```

- Vous pouvez mettre les valeurs dans un fichier dédié à ce but, puis instruire **Squid** de lire les valeurs de ce fichier :

```
acl example_sites dstdomain '/etc/squid/example_sites.txt'
```

Le contenue de **/etc/squid/example_sites.txt** ressemble à ceci :

```
# Write one value (domain name) per line
example.net
example.org # Temporarily remove example.org from example_sites acl
example.com
```



IMPORTANT

Les ACL doivent être combinés à des directives de contrôle d'accès pour permettre ou interdire l'accès à certaines ressources. **http_access** est une de ces directives utilisées pour donner l'accès en vue d'effectuer des transactions HTTP via **Squid** :

Contrôler l'accès HTTP par les ACL
Pour autoriser ou interdire l'accès aux clients, vous devez combiner les ACL à la directive **http_access**.

Dans le fichier **/etc/squid/squid.conf**, modifiez la directive **http_access**, où *ACL_NAME* indique les ressources auxquelles on peut accéder ou non :

```
http_access allow|deny [!]ACL_NAME
```

Exemple 12.5. Autoriser ou refuser l'accès aux clients

Les paramètres de configuration suivants donnent accès à l'hôte local :

```
http_access allow localhost
```

Cette configuration refuse l'accès à l'hôte local :

```
http_access deny localhost
```

Certains noms d'ACL commencent par un point d'exclamation. Si tel est le cas, inclure le point d'exclamation également :

```
http_access deny !Safe_ports
```

12.4. AUTHENTIFICATION SQUID

Pour l'authentification, le code source de **Squid** connecte avec quelques back-ends d'authentification, également appelés **helpers**, tels que SMB (serveur SMB comme Windows NT ou Samba), DB (une base de données SQL), ou LDAP (Lightweight Directory Access Protocol). Les utilisateurs sont authentifiés si **Squid** est configuré pour utiliser les ACL **proxy_auth**.

Indiquer à **Squid** quel programme helper d'authentification utiliser avec une directive **auth_param** dans **/etc/squid/squid.conf**. Spécifier le nom du programme et les options en ligne de commande si nécessaire.

```
auth_param scheme parameter [setting]
```

Exemple 12.6. Ajout d'ACL proxy_auth

Ajouter des entrées d'ACL **proxy_auth** à votre configuration **Squid** en indiquant les noms d'utilisateurs. Dans cet exemples, les utilisateurs nommés lisa, sarah, joe, et frank sont autorisés d'utiliser le proxy à tout moment. Les autres utilisateurs ne peuvent le faire que pendant la journée.

```
acl foo proxy_auth REQUIRED
acl bar proxy_auth lisa sarah frank joe
acl daytime time 08:00-17:00
http_access allow foo daytime
http_access allow bar
http_access deny all
```

12.4.1. Authentication avec LDAP

Dans cette installation, **Squid** utilise LDAP pour authentifier les utilisateurs avant de les autoriser à surfer sur l'internet. Le code source **Squid** connecte à un back-end d'authentification (LDAP). Les utilisateurs doivent ensuite saisir leur nom d'utilisateur et mot de passe avant de pouvoir continuer sur les pages web. **Squid** utilise l'assistant d'authentification LDAP de **Squid**, **squid_ldap_auth**, ce qui permet à **Squid** de se connecter à un répertoire LDAP afin de valider le nom d'utilisateur et le mot de passe pour une authentication HTTP de base.

Mofifier le fichier **/etc/squid/squid.conf** comme suit afin de pouvoir connecter **Squid** à ldap.example.com:

```
auth_param basic program /usr/lib64/squid/basic_ldap_auth -b
"dc=example,dc=com" -f "uid=%s" -c 2 -t 2 -h ldap.example.com
otherldap.example.com
```

Si vous souhaitez authentifier des utilisateur **Squid** sur un serveurLDAP via un canal SSL/TLS sécurisé, passer l'argument **-ZZ** au programme **squid_ldap_auth**.

