# Linux IP Masquerade HOWTO (version française)

David Ranch (version orginale), *dranch@trinnet.net*
Pejvan AHMAD-BEIGUI (traduction - v1.95-b : le 28 juin 2001), *pejvan.ahmad-beigui@ensimag.imag.fr*
v1.95, November 14, 2000

Ce document décrit la mise en application de l'IP Masquerading sur une machine Linux. IP Masq est une forme de Traduction d'Adresse Reseau (Network Address Translation ou NAT en anglais) qui permet à un ou plusieurs ordinateurs qui ne possédent pas d'adresses IP, de communiquer sur Internet grâce à l'unique adresse IP d'une machine Linux. Ces ordinateurs étant connectés de manière interne sur le serveur Linux.

## Contents

# 1 Introduction

## 1.1 Introduction à l'IP Masquerading ou IP MASQ en abrégé

Ce document décrit la mise en application de l'IP Masquerading sur une machine Linux. IP Masq est une forme de Traduction d'Adresse Reseau (Network Address Translation ou NAT en anglais) qui permet à un ou plusieurs ordinateurs, qui ne possèdent pas d'adresses IP, de communiquer sur Internet grâce à l'unique adresse IP d'une machine Linux. Ces ordinateurs étant connectés de manière interne sur le serveur Linux. Cette connexion peut se faire avec les différentes techonologies LAN (Local Area Networks ou en français, réseaux locaux) tels que Ethernet, TokenRing, FDDI mais aussi par d'autres types de connexions tels que le PPP ou le SLIP. Ce document utilisera Ethernet comme exemple principal puisque c'est le scenario le plus commun.

> ***Ce document se destine aux utilisateurs d'un des deux noyaux stable Linux : 2.0.38+ et 2.2.17+ sur un compatible PC (NDT : j'utilise comme routeur linux un vieux Sparc Classic et ca fonctionne parfaitement). Les noyaux plus anciens tels que 1.2.x, 1.3.x, et 2.1.x NE sont PAS traités dans ce document et, peuvent être considérés comme défectueux pour certaines versions. Nous vous recommandons de faire la mise à jour vers un des noyaux Linux stables avant d'utiliser l'IP Masquerading. Les nouveaux noyaux 2.3 et 2.4 avec le nouveau code NetFilter ne sont pas encore traités mais les URLs sont fournis ci-dessous. Une fois que les caractéristiques de Netfilter seront finalisées, le nouveau code sera traité dans ce HOWTO. Si vous voulez configurer IP Masq sur un Macintosh, contactez par email (en anglais) Taro Fukunaga,*** **tarozax@earthlink.net** ***pour recevoir une copie de sa version abregée du HOWTO pour MkLinux.***

## 1.2 Avant-Propos, Feedback & Credits

En tant que nouvel utilisateur, j'ai trouvé la configuration de l'IP masquerade sous Linux très déroutante (noyau 1.2.x à cette époque). Bien qu'il y ait eu une FAQ et une mailing list, il n'avait pas de document dedié. Il y avait aussi de la demande sur la mailing list pour un tel HOWTO. J'ai alors decidé d'écrire ce HOWTO comme point de départ pour les nouveaux utilisateurs et de poser les fondations qui permettraient aux autres utilisateurs de l'étoffer dans le futur. Si vous avez des suggestions, des corrections, etc. à nous soumettre au sujet de ce document pour nous permettre de l'améliorer, n'hesitez pas.

Ce document était basé sur la FAQ originale de Ken Eves, et des nombreux messages salutaires de la mailing list de l'IP Masquerade. Je remercie tout particulierement M. Matthew Driver dont le message sur la mailing list m'a inspiré l'organisation et finalement la rédaction de ce document. Dernièrement, David Ranch a réécrit ce HOWTO et a ajouté un nombre conséquent de sections pour le rendre aussi complet que possible.

Envoyez nous vos feedbacks et commentaires (en anglais) à *ambrose@writeme.com* et *dranch@trinnet.net* si vous avez des corrections à nous soumettre ou si des information/URLS/etc. manquent. Si vous avez des commentaires sur la traduction de ce document, ou des erreurs/améliorations à signaler, vous pouvez me contacter a : *pejvan.ahmad-beigui@ensimag.imag.fr* . Votre aide inestimable va certainement influencer la prochaine version de ce HOWTO !

