Pédants Principes PPP

Frédéric Haby

Frédéric.haby@freebsd-fr.org

Pédants Principes PPP

par Frédéric Haby

Publié $FreeBSD: head/fr_FR.ISO8859-1/books/ppp-primer/book.sgml 38826 2012-05-17 19:12:14Z hrs $

Ce document est un guide pas à pas pour la configuration de FreeBSD dans le cadre de son utilisation comme routeur/passerelle de connexions à la demande dans un réseau local. Toutes les commandes fonctionnent avec un système FreeBSD 2.2 ou ultérieur, sauf indication contraire.

Version française de Frédéric Haby <`frederic.haby@mail.dotcom.fr`>.

Table des matières

Introduction

Le protocole de connexion PPP en mode utilisateur de la version 2.2 de FreeBSD (appelé aussi “IIJ-PPP”) supporte maintenant la réécriture de paquets pour les connexions à l’Internet. Cette possibilité, connue aussi sous le nom de “*Mascarade (Masquerading)*”, “*Alias IP (IP Aliasing)*”, ou “*Traduction d’Adresse Réseau (Network Adresse Translation)*”, permet à un système FreeBSD d’interagir comme routeur de connexion à la demande entre un réseau local et un fournisseur d’accès à Internet. Les machines du réseau local peuvent utiliser l’unique connexion de la machine FreeBSD à l’Internet pour échanger des informations avec ce dernier.

But de ce document

Ce document[1] vous explique comment:

- Configurer le système FreeBSD pour permettre les connexions vers l’extérieur,
- Partager une seule connexion vers l’extérieur avec les autres machines du réseau,
- Configurer les machines Windows pour utiliser la machine FreeBSD comme passerelle vers l’Internet.

Bien que l’objectif de ce document soit avant tout d’expliquer comment configurer la traduction d’adresses IP, il fournit aussi des exemples d’installation et de configuration des autres composants. Chaque section est indépendante et peut être utilisée pour configurer divers aspects d’une passerelle FreeBSD entre plusieurs réseaux.

Notes

1. N.d.T: Si vous souhaitez consulter le document original en langue anglaise, il se trouve sur http://www.FreeBSD.org/ppp/ppp.html (http://www.FreeBSD.org/tutorials/ppp/ppp.html).

Chapitre 1. Mise en oeuvre du réseau local

Bien que le programme `ppp` puisse, et soit généralement utilisé pour fournir des services à la *seule* machine FreeBSD, il peut aussi servir à en faire une “passerelle” ou “routeur” entre les autres dispositifs reliés au réseau local et l’Internet ou tout autre service connecté.

1.1. Topologie classique du réseau

Ce document s’applique à un réseau local dont la topologie est - classiquement - la suivante:

```
+=========+      =======> Connexion à Internet ( Wanadoo, AOL, etc. )
|         |      \
| FreeBSD |=======
| "Curly" |
|         |
+====+====+
     |
|================+===========+====|  <== Réseau Local Ethernet
     |           |            |
+====+====+  +====+====+  +====+====+
|         |  |         |  |         |
|  Win95  |  |   WFW   |  |  WinNT  |
| "Larry" |  |  "Moe"  |  | "Shemp" |
|         |  |         |  |         |
+=========+  +=========+  +=========+
```

1.2. Hypothèses sur le réseau local

Pour les besoins de notre exemple, nous supposerons que :

Il y a trois stations et un serveur reliés par le réseau Ethernet local:

- un serveur FreeBSD (“Curly”) avec un contrôleur Ethernet NE-2000 configuré sur “ed0”,
- une station de travail Windows 95 (“Larry”) avec les pilotes TCP/IP 32-bits natifs de Microsoft,
- une station de travail Windows for Workgroups (“Larry”) avec les extension TCP/IP 16-bits de Microsoft,
- une station de travail Windows NT (“Larry”) avec les pilotes TCP/IP 32-bits natifs de Microsoft.

Les adresses IP du réseau Ethernet sur lequel se base notre exemple ont été attribuées dans l’espace d’adresses privées défini par la RFC-1597, comme suit:

Nom	Adresse IP	Commentaire
`Curly`	`192.168.1.1`	La machine FreeBSD
`Larry`	`192.168.1.2`	La machine Windows 95

Nom	Adresse IP	Commentaire
`Moe`	`192.168.1.3`	La machine Windows for Workgroups
`Shemp`	`192.168.1.4`	La machine Windows NT

Nous supposerons aussi que le modem est connecté sur le premier port série (`/dev/cuaa0` ou *COM1:* selon la terminologie DOS) de la machine FreeBSD.

Enfin, nous supposerons que notre fournisseur d'accès Internet nous fournit automatiquement les adresses IP des deux extrémités de la liaison PPP (point à point) : celle de notre machine FreeBSD et celle de la machine du côté du fournisseur (adressage dynamique). Les détails de configuration de la connexion de la machine FreeBSD à l'extérieur sont donnés dans le chapître 2, "Configuration du système FreeBSD".

Chapitre 2. Configuration du système FreeBSD

Il y a trois informations à fournir au système FreeBSD avant de le relier au réseau local de notre exemple:

- Son nom de machine (`hostname`), “Curly” dans notre exemple,
- La configuration de son interface réseau,
- Le fichier `/etc/hosts` (qui liste les noms et les adresses IP des autres machines du réseau).

Si vous avez installé FreeBSD via une connexion réseau, il est possible que certaines de ces informations soient déjà définies.

Même si vous pensez que votre système FreeBSD a déjà été correctement configuré au moment de l’installation, il vous mieux vous assurer de la validité de chacune de ces informations pour vous éviter des ennuis par la suite.

2.1. Contrôler le nom de la machine FreeBSD

Il est possible que le nom de la machine FreeBSD ait été défini et enregistré lors de l’installation du système. Pour le vérifier, entrez, à l’invite du système, la commande suivante:

```
# hostname
```

Le nom de la machine FreeBSD sera affiché sur une seule ligne. Si ce nom paraît correct (c’est très subjectif :-), vous pouvez passer au paragraphe Contrôler la configuration de l’interface Ethernet.

Dans notre exemple, le résultat de la commande `hostname` devrait être “curly.my.domain” si le nom de la machine a été correctement défini à l’installation, ou ensuite. (A ce stade, ne vous préoccupez pas du suffixe “.my.domain”, nous verrons cela par la suite. Ce qui compte maintenant est le nom qui précède le premier “.”)

Si le nom de machine n’a pas été défini à l’installation, la réponse de `hostname` sera probablement “myname.my.domain”. Vous devez alors éditer le fichier `/etc/rc.conf` pour définir le nom du système.

2.1.1. Définir le nom de la machine FreeBSD

> **Note :** Vous devez être en session sous “root” pour éditer les fichiers de configuration du système.

> **Note :** Si vous introduisez des erreurs dans les fichiers de configuration, il y a de fortes chances que votre système *ne redémarre* pas correctement! Soyez prudents !

Le nom de machine du système FreeBSD est défini au démarrage par le fichier `/etc/rc.conf`. Utilisez l’éditeur par défaut (“ee”) pour éditer ce fichier.

Sous une session “root”, visualisez le fichier dans l’éditeur avec la commande:

```
# ee /etc/rc.conf
```

Avec les touches flèches, allez jusqu'à la ligne qui définit le nom de machine du système FreeBSD. Par défaut, vous y trouvez:

```
---
### Basic network options: ###
hostname="myname.my.domain"     # Set this!
---
```

Remplacez ces lignes, par exemple, par:

```
---
### Basic network options: ###
hostname="curly.my.domain"      # Set this!
---
```

Une fois la modification faite, appuyez sur la touche **Echap** pour aller dans le menu de commandes de l'éditeur. Choisissez “leave editor” (quitter l'éditeur) et sélectionnez bien l'option “save changes” (enregistrer les modifications) quand la question vous est posée.

2.2. Contrôler la configuration de l'interface Ethernet

Rappelons nos hypothèses : l'interface Ethernet de la machine s'appelle “ed0”. C'est la valeur par défaut pour les contrôleurs NE-1000, NE-2000, WD/SMC modèles 8003, 8013 et Elite Ultra (8216).

Les fichiers d'interface d'autres modèles de contrôleurs peuvent avoir d'autres noms sous FreeBSD. Si vous n'êtes pas certain du nom correspondant à votre contrôleur, consultez les Questions Fréquemment Posées à propos de FreeBSD (“FAQ”) et substituez la bonne valeur (par ex.: '*de0*','*zp0*', ou équivalent) dans les exemples suivants.

