Linux IP Masquerade HOWTO

David Ranch, *dranch@trinnet.net*; Ambrose Au, *ambrose@writeme.com*
Version Française par Mathieu Arnold, *arn_mat@club-internet.fr*

v1.77, 27 Juillet 1999

Ce document décrit la procédure pour mettre en place la fonctionnalité IP Masquerade sur un ordinateur Linux. L'IP MASQ est une forme de NAT (Network Address Translation, c'est a dire, Translation d'adresse réseau) qui permet de connecter sur Internet des ordinateurs n'ayant pas d'adresses IP réservées, à travers l'IP de votre ordinateur Linux.

Table des matières

1 Introduction

1.1 Introduction à l'IP Masquerading aka IP MASQ

Ce document décrit la façon de mettre en oeuvre l'IP MASQ sur une machine Linux. IP MASQ est une forme de NAT qui permet de connecter sur Internet des ordinateurs n'ayant pas d'adresses IP réservées, à travers l'IP de votre ordinateur Linux. Il est possible de connecter vos machines internes a la machine Linux avec des technologies réseau telles l'Ethernet, le TokenRing, le FDDI, comme d'autres formes de connexions comme une connexion PPP ou SLIP. Ce document utilise l'Ethernet comme exemple car c'est le scénario le plus commun.

> ***Ce document est destiné aux utilisateurs utilisant un des noyaux Linux stable : 2.0.36+ et 2.2.9+ sur un compatible PC. Les noyaux les plus anciens comme les 1.2, 1.3 et 2.1 NE SONT PAS couverts par ce document, et certaines versions sont considérées comme dangereuses. Mettez a jour votre noyau pour utiliser un noyau stable avant de vous mettre a utiliser l'IP MASQ.***
>
> ***Si vous configurez l'IP MASQ pour être utilisé sur un Macintosh, envoyez un email à Taro Fukunaga,*** **tarozax@earthlink.net** ***pour recevoir un exemplaire de sa version de ce HOWTO pour MkLinux.***

1.2 Mise en garde, Feedback et Crédits

En tant que débutant, je trouve qu'il est très déroutant de mettre en place l'IP masquerade sur un noyau, (un 1.2 a cette époque là). Bien qu'il y ait une FAQ et une mailing list, il n'y avais pas de documents qui soit spécifiques à cela. Il y avais aussi quelques demandes sur la mailing list pour des documents tels qu'un HOWTO. J'ai donc décidé d'écrire ce document comme un point de départ pour un nouvel utilisateur et peut-être comme une première brique pour que d'autres utilisateurs expérimentés ajoutent leurs briques. Si vous avez la moindre idée a propos de ce document, une correction, n'hésitez pas à nous en faire part afin que nous puissions l'améliorer.

Ce document étais a l'origine largement inspiré de la FAQ de Ken Eves, ainsi que de nombreux messages très utiles de la mailing list d'IP Masquerading. Je tiens à adresser des remerciements tout particuliers à M. Matthew Driver qui, par ses messages sur la mailing list, m'a donné envie de mettre en place l'IP Masquerading et d'écrire ce document. Récemment, David Ranch a réécrit ce HOWTO et a ajouté un bon paquet de sections pour rendre ce document encore plus complet possible.

N'hésitez pas à nous envoyer tout commentaire à *ambrose@writeme.com* et *dranch@trinnet.net* pour toute erreur ou tout oubli dans ce document. L'avenir de cet HOWTO sera fortement influencé par les réactions que j'aurais de votre part.

Ce HOWTO a été conçu pour être un guide compréhensible pour que l'IP Masquerading fonctionne chez vous le plus vite possible. Comme ni Ambrose ni David ne sont des techniciens, il se peut que vous trouviez les informations de ce document moins générales et/ou moins objectives. Les dernières nouvelles et informations sur ce HOWTO et l'IP MASQ pourront être trouvées sur le site web de l'*IP Masquerading Ressource*, dont nous nous occupons activement. Si vous désirez poser des questions techniques à propos d'IP Masquerade, veuillez souscrire à la mailing list IP Masquerade au lieu d'envoyer des mails à Ambrose ou David. La majorité des questions se rapportant au masquerading sont bien connues des utilisateurs et trouveront réponse sur la liste. De plus, la réponse donnée sur la liste sera beaucoup plus rapide que si elle est posée à Ambrose ou David.

La dernière version de ce document peut être obtenue aux adresses suivantes, il y a aussi des versions HTML et PostScript :

- *http ://ipmasq.cjb.net/ : The IP Masquerade Resources*
- *http ://ipmasq2.cjb.net/ : The IP Masquerade Resources MIRROR*
- *The Linux Documentation Project*
- *Dranch's Linux page*
- Référez vous aussi à *IP Masquerade Resource Mirror Sites Listing* `<http://ipmasq.cjb.net/index.html##mirror>` pour la liste des miroirs proche de vous.

1.3 Copyright & Dénégation

Les informations contenues dans ce document sont correcte dans l'état des connaissances d'Ambrose et David. Toutefois, la fonctionnalité IP MASQ de Linux est écrite par des humains, et donc, il y a des chances que des bugs apparaissent de temps en temps.

Personne n'est responsable pour un quelconque dommage sur vos ordinateur(s) ainsi qu'une quelconque autre perte due à l'utilisation des informations contenues dans ce document. C'est à dire :

> **LES AUTEURS ET TOUS LES MAINTENEURS NE SONT EN AUCUN CAS RESPONSABLES DES DOMMAGES OCCASIONNÉS PAR L'USAGE DES INFORMATIONS CONTENUES DANS CE DOCUMENT, QUELS QU'ILS SOIENT.**

2 Connaissances de base

2.1 Qu'est-ce que l'IP Masquerade ?

LÍP Masquerade est une fonctionnalité réseau de Linux similaire aux nombreuses NAT (Network Address Translation) présents dans beaucoup de par feux commerciaux et de routeurs réseau. Par exemple, une machine Linux connectée a internet via une liaison PPP, Ethernet, etc., l'IP MASQ permet a d'autres ordinateurs connectés a la machine Linux (via PPP, Ethernet, etc.) d'accéder aussi a internet. L'IP MASQ de Linux permet cela même si les machines internes n'ont pas *d'adresse IP officielle*.

L'IP MASQ permet a un ensemble de machines d'accéder a internet de manière *invisible* via la passerelle MASQ. Pour les autres machines sur internet, tout le trafic sortant aura comme source la machine Linux IP MASQ elle même. En plus de ces fonctionnalités, l'IP Masquerade fournis les bases de la création d'un environnement réseau totalement sécurisé. Avec un firewall bien intégré, braver la sécurité d'un système utilisant le masquerading ainsi que le réseau interne est peine perdue.

2.2 Où cela en est ?

L'IP Masquerade est utilisé depuis quelques années et est arrivé a maturité lorsque Linux est arrivé dans les noyaux 2.2. Cette fonctionnalité est supportée en standard depuis la série 1.3. De nombreuses personnes, et même des entreprises l'utilisent, avec des résultats excellents.

Les utilisation courantes comme la navigation web, le telnet, ftp, ping, traceroute, etc marche très bien avec l'ip masquerade. D'autres protocoles de communication comme le ftp, l'irc et le RealAudio marchent très bien avec le module IP MASQ approprié. D'autres programmes spécifiques a un réseau comme les stream audio (MP3s, True Speech, etc) marchent aussi. Certains ont même obtenus de bon résultats avec des logiciels de vidéoconférence.

Veuillez consulter la section 6.2 () pour une liste complète des logiciels supportés.

IP Masquerade marche bien comme serveur pour d'autres 'machines clientes' avec différents OS. Il y a eu des échos de succès avec des machines MASQuées utilisant :

- Unix : Sun Solaris, *BSD, Linux, Digital UNIX, etc.
- Microsoft Windows 95/98, Windows NT, et Windows pour Workgroups (avec le paquetage TCP/IP)
- IBM OS/2
- Apple Macintosh MacOS avec soit MacTCP soit Open Transport
- Les systèmes basés sur le DOS avec les pilotes packet et le paquetage NCSA Telnet
- Les VAX
- Compaq/Digital Alpha avec Linux ou Windows NT
- même les Amiga avec AmiTCP ou la pile AS225

La liste continue, mais en fait, si votre OS parle TCP/IP, il devrais marcher avec IP Masquerade !

2.3 A qui peut être utile IP Masquerade ?

- Si vous avez un hôte Linux connecté à Internet et
- si vous avez un ou plusieurs ordinateurs utilisant TCP/IP, connecté à un hôte Linux sur un réseau local, et/ou
- si votre hôte Linux a un ou plusieurs modems et joue le rôle de serveur PPP ou SLIP, et que
- ces **AUTRES** machines ne possèdent pas d'adresse IP officielle ou publique (c'est a dire, avec une adresse IP privée).
- et bien sûr, si vous désirez que ces **AUTRES** machines soient également connectées sur Internet sans débourser un centime de plus pour acheter des IP officielles ou publiques a votre FAI et configurer un routeur sous Linux ou même acheter un routeur.

2.4 Qui n'a pas besoin d'IP Masquerade ?

- Si votre machine est un hôte isolé, connecté sur Internet, (bien que mettre en place un firewall soit une bonne idée), ou
- si vous avez déjà obtenu des adresses officielles pour vos **AUTRES** machines,

– et bien sûr, si vous n'aimez pas l'idée de connecter toutes les **AUTRES** machines gratuitement, de cette manière en utilisant Linux et vous avez plus confiance en utilisant de chers produits commerciaux qui font exactement la même chose...

2.5 Comment fonctionne IP Masquerade ?

D'après la FAQ IP Masquerade originelle, de Ken Eves :

```
Voici un schéma du plus simple cas possible~:

   SLIP/PPP         +------------+                        +-------------+
   vers le FAI      |  Linux     |       SLIP/PPP         | Peu_importe |
  <---------- modem1|   #1       |modem2 ----------- modem3|             |
    111.222.333.444 |            |           192.168.0.100 |             |
                    +------------+                        +-------------+

  Dans le schéma ci-dessus, un ordinateur sous Linux (Linux #1), utilisant
  IP_MASQUERADING est connecté à Internet par un lien SLIP ou PPP,
  utilisant modem1. Il possède l'adresse IP (officielle)
  111.222.333.444. Il a aussi un modem2 de branché qui permet aux
  appelants de se connecter et d'initier une connexion PPP ou SLIP.

  Le second système (qui n'utilise par forcément Linux comme système
  d'exploitation) se connecte par modem sur l'hôte Linux #1 et entame une
  liaison SLIP ou PPP. Il NE possède PAS d'adresse IP officielle donc il
  utilise l'adresse privée 192.168.0.100 (voir ci-dessous pour plus
  d'informations).

  Avec l'option IP Masquerade et un routage configuré correctement, la
  machine "Peu_importe" peut interagir avec Internet comme si elle était
  directement connectée (à quelques petites exceptions près).
```

```
Pour citer Pauline Middlelink~:

  N'oublie pas de rappeler que la machine "Peu_importe" doit déclarer l'hôte
  Linux #1 comme passerelle (que cela soit la route par défaut ou juste un
  sous réseau importe peu). Si la machine "Peu_importe" ne peut pas le
  faire, l'hôte Linux devra être configurée pour faire du proxy arp pour
  toutes les adresses routées, mais la mise en place et la configuration du
  proxy arp est hors du domaine de ce document.

Ce qui suit est l'extrait d'un article de comp.os.linux.networking qui a été
modifié pour utiliser les noms des machines de l'exemple ci-dessus~:

   o J'indique à la machine "Peu_importe" que le serveur Linux est sa
     passerelle.
   o Quand un paquet en provenance de Peu_importe arrive sur la machine
     Linux, elle va lui assigner un nouveau numéro de port, et indiquer sa
     propre adresse IP dans l'entête du paquet, tout en sauvegardant
     l'entête originale. Le serveur IP MASQ va alors envoyer le paquet
     modifié à travers son interface SLIP ou PPP, vers Internet.
```

```
  o Lorsqu'un paquet en provenance d'Internet revient sur la machine Linux,
    Linux regarde si le numéro de port est un de ceux assignés à l'étape
    précédente, elle va modifier à nouveau l'entête pour y remettre les
    numéros de port et adresses IP originaux, et alors envoyer le paquet à
    la machine Peu_importe.
  o L'hôte qui a envoyé le paquet ne verra jamais la différence.
```

Un autre example d'IP Masquerading :

Voici ci-dessous le schéma d'un exemple classique :

```
+----------+
|          |  Ethernet
| Ordi A   |::::::
|          |.2  ~:192.168.0.x
+----------+    ~:
                ~:     +----------+   lien
+----------+    ~:   .1|  Linux   |   PPP
|          |    ~::::::| Masq-Gate|:::::::::::::::// Internet
| Ordi B   |::::::     |          | 111.222.333.444
|          |.3  ~:     +----------+
+----------+    ~:
                ~:
+----------+    ~:
|          |    ~:
| Ordi C   |::::::
|          |.4
+----------+

|                      |          |
| <- Réseau interne -> |          | <- Réseau externe ----->
|                      |          |
```

Dans cet exemple, il y a (4) ordinateurs qui nous intéressent. Il y a aussi certainement quelquechose loin a droite ou votre connexion PPP arrive (un serveur, etc.) et qu'il y a une machine distant très très loin a droite sur internet avec laquelle nous voulons communiquer. La machine Linux `Masq-Gate` est la passerelle IP MASQ que tout le réseau de machines `Ordi A`, `Ordi B` et `Ordi C` utilisent pour se rendre sur internet. Le réseau interne utilise un des nombreux réseaux privés décrit dans la *RFC-1918*, dans ce cas, le réseau de classe C 192.168.0.0. La machine Linux ayant l'adresse TCP/IP 192.168.0.1 alors que les autres ont les adresses :

- Ordi A : 192.168.0.2
- Ordi B : 192.168.0.3
- Ordi C : 192.168.0.4

Les trois machines `A`, `B` et `C`, qui peuvent utiliser n'importe quel système d'exploitation, du moment qu'elles utilisent TCP/IP - comme par exemple **Windows 95**, **Macintosh MacTCP ou Open Transport** ou même une autre **machine Linux**, peuvent se connecter à n'importe qu'elle machine sur Internet. Toutefois `masq-gate` convertit toutes leurs connexions de façon à ce qu'elles semblent provenir de `masq-gate` elle même. MASQ s'arrange pour que toutes les données revenant d'Internet retournent au système qui en est à l'origine. Ainsi, les ordinateurs du réseau interne voient une route directe vers Internet et ne sont pas au courant du fait que leurs données ont été "masqueradées". Cela s'appelle une connexion transparente.

NOTE : Allez voir la 7 () pour plus de détails sur :

- La différence entre NAT, MASQ et serveurs Proxy.
- Comme le firewalling de paquets marche.

2.6 Ce qui est requis pour utiliser IP Masquerade sur un Linux 2.0.x

**** Veuillez s'il vous plaît consulter l'*IP Masquerade Resource* pour les dernières informations.****

- Un matériel décent. référez vous a la 7.1 () pour plus de détails
- Les sources d'un noyau de la série 2.0, disponibles sur *ftp ://ftp.lip6.fr/pub/linux/kernel/sources/v2.0/*
  (La majorité des distributions de linux récentes 7 () telle la RedHat 5.2 ont le un noyau modulaire avec toutes les options nécessaires a l'IP Masquerading compilées d'origine. Dans ce genre de cas, il n'y a pas besoin de recompiler le noyau. Si vous METTEZ À JOUR votre noyau, alors, vous devriez savoir ce dont que vous aurez besoin d'installer ou de mettre a jour, nous y reviendrons un peu plus loin).
- Les modules chargeables à la demande, de préférence la version 2.1.85 (ou ultérieure), disponible sur *ftp ://ftp.lip6.fr/pub/linux/kernel/sources/v2.1/modules-2.1.85.tar.gz* (modules-1.3.57 étant le minimum)
- Un réseau ou LAN TCP/IP fonctionnant convenablement (se référer à *Linux NET-3 HOWTO* `<http://www.freenix.org/unix/linux/HOWTO/NET-3-HOWTO.html>` (**en français**) et au *Network Administrator's Guide* `<ftp://ftp.ibp.fr/pub/linux/french/books/nag.french.eoit-1.0.tar.gz>`)
  Allez aussi faire un tour chez *Trinity OS Doc* `<http://www.ecst.csuchico.edu/~dranch/LINUX/TrinityOS.wri>`, une documentation très simple sur Linux et le réseau incluant des sujets comme IP MASQ, sécurité, DNS, DHCP, Sendmail, PPP, DialD, NFS, IPSEC-based VPNs, et des sections sur les performances pour n'en nommer que quelques unes. Plus de 50 sections en tout.
- Un accès Internet pour votre hôte Linux Se référer à *Linux ISP Hookup HOWTO* `<http://www.freenix.org/unix/linux/HOWTO/ISP-Hookup-HOWTO.html>` *Linux PPP HOWTO* `<http://www.freenix.org/unix/linux/HOWTO/PPP-HOWTO.html>`, *TrinityOS* `<http://www.ecst.csuchico.edu/~dranch/LINUX/TrinityOS.wri>`, *Linux DHCP mini-HOWTO* `<http://www.freenix.org/unix/linux/HOWTO/mini/DHCP.html>` et à *Linux Cable Modem mini-HOWTO* `<http://www.freenix.org/unix/linux/HOWTO/mini/Cable-Modem.html>`
- Ipfwadm 2.3, téléchargeable sur *ftp ://ftp.xos.nl/pub/linux/ipfwadm/ipfwadm-2.3.tar.gz*
  Plus d'informations sur *Linux IPFWADM page* `<http://www.xos.nl/linux/ipfwadm/>`
- Si vous etes interessés par l'utilisation d'IPCHAINS sur un noyau 2.0.36+, allez voir *Willy Tarreau's IPCHAINS enabler for 2.0.36* `<http://www-miaif.lip6.fr/willy/pub/linux-patches/>` ou *Rusty's IPCHAINS for 2.0.x kernels*
- Sachez configurer, compiler et installer un nouveau noyau Linux comme décrit dans *Linux Kernel HOWTO* `<http://www.freenix.org/unix/linux/HOWTO/Kernel-HOWTO.html>`
- Vous pouvez aussi appliquer divers paths optionnels pour ajouter des fonctionnalités comme :
  - Des redirecteurs de ports : Avec ces outils, vous pouvez faire marcher quelques programmes non MASQ derriere un serveur MASQ. De plus, vous pouvez configurer un serveur MASQ pour permettre aux utilisateurs d'internet de contacter des serveurs web, telnet, smtp, ftp (avec un patch), etc. Référez vous a la section 6.7 () de ce HOWTO pour plus d'informations. Voici une liste des différents patchs IP MASQ pour les noyaux 2.0 :
    - *IP PortForwarding (IPPORTFW)* de Steven Clarke's **RECOMMANDÉ**
    - *IP AutoForward* et *a mirror* `<ftp://ftp.netis.com/pub/members/rlynch/ipautofw.tar.gz>` (IPAUTOFW) - *NON recommandé*
    - *REDIR* `<http://ipmasq.cjb.net/redir_0.7.orig.tar.gz>` pour TCP (REDIR) - NON recommandé
    - *UDP redirector* (UDPRED) - NON recommandé
  - PORTFWed FTP :
    - Si vous allez faire de la redirection de port pour un serveur FTP interne, vous aurez besoin de *Fred Viles's FTP server patch*. Pour plus de détails, allez faire un tour du coté de la section 6.7 () de ce

HOWTO.
- X-Windows display forwarders :
  - *X-windows forwarding (DXCP)* <ftp://sunsite.unc.edu/pub/Linux/X11/compress/dxpc-3.7.0.tar.gz>
- ICQ MASQ module
  - *Andrew Deryabin's ICQ MASQ module*
- PPTP (GRE) et SWAN (IPSEC) VPNs redirecteur de tunnels :
  - *John Hardin's VPN Masquerade forwarders* ou le vieux patch pour juste le *PPTP Support* <http://ipmasq.cjb.net/ip_masq_pptp.patch.gz>.
- Patches spécifiques aux jeux :
  - Le patch Lamb's *LooseUDP for 2.0.36+* <ftp://ftp.netcom.com/pub/mu/mumford/loose-udp-2.0.36.patch.gz> de Glenn.
    Notez que certains navigateurs WWW décompresseront automatiquement ce fichier .gz. Pour télécharger ce fichier, gardez la touche SHIFT enfoncée lorsque vous cliquez sur l'URL ci dessus.
    Allez aussi faire un tour sur la *page sur le NAT* <http://www.alumni.caltech.edu/dank/peer-nat.html> de Dan Kegel pour plus d'informations. D'autres informations peuvent aussi être touvées dans la section 6.3.1 () et dans la 7 ().

Allez faire un tour sur *IP Masquerade Resource* <http://ipmasq.cjb.net/> pour plus de détails sur ces patchs.

2.7 Ce qui est requis pour utiliser IP Masquerade sur un Linux 2.2.x

**** Référez vous à *IP Masquerade Resource* <http://ipmasq.cjb.net/> pour les dernieres informations (en anglais). ****

- Les sources du noyau 2.2.x sont disponibles depuis *ftp ://ftp.lip6.fr/pub/linux/kernel/sources/v2.2/*
  NOTE : La majorité des distributions de linux récentes 7 () telle la RedHat 5.2 - livrée avec un noyau 2.0.36 - ne sont peut être pas prêtes pour les 2.2. Des outils comme le DHCP, NetUtils, auront a être mis a jour. Plus de détails sont donnés par la suite.
- Modules chargeables dynamiquement, de préférence avec un 2.1.121 ou plus récent. disponible depuis : *http ://www.pi.se/blox/modules/*
- Un réseau ou LAN TCP/IP fonctionnant convenablement (se référer à *Linux NET-3 HOWTO* <http://www.freenix.org/unix/linux/HOWTO/NET-3-HOWTO.html> (**en français**) et au *Network Administrator's Guide* <ftp://ftp.ibp.fr/pub/linux/french/books/nag.french.eoit-1.0.tar.gz>)
  Allez aussi faire un tour chez *Trinity OS Doc* <http://www.ecst.csuchico.edu/~dranch/LINUX/TrinityOS.wri>, une documentation très simple sur Linux et le réseau incluant des sujets comme IP MASQ, sécurité, DNS, DHCP, Sendmail, PPP, DialD, NFS, IPSEC-based VPNs, et des sections sur les performances pour n'en nommer que quelques unes. Plus de 50 sections en tout.
- Un accès Internet pour votre hôte Linux Se référer à *Linux ISP Hookup HOWTO* <http://www.freenix.org/unix/linux/HOWTO/ISP-Hookup-HOWTO.html> *Linux PPP HOWTO* <http://www.freenix.org/unix/linux/HOWTO/PPP-HOWTO.html>, *TrinityOS* <http://www.ecst.csuchico.edu/~dranch/LINUX/TrinityOS.wri>, *Linux DHCP mini-HOWTO* <http://www.freenix.org/unix/linux/HOWTO/mini/DHCP.html> et à *Linux Cable Modem mini-HOWTO* <http://www.freenix.org/unix/linux/HOWTO/mini/Cable-Modem.html>
- IP Chains 1.3.8 ou plus récent, disponible sur *http ://www.rustcorp.com/linux/ipchains/*
  D'informations additionnelles sur les différentes versions, visitez *Linux IP Firewalling Chains page* <http://www.rustcorp.com/linux/ipchains/>
- Sachez configurer, compiler et installer un nouveau noyau Linux comme décrit dans *Linux Kernel HOWTO* <http://www.freenix.org/unix/linux/HOWTO/Kernel-HOWTO.html>
- Vous pouvez aussi appliquer divers paths optionnels pour ajouter des fonctionnalités comme :
  - redirecteur de ports TCP/IP :

– *IP PortForwarding (IPMASQADM) - RECOMMANDÉ* `<http://juanjox.kernelnotes.org/>` ou son vieux *miroir*.

Module ICQ MASQ

– *Andrew Deryabin's ICQ MASQ module*

Allez faire un tour sur *IP Masquerade Resource* `<http://ipmasq.cjb.net/>` pour plus d'informations sur ces patches là et d'autres.

3 Mise en place d'IP Masquerade

Si votre réseau privé contient des informations vitales, pensez aux implications de sécurité avant d'utiliser IP Masquerade. Par défaut IP MASQ constitue une passerelle pour vous, pour atteindre Internet, mais la réciproque est vraie et quelqu'un sur Internet pourrait pénétrer sur votre réseau privé.

Une fois que vous avez un IP MASQ qui marche, il est CHAUDEMENT recommandé d'implémenter de FORTES règles d'IPFWADM/IPCHAINS. Référez vous aux sections 6.4 () et 6.5 () ci dessous pour plus de détails.

3.1 Compiler le noyau pour le support d'IP Masquerade

Si votre distribution de Linux a déjà les fonctionnalités necessaires et les modules de compilés, comme :

– IPFWADM/IPCHAINS
– IP forwarding
– IP masquerading
– IP Firewalling
– etc.

Et tous les modules relatif au MASQ de compilés (la majorité des noyaux auront ce dont vous avez besoin) ; alors, vous n'aurez pas a recompiler votre noyau. Si vous n'etes pas sru que votre distribution linux est prete pour le MASQ allez voir la section 7 () ou la page *IP Masquerade Resource* `<http://ipmasq.cjb.net/>` pour plus de détails. Si vous n'arrivez pas a savoir si votre distribution supporte le MASQ, alors, SUPPOSEZ QU'ELLE NE LE SUPPORTE PAS

Que votre distribution le supporte ou pas, lire la suite de cette section est chaudement recommandé car elle contient pas mal d'informations interessantes.

3.1.1 Noyaux Linux 2.0.x

Référez vous a la section 2.5 () pour les logiciels, patchs requis.

- Vous devez tout d'abord disposer des sources du noyau (de préférence la derniere version 2.0.36 ou plus récente).
- Si c'est la première fois que vous compilez votre noyau, ne soyez pas effrayé. En fait, c'est plutôt simple, et tout est expliqué dans plusieurs liens dans la section 2.5 ().
- Décompressez les sources du noyau dans `/usr/src/` avec la commande `tar xvzf linux-2.0.x.tar.gz -C /usr/src`, où x est le numéro de révision du noyau. Une fois terminé, vérifiez qu'il y a un répertoire ou un lien symbolique nommé `/usr/src/linux/`.