**Ce HOWTO est destiné à être aussi complet que possible pour permettre la mise en place de votre réseau ipmasqueradée aussi rapidement que possible. David n'est pas un rédacteur technique professionnel. Vous pourrez donc trouver les informations de ce document pas assez généraux et/ou objectifs. Les dernières news et infos concernant ce HOWTO et les autres détails sur l'IP MASQ se trouvent à l' *IP Masquerade Resource* `http://ipmasq.cjb.net/` , site web que nous mettons à jour activement. Si vous avez des questions techniques sur l'IP Masquerade, contactez SVP la Mailing List plutôt que d'envoyer un email à David. La plupart des problèmes sur l'IP MASQ sont les mêmes pour TOUS et peuvent être facilement résolus par quelqu'un sur la Mailing List. De plus, la réponse vous parviendra bien plus rapidement sur la liste que le reply de David.**

La dernière version de ce document se trouve (sous differents formats dont l'HTML et le PostScript) sur les sites suivants

- *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*
- Vous pouvez aussi consulter l' *IP Masquerade Resource Mirror Sites Listing* `http://ipmasq.cjb.net/index.html#mirror` pour obtenir la liste de nos sites mirroirs.

## 1.3 Copyright & Désistement

Ce document est `copyright(c) 2000 David Ranch` pour la version originale et `copyright(c) 2001 Pejvan AHMAD-BEIGUI` pour la version française et est un document GRATUIT. Vous pouvez le redistribuer suivant les termes de la GNU General Public License.

Les informations contenues dans ce document, sont à notre connaissance, corrects. Cependant, lIP Masquerading de Linux est écrits par des humains et peut donc contenir des erreurs, bugs, etc.

Aucune personne, groupe ou autre organisme ne peut etre tenu responsable des dommages causés à votre (vos) ordinateur(s) ou des pertes dus à l'utiisations des informations de ce document. i.e :

**LES AUTHEURS ET LES PERSONNES PARTICIPANT AU DEVELOPPEMENT DE CE DOCUMENT NE PEUVENT ETRE TENUS RESPONSABLES**

**DES DOMMAGES CAUSES PAR L'UTILISATION DES INFORMATIONS CONTENUES DANS CE DOCUMENT.**

Ok, avec tout ca dernière nous... que le spectacle commence.

# 2 Connaissances Préliminaires

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

L'IP Masquerade est une fonctionnalité réseau de Linux similaire à la Translation d'Adresse Réseau un-vers-plusieurs que l'on trouve dans beaucoup de firewalls et de routeurs commerciaux. Par exemple, si une machine Linux est connectée à Internet via PPP, Ethernet, etc., l'IP Masquerading permet aux ordinateurs "internes" connectés à cette machine Linux (via PPP, Ethernet, etc.) d'accéder aussi à Internet. L'IP Masquerading fonctionne même si ces machines internes n'ont pas d'*adresses IP officielles*.

MASQ permet à un groupe de machines d'avoir accès à Internet via la passerelle MASQ de manière *transparente*. Pour les autres ordinateurs connectés à Internet, tout le traffic généré va sembler provenir du serveur Linux IP MASQ lui-même. En plus de ces fonctionnalités, IP Masquerade fournit les bases de la création d'un environnement réseau de HAUTE sécurité. Avec un firewall bien configuré, casser la securité d'un système de masquerading et d'un LAN interne bien configuré devrait être très difficile.

Si vous voulez savoir en quoi MASQ diffère des solutions 1:1 NAT and Proxy, reportez vous à la partie 7.5 () de la FAQ.

## 2.2 Situation Actuelle

IP Masquerade est sorti il y a plusieurs années maintenant et il est plutôt mature depuis les noyaux 2.2.x. Depuis le noyau 1.3.x, Linux est fourni avec MASQ. Aujourd'hui de nombreuses personnes et entreprises l'utilisent avec d'excellents résultats.

Les utilisations courantes du réseau tels que la navigation Web, les TELNET, PING, TRACEROUTE, etc. fonctionnent bien avec IP Masquerade. D'autres types de communications, tels que FTP, IRC, et Real Audio fonctionnent bien avec les modules IP MASQ appropriés chargés en memoire. Certains programmes réseaux tels que les streaming audio (MP3s, True Speech, etc.) fonctionnent aussi. Quelques personnes sur la mailing list ont même réussi à obtenir de bons résultats avec des logiciels de video conferencing.