De la même façon que pour le nom de la machine, l'interface Ethernet du système FreeBSD a peut-être été configurée à l'installation.

Pour consulter la configuration des interfaces (Ethernet et autres) de votre système FreeBSD), utilisez la commande suivante:

```
# ifconfig -a
```

Soit en clair: "Donnes-moi la CONFIGuration des *I*nter*F*aces de mes périphériques réseau."

Par exemple:

```
# ifconfig -a
ed0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
inet 192.168.1.1 netmask 0xffffff00 broadcast 192.168.1.255
ether 01:02:03:04:05:06
lp0: flags=8810<POINTOPOINT,SIMPLEX,MULTICAST> mtu 1500
tun0: flags=8050<POINTOPOINT,RUNNING, MULTICAST> mtu 1500
l0: flags=c010<OINTOPOINT,LINK2,MULTICAST> mtu 552
ppp0: flags=8010<POINTOPOINT,MULTICAST> mtu 1500
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
```

```
inet 127.0.0.1 netmask 0xff000000
#
```

Sur cet exemple, les périphériques suivants sont reconnus :

- `ed0`: l'interface Ethernet,
- `lp0`: l'interface parallèle (que nous n'utiliserons pas ici),
- `tun0`: l'interface "tunnel"; *Il s'agit de celle utilisée par le PPP utilisateur*
- `ppp0`: un autre interface PPP (pour le module `ppp` intégré au noyau, que n'utiliserons pas ici),
- `lo0`: l'interface "en boucle" - "loopback" - que nous n'utiliserons pas non plus).

Dans l'exemple, l'interface "ed0" est active ("UP") et fonctionne ("RUNNING"). Les indications importantes sont:

1. L'interface est active (son status est "UP"),
2. Elle a une adresse Ethernet ("inet"), dans notre cas: 192.168.1.1,
3. Elle a un masque de sous-réseau ("netmask"; 0xffffff00 qui équivaut à 255.255.255.0) correct,
4. Elle a une adresse de diffusion ("broadcast"; ici, 192.168.1.255) valide.

Si la ligne décrivant l'interface Ethernet mentionne quelque chose comme:

```
ed0: flags=8802<BROADCAST,SIMPLEX,MULTICAST> mtu 1500
ether 01:02:03:04:05:06
```

c'est que cette interface n'est pas encore configurée.

Si l'interface Ethernet est déjà configurée, vous pouvez passer directement au paragraphe
Renseigner la liste des autres machines du réseau. Sinon, continuez en séquence.

2.2.1. Configurer votre interface Ethernet

> **Note :** Vous devez être en session sous "root" pour éditer les fichiers de configuration du système.

> **Note :** Si vous introduisez des erreurs dans les fichiers de configuration, il y a de fortes chances que votre système *ne redémarre* pas correctement! Soyez prudents !

Les paramètres de configuration des interfaces réseau sont définis au démarrage par le fichier `/etc/rc.conf`. Utilisez l'éditeur par défaut ("ee") pour éditer ce fichier.

Sous une session "root", visualisez le fichier dans l'éditeur avec la commande:

`# ee /etc/rc.conf`

Vers la vingtième ligne environ du fichier, se trouve la section qui définit quelles interfaces réseau doivent être mises en service au démarrage. Le fichier de configuration par défaut contient la ligne :

```
network_interfaces="lo0"        # List of network interfaces (lo0 is loopback)
```

que vous devrez modifier pour dire à FreeBSD que vous voulez ajouter un autre périphérique, en l'occurence l'interface "ed0", en:

```
network_interfaces="lo0 ed0"    # List of network interfaces (lo0 is loopback)
```

(Faites attention à l'espace entre la définition de l'interface en boucle "lo0" et l'interface Ethernet "ed0".)

> **Note :** Si le nom correspondant à votre interface n'est pas "ed0",n'oubliez pas de mettre la valeur adéquate à la place.

Si vous avez installé FreeBSD via une connexion réseau, la ligne "`network_interfaces`" comporte peut-être déjà un référence à votre carte Ethernet. Dans ce cas, assurez vous que son nom soit correct.

Les lignes qui suivent celles où sont listées les interfaces à mettre en service décrivent les paramètres de chacune de ces interfaces. Dans le fichier `/etc/rc.conf` par défaut, il n'y a qu'un seule ligne qui dit:

```
ifconfig_lo0="inet 127.0.0.1"   # default loopback device configuration
```

Vous devrez ajouter une ligne qui définisse les caractéristiques de votre interface "ed0".

Si vous avez installé FreeBSD via une connexion réseau, la ligne "`ifconfig_ed0=`" existe peut-être déjà. Vérifiez alors qu'elle contienne bien les bonnes valeurs.

En nous référant à notre exemple, nous insérerons la ligne suivante, immédiatement après la définition de l'interface "en boucle":

```
ifconfig_ed0="inet 192.168.1.1 netmask 255.255.255.0"
```

Après modification, cette section du fichier `/etc/rc.conf` doit ressembler à:

```
---
network_interfaces="lo0 ed0"    # List of network interfaces (lo0 is loopback)
ifconfig_lo0="inet 127.0.0.1"   # default loopback device configuration
ifconfig_ed0="inet 192.168.1.1 netmask 255.255.255.0"
---
```

Un fois que vous avez fait toutes les modifications nécessaires, appuyez sur la touche **Echap** pour aller dans le menu de commandes de l'éditeur. Choisissez "leave editor" (quitter l'éditeur) et sélectionnez bien l'option "save changes" (enregistrer les modifications) quand la question vous est posée.

2.3. Autoriser la transmission de paquets

Par défaut, FreeBSD n'active pas le transmission de paquets entre les différentes interfaces réseau d'une machine. En d'autres termes, les fonctions de routage (aussi appelées de passerelle) ne sont pas disponibles.

Si vous voulez utiliser le programme `ppp` comme poste Internet autonome et non comme passerelle entre le réseau local et votre fournisseur d'accès à Internet, vous pouvez passer directement au paragraphe Renseigner la liste des autres machines du réseau.

Si vous voulez utiliser le programme `ppp` pour connecter à la fois votre machine FreeBSD (pour servir de routeur) et les autres postes du réseau local, vous devez autoriser la transmission de paquets IP.

Pour cela, vous devez éditer le fichier `/etc/rc.conf`. Visualisez le fichier dans l'éditeur avec la commande:

`# ee /etc/rc.conf`

Vers la quatre-vingt-cinquième ligne environ, vous trouverez la section qui contrôle la transmission de paquets. Elle ressemblera a ceci:

```
---
gateway_enable="NO"  # Set to YES if the host will be a gateway
---
```

Modifiez-la en:

```
---
gateway_enable="YES"  # Set to YES if the host will be a gateway
---
```

et quittez l'éditeur (en enregistrant vos modifications!).

> **Note :** Cette ligne contient peut-être déjà `gateway_enable="YES"` si la transmission de paquets IP a été autorisée au moment de l'installation du système FreeBSD.

2.4. Renseigner la liste des autres machines du réseau (`/etc/hosts`)

La dernière étape de la configuration du réseau local vu de la machine FreeBSD consiste à créer la liste des noms et des adresses TCP/IP des différentes machines raccordées au réseau local. Cette liste est définie dans le fichier `/etc/hosts`.

Par défaut, ce fichier ne contient qu'une ligne: le nom et l'adresse de l'interface "en boucle". Par convention, ce nom est toujours "localhost" et son adresse 127.0.0.1. (Reportez-vous à l'exemple de configuration des interfaces au paragraphe Contrôler la configuration de l'interface Ethernet.)

Pour éditer le fichier `/etc/hosts`, tapez la commande:

`# ee /etc/hosts`

Allez jusqu'à la fin du fichier (en lisant au passage les commentaires, ce sont des informations utiles!) et ajoutez (toujours dans le cas du réseau de notre exemple) les noms et les adresses IP suivantes:

```
192.168.1.1   curly   curly.my.domain # machine FreeBSD
192.168.1.2   larry   larry.my.domain # machine Windows 95
192.168.1.3   moe     moe.my.domain   # machine Windows for Workgroups
192.168.1.4   shemp   shemp.my.domain # machine Windows NT
```

(Vous ne devez pas modifier la ligne “`127.0.0.1 localhost`”.)

Une fois que vous avez ajouté ces lignes, appuyez sur la touche **Echap** pour aller dans le menu de commandes de l’éditeur. Choisissez “leave editor” (quitter l’éditeur) et sélectionnez bien l’option “save changes” (enregistrer les modifications) quand la question vous est posée.