```
auth_param basic program /usr/lib64/squid/basic_ldap_auth -v 3 -ZZ -b
"dc=yourcompany,dc=com" -D uid=some-user,ou=People,dc=yourcompany,dc=com
-w password -f uid=%s ldap.yourcompany.com
```

Si vous souhaitez vous authentifier auprès de serveurs OpenLDAP, comme TLS et SSL, vous devrez spécifier **auth_param** dans le fichier **/etc/squid/squid.conf**:

1. Modifier **/etc/squid/squid.conf** pour TLS :

   ```
   auth_param basic program /usr/lib64/squid/basic_ldap_auth -Z -b
   "dc=example,dc=com" -f "uid=%s" -c 2 -t 2 -h ldap.example.com
   ```

   et pour SSL :

   ```
   auth_param basic program /usr/lib64/squid/basic_ldap_auth -b
   "dc=example,dc=com" -f "uid=%s" -c 2 -t 2 -H
   ldaps://ldap.example.com
   ```

   Quand

   ```
   -b - Specifies the base DN under which the users are located.
   -f - Specifies LDAP search filter to locate the user DN.
   -c - Specifies timeout used when connecting to LDAP servers.
   -t - Specifies time limit on LDAP search operations.
   -h - Specifies the LDAP server to connect to.
   -H - Specities the LDAP server to connect to by LDAP URI
   ```

2. Redémarrer le service **Squid**

   ```
   ~]# systemctl restart squid
   ```

12.4.2. Authentication avec Kerberos

Suivez la procédure pour configurer le proxy **Squid** dans Red Hat Enterprise Linux 7 afin d'utiliser l'authentification **Kerberos**. De plus, comme prérequis, commencez par installer Samba, le serveur de fichiers Common Internet File System (CIFS) pour Red Hat Enterprise Linux. Pour plus d'informations sur l'installation de Samba, voyez la section Samba dans le Guide de l'administrateur systèmes de Red Hat Enterprise Linux 7.

Procédure 12.4. Configurer Squid dans Red Hat Enterprise Linux 7 pour utiliser l'authentification Kerberos

1. Configurer **Squid** pour rejoindre un domaine AD (Active Directory).

   1. Modifier le fichier **/etc/krb5.conf** :

      ```
      [libdefaults]
               default_realm = EXAMPLE.COM
               dns_lookup_kdc = no
               dns_lookup_realm = no
               default_keytab_name = /etc/krb5.keytab
          ; for Windows 2003
               default_tgs_enctypes = rc4-hmac des-cbc-crc des-cbc-md5
      ```

```
          default_tkt_enctypes = rc4-hmac des-cbc-crc des-cbc-md5
          permitted_enctypes = rc4-hmac des-cbc-crc des-cbc-md5

    ; for Windows 2008 with AES
    ;      default_tgs_enctypes = aes256-cts-hmac-sha1-96 rc4-
hmac des-cbc-crc des-cbc-md5
    ;      default_tkt_enctypes = aes256-cts-hmac-sha1-96 rc4-
hmac des-cbc-crc des-cbc-md5
    ;      permitted_enctypes = aes256-cts-hmac-sha1-96 rc4-hmac
des-cbc-crc des-cbc-md5

    [realms]
     EXAMPLE.COM = {
      kdc = 192.168.0.1
      admin_server = 192.168.0.1
     }

    [domain_realm]
     example.com = EXAMPLE.COM
     .example.com = EXAMPLE.COM

    [logging]
      kdc = FILE:/var/log/kdc.log
      admin_server = FILE:/var/log/kadmin.log
      default = FILE:/var/log/krb5lib.log
```

2. Vérifier par la commande **kinit** :

```
~]# kinit testuser1
```

```
~]# kinit administrator
```

3. Modifier le fichier **/etc/samba/smb.conf** comme suit :

```
[global]
workgroup = EXAMPLE
password server = 192.168.0.1
# Remember to put the realm all in CAPS:
realm = EXAMPLE.COM
security = ads
idmap uid = 16777216-33554431
idmap gid = 16777216-33554431
template shell = /bin/bash
winbind use default domain = true
winbind offline logon = false
winbind enum users = yes
winbind enum groups = yes
encrypt passwords = yes
log file = /var/log/samba/log.%m
max log size = 50
passdb backend = tdbsam
load printers = yes
cups options = raw
kerberos method = system keytab
```

4. Rejoindre le domaine AD

```
~]# net ads join -U Administrator
```

2. Créer un onglet pour HTTP/fqdn par la commande **net ads keytab**

```
~]# kinit administrator
~]# export KRB5_KTNAME=FILE:/etc/squid/HTTP.keytab
~]# net ads keytab CREATE
~]# net ads keytab ADD HTTP
```

et vérifier le fichier *keytab*

```
~]# klist -k /etc/squid/HTTP.keytab
```

NOTE

Veillez à ce que le nom d'hôte soit bien défini dans le fichier **/etc/hosts**

3. Les fichiers doivent être inclus dans **Squid**.