A noter aussi que faire de l'IP Masquerading avec UNE seule carte réseau (NIC) pour MASQuer entre des réseaux Ethernet interne et externe N'est PAS recommandé. Pour plus de détails, reportez vous SVP à la partie 7.25 () de la FAQ.

Dans tous les cas, reportez vous SVP à la partie 6.2 () pour une liste plus complète des logiciels fonctionnant sous IP MASQ.

IP Masquerade fonctionne bien comme serveur pour des 'machines clientes' tournant sous différents systèmes d'exploitations (operating sytems ou OS en anglais) et différents materiels dont :

- Unix: Sun Solaris, *BSD, Linux, Digital UNIX, etc.
- Microsoft Windows 2000, NT (3.x et 4.x), 95/98/ME, Windows for Workgroups (avec le package TCP/IP)
- IBM OS/2
- ordinateurs Apple Macintosh sous MacOS avec soit MacTCP soit Open Transport

- systèmes sous DOS avec les packet drivers et le package NCSA Telnet
- VAXen
- Compaq/Digital Alpha sous Linux et NT
- même les ordinateurs Amiga avec AmiTCP ou la pile AS225.

Cette liste continue encore et toujours mais ce qu'il faut bien comprendre, c'est que si votre OS comprend le TCP/IP, il devrait fonctionner avec l'IP Masquerade !

## 2.3 Qui Peut Profiter de l'IP Masquerade?

- Si vous avez un serveur Linux connecté à Internet
- Si vous avez des ordinateurs connectés à ce serveur dans un sous-réseau local et si ces ordinateurs "parlent" le TCP/IP et/ou
- si votre serveur Linux a un ou plusieurs modems et se comporte comme un serveur PPP ou SLP pour connecter d'autres ordinateurs
- ces **AUTRES** ordinateurs n'ont pas d'adresses IP officielles ou publiques (i.e. avec des adresses privées).
- Et bien sûr, si vous voulez que ces **AUTRES** ordinateurs communiquent sur Internet sans depenser d'argent supplémentaire pour l'acquisition d'adresses IP publiques / officielles ou, soit configurer Linux pour en faire un routeur soit acheter un routeur externe.

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

- Si vous avez une ordinateur Linux isolé connecté à Internet (bien que mettre en place un firewall serait une bonne idée), ou
- si vous avez déjà des adresses IP publiques pour vos **AUTRES** machines, et
- et bien sûr, si vous n'aimez pas trop l'idée de devoir utiliser Linux en 'free ride' et vous vous sentez plus à l'aise avec des outils commerciaux mais chers qui font exactement la même chose.

## 2.5 Comment fonctionne IP Masquerade ?

Tiré de la FAQ sur l'IP Masquerade de Ken Eves:

```
Voici un dessin de la configuration la plus simple:

 SLIP/PPP         +------------+                        +-------------+
 vers votre FAI   |  Linux     |         SLIP/PPP       | un          |
<---------- modem1|    #1      |modem2 ----------- modem3| ordinateur  |
  111.222.121.212 |            |           192.168.0.100 |             |
                  +------------+                        +-------------+


  Sur le dessin ci-desus, une machine Linux, Linux #1, avec IP_MASQUERADING d'installe
  est connecte a Internet par le modem1 via SLIP/ou/PPP. Elle a une adresse IP publique :
```

```
    111.222.121.212. Elle a aussi un modem2 qui permet a l'appelant des connexions SLIP/ou/PPP.

    La seconde machine (qui n'a pas besoin de tourner sous Linux) ce connecte a la machine
    Linux #1 et commence une session SLIP/ou/PPP. Elle N'a PAS d'adresse IP publique
    sur Internet c'est pourquoi elle utilise l'adresse privee 192.168.0.100 (voir ci-dessous).

    Avec IP Masquerade et une configuration de routage correcte, la machine "un ordinateur"
    peut interagir avec Internet comme si elle y etait directement connectee (a quelques
    petites exceptions pres).

Citons Pauline Middelink:
  N'oublions pas de mentionner le fait que la machine "un ordinateur" doit avoir Linux #1
  configure comme sa passerelle (que ca soit la route par defaut ou juste un sous reseau
  n'a pas d'importance). Si la machine "un ordinateur" ne peut pas faire ca, alors la
  machine Linux doit etre configuree de telle sorte que le proxy arp fonctionne pour toute
  les adresses de routage. Mais la mise en place et la configuration d'un proxy arp depasse
  le cadre de ce document.

L'extrait suivant est tire d'un post sur comp.os.linux.networking qui a ete modifie
de facon a tenir compte des noms utilises dans l'exemple precedent :

   o Je dis a la machine "un ordinateur" que mon Linux, connecte via PPP ou SLIP, est
   sa passerelle.
   o Quand un paquet provenant d"un ordinateur" arrive a ma machine Linux, elle va lui
   assigner un nouveau numero de port source TCP/IP et va coller sa propre adresse IP
   dans l'entete du paquet, tout en sauvegardant l'entete originale. Le server MASQ
   va ensuite envoyer le paquet ainsi modifie sur Internet via son interface SLIP/PPP.
   o Quand un paquet arrive d'Internet vers la machine Linux, Linux va examiner si son
    numero de port est l'un des numeros qu'il avait assigne precedement. Si c'est le cas,
    le server MASQ va recuperer le port et l'adresse IP originale et les remettre dans
    l'entete de paquet qui est revenu. Enfin Linux va renvoyer ce paquet a la machine
    "un ordinateur".
   o La machine qui a envoye le paquet ne fera pas la difference.
```