2.5. Tester le système FreeBSD

Félicitations! Une fois que vous en êtes arrivé là, votre machine FreeBSD est maintenant une machine UNIX connectée au réseau! Si vous avez dû modifier votre fichier `/etc/rc.conf`, il faut redémarrer votre machine FreeBSD. Cela va faire deux choses importantes:

- Appliquer les modifications de configuration des interfaces, et:
- Vérifier que le système redémarre sans erreur de configuration flagrante.

Une fois que le système a redémarré, vous devriez tester vos interfaces réseaux.

2.5.1. Vérifier le bon fonctionnement de l’interface “en boucle”

Pour vérifier que l’interface “en boucle” est correctement configurée, ouvrez une session sous “root” et tapez:

```
# ping localhost
```

Vous devriez voir défiler des messages du type:

```
# ping localhost

PING localhost.my.domain. (127.0.0.1): 56 bytes
64 bytes from 127.0.0.1: icmp_seq=0 ttl=255 time=0.219 ms
64 bytes from 127.0.0.1: icmp_seq=1 ttl=255 time=0.200 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=255 time=0.187 ms
[...]
```

jusqu’à ce que vous tapiez **Ctrl-C** pour arrêter ce délire.

2.5.2. Vérifier le bon fonctionnement de l’interface Ethernet

Pour vérifier que l’interface Ethernet est correctement configurée, tapez:

```
# ping curly
```

Vous devriez voir:

```
# ping curly

PING curly.my.domain. (192.168.1.1): 56 bytes
64 bytes from 192.168.1.1: icmp_seq=0 ttl=255 time=0.219 ms
64 bytes from 192.168.1.1: icmp_seq=1 ttl=255 time=0.200 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=255 time=0.187 ms
```

```
[...]
```

L’information importante dans ces deux exemples est que les noms (“localhost” et “curly”) correspondent bien à leurs adresses IP (127.0.0.1 et 192.168.1.1). Cela confirme que le fichier `/etc/hosts` est correct.

Si l’adresse IP pour "curly" n’est pas 192.168.1.1 ou que l’adresse pour "localhost" n’est pas 127.0.0.1, retournez au paragraphe Pouet Pouet.

Si les noms et les adresses apparaîssent correctement dans les résultats de la commande `ping` mais que des messages d’erreur sont affichés, quelque chose ne va pas dans la configuration des interfaces. Retournez au paragraphe Contrôler le nom de la machine FreeBSD et vérifiez de nouveau votre configuration.

Si maintenant, tout va bien, passez à la section suivante.

Chapitre 3. Configurer la liaison PPP vers l’extérieur

Il y a deux façons d’utiliser le programme `ppp`: en mode “Interactif” et en mode “Automatique”.

En mode Interactif, vous:

- Etablissez manuellement la connexion avec votre fournisseur d’accès Internet,
- Surfez, transférez des fichiers et envoyez du courrier électronique, etc.
- Coupez manuellement la connexion avec votre fournisseur d’accès Internet.

En mode Automatique, le programme `ppp` surveille silencieusement les opérations du système FreeBSD et se connecte et se déconnecte automagiquement à votre fournisseur d’accès selon le besoin, intégrant de façon transparente l’Internet à votre réseau.

Dans cette section, les configurations pour les deux modes seront étudiées avec un approfondissement sur la configuration du mode Automatique.

3.1. Sauvegarder vos fichiers de configuration `ppp` d’origine

> **Note :** Les versions les plus récentes de FreeBSD possèdent des fichiers d’exemple dans `/usr/share/examples/ppp`, aussi, cette étape peut ne pas être nécessaire.

Avant de modifier les fichiers qu’utilise `ppp`, il est bon de sauvegarder les fichiers par défaut créés à l’installation de FreeBSD.

Ouvrez une session sous le compte “root”, et suivez la procédure ci-dessous:

Allez dans le répertoire `/etc`:

```
# cd etc
```

Faites un copie des fichiers d’origine du répertoire `ppp`:

```
# cp -R ppp ppp.ORIGINAL
```

Vous devriez maintenant avoir un répertoire `ppp` et un répertoire `ppp.ORIGINAL` dans le répertoire `etc`.

3.2. Créer vos propres fichiers de configuration de `ppp`

Par défaut, la procédure d’installation de FreeBSD crée un certain nombre de fichiers d’exemple de configuration dans le répertoire `/etc/ppp`. Prenez un peu de temps pour parcourir ces fichiers. Ce sont des exemples empruntés à des systèmes qui fonctionnent et ils illustrent les différentes possibilités du programme `ppp`.

Je vous encourage *fortement* à étudier ces exemples et à les adapter à votre configuration.

Pour plus d’informations sur le programme `ppp`, lisez les pages de manuel:

```
# man ppp
```

Pour plus d'informations sur le langage `chat`, utilisé par `ppp` pour établir la connexion, lisez les pages de manuel:

```
# man chat
```

La suite de cette section décrit le contenu recommandé des fichiers de configurations de `ppp`.

3.2.1. Le fichier `/etc/ppp/ppp.conf`

Le fichier `/etc/ppp/ppp.conf` contient les informations et les paramètres nécessaires pour établir une connexion vers l'extérieur. Le Manuel FreeBSD (../handbook/index.html) décrit en détail le contenu et la syntaxe de ce fichier.

N'est décrite ici que la configuration minimale nécessaire pour arriver à établir une connexion.

Voici le fichier `/etc/ppp/ppp.conf` que nous utiliserons sur notre passerelle entre le réseau local et l'Internet:

```
##################################################################
# Fichier de Configuration de PPP ('/etc/ppp/ppp.conf')
#
# Paramètres par défaut: Ces commandes sont toujours exécutées
# quand PPP est appelé et s'appliquent à toutes les configurations
##################################################################
default:
set device /dev/cuaa0
set speed 57600
disable pred1
deny pred1
disable lqr
deny lqr
set dial "ABORT BUSY ABORT NO\\sCARRIER TIMEOUT 5 \"\" ATE1Q0M0
OK-AT-OK\\dATDT\\T TIMEOUT 40 CONNECT"
set redial 3 10
#
#
##################################################################
#
# Configuration utilisée en mode interactif :
#
# appelée par: `ppp -alias interactive`
#
##################################################################
interactive:
set authname Your_User_ID_On_Remote_System
set authkey Your_Password_On_Remote_System
set phone 1-800-123-4567
set timeout 300
set openmode active
accept chap
#
##################################################################
#
# Configuration utilisée en mode à la demande ( automatique)
#
```

```
# appelée par: 'ppp -auto -alias demand'
#
####################################################################
demand:
set authname Your_User_ID_On_Remote_System
set authkey Your_Password_On_Remote_System
set phone 1-800-123-4567
set timeout 300
set openmode active
accept chap
set ifaddr 127.1.1.1/0 127.2.2.2/0 255.255.255.0
add 0 0 127.2.2.2
################################################################
# Fin de /etc/ppp/ppp.conf
```

Ce fichier, recopié tel quel d'un système opérationnel, comporte trois parties:

3.2.1.1. La section "default"

La section "default:" contient les informations et paramètres utilisés par toutes les autres sections du fichier. Cette section est donc implicitement ajoutée à toutes les autres.

C'est l'endroit idéal pour définir les valeurs par défaut qui s'appliquent à toutes les connexions vers l'extérieur; en particulier, la configuration du modem et le dialogue d'initialisation de la connexion, qui ne changent pas, quel que soit le système distant auquel vous vous connectez.

Voici le détail de chacune des lignes de la section "default:" du fichier `/etc/ppp/ppp.conf`:

```
set device /dev/cuaa0
```

Cette instruction dit au programme `ppp` d'utiliser le premier port série. Sous FreeBSD, le périphérique "/dev/cuaa0" correspond au port appelé "COM1:" de DOS, Windows, Windows 95, etc.

Si votre modem est sur le port "COM2": vous devrez mettre "/dev/cuaa1". "/dev/cuaa2" serait le port "COM3:".

```
set speed 57600
```

Cette ligne indique au programme `ppp` la vitesse en émission et en réception entre le port série et le modem. Bien que le modem de notre configuration fonctionne à 28.8 Kbps, donner la valeur 57600 permet à la liaison série de fonctionner suffisamment vite pour gérer la compression de données intégrée aux modems les plus récents.