– Appliquez les patches appropriés et/ou optionnels au sources du noyau. Depuis la version 2.0.36, IP MASQ ne requiert plus de patches spécifiques pour que tout marche. Les fonctionnalités comme IPPORTFW, PPTP, and Redirection Xwindow sont optionnelles. Référez vous à la section 2.5 () pour des liens et a la page *IP Masquerade Resources* `<http://ipmasq.cjb.net/>` pour des informations a jours ainsi que les adresses des patchs additionnels.
– Il y a ici le MINIMUM d'options a etre incluses dans le noyau. Vous aurez aussi a configurer le noyau pour vos interfaces réseau. Consulter le *Linux Kernel HOWTO* `<http://www.freenix.org/unix/linux/HOWTO/Kernel-HOWTO.html>` et le fichier README dans le répertoire des sources du noyau pour plus d'informations sur la compilation d'un noyau.
– Notez les ***RÉPONSES YES ou NO*** données aux options suivantes. Toutes les options citées ici ne seront pas disponibles avec un noyau standard, certaines sont issues de l'application d'un patch :

```
* Prompt for development and/or incomplete code/drivers (CONFIG_EXPERIMENTAL) [Y/n/?]
  - YES : Permet de selectionner l'IP Masquerade (qui est experimental dans les 2.0)

* Enable loadable module support (CONFIG_MODULES) [Y/n/?]
  - YES : Permet de charger les modules IP MASQ

* Networking support (CONFIG_NET) [Y/n/?]
  - YES : Permet l'utilisation du réseau

* Network firewalls (CONFIG_FIREWALL) [Y/n/?]
  - YES : Permet d'utiliser IPFWADM

* TCP/IP networking (CONFIG_INET)
  - YES : Permet d'utiliser le protocole TCP/IP

* IP: forwarding/gatewaying (CONFIG_IP_FORWARD)
  - YES : Permet de faire du relayage / routage de paquets IP - Contrôlé par IPFWADM

* IP: syn cookies (CONFIG_SYN_COOKIES) [Y/n/?]
  - YES : CHAUDEMENT recommandé pour une sécurité réseau de base

* IP: firewalling (CONFIG_IP_FIREWALL) [Y/n/?]
  - YES : Permet d'utiliser les fonctionnalités de firewall

* IP: firewall packet logging (CONFIG_IP_FIREWALL_VERBOSE) [Y/n/?]
  - YES : (OPTIONNEL mais CHAUDEMENT recommandé) : Permet de logger des hits du firewall

* IP: masquerading (CONFIG_IP_MASQUERADE [Y/n/?]
  - YES : Permet a IP MASQ de réadresser les paquets TCP/IP

* IP: ipautofw masquerade support (EXPERIMENTAL) (CONFIG_IP_MASQUERADE_IPAUTOFW) [Y/n/?]
  - NO : IPAUTOFW est une méthode simple de relayage de port. Bien qu'elle marche,
         IPPORTFW marche mieux, par conséquent, IPAUTOFW N'EST PAS recommandé

* IP: ipportfw masq support (EXPERIMENTAL) (CONFIG_IP_MASQUERADE_IPPORTFW) [Y/n/?]
  - YES : Cette option n'est disponible qu'après l'application d'un patch pour les
          noyaux 2.0

          Avec cette option, des ordinateurs externes sur internet peuvent communiquer
          directement avec des machines MASQuée spécifiques. Cette fonctionnalité est
          typiquement utilisée pour accéder a des serveurs smtp, telnet, web
          internes. Pour le ftp, un patch supplémentaire sera nécessaire comme il l'est
```

```
              décrit dans la FAQ. De plus amples informations sur le relayage de port se
              trouve dans la section forwarding de ce HOWTO.

    * IP: ICMP masquerading (CONFIG_IP_MASQUERADE_ICMP) [Y/n/?]
      - YES : Permet de masquerader les paquets ICMP. Bien qu'optionnelle, beaucoup de
              programmes ne fonctionneront pas correctement sans cette fonctionnalité.

    * IP: loose UDP port managing (EXPERIMENTAL) (CONFIG_IP_MASQ_LOOSE_UDP) [Y/n/?]
      - YES : Cette option n'est disponible qu'après l'application d'un patch pour les
              noyaux 2.0

              Avec cette option, des ordinateurs internes masqués peuvent jouer a des jeux
              ne supportant pas le NAT sur internet. plus de détails sont donnés dans la FAQ

    * IP: always defragment (CONFIG_IP_ALWAYS_DEFRAG) [Y/n/?]
      - YES : Cette fonctionnalité optimise les connexions IP MASQ - CHAUDEMENT recommandé

    * IP: optimize as router not host (CONFIG_IP_ROUTER) [Y/n/?]
      - YES : Cela optimise le noyau pour l'utilisation en réseau

    * IP: Drop source routed frames (CONFIG_IP_NOSR) [Y/n/?]
      - YES : CHAUDEMENT recommandé pour une sécurité réseau de base

    * Dummy net driver support (CONFIG_DUMMY) [M/n/y/?]
      - YES : Bien qu'optionnelle, cette fonctionnalité peut aider en cas de problèmes

    * /proc filesystem support (CONFIG_PROC_FS) [Y/n/?]
      - YES : Requise pour permettre d'autoriser le systeme de relayage
```

NOTE : Ce sont juste les composants dont vous avez besoin pour l'IP Masquerade. Vous avez aussi à ajouter toute autre option réseau et matérielles nécessaire pour votre configuration personnelle.

– Une fois le noyau compilé, vous avez aussi à compiler et installer les modules :

```
make modules; make modules_install
```

– Ensuite, ajoutez quelques lignes dans votre `/etc/rc.d/rc.local` pour charger le script de configuration de l'IP MASQ automatiquement a chaque démarrage :

```
.
.
.
# script rc.firewall - Démarre l'IP MASQ et le firewall
/etc/rc.d/rc.firewall
.
.
.
```

3.1.2 Noyaus Linux 2.2.x

Référez vous a la section 2.6 () pour les logiciels, patchs requis.

– Tout d'abord, Vous avez besoin des sources du noyau 2.2.x. De préférence, une version ultérieure au 2.2.1.
– Si c'est la première fois que vous compilez votre noyau, ne soyez pas effrayé. En fait, c'est plutôt simple, et tout est expliqué dans plusieurs liens dans la section 2.6 ().

- Décompressez les sources du noyau dans /usr/src/ avec la commande tar xvzf linux-2.2.x.tar.gz -C /usr/src, où x est le numéro de révision du noyau. Une fois terminé, vérifiez qu'il y a un répertoire ou un lien symbolique nommé /usr/src/linux/.
- Appliquez les patches appropriés et/ou optionnels au sources du noyau. Depuis la version 2.2.1, IP MASQ ne requiert plus de patches spécifiques pour que tout marche. Les fonctionnalités comme PPTP, and Redirection Xwindow sont optionnelles. Référez vous à la section 2.6 () pour des liens et a la page *IP Masquerade Resources* <http://ipmasq.cjb.net/> pour des informations a jours ainsi que les adresses des patchs additionnels.
- Il y a ici le MINIMUM d'options a etre incluses dans le noyau. Vous aurez aussi a configurer le noyau pour vos interfaces réseau. Consulter le *Linux Kernel HOWTO* <http://www.freenix.org/unix/linux/HOWTO/Kernel-HOWTO.html> et le fichier README dans le répertoire des sources du noyau pour plus d'informations sur la compilation d'un noyau.
- Notez les ***RÉPONSES YES ou NO*** données aux options suivantes. Toutes les options citées ici ne seront pas disponibles avec un noyau standard, certaines sont issues de l'application d'un patch :

```
* Prompt for development and/or incomplete code/drivers (CONFIG_EXPERIMENTAL) [Y/n/?]
  - YES : Bien que n'étant pas nécessaire pour l'IP MASQ, cette option permet de créer
          certains modules MASQ et de choisir quelques options comme le relayage de
          ports

* Enable loadable module support (CONFIG_MODULES) [Y/n/?]
  - YES : Permet de charger les modules IP MASQ

* Networking support (CONFIG_NET) [Y/n/?]
  - YES : Permet l'utilisation du réseau

* Packet socket (CONFIG_PACKET) [Y/m/n/?]
  - YES : Bien qu'OPTIONNELLE, cette fonctionnalité permet d'utiliser tcpdump pour
          débugger certains problèmes avec IP MASQ

* Kernel/User netlink socket (CONFIG_NETLINK) [Y/n/?]
  - YES : Bien qu'OPTIONNELLE, cette fonctionnalité permet de logger des hits du
          firewall

* Routing messages (CONFIG_RTNETLINK) [Y/n/?]
  - NO : Cette option n'a rien a voir avec les logs du firewall

* Network firewalls (CONFIG_FIREWALL) [Y/n/?]
  - YES : Permet d'utiliser IPCHAINS

* TCP/IP networking (CONFIG_INET) [Y/n/?]
  - YES : Permet d'utiliser le protocole TCP/IP

* IP: advanced router (CONFIG_IP_ADVANCED_ROUTER) [Y/n/?]
  - NO : Ce n'est requis que si vous utilisez aussi CONFIG_IP_ROUTE_VERBOSE et des
         routages spéciaux (indépendants d'ipchains/masq)

* IP: verbose route monitoring (CONFIG_IP_ROUTE_VERBOSE) [Y/n/?]
  - YES : Très utile si vous voulez utiliser le code de routage pour mettre de coté les
          paquets spoofés (CHAUDEMENT recommandé) et que vous voulez les logger.

* IP: firewalling (CONFIG_IP_FIREWALL) [Y/n/?]
  - YES : Permet d'utiliser les fonctionnalités de firewall
```

```
* IP: firewall packet netlink device (CONFIG_IP_FIREWALL_NETLINK) [Y/n/?]
  - YES : Bien qu'OPTIONNELLE, cette fonctionnalité permet d'améliorer les logs du
          firewall

* IP: always defragment (required for masquerading) (CONFIG_IP_ALWAYS_DEFRAG) [Y/n/?]
  - YES : Cette fonctionnalité est REQUISE si vous voulez avoir accès aux
          fonctionnalités d'IP MASQ et/ou de proxy transparent. Cette fonctionnalité
          optimise les connexions IP MASQ

* IP: masquerading (CONFIG_IP_MASQUERADE) [Y/n/?]
  - YES : Permet a IP MASQ de réadresser les paquets TCP/IP

* IP: ICMP masquerading (CONFIG_IP_MASQUERADE_ICMP) [Y/n/?]
  - YES : Permet de masquerader les paquets icmp (les paquets d'erreur seront masqués de
          que la réponse soit oui ou non). Cette fonctionnalité est importante si vous
          rencontrez des problèmes avec vos connexions.

* IP: masquerading special modules support (CONFIG_IP_MASQUERADE_MOD) [Y/n/?]
  - YES : Bien qu'OPTIONNELLE, cette fonctionnalité permet de choisir le relayage de
          port TCP/IP pour permettre a des ordinateurs externe d'accéder a certaines
          machines MASQuées.

* IP: ipautofw masq support (EXPERIMENTAL) (CONFIG_IP_MASQUERADE_IPAUTOFW) [N/y/m/?]
  - NO : IPAUTOFW est une méthode simple pour le relayage de ports. C'est juste une
         grosse bidouille, et il est préférable d'utiliser les modules spécifiques a un
         protocole. IPAUTOFW N'EST PAS recommandé.

* IP: ipportfw masq support (EXPERIMENTAL) (CONFIG_IP_MASQUERADE_IPPORTFW) [Y/m/n/?]
  - YES : Permet le IPPORTFW.

          Avec cette option, des ordinateurs externes sur internet peuvent communiquer
          directement avec des machines MASQuée spécifiques. Cette fonctionnalité est
          typiquement utilisée pour accéder a des serveurs smtp, telnet, web
          internes. Pour le ftp, un patch supplémentaire sera nécessaire comme il l'est
          décrit dans la FAQ. De plus amples informations sur le relayage de port se
          trouve dans la section forwarding de ce HOWTO.

* IP: ip fwmark masq-forwarding support (EXPERIMENTAL) (CONFIG_IP_MASQUERADE_MFW) [Y/m/n/?]
  - NO : Permet de configurer le relayage avec IPCHAINS directement. C'est du code
        expérimental, et la méthode recommandée est d'utiliser IPMASQADM et IPPORTFW.

* IP: optimize as router not host (CONFIG_IP_ROUTER) [Y/n/?]
  - YES : Cela optimise le noyau pour l'utilisation en réseau mais cela ne change pas
          grand chose en fait.

* IP: GRE tunnels over IP (CONFIG_NET_IPGRE) [N/y/m/?]
  - NO : Cette sélection OPTIONNELLE sert a autoriser les tunnels PPTP et GRE via la
         machine MASQ

* IP: TCP syncookie support (not enabled per default) (CONFIG_SYN_COOKIES) [Y/n/?]
  YES : CHAUDEMENT recommandé pour une sécurité réseau de base

* IP: Drop source routed frames (CONFIG_IP_NOSR) [Y/n/?]
  - YES : CHAUDEMENT recommandé pour une sécurité réseau de base
```

```
    * Network device support (CONFIG_NETDEVICES) [Y/n/?]
      - YES : Permet d'utiliser la sous couche réseau de Linux

    * Dummy net driver support (CONFIG_DUMMY) [M/n/y/?]
      - YES : Bien qu'optionnelle, cette fonctionnalité peut aider en cas de problèmes

    * /proc filesystem support (CONFIG_PROC_FS) [Y/n/?]
      - YES : Requise pour permettre d'autoriser le systeme de relayage
```

NOTE : Voici juste les composants qu'il faut pour que l'IP Masquerade marche, sélectionnez les options spécifiques dont vous avez besoin pour votre système.

– Après avoir compilé le noyau, vous devriez compiler et installer les modules :

```
make modules; make modules_install
```

– Ensuite, ajoutez quelques lignes dans votre `/etc/rc.d/rc.local` pour charger le script de configuration de l'IP MASQ automatiquement a chaque démarrage :

```
.
.
.
# script rc.firewall - Démarre l'IP MASQ et le firewall
/etc/rc.d/rc.firewall
.
.
.
```

3.2 Assignation d'adresse IP pour le réseau local

Puisque toutes les machines **INTERNES MASQUÉES** ne doivent pas d'adresse IP officielle, il doit y avoir une manière spécifique et générale de leur en allouer sans entrer en collisions avec l'adresse IP de quelqu'un d'autre.

Selon la FAQ originelle d'IP Masquerade :

Le document officiel qui indique quelles adresses IP assigner à un réseau non connecté ou "privé" est la *RFC 1918*. Il existe 3 plages réservées spécialement à cet effet.

```
Section 3~: Adressage de réseaux privés

L' "Internet Assigned Numbers Authority" (IANA) a réservé les 3 plages
suivantes pour leur utilisation par des réseaux privés~:

              10.0.0.0        -   10.255.255.255
              172.16.0.0      -   172.31.255.255
              192.168.0.0     -   192.168.255.255

Nous ferons référence à la première en tant que la "plage de 24 bits", la
deuxième comme "plage de 20 bits" et la troisième comme "plage de 16 bits".
Notez que la première plage n'est rien d'autre qu'un réseau de classe A, la
deuxième un ensemble de 16 réseaux de classe B contigus, et la troisième un
ensemble de 255 réseaux de classe C contigus.
```

Pour la petite histoire, ma préférence va au réseau 192.168.0.0 avec le masque de sous réseau de classe C 255.255.255.0, et ce HOWYO reflete cela. Bien sur, n'importe quel réseau privé décrit ci dessus est valide, tant que vous utilisez le bon masque de sous réseau.

Ainsi, si vous utilisez un réseau de classe C, vous devrez utiliser les adresses IP 192.168.0.1, 1.92.168.0.2, 1.92.168.0.3, ..., 192.168.0.x

192.168.0.1 est habituellement la machine passerelle, qui est ici votre machine Linux se connectant à Internet. Remarquez que 192.168.0.0 et 192.168.0.255 sont respectivement les adresses de réseau et de broadcast (ces adresses sont RÉSERVÉES). Évitez d'utiliser ces adresses sur vos machines ou votre réseau ne fonctionnera pas correctement.

3.3 Configurer les règles d'IP Forwarding

A ce point du document, vous devriez avoir votre noyau et les autres packages installés. De plus, les adresses IP, la passerelle, et le DNS devraient être installés sur votre serveur MASQ. Si vous ne savez pas comment configurer votre carte réseau, référez vous aux sections 2.5 () ou 2.6 ()

Maintenant, la seule chose à faire est de configurer l'outil de firewalling IP (ipfwadm) pour faire suivre / masquer les paquets appropriés à la machine qui convient :

> ** Ceci peut être fait de diverses façons. Les suggestions et exemples suivants fonctionnent pour moi, mais il se peut que vous ayez d'autres idées ou besoins.
>
> **** Cette section ne fournis que les rudiments d'un firewall pour faire marcher l'IP MASQ. Une fois que l'IP MASQ a été suffisamment testé (comme décrit plus loin dans ce HOWTO), référez vous aux section 6.4 () et/ou 6.5 () pour des regles plus sécurisées. En plus, allez lire les mages man d'IPFWADM (2.0) te IPCHAINS (2.2) pour plus de détails.**

3.3.1 Noyaux Linux 2.0.x

Créez le fichier `/etc/rc.d/rc.firewall` avec l'exemple SIMPLE qui suit :

```
#!/bin/sh
#
# rc.firewall - exemple SIMPLE de configuration IP MASQ pour les noyaux 2.0 avec IPFWADM
#
# Charge tous les modules MASQ requis
#
# NOTE : Ne chargez que les modules dont vous avez besoin. Tous les modules IP MASQ
#        disponibles sont cités ci dessous, mais sont mis en commentaires.

# Necessaire pour charger les modules
#
/sbin/depmod -a

# Supporte le masquerading des transferts de fichiers FTP utilisant la méthode PORT.
#
/sbin/modprobe ip_masq_ftp

# Supporte le masquerading de RealAudio via UDP. Sans ce module, RealAudio FONCTIONNERA,
# mais en mode TCP. Cela peut causer une réduction dans la qualité du son.
#
#/sbin/modprobe ip_masq_raudio

# Supporte le masquerading des transferts IRC DCC.
#
```

```
#/sbin/modprobe ip_masq_irc

# Supporte le masquerading de Quake et QuakeWorld par défaut. Ce module sert si vous avez
# plusieurs utilisateurs derriere le serveur MASQ. Si vous allez jouer a Quake I, II et
# III, utilisez le deuxieme exemple.
#
#Quake I / QuakeWorld (ports 26000 et 27000)
#/sbin/modprobe ip_masq_quake
#
#Quake I/II/III / QuakeWorld (ports 26000, 27000, 27910, 27960)
#/sbin/modprobe ip_masq_quake ports=26000,27000,27910,27960

# Supporte le masquerading du systeme de visioconférences CuSeeMe
#
#/sbin/modprobe ip_masq_cuseeme

# Supporte le masquerading du systeme de visioconférences VDO Live
#
#/sbin/modprobe ip_masq_vdolive


# CRITIQUE : Permet le relayage de paquets IP car il est désactivé par défaut depuis le
#            noyau 2.0.34
#
# Utilisateurs RedHat : Vous pouvez aussi changer l'option dans /etc/sysconfig/network de :
#
#                       FORWARD_IPV4=false
#                             à
#                       FORWARD_IPV4=true
#
echo "1" > /proc/sys/net/ipv4/ip_forward

# Utilisateurs d'IP dynamiques :
#
# Si vous avez une adresse IP dynamique via SLIP, PPP ou DHCP, décommentez l'option
# suivante. Elle permet d'utiliser une bidouille dans l'IP MASQ qui sert pour les IP
# dynamiques, ce qui rends la vie avec DialD, PPPd et les programmes similaires bien plus
# facile.
#
#echo "1" > /proc/sys/net/ipv4/ip_dynaddr


# Durée de vie des connexion MASQuées
#
#    2 hrs pour une session TCP
#   10 sec apres qu'un paquet "FIN" soit passé
#  160 sec pour le traffic UDP (Important pour les utilisateurs d'ICQ)
#
/sbin/ipfwadm -M -s 7200 10 160


# DHCP : Pour Ceux qui recoivent leur IP externe d'un serveur soit DHCP soit BOOTP comme
#        les utilisateurs d'ADSL ou du Câble, Il est nécessaire d'utiliser la commande
#        suivante avant le deny. Le "bootp_client_net_if_name" doit être remplacé par le
#        nom de l'interface qui recevra son IP du serveur DHCP/BOOTP. Cela devrais être
```

```
#        quelque chose comme "eth0", "eth1", etc.
#
#        Cet exemple est pour l'instant commenté.
#
#
#/sbin/ipfwadm -I -a accept -S 0/0 67 -D 0/0 68 -W bootp_clients_net_if_name -P udp


# Permettre un forwarding et masquerading IP simple
#
# NOTE : L'exemple suivant est un exemple pour un réseau dont les adresses sont dans le
#        réseau 192.168.0.0 avec le masque de sous réseau 255255.255.0, ou 24 bits.
#
#        Changez ces paramètres pour prendre en compte votre configuration locale.
#
/sbin/ipfwadm -F -p deny
/sbin/ipfwadm -F -a m -S 192.168.0.0/24 -D 0.0.0.0/0
```

Une fois que vous avez terminé d'éditer votre `/etc/rc.d/rc.firewall`, rendez le executable en tapant "`chmod 700 /etc/rc.d/rc.firewall`".

Vous pouvez aussi autoriser l'IP MASQ par machine et non en autorisant tout le réseau. Par exemple, vous pourriez vouloir que seules les machines 192.168.0.2 et 192.168.0.8 aient accès a internet, et pas les autres machines. Il suffit de changer la section "Permettre un forwarding et masquerading IP simple" (ci dessus) du fichier `/etc/rc.d/rc.firewall`.

```
# Permettre un forwarding et masquerading IP simple
#
# NOTE : L'exemple suivant est un exemple qui autorise le MASQ pour uniquement les
#        machines 192.168.0.2 et 192.168.0.6.
#
#        Utilisez l'exemple ci dessous pour spécifier les machines ayant le droit
#        d'accéder a internet. n'oubliez pas de l'adapter a votre configuration.
#
/sbin/ipfwadm -F -p deny
/sbin/ipfwadm -F -a m -S 192.168.0.2/32 -D 0.0.0.0/0
/sbin/ipfwadm -F -a m -S 192.168.0.8/32 -D 0.0.0.0/0
```

Il semble que l'erreur la plus courante soit de faire de la police par défaut le masquerading en tapant :

`ipfwadm -F -p masquerade`

NE FAITE PAS de votre configuration par défaut le MASQ. Autrement, quelqu'un de mal intentionné pourrais manipuler ses tables de routage pour créer un tunnel a travers votre passerelle, utilisant le masquerading pour masquer sa propre identité !

Encore une fois, vous pouvez ajouter au fichier `/etc/rc.d/rc.firewall`, ou l'un des autres fichiers rc, ou même les taper a la main a chaque fois que vous en avez besoin.

Allez faire un tour dans les sections 6.4 () et 6.5 () pour plus de détails sur l'établissement de règles plus strictes.

3.3.2 Noyaux Linux 2.2.x

Notez que **IPFWADM n'est plus l'outil de firewall a utiliser** pour manipuler les regles d'IP MASQ pour les noyaux 2.1 et 2.2. Ces noyaux utilisent le nouvel outil IPCHAINS. Pour plus de détails sur les raisons de ce changement, référez vous a la 7 ().

Créez le fichier `/etc/rc.d/rc.firewall` avec les règles suivantes :

```
#!/bin/sh
#
# rc.firewall - exemple SIMPLE de configuration IP MASQ pour les noyaux 2.1 et 2.2 avec
#               IPFWADM
#
# Charge tous les modules MASQ requis
#
# NOTE : Ne chargez que les modules dont vous avez besoin. Tous les modules IP MASQ
#        disponibles sont cités ci dessous, mais sont mis en commentaires.

# Necessaire pour charger les modules
#
/sbin/depmod -a

# Supporte le masquerading des transferts de fichiers FTP utilisant la méthode PORT.
#
/sbin/modprobe ip_masq_ftp

# Supporte le masquerading de RealAudio via UDP. Sans ce module, RealAudio FONCTIONNERA,
# mais en mode TCP. Cela peut causer une réduction dans la qualité du son.
#
#/sbin/modprobe ip_masq_raudio

# Supporte le masquerading des transferts IRC DCC.
#
#/sbin/modprobe ip_masq_irc


# Supporte le masquerading de Quake et QuakeWorld par défaut. Ce module sert si vous avez
# plusieurs utilisateurs derriere le serveur MASQ. Si vous allez jouer a Quake I, II et
# III, utilisez le deuxieme exemple.
#
#Quake I / QuakeWorld (ports 26000 et 27000)
#/sbin/modprobe ip_masq_quake
#
#Quake I/II/III / QuakeWorld (ports 26000, 27000, 27910, 27960)
#/sbin/modprobe ip_masq_quake ports=26000,27000,27910,27960

# Supporte le masquerading du systeme de visioconférences CuSeeMe
#
#/sbin/modprobe ip_masq_cuseeme

# Supporte le masquerading du systeme de visioconférences VDO Live
#
#/sbin/modprobe ip_masq_vdolive


# CRITIQUE : Permet le relayage de paquets IP car il est désactivé par défaut depuis le
#            noyau 2.0.34
#
# Utilisateurs RedHat : Vous pouvez aussi changer l'option dans /etc/sysconfig/network de :
#
#                       FORWARD_IPV4=false
```

```
#                             à
#                       FORWARD_IPV4=true
#
echo "1" > /proc/sys/net/ipv4/ip_forward


# Utilisateurs d'IP dynamiques :
#
# Si vous avez une adresse IP dynamique via SLIP, PPP ou DHCP, décommentez l'option
# suivante. Elle permet d'utiliser une bidouille dans l'IP MASQ qui sert pour les IP
# dynamiques, ce qui rends la vie avec DialD, PPPd et les programmes similaires bien plus
# facile.
#
#echo "1" > /proc/sys/net/ipv4/ip_dynaddr


# Durée de vie des connexion MASQuées
#
#    2 hrs pour une session TCP
#   10 sec apres qu'un paquet "FIN" soit passé
#  160 sec pour le traffic UDP (Important pour les utilisateurs d'ICQ)
#
/sbin/ipchains -M -S 7200 10 160


# DHCP : Pour Ceux qui recoivent leur IP externe d'un serveur soit DHCP soit BOOTP comme
#        les utilisateurs d'ADSL ou du Câble, Il est nécessaire d'utiliser la commande
#        suivante avant le deny. Le "bootp_client_net_if_name" doit être remplacé par le
#        nom de l'interface qui recevra son IP du serveur DHCP/BOOTP. Cela devrais être
#        quelque chose comme "eth0", "eth1", etc.
#
#        Cet exemple est pour l'instant commenté.
#
#/sbin/ipchains -A input -j ACCEPT -i bootp_clients_net_if_name -s 0/0 67 -d 0/0 68 -p udp

# Permettre un forwarding et masquerading IP simple
#
# NOTE : L'exemple suivant est un exemple pour un réseau dont les adresses sont dans le
#        réseau 192.168.0.0 avec le masque de sous réseau 255255.255.0, ou 24 bits.
#
#        Changez ces paramètres pour prendre en compte votre configuration locale.
#
/sbin/ipchains -P forward DENY
/sbin/ipchains -A forward -s 192.168.0.0/24 -j MASQ
```

Une fois que vous avez terminé d'éditer votre `/etc/rc.d/rc.firewall`, rendez le executable en tapant `"chmod 700 /etc/rc.d/rc.firewall"`.