### Un Autre Exemple d'IP Masquerading :

Un exemple typique est donné dans le diagramme ci-dessous :

```
    +----------+
    |          |  Ethernet
    | A        |::::::
    |          |.2   : 192.168.0.x
    +----------+     :
                     :      +----------+   PPP
    +----------+     :   .1 |  Linux   |   link
    |          |     ::::::| Masq-Gate|:::::::::::::::::::// Internet
    | B        |::::::      |          |  111.222.121.212
    |          |.3   :      +----------+
    +----------+     :
                     :
    +----------+     :
```

```
|          |     :
| C        |::::::
|          |.4
+----------+

|                         |           |
| <- Reseau Interne -->   |           | <- Reseau Externe ---->
|                         |           |
```

Dans cet exemple, il y a (4) ordinateurs qui méritent notre attention. Il y a aussi sans doute quelque chose tout à droite d'où provient votre connexion PPP à Internet (serveur terminal, etc.) et il y a aussi un serveur distant (très très loin de la droite de cette page) sur Internet avec qui vous voulez communiquer. Le serveur Linux `Masq-Gate` est la passerelle d'IP Masquerading pour TOUT le réseau interne constitué des machines `A`, `B` et `C`. C'est par là que se fera leur accès à Internet. Le réseau interne utilise une des *adresses reservées pour les réseaux privés par le RFC-1918* . Dans notre cas, c'est les adresses de Classe C 192.168.0.0. Le serveur Linux a l'adresse 192.168.0.1 alors que les autres ordinateurs ont les adresses suivantes :

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

Les trois machines, `A`, `B` et `C`, peuvent tourner sous n'importe quel système d'exploitation pour peu qu'ils puissent communiquer par TCP/IP. Les OS tels que **Windows 95**, **Macintosh MacTCP ou OpenTransport** et **Linux** peuvent se connecter à d'autres machines sur Internet. Lorsqu'il est lancé, le serveur IP Masquerade ou `portail MASQ` convertit toutes les connexions internes de telle sorte qu'ells semblent provenir du `passerelle MASQ` lui même. MASQ reconvertit ensuite les données qui lui reviennent sur un port masqueradé et ces données sont renvoyées vers la machine qui en est à l'origine. A cause de cela, les ordinateurs du réseau interne voient une route directe vers Internet et ne sont pas au courant que leurs données sont masqueradées. C'est ce que l'on appelle une connexion "transparente".

NB: Vous pouvez vous reporter à 7 () pour de plus amples détails sur les sujets tels que :

- les différences entre NAT, MASQ, et les serveurs Proxy.
- le fonctionnement des firewalls pour les trames

## 2.6 Configurations Requises pour IP Masquerade sous Linux 2.2.x

**** Reportez vous SVP à l' *IP Masquerade Resource* `http://ipmasq.cjb.net/` pour les informations les plus récentes****

- Les sources du noyau 2.2.x disponible sur le site : *http://www.kernel.org/*
  NOTE #1: Les noyaux Linux 2.2.x inférieurs à 2.2.16 ont un trou de sécurité (TCP root exploit) et les versions inférieures à 2.2.11 ont un bug de fragmentation dans IPCHAINS. Pour cette raison, les personnes qui utilisent des règles IPCHAINS très restrictives sont attaquables. Mettez à jour votre noyau vers une version corrigée. NOTE #2: La plupart des nouvelles distributions compatibles7 () tels que Redhat 5.2 peuvent ne pas correspondre à l'installation de Linux 2.2.x requise. Des utilitaires tels que DHCP, NetUtils, etc. vont devoir être mis à jour. Vous pourrez trouver plus de details dans ce HOWTO.