Si vous avez des problèmes pour dialoguer avec votre modem à cette vitesse, descendez à une valeur inférieure: 38400, voir aussi faible que 19200.

```
disable pred1
deny pred1
```

Ces deux lignes désactivent la compression "CCP/Predictor 1" dont est capable le programme `ppp`. La version actuelle de `ppp` supporte la compression de données selon la définition provisoire des standards Internet. Malheureusement, de nombreux fournisseurs d'accès ont des équipements qui n'offrent pas cette possibilité. Comme la plupart des modems récents sont capables de compression à la volée, vous ne perdrez probablement pas grand-chose en termes de performances en désactivant cette possibilité sur votre machine FreeBSD et en interdisant au site distant de vous la réclamer.

```
disable lqr
deny lqr
```

Ces deux lignes contrôlent l'"audit de la qualité de la liaison (Link Quality Reporting)" qui est partie intégrante des spécifications du protocole PPP. (Voyez la RFC-1989 pour plus de détails.)

La première ligne, "disable lqr" instruit le programme `ppp` de ne pas émettre de demandes de contrôle de la qualité de la liaison à la machine distante.

La seconde ligne, "deny lqr" instruit le progamme `ppp` de ne pas accepter de demandes de contrôle de la qualité de la liaison de la machine distante.

Comme la plupart des modems ont des procédures intégrées de détection et de correction d'erreur et que l'audit LQR est rarement intégralement implémentée par les constructeurs, il vaut en général mieux ne pas utiliser cette possibilité.

```
set dial "ABORT BUSY ABORT NO\\sCARRIER TIMEOUT 5 \"\
" ATE1Q0M0 OK-AT-OK\\dATADT\\T TIMEOUT 40 CONNECT"
```

> **Note :** (Cette instruction doit être écrite sur une seule ligne; ne tenez pas compte du formatage sur deux lignes, si elle se présente ainsi dans ce document.)

Cette ligne décrit le dialogue entre le programme `ppp` et le modem qui doit permettre d'établir la connexion et fournit des instructions rudimentaires sur la marche à suivre:

- Une tentative d'appel échoue si le modem renvoie le code d'erreur "BUSY (occupé)",
- Une tentative d'appel échoue aussi si le modem renvoie le code d'erreur "NO CARRIER (pas de porteuse),",
- Le programme `ppp` doit constater, avec un délai maximum de cinq secondes entre chaque événement, la séquence d'événements suivants:
  - Initialement, le programme `ppp` n'attend pas de réponse du modem (c'est le sens de la chaîne `\"\"` dans l'instruction),
  - Le programme enverra au modem la chaîne "ATE1Q0M0" et attendra la réponse "OK". Si cette réponse n'arrive pas, il enverra la commande d'interrogation "AT" et attendra à nouveau une réponse "OK",
  - Le programme devra attendre une seconde (c'est le sens de la chaîne "\\d" dans l'instruction), et envoyer la séquence d'appel au modem. Le préfixe "ATDT" est la commande d'appel standard pour les lignes téléphoniques à fréquences vocales. Si vous n'avez pas de ligne à fréquence vocale, remplacez "ATDT" par

"ATDP". La chaîne "\\T" sera remplacée par le numéro de téléphone réel (qui sera défini plus loin par la ligne "set dial").

- Pour finir, avec un délai d'attente de quarante secondes au maximum, le programme `ppp` doit recevoir un code d'acquittement "CONNECT" du modem.

Un échec à une phase quelconque de ce dialogue sera interpreté comme un échec d'appel et la connexion ne sera pas établie.

(Pour avoir une description détaillée du mini-langage de commande utilisé par `ppp`, reportez-vous aux pages de manuel de `chat`.)

```
set redial 3 10
```

Cette ligne indique au programme `ppp` qu'en cas d'échec, il peut faire jusqu'à trois nouvelles tentatives avec un temps d'attente de dix secondes entre chaque nouvel appel.

3.2.1.2. La section "interactive"

La section "interactive:" contient les valeurs et les paramètres utilisés par les connexions PPP interactives avec un système distant particulier. Les lignes de la section "default:" sont automatiquement incluses dans cette section.

L'exemple donné ci-dessous suppose que vous vous connectiez à un système qui sait identifier un utilisateur sans avoir à utiliser une série de commandes baroques. Nous supposerons que le protocole CHAP est utilisé pour initialiser la session.

A priori, si un système Windows 95 peut établir la connexion d'un simple clic sur le bouton "Connecter", la configuration donnée en exemple devrait fonctionner.

Si, à l'inverse, quand vous vous connectez chez votre fournisseur d'accès avec les utilitaires de Microsoft Windows 95, vous devez utiliser l'option "Utiliser des procédures de connexion"[1]. il vous faudra alors lire les pages de manuel de `ppp` pour trouver des exemples de fichiers de configuration de PPP comportant des procédures "expect / reponse" pour établir votre connexion et vous devrez utiliser la commande "set login" à cet effet.

Ou mieux, trouvez-vous un fournisseur d'accès qui offre l'authentification par PAP ou CHAP!

La configuration donnée en exemple a été utilisée avec succès pour se connecter:

- à Divers Shiva LanRovers,
- au réseau IBM (http://www.ibm.net),
- à AT&T WorlNet (http://www.att.com/worldnet),
- à Erol's (http://www.erols.com).

Voici la description de chacune des lignes de la section "interactive:" du fichier `/etc/ppp/ppp.conf` de l'exemple:

```
set authname Votre_Nom_d_Utilisateur_du_Système_Distant
```

Cette ligne définit avec quel nom d'utilisateur vous vous connectez au système distant.

```
set authkey Votre_Mot_de_Passe_Utilisateur_du_Système_Distant
```

Cette ligne définit le mot de passe que vous utilisez pour vous connecter au système distant.

```
set phone 012345678789
```

C'est le numéro de téléphone avec lequel vous joignez le système distant.

```
set timeout 300
```

Cette ligne indique au programme `ppp` de couper la ligne s'il n'y a aucun échange pendant cinq minutes (300 secondes). Vous pouvez adapter cette ligne à votre besoin particulier.

```
set openmode active
```

Cette ligne dit au programme `ppp` de négocier la connexion dès que la liaison entre les modems est établie. Certains sites distants le font automatiquement, d'autres non. Cette instruction permet à votre site d'initialiser la négociation.

```
accept chap
```

Cette ligne dit au programme `ppp` d'utiliser le protocole CHAP "Challenge Handsake Autentification Protocole (protocole d'authentification par poignée de main avec défi de décryptage)" pour vous authentifier. Les valeurs échangées entre le système local et le système distant pour transmettre le nom d'utilisateur et le mot de passe sont définies par les lignes "authname" et "authkey".

3.2.1.3. La section "demand"

La section "demand:" contient les valeurs et les paramètres utilisés par les connexions PPP automatiques avec un système distant particulier. Les lignes de la section "default:" sont automatiquement incluses dans cette section.

A l'exception des deux dernières lignes, cette section est identique à la section qui définit la configuration "interactive".

Comme indiqué plus tôt, les exemples donnés dans cette section suppose que vous vous connectiez à un système qui comprenne le protocole CHAP pour initialiser la session.

Voici la description de chacune des lignes de la section "demand:" du fichier `/etc/ppp/ppp.conf` de l'exemple:

```
set authname Votre_Nom_d_Utilisateur_du_Système_Distant
```

Cette ligne définit avec quel nom d'utilisateur vous vous connectez au système distant.

```
set authkey Votre_Mot_de_Passe_Utilisateur_du_Système_Distant
```

Cette ligne définit le mot de passe que vous utilisez pour vous connecter au système distant.

```
set phone 012345678789
```

C'est le numéro de téléphone avec lequel vous joignez le système distant.

```
set timeout 300
```

Cette ligne indique au programme `ppp` de couper la ligne s'il n'y a aucun échange pendant cinq minutes (300 secondes). Vous pouvez adapter cette ligne à votre besoin particulier.

```
set openmode active
```

Cette ligne dit au programme `ppp` de négocier la connexion dès que la liaison entre les modems est établie. Certains sites distants le font automatiquement, d'autres non. Cette instruction permet à votre site d'initialiser la négociation.

```
accept chap
```

Cette ligne dit au programme `ppp` d'utiliser le protocole CHAP "Challenge Handsake Autentification Protocole (protocole d'authentification par poignée de main avec défi de décryptage)" pour vous authentifier. Les valeurs échangées entre le système local et le système distant pour transmettre le nom d'utilisateur et le mot de passe sont définies par les lignes "authname" et "authkey"

```
set ifaddr 127.1.1.1/0 127.2.2.2/0 255.255.255.0
```

Cette commande définit une paire d'adresses IP fictives pour les deux extrémités de la liaison point-à-point. Elle dit au programme `ppp` d'utiliser l'adresse 127.1.1.1 pour l'extrémité locale de la liaison gérée par le pilote de périphérique "tun0 (tunnel)"(Reportez-vous au manuel FreeBSD (../handbook/index.html) pour une description de ce pilote) et l'adresse 127.2.2.2 pour l'extrémité distante. Le "/0" qui les suit précise le nombre de digits significatifs de ces adresses, qui peuvent, (et doivent en fait) être négociées entre les deux systèmes une fois la liaison établie. 255.255.255.0 est le masque de sous-réseau qui s'applique à ces deux pseudo-interfaces.