Vous pouvez aussi autoriser l'IP MASQ par machine et non en autorisant tout le réseau. Par exemple, vous pourriez vouloir que seules les machines 192.168.0.2 et 192.168.0.8 aient accès a internet, et pas les autres machines. Il suffit de changer la section "Permettre un forwarding et masquerading IP simple" (ci dessus) du fichier `/etc/rc.d/rc.firewall`.

```
# Permettre un forwarding et masquerading IP simple
#
# NOTE : L'exemple suivant est un exemple qui autorise le MASQ pour uniquement les
```

```
#       machines 192.168.0.2 et 192.168.0.6.
#
#       Utilisez l'exemple ci dessous pour spécifier les machines ayant le droit
#       d'accéder a internet. n'oubliez pas de l'adapter a votre configuration.
#
/sbin/ipchains -P forward deny
/sbin/ipchains -A forward -s 192.168.0.2/32 -j MASQ
/sbin/ipchains -A forward -s 192.168.0.8/32 -j MASQ
```

Il semble que l'erreur la plus courante soit de faire de la police par défaut le masquerading en tapant :

```
/sbin/ipchains -P forward masquerade
```

NE FAITE PAS de votre configuration par défaut le MASQ. Autrement, quelqu'un de mal intentionné pourrais manipuler ses tables de routage pour créer un tunnel a travers votre passerelle, utilisant le masquerading pour masquer sa propre identité !

Encore une fois, vous pouvez ajouter au fichier `/etc/rc.d/rc.firewall`, ou l'un des autres fichiers rc, ou même les taper a la main a chaque fois que vous en avez besoin.

Allez faire un tour dans les sections 6.4 () et 6.5 () pour plus de détails sur l'établissement de règles plus strictes, référez vous aussi au *Linux IP CHAINS HOWTO*

4 Configurer les autres machines MASQuées internes

En plus d'affecter les adresses IP pour chaque machine MASQuée, vous devrez indiquer la bonne adresse de la passerelle et le bon serveur DNS. En général, c'est plutôt simple. Vous entrez juste l'adresse de votre machine Linux (généralement 192.168.0.1) en tant qu'adresse de passerelle.

Pour le DNS, vous pouvez utiliser n'importe quel DNS qui soit utilisable. Le plus simple est d'utiliser celui qu'utilise votre serveur Linux. Vous pouvez aussi, si vous le désirez, ajouter des "suffixes d'ordre de recherche" DNS.

Après avoir correctement configurées ces adresses IP, n'oubliez pas de relancer leurs services réseaux, ou de les rebooter.

Les instructions de configuration qui suivent supposent que vous utilisez un réseau de classe C, et que votre machine Linux a pour adresse 192.168.0.1. Notez que 192.168.0.0 et 192.168.0.255 sont des adresses TCP/IP réservées.

Au jour d'aujourd'hui, les plates formes suivantes ont été testés comme machine MASQuées.

- Linux 1.2.x, 1.3.x, 2.0.x, 2.1.x, 2.2.x
- Solaris 2.51, 2.6, 7
- Windows 95, OSR2, 98
- Windows NT 3.51, 4.0, 2000 (workstation et server)
- Windows For Workgroup 3.11 (avec le paquetage TCP/IP)
- Windows 3.1 (avec ke paquetage Netmanage Chameleon)
- Novell 4.01 Server avec le service TCP/IP
- OS/2 (Warp v3 inclu)
- Macintosh OS (avec MacTCP ou Open Transport)
- DOS (avec le paquetage NCSA Telnet, DOS Trumpet fonctionne partiellement)
- Amiga (avec AmiTCP ou AS225-stack)
- Les VAX 3520 et 3100 avec UCX (la pile TCP/IP pour VMS)
- Alpha/AXP avec Linux RedHat

– SCO Openserver (v3.2.4.2 et 5)
– IBM RS/6000 avec AIX

4.1 Configurer Microsoft Windows 95

1. Si vous n’avez pas installé votre carte réseau et son driver, faites le maintenant. La description de cela dépasse le cadre de ce document.
2. Allez dans le *’Panneau de configuration’* –> *’Réseau’*.
3. Cliquez sur *’Ajouter’* –> *’Protocole’* –> *’Constructeur : Microsoft’* –> *’Protocole : TCP/IP’* si vous ne l’avez pas déjà.
4. Sélectionnez le TCP/IP lié a votre carte réseau et cliquez sur *’Propriétés’*. Allez maintenant dans l’onglet *’adresse IP’* et mettez comme adresse IP 192.168.0.x ($1 < x < 255$). Fixez le *’Masque de sous réseau’* à 255.255.255.0
5. Maintenant, allez dans l’onglet *’Passerelle’*, Entrez 192.168.0.1 comme passerelle et cliquez sur *’Ajouter’*.
6. Dans l’onglet *’Configuration DNS’*, assurez vous de bien avoir mis un nom pour cette machine et entré votre nom de domaine officiel. Si vous n’avez pas de nom de domaine, mettez celui de votre FAI. Maintenant, ajoutez tous les serveurs DNS qui sont configurés sur votre machine Linux (on les trouve dans le fichier `/etc/resolv.conf`). D’habitude, ces serveurs DNS sont situés chez votre FAI, mais il se peut que vous en aviez un en local pour faire soit votre propre cache ou serveur DNS autoritaire. Vous pouvez aussi ajouter quelques domaines de recherche.
7. Laissez les autres paramètres tels quels, à moins que vous sachiez ce que vous faites.
8. Cliquez sur *’OK’* dans toutes les boîtes de dialogue et relancez le système.
9. `Ping`uez la machine Linux pour tester la connexion réseau : *’Démarrer’* –> *’Exécuter’*, tapez : `ping 192.168.0.1` (C’est seulement un test de connexion locale, vous ne pouvez pas encore `ping`uer l’extérieur). Si vous ne voyez pas de réponses a vos pings, vérifiez votre configuration réseau.
10. Vous pouvez éventuellement créer un fichier `HOSTS` dans le répertoire `c :\windows`, pour que vous puissiez utiliser les noms d’hôtes des autres machines de votre réseau local sans recourir a un serveur dns. Il y a un exemple nommé `HOSTS.SAM` dans le répertoire `c :\windows`.

4.2 Configurer Windows NT

1. Si vous n’avez pas encore installé votre carte réseau et son driver, faites le maintenant. La description de cela dépasse le cadre de ce document.
2. Allez dans le *’Panneau de configuration’* –> *’Réseau’*.
3. Ajoutez le protocole TCP/IP et les composants qui s’y rattachent depuis le menu *’Ajout de logiciels’* si vous n’avez pas encore installé le service TCP/IP.
4. Dans la section *’Logiciel et carte réseau’*, sélectionnez le *’Protocole TCP/IP’* dans la boite de choix *’Logiciels réseaux installés’*.
5. Dans *Configuration TCP/IP*, sélectionnez l’adaptateur réseau appropriées, par exemple `[1]Novell NE2000 Adapter`. Entrez l’adresse IP 192.168.0.x ($1 < x < 255$), positionnez le masque de sous réseau sur 255.255.255.0 et la passerelle par défaut à 192.168.0.1.
6. Ne sélectionnez pas *’Configuration automatique DHCP’*, mettez n’importe quoi dans la case *’Serveur WINS’*, et *’Autorisez le relayage de paquets’* à moins que vous ne fassiez partie d’un domaine Windows NT et que vous sachiez EXACTEMENT ce que vous faites.
7. Cliquez sur *’DNS’*, et remplissez les informations appropriés, maintenant, ajoutez tous les serveurs DNS qui sont configurés sur votre machine Linux (on les trouve dans le fichier `/etc/resolv.conf`). Cliquez sur *’OK’* une fois que c’est fini.

8. Cliquez sur *'Configuration avancée'*, décochez *'Utiliser le DNS pour la résolution de noms'*, et cochez *'Utiliser LMHOSTS'*, a moins que vous sachiez ce que font ces options. Si vous voulez utiliser un fichier `LMHOSTS`, il est situé dans `C :\winnt\system32\drivers\etc`.
9. Cliquez alors sur *'OK'* sur toutes les boites de dialogue, et redémarrez le système.
10. `Ping`ez la machine Linux pour tester la connexion réseau : *'Fichier'* –> *'Exécuter'*, taper : `ping 192.168.0.1` (C'est juste un test de connexion locale, vous ne pouvez pas encore `ping`er le monde extérieur). Si vous ne voyez pas de réponses a vos pings, vérifiez votre configuration réseau.

4.3 Configurer Windows pour Workgroup 3.11

1. Si vous n'avez pas encore installé votre carte réseau et son driver, faites le maintenant. La description de cela dépasse le cadre de ce document.
2. Installez le package TCP/IP 32b si ce n'est pas déjà fait.
3. Dans *'Groupe Principal'* –> *'Installation'* –> *'Configuration réseau'*, cliquez sur *'Drivers'*.
4. Sélectionnez *'Microsoft TCP/IP-32 3.11b'* dans la section *'Drivers Réseaux'*. Choisissez *Configuration*.
5. Saisissez l'adresse IP 192.168.0.x (1 < x < 255), et positionnez le masque de sous réseau à 255.255.255.0 et la passerelle par défaut à 192.168.0.1.
6. Ne sélectionnez pas *'Configuration automatique DHCP'* ou mettez n'importe quoi dans la case *'Server WINS'*, à moins que vous ne fassiez partie d'un domaine Windows NT et que vous sachiez ce que vous faites.
7. Cliquez sur *'DNS'*, et remplissez les informations appropriés, ajoutez tous les serveurs DNS qui sont configurés sur votre machine Linux (on les trouve dans le fichier `/etc/resolv.conf`). Cliquez sur *'OK'* une fois que c'est fini.
8. Cliquez sur *'Configuration avancée'*, cochez *'Utiliser le DNS pour la résolution de noms'*, et *'Utiliser LMHOSTS'* situé dans `c :\windows`.
9. Cliquez alors sur *'OK'* sur toutes les boites de dialogue, et redémarrez le système.
10. `Ping`ez la machine Linux pour tester la connexion réseau : *'Fichier'* –> *'Exécuter'*, taper : `ping 192.168.0.1` (C'est juste un test de connexion locale, vous ne pouvez pas encore `ping`er le monde extérieur. Si vous ne voyez pas de réponses a vos pings, vérifiez votre configuration réseau.

4.4 Configurer les systèmes UNIX

1. Si vous n'avez pas encore installé votre carte réseau et recompilé votre noyau avec le driver adéquat, faites le maintenant. La description de cela dépasse le cadre de ce document.
2. Installez des outils TCP/IP, comme par exemple le package net-tools, si ce n'est déjà fait.
3. Affectez *IPADDR* à 192.168.0.x (1 < x < 255), puis *NETMASK* à 255.255.255.0, *GATEWAY* à 192.168.0.1 et *BROADCAST* à 192.168.0.255.

   Par exemple, sur les systèmes RedHat Linux, vous pouvez éditer le fichier `/etc/sysconfig/network-scripts/ifcfg-eth0`, ou simplement le faire par l'intermédiaire du *Control Panel*. C'est différent sur d'autres unices comme SunOS, BSDi, Slackware Linux, Solaris, SuSe, Debian, etc. Référez vous a la documentation de votre unix pour plus d'informations.
4. Ajoutez l'adresse IP de votre DNS et votre ordre de recherche DNS dans `/etc/resolv.conf`. Et pour certaines versions d'unix, éditez votre `/etc/nsswitch.conf` pour autoriser le service DNS.
5. Il sera éventuellement nécessaire de mettre à jout le fichier `/etc/networks`, selon votre configuration.
6. Redémarrez les services adéquats, ou, plus simplement, redémarrez votre système.
7. Testez votre connexion avec la passerelle en utilisant la commande `ping` : `ping 192.168.0.1`. Ceci est juste un test sur votre réseau local, vous ne pouvez pas encore `ping`er l'extérieur. Si vous ne voyez pas de réponses a vos pings, vérifiez votre configuration réseau.

4.5 Configuration sous DOS avec le package NCSA

1. Si vous n'avez pas encore installé votre carte réseau, faites le maintenant. La description de cela dépasse le cadre de ce document.
2. Chargez le driver adéquat. Par exemple, si vous utilisez une carte ethernet NE2000 configurée pour utiliser le port d'E/S 0x300 et l'IRQ 10, tapez `nwpd 0x60 10 0x300`.
3. Créez un nouveau répertoire, et décompressez-y l'archive NCSA Telnet : `pkunzip tel2308b.zip`
4. Utilisez un éditeur de texte pour ouvrir le fichier `config.tel`.
5. Affectez `myip=192.168.0.x` ($1 < x < 255$), et `netmask=255.255.255.0`.
6. Dans cet exemple, vous auriez à régler `hardware=packet, interrupt=10, ioaddr=60`.
7. Vous devriez avoir au moins une seule machine déclarée comme passerelle, à savoir la machine sous Linux :

   ```
   name=default
   host=le_nom_de_votre_hote_linux
   hostip=192.168.0.1
   gateway=1
   ```

8. Pour mettre en place le DNS :

   ```
   name=dns.domain.com~; hostip=123.123.123.123; nameserver=1
   ```

   NB : remplacez les champs par les informations qu'utilise votre machine Linux.
9. Sauvegardez votre nouveau fichier `config.tel`.
10. Lancez un telnet vers la machine Linux pour tester la connexion réseau : `telnet 192.168.0.1`. Si vous n'obtenez pas un prompt de login, Vérifiez votre configuration réseau.

4.6 Configuration des systèmes MacOS utilisant MacTCP

1. Si vous n'avez pas encore installé le driver pour votre carte Ethernet, ça serait une excellente idée de le faire maintenant. La description de cela dépasse le cadre de ce document.
2. Ouvrez le *'Tableau de bord MacTCP'*. Selectionnez le driver réseau adapté (Ethernet, PAS EtherTalk) et cliquez sur le bouton.
3. Dans la section *'Obtenir l'adresse'*, sélectionnez *'Manuellement'*.
4. Dans *'Adresse IP'*, choisissez *'class C'* dans le menu déroulant. Vous pouvez ignorer le reste de cette boite de dialogue.
5. Remplissez la section *'Information DNS'* avec les informations qui conviennent.
6. Dans *'Adresse de la passerelle'*, entrez 192.168.0.1.
7. Cliquez sur *'OK'* pour sauvegarder les changements. Dans la fenêtre principale du *'Tableau de bord MacTCP'*, entrez l'adresse IP de votre Mac (192.168.0.x, $1 < x < 255$) dans la zone *'Adresse IP'*.
8. Refermez le *'Tableau de bord MacTCP'*. Si une boite de dialogue vous demande de redémarrer le système, faites le.
9. Vous pouvez si vous le désirez `pinger` l'hôte Linux pour tester la connexion réseau. Si vous avez le programme freeware *'MacTCP Watcher'*, cliquez sur le bouton *Ping* et entrez l'adresse de votre hôte Linux (192.168.0.1) dans la boîte de dialogue qui apparait. (C'est uniquement une connexion locale, vous ne pouvez pas encore `pinger` l'extérieur). Si vous ne voyez pas de réponses a vos pings, vérifiez votre configuration réseau.
10. Vous pouvez, si vous le désirez, créer un fichier `Hosts` dans votre dossier système, pour pouvoir utiliser les noms d'hôte des machines de votre réseau local. Le fichier devrait déjà exister dans votre dossier système, et contenir quelques exemples commentés, que vous n'avez qu'à modifier pour correspondre à vos besoins.

4.7 Configuration des systèmes MacOS utilisant Open Transport

1. Si vous n'avez pas encore installé le driver pour votre carte Ethernet, ça serait une excellente idée de le faire maintenant. La description de cela dépasse le cadre de ce document.
2. Ouvrez le *'Tableau de bord TCP/IP'* et choisissez *'Mode utilisateur...'* dans le menu *'Edition'*. Assurez nous que le mode utilisateur est mis au niveau *'Avancé'* et cliquez sur le bouton *'OK'*.
3. Choisissez *'Configurations...'* depuis le menu *'Fichier'*. Sélectionnez la configuration *'Par défaut'* et cliquez sur le bouton *'Recopier'*. Entrez 'IP Masq' (ou quelque chose d'autre du moment que vous puissiez etre sur qu'il s'agit d'une configuration spéciale) dans la boite de dialogue *'Configuration de copie'*. Cliquez sur le bouton *'OK'* puis sur *'Rendre active'*.
4. Sélectionnez *'Ethernet'* depuis le menu *'Se connecter via...'*.
5. Sélectionnez l'option qui convient dans le menu *'Configuration'*. Si vous ne savez pas quelle option choisir, vous devriez sans doute resélectionner la configuration par défaut et quitter. Je choisis *'Manuellement'*.
6. Saisissez l'adresse IP de votre Mac (192.168.0.x, 1 < x < 255) dans la zone *'Adresse IP'*.
7. Mettez le *'Masque de sous réseau'* à 255.255.255.0.
8. L'*'Adresse de routeur'* est 192.168.0.1.
9. Remplissez la case *'Adresse du DNS'* en y mettant votre adresse IP.
10. Entrez le nom de votre domaine Internet (par exemple 'microsoft.com') dans la boite de dialogue *'Ordre de recherche DNS'*.
11. La procédure suivante est optionnelle. L'utilisation de valeurs incorrectes peut entrainer des comportements inattendus. Si vous ne savez pas ce que vous faites, il vaut mieux ne pas y toucher, et si nécessaire vider les cases et zones de sélection. Pour ce que j'en sais, il n'est pas possible, par l'intermédiaire des boites de dialogue, de demander au système de ne pas utiliser un fichier "Hosts" sélectionné précédemment. Si vous saviez comment faire, je serais très intéressé.

    Sélectionnez l'option *'802.3'* si votre réseau nécessite des paquets de type 802.3.
12. Cliquez sur le bouton *'Options...'* pour vous assurer que le TCP/IP est activé. J'utilise l'option *'Charger uniquement si besoin'*. Si vous lancez et quittez des applications utilisant TCP/IP assez souvent, sans relancer votre machine, vous pourrez sans doute désélectionner *'Charger uniquement si besoin'* pour diminuer les effets sur le gestionnaire mémoire de votre machine. Lorsque l'option est désélectionnée, les piles du protocole TCP/IP sont toujours en mémoire et prêtes à l'emploi. Si l'option est cochée, la pile TCP/IP est automatiquement chargée lorsqu'elle est nécessaire, et déchargée sinon. Le processus de la charger et la décharger en mémoire peut fragmenter la mémoire de votre ordinateur.
13. `Ping`ez la machine Linux pour tester la connexion réseau. Si vous avez le programme freeware *MacTCP Watcher*, cliquez sur le bouton *'Ping'*, et entrez l'adresse de votre machine Linux (192.168.0.1) dans la boite de dialogue qui apparait. (C'est une connexion locale, vous ne pouvez pas encore `ping`er l'extérieur). Si vous ne voyez pas de réponses a vos pings, vérifiez votre configuration réseau.
14. Vous pouvez aussi créer un fichier `Hosts` dans votre dossier Système, pour pouvoir utiliser les noms d'hotes de votre réseau local. Le fichier peut exister ou non dans votre dossier Système. Si c'est le cas, il devrait contenir des exemples (en commentaires) que vous pouvez modifier selon vos souhaits. Sinon, vous pouvez obtenir une copie d'un système utilisant MacTCP, ou juste créer le votre (cela ressemble fortement au fichier `/etc/hosts` sur un système Unix, qui est décrit dans la RFC 952). Une fois le fichier créé, ouvrez le *'Tableau de bord TCP/IP'*, cliquez sur le bouton *'Sélectionner le fichier Hosts...'*, et ouvrez le fichier `Hosts`.
15. Cliquez sur *'Fermer'* ou choisissez *'Fermer'* ou *'Quitter'* depuis le menu *'Fichier'*, et cliquez alors sur le bouton *'Enregistrer'* pour enregistrer vos changements.
16. Les changements prennent effet immédiatement, mais cela ne fera pas de mal de rebouter le système.

4.8 Configurer un réseau Novell utilisant le DNS

1. Si vous n'avez pas encore installé le gestionnaire de périphérique de votre adaptateur Ethernet, faites le dès maintenant. La description de cela dépasse le cadre de ce document.
2. Téléchargez tcpip16.exe depuis *The Novell LanWorkPlace page* <ftp://ftp.novell.com/pub/updates/unixconn/lwp5>
3. Editez `c :\nwclient\startnet.bat` (voici une copie du mien) :

```
SET NWLANGUAGE=ENGLISH
LH LSL.COM
LH KTC2000.COM
LH IPXODI.COM
LH tcpip
LH VLM.EXE
F:
```

4. Editez `c :\nwclient\net.cfg` (changez le `Link drivers`, NE2000 dans mon cas) :

```
Link Driver KTC2000
        Protocol IPX 0 ETHERNET_802.3
        Frame ETHERNET_802.3
        Frame Ethernet_II
        FRAME Ethernet_802.2

NetWare DOS Requester
           FIRST NETWORK DRIVE = F
           USE DEFAULTS = OFF
           VLM = CONN.VLM
           VLM = IPXNCP.VLM
           VLM = TRAN.VLM
           VLM = SECURITY.VLM
           VLM = NDS.VLM
           VLM = BIND.VLM
           VLM = NWP.VLM
           VLM = FIO.VLM
           VLM = GENERAL.VLM
           VLM = REDIR.VLM
           VLM = PRINT.VLM
           VLM = NETX.VLM

Link Support
        Buffers 8 1500
        MemPool 4096

Protocol TCPIP
        PATH SCRIPT     C:\NET\SCRIPT
        PATH PROFILE    C:\NET\PROFILE
        PATH LWP_CFG    C:\NET\HSTACC
        PATH TCP_CFG    C:\NET\TCP
        ip_address      192.168.0.xxx
        ip_router       192.168.0.1
```

5. Changez l'adresse IP dans le champ *'ip_ address'* (192.168.0.x, $1 < x < 255$) et finalement, créez `c :\bin\resolv.cfg` :

```
SEARCH DNS HOSTS SEQUENTIAL
NAMESERVER xxx.xxx.xxx.xxx
NAMESERVER yyy.yyy.yyy.yyy
```

6. Maintenant, éditez les entrées *'NAMESERVER'* et remplacez les par l'adresse ip de votre serveur DNS local.
7. Testez votre connexion avec la passerelle en utilisant la commande `ping` : `ping 192.168.0.1`. Ceci est juste un test sur votre réseau local, vous ne pouvez pas encore `ping`er l'extérieur. Si vous ne voyez pas de réponses a vos pings, vérifiez votre configuration réseau.

4.9 Configurer OS/2 Warp

1. Si vous n'avez toujours pas configuré votre adaptateur réseau Ethernet, c'est le moment de le faire. La description de cela dépasse le cadre de ce document.
2. Installez le protocole TCP/IP s'il n'est pas déjà présent.
3. Allez dans les paramètres *'Programs/TCP/IP(LAN)/TCP/IP'*
4. Dans *'Network'*, ajoutez votre adresse TCP/IP (192.168.0.x) et configurez votre masque de sous réseau (255.255.255.0)
5. Dans *'Routing'* cliquez sur *'Ajouter'*. Sélectionnez *'default'* pour le *Type* and entrez l'adresse de votre machine Linux dans le champs *'Router Address'* (192.168.0.1).
6. Utilisez la même adresse DNS (Serveur de noms) que celle de votre machine Linux.
7. Fermez le panneau de contrôle de TCP/IP. Répondez oui au (à la) question(s) suivante(s).
8. Reboutez votre système.
9. Vous devriez être en mesure de `ping`er votre hôte Linux pour tester la configuration réseau. Taper `'ping 192.168.0.1'` dans une boîte de commande OS/2. Si vous recevez les paquets IP, tout fonctionne correctement.

4.10 Configurer les autres systèmes

Ces systèmes devraient suivre la même logique d'installation. Lisez les sections précédentes. Si vous êtes intéressés par l'écriture de la documentation sur n'importe quel système, comme OS/2, ou une variété quelconque de système Unix, envoyez s'il vous plait des instructions détaillées à *ambrose@writeme.com* (Note du traducteur : en anglais bien sûr).

5 Tester IP Masquerade

Enfin, il est temps de faire un vrai test de votre IP masquerading après tout ce dur labeur. Si vous n'avez pas encore rebooté votre machine Linux, faites le maintenant, et assurez vous que votre machine démarre sans problèmes. Exécutez vos règles dans `/etc/rc.d/rc.firewall`. Ensuite, assurez vous que votre réseau local fonctionne bien.