```
~]# rpm -q squid
    squid-3.1.10-1.el6.x86_64
```

```
~]# rpm -ql squid | grep kerb
```

```
/usr/lib64/squid/negotiate_kerberos_auth
    /usr/lib64/squid/negotiate_kerberos_auth_test
    /usr/lib64/squid/squid_kerb_auth
    /usr/lib64/squid/squid_kerb_auth_test
```

4. Modifier **/etc/squid/squid.conf** comme suit

```
auth_param negotiate program /usr/lib64/squid/squid_kerb_auth -d -s
HTTP/squid.example.com@EXAMPLE.COM
auth_param negotiate children 10
auth_param negotiate keep_alive on
acl kerb_auth proxy_auth REQUIRED
(content truncated)

http_access allow kerb_auth
http_access allow manager localhost
http_access deny manager
http_access deny !Safe_ports
http_access deny CONNECT !SSL_ports
http_access allow localnet
http_access allow localhost
http_access deny all
(content truncated)
```

5. Définir le fichier *.keytab* sur lecture (read) par le propriétaire du processus **Squid** :

```
~]# chgrp squid /etc/squid/HTTP.keytab
```

```
~]# chmod g+r /etc/squid/HTTP.keytab
```

6. Ajouter les lignes suivantes au fichier **/etc/sysconfig/squid** :

```
KRB5_KTNAME="/etc/squid/HTTP.keytab "
export KRB5_KTNAME
```

7. Démarrer le service **Squid**

```
~]# service squid start
```

8. Configuration un client Kerberos et configurer votre navigateur pour qu'il utilise le serveur proxy de **Squid**. Obtenir un ticket Kerberos du KDC (Key Distribution Center).

```
~]# kinit testuser1
```

   Essayez d'accéder à un site. Le navigateur ne doit pas vous demander un nom d'utilisateur ou un mot de passe.

12.5. UTILISER SQUID POUR UN ACCÈS LIMITÉ

Principalement, **Squid** est utilisé pour bloquer l'accès à certains contenus web. Normalement, certains ports sont bloqués ou ce sont certains sites.

12.5.1. Limiter un accès en bloquant un port

Par cette méthode, aussi appelée filtrage de port, vous pouvez bloquer un port spécifique avec le serveur proxy de **Squid**. Ce faisant, vous pouvez restreindre l'utilisation de certains protocoles, services, sites Web, ou applications. Par exemple, pour bloquer le trafic FTP, il suffit de bloquer le port 21/TCP. De la même manière, vous pouvez bloquer tous les sites HTTPS en bloquant le port 443/TCP.

Procédure 12.5. Bloquer des numéros de port

1. Connectez vous en tant que superutilisateur, et ouvrir le fichier de configuration de **Squid** :

```
~]# vi /etc/squid/squid.conf
```

2. Bloquer les ports par des ACL.

```
acl Bad_ports port 443          #(create acl for port 443/tcp)
```

3. Enregistrez les changements.

4. Redémarrer **Squid** pour faire appliquer la nouvelle configuration :

```
~]# service squid reload
```

Le fichier de configuration de **Squid** contient des lignes avec *acl Safe_ports port*. Par défaut, ces numéros de port sont ajoutés en tant que "Safe_Ports" sont ouverts à la navigation.

```
acl Safe_ports port 80
acl Safe_ports port 21
acl Safe_ports port 443
acl Safe_ports port 70
acl Safe_ports port 210
acl Safe_ports port 1025-65535
acl Safe_ports port 280
acl Safe_ports port 488
acl Safe_ports port 591
acl Safe_ports port 777
```

Vous pouvez désactiver toutes les lignes listées dans **/etc/squid/squid.conf** pour bloquer les ports qu'il faut.

Exemple 12.7. Bloquer le port 777/tcp

Pour bloquer le port 777/tcp, ajouter un signe de hachage devant la ligne dont il s'agit, comme suit :