Rappelez-vous, nous avons supposé que votre fournisseur d'accès vous assigne les adresses des deux extrémités de la ligne! Si votre fournisseur vous a donné une adresse particulière, vous pouvez la mentionner sur cette ligne *à la place* de l'adresse 127.1.1.1.

Inversement, si votre fournisseur vous a donné une adresse particulière pour son côté de la liaison, vous pouvez la mentionner sur cette ligne *à la place* de l'adresse 127.2.2.2.

Dans les deux cas, il vaut mieux conserver le "/0" après les deux adresses. Cela laisse la possibilité au programme `ppp` de changer d'adresse(s) *s'il en a besoin*.

```
add 0 0 127.2.2.2
```

Cette dernière ligne dit au programme `ppp` d'ajouter une route par défaut pour le trafic vers l'adresse (fictive) 127.2.2.2 du fournisseur d'accès.

> **Note :** Si vous avez donnée l'adresse que vous a indiqué votre fournisseur d'accès à la place de l'adresse 127.2.2.2 à la ligne "set ifaddr", vous devez utiliser la même adresse ici, à la place de 127.2.2.2.

En ajoutant cette route "fictive" pour le trafic IP, le programme `ppp` peut, successivement, avant et au moment d'établir la liaison:

- Accepter des paquets que FreeBSD ne sait pas encore comment rediriger,
- Etablir la connexion avec votre fournisseur d'accès "à la volé",
- Reconfigurer les adresses IP des deux côtés de la ligne,
- Transmettre les paquets entre votre sytème local et votre fournisseur d'accès,

automatiquement!

Une fois le nombre de secondes défini par l'instruction "timeout" de la section "default" écoulé sans trafic TCP/IP, le programme `ppp` coupe automatiquement la connexion, et le processus peut recommencer.

3.2.2. Le fichier `/etc/ppp/ppp.linkup`

L'autre fichier nécessaire à la configuration de `ppp` est `/etc/ppp/ppp.linkup`. Ce fichier contient les instructions qui définissent ce que `ppp` doit faire une fois que la connexion est établie.

Dans le cas d'une connexion à la demande, le programme `ppp` doit supprimer la route par défaut vers l'adresse IP fictive du système distant (127.2.2.2 dans notre exemple plus haut) et définir une nouvelle route vers l'adresse IP réelle de ce système (obtenue à l'établissement de la connexion).

Voici un bon exemple de fichier `/etc/ppp/ppp.linkup`:

```
#########################################################################=
#
# Fichier PPP Link Up ('/etc/ppp/ppp.linkup')
#
#  Ce fichier est consulté une fois que PPP a établi une connexion
#
#  Ce fichier est lu dans l'ordre suivant :
```

```
#
#  1) En premier, l’adresse IP qui nous a été assignée est recherchée
#     et la/les commande(s) associée(s) exécutée(s)
#
#  2) Si l’adresse IP n’est pas trouvée, alors l’étiquette donnée au
#     démarrage de PPP est recherchée, et la/les commande(s) associée(s)
#     exécutée(s)
#
#  3) Si ni l’une ni l’autre n’ont été trouvées, les commandes associées
#     à l’étiquette ’MYADDR:’ sont exécutées.
#
##########################################################################=

#
# Cette section correspond à la configuration "demand" de
#    /etc/ppp/ppp.conf:
demand:
delete ALL
add 0 0 HISADDR
#
# Toutes les autres configurations de /etc/ppp/ppp.conf utilisent
#    ce qui suit:
#
MYADDR:
add 0 0 HISADDR
########################################################################
# Fin de /etc/ppp/ppp.linkup
```

Remarquez qu’une section de ce fichier s’appelle “demand:”, tout comme dans le fichier `/etc/ppp/ppp.conf`. Cette section dit au programme `ppp` ce qu’il doit faire, une fois la connexion à la demande établie:

1. Supprimer toutes les informations de routage qu’il a déjà créées,
2. Ajouter une route par défaut vers l’adresse réelle de la machine distante.

Il est capital que les configurations définies dans le fichier `/etc/ppp/ppp.conf` et qui comportent les lignes “set ifaddr” et “add 0 0” (i.e.: les configurations de connexion à la demande) exécutent les instructions “delete ALL” et “add 0 0 HISADDR” du fichier `/etc/ppp/ppp.linkup`.

C’est le mécanisme qui contrôle la configuration de la liaison dans le cas d’une connexion à la demande.

Toutes les configurations qui ne sont pas explicitement définies dans le fichier `/etc/ppp/ppp.linkup` utiliseront les commandes listées dans la section “MYADDR:” de ce fichier. C’est ce que feront les connexions manuelles (comme dans notre exemple “interactive:”). Cette section ajoute simplement une route par défaut vers l’adresse IP du fournisseur d’accès (à l’autre extrémité de la ligne).

3.3. Alias IP

Toutes les étapes précédentes s'appliquent à un systeme FreeBSD utilisé pour se connecter à la demande à un fournisseur d'accès Internet.

Si vous cherchez simplement, en lisant ce guide, à connecter votre systeme FreeBSD à l'Internet, vous pouvez passer directement au chapitre Tester le réseau.

Une possibilité très intéressante du programme `ppp` en mode "connexion à la demande" est sa capacité à aiguiller automatiquement le trafic vers les autres systèmes du réseau local. On donne à cela des noms divers "alias IP (IP Aliasing)", "traduction d'adresse réseau (Network Adress Translation)", "mascarade d'adresse (Adress Masquerading)" ou "mandatement transparent (Transparent Proxying)".

Quelle que soit la terminologie employée, ce fonctionnement n'est cependant pas automatique. Si le programme `ppp` est démarré de façon habituelle, il ne fera pas transiter de paquets entre l'interface avec le réseau local et la liaison vers l'extérieur. En effet, seul le système FreeBSD est connecté à l'Internet; les autres systèmes ne peuvent pas partager la même connexion.

Par exemple, si le programme `ppp` est lancé par:

```
# ppp interactive
```

ou:

```
# ppp -auto demand
```

correspondant respectivement aux modes manuel et automatique, alors il établira une connexion à l'Internet pour la *seule* machine FreeBSD.

Pour démarrer le programme `ppp` pour l'utiliser comme passerelle entre le réseau local et l'Internet, il faut utiliser l'une des commandes suivantes:

```
# ppp -alias interactive
```

ou:

```
# ppp -auto -alias demand
```

Alternativement, vous pouvez utiliser l'instruction "alias enable yes" dans votre fichier de configuration de `ppp` (Reportez-vous aux pages de manuel pour plus de détails).

Gardez cela présent à l'esprit si vous passez maintenant au chapitre Configurer les systèmes Windows.

Notes

1. N.d.T: ne disposant du matériel nécessaire au contrôle de certains de ces libellés dans les versions françisées de Windows, ils ont été traduits du document en langue anglaise. En cas d'inexactitude, le lecteur voudra bien rectifier de lui-même.

Chapitre 4. Configurer les Systèmes Windows

Comme indiqué au Chapitre Mise en oeuvre du réseau local, le réseau de notre exemple comporte un système FreeBSD (“Curly”) qui sert de passerelle (ou routeur) pour un réseau local composé de trois moutures différentes de systèmes Windows. Pour que ces stations puissent utiliser Curly comme routeur, elles doivent être correctement configurées. Notez bien que l’on ne vous explique pas ici comment configurer les postes Windows pour qu’ils puissent se connecter eux-mêmes. Si vous cherchez de bonnes explications sur le sujet, voyez http://www.aladdin.co.uk/techweb.