Maintenant, faites ce qui suit :

1. Depuis une machine, interne MASQuée, essayez de `ping`er une machine locale (par exemple : *ping 192.168.0.10*). C'est pour vérifier que votre TCP/IP marche bien sur cette machine. Si cela ne marche pas, assurez vous de bien avoir configuré le TCP/IP sur le PC MASQué comme décrit plus haut dans ce HOWTO.

2. Sur le serveur MASQ, `ping`ez des adresses IP internes (par exemple : *ping 192.168.0.10*). Maintenant, `ping`ez l'adresse IP externe connectée a internet. Cette adresse IP devrais être votre adresse PPP, Ethernet, etc. connectée a votre FAI. Si vous ne connaissez pas cette adresse IP, lancez la commande *"/sbin/ifconfig"* depuis votre serveur MASQ pour récupérer votre adresse Internet. Cela confirmera que le serveur MASQ a une connectivité totale a Internet.
3. Revenez sur une machine MASQuée, et essayez de `ping`er l'adresse IP de la carte Ethernet interne de votre serveur MASQ (par exemple : *ping 192.168.0.1*). Cela prouvera que le routage de votre réseau interne marche. Si ça ne marche pas, assurez vous que la carte Ethernet de votre serveur a bien le "link". C'est habituellement une diode qui est sur le derriere de la carte Ethernet et aussi sur le hub/switch Ethernet (si vous en utilisez un).
4. Depuis une machine MASQuée, `ping`ez l'adresse externe de votre serveur MASQ que vous avez déjà testé dans l'item 2 ci dessus. Cette adresse IP devrais être votre adresse PPP, Ethernet, etc. connectée a votre FAI. Ce test de `ping` prouvera que votre masquerading marche (enfin, le masquerading ICMP au moins). Si ça ne marche pas, assurez vous d'avoir bien coché "ICMP Masquerading" dans la configuration de votre noyau et que vous avez bien "IP Forwarding" dans votre `/etc/rc.d/rc.firewall`. Assurez vous que les règles de votre `/etc/rc.d/rc.firewall` sont bien chargées. Essayez de lancer le script `/etc/rc.d/rc.firewall` manuellement pour vérifier qu'il fonctionne bien.

   Si ça ne marche toujours pas, allez voir informations données par la sortie des commandes suivantes :
   - "*ifconfig*" : Assurez vous que vous avez bien votre connexion internet d'établie et que vous avez la bonne adresse IP pour votre connexion internet.
   - "*netstat -rn*" : Assurez vous que votre passerelle par défaut est bien (la colonne avec une adresse IP en gateway)
   - "*cat /proc/sys/net/ipv4/ip_forward*" : Assurez vous bien que ça réponde "`1`" ce qui prouvera que l'ip forwarding est activé
   - "*/sbin/ipfwadm -F -l*" pour les 2.0 ou "*/sbin/ipchains -F -L*" pour les 2.2.x : Assurez vous de bien avoir le MASQ d'intégré
5. Pour une machine MASQuée, `ping`ez une adresse IP statique sur internet (par exemple : *ping 152.19.254.81* (c'est `http ://metalab.unc.edu/` maison du LDP). Si ça marche, cela signifie que le masquerading ICMP marche via internet. Si cela n'a pas marché, revérifiez votre connexion Internet. Si ça ne marche toujours pas, assurez vous que les règles de votre rc.firewall sont bien chargées et que vous avec compilé l'"ICMP Masquerading" dans votre noyau.
6. Maintenant, faites un TELNET sur une adresse IP distante (par exemple : *telnet 152.2.254.81* (metalab.unc.edu - notez que cela peut prendre longtemps avant d'avoir le prompt de login car c'est un serveur très chargé). Vous avez eu le prompt de login ? Si ça a marché, cela signifie que le masquerading TCP marche aussi. Si non, essayez de faire un telnet sur une autre machine qui le supporte comme 198.182.196.55 (www.linux.org). Si ça ne marche toujours pas, vérifiez que vous utilisez bien des règles simples pour l'instant.
7. Maintenant, essayez de faire un telnet sur un nom de machine (par exemple : *"telnet metalab.unc.edu"* (152.2.254.81)). Si ça marche aussi, cela signifie que le DNS marche aussi. Si ça ne marche pas mais que l'étape 4 a marchée, assurez vous de bien avoir configuré les bons serveurs DNS sur votre ordinateur MASQué
8. Comme dernier test, essayez de naviguer sur quelques sites web sur *'Internet'* depuis l'une de vos machines **MASQuées**, et voyez si vous pouvez les atteindre. Par exemple, essayez d'accéder au *site du LDP*. Si ça marche, vous pouvez être fier, tout marche sans problèmes !

Si vous voyez la page du LDP **FÉLICITATIONS ! Ça marche !** Si le site web apparaît bien, les autres outils réseau comme `ping`, `telnet`, `ssh` et avec leurs modules chargés : `ftp`, Real Audio, les DCC IRC, Quake I,][,]I[, CuSeeme, vdolive, etc. devraient aussi marche ! Si le `ftp`, l'`irc`, ou autres marchent mal ou pas du tout, assurez vous que leurs modules respectifs sont bien chargés en lançant `lsmod` et aussi, assurez vous de bien charger les modules avec les bons ports. Si vous ne voyez pas un module dont vous avez besoin, assurez vous qu'il est bien chargé dans le `/etc/rc.d/rc.firewall` (par exemple : enlevez le # qui se trouve au début de la ligne).

6 Autres sujets relatifs à IP Masquerade et au support logiciel

6.1 Problèmes avec IP Masquerade

Certains protocoles TCP/IP propriétaires ne marcheront pas avec l'IP masquerading Linux, soit car ils supposent des choses sur les numéros de port ou soit qu'ils encodent les adresses/ports TCP/IP dans leurs les données. Ces protocoles ont besoin de proxy ou de modules intégrés dans le code du masquerading pour fonctionner.

6.2 Services entrants

A la base, le masquerading ne peut pas du tout prendre en charge les services entrants mais il y a plusieurs façons de les autoriser

Si vous n'avez pas besoin d'une grande sécurité, vous pouvez simplement rediriger ou forwarder les ports. Il y a de nombreuses façons de faire cela toutefois, la meilleur solution est d'utiliser IPPORTFW. Pour plus d'informations, référez vous a la section 6.7 ().

Si vous désirez avoir des niveaux d'autorisation sur les connexions entrantes, vous aurez alors soit à utiliser les `TCP Wrappers` soit `Xinetd` pour autoriser seulement des adresses IP données, ou utiliser d'autre outils. La boîte à outils pour firewall TIS (TIS Firewall Toolkit) est un bon produit pour ceux qui cherchent des outils et des informations.

Plus de détails sur la sécurité peuvent être trouvés dans le document *TrinityOS* `<http://www.ecst.csuchico.edu/dranch/LINUX/TrinityOS.wri>` et sur *la page de l'IP Masquerade* `<http://ipmasq.cjb.net>`.

6.3 Programmes clients supportés et autres remarques pour la configuration

> **** Voyez la *page* `<http://dijon.nais.com/~nevo/masq/>` sur les applications fonctionnant au travers d'IP Masquerading contient plein de bonnes informations. Malheureusement, Cette page n'est plus vraiment mise a jour, si vous voulez vous en occuper, envoyez un mail a *ambrose@writeme.com* et/ou *dranch@trinnet.net*.**

En général, toutes les applications qui utilisent TCP et/ou UDP de manière standard devraient fonctionner. Si vous avez une quelconque suggestion ou question à propos des applications compatibles avec IP masquerade, visitez *la page de l'IP Masquerade* `<http://ipmasq.cjb.net>`.

6.3.1 Les clients réseau qui fonctionnent avec l'IP Masq

Clients génériques

Archie

toutes les plateformes, client de recherche de fichiers (tous les clients ne fonctionnent pas) ;

FTP

toutes les plateformes, avec le module `ip_masq_ftp.o` (tous les sites ne fonctionnent pas avec certains clients ; par exemple, certains sites ne peuvent pas être atteints en utilisant ws_ftp32 mais fonctionnent avec Netscape) ;

Gopher
: toutes les plateformes ;

HTTP
: toutes les plateformes, naviguer sur le web ;

IRC
: toutes les plateformes, avec le module `ip_masq_irc.o` ;

NNTP
: toutes les plateformes, client news USENET ;

ping
: toutes plateformes, avec le patch ICMP

POP
: toutes les plateformes, clients de courrier électronique ;

SSH
: toutes plateformes, client Telnet/FTP sécurisé ;

SMTP
: toutes les plateformes, clients de courrier électronique ;

Telnet
: toutes les plateformes, sessions distantes ;

traceroute
: surtout les plateformes UNIX, certaines variantes ne devraient pas fonctionner ;

VRML
: Windows (peut être toutes les plateformes), réalité virtuelle ;

WAIS
: toutes les plateformes.

Clients Multimédia et de communication

Alpha Worlds
: Windows, programme client-serveur de discussion 3D

CU-SeeMe
: toutes les plateformes, avec le module cuseeme, voir la section 6.8.2 () pour les détails.

ICQ
: Tous clients supportés. Requiert d'avoir compilé le noyau avec IPPORTFW et qu'ICQ soit configuré pour être derrière un proxy non socks. Une description complète de la configuration se trouve dans la section 6.9 ().

Internet Phone 3.2
: Windows, communications audio. Vous ne pouvez être contacté que si vous initiez la connexion, mais on ne peut pas vous appeler si vous n'avez pas mis en place un port forwarding spécifique. Référez vous a la section 6.7 () pour plus de détails.

Internet Wave Player
: Windows, flux audio par réseau

Powwow

Windows, communication audio. Vous ne pouvez être contacté que si vous initiez la connexion, mais on ne peut pas vous appeler si vous n'avez pas mis en place un port forwarding spécifique. Référez vous a la section 6.7 () pour plus de détails.

Real Audio Player 2.0

Windows, flux audio par réseau, on obtient une meilleur qualité avec le module `ip_masq_raudio`.

True Speech Player 1.1b

Windows, flux audio par réseau

VDOLive

Windows, avec le patch *ip_ masq_ vdolive*

Worlds Chat 0.9a

Windows, programme client-serveur de discussion 3D

Jeux - Référez vous a la section 6.10 () pour plus d'informations sur comment utiliser le patch LooseUDP.

Battle.net

Fonctionne mais requiert que les ports TCP 116 et 118 et le port UDP 6112 soient IPPORTFW sur la machine qui joue. Référez vous a la section 6.7 () pour plus de détails. Notez que les serveurs FSGS et Bnetd requièrent toujours IPPORTFW car ils n'ont pas été réécrits pour marcher avec le NAT.

BattleZone 1.4

Fonctionne avec le patch LooseUDP et les nouvelles *.DLLs d'Activision* qui marchent avec le NAT.

Dark Reign 1.4

Fonctionne avec le patch LooseUDP ou requiert que les ports TCP 116 et 118 et le port UDP 6112 soient IPPORTFW sur la machine qui joue. Référez vous a la section 6.7 () pour plus de détails

Diablo

Fonctionne avec le patch LooseUDP ou requiert que les ports TCP 116 et 118 et le port UDP 6112 soient IPPORTFW sur la machine qui joue. Les dernières version de Diablo utilisent uniquement les ports TCP et UDP 6112. Référez vous a la section 6.7 () pour plus de détails

Heavy Gear 2

Fonctionne avec le patch LooseUDP ou requiert que les ports TCP 116 et 118 et le port UDP 6112 soient IPPORTFW sur la machine qui joue. Référez vous a la section 6.7 () pour plus de détails

Quake I/][/]I[

Fonctionne sans problèmes avec le module *ip_ masq_ quake* si il y a plusieurs joueurs derrière la machine MASQ. Ce module ne supporte que Quake I et QuakeWorld par défaut. Si vous avez besoin d'utiliser Quake][ou des ports non standards, regardez comment s'installe le module dans les sections 3.3 () et 3.3.1 ().

StarCraft

Fonctionne avec le patch LooseUDP ou requiert que les ports TCP et UDP 6112 soient IPPORTFW sur la machine qui joue. Référez vous a la section 6.7 () pour plus de détails

WorldCraft

Fonctionne avec le patch LooseUDP.

Autres clients

Linux net-acct package

Linux, package d'administration par réseau

NCSA Telnet 2.3.08

DOS, une suite de logiciels contenant telnet, ftp, ping, etc...

PC-anywhere pour Windows
MS-Windows, controle d'un PC à distance avec TCP/IP, fonctionne uniquement si la machine est un client et non un hôte si vous n'avez pas joué avec IPPORTFW. Référez vous a la section 6.7 () pour plus de détails

Socket Watch
utilise ntp - network time protocol

6.3.2 Clients qui ne fonctionnent pas

Tous les programmes H.323
MS Netmeeting, Intel Internet Phone Beta 2... Connexion ok, mais la voix ne peut que sortir de votre réseau. Allez faire un tour sur la passerelle H.323 *Equivalence's PhonePatch* `<http://www.equival.com.au/phonepatch/index.html>` qui est une solution possible.

Intel Streaming Media Viewer Beta 1
Connexion impossible au serveur.

Netscape CoolTalk
Connexion à l'hôte distant impossible.

WebPhone
Ne peut pas fonctionner (il fait des suppositions invalides sur les adresses).

6.4 Regles ipfwadm plus sures

Cette section constitue un guide plus précis sur l'utilisation de l'outil de reglage du firewall sur les noyaux 2.0, ipfwadm. Allez voire plus bas pour ipchains.

Voici un exemple de script d'initialisation pour un système qui fait office de firewall et de masquerading derriere un lien PPP avec une IP statique (des instructions pour les ip dynamiques sont données, mais commentées). L'interface à laquelle on fait confiance est 192.168.0.1 (celle du réseau local) et l'adresse IP de l'interface PPP a été changée pour des raisons de sécurité. Toutes les interfaces sont listées individuellement pour intercepter l'IP spoofing et les routages inexacts. Tout ce qui n'est pas explicitement autorisé est **INTERDIT** (disons plutôt rejeté). Si machine IP MASQ ne marche plus apres avoir lancé ce script, assurez vous de bien avoir édité le script pour qu'il sont en accord avec votre configuration, allez aussi jeter un coup d'oeil dans votre syslog `/var/log/messages` ou `/var/adm/messages` pour y trouver une ou deux erreurs de firewall.

Pour des exemples plus précis sur l'implémentation de règles sures pour PPP, modem Câble, etc., référez vous à *TrinityOS - Section 10* `<http://www.ecst.csuchico.edu/dranch/LINUX/TrinityOS.wri>` et *GreatCircle's Firewall WWW page*

NOTE : Si vous avez une adresse TCP/IP assignée dynamiquement par votre FAI (PPP, ADSL, Câble, etc.), vous **ne pouvez pas charger ce script au boot**. Vous allez soit avoir a le recharger a chaque fois que vous changez d'adresse IP, soit rendre votre `/etc/rc.d/rc.firewall` plus intelligent. Pour les utilisateurs de PPP, lisez attentivement et décommentez les lignes qui vous concernent dans la section "Récupération dynamique de l'IP en PPP" ci dessous. Vous pourrez aussi trouver plus de détails dans *TrinityOS - Section 10* `<http://www.ecst.csuchico.edu/dranch/LINUX/TrinityOS.wri>` pour plus de détails sur les règles plus sures avec des IP Dynamiques.

Rappelez vous qu'il y a plein d'outils graphiques pour régler votre Firewall. Référez vous a la section 7 () pour plus de détails.

Enfin, si vous utilisez une adresse PPP statique, changez la ligne "ppp_ip = "votre_adresse_ip_statique"" pour refléter votre adresse.

```
#!/bin/sh
#
# /etc/rc.d/rc.firewall, un exemple de regles plutot sures de firewall avec ipfwadm
#

PATH=/sbin:/bin:/usr/sbin:/usr/bin

# pour les tests, attend un moment puis efface toutes les règles du
# firewall. Décommentez les lignes suivantes si vous voulez que le firewall soit
# désactivé automatiquement après 10 minutes.
# (sleep 600; \
# ipfwadm -I -f; \
# ipfwadm -I -p accept; \
# ipfwadm -O -f; \
# ipfwadm -O -p accept; \
# ipfwadm -F -f; \
# ipfwadm -F -p accept; \
# ) &


#########################################################################################
# Necessaire pour charger les modules
#
/sbin/depmod -a

# Supporte le masquerading des transferts de fichiers FTP utilisant la méthode PORT.
#
/sbin/modprobe ip_masq_ftp

# Supporte le masquerading de RealAudio via UDP. Sans ce module, RealAudio FONCTIONNERA,
# mais en mode TCP. Cela peut causer une réduction dans la qualité du son.
#
#/sbin/modprobe ip_masq_raudio

# Supporte le masquerading des transferts IRC DCC.
#
#/sbin/modprobe ip_masq_irc

# Supporte le masquerading de Quake et QuakeWorld par défaut. Ce module sert si vous
# avez plusieurs utilisateurs derriere le serveur MASQ. Si vous allez jouer a Quake I,
# ][ et ]I[, utilisez le deuxieme exemple.
#
#Quake I / QuakeWorld (ports 26000 et 27000)
#/sbin/modprobe ip_masq_quake
#
#Quake I/II/III / QuakeWorld (ports 26000, 27000, 27910, 27960)
#/sbin/modprobe ip_masq_quake ports=26000,27000,27910,27960

# Supporte le masquerading du systeme de visioconférences CuSeeMe
#
```

```
#/sbin/modprobe ip_masq_cuseeme

# Supporte le masquerading du systeme de visioconférences VDO Live
#
#/sbin/modprobe ip_masq_vdolive


# CRITIQUE : Permet le relayage de paquets IP car il est désactivé par défaut depuis le
#            noyau 2.0.34
#
# Utilisateurs RedHat : Vous pouvez aussi changer l'option dans /etc/sysconfig/network
#                              de :
#                        FORWARD_IPV4=false
#                              à :
#                        FORWARD_IPV4=true
#
echo "1" > /proc/sys/net/ipv4/ip_forward

# Utilisateurs d'IP dynamiques :
#
# Si vous avez une adresse IP dynamique via SLIP, PPP ou DHCP, décommentez l'option
# suivante. Elle permet d'utiliser une bidouille dans l'IP MASQ qui sert pour les IP
# dynamiques, ce qui rends la vie avec DialD, PPPd et les programmes similaires bien
# plus facile.
#
#echo "1" > /proc/sys/net/ipv4/ip_dynaddr


# Spécifiez votre adresse IP statique ici.
#
#   Si vous utilisez une adresse IP DYNAMIQUE, vous avez besoin de faire comprendre a ce
#   jeu de règles que votre IP a changée a chaque fois qu'elle change. Pour faire cela,
#   décommentez le script suivant. (Notez qu'il y a une différence importante entre les
#   guillemets simples et doubles).
#
#   Vous aurez aussi besoin soit de créer le lien suivant ou que votre /etc/ppp/ip-up
#   existant lance votre /etc/rc.d/rc.firewall.
#
#       ln -s /etc/rc.d/rc.firewall /etc/ppp/ip-up
#
#   Si le fichier /etc/ppp/ip-up existe déjà, vous devriez l'éditer et ajouter une ligne
#   contenant "/etc/rc.d/rc.firewall" aux alentours de la fin du fichier.
#
#   Si vous ne vous en etes pas encore rendu comptes, le script /etc/ppp/ip-up est
#   toujours lancé lorsqu'une connexion PPP est établie. Grâce a cela, nous pouvons dire
#   au script d'aller chercher la nouvelle adresse IP pour pouvoir mettre a jour les
#   règles sures.
#
#   Utilisateurs PPP :
#        Le bout de script suivant devrais marcher sans problèmes.
```

```
#
#   Utilisateurs de DHCP :
#        Si vous obtenez votre adresse TCP/IP via DHCP, vous aurez a remplacer le mot
#        ppp0 par le nom de votre interface via laquelle vous êtes connectés a Internet
#        (eth0, eth1, etc). Il devrais aussi être noté que le DHCP peut changer votre IP
#        adresse. Pour régler cela vous devriez configurer votre client DHCP pour
#        relancer vos règles de firewall a chaque vois que votre bail est
#        renouvelé. Pour les utilisateurs de DHCPcd, utilisez l'option -c.
#
#ppp_ip = "`/sbin/ifconfig ppp0 | perl -ne 'if (/inet addr:([^ ]*)/) {print $1}'`"
#
ppp_ip = "votre_adresse_ip_statique"


# Durée de vie des connexion MASQuées
#
#    2 hrs pour une session TCP
#   10 sec apres qu'un paquet "FIN" soit passé
#  160 sec pour le traffic UDP (Important pour les utilisateurs d'ICQ)
#
/sbin/ipfwadm -M -s 7200 10 160


#########################################################################################
# Connexions entrantes, efface tout et positionne le comportement par défaut à reject
# (refus). En fait, le comportement par défaut est inadéquat puisqu'il y a une règle
# pour tout intercepter, avec refus et logging.
#
/sbin/ipfwadm -I -f
/sbin/ipfwadm -I -p reject

# interface locale, machines locales. Aller n'importe où est autorisé.
#
/sbin/ipfwadm -I -a accept -V 192.168.0.1 -S 192.168.0.0/24 -D 0.0.0.0/0

# interface distante, prétendant être une machine locale. C'est de l'IP spoofing, on
# refuse.
#
/sbin/ipfwadm -I -a reject -V $ppp_ip -S 192.168.0.0/24 -D 0.0.0.0/0 -o

# interface distante, n'importe qu'elle source, l'accès à notre adresse PPP est valide
#
/sbin/ipfwadm -I -a accept -V $ppp_ip -S 0.0.0.0/0 -D $ppp_ip/32

# l'interface loopback est valide.
#
/sbin/ipfwadm -I -a accept -V 127.0.0.1 -S 0.0.0.0/0 -D 0.0.0.0/0

# une fois toutes les règles faites, toutes les autres connexions entrantes sont
# refusées et logguées.
```

```
#
/sbin/ipfwadm -I -a reject -S 0.0.0.0/0 -D 0.0.0.0/0 -o


#########################################################################################
# Connexions sortantes,efface tout et positionne le comportement par défaut à reject
# (refus).En fait, le comportement par défaut est inadéquat puisqu'il y a une règle pour
# tout intercepter, avec refus et logging.
#
/sbin/ipfwadm -O -f
/sbin/ipfwadm -O -p reject

# interface locale, machines locales. N'importe quelle source allant vers le réseau
# local est valide.
#
/sbin/ipfwadm -O -a accept -V 192.168.0.1 -S 0.0.0.0/0 -D 192.168.0.0/24

# destination vers le réseau local à partir de l'interface sortante. C'est du routage
# piraté, tout refuser.
#
/sbin/ipfwadm -O -a reject -V $ppp_ip -S 0.0.0.0/0 -D 192.168.0.0/24 -o

# sortante depuis le réseau local sur l'interface sortante. C'est du masquerading
# pirate, tout refuser.
#
/sbin/ipfwadm -O -a reject -V  $ppp_ip -S 192.168.0.0/24 -D 0.0.0.0/0 -o

# sortante depuis le réseau local sur l'interface sortante. C'est du masquerading
# pirate, tout refuser.
#
/sbin/ipfwadm -O -a reject -V $ppp_ip -S 0.0.0.0/0 -D 192.168.0.0/24 -o

# Tout ce qui sort d'autre de l'interface distante est bon
#
/sbin/ipfwadm -O -a accept -V $ppp_ip -S $ppp_ip/32 -D 0.0.0.0/0

# l'interface loopback est valide.
#
/sbin/ipfwadm -O -a accept -V 127.0.0.1 -S 0.0.0.0/0 -D 0.0.0.0/0

# une fois toutes les règles faites, toutes les autres connexions sortantes sont
# refusées et logguées.
#
/sbin/ipfwadm -O -a reject -S 0.0.0.0/0 -D 0.0.0.0/0 -o


#########################################################################################
# Connexions à faire suivre (forwarding), efface tout et positionne le comportement par
# défaut à reject (refus). En fait, le comportement par défaut est inadéquat puisqu'il y
# a une règle pour tout intercepter, avec refus et logging.
```

```
#
/sbin/ipfwadm -F -f
/sbin/ipfwadm -F -p reject

# Masquerade depuis le réseau local sur l'interface locale vers n'importe où
#
/sbin/ipfwadm -F -a masquerade -W ppp0 -S 192.168.0.0/24 -D 0.0.0.0/0

# une fois toutes les règles faites, toutes les autres connexions à faire suivre sont
# refusées et logguées.
#
/sbin/ipfwadm -F -a reject -S 0.0.0.0/0 -D 0.0.0.0/0 -o
```

Avec IPFWADM, vous pouvez bloquer le trafic vers ou depuis un site particulier en utilisant -I, -O ou -F appropriées. Souvenez vous que les règles sont analysées de haut en bas, et "`-a`" signifie ajoute (*"append"*) à l'ensemble des règles existantes. Avec cela en mémoire, toute restriction spécifique doit venir avant les regles globales. Par exemple :

En utilisant les règles -I. Probablement le plus rapide mais stoppe uniquement les machines locales, le firewall peut encore accéder au site "interdit". C'est peut être d'ailleurs le comportement que vous désirez.

Dans le `/etc/rc.d/rc.firewall` :

```
... début des règles -I ...

# rejette et loggue l'interface locale et la machine locale allant sur 204.50.10.13
#
/sbin/ipfwadm -I -a reject -V 192.168.0.1 -S 192.168.0.0/24 -D 204.50.10.13/32 -o

# interface locale, machines locales. Aller n'importe où est autorisé.
#
/sbin/ipfwadm -I -a accept -V 192.168.0.1 -S 192.168.0.0/24 -D 0.0.0.0/0

... fin des règles -I ...
```

En utilisant les règles -O. C'est le plus lent puisque les paquets passent d'abord à travers le masquerading, mais cette règle empèche même au firewall d'accéder au site interdit.

```
... début des règles -O ...

# rejette et loggue les connexions sortantes vers 204.50.10.13
#
/sbin/ipfwadm -O -a reject -V $ppp_ip -S $ppp_ip/32 -D 204.50.10.13/32 -o

# tout le reste, sortant vers l'interface distante est valide
#
/sbin/ipfwadm -O -a accept -V $ppp_ip -S $ppp_ip/32 -D 0.0.0.0/0

... fin des règles -O ...
```

En utilisant les règles -F. Probablement plus lent qu'en utilisant les règles -I, et cela stoppe uniquement les machines pour lesquelles on effetue du masquerading (c'est à dire les machines internes). Le firewall peut encore accéder au site interdit.

```
... début des règles -F ...
```

```
# Rejette et loggue les connexions depuis le réseau local sur l'interface PPP vers
# 204.50.10.13.
#
/sbin/ipfwadm -F -a reject -W ppp0 -S 192.168.0.0/24 -D 204.50.10.13/32 -o

# Masquerade depuis le réseau local sur l'interface locale vers n'importe où
#
/sbin/ipfwadm -F -a masquerade -W ppp0 -S 192.168.0.0/24 -D 0.0.0.0/0

... fin des règles -F ...
```

Il n'y a pas besoin d'une règle spéciale pour autoriser 192.168.0.0/24 à se connecter sur 204.50.11.0, ce comportement est inclus dans les règles globales.