- Modules noyau, de préférénce 2.1.121 ou mieux, disponibles sur les sites suivants : *http://www.pi.se/blox/modutils/index.html* or *ftp://ftp.ocs.com.au/pub/modutils/*
- Un réseau fonctionnant sous TCP/IP ou un LAN traité dans *Linux NET-3-4 HOWTO* `http://www.linuxdoc.org/HOWTO/NET3-4-HOWTO.html` et le *Network Administrator's Guide* `http://www.linuxdoc.org/LDP/nag/nag.html`
  Vous pouvez aussi regarder le document *TrinityOS* `http://www.ecst.csuchico.edu/~dranch/LINUX/index-linux.html#TrinityOS` . TrinityOS est un guide très complet pour le réseau sous linux. Il traite de nombreux thèmes dont : l'IP MASQ, la sécurité,les DNS, DHCP, Sendmail, PPP, Diald, NFS, VPNs basés sur IPSEC et possède une section sur les performances. Il contient plus de 50 sections au total !
- La connexion de votre machine Linux à Internet est traitée dans : *Linux ISP Hookup HOWTO* `http://www.linuxdoc.org/HOWTO/ISP-Hookup-HOWTO.html` , *Linux PPP HOWTO* `http://www.linuxdoc.org/HOWTO/PPP-HOWTO.html` , *TrinityOS* `http://www.ecst.csuchico.edu/~dranch/LINUX/index-linux.html#TrinityOS` , *Linux DHCP mini-HOWTO* `http://www.linuxdoc.org/HOWTO/mini/DHCP/index.html` , *Linux Cable Modem mini-HOWTO* `http://www.linuxdoc.org/HOWTO/Cable-Modem/index.html` and *http://www.linuxdoc.org/HOWTO/mini/ADSL.html* `http://www.linuxdoc.org/HOWTO/mini/ADSL.html`
- IP Chains 1.3.9 ou supérieur est disponible ici : *http://netfilter.filewatcher.org/ipchains/* .
  Des informations supplémentaires sur les versions requises et les dernières mises à jour de IPCHAINS HOWTO etc. se trouvent à l'URL suivant : *Linux IP Chains page* `http://netfilter.filewatcher.org/ipchains/`
- La configuration, la compilation et l'installation d'un nouveau noyau Linux sont expliquées dans le *Linux Kernel HOWTO* `http://www.linuxdoc.org/HOWTO/Kernel-HOWTO.html`
- Vous pouvez télécharger de nombreux utilitaires optionnels pour IP Masquerade qui permettent des fonctionnalités supplémentaires tels que :
  - port-forwarder ou re-diriger des ports TCP/IP :
    * *IP PortForwarding (IPMASQADM) - RECOMMENDED* `http://juanjox.kernelnotes.org/` ou l'ancien *mirror* .

  module ICQ MASQ

  - *module ICQ MASQ d'Andrew Deryabin*

  Solutions de PORTFW pour FTP:

  - Les deux noyaux 2.2.x et 2.0.x ont un Module MASQ pour PORTFWer un FTP vers une machine MASQuée. Reportez vous SVP à la page relative aux applications ( *IPMASQ WWW site* ) pour les détails complets.
  - Il y a un proxy FTP complet fait par SuSE qui possède une fonction similaire au PORTFWing pour accéder à un serveur FTP interne. Pour plus de details, reportez vous SVP à l'URL : *SuSe Proxy URL*

  IPROUTE2 pour la vraie 1:1 NAT, le routage basé sur la source (Policy-based / source routing), et le façonnage du Traffic :

  - *ftp://ftp.inr.ac.ru/ip-routing*
  - La documentation se trouve ici : *http://www.compendium.com.ar/policy-routing.txt*
  - L' *Advanced Routing HOWTO*

– Voici quelques serveurs mirroirs pour le code source:
*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/*
les RPMs sont disponibles ici : *ftp://omni.rk.tusur.ru/Tango/* et la : *ftp://ftp4.dgtu.donetsk.ua/pub/RedHat/Contrib-Donbass/KAD/*

Reportez vous SVP à l' *IP Masquerade Resource* `http://ipmasq.cjb.net/` pour plus d'informations sur ces patches ou d'autres éventuels patches.