4.1. Configurer Windows 95

Il est relativement simple de configurer Windows 95 pour relier un poste à votre réseau local. Le configuration réseau Windows 95 doit être légérement modifiée pour pouvoir utiliser le système FreeBSD comme passerelle par défaut vers votre fournisseur d’accès. Effectuez les étapes suivantes:

Créer le fichier Windows 95 `hosts`

Pour vous connecter aux autres postes TCP/IP du réseau local, vous devrez installer une copie du fichier `hosts` que vous avez créé sur votre système FreeBSD au paragraphe Renseigner la liste des autres machines du réseau.

- Cliquez sur le bouton “Démarrer”, choisissez “Exécuter ...”, lancez **`notepad \WINDOWS\HOSTS`** et cliquez sur “OK”,
- Sous l’éditeur, entrez les noms et les adresses des machines recopiés du fichier `hosts` du paragraphe Renseigner la liste des autres machines du réseau,
- Une fois que c’est fait, quittez `notepad` (assurez-vous d’avoir enregistré le fichier!).

Définir le configuration réseau TCP/IP Windows 95

- Cliquez sur le bouton “Démarrer” de la barre de tâches, Choisissez “Paramètres” et “Panneau de Configuration”,
- Double-cliquez sur l’icône réseau pour l’ouvrir.

  Les paramètres des éléments du réseau sont tous affichés.
- Sélectionnez l’option “Configuration” et faites défiler la liste des composants installés pour choisir la ligne “TCP/IP ->*`votre interface`*” (où *`votre interface`* est le nom ou le modèle de votre contrôleur Ethernet),

  Si “TCP/IP” ne figure pas dans la liste des composants réseau installés, cliquez sur le bouton “Ajouter” pour l’installer avant de continuer

  > **Astuce :** Ajouter | Protocoles | Microsoft | TCP/IP | OK.

- Cliquez sur le bouton “Propriétés” pour afficher la liste des paramètres associés au composant TCP.

Configurer les informations relatives à l’adresse IP

- Choisissez l'onglet "Adresse IP"
- Enfoncez le bouton "Spécifier une adresse IP".

  (Dans notre exemple, le système Windows 95 est celui que nous avons appelé "Larry".)
- Dans la zone "Adresse IP", entrez "192.168.1.2",
- Entrez "255.255.255.0" dans le zone "Masque de Sous-réseau".

Configurer les informations relatives à la passerelle

- Cliquez sur l'onglet "Passerelle".

  Dans le réseau de notre exemple, c'est le système FreeBSD qui sera notre passerelle vers l'Internet (routant les paquets entre le réseau local Ethernet et la connexion PPP). Entrez l'adresse IP de l'interface Ethernet de la machine FreeBSD, "192.168.1.1", dans la zone "Nouvelle Passerelle" et cliquez sur le bouton "Ajouter". S'il y a d'autres passerelles définies dans les "Passerelles Installées", peut-être devrez-vous les supprimer.

Configurer les informations relatives au DNS

Nous supposons que votre fournisseur d'accès Internet vous a donné une liste de serveurs de noms de domaines (Domain Name Servers) - ou "Serveurs DNS" - que vous devez utiliser. Si vous voulez mettre en service un serveur DNS sur votre machine FreeBSD, reportez-vous au chapitre Exercices pour l'étudiant intéressé pour avoir des indications sur l'installation d'un serveur DNS sur une machine FreeBSD.

- Cliquez sur l'onglet "Configuration DNS",
- Assurez-vous que l'option "Activer DNS" est bien sélectionnée,

  (Si cette option n'est pas sélectionnée, seules les machines définies dans votre fichier `hosts` seront accessibles et vous ne pourrez pas "surfer" comme vous le souhaitez!)
- Dans la zone "Hôte", entrez le nom du poste Windows 95, dans notre cas : "Larry",
- Dans la zone "Domaine", entrez le nom du réseau local, dans notre cas : "my.domain",
- Dans la zone "Ordre de recherche DNS", entrez les adresses IP des serveurs DNS que votre fournisseur vous a données, cliquez sur "Ajouter" à chaque adresse que vous définissez. Répétez l'opération autant de fois que nécessaire pour entrer toutes les adresses que votre fournisseur vous a indiquées.

Autres options TCP/IP Windows 95

Pour nos besoins, les paramètres des sections "Avancées", "Configuration WINS" et "Liens" sont inutiles.

Si vous souhaitez utiliser le service de noms Internet de Windows (Windows Internet Naming Service - "WINS"), vous trouverez plus d'informations sur http://www.localnet.org, en particulier en ce qui concerne le partage de fichiers de façon transparente à travers l'Internet.

Pour finir

- Cliquez sur "OK" pour fermer la fenêtre "Propriétés TCP/IP",
- Cliquez sur "OK" pour fermer la fenêtre "Panneau de Contrôle Réseau",

- Redémarrez votre ordinateur si on vous le demande.

C'est tout!

4.2. Configurer Windows NT

Il est relativement simple de configurer Windows NT pour relier un poste à votre réseau local. Les procédures sont semblables à celles suivies pour Windows 95 à quelques différences près quant à l'interface utilisateur.

Les étapes décrites ci-dessous s'appliquent à Windows NT 4.0 Workstation, mais les principes sont les mêmes pour Windows NT 3.5x. Si vous installez un poste Windows NT 3.5x, vous pouvez vous référer au paragraphe Configurer Windows for Workgroups, en effet, l'interface utilisateur est la même pour NT 3.5x et WfW.

Effectuez les opérations suivantes:

Créer le fichier Windows NT `hosts`

Pour vous connecter aux autres postes TCP/IP du réseau local, vous devrez installer une copie du fichier `hosts` que vous avez créé sur votre système FreeBSD au paragraphe Configurer Windows for Workgroups.

- Cliquez sur le bouton "Démarrer", choisissez "Exécuter ...", lancez **`notepad \WINDOWS\SYSTEM\DRIVERS\ETC\HOSTS`** et cliquez sur "OK",
- Sous l'éditeur, entrez les noms et les adresses des machines recopiés du fichier `hosts` du paragraphe Renseigner la liste des autres machines du réseau,
- Une fois que c'est fait, quittez `notepad` (assurez-vous d'avoir enregistré le fichier!).

Définir le configuration réseau TCP/IP Windows NT

- Cliquez sur le bouton "Démarrer" de la barre de tâches, Choisissez "Paramètres" et "Panneau de Configuration",
- Double-cliquez sur l'icône réseau pour l'ouvrir,
- Dans l'onglet "Identification", vérifiez les informations des zones "Nom d'Ordinateur" et "Workgroup". Dans notre exemple, "Shemp" sera le nom de la machine, et "Stooges" celui du groupe de travail. Choisissez "Modifier" pour corriger ces valeurs au besoin,
- Choisissez l'onglet "Protocoles".

  Les Protocoles Réseau installés seront affichés. Il peut y en avoir un certain nombre, mais le seul qui nous intéresse ici est le "Protocole TCP/IP". Si le "Protocole TCP/IP" n'apparaît pas dans la liste, cliquez sur le bouton "Ajouter" pour l'installer.

  > **Astuce :** Ajouter | Protocole TCP/IP | OK.

- Sélectionnez "Protocole TCP/IP" et cliquez sur le bouton "Propriétés".

  Les différentes propriétés associées à TCP/IP apparaîtront.

Configurer les informations relatives à l'adresse IP

Vérifiez que l'interface Ethernet est bien définie dans la zone "Carte Réseau"; si ce n'est pas le cas, parcourez la liste des contrôleurs jusqu'à ce que l'interface adéquate soit affichée.

- Choisissez l'option "Spécifier une Adresse IP" pour activer les trois zones de texte,

  Dans notre exemple, le poste Windows NT est celui qui s'appele "Shemp".
- Dans la zone "Adresse IP", entrez "192.168.1.4",
- Entrez "255.255.255.0" dans le zone "Masque de Sous-réseau.".

Configurer les informations relatives à la passerelle

Dans le réseau de notre exemple, c'est le système FreeBSD qui sera notre passerelle vers l'Internet (routant les paquets entre le réseau local Ethernet et la connexion PPP).

- S'il n'y a rien encore dans la zone "Nouvelle Passerelle", entrez l'adresse IP de l'interface Ethernet de la machine FreeBSD, "192.168.1.1", dans cette zone et cliquez sur le bouton "Ajouter". Sinon, utilisez le bouton "Avancées" et la fonction "Ajouter" pour ajouter cette nouvelle passerelle.

  S'il y a d'autres passerelles définies dans les "Passerelles Installées", peut-être devrez-vous les supprimer.

Configurer les informations relatives au DNS

Nous supposons que votre fournisseur d'accès Internet vous a donné une liste de serveurs de noms de domaines (Domain Name Servers) - ou "Serveurs DNS" - que vous devez utiliser.