Il y a plus d'une façon d'écrire les règles précédentes. Par exemple, au lieu de -V 192.168.0.1, vous pouvez utiliser -W eth0, au lieu de -V $ppp_ip, vous pouvez utiliser -W ppp0. Le -V a été supprimé lors du passage a ipchains, mais pour les utilisateurs de ipfwadm, c'est une question de goût personnel.

6.5 Chaines IP Firewalling (ipchains)

Cette section constitue un guide plus précis sur l'utilisation de l'outil de reglage du firewall sur les noyaux 2.2, ipchains. Allez voire plus haut pour ipfwadm.

Voici un exemple de script d'initialisation pour un système qui fait office de firewall et de masquerading derriere un lien PPP avec une IP statique (des instructions pour les ip dynamiques sont données, mais commentées). L'interface à laquelle on fait confiance est 192.168.0.1 (celle du réseau local) et l'adresse IP de l'interface PPP a été changée pour des raisons de sécurité. Toutes les interfaces sont listées individuellement pour intercepter l'IP spoofing et les routages inexacts. Tout ce qui n'est pas explicitement autorisé est **INTERDIT** (disons plutôt rejeté). Si machine IP MASQ ne marche plus apres avoir lancé ce script, assurez vous de bien avoir édité le script pour qu'il sont en accord avec votre configuration, allez aussi jeter un coup d'oeil dans votre syslog `/var/log/messages` ou `/var/adm/messages` pour y trouver une ou deux erreurs de firewall.

Pour des exemples plus précis sur l'implémentation de règles sures pour PPP, modem Câble, etc., référez vous à *TrinityOS - Section 10* `<http://www.ecst.csuchico.edu/dranch/LINUX/TrinityOS.wri>` et *GreatCircle's Firewall WWW page*

NOTE : Si vous avez une adresse TCP/IP assignée dynamiquement par votre FAI (PPP, ADSL, Câble, etc.), vous **ne pouvez pas charger ce script au boot**. Vous allez soit avoir a le recharger a chaque fois que vous changez d'adresse IP, soit rendre votre `/etc/rc.d/rc.firewall` plus intelligent. Pour les utilisateurs de PPP, lisez attentivement et décommentez les lignes qui vous concernent dans la section "Récupération dynamique de l'IP en PPP" ci dessous. Vous pourrez aussi trouver plus de détails dans *TrinityOS - Section 10* `<http://www.ecst.csuchico.edu/dranch/LINUX/TrinityOS.wri>` pour plus de détails sur les règles plus sures avec des IP Dynamiques.

Rappelez vous qu'il y a plein d'outils graphiques pour régler votre Firewall. Référez vous a la section 7 () pour plus de détails.

Enfin, si vous utilisez une adresse PPP statique, changez la ligne "ppp_ip = "votre_adresse_ip_statique"" pour refléter votre adresse.

```
#!/bin/sh
#
```

```
# /etc/rc.d/rc.firewall: un exemple de regles plutot sures de firewall avec ipchains
#

PATH=/sbin:/bin:/usr/sbin:/usr/bin

# pour les tests, attend un moment puis efface toutes les règles du
# firewall. Décommentez les lignes suivantes si vous voulez que le firewall soit
# désactivé automatiquement après 10 minutes.
# (sleep 600; \
# ipchains -F \
# ipchains -X \
# ) &

#########################################################################################
# Necessaire pour charger les modules
#
/sbin/depmod -a

# Supporte le masquerading des transferts de fichiers FTP utilisant la méthode PORT.
#
/sbin/modprobe ip_masq_ftp

# Supporte le masquerading de RealAudio via UDP. Sans ce module, RealAudio FONCTIONNERA,
# mais en mode TCP. Cela peut causer une réduction dans la qualité du son.
#
#/sbin/modprobe ip_masq_raudio

# Supporte le masquerading des transferts IRC DCC.
#
#/sbin/modprobe ip_masq_irc

# Supporte le masquerading de Quake et QuakeWorld par défaut. Ce module sert si vous
# avez plusieurs utilisateurs derriere le serveur MASQ. Si vous allez jouer a Quake I,
# ][ et ]I[, utilisez le deuxieme exemple.
#
#Quake I / QuakeWorld (ports 26000 et 27000)
#/sbin/modprobe ip_masq_quake
#
#Quake I/II/III / QuakeWorld (ports 26000, 27000, 27910, 27960)
#/sbin/modprobe ip_masq_quake ports=26000,27000,27910,27960

# Supporte le masquerading du systeme de visioconférences CuSeeMe
#
#/sbin/modprobe ip_masq_cuseeme

# Supporte le masquerading du systeme de visioconférences VDO Live
#
#/sbin/modprobe ip_masq_vdolive
```

```
# CRITIQUE : Permet le relayage de paquets IP car il est désactivé par défaut depuis le
#            noyau 2.0.34
#
# Utilisateurs RedHat : Vous pouvez aussi changer l'option dans /etc/sysconfig/network
#                                de :
#                          FORWARD_IPV4=false
#                                à :
#                          FORWARD_IPV4=true
#
echo "1" > /proc/sys/net/ipv4/ip_forward

# Utilisateurs d'IP dynamiques :
#
# Si vous avez une adresse IP dynamique via SLIP, PPP ou DHCP, décommentez l'option
# suivante. Elle permet d'utiliser une bidouille dans l'IP MASQ qui sert pour les IP
# dynamiques, ce qui rends la vie avec DialD, PPPd et les programmes similaires bien
# plus facile.
#
#echo "1" > /proc/sys/net/ipv4/ip_dynaddr


# Spécifiez votre adresse IP statique ici.
#
#   Si vous utilisez une adresse IP DYNAMIQUE, vous avez besoin de faire comprendre a ce
#   jeu de règles que votre IP a changée a chaque fois qu'elle change. Pour faire cela,
#   décommentez le script suivant. (Notez qu'il y a une différence importante entre les
#   guillemets simples et doubles).
#
#   Vous aurez aussi besoin soit de créer le lien suivant ou que votre /etc/ppp/ip-up
#   existant lance votre /etc/rc.d/rc.firewall.
#
#       ln -s /etc/rc.d/rc.firewall /etc/ppp/ip-up
#
#   Si le fichier /etc/ppp/ip-up existe déjà, vous devriez l'éditer et ajouter une ligne
#   contenant "/etc/rc.d/rc.firewall" aux alentours de la fin du fichier.
#
#   Si vous ne vous en etes pas encore rendu comptes, le script /etc/ppp/ip-up est
#   toujours lancé lorsqu'une connexion PPP est établie. Grâce a cela, nous pouvons dire
#   au script d'aller chercher la nouvelle adresse IP pour pouvoir mettre a jour les
#   règles sures.
#
#   Utilisateurs PPP :
#        Le bout de script suivant devrais marcher sans problèmes.
#
#   Utilisateurs de DHCP :
#        Si vous obtenez votre adresse TCP/IP via DHCP, vous aurez a remplacer le mot
#        ppp0 par le nom de votre interface via laquelle vous êtes connectés a Internet
#        (eth0, eth1, etc). Il devrais aussi être noté que le DHCP peut changer votre IP
#        adresse. Pour régler cela vous devriez configurer votre client DHCP pour
#        relancer vos règles de firewall a chaque vois que votre bail est
```

```
#       renouvelé. Pour les utilisateurs de DHCPcd, utilisez l'option -c.
#
extint = "ppp0"
#extip = "`/sbin/ifconfig $extint | perl -ne 'if (/inet addr:([^ ]*)/) {print $1}'`"
#
extip = "votre_adresse_ip_statique"

# L'ip interne e
intint="eth0"
intnet="192.168.0.1/24"


# Durée de vie des connexion MASQuées
#
#    2 hrs pour une session TCP
#   10 sec apres qu'un paquet "FIN" soit passé
#   60 sec pour le traffic UDP (Les utilisateurs d'ICQ doivent configurer leur ICQ avec
#                              un timeout de firewall de 30 secondes)
#
ipchains -M -S 7200 10 60

#########################################################################################
# Connexions entrantes, efface tout et positionne le comportement par défaut à reject
# (refus). En fait, le comportement par défaut est inadéquat puisqu'il y a une règle
# pour tout intercepter, avec refus et logging.
#
ipchains -F input
ipchains -P input REJECT

# interface locale, machines locales. Aller n'importe où est autorisé.
#
ipchains -A input -i $intint -s $intnet -d 0.0.0.0/0 -j ACCEPT

# interface distante, prétendant être une machine locale. C'est de l'IP spoofing, on
# refuse.
#
ipchains -A input -i $extint -s $intnet -d 0.0.0.0/0 -l -j REJECT

# interface distante, n'importe qu'elle source, l'accès à notre adresse PPP est valide
#
ipchains -A input -i $extint -s 0.0.0.0/0 -d $extip/32 -j ACCEPT

# l'interface loopback est valide.
#
ipchains -A input -i lo -s 0.0.0.0/0 -d 0.0.0.0/0 -j ACCEPT

# une fois toutes les règles faites, toutes les autres connexions entrantes sont
# refusées et logguées.
#
ipchains -A input -s 0.0.0.0/0 -d 0.0.0.0/0 -l -j REJECT
```

```
#########################################################################################
# Connexions sortantes,efface tout et positionne le comportement par défaut à reject
# (refus).En fait, le comportement par défaut est inadéquat puisqu'il y a une règle pour
# tout intercepter, avec refus et logging.
#
ipchains -F output
ipchains -P output REJECT

# interface locale, machines locales. N'importe quelle source allant vers le réseau
# local est valide.
#
ipchains -A output -i $intint -s 0.0.0.0/0 -d $intnet -j ACCEPT

# destination vers le réseau local à partir de l'interface sortante. C'est du routage
# piraté, tout refuser.
#
ipchains -A output -i $extint -s 0.0.0.0/0 -d $intnet -l -j REJECT

# sortante depuis le réseau local sur l'interface sortante. C'est du masquerading
# pirate, tout refuser.
#
ipchains -A output -i $extint -s $intnet -d 0.0.0.0/0 -l -j REJECT

# Tout ce qui sort d'autre de l'interface distante est bon
#
ipchains -A output -i $extint -s $extip/32 -d 0.0.0.0/0 -j ACCEPT

# l'interface loopback est valide.
#
ipchains -A output -i lo -s 0.0.0.0/0 -d 0.0.0.0/0 -j ACCEPT

# une fois toutes les règles faites, toutes les autres connexions sortantes sont
# refusées et logguées.
#
ipchains -A output -s 0.0.0.0/0 -d 0.0.0.0/0 -l -j REJECT

#########################################################################################
# Connexions à faire suivre (forwarding), efface tout et positionne le comportement par
# défaut à reject (refus). En fait, le comportement par défaut est inadéquat puisqu'il y
# a une règle pour tout intercepter, avec refus et logging.
#
ipchains -F forward
ipchains -P forward DENY

# Masquerade depuis le réseau local sur l'interface locale vers n'importe où
#
ipchains -A forward -i $extint -s $intnet -d 0.0.0.0/0 -j MASQ
#
# une fois toutes les règles faites, toutes les autres connexions à faire suivre sont
```

```
# refusées et logguées.
#
ipchains -A forward -s 0.0.0.0/0 -d 0.0.0.0/0 -l -j REJECT
```

Avec IPCHAINS, vous pouvez bloquer le trafic vers ou depuis un site particulier en utilisant les règles "input", "output" et "forward". Souvenez vous que les règles sont analysées de haut en bas, et "`-A`" signifie ajoute (*"append"*) à l'ensemble des règles existantes. Avec cela en mémoire, toute restriction spécifique doit venir avant les regles globales. Par exemple :

En utilisant les règles "input". Probablement le plus rapide mais stoppe uniquement les machines locales, le firewall peut encore accéder au site "interdit". C'est peut être d'ailleurs le comportement que vous désirez.

Dans le `/etc/rc.d/rc.firewall` :

```
... début des règles "input" ...

# rejette et loggue l'interface locale et la machine locale allant sur 204.50.10.13
#
/sbin/ipfwadm -I -a reject -V 192.168.0.1 -S 192.168.0.0/24 -D 204.50.10.13/32 -o

# interface locale, machines locales. Aller n'importe où est autorisé.
#
/sbin/ipfwadm -I -a accept -V 192.168.0.1 -S 192.168.0.0/24 -D 0.0.0.0/0

... fin des règles -I ...
```

En utilisant les règles "output. C'est le plus lent puisque les paquets passent d'abord à travers le masquerading, mais cette règle empèche même au firewall d'accéder au site interdit.

```
... début des règles "output" ...

# rejette et loggue les connexions sortantes vers 204.50.10.13
#
/sbin/ipfwadm -O -a reject -V $ppp_ip -S $ppp_ip/32 -D 204.50.10.13/32 -o

# tout le reste, sortant vers l'interface distante est valide
#
/sbin/ipfwadm -O -a accept -V $ppp_ip -S $ppp_ip/32 -D 0.0.0.0/0

... fin des règles -O ...
```

En utilisant les règles "forward". Probablement plus lent qu'en utilisant les règles -I, et cela stoppe uniquement les machines pour lesquelles on effetue du masquerading (c'est à dire les machines internes). Le firewall peut encore accéder au site interdit.

```
... début des règles "forward" ...

# Rejette et loggue les connexions depuis le réseau local sur l'interface PPP vers
# 204.50.10.13.
#
/sbin/ipfwadm -F -a reject -W ppp0 -S 192.168.0.0/24 -D 204.50.10.13/32 -o

# Masquerade depuis le réseau local sur l'interface locale vers n'importe où
#
```

```
/sbin/ipfwadm -F -a masquerade -W ppp0 -S 192.168.0.0/24 -D 0.0.0.0/0
```

```
... fin des règles -F ...
```

Il n'y a pas besoin d'une règle spéciale pour autoriser 192.168.0.0/24 à se connecter sur 204.50.11.0, ce comportement est inclus dans les règles globales.

Contrairement a ipfwadm, il n'y a qu'une seule façon de spécifier l'interface dans ipchains, utilisez "-i interface".

6.6 MASQuer de multiples sous réseaux internes

MASQuer plus d'un sous réseau interne est assez simple. Vous n'avez qu'a vous assurer que tous vos réseaux fonctionnent bien (en interne et avec l'extérieur). Ensuite, vous aurez a autoriser le traffic a passer via vos interface réseaux vers Internet.

Ensuite, vous aurez a autoriser les autres interfaces internes, cet exemple montre comment masquer les interfaces eth1 (192.168.0.1) et eth2 (192.168.1.1) derriere eth0. Dans votre `rc.firewall` juste apres la ligne autorisant le MASQ, ajoutez les lignes :

– Noyaux 2.0 avec IPFWADM

```
# Permet aux interfaces internes de communiquer ensemble
/sbin/ipfwadm -F -a -V 192.168.0.1 -D 192.168.1.0/24
/sbin/ipfwadm -F -a -V 192.168.1.1 -D 192.168.0.0/24

# Permet aux interfaces internes d'etre MASQuées pour Internet
/sbin/ipfwadm -F -a masq -W eth0 -S 192.168.0.0/24 -D 0.0.0.0/0
/sbin/ipfwadm -F -a masq -W eth0 -S 192.168.1.0/24 -D 0.0.0.0/0
```

– Noyaux 2.2 avec IPCHAINS

```
#Enable internal interfaces to communication between each other
/sbin/ipchains -A forward -i eth1 -d 192.168.1.0/24
/sbin/ipchains -A forward -i eth2 -d 192.168.0.0/24

#Enable internal interfaces to MASQ out to the Internet
/sbin/ipchains -A forward -j MASQ -i eth0 -s 192.168.0.0/24 -d 0.0.0.0/0
/sbin/ipchains -A forward -j MASQ -i eth0 -s 192.168.1.0/24 -d 0.0.0.0/0
```

6.7 L'IP Masquerade et la numérotation à la demande.

1. Si vous voulez que votre réseau se connecte automatiquement à Internet, soit avec le package *diald* soit avec les dernieres versions du paquetage *PPPd*. Diald est la solution de numérotation à la demande recommandée.
2. Pour mettre en place *Diald*, veuillez vous référer à la page (en anglais) *Setting Up Diald for Linux Page* `<http://home.pacific.net.sg/~harish/diald.config.html>` ou *TrinityOS - Section 23* `<http://www.ecst.csuchico.edu/dranch/LINUX/TrinityOS.wri>`
3. Une fois que diald et IP masq auront été installés, vous pouvez aller sur n'importe laquelle des machines clients et initier une connexion web, telnet ou ftp et cela fera que votre machine Linux se connectera a Internet.
4. Un *timeout* (délai d'attente dépassé) sera inévitable sur la première connexion, mais c'est le lot des modems analogiques. Le temps mis à établir la connexion va provoquer un timeout de votre programme client. Ce n'est quand meme pas quelque chose qui arrive souvent. Il suffit de recommencer, et ça

passera. Vous pouvez aussi essayer de faire un `echo "1" > /proc/sys/net/ipv4/ip_dynaddr` pour aider la premiere utilisation.

6.8 IPPORTFW, IPMASQADM, IPAUTOFW, REDIR, UDPRED, et autres outils de redirection de ports

IPPORTFW, IPAUTOFW, REDIR, UDPRED, et autres sont des programmes génériques pour faire de la redirection de ports avec Linux IP MASQ. Ces outils sont normalement utilisés en remplacement de modules IP MASQ spécifiques comme les modules FTP ou Quake existant. Avec les redirecteurs de ports, vous pouvez rediriger des connexions arrivant d'Internet vers le réseau interne derriere le serveur MASQ. Cette redirection inclue des protocoles réseaux comme telnet, www, smtp, ftp (avec un patch, voir plus bas), ICQ, et plein d'autres.

NOTE : Si vous n'avez besoin que de la redirection de ports et pas besoin de l'IP MASQ, vous aurez **QUAND MÊME** à autoriser l'IP MASQ dans le noyau et avec soit IPFWADM soit IPCHAINS pour avoir la possibilité d'utiliser la redirection de ports. Plus tard, lorsque IP Masquerade est devenu mature, ces outils sont remplacés par IPPORTFW qui est une solution plus intelligente. Du fait de la présence d'outils plus récents, il est **HAUTEMENT RECOMMANDÉ** de ne pas utiliser les vieux outils comme IPAUTOFW et REDIT car ils ne font pas part au noyau de leur présence et leur présence peut même **PLANTER** votre serveur Linux si la charge monte.

Avant de se lancer corps et âme dans l'installation de soit IPPORTFW pour les 2.0 soit IPMASQADM avec le support IPPORTFW pour les 2.2, la sécurité du réseau peut être grandement compromise avec l'utilisation d'un tel outil. La raison est que ces outils créent un trou dans le firewall. Bien que cela ne pose pas de problèmes a votre machine Linux, il se peut que cela en soit un pour la machine interne vers lesquels sont transférés les paquets. Ne vous inquiétez pas trop quand même, Voici ce que Steven Clarke (l'auteur d'IPPORTFW) réponds a cela :

```
   "Le relayage de ports est seulement appelé depuis les fonctions de
masquerading, Par conséquent, il rentre dans les mêmes règles
d'IPFWADM/IPCHAINS. Le masquerading est une extension du relayage IP. Par
conséquent, ipportfw ne voit que les paquets qui rentrent dans les règles
d'entrée et de sortie."
```

Par conséquent, il est important d'avoir de bonnes regles de firewall. Allez lire les sections 6.4 () et 6.5 () pour plus de détails sur l'établissement de bonnes règles.

Donc, pour installer IPPORTFW, que ce soit pour les 2.0 ou 2.2, vous aurez a recompiler votre noyau avec le support pour IPPORTFW.

- Les utilisateurs des 2.0 auront a appliquer un patch aux sources de leur noyau (allez lire plus loin).
- Les utilisateurs des 2.2 ont déjà l'option IPPORTFW disponible via IPMASQADM.

6.8.1 IPPORTFW sur un noyau 2.0

Premièrement, assurez vous que vous avez le dernier noyau 2.0 décompressé dans `/usr/src/linux`. Si vous n'avez pas encore fait cela allez voir la section 3.1 () pour plus de détails. Ensuite, téléchargez le programme "`ipportfw.c`" et le "`subs-patch-x.gz`" depuis la section 2.5 () dans le répertoire `/usr/src`

NOTE : Remplacez le x de "`subs-patch-x.gz`" par le nom de la dernière version disponible sur le site.

Maintenant, copiez le patch IPPORTFW (`subs-patch-x.gz`) dans le repertorie Linux :

```
cp /usr/src/subs-patch-1.37.gz /usr/src/linux
```

Ensuite, appliquez le patch pour créer l'option IPPORTFW au noyau :

```
cd /usr/src/linux
zcat subs-patch-1.3x.gz | patch -p1
```

Ensuite, Si vous avez dans l'idée de faire du forwarding de trafic FTP vers un serveur interne, appliquez un **NOUVEAU** *IP_MASQ_FTP* patch de la section 2.5 (). Plus de détails sont donnés un peu plus loin.

Bon, maintenant, il est temps de compiler votre noyau comme il l'est expliqué dans la section 3.1 (). Dites bien oui a l'option IPPORTFW qui est maintenant disponible quand vous configurez votre noyau. Une fois que votre noyau est compilé et que vous avz rebooté, revenez ici.

Maintenant que vous avez un noyau tout neuf, compilez et installez le programme "`IPPORTFW`" :

```
cd /usr/src
gcc ipportfw.c -o ipportfw
mv ipportfw /usr/local/sbin
```

Maintenant, nous allons prendre l'exemple suivant : rediriger le trafic internet arrivant sur le port 80 d'être transféré a la machine masquée ayant l'adresse IP 192.168.0.10.

NOTE : Une fois que vous avez autorisé le relayage de ports sur le port 80, ce port ne peut plus être utilisé par le serveur MASQ. Pour être plus précis, si vous avez un serveur web qui tourne déjà sur le serveur MASQ et que vous avez mis le relayage en place, tous les utilisateurs d'Internet verront les pages web du serveur interne, et non celles du serveur MASQ. La seule parade a cela est de forwarder un autre port, disons par exemple le 8080, vers votre machine interne. Bien que cela marchera, les utilisateurs auront a ajouter un *:8080* a l'URL quand ils voudront contacter le serveur WWW.

Quoi qu'il en soit, pour autoriser la redirection de port, éditez le fichier `/etc/rc.d/rc.firewall`. Et ajoutez les lignes suivantes, assurez vous de bien remplacer le mot `$extip` par votre adresse internet.

NOTE : Si vous avez une IP dynamique, vous aurez a rendre votre `/etc/rc.d/rc.firewall` plus intelligent. Pour ce faire, référez vous aux sections 6.4 () et 6.5 () ou à *TrinityOS - Section 10* `<http://www.ecst.csuchico.edu/dranch/LINUX/TrinityOS.wri>` pour plus de détails sur les ip dynamiques.

```
/etc/rc.d/rc.firewall
--

#echo "Mise en place d'IPPORTFW vers le LAN externe."
#
/usr/local/sbin/ipportfw -C
/usr/local/sbin/ipportfw -A -t$extip/80 -R 192.168.0.10/80

--
```

Et voilà ! Relancez votre `/etc/rc.d/rc.firewall` et testez !

Si vous avez un message d'erreur comme "`ipfwadm : setsockopt failed : Protocol not available`", vous avez votre ancien noyau. Assurez vous que vous avez bien installé le nouveau noyau, lancez `lilo` et rebootez.

Redirection de ports pour les serveurs FTP :

Si vous avez dans l'idée d'utilise la redirection de ports, les choses se compliquent... La raison est que le module standard `ip_masq_ftp` du noyau n'a pas été écrit dans ce but. Heureusement, Fred Viles à modifié le module pour que ça marche. Si vous êtes juste curieux de savoir comment ça marche, téléchargez le patch suivant, Fred a mis plein de commentaires dans son code. Notez aussi que ce patch est légèrement expérimental et doit être traité comme tel. Il faut aussi noter que ce patch est disponible uniquement pour les noyaux 2.0. Le portage vers les noyaux 2.2 a été entrepris, si vous voulez aider a terminer ce port, envoyez un email directement à *Fred Viles - fv@episupport.com* (en anglais bien sur ;).

Donc, Pour faire marcher le patch, vous allez avoir à :

– Appliquer le patch IPPORTFW au noyau comme indiqué plus haut dans cette section.
– Télécharger le "`msqsrv-patch-36`" depuis le serveur ftp de Fred Viles (voir la section 2.5 ()) et mettez le dans `/usr/src/linux`.
– Patchez le noyau avec ce nouveau code en tapant : "`cat msqsrv-patch-36 | patch -p1`".
– Ensuite, remplacez le fichier "`ip_masq_ftp.c` du noyau par le nouveau :

```
mv /usr/src/linux/net/ipv4/ip_masq_ftp.c /usr/src/linux/net/ipv4/ip_masq_ftp.c.orig
mv /usr/src/linux/ip_masq_ftp.c /usr/src/linux/net/ipv4/ip_masq_ftp.c
```

– Enfin, compilez et installez le noyau avec le nouveau code en place.

Une fois cela fait, éditez le fichier `/etc/rc.d/rc.firewall` et ajoutez les lignes suivantes (remplacez bien $extip par votre ip externe) :

NOTE : Si vous avez une IP dynamique, vous aurez a rendre votre `/etc/rc.d/rc.firewall` plus intelligent. Pour ce faire, référez vous aux sections 6.4 () et 6.5 () ou à *TrinityOS - Section 10* `<http://www.ecst.csuchico.edu/dranch/LINUX/TrinityOS.wri>` pour plus de détails sur les ip dynamiques.