```
#acl Safe_ports port 777         # multiling http
```

12.5.2. Limiter un accès en bloquant des adresses ou des sites spécifiques

Configurer **Squid** pour que votre réseau désactive l'accès à certains sites.

Procédure 12.6. Bloquer un site particulier

1. Autoriser l'accès à **Squid** sur votre réseau. Ouvrir le fichier **/etc/squid/squid.conf** et chercher "Access Controls". Défilez vers le bas *INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR CLIENTS*. Il vous faudra adapter la liste en fonction de vos réseaux d'IP internes à partir desquels on pourra naviguer. Dans cet exemple, l'ACL donne accès aux réseaux locaux 192.168.1.0/24 et 192.168.2.0/24.

   ```
   # INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR CLIENTS
   acl our_networks src 192.168.1.0/24 192.168.2.0/24
   http_access allow our_networks
   ```

2. Créer un fichier qui contient une liste de sites que vous souhaitez bloquer. Nommez les fichiers, par exemple, **/usr/local/etc/allowed-sites.squid** et **/usr/local/etc/restricted-sites.squid**.

   ```
   ~]# cat /usr/local/etc/allowed-sites.squid
   www.redhat.com
   fedoraproject.org
   ```

   ```
   ~]# cat /usr/local/etc/restricted-sites.squid
   www.badsites.com
   illegal.com
   ```

Ils peuvent ensuite être utilisés pour bloquer les sites ayant des restrictions.

```
~]# vi /etc/squid/squid.conf
```

```
acl our_networks src 192.168.1.0/24 192.168.2.0/24
acl GoodSites dstdomain "/usr/local/etc/allowed-sites.squid"
acl BadSites dstdomain "/usr/local/etc/restricted-sites.squid"

http_access allow our_networks
http_access deny BadSites
http_access allow home_network business_hours GoodSites
```

Enregistrer et fermer le fichier.

3. Re-démarrer le serveur proxy de **Squid** :

```
~]# systemctl restart squid
```

4. Configurer votre navigateur web pour qu'il utilise le nom DNS ou l'adresse IP de votre serveur **Squid** et qu'il corresponde au port en cours.

12.6. RESSOURCES SUPPLÉMENTAIRES : DOCUMENTATION INSTALLÉE

Les pages man suivantes fournissent des ressources supplémentaires à propos de **Squid**.

- **squid(8)**
- **squidclient(1)**
- **basic_ldap_auth(8)**
- **ext_ldap_group_acl(8)**
- **ext_session_acl(8)**
- **ext_unix_group_acl(8)**
- **negotiate_kerberos_auth(8)**

ANNEXE A. HISTORIQUE DE RÉVISION

Version 0.9-30 **Tue 18 Oct 2016** **Mirek Jahoda**
Version pour la distribution GA 7.3.

Version 0.9-27 **Fri 24 Jun 2016** **Mirek Jahoda**
Ajout de *Squid*.

Version 0.9-25 **Wed 11 Nov 2015** **Jana Heves**
Distribution 7.2 GA.

Version 0.9-15 **Tue 17 Feb 2015** **Christian Huffman**
Version pour la distribution 7.1 GA

Version 0.9-14 **Fri Dec 05 2014** **Christian Huffman**
Mise à jour des sections nmtui et NetworkManager GUI.

Version 0.9-12 **Wed Nov 05 2014** **Stephen Wadeley**
Améliorations dans *Gestion des réseaux IP* , *802.1Q VLAN tagging*, et *Associations de réseaux*.

Version 0.9-11 **Tues Oct 21 2014** **Stephen Wadeley**
Améliorations dans *Bonding*, *Pontage de réseaux*, et *Teaming*.

Version 0.9-9 **Tue Sep 2 2014** **Stephen Wadeley**
Améliorations dans *Bonding* et *Dénomination de périphériques réseaux consistante*.

Version 0.9-8 **Tue July 8 2014** **Stephen Wadeley**
Distribution du Guide de Gestion des réseaux de Red Hat Enterprise Linux 7.0 GA .

Version 0-0 **Wed Dec 12 2012** **Stephen Wadeley**
Initialisation du guide Red Hat Enterprise Linux 7 Networking Guide.

A.1. REMERCIEMENTS

Certaines portions de ce texte ont déjà été publiées dans le *Guide de Déploiement de Red Hat Enterprise Linux 6*, copyright © 2014 Red Hat, Inc., disponible à l'adresse suivante https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/index.html.

INDEX

Symboles

A

B

C

D

E

I

L

M

N

P

R

S