Si vous voulez mettre en service un serveur DNS sur votre machine FreeBSD, reportez-vous au chapitre Exercices pour l'étudiant intéressé pour avoir des indications sur l'installation d'un serveur DNS sur une machine FreeBSD.

- Cliquez sur le bouton "DNS",
- Dans la zone "Nom d'Hôte", entrez le nom du poste Windows NT, dans notre cas : "Shemp",
- Dans la zone "Domaine", entrez le nom du réseau local, dans notre cas : "my.domain",
- Dans la zone "Ordre de recherche du service DNS", entrez les adresses IP des serveurs DNS que votre fournisseur vous a données, cliquez sur "Ajouter" à chaque adresse que vous définissez. Répétez l'opération autant de fois que nécessaire pour entrer toutes les adresses que votre fournisseur vous a indiquées.

Autres options TCP/IP Windows NT

Pour nos besoins, les paramètres des sections "Adresses WINS" et "Routage" sont inutiles.

Si vous souhaitez utiliser le service de noms Internet de Windows (Windows Internet Naming Service - "WINS"), vous trouverez plus d'informations sur http://www.localnet.org, en particulier en ce qui concerne le partage de fichiers de façon transparente à travers l'Internet.

Pour finir

- Cliquez sur “OK” pour fermer la fenêtre “Propriétés TCP/IP”,
- Cliquez sur “OK” pour fermer la fenêtre “Panneau de Contrôle Réseau”,
- Redémarrez votre ordinateur si on vous le demande.

C’est tout!

4.3. Configurer Windows for Workgroups

Pour qu’un poste Windows for Workgroups puisse être utilisé en réseau, il faut avoir installé les pilotes qui sont sur la disquette Microsoft TCP/IP. Ces pilotes ne sont pas inclus avec WfW et les disquettes qui vont avec. S’il vous en faut une copie, ils sont disponibles sur ftp://ftp.microsoft.com/peropsys/windows/public/tcpip. (ftp://ftp.microsoft.com/peropsys/windows/public/tcpip)

Une fois que les pilotes TCP/IP ont été installés, effectuez les opérations suivantes:

Créer le fichier Windows for Workgroups `hosts`

Pour vous connecter aux autres postes TCP/IP du réseau local, vous devrez installer une copie du fichier `hosts` que vous avez créé sur votre système FreeBSD au paragraphe Renseigner la liste des autres machines du réseau.

- Dans le Gestionnaire de Programmes, cliquez sur le bouton “Fichier”, choisissez “Exécuter ...”, lancez **`notepad \WINDOWS\HOSTS`** et cliquez sur “OK”,
- Sous l’éditeur, entrez les noms et les adresses des machines recopiés du fichier `hosts` de Renseigner la liste des autres machines du réseau,
- Une fois que c’est fait, quittez `notepad` (assurez-vous d’avoir enregistré le fichier!).

Définir le configuration réseau TCP/IP Windows for Workgroups

- Dans le Gestionnaire de Programmes, ouvrez le groupe “Réseau” en double-cliquant sur l’icône,
- Double-cliquez sur l’icône “Paramètres Réseau”,
- Dans la boîte de dialogue “Pilotes Réseau”, double-cliquez sur l’option “Microsoft TCP/IP-32”.

Configurer les informations relatives à l’adresse IP

Vérifiez que l’interface Ethernet est bien définie dans la liste “Carte Réseau”; si ce n’est pas le cas, parcourez la liste des contrôleurs jusqu’à ce que l’interface adéquate soit affichée et sélectionnez-la en cliquant dessus.

- Vérifiez que l’option “Activer la Configuration DHCP Automatique” n’est pas sélectionnée. Si c’est le cas, cliquez dessus pour effacer le “X”.
- Dans notre exemple, le poste Windows for Workgroups est celui qui s’appelle “Moe”; dans la zone “Adresse IP”, entrez “192.168.1.3”,
- Entrez “255.255.255.0” dans le zone “Masque de Sous-réseau”.

Configurer les informations relatives à la passerelle

Dans le réseau de notre exemple, c'est le système FreeBSD qui sera notre passerelle vers l'Internet (routant les paquets entre le réseau local Ethernet et la connexion PPP).

- Entrez l'adresse IP de l'interface Ethernet de la machine FreeBSD, "192.168.1.1", dans la zone "Passerelle par Défaut".

Configurer les informations relatives au DNS

Nous supposons toujours que votre fournisseur d'accès Internet vous a donné une liste de serveurs de noms de domaines (Domain Name Servers) - ou "Serveurs DNS" - que vous devez utiliser.

Si vous voulez mettre en service un serveur DNS sur votre machine FreeBSD, reportez-vous au chapitre Exercices pour l'étudiant intéressé pour avoir des indications sur l'installation d'un serveur DNS sur une machine FreeBSD.

- Cliquez sur l'onglet "Configuration DNS",
- Dans la zone "Hôte", entrez le nom du poste Windows for Workgroups, dans notre cas : "Moe",
- Dans la zone "Domaine", entrez le nom du réseau local, dans notre cas : "my.domain",
- Dans la zone "Ordre de recherche DNS", entrez les adresses IP des serveurs DNS que votre fournisseur vous a données, cliquez sur "Ajouter" à chaque adresse que vous définissez. Répétez l'opération autant de fois que nécessaire pour entrer toutes les adresses que votre fournisseur vous a indiquées,
- Cliquez sur "OK" pour fermer la fenêtre "Configuration DNS".

Pour finir

- Cliquez sur "OK" pour fermer la fenêtre "Propriétés TCP/IP",
- Cliquez sur "OK" pour fermer la fenêtre "Panneau de Contrôle Réseau",
- Redémarrez votre ordinateur si on vous le demande.

C'est tout!

Chapitre 5. Tester le réseau

Une fois que vous aurez terminé les opérations nécessaires décrites plus haut, votre passerelle PPP vers l'Internet devrait être opérationnelle.

5.1. Tester la connexion vers l'extérieur

La première chose à vérifier est que la connexion s'établisse entre votre modem et votre fournisseur d'accès.

5.2. Tester le réseau local

*** à faire ***

Chapitre 6. Exercices pour l'étudiant intéressé

6.1. Gérer un mini-DNS

Bien que gérer une hiérarchie du service de noms de domaines ("Domain Name Service" - DNS) puisse être une tâche diabolique, il est tout à fait faisable d'installer un mini-serveur DNS sur le sytème FreeBSD qui vous sert aussi de passerelle vers votre fournisseur d'accès.

A partir des fichiers existants dans `/etc/namedb` après installation de FreeBSD, il est possible de définir un serveur DNS qui ait autorité sur le réseau de notre exemple et serve d'interface avec l'architecture DNS de l'Internet.

Pour cette configuration minimale, il suffit de trois fichiers:

- `/etc/namedb/named.boot`
- `/etc/namedb/named.root`
- `/etc/namedb/mydomain.db`

Le fichier `/etc/namedb/named.root` est automatiquement initialisé à l'installation de FreeBSD, les deux autres fichiers doivent être renseignés manuellement.

6.1.1. Le fichier `/etc/namedb/named.boot`

Le fichier `/etc/namedb/named.boot` décrit la configuration initiale du serveur DNS. Essentiellement, il indique au serveur de noms:

1. Où sont les fichiers de configuration,
2. De quels "noms de domaines" il a la responsabilité,
3. Où trouver les autres serveurs DNS.

Avec l'éditeur `ee`, créez un fichier `/etc/namedb/named.boot` comme suit:

```
; fichier de démarrage pour un mini-serveur de noms de domaines

directory       /etc/namedb

; type    domain              source machine/fichier   fichier de sauvegarde

cache     .                   named.root
primary   my.domain.          mydomain.db
```

Les lignes qui commencent par un point-virgule sont des commentaires. Voici la signification des autres lignes:

- `directory /etc/namedb`

Dit au serveur de noms où trouver les fichiers de configuration référencés dans la suite du fichier `/etc/namedb/named.boot`.

- ```
  cache . named.root
  ```

  Dit au serveur de noms que la liste des serveurs de la racine ("Root Servers") se trouve dans le fichier `named.root`. (Ce fichier est inclus dans l'installation standard et n'est pas décrit ici.)

- ```
  primary my.domain mydomain.db
  ```

  Dit au serveur de noms qu'il a "autorité" sur un domaine DNS appelé "my.domain" et que la liste des adresses IP des machines du domaine "my.domain" (le réseau local) se trouve dans le fichier `/etc/namedb/mydomain.db`.