Cet exemple, comme celui ci dessus, permettra à tout le trafic ftp (sur le port 21) arrivant sur l'adresse IP internet d'être transféré a la machine interne ayant l'adresse IP 192.168.0.10.

NOTE : Une fois que vous avez autorisé le relayage du port 21, ce port ne peut plus être utilisé par le serveur MASQ. Pour être plus spécifique, si vous aviez déjà un serveur ftp sur le serveur MASQ, il ne sera plus accessible de l'extérieur.

```
/etc/rc.d/rc.firewall
--

#echo "Mise en place d'IPPORTFW vers le LAN externe."
#
/usr/local/sbin/ipportfw -C
/usr/local/sbin/ipportfw -A -t$extip/21 -R 192.168.0.10/21

--
```

Et voilà ! Relancez votre `/etc/rc.d/rc.firewall` et testez !

Si vous avez un message d'erreur comme "`ipfwadm : setsockopt failed : Protocol not available`", vous avez votre ancien noyau. Assurez vous que vous avez bien installé le nouveau noyau, lancez `lilo` et rebootez. Si vous êtes sur de bien utiliser votre nouveau noyau, tapez : "`ls /proc/net`" et vérifiez qu'il y a bien un "`ip_portfw`". Si il n'y est pas, vous avez du commettre une erreur dans la configuration du noyau. Essayez encore.

6.8.2 IPMASQADM avec IPPORTFW pour les noyaux 2.2

Tout d'abord, vérifiez que vous avez bien le dernier noyau 2.2 décompressé `/usr/src/linux`. Si vous ne l'avez pas encore fait, Allez voir la section 3.1 () pour tous les détails. Ensuite, téléchargez le programme "`ipmasqadm.c`" depuis la section 2.6 () dans le repertoire `/usr/src`.

Ensuite, vous aurez a compiler votre noyau comme il l'est expliqué dans la section 3.1 (). Vérifiez que vous avez bien dit YES a l'option IPPORTFW quand vous avez configuré votre noyau. Une fois la compilation terminée et que vous avez rebooté, revenez ici.

Maintenant, compilez et installez ipmasqadm :

```
cd /usr/src
tar xzvf ipmasqadm-x.tgz
cd ipmasqadm-x
make
make install
```

Maintenant, nous allons prendre l'exemple suivant : rediriger le trafic internet arrivant sur le port 80 d'être transféré a la machine masquée ayant l'adresse IP 192.168.0.10.

NOTE : Actuellement, le module `ip_masq_ftp` modifié ne marche pas pour les noyaux 2.2. Si vous vous sentez l'âme expérimentale, essayez de le porter et envoyez vos résultats a Ambrose et David.

NOTE : Une fois que vous avez autorisé le relayage de ports sur le port 80, ce port ne peut plus être utilisé par le serveur MASQ. Pour être plus précis, si vous avez un serveur web qui tourne déjà sur le serveur MASQ et que vous avez mis le relayage en place, tous les utilisateurs d'Internet verront les pages web du serveur interne, et non celles du serveur MASQ.

Quoi qu'il en soit, pour autoriser la redirection de port, éditez le fichier `/etc/rc.d/rc.firewall`. Et ajoutez les lignes suivantes, assurez vous de bien remplacer le mot `$extip` par votre adresse internet.

NOTE : Si vous avez une IP dynamique, vous aurez a rendre votre `/etc/rc.d/rc.firewall` plus intelligent. Pour ce faire, référez vous aux sections 6.4 () et 6.5 () ou à *TrinityOS - Section 10* <http://www.ecst.csuchico.edu/dranch/LINUX/TrinityOS.wri> pour plus de détails sur les ip dynamiques.

```
/etc/rc.d/rc.firewall
--

#echo "Mise en place d'IPPORTFW vers le LAN externe."
#
/usr/sbin/ipmasqadm portfw -f
/usr/sbin/ipmasqadm portfw -a -P tcp -L $extip 80 -R 192.168.0.10 80

--
```

Et voilà ! Relancez votre `/etc/rc.d/rc.firewall` et testez !

Si vous avez un message d'erreur comme "`ipfwadm : setsockopt failed : Protocol not available`", vous avez votre ancien noyau. Assurez vous que vous avez bien installé le nouveau noyau, lancez `lilo` et rebootez. Si vous êtes sur de bien utiliser votre nouveau noyau, tapez : "`ls /proc/net`" et vérifiez qu'il y a bien un "`ip_portfw`". Si il n'y est pas, vous avez du commettre une erreur dans la configuration du noyau. Essayez encore.

6.9 CU-SeeMe et Linux IP-Masquerade

L'ip masquerade supporte CuSeeMe via le module "`ip_masq_cuseeme`" du noyau. Ce module devrais être chargé depuis le script `/etc/rc.d/rc.firewall`. Une fois que le module est chargé, tous devriez etre capable d'établir et recevoir des connexions CuSeeMe vers des serveurs et/ou des utilisateurs.

NOTE : Il est recommandé d'utiliser IPPORTFW plutôt que IPAUTOFW pour utiliser CuSeeMe.

Si vous avez besoin de plus d'explications sur la configuration de CuSeeMe, allez voire *Michael Owings's CuSeeMe page* <http://www.swampgas.com/vc/ipmcus.htm> pour un Mini HOWTO ou *The IP Masquerade Resources* pour un miroir du mini HOWTO.

6.10 Mirabilis ICQ

Il y a deux méthodes pour faire marcher ICQ derrière un serveur MASQ. L'une de ces méthodes est d'utiliser le nouveau module ICQ MASQ, et l'autre est d'utiliser IPPORTFW.

Le module ICQ a quelques bénéfices et quelques limitations. Il permet d'avoir plusieurs utilisateurs ICQ derriere le serveur MASQ. Il ne requiert pas de changements dans la configuration des clients. Toutefois, il ne permet pas les transferts de fichiers et les chat en temps réel.

Avec IPPORTFW, Vous aurez a faire quelques changements a votre machine Linux et aux clients ICQ, mais tous les types de messages ICQ (urls, chat, transferts de fichiers) marcheront.

Si vous etes interessés par le module ICQ d'Andrew Deryabin's *djsf@usa.net* pour les noyaux 2.2, référez vous a la section 2.6 () pour plus de détails.

Si vous préférez utiliser la méthode classique pour faire marcher ICQ derriere votre serveur MASQ, suivez les étapes suivantes :

- Tout d'abord, vous avez besoin d'avoir l'IPPORTFW de compilé. Référez vous a la section 6.7 () pour plus de détails.
- Ensuite, vous aurez besoin d'ajouter les lignes suivantes a votre fichier `/etc/rc.d/rc.firewall`. Cet exemple suppose que 10.1.2.3 est votre adresse internet externe et que votre machine utilisant ICQ est 192.168.0.10 :
  - Voici ce qu'il faut mettre pour un noyau 2.0 avec IPFWADM ·

```
J'ai inclus deux exemples pour l'utilisateur, les deux marchent tres bien :

Exemple #1
--
/usr/local/sbin/ipportfw -A -t10.1.2.3/2000 -R 192.168.0.10/2000
/usr/local/sbin/ipportfw -A -t10.1.2.3/2001 -R 192.168.0.10/2001
/usr/local/sbin/ipportfw -A -t10.1.2.3/2002 -R 192.168.0.10/2002
/usr/local/sbin/ipportfw -A -t10.1.2.3/2003 -R 192.168.0.10/2003
/usr/local/sbin/ipportfw -A -t10.1.2.3/2004 -R 192.168.0.10/2004
/usr/local/sbin/ipportfw -A -t10.1.2.3/2005 -R 192.168.0.10/2005
/usr/local/sbin/ipportfw -A -t10.1.2.3/2006 -R 192.168.0.10/2006
/usr/local/sbin/ipportfw -A -t10.1.2.3/2007 -R 192.168.0.10/2007
/usr/local/sbin/ipportfw -A -t10.1.2.3/2008 -R 192.168.0.10/2008
/usr/local/sbin/ipportfw -A -t10.1.2.3/2009 -R 192.168.0.10/2009
/usr/local/sbin/ipportfw -A -t10.1.2.3/2010 -R 192.168.0.10/2010
/usr/local/sbin/ipportfw -A -t10.1.2.3/2011 -R 192.168.0.10/2011
/usr/local/sbin/ipportfw -A -t10.1.2.3/2012 -R 192.168.0.10/2012
/usr/local/sbin/ipportfw -A -t10.1.2.3/2013 -R 192.168.0.10/2013
/usr/local/sbin/ipportfw -A -t10.1.2.3/2014 -R 192.168.0.10/2014
/usr/local/sbin/ipportfw -A -t10.1.2.3/2015 -R 192.168.0.10/2015
/usr/local/sbin/ipportfw -A -t10.1.2.3/2016 -R 192.168.0.10/2016
/usr/local/sbin/ipportfw -A -t10.1.2.3/2017 -R 192.168.0.10/2017
/usr/local/sbin/ipportfw -A -t10.1.2.3/2018 -R 192.168.0.10/2018
/usr/local/sbin/ipportfw -A -t10.1.2.3/2019 -R 192.168.0.10/2019
/usr/local/sbin/ipportfw -A -t10.1.2.3/2020 -R 192.168.0.10/2020
--

Exemple #2
--
port=2000
while [ $port -lt 2020 ]
 do
```

```
   /usr/local/sbin/ipportfw -A t10.1.2.3/$port -R 192.168.0.10/$port
   port=$((port+1)
  done
--
```

– Et voila ce qu'il faut mettre pour un noyau 2.2 avec IPCHAINS :

```
J'ai inclus deux exemples pour l'utilisateur, les deux marchent tres bien :

Exemple #1
--
/usr/local/sbin/ipmasqadm portfw -a -P tcp -L 10.1.2.3 2000 -R 192.168.0.10 2000
/usr/local/sbin/ipmasqadm portfw -a -P tcp -L 10.1.2.3 2001 -R 192.168.0.10 2001
/usr/local/sbin/ipmasqadm portfw -a -P tcp -L 10.1.2.3 2002 -R 192.168.0.10 2002
/usr/local/sbin/ipmasqadm portfw -a -P tcp -L 10.1.2.3 2003 -R 192.168.0.10 2003
/usr/local/sbin/ipmasqadm portfw -a -P tcp -L 10.1.2.3 2004 -R 192.168.0.10 2004
/usr/local/sbin/ipmasqadm portfw -a -P tcp -L 10.1.2.3 2005 -R 192.168.0.10 2005
/usr/local/sbin/ipmasqadm portfw -a -P tcp -L 10.1.2.3 2006 -R 192.168.0.10 2006
/usr/local/sbin/ipmasqadm portfw -a -P tcp -L 10.1.2.3 2007 -R 192.168.0.10 2007
/usr/local/sbin/ipmasqadm portfw -a -P tcp -L 10.1.2.3 2008 -R 192.168.0.10 2008
/usr/local/sbin/ipmasqadm portfw -a -P tcp -L 10.1.2.3 2009 -R 192.168.0.10 2009
/usr/local/sbin/ipmasqadm portfw -a -P tcp -L 10.1.2.3 2010 -R 192.168.0.10 2010
/usr/local/sbin/ipmasqadm portfw -a -P tcp -L 10.1.2.3 2011 -R 192.168.0.10 2011
/usr/local/sbin/ipmasqadm portfw -a -P tcp -L 10.1.2.3 2012 -R 192.168.0.10 2012
/usr/local/sbin/ipmasqadm portfw -a -P tcp -L 10.1.2.3 2013 -R 192.168.0.10 2013
/usr/local/sbin/ipmasqadm portfw -a -P tcp -L 10.1.2.3 2014 -R 192.168.0.10 2014
/usr/local/sbin/ipmasqadm portfw -a -P tcp -L 10.1.2.3 2015 -R 192.168.0.10 2015
/usr/local/sbin/ipmasqadm portfw -a -P tcp -L 10.1.2.3 2016 -R 192.168.0.10 2016
/usr/local/sbin/ipmasqadm portfw -a -P tcp -L 10.1.2.3 2017 -R 192.168.0.10 2017
/usr/local/sbin/ipmasqadm portfw -a -P tcp -L 10.1.2.3 2018 -R 192.168.0.10 2018
/usr/local/sbin/ipmasqadm portfw -a -P tcp -L 10.1.2.3 2019 -R 192.168.0.10 2019
/usr/local/sbin/ipmasqadm portfw -a -P tcp -L 10.1.2.3 2020 -R 192.168.0.10 2020
--

Exemple #2
--
port=2000
while [ $port -lt 2020 ]
 do
  /usr/local/sbin/ipmasqadm portfw -a -P tcp -L 10.1.2.3 $port -R 192.168.0.10 $port
  port=$((port+1)
 done
--
```

– Une fois que votre nouveau `rc.firewall` est pret, rechargez le en tapant "`/etc/rc.d/rc.firewall`". Si vous avez des erreurs, soit vous n'avez pas IPPORTFW dans votre noyau, soit vous avez faite une faute de frappe dans le script.

– Maintenant, il est temps d'aller configurer ICQ, allez dans les préférences –> connection, configurez pour avoir *'I am using a permanent internet connection (LAN)'* et *'I am behind a firewall or proxy'*, maintenant, cliquez sur *'Firewall Settings'* et choisissez *'I don't use a SOCKS Proxy server...'*. Notez aussi qu'il étais recommandé de mettre 30 secondes dans *'Firewall session time out after'* MAIS beaucoup d'utilisateurs ont trouvé qu'ICQ devenait peu fiable. ICQ est bien plus fiable avec l'option par défaut (ne cochez donc pas cette option) et changez le timeout du serveur MASQ a 160 secondes. Vous pouvez voir comment changer ce timeout dans les sections 3.3 () et 3.3.1 (). Finalement, cliquez sur *'Next'* et configurez ICQ pour *'Use the following TCP listen ports...'* de 2000 à 2020. Cliquez sur *'Done'*.
Maintenant, ICQ vous dit que vous avez a redémarrer ICQ pour que les changements soit pris en compte.

Pour être honnête j'ai du redémarrer la machine Windows 9x pour que ça marche, mais d'autre personnes disent que cela n'a pas été la peine. Donc, essayez les deux...

- Un utilisateur m'a aussi rapporté que le fait de rediriger le port 4000 vers sa machine ICQ marche sans problèmes. Il a dit que tout fonctionnais (chat, transferts de fichiers, etc) SANS AVOIR a reconfigurer ICQ. Il se peut que ça marche, c'est pour cela que je vous en fait part.

6.11 Joueurs : Le patch LooseUDP

Le patch LooseUDP permet aux jeux comprenant le NAT qui utilisent des connexions UDP de marcher et même bien derrière un serveur MASQ. Actuellement, LooseUDP est disponible comme patch pour les noyaux 2.0.36+ et est intégré dans les noyaux 2.2.3+. Pour le faire marcher, quelques petites choses sont nécessaires :

- Ayez la dernière version des source du noyau décompressées dans `/usr/src/linux`.
- Pour les 2.0, il est NÉCESSAIRE d'installer IPPORTFW (voir la section 2.5 ()) comme expliqué dans la section 6.7 ().
- Téléchargez le patch LooseUDP depuis la section 2.5 ().

Maintenant, mettez le patch LooseUDP dans le répertoire `/usr/src/linux`. Quand c'est fait, tapez :

```
Pour un patch compressé : zcat loose-udp-2.0.36.patch.gz | patch -p1

Pour un patch non compressé : cat loose-udp-2.0.36.patch | patch -p1
```

Maintenant, suivant la version du "patch", vous verrez le texte suivant :

```
patching file ‘CREDITS’
patching file ‘Documentation/Configure.help’
patching file ‘include/net/ip_masq.h’
patching file ‘net/ipv4/Config.in’
patching file ‘net/ipv4/ip_masq.c’
```

Si vous voyez le texte "Hunk FAILED" une fois et une seule au début du patch, ne vous inquiétez pas, vous avez probablement une vieille version du patch (ça à été réparé depuis) mais ça marche quand même. Si ça plante complètement, vérifiez que vous avez appliqué le patch IPPORTFW en PREMIER.

Une fois que le patch est installé, reconfigurer le noyau comme c'est expliqué dans la section 3.1 () et dites bien YES à l'option "IP : loose UDP port managing (EXPERIMENTAL) (CONFIG_IP_MASQ_LOOSE_UDP) [Y/n/ ?]".

Une fois que vous tournez avec le patch LooseUDP, vous devriez pouvoir jouer a un bon paquet de jeux. Quelques URL se trouvent dans la section 6.3.1 () pour savoir comment pouvoir utiliser des jeux comme BattleZone et d'autre.

7 Foire Aux Questions

Si vous avez des suggestions a faire, envoyez les à *ambrose@writeme.com* et *dranch@trinnet.net* (NdT : en anglais). S'il vous plait, poser clairement la question et la réponse appropriée. Merci !

7.1 Quelles sont les distributions de Linux qui sont livrées avec les fonctionnalités d'IP MASQ a la base ?

Si votre distribution de Linux ne supporte pas l'IP MASQ a la base, ne vous inquiétez pas. Tout ce que vous aurez a faire est de recompiler votre noyau comme il l'est expliqué plus haut dans ce HOWTO a la section

3.1 ().

NOTE : Si vous pouvez nous aider a remplir cette table, envoyez un email à *ambrose@writeme.com* ou *dranch@trinnet.net* (NdT : en anglais).

```
Caldera       < v1.2 : NON - ?
Caldera         v1.3 : OUI - basé sur un 2.0.35
Caldera         v2.2 : OUI - basé sur un 2.2.5
Debian          v1.3 : NON - ?
Debian          v2.0 : NON - ?
Debian          v2.1 : NON - basé sur un 2.2.1
DLX Linux       v?   :  ?  - ?
DOS Linux       v?   :  ?  - ?
Hal91 Linux     v?   :  ?  - ?
Linux Mandrake  v5.3 : OUI - ?
Linux Mandrake  v6.0 : OUI - basé sur un 2.2.5
Linux PPC       vR4  : NON - ?
Linux Pro       v?   :  ?  - ?
LinuxWare       v?   :  ?  - ?
MkLinux         v?   :  ?  - ?
MuLinux         v3rl : OUI - ?
Redhat        < v4.x : NON - ?
Redhat          v5.0 : OUI - ?
Redhat          v5.1 : OUI - ?
Redhat          v5.2 : OUI - basé sur un 2.0.36
Redhat          v6.0 : OUI - basé sur un 2.2.5
Slackware       v3.0 :  ?  - ?
Slackware       v3.1 :  ?  - basé sur un 2.0.29
Slackware       v3.2 :  ?  - ?
Slackware       v3.3 :  ?  - basé sur un 2.0.34
Slackware       v3.4 :  ?  - ?
Slackware       v3.5 :  ?  - ?
Slackware       v3.6 :  ?  - ?
Slackware       v3.9 :  ?  - basé sur un 2.0.37pre10
Slackware       v4.0 :  ?  - ?
Stampede Linux  v?   :  ?  - ?
SuSE            v5.2 : OUI - ?
SuSE            v5.3 : OUI - ?
SuSE            v6.0 : OUI - ?
SuSE            v6.1 : OUI - basé sur un 2.2.5
Tomsrbt Linux   v?   :  ?  - ?
TriLinux        v?   :  ?  - ?
TurboLinux      v?   :  ?  - ?
Yggdrasil Linux v?   :  ?  - ?
```

7.2 Quel est le minimum requis et quelles sont les limitations de l'IP MASQ ? Est ce que ça marche bien ?

Un 486/66 avec 16 Mo de RAM, étais largement suffisant pour masquerader les 1.54 Mo/s d'une T1 ! Le MASQ est aussi connu pour marcher très bien sur un 386SX-16s avec 8 Mo de RAM. Bon, il faut aussi savoir que l'IP Masq commence a déconner a partir de 500 connexions MASQ.

La seule application qui est connue pour bloquer temporairement une machine Linux est GameSpy. Pourquoi ? Quand il raffraichis ses listes, il crée 10000 petites connexions dans très peu de temps. Jusqu'à ce que les sessions timeout, la table des connexion MASQ est pleine. Allez visiter la section 7.17 () pour plus de détails.

Tant qu'on y est :

Il y a une limite en dur dans le noyau à 4096 connexions TCP ou UDP. Cette limite peut être changée en changeant les valeurs dans `/usr/src/linux/net/ipv4/ip_masq.h`. Jusqu'à 32000, ça doit être bon. Si vous voulez changer cette limite, vous aurez besoin de changer les valeurs de PORT_MASQ_BEGIN & PORT_MASQ_END pour avoir un espace suffisant au dessus de 32K et en dessous de 64K.

7.3 J'ai vérifié toute ma configuration, et je n'arrive toujours pas a faire marcher l'IP MASQ. Que faire ?

- Restez calme. Allez vous chercher une tasse de thé, café, soda, etc., et allez faire une sieste. Une fois que votre esprit est plus clair, essayez les suggestions ci dessous. La mise en place de l'ip masquerading n'est PAS une chose difficile, mais il y a pas mal de concepts qui sont nouveaux pour vous.
- Encore une fois, refaites toutes les étapes de la section 5 (). 99% du temps, vous n'êtes même pas allés lire cette section.
- Allez lire *IP Masquerade Mailing List Archives* `<http://www.indyramp.com/lists/masq/>`, il est tres probable que la réponse a votre question trouvera réponse en faisant une simple recherche dans les archives.
- Allez parcourir le document *TrinityOS* `<http://www.ecst.csuchico.edu/dranch/LINUX/TrinityOS.wri>`. Il couvre l'IP MASQ pour les noyaux 2.0 et 2.2 et beaucoup d'autres sujets comme PPPd, DialD, DHCP, DNS, Sendmail, etc.
- Assurez vous que vous n'avez pas lancés les démons ROUTED ou GATED. Pour vérifiez, tapez "`ps aux | grep -e routed -e gated`""
- Posez votre question sur la liste IP MASQ (allez voir la section suivante de la FAQ pour plus de détails). S'il vous plaît, ne posez votre question sur la liste que si vous n'avez pas trouvé la réponse dans les archives. Assurez vous de bien inclure toutes les informations requises dans la section 5 () dans votre email !
- Posez votre question sur un newsgroup en rapport avec Linux.
- Envoyez un email à *ambrose@writeme.com* et/ou *dranch@trinnet.net*. Mais vous aurez plus de chance d'obtenir une réponse venant de la liste IP MASQ que de nous.
- Vérifiez votre configuration une fois de plus :-)

7.4 Comment est ce que je m'inscrit a la liste IP MASQ et/ou a la liste IP MASQ developers, et ou puis-je trouver leurs archives ?

Il y a deux moyens de s'inscrire aux deux listes IP MASQ. La première est d'envoyer un email à *masq-request@indyramp.com*. Pour vous inscrire a la liste IP MASQ developers, envoyez un email à *masq-dev-request@indyramp.com*. Voyez ci dessous pour plus de détails.

- Inscription par email : Mettez le mot "`subscribe`" soit dans le sujet soit dans le corps du message. Si vous voulez juste vous inscrire à la liste en mode digest (tous les emails vous sont envoyés chaque semaine en une seule fois dans un gros message), mettez les mots "`subscribe digest`" dans le sujet ou le corps.
- Une fois que le serveur a reçu votre inscription, il vous inscrira a la liste en question et vous enverra un mot de passe. Gardez ce mot de passe car vous en aurez besoin plus tard pour vous désinscrire ou changer vos options.

La deuxième méthode est de passer via un navigateur web et de s'inscrire depuis le formulaire *http ://www.indyramp.com/masq-list/* pour la liste principale ou sur *http ://www.indyramp.com/masq-dev-list/* pour la liste developers.

Une fois inscrit, vous recevrez les mails des listes ou vous vous êtes inscrits. Notez aussi qu'il n'est pas nécessaire d'être inscrit pour consulter les archives des deux listes. Pour ce faire, allez visiter les adresses ci dessus.

Enfin, notez que vous ne pouvez poster des messages que depuis des adresses depuis lesquelles vous êtes inscrits.

Si vous avez un quelconque problème en rapport avec les listes de diffusions, contactez *Robert Novak*.