Une fois créé et sauvegardé le fichier `/etc/namedb/named.boot`, passez à l'étape suivante pour créer le fichier `/etc/namedb/named.db`.

6.1.2. Le fichier `/etc/namedb/named.db`

Le fichier `/etc/namedb/named.boot` liste les noms et les adresses IP de *toutes* les machines du réseau local.

Pour une description détaillée des instructions de ce fichier, reportez-vous aux pages de manuel de `named`.

Le fichier `/etc/namedb/named.boot` de notre serveur DNS minimum contient les lignes suivantes:

```
@        IN SOA  my.domain. root.my.domain.  (
                               961230  ; Serial
                               3600    ; Refresh
                               300     ; Retry
                               3600000 ; Expire
                               3600 )  ; Minimum
       IN NS   curly.my.domain.

curly.my.domain.        IN A    192.168.1.1     # The FreeBSD box
larry.my.domain.        IN A    192.168.1.2     # The Win'95 box
moe.my.domain.          IN A    192.168.1.3     # The WfW box
shemp.my.domain.        IN A    192.168.1.4     # The Windows NT box

$ORIGIN 1.168.192.IN-ADDR.ARPA
                IN NS   curly.my.domain.
1               IN PTR  curly.my.domain.
2               IN PTR  larry.my.domain.
3               IN PTR  moe.my.domain.
4               IN PTR  shemp.my.domain.

$ORIGIN 0.0.127.IN-ADDR.ARPA
                IN NS   curly.my.domain.
1               IN PTR  localhost.my.domain.
```

Brièvement, ce fichier déclare que le serveur DNS local est:

- Le "Début d'Autorité" (Start Of Authority) pour le domaine appelé "my.domain",
- Le serveur de noms (Name Server - "NS") pour "my.domain",
- Responsable de la résolution inverse des adresses IP qui commencent par "192.168.1" et "127.0.0" ("$ORIGIN ...").

Pour ajouter de nouvelles machines, vous devez ajouter deux lignes pour chaque système; une dans la section du haut où les noms des systèmes sont associés aux adresses Internet ("IN A"), et une autre ligne qui associe inversement l'adresse au nom de la machine ("IN PTR"), dans la section "$ORIGIN 1.168.192.IN-ADDR.ARPA".

6.1.3. Démarrer le serveur DNS

Par défaut, le serveur DNS (`/etc/sbin/named`) n'est pas lancé au démarrage du système. Vous pouvez modifier ce comportement en rectifiant une seule ligne dans le fichier `/etc/rc.conf`, comme suit:

Avec l'éditeur `ee`, visualisez le fichier `/etc/rc.conf`. Descendez d'une quarantaine de lignes jusqu'à ce que vous trouviez la section qui dit:

```
---
named_enable="NO"                        # Run named, the DNS server (or NO).
named_flags="-b /etc/namedb/named.boot" # Flags to named (if enabled).
---
```

et modifiez-la en:

```
---
named_enable="YES"                       # Run named, the DNS server (or NO).
named_flags="-b /etc/namedb/named.boot" # Flags to named (if enabled).
---
```

Sauvegardez le fichier et redémarrez le système.

Vous pouvez aussi démarrer le serveur DNS avec la commande suivante:

`# named -b /etc/namedb/named.boot`

Chaque fois que vous modifiez les fichiers du répertoire `/etc/namedb/named.boot`, vous devez faire relire ces modifications par le serveur DNS avec la commande suivante:

`# kill -HUP `cat /var/run/named.pid``

6.2. Essayer les filtres PPP

Le programme `ppp` a la capacité d’appliquer des règles de filtrage au trafic qu’il route. Bien que cela ne soit pas aussi sécurisé qu’un véritable coupe-feu, cela autorise quelques contrôles d’accès à la liaison.

(“`man ipfw`” pour plus d’informations sur la configuration d’un système FreeBSD plus sécurisé.)

La documentation complète des différents filtres et règles utilisables avec le programme `ppp` se trouve dans les pages de manuel de `ppp`.

Il y a quatre classes de filtres qui s’appliquent au programme `ppp`:

- `alive` - filtre compteur d’accès (ou “Keep Alive” - garder en vie)

  Ils définissent quels événements sont ignorés par l’instruction `set timeout=`.
- `dial` - filtre d’appel

  Ils définissent quels événements sont ignorés par `ppp` en mode connexion à la demande.
- `in` - filtre d’entrée

  Ils définissent quels paquets entrants sont acceptés ou refusés par le programme `ppp`.
- `out` - filtre de sortie

  Ils définissent quels paquets sortants sont acceptés ou refusés par le programme `ppp`.

L’exemple qui suit est un extrait de la configuration d’un système opérationnel qui donne une bonne base pour un fonctionnement Internet “normal”, tout en empêchant que `ppp` accepte n’importe quoi sur la liaison. Les commentaires décrivent la logique de chaque jeu de règles.:

```
#
# filtres KeepAlive
# ne pa prendre en compte les paquets ICMP,DNS et RIP
#
 set filter alive 0 deny icmp
 set filter alive 1 deny udp src eq 53
 set filter alive 2 deny udp dst eq 53
 set filter alive 3 deny udp src eq 520
 set filter alive 4 deny udp dst eq 520
 set filter alive 5 permit 0/0 0/0
#
# filtre d’appel:
#  Note:  ICMP établira la connexion dans cette configuration!
#
 set filter dial 0 permit 0/0 0/0
#
# autoriser les paquets ident
#
 set filter in 0 permit tcp dst eq 113
 set filter out 0 permit tcp src eq 113
#
# autoriser les connexions telnet à l’Internet
#
 set filter in 1 permit tcp src eq 23 estab
```

```
 set filter out 1 permit tcp dst eq 23
#
# autoriser l'accès ftp à l'Internet
#
 set filter in 2 permit tcp src eq 21 estab
 set filter out 2 permit tcp dst eq 21
 set filter in 3 permit tcp src eq 20 dst gt 1023
 set filter out 3 permit tcp dst eq 20
#
# autoriser les requêtes DNS
#
 set filter in 4 permit udp src eq 53
 set filter out 4 permit udp dst eq 53
#
# autoriser les transferts de zone DNS
#
 set filter in 5 permit tcp src eq 53
 set filter out 5 permit tcp dst eq 53
#
# autoriser l'accès depuis/vers le réseau local
#
 set filter in 6 permit 0/0 192.168.1.0/24
 set filter out 6 permit 192.168.1.0/24 0/0
#
# autoriser les réponses au ping et traceroute
#
 set filter in 7 permit icmp
 set filter out 7 permit icmp
 set filter in 8 permit udp dst gt 33433
 set filter out 9 permit udp dst gt 33433
#
# autoriser cvsup
#
 set filter in 9 permit tcp src eq 5998
 set filter out 9 permit tcp dst eq 5998
 set filter in 10 permit tcp src eq 5999
 set filter out 10 permit tcp dst eq 5999
#
# autoriser NTP pour la synchronization des horloges
#
 set filter in 11 permit tcp src eq 123 dst eq 123
 set filter out 11 permit tcp src eq 123 dst eq 123
 set filter in 12 permit udp src eq 123 dst eq 123
 set filter out 12 permit udp src eq 123 dst eq 123
#
# SMTP serait une bonne idée!
#
 set filter in 13 permit tcp src eq 25
 set filter out 13 permit tcp dst eq 25
#
#
# nous utilisons beaucoup whois, donc nous le laissons passer
#
```

```
 set filter in 14 permit tcp src eq 43
 set filter out 14 permit tcp dst eq 43
 set filter in 15 permit udp src eq 43
 set filter out 15 permit udp dst eq 43
#
# si aucune des conditions ci-dessus n'est remplie, le paquet est refusé
#-------
```

Il peut y avoir jusqu'à vingt règles dans chaque classe de filtres. Dans chaque classe, les règles sont numérotées séquentiellement de 0 à 20, *mais aucune règle dans une classe particulière n'est appliquée tant que la règle "0" n'est pas définie!*

Si vous décidez de *ne pas* utiliser de règles de filtrage dans votre configuration du programme `ppp`, alors *TOUT* le trafic de/vers votre système sera accepté lorsqu'il est connecté à votre fournisseur d'accès.

Si vous décidez d'appliquer des règles de filtrages, ajoutez les lignes précédentes à votre fichier `/etc/ppp/ppp.conf` dans l'une des sections "default", "demand" ou "interactive" (ou à toutes - c'est à vous de voir).