7.5 En quoi est ce que IP MASQ diffère des services NAT/Proxy ?

```
Proxy:  Serveurs proxy sont disponibles pour : Win95, NT, Linux, Solaris, etc.

          Pour :   + (1) adresse IP ; économique
                   + Possibilité de cache pour de meilleurs performances (WWW, etc.)

          Contre : - Toutes les applications utilisées derrière le proxy doivent
                     supporter les services SOCKS et être configurés pour utiliser
                     les services du Proxy.
                   - Explose les statistiques et compteurs Web

        Un serveur proxy utilise seulement une adresse IP, comme IP MASQ, et sert de
        passerelle entre les clients sur le réseau privé (navigateur Web, etc.). Ce
        proxy reçoit des requêtes comme TELNET, FTP, WWW, etc. du réseau interne sur
        une interface. Il initie a son tour une requête comme si quelqu'un sur le
        proxy faisait la requête. Une fois que le serveur distant a répondu, il
        transfert les adresses TCP/IP au client MASQ interne et envoie la requête a
        la machine interne. C'est pour cela que ça s'appelle un serveur PROXY.

           Note : Toutes les applications que vous voudrez utiliser sur le réseau
                  interne doivent supporter un serveur proxy comme Netscape et
                  quelques bons clients TELNET et FTP. Ceux qui ne supportent pas un
                  serveur proxy ne fonctionneront pas.

        Une autre chose sympa a propos des serveurs proxy est que certains d'entre
        eux cachent aussi leurs données (Squid pour le WWW). Donc, imaginons que
        vous avez 50 clients qui lancent Netscape en même temps, si ils ont toujours
        la page par défaut de configurée, vous auriez 50 copies de la même page
        chargée 50 fois via la connexion internet. Avec un proxy cache, seule une
        copie de la page serais téléchargée via la connexion internet et elle serais
        ensuite distribuée a la demande aux machines internes. Non seulement ca
        sauvegarde la bande passante de la connexion internet mais en plus, les
        machines internes ont un acces beaucoup plus rapide au net.



MASQ:   Le masquerading est disponible sous Linux, et certains routeurs RNIS tels
 OU     les Zytel Prestige128, Cisco 770, NetGear ISDN routers, etc.
1:+
 NAT
```

```
  Pour :   + Seule une adresse IP est necessaire (économique)
           + Ne requiert pas de support dans les applications.
           + Utilise des logiciels de firewall, ce qui permet de rendre
             votre réseau plus sur.

  Contre : - Requiert une machine Linux ou un routeur RNIS dédié (d'autres
             produits peuvent l'avoir toutefois...)
           - Le trafic entrant ne peut pas accéder a votre réseau interne, a
             moins que votre LAN n'initie le trafic, ou qu'un logiciel de
             redirection de ports soit installé. Beaucoup de serveurs NAT ne
             savent pas le faire.
           - Certains protocoles doivent être pris en charge uniquement via
             des redirecteurs de firewall, etc. Linux supporte tout cela
             (FTP, IRC, etc.) mais beaucoup de routeurs ne le font PAS
             (NetGear le fait).

MASQ ou 1:+ NAT est similaire a un serveur proxy dans le sens que le serveur
fera de la translation d'adresse IP pour tromper les serveurs distants (WWW
par exemple) comme si le serveur MASQ avait fait la requête et non pas une
machine interne.

La différence majeure entre le MASQ et un PROXY est que le serveur MASQ n'a
pas besoin de changer la configuration des machines clientes. Configurez les
juste pour utiliser la machine Linux comme étant leur passerelle par défaut
et tout marchera sans problemes. Vous aurez besoin d'installer des modules
Linux spéciaux sur le serveur pour certains services (RealAudio, FTP,
etc. pour que cela marche) !

Beaucoup de gens utilisent un serveur MASQ pour faire du TELNET, du FTP,
etc. *ET* mettent aussi en place un serveur proxy cache pour le trafic WWW
pour augmenter les performances.


NAT:    Des serveurs NAT sont disponibles pour Windows 95/98/NT/2000, Linux,
        Solaris, et certains bons routeurs RNIS (pas les Ascend).

  Pour :   + Très configurable
           + Pas de logiciels vraiment nécessaires.

  Contre : - Requiert un sous réseau de votre FAI (cher)

Le NAT (translation d'adresses réseau) est le nom d'une machine ayant a sa
disposition un certain nombre d'adresses IP valides qu'il peut
utiliser. Lorsqu'une connexion vers l'extérieur est requise par un
utilisateur interne, cette machine associe une adresse IP valide disponible
a l'IP privée qui requiert la connexion. Apres ca, tout le traffic est
réécrit de l'adresse IP NAT publique pour l'adresse IP NAT
privée. Lorsqu'une adresse NAT publique reste sans être utilisée pendant un
certain laps de temps, l'adresse IP publique retourne dans le groupe des IP
publiques disponibles.
```

```
Le problème principal du NAT est qu'une fois que toutes les adresses IP
publiques sont utilisées, ceux qui veulent obtenir une connexion ensuite ne
le pourront pas tant qu'une adresse ne se libère pas.
```

7.6 Existe-t-il des outils de configuration de firewall graphiques ?

Oui ! Il y a un peu de tout. Mais la majorité est faite pour marcher avec IPFWADM. Voici une courte liste des outils disponibles par ordre alphabétique. Si vous n'en connaissez pas d'autres, ou que vous avez des remarques (bonnes/mauvaise/méchantes) sur ceux qui sont ici, envoyez un email a Ambrose ou David.

- John Hardin's *IPFWADM Dot file generator* - Une version IPCHAINS est en cours.
- Sonny Parlin's *FWCONFIG* `<http://www.mindstorm.com/sparlin/fwconfig.shtml>` pour IPFWADM et IPCHAINS
- William Stearns's *Mason* `<http://www.pobox.com/wstearns/mason/>` - Un système de création de règles a la volée.

7.7 Est-ce que IP Masquerade marche avec une IP dynamique ?

Oui, bien sur que cela marche avec une adresse IP dynamique assignée par votre FAI, d'habitude, via un serveur DHCP. Aussi longtemps que vous avez une adresse Internet valide, cela devrait marcher. Bien entendu, les IP statiques conviennent aussi.

7.8 Puis-je utiliser des modems cable, DSL, liaison satellite, etc pour me connecter à Internet en utilisant IP masquerading ?

Bien sûr, tant que Linux supporte l'interface réseau, cela marchera.

7.9 Puis-je utiliser DialD ou la fonctionnalité de connexion à la demande de PPPd avec IP MASQ ?

Bien sur ! IP MASQ est totalement transparent à DialD ou PPP. La seule chose qui peut poser problème si vous avez des règles de firewall méchantes et que vous avez une adresse dynamique. Référez vous a l'item de la FAQ "Est-ce que IP Masquerade marche avec une IP dynamique ?" ci dessus pour plus de détails.

7.10 Quels sont les programmes qui fonctionnent avec l'IP Masquerade ?

Il est très difficile d'avoir une liste exhaustive des "programmes qui marchent". Toutefois, la majorité des applications internet sont supportées, telles que surfer sur Internet (Netscape, MSIE, etc.), ftp (comme WS_FTP), Real Audio, telnet, SSH, POP3 (courrier entrant - pine, Outlook), SMTP (courrier sortant), etc.

Les programmes qui impliquent des protocoles plus compliqués ou des méthodes de connexion spéciales necessitent des outils d'aide spéciaux.

Pour plus de détails, référez vous à cette page : *programmes qui marchent avec le Linux IP masquerading* `<http://dijon.nais.com/~nevo/masq/>` par Lee Nevo.

7.11 Comment puis-je faire marcher l'IP Masquerading sur une RedHat, une Debian, une Slackware, etc ?

La distribution de Linux que vous avez, les procédures pour mettre en place l'IP masquerading mentionnées dans ce HOWTO devraient suffire. Certaines distributions auront peut être des programmes graphiques ou des fichiers de configuration spéciaux pour rendre les réglages plus simples. Nous faisons en sorte de rendre ce HowTo aussi simple que possible.

7.12 Mes connexions TELNET semblent mourrir si je ne les utilise pas pendant quelques temps. Pourquoi ?

Par défaut, IP MASQ à un time out de connexion pour les sessions TCP, les TCP FIN et le traffic UDP qui est a 15 minutes. Il est recommendé d'utiliser les valeurs suivantes (comme elles le sont dans le fichier `/etc/rc.d/rc.firewall` décrit dans ce HOWTO) :

Linux 2.0.x avec IPFWADM :

```
# Durée de vie des connexion MASQuées
#
#    2 hrs pour une session TCP
#   10 sec apres qu'un paquet "FIN" soit passé
#  160 sec pour le traffic UDP (Important pour les utilisateurs d'ICQ)
#
/sbin/ipfwadm -M -s 7200 10 160
```

Linux 2.2.x avec IPCHAINS :

```
# Durée de vie des connexion MASQuées
#
#    2 hrs pour une session TCP
#   10 sec apres qu'un paquet "FIN" soit passé
#   60 sec pour le traffic UDP (Les utilisateurs d'ICQ doivent configurer leur ICQ avec
#                               un timeout de firewall de 30 secondes)
#
ipchains -M -S 7200 10 60
```

7.13 Lorsque ma connexion Internet arrive, rien ne marche, ensuite, si je recommence, tout marche. Pourquoi ?

La raison est que vous utilisez une adresse dynamique et que lorsque votre connexion internet arrive, IP Masquerade ne connaît pas son adresse IP. Il existe une solution. Dans votre jeu de règles `/etc/rc.d/rc.firewall`, ajoutez la ligne suivante ;

```
# Utilisateurs d'IP dynamiques :
#
# Si vous avez une adresse IP dynamique via SLIP, PPP ou DHCP, décommentez l'option
# suivante. Elle permet d'utiliser une bidouille dans l'IP MASQ qui sert pour les IP
# dynamiques, ce qui rends la vie avec DialD, PPPd et les programmes similaires bien
# plus facile.
#
echo "1" > /proc/sys/net/ipv4/ip_dynaddr
```

7.14 L'IP MASQ semble fonctionner comme un charme, mais certains sites ne marchent pas. Cela se produit le plus souvent avec des sites WWW ou FTP.

Il y a deux raisons possibles à cela. La première est très commune, le deuxième, très peu.

- A partir des noyaux 2.0.36 et 2.2.9, il y a un BUG dans le code MASQ qui a qui a à voir avec les paquets qui ont le bit DF ou "Don't Fragment" (ne pas découper) de positionné. En gros, lorsque la machine MASQ se connecte à internet avec un MTU inférieur a 1500, quelques paquets auront le bit DF de positionné. Bien que changer le MTU de votre Linux à 1500 semblera résoudre le problème, le bug est toujours là. Le problème semble venir du fait que le code MASQ ne réécrit pas correctement les paquets ICMP avec le code ICMP 3.4 revenant a l'ordinateur MASQ à la source. A cause de cela, le paquet est détruit. Si vous êtes un programmeur réseau et que vous pensez pouvoir le faire... ESSAYEZ !
  Ne vous inquietez pas trop quand même. Pour contourner ce bug, changez le MTU de votre lien Internet à 1500. Certains utilisateurs crieront car cela peut créer quelques latences avec des programmes comme TELNET ou des jeux, mais l'impact est vraiment faible. D'un autre coté, la majorité du trafic HTTP et FTP deviendra plus rapide !
  Pour réparer cela, regardez quel est le MTU de votre lien Internet. Pour le savoir, lancez "`/sbin/ifconfig`". Maintenant, regardez les lignes correspondant a votre connexion Internet et regardez le MTU. Il doit être a 1500. Pour l'Ethernet, c'est la valeur par défaut, mais sur un lien PPP, la valeur par défaut est de 576.
- Pour règler ce probleme de MTU sur votre lien PPP, éditez votre fichier `/etc/ppp/options` et mettez y les lignes "`mtu 1500`" et "`mru 1500`". Sauvez et relancez votre lien PPP. Comme ci dessus, vérifiez que votre lien PPP à bien les valeurs correctes pour MTU et MRU.
- Pour réparer un probleme de MTU sur un lien Ethernet vers votre modem ADSL, Cable, etc, vous allez devoir éditer vos scripts de démarrage. Allez consulter *TrinityOS - Section 16* `<http://www.ecst.csuchico.edu/dranch/LINUX/TrinityOS.wri>` pour plus d'informations.
- Enfin, bien que ce ne soit pas un probleme tres commun, certaines personnes ont trouvé leur solution. avec les utilisateurs PPP, sur quel port se connecte PPPd ? À un port `/dev/cua*` ou un `/dev/ttyS*` ? Il doit se connecter sur un port `/dev/ttyS*`. Les vieux cua font se comporter les connexions de manières bizarres.

7.15 IP Masquerading semble être lent.

Il peut y avoir quelques raisons a cela :

- Assurez vous que vous n'avez pas les réseaux internes et externes lancés sur la même carte réseau avec de l'IP alias. Si vous le faites, vous devriez vraiment aller dépenser 100F dans une deuxième carte réseau pour que chaque réseau ait la sienne.
- Si vous avez un modem externe, assurez vous d'avoir un bon câble. Beaucoup de PC ont ces cables plats qui connectent le port série a la carte mère ou la carte d'E/S. Si vous avez un câble comme cela, assurez vous qu'il est dans de bonnes conditions. J'ai personnellement un anneau de ferrite (ces espèces anneaux en métal gris/noir) autour de tous mes cables plats.
- Assurez vous que votre MTU est a 1500 comme il l'est décrit ci dessus.
- Assurez vous que votre port série est à un UARL 16550A ou mieux. Lancez "`dmesg | less`" pour vérifier.
- Assurez vous que le port série sur lequel votre connexion tourne est a 115200 (ou plus rapide si votre modem et votre port série le supportent... comme les adaptateurs RNIS).
  - Noyaux 2.0 : Ces noyaux sont un peu vieux et ne peuvent pas obliger le noyau a passer le port série à 115200. Par conséquent, dans l'un de vos scripts de démarrage comme `/etc/rc.d/rc.local` ou `/etc/rc.d/rc.serial`, lancez les commandes suivantes si votre modem se trouve sur le COM2 :
    - `setserial /dev/ttyS1 spd_vhi`
    - dans votre script PPPd, éditez la ligne pour inclure la vitesse de 38400 d'après la page man de pppd.

  - Noyaux 2.2 : Contrairement aux 2.0, les 2.1 et 2.2 n'ont pas besoin d'utiliser setserial.
    - Donc, dans votre script PPPd, éditez la ligne lançant pppd et ajoutez la vitesse de 115200.
- Réglez la fenêtre de réception TCP à au moins 8192.
  - Bien que cela soit COMPLÈTEMENT hors du sujet de ce document, cela peut aider un petit peu dans certaines conditions. Pour plus de détails, référez vous a la section optimisation des réseaux de *TrinityOS - Section 16*.
- Réglez IRQ-Tune pour vos ports séries.
  - Sur la majorité du matériel PC, l'utilisation de *IRQTUNE* `<http://www.best.com/cae/irqtune/>` de Craig Estey's accroît de manière significative les performances des ports pour les connexions PPP et SLIP.

7.16 Maintenant que j'ai l'IP MASQ qui marche, j'ai plein de notifications et d'erreurs bizarres dans mon SYSLOG. Comment comprendre les erreurs IPFWADM/IPCHAINS dans le texte ?

Il y a probablement deux choses normales que vous allez voire :

- **MASQ : Failed TCP Checksum error :** Vous verrez cette erreur lorsqu'un paquet venant de l'Ethernet a sa section de données corrompue alors que le reste du paquet semble bon. Lorsque votre machine Linux reçoit ce paquet, il calcule le CRC du paquet et détermine qu'il est corrompu. Sur la plupart des machines tournant sur des OS comme Microsoft Windows, le paquet est juste mis silencieusement a la poubelle, la seule différence est que Linux le dit. Si vous avez beaucoup de paquets de ce genre a travers votre lien PPP, allez déjà lire l'entrée précédente de la FAQ sur les lenteurs.
- Si toutes les astuces ne servent a rien, essayez d'ajouter la ligne `"-vj"` a votre `/etc/ppp/options` et redémarrez PPPd.
- **Firewall hits** : Vous êtes connectés a Internet avec un firewall décent, vous allez voir le nombre de gens qui tentent de pénétrer votre machine Linux ! Alors, qu'est ce que ces logs signifient ?
  D'après *TrinityOS - Section 10* `<http://www.ecst.csuchico.edu/dranch/LINUX/TrinityOS.wri>` :

```
Dans les règles suivantes, les lignes ayant soit DENY soit REJECT ont aussi un
"-o" pour dire au firewall d'envoyer les messages au SYSLOG. Tout cela est
stocké soit dans :

        Redhat:         /var/log
        Slackware:      /var/adm

Si vous regardez a vos logs de firewall, vous verrez des choses comme cela~:

----------------------------------------------------------------------
IPFWADM:
Feb 23 07:37:01 Roadrunner kernel: IP fw-in rej eth0 TCP 12.75.147.174:1633
        100.200.0.212:23 L=44 S=0x00 I=54054 F=0x0040 T=254

IPCHAINS:
Packet log: input DENY eth0 PROTO=6 12.75.147.174:1633 100.200.0.212:23
        L=44 S=0x00 I=54054 F=0x0040 T=254
----------------------------------------------------------------------

Il y a beaucoup d'informations sur ces lignes. Découpons donc tout cela pour
que vous puissiez comprendre ce qui s'est passé sur votre machine. Notez que
cet exemple explique pour IPFWADM, bien qu'il soit compréhensible aux
utilisateurs d'IPCHAINS.
```

```
--------------

- Ce "hit" du firewall est arrivé le "Feb 23 07:37:01".

- Sur l'ordinateur "RoadRunner".

- Il est arrivé sur le protocole IP ou TCP/IP.

- Il est arrivé dans ("fw-in") le firewall
        * d'autres logs pourraient dire "fw-out" pour la sortie
          ou "fw-fwd" pour le forwarding

- Ensuite, le hit a été "rejETÉ".
        * d'autres logs pourraient dire "deny" ou "accept"

- Ce hit du firewall est arrivé sur l'interface "eth0" (lien Internet).

- C'était un paquet "TCP".

- Ce hit est arrivé de l'adresse IP "12.75.147.174" avec comme port de retour
  "1633".

- Ce hit étais adressé a l'IP "100.200.0.212" sur le port "23" ou TELNET.
        * Si vous ne saviez pas que le port 23 correspondait au TELNET, allez
          jeter un coup d'oeil au fichier /etc/services.

- Ce paquet faisait "44" octets de long.

- Ce paquet n'avais aucun TOS (type de service)
        --Ne vous inquiétez pas si vous ne comprenez pas ce que cela signifie...
          vous n'avez pas envie de savoir.
        * pour IPCHAINS, il faut diviser ce nombre par 4.

- Ce paquet avec comme "IP ID" "18".
        --Ne vous inquiétez pas si vous ne comprenez pas ce que cela signifie...
          vous n'avez pas envie de savoir.

- Ce paquet avec avais un fragment de décalage de 16 bits incluant tout paquet
  TCP/IP avec comme drapeau "0x0000".
        --Ne vous inquiétez pas si vous ne comprenez pas ce que cela signifie...
          vous n'avez pas envie de savoir.
        * Une valeur commençant par "0x2..." ou "0x3..." aurais signifié que le
          drapeau "Autres Fragments" était positionné et que d'autres morceaux
          de paquets étaient a venir pour compléter cet ÉNORME paquet.
        * Une valeur de "0x4..." et "0x5..." signifie que le drapeau "Ne pas
          fragmenter" est positionné.
        * Toutes les autres valeurs est le décalage du fragment (divisé par 8)
          pour être réassemblé plus tard.
```

```
- Ce paquet avais un TTL de 20.
        * Chaque saut au travers d'Internet soustrait 1 de ce nombre. En
          général, les paquets commencent leur vie avec un TTL de 255 et si
          jamais ce nombre arrive a zéro, cela signifie que ce paquet a été
          perdu et qu'il sera supprimé.
```

7.17 Puis-je configurer IP MASQ pour permettre a des utilisateurs externes de contacter directement des machines masquées ?

Bien sur ! avec IPPORTFW, vous pouvez autoriser TOUT ou juste certaines machines sur Internet a contacter toutes vos machines masquées. **Ce sujet est développé dans la section 6.7 () de ce HOWTO.**

7.18 J'ai des "kernel : ip_masq_new(proto=UDP) : no free ports." qui apparaissent dans mon SYSLOG. Que se passe-t-il ?

L'une de vos machines internes crée un nombre anormalement haut de paquets a destination d'Internet. Comme l'IP MASQ construit une table et transmet ces paquets sur Internet, la table se remplis vite. Une fois cette table pleine, vous aurez droit à cette erreur.

La seule application que je connaisse et qui crée cette situation temporairement est un programme de jeu appelé "GameSpy". Pourquoi ? Car il récupère une liste de serveurs et ensuite, ping chaque serveur de la liste (environ 1000 serveurs). En créant tous ces pings, il crée une dizaine de milliers de connexions dans un très court laps de temps. Jusqu'à ce que les sessions expirent grâce aux timeout de l'IP MASQ, la table MASQ est pleine.

Que faire alors ? Franchement, n'utilisez pas de programme comme cela. Si vous avez cette erreur dans vos logs, trouvez le programme, et stoppez le. Si vous aimez beaucoup GameSpy, ne rafraîchissez pas la liste des serveurs trop souvent. Une fois que vous arrêtez ce genre de programmes, l'erreur MASQ s'en ira quand les connexions commenceront a expirer.

7.19 J'ai des "ipfwadm : setsockopt failed : Protocol not available" lorsque j'essaye d'utiliser IPPORTFW !

Si vous obtenez ce message d'erreur, alors que vous venez d'installer un nouveau noyau, assurez vous de bien avoir copié le noyau et de bien avoir relancé LILO.

Référez vous à la fin de la section 6.7 () pour plus de détails.

7.20 Les partages Microsoft ne marchent pas a travers l'IP MASQ !

Pour supporter convenablement le protocole SMB, un module IP MASQ aurais besoin d'être écrit, mais il y a trois façons de contourner ce problème. Allez lire *cet article de la base de connaissances de Microsoft.*

La première façon est de configurer IPPORTFW et de rediriger les ports TCP 137, 138 et 139 vers la machine masquée. Bien que cette solution marche, elle ne permet d'accéder qu'a UNE seule machine interne.

La deuxieme solution est d'installer et configurer *Samba* sur le serveur MASQ. Avec Samba, vous pouvez mapper vos partages internes sur le serveur et les exporter aux clients externes. La configuration de Samba est entièrement décrite dans le SMB HOWTO et dans TrinityOS.

La troisième solution est de configurer un VPN (réseau privé virtuel) entre les deux machines ou entre les deux réseaux. Cela peut être fait soit avec PPTP, soit avec IPSEC. Il y a un patch 7.24 () pour Linux et aussi une implémentation complete d'IPSEC pour les noyaux 2.0 et 2.2. Cette solution est probablement la plus sure des trois solutions.

Toutes ces solutions ne sont pas décrites dans ce HOWTO. Je vous recommande d'aller lire TrinityOS pour de l'aide sur IPSEC et la page de John Hardin's sur PPTP pour plus d'informations.

S'IL VOUS PLAÎT, comprenez bien que le protocole SMB de Microsoft n'est PAS sur du TOUT. A cause de cela, faire transiter des fichiers, impressions et ouvertures de sessions SMB via Internet n'est PAS une bonne idée.

7.21 L'IRC ne marche pas correctement pour les utilisateurs MASQ. Pourquoi ?

La raison la plus probable est que la majorité des distributions Linux ont un serveur IDENT qui n'arrive pas a prendre en charge les liens MASQ. Ne vous inquiétez pas, il y a des IDENT de par le monde qui fonctionnent.

L'installation de ces logiciels est hors de propos dans ce HOWTO, mais chacun a sa propre documentation. Voici quelques adresses :
- *Mident* `<ftp://ftp.code.org/pub/linux/midentd/>` est largement utilisé par la majorité des utilisateurs d'IRC.
- *Sident* `<http://insecurity.net/sidentd.gz>`
- *Autres Idents incluant Oidentd* `<ftp://sunsite.unc.edu/pub/Linux/system/network/daemons/>`

Notez que certains serveurs IRC n'autoriseront pas de multiples connexions d'une machine, même si les ident sont différents et que les utilisateurs le sont aussi. Plaignez vous a l'administrateur du serveur IRC. :-)

7.22 mIRC n'arrive pas a faire des DCC send.

C'est un problème de configuration de votre copie de mIRC. Pour réparer cela, déconnectez vous du serveur IRC. Maintenant, dans mIRC, allez dans fichier/options/Connect, cliquez sur Edit et vérifiez que vous vous connectez bien sur le port 6667. Si vous avez besoin d'accéder via d'autres ports, allez voir plus bas. Ensuite, allez dans Fichier/options/Local Info et effacez le nom de machine ainsi que l'adresse IP, ensuite, sélectionnez "local host" et "ip adress" (ip adress sera peut être coché mais désactivé). Ensuite, dans "Lookup Method", configurez le pour "normal". Cela ne marchera PAS si vous avez coché "server". Voilà. Reconnectez vous au serveur irc.

Si vous avez besoin de vous connecter a des serveurs irc sur un autre port que 6667 (par exemple 6969), vous allez avoir besoin d'éditer le fichier `/etc/rc.d/rc.firewall` a l'endroit ou vous chargez les modules IP MASQ. Éditez la ligne "modprobe ip_masq_irc" et ajoutez "ports=6667,6969". Vous pouvez ajouter autant de ports que vous voulez tant qu'ils sont séparés par des virgules.

Ensuite, déconnectez tous les clients IRC de toutes les machines masquées et rechargez le module IP MASQ IRC :

```
/sbin/rmmod ip_masq_irc
/etc/rc.d/rc.firewall
```

7.23 Est ce que l'IP MASQ peut marcher avec une seule carte réseau (IP Aliasing) ?

Oui et non. La fonctionnalité "IP Alias" du noyau sert a mettre en place plusieurs interfaces comme eth0 :0, eth0 :1, etc mais il n'est pas recommandé d'utiliser des interfaces aliasées pour faire de l'IP Masquerading. Pourquoi ? Réaliser un firewall sécurisé devient plus dur avec une seule interface réseau. De plus, vous allez rencontrer un nombre anormalement élevé d'erreurs car les paquets reçus seront quasiment réenvoyés au même moment. A cause de cela, et aussi vu qu'une carte réseau coûte moins de 100 F, je recommande grandement d'acheter une deuxième carte réseau.

Vous devriez aussi comprendre que l'IP MASQ ne marchera en sortie que sur une interface physique comme ppp0, eth0, etc... Masquer une interface aliasée comme eth0 :0, eth1 :2, etc ne marchera PAS. En d'autres termes, cela NE MARCHERA PAS :

- /sbin/ipfwadm -F -a m -W eth0 :1 -S 192.168.0.0/24 -D 0.0.0.0/0
- /sbin/ipchains -A forward -i eth0 :1 -s 192.168.0.0/24 -j MASQ"

Si vous êtes toujours intéressés par le fait d'utiliser des interfaces aliasées, vous allez avoir à autoriser la fonctionnalité "IP Alias" du noyau. Recompilez et redémarrez. Une fois que vous avez votre nouveau noyau, vous allez devoir configurer Linux pour utiliser la nouvelle interface (eth0 :0, etc). Après cela, vous pourrez la traiter comme n'importe quelle interface avec quelques restrictions comme celle ci dessus.

7.24 J'essaye d'utiliser NETSTAT pour montrer mes connexions masquées, mais ça ne marche pas.

Il y a un problème dans le programme netstat. Après avoir rebooté linux, lancer un `netstat -M` fonctionne, mais après qu'un ordinateur MASQué ai fait un peu de trafic ICMP comme un ping ou un traceroute, etc. Vous verrez peut être la chose suivante :

```
masq_info.c: Internal Error 'ip_masquerade unknown type'.
```

Pour contourner ce problème, utilisez la commande "`/sbin/ipfwadm -M -l`". Vous noterez aussi qu'une fois que les entrées ICMP masquées expirent, netstat remarche.

7.25 Je voudrais réussir a faire marcher Microsoft PPTP (tunnel GRE) et/ou des tunnels IPSEC (Linux SWAN) à travers l'IP MASQ.

C'EST possible. Toutefois, c'est légèrement hors de propos ici. Allez consulter la page *PPTP Masq* de John Hardin pour plus de détails.

7.26 Je veux faire marcher le jeu en réseau XYZ à travers IP MASQ, mais cela ne marche pas !

Tout d'abord, allez faire un tour sur *Lee Nevo's MASQ Applications page* `<http://dijon.nais.com/nevo/masq/>`. Si la solution a votre probleme ne s'y trouve pas, essayez de patcher votre noyau avec le patch *LooseUDP* `<ftp://ftp.netcom.com/pub/mu/mumford/loose-udp-2.0.36.patch.gz>` qui est décrit dans la section 6.10 (). Allez aussi faire un tour sur la *NAT Page* `<http://www.alumni.caltech.edu/dank/peer-nat.html>` de Dan Kegel pour plus d'informations.

Si vous en avez le courage et les capacités, utilisez "`tcpdump`" pour écouter votre réseau. Essayez de découvrir quels protocoles et ports votre jeu XYZ utilise. Avec ces informations en main, inscrivez vous a la liste *IP Masq email list* et envoyez les résultats pour de l'aide.

7.27 IP MASQ marche très bien, et puis au bout d'un certain temps, il s'arrête de marcher. Un reboot semble corriger le problème pour un certain temps. Pourquoi ?

Je parie que vous utilisez IPAUTOFW et/ou que vous l'avez compilé dans votre noyau, non ? C'est un problème bien connu d'IPAUTOFW. Il est recommandé de ne même pas configurer IPAUTOFW dans le noyau et d'utiliser IPPORTFW à la place. Tout cela est couvert dans la section 6.7 ().

7.28 Les machines masquées ne peuvent pas envoyer ni recevoir des emails !

Bien que cela ne soit pas un problème de masquerading, mais que cela arrive a pas mal de gens, ce problème mérite d'être mentionné.

SMTP : Le problème est que vous utilisez certainement une machine Linux comme serveur de mail et vous avez l'erreur suivante :

```
"error from mail server : we do not relay"
```

Les dernières version de Sendmail et d'autres MTA (mail transfer agents) désactivent le relayage par défaut (c'est une bonne chose). Donc, faites les chose suivante :

– Sendmail : Autoriser un relayage spécifique pour vos machines internes en éditant le fichier `/etc/sendmail.cw` et ajoutez le nom de la machine et le nom du domaine de votre machine masquée. Vous devriez aussi vérifier que l'IP et le nom complet de la machine en question se trouvent dans le fichier `/etc/hosts`. Une fois que cela a été fait, vous aurez a redémarrer Sendmail pour qu'il relise son fichier de configuration. Tout cela est expliqué dans *TrinityOS - Section 25* `<http://www.ecst.csuchico.edu/dranch/LINUX/TrinityOS.wri>`

POP-3 : Certains utilisateurs configurent leurs ordinateurs masqués pour aller récupérer leurs emails sur des serveurs POP-3 externes. Cela ne devrais pas poser de problèmes, mais certains serveurs essayent de faire des requêtes IDENT sur le port 113. La majorité des systèmes de part le monde ont comme police par défaut mise a DENY. Ce n'est pas bien du tout. Réglez la a REJECT et relancez votre `rc.firewall`.

7.29 J'ai besoin que différents réseaux sortent sur des adresses IP différentes (IPROUTE2)

Disons que vous avez le problème suivant :

```
LAN ----------> IP officielle
192.168.1.x --> 123.123.123.11
192.168.2.x -->123.123.123.12
```

Tout d'abord, vous devez comprendre que IPFWADM et IPCHAINS entrent en jeu apres que le système de routage aie décidé ou envoyer le paquet. Cela devrais être écrit en grosses lettres rouges dans toutes les documentations IPFWADM/IPCHAINS/IPMASQ. Vous aurez donc tout d'abord a bien faire votre routage et ensuite, ajouter IPFWADM/IPCHAINS, et/ou MASQ.

Dans le cas précédent, Vous allez devoir persuader le système de routage de diriger les paquets de 192.168.1.x vers 123.123.123.11 et les paquets de 192.168.2.x vers 123.123.123.12. C'est la partie la plus ardue. Ajouter du MASQ par dessus un routage correct est assez aisé.

Pour faire de routage bizarre, vous allez utiliser IPROUTE2.

Le site FTP principal est :

- *ftp ://ftp.inr.ac.ru/ip-routing*
  Les miroirs sont :
- *ftp ://linux.wauug.org/pub/net*
- *ftp ://ftp.nc.ras.ru/pub/mirrors/ftp.inr.ac.ru/ip-routing/*
- *ftp ://ftp.gts.cz/MIRRORS/ftp.inr.ac.ru/*
- *ftp ://ftp.funet.fi/pub/mirrors/ftp.inr.ac.ru/ip-routing/ (STM1 to USA)*
- *ftp ://sunsite.icm.edu.pl/pub/Linux/iproute/*
- *ftp ://ftp.sunet.se/pub/Linux/ip-routing/*
- *ftp ://ftp.nvg.ntnu.no/pub/linux/ip-routing/*
- *ftp ://ftp.crc.ca/pub/systems/linux/ip-routing/*
- *ftp ://ftp.paname.org (France)*
- *ftp ://donlug.ua/pub/mirrors/ip-route/*
- *ftp ://omni.rk.tusur.ru/mirrors/ftp.inr.ac.ru/ip-routing/*

Des RPM sont disponibles sur :

- *ftp ://omni.rk.tusur.ru/Tango/*
- *ftp ://ftp4.dgtu.donetsk.ua/pub/RedHat/Contrib-Donbass/KAD/*

NOTE : Les instructions suivantes sont données uniquement car il n'y a que peu d'informations disponibles sur IPROUTE2. Allez lire *http ://www.compendium.com.ar/policy-routing.txt* pour avoir le début d'un HOWTO sur d'IPROUTE2.

Les commandes `"iprules"` et `"iproute"` sont les mêmes que `"ip rules"` et `"ip route"` (Je préfère les premières car elles sont plus facile a retrouver). Toute les commandes ci dessous n'ont pas été testées, si elles ne marchent pas, contactez l'auteur d'IPROUTE2... Pas David Ranch, Ambrose Au, ou qui que ce soit sur la liste Masq, car cela n'a rien a voir avec l'IP Masquerading.

Les premières commandes sont à exécuter une fois au boot, disons dans `/etc/rc.d/rc.local`.

```
# Autoriser les deux LAN a communiquer, sans masq.
/sbin/iprule add from 192.168.0.0/16 to 192.168.0.0/16 table main pref 100
# Tout le trafic restant de 192.168.1.x va sortir, on le prends en charge via la table 101.
/sbin/iprule add from 192.168.1.0/24 to 0/0 table 101 pref 102
# Tout le trafic restant de 192.168.2.x va sortir, on le prends en charge via la table 102.
/sbin/iprule add from 192.168.2.0/24 to 0/0 table 102 pref 102
```

Ces commandes doivent être lancées lorsque eth0 est configurée, par exemple, dans `/etc/sysconfig/network-scripts/ifup-post` (pour les RedHat). Assurez vous les lancer a la main pour être sur qu'elles marchent.

```
# La table 101 force tous les paquets y arrivant à être envoyés via 123.123.123.11
/sbin/iproute add table 101 via 62123.123.123.11
# La table 102 force tous les paquets y arrivant à être envoyés via 123.123.123.12
/sbin/iproute add table 102 via 62123.123.123.12
```

A cette étape, vous devriez voir les paquets venant de 192.168.1.x a destination du monde extérieur sont routés via 123.123.123.11, et que les paquets de 192.168.2.x le sont via 123.123.123.12.

Une fois que le routage est correct, vous pouvez ajouter vos règles d'IPFWADM ou d'IPCHAINS. Pour IPCHAINS, on pourrais mettre :

```
/sbin/ipchains -A forward -i ppp+ -j MASQ
```

Si tout s'est bien passé, le code MASQ devrais voir les paquets routés sur 123.123.123.11 et 123.123.123.12, et utiliser ces adresses comme adresses source de MASQ.

7.30 Pourquoi est ce que les noyaux 2.1 et 2.2 utilisent IPCHAINS et non plus IPFWADM ?

IPCHAINS inclu plus de fonctionnalités qu'IPFWADM, dont :

- La "Qualité de Service" (QoS)
- Un système en arbre contrairement au système linéaire d'IPFWADM (par exemple, cela permet des choses comme "si ça vient de ppp0, va directement a cette règle (qui contient ses règles propres)").
- IPCHAINS est plus souple a configurer. Par exemple, il inclue la commande "replace" (en plus d'"insert" et "add"). Il est aussi possible de nier des règles (par exemple "ignorer tous les paquets sortant qui ne viennent pas de mon IP" pour ne pas être la source d'une attaque spoofée).
- IPCHAINS peut filtrer n'importe quel protocole IP, pas seulement TCP, UDP et ICMP.

7.31 Je viens de passer a un noyau 2.2 et ça ne marche plus.

Il y a plusieurs raisons qui peuvent faire que cela ne marche plus, en supposant que vous avez une bonne connection à Internet et à votre LAN :

- Assurez vous que vous avez les fonctionnalités et les modules nécessaires compilés et chargés. Référez vous aux sections précedentes pour cela.
- Vérifiez `/usr/src/linux/Documentation/Changes` et assurez vous que vous avez les bonnes versions des outils réseau installés.
- Assurez vous d'avoir bien activé l'IP forwarding. Essayez de lancer `echo "1" > /proc/sys/net/ipv4/ip_forwarding`.
- Assurez vous d'avoir bien suivi tous les tests de la section 5 () de ce howto.
- Vous devez utiliser *ipchains* `<http://www.rustcorp.com/linux/ipchains/>` pour manipuler les regles ipmasq et firewal.
- IPPORTFW et IPAUTOFW ont été remplacés par *IPMASQADM* `<http://juanjox.kernelnotes.org/>`. Vous allez devoir appliquer des patchs au noyau, et à le recompiler, compiler le nouvel outil IPMASQADM, et ensuite, convertir vos anciennes regles IPAUTOFW/IPPORTFW pour utiliser la nouvelle syntaxe. Tout cela est intégrallement décrit dans la section 6.7 ().
- Refaites toute la préparation et la configuration encore ! La plupart du temps, c'est juste une erreur de typo ou une erreur stupide que vous verrez facilement.

7.32 Je viens de mettre à jour mon noyau avec un 2.0.36 ou plus récent et ça ne marche plus !

Il y a plusieurs chose que vous devriez vérifier, en supposant que vous avez une bonne connection à Internet et à votre LAN :

- Assurez vous que vous avez bien toutes les fonctionnalités et modules de compilés et chargés. Référez vous aux sections précédentes pour plus de détails.
- Verifiez dans `/usr/src/linux/Documentation/Changes` que vous avez bien mis a jour le minimum pour survivre.
- Assurez vous de bien avoir activé l'IP forwardinf. Essayez `echo "1" > /proc/sys/net/ipv4/ip_forward`.
- Assurez vous d'avoir bien suivi tous les tests de la section 5 () de ce howto.
- Vous devriez utiliser *ipfwadm* `<http://www.xos.nl/>` pour manipuler les regles ipmasq et firewall. Vous aurez a patcher les noyaux 2.0.x pour utiliser ipchains.
- Refaites toute la préparation et la configuration encore ! La plupart du temps, c'est juste une erreur de typo ou une erreur stupide que vous verrez facilement.

7.33 J'ai besoin de faire marcher des connexions EQL avec l'IP MASQ.

L'EQL n'a rien a voir avec l'IP MASQ bien qu'ils soient souvent utilisés ensembles sur des machines Linux. A cause de cela, je vous recommenderais d'aller récupérer la derniere version de l'*EQL HOWTO de Robert Novak* qui devrais répondre a toutes vos questions.

7.34 Je n'arrive pas à faire marcher l'IP Masquerade ! Quelles sont les possibilités pour le faire avec Windows ?

Ok, c'est vous qui l'avez voulu. Si vous voulez une solution Windows NAT/proxy, voici une liste décente. Je n'ai aucune préférence entre ces différents utilitaires, vu que je n'en ai utilisé aucun. (NdT : j'utilise souvent Winroute et il est très bien).

- Firesock (des créateurs de Trumpet Winsock)
  - Joue le role de proxy
  - *http ://www.trumpet.com.au*
- Iproute
  - Programme DOS fait pour tourner sur un 286.
  - Requiert une machine Linux pour le MASQ.
  - *http ://www.mischler.com/iproute/*
- Microsoft Proxy
  - Requiert Windows NT Server
  - Assez cher
  - *http ://www.microsoft.com*
- NAT32
  - Compatible Windows 95/98/NT
  - *http ://www.nat32.com*
  - Coûte $25 pour Win9x et $47 pour Win9x et WinNT.
- SyGate
  - *http ://www.sygate.com*
- Wingate
  - Joue le role de proxy
  - Coûte $30 pour 2-3 IP.
  - *http ://www.wingate.com*
- Winroute
  - Fait du NAT.
  - *http ://www.winroute.cz/en/*

Mais ne dites pas que c'est moi qui vous y envoie !

7.35 Comment je m'inscris à la liste IP Masquerade ?

Pour s'inscrire à la liste IP Masquerading envoyez un mail à *masq-subscribe@indyramp.com.*

Le sujet et le corps de ce message sont **ignorés**. Ceci vous permet de recevoir tous les messages de la liste alors qu'ils arrivent. Vous avez la possibilité de recevoir les messages sous cette forme, mais si vous pouviez plutot vous abonner au condensé (digest), choisissez le plutôt. Le digest charge moins les machines qui servent les listes. Notez aussi qu'il n'est possible de poster qu'à partir de l'adresse où vous êtes enregistrés.

Pour plus de commandes, envoyez un email à *masq-help@tori.indyramp.com.*

7.36 Je veux aider au développement de l'IP Masquerading. Comment faire ?

Abonnez vous à la liste de développement de l'IP Masquerading, et contactez les grands développeurs là-bas, en envoyant un email à *masq-dev-subscribe@tori.indyramp.com* (ou pour le digest *masq-dev-digest-subscribe@tori.indyramp.com*).

NE posez pas de questions n'ayant pas de rapport avec le développement sur cette liste ! !

7.37 Où puis-je trouver plus d'informations sur l'IP Masquerading ?

Vous pouvez trouver plus d'informations sur l'IP Masquerading à *Linux IP Masquerade Resource* `<http://ipmasq.cjb.net/>` que David et moi maintenons. référez vous a la section 6.2 pour la disponibilité.

Vous pouvez aussi trouver plus d'informations à *The Semi-Original Linux IP Masquerading Web Site* `<http://www.indyramp.com/masq/>` maintenu par Indyramp Consulting, qui fournissent aussi les listes ipmasq.

7.38 Je veux traduire ce HOWTO dans une autre langue. Comment faire ?

Assurez vous que le langage dans lequel vous voulez traduire n'est pas déjà couvert par quelqu'un d'autre ; une liste des traductions faites est disponible sur *Linux IP Masquerade Resource* `<http://ipmasq.cjb.net/>`.

Envoyez un email à *ambrose@writeme.com* et je vous enverrais la derniere version du SGML de ce HOWTO.

7.39 Ce HOWTO semble à l'abandon, vous vous en occupez toujours ? Pouvez vous inclure plus d'informations sur ... ? Comptez vous l'améliorer ?

Oui, ce HOWTO est toujours maintenu. Il fut un temps ou j'étais coupable d'être trop occupé à travailler sur deux boulots et je n'ai pas beaucoup de temps pour travailler dessus, toutes mes excuses. Toutefois, avec l'arrivée de David Ranch et tant que mainteneur aux alentours de la version 1.5, les choses ont pas mal bougées.

Si vous pensez à un sujet qui devrais être inclus dans ce HOWTO, envoyez nous un email à *ambrose@writeme.com* et *dranch@trinnet.net*. Cela serais encore mieux si vous pouviez nous fournir cette information. David et moi inclurons cette information dans le HOWTO si elle nous semble appropriée. Et merci pour votre contribution.

Nous avons beaucoup de nouvelles idées et de plans pour mettre à jour ce HOWTO, comme des études de cas, qui couvreraient differentes configurations réseau mettant en jeu l'IP Masquerading, plus de choses sur la sécurité, l'utilisation d'ipchains, des exemples sur ipfwadm/ipchains, plus de FAQ, plus de choses sur les utilitaires de forwarding de port et de protocoles tels masqasmin, etc. Si vous pensez que vous pouvez aider, faites le. Merci.

7.40 J'ai reussi à faire marcher l'IP Masquerade, c'est génial ! Qu'est ce que je pourrais faire pour vous remercier ?

- Pouvez vous traduire ce HOWTO dans une nouvelle langue ?
- Remerciez les développeurs, et appréciez le temps et les efforts qu'ils ont passés dessus.
- Inscrivez vous a la liste IP MASQ et aidez les nouveaux arrivants.
- Envoyez nous un email pour nous faire savoir que vous êtes contents.
- Faites connaitre Linux autour de vous, et aidez les gens qui ont des problèmes.

8 Divers

8.1 Ressources utiles

Note du Traducteur : Tous les documents cités dans cette section sont en anglais, à moins d'une mention spéciale de ma part.

- *IP Masquerade Resource page* <`http://ipmasq.cjb.net/`> devrait contenir assez d'information pour l'installation d'IP Masquerade sur les noyaux 2.0, 2.2 et même les 1.2 !
- L'*archive de la mailing list IP Masquerade* <`http://www.indyramp.com/masq/list/`> contient certains des messages envoyés récement sur la liste.
- Ce document, en anglais : *Linux IP Masquerade mini HOWTO* <`http://ipmasq.cjb.net/ipmasq-HOWTO.html`> pour les noyaux 2.2.x et 2.0.x.
- Le *IP Masquerade HOWTO pour les noyaux 1.2.x* <`http://ipmasq.cjb.net/ipmasq-HOWTO-1.2.x.txt`> si vous utilisez un vieux noyau
- La *FAQ IP Masquerade* <`http://www.indyramp.com/masq/ip_masquerade.txt`> contient des informations générales
- *Linux IPCHAINS HOWTO* <`http://www.freenix.org/unix/linux/HOWTO-vo/IPCHAINS-HOWTO.html`> et *http ://www.rustcorp.com/linux/ipchains/* contiennent plein d'informations sur l'utilisation d'ipchains, ainsi que les sources et les binaires pour ipchains.
- La page *X/OS Ipfwadm page* <`http://www.xos.nl/linux/ipfwadm/`> contient les sources, binaires, de la documentation et d'autres informations sur le package `ipfwadm`.
- Une page sur les *applications qui marchent avec l'IP Masquerading de Linux* <`http://dijon.nais.com/~nevo/masq/`> par Lee Nevo fournis des trucs et astuces pour faire marcher tout ça avec l'IP Masquerading.
- Le *LDP Network Administrator's Guide* <`http://metalab.unc.edu/mdw/LDP/nag/nag.html`> est incontournable pour les débutants essayant d'installer un réseau.
- *Trinity OS Doc* <`http://www.ecst.csuchico.edu/~dranch/LINUX/TrinityOS.wri`>, Une documentation très complete sur l'utilisation de Linux en réseau.
- Le *Linux NET-3 HOWTO* <`http://www.freenix.org/unix/linux/HOWTO/NET-3-HOWTO.html`> (**en français**) contient aussi beaucoup d'informations utiles sur l'utilisation du réseau sous Linux.
- Le *Linux ISP Hookup HOWTO* <`http://www.freenix.org/unix/linux/HOWTO/ISP-Hookup-HOWTO.html`> (**en français**) et le *PPP HOWTO* <`http://www.freenix.org/unix/linux/HOWTO/PPP-HOWTO.html`> (**en français**) vous donnent des informations pour connecter votre hôte Linux sur Internet.
- Le *Linux Ethernet-Howto* <`http://www.freenix.org/unix/linux/HOWTO/Ethernet-HOWTO.html`> (**en français**) est une bonne source d'informations sur la mise un place d'un réseau local utilisant Ethernet.
- Vous pouvez également être intéressé par le *Linux Firewalling and Proxy Server HOWTO* <`http://www.freenix.org/unix/linux/HOWTO/Firewall-HOWTO.html`> (**en français**)
- Le *Linux Kernel HOWTO* <`http://www.freenix.org/unix/linux/HOWTO/Kernel-HOWTO.html`> (**en français**) vous guidera pour pour la recompilation de votre noyau.
- D'autres HOWTOs, en *français* <`http://www.freenix.org/unix/linux/HOWTO/`>, ou en *anglais* <`http://www.caldera.com/LDP/HOWTO/HOWTO-INDEX-3.html`>.
- Poster dans les newsgroups USENET : *comp.os.linux.networking*, ou, en français, *fr.comp.os.linux.configuration* ou *fr.comp.os.linux.moderated*

8.2 Ressources sur l'IP Masquerade

Le site *Linux IP Masquerade Resource* <`http://ipmasq.cjb.net/`> est dédié à l'IP Masquerading, aussi maintenu par David Ranch et moi. Les dernières informations y sont toujours disponibles, et il peut y avoir des choses non disponibles dans ce HOWTO.

Vous trouverez les ressources sur l'IP Masquerading aux endroits suivants :

– *http ://ipmasq.cjb.net/*, site primaire, redirigé sur *http ://www.tor.shaw.wave.ca/˜ambrose/*
– *http ://ipmasq2.cjb.net/*, site secondaire, redirigé sur *http ://www.geocities.com/SiliconValley/Heights/2288/*

8.3 Remerciements

En ordre alphabétique :

– Gabriel Beitler, gbeitler@aciscorp.com
section 3.3.8 sur Novell.
– Juan Jose Ciarlante, irriga@impsat1.com.ar
Pour son travail sur IPMASQADM, son travail sur le code des 2.1 et 2.2, le patch LooseUDP original, etc...
– Steven Clarke, steven@monmouth.demon.co.uk
pour ses contributions à IPPORTFW.
– Andrew Deryabin, djsf@usa.net
pour ses contributions au module ICQ MASQ.
– Ed Doolittle, dolittle@math.toronto.edu
pour ses suggestions sur l'option `-V` d'`ipfwadm` pour accroître la sécurité.
– Matthew Driver, mdriver@cfmeu.asn.au
Pour avoir aidé sur ce HOWTO, et avoir fourni la section 3.3.1 (mise en place de Windows 95).
– Ken Eves, ken@eves.com
pour la FAQ qui fournis des informations essentielles à ce HOWTO.
– John Hardin, jhardin@wolfenet.com
pour ses outils de forwarding PPTP et IPSEC.
– Glenn Lamb, mumford@netcom.com
pour le patch LooseUDP
– Ed. Lott, edlott@neosoft.com
pour une longue liste de systèmes et logiciels testés.
– Nigel Metheringham, Nigel.Metheringham@theplanet.net
Pour avoir apporté sa version de l'IP Packet Filtering et IP Masq HOWTO, qui rends ce HOWTO encore meilleur.
les section 4.1, 4.2, et d'autres.
– Keith Owens, kaos@ocs.com.au
pour son excellent guide sur ipfwadm de la section 4.2
pour une correction à `ipfwadm -deny` qui évitais un trou de sécurité, et clarifiait le status des `ping` via IP MASQ.
– Michael Owings, mikey@swampgas.com
pour avoir fournis la section a propos de CU-SeeMe et le Linux IP-Masquerade Teeny How-To
– Rob Pelkey, rpelkey@abacus.bates.edu
pour avoir fournis les sections 3.3.6 et 3.3.7 (mise en place de MacTCP et Open Transport)
– Harish Pillay, h.pillay@ieee.org
pour la secion 4.5 (dial-on-demand avec Diald)
– Mark Purcell, purcell@rmcs.cranfield.ac.uk
pour la section 4.6 (IPautofw)
– David Ranch, dranch@trinnet.net
pour son aide a maintenir ce HOWTO, la page Linux IP Masquerade Resource, le document TrinityOS, ... , trop de choses pour tout mettre ici :-)
– Paul Russell, rusty@rustcorp.com.au
pour son travail sur IPCHAINS, des patchs IP MASQ, etc.
– Ueli Rutishauser, rutish@ibm.net
pour la section 3.3.9 (mise en place d'OS/2 Warp)

- Fred Viles, fv@episupport.com
  pour ses patchs FTP.
- John B. (Brent) Williams, forerunner@mercury.net
  pour la section 3.3.7 (mise en place d'Open Transport)
- Enrique Pessoa Xavier, enrique@labma.ufrj.br
  Pour une suggestion sur BOOTp.
- Tous les habitués de la liste IP MASQ, masq@tiffany.indyramp.com
  pour avoir fournis leur aide aux utilisateurs de Linux.
- D'autres programmeurs ou rédacteurs de documentations :
  - Delian Delchev, delian@wfpa.acad.bg
  - David DeSimone (FuzzyFox), fox@dallas.net
  - Jeanette Pauline Middelink, middelin@polyware.iaf.nl
  - Miquel van Smoorenburg, miquels@q.cistron.nl
  - Jos Vos, jos@xos.nl
  - Et tous ceux que j'ai pu oublier (faites-le moi savoir !)
- tous les utilisateurs envoyant des suggestions et des critiques à la mailing list, et plus particulièrement ceux qui m'ont fait part d'erreurs dans ce document et les clients supportés ou non.
- Je vous demande pardon si je n'ai pas inclus les informations que certains utilisateurs m'ont envoyées. De nombreuses idées et suggestions me sont envoyées, mais je n'ai pas le temps de les vérifier, ou je les égare. J'essaie de faire de mon mieux pour incorporer toutes les informations qu'on m'envoye pour rédiger ce HOWTO. Je vous remercie pour votre effort, et j'espère que vous comprenez ma situation.

8.4 Réference

- IP masquerade FAQ de Ken Eves
- Archive de la mailing list IP Masquerade de Indyramp Consulting
- Le site WWW IP Masquerade d'Ambrose Au
- La page sur Ipfwadm par X/OS
- Divers HOWTOs liés au réseau sous Linux
- Certains sujets abordés dans TrinityOS de David Ranch