## 2.7 Configurations Requises pour IP Masquerade sous Linux 2.3.x and 2.4.x

**** Reportez vous SVP à l' *IP Masquerade Resource* `http://ipmasq.cjb.net/` pour les informations les plus récentes****

- Les nouveaux noyaux 2.3.x et 2.4.x utilisent un nouveau système appelé NetFilter (un peu à la manière des noyaux 2.2.x qui utilisèrent IPCHAINS). Heureusement, **contrairement** à la migration vers IPCHAINS, le nouvel utilitaire NetFilter arrive avec des modules noyaux qui permettent d'utiliser de façon NATIVE la syntaxe d'IPCHAINS et d'IPFWADM et donc il n'est pas nécessaire de réecrire tous vos anciens scripts. Mais il pourrait y avoir quelques avantages à le faire (vitesse, nouvelles fonctionnalités, etc.) mais tout ca dépend de la qualité de vos anciennes règles. De nombreux changements d'architectures ont eu lieu dans ce nouveau code et permettent beaucoup plus de flexibilité, et de fonctionnalités à venir, etc. Voici quelques avantages et inconvenients des nouvelles fonctionnalités :

  **avantages:**

  – VRAIE 1:1 NAT pour ceux qui veulent avoir des sous-reseaux TCP/IP et qui veulent jouer avec
  – PORTFWing integré. IPMASQADM n'est donc plus obligatoire
  – le PORTFWing integré fonctionne pour le traffic externe et interne. Ce qui signifie que les utilisateurs qui utilisaient PORTFW pour le traffic externe et le REDIR pour le traffic interne ne vont plus avoir à utiliser deux utilitaires !
  – Capacité de routage Policy-Based complète (routage d'adresse TCP/IP base sur la source)
  – Compatible avec les capacités FastRoute de Linux pour obtenir un packet forwarding beaucoup plus rapide (changement de réseau Linux)
  – compatibilité complète avec TCP/IP v4, v6, et même DECnet (ack!)
  – compatible avec les wildcard pour les noms d'interface comme : ppp* pour PPP0, PPP1, etc.
  – permet le filtrage des entrées et des sorties sur les INTERFACES
  – Filtrage Ethernet MAC

- limitation du taux des paquets de type Denial of Service (DoS)
- fonctionnalité très simple, générique, de type stateful d'inspection
- la fonction Packet REJECT possède maintenant un renvoie de messages ICMP configurable par l'utilisateur
- Différents niveaux de logging (différents paquets peuvent aller dans differents niveaux de SYSLOG)

**inconvénients:**

- Puisque NetFilter est une architecture toute neuve, la plupart des anciens modules noyaux de MASQ vont devoir être réécrits. C'est à dire que pour le moment, seul le module FTP a été mis à jour et les modules suivants ont encore besoin de l'eêtre : ip_masq_cuseeme.o ip_masq_icq.o ip_masq_quake.o ip_masq_user.o ip_masq_irc.o ip_masq_raudio.o ip_masq_vdolive.o

  Il existe un document expliquant comment faire ce portage ici : *http://netfilter.kernelnotes.org/unreliable-guides/netfilter-hacking-HOWTO-5.html* `http://netfilter.kernelnotes.org/unreliable-guides/netfilter-hacking-HOWTO-5.html` , Si vous avez le temps, votre talent serait d'une grande aide pour porter ces modules plus rapidement.

Dans la version actuelle de ce HOWTO, NetFilter N'est PAS traité. Une fois que les fonctionnalité de NetFilter seront fixées, NetFilter sera traité dans -ce- HOWTO ou peut être éventuellement dans un nouveau HOWTO. D'ici là, vous pouvez vous tournez vers ces liens pour la documentation de NetFilter. Pour le moment, la configuration et la résolution des conflits et problèmes du nouveau NetFilter va être similaire à 95% à celles du IPCHAINS actuel. En raison de tout ça, ce HOWTO va être utile aux utilisateurs de firewall NetFilter et de NAT.

*http://netfilter.filewatcher.org/unreliable-guides/index.html* et plus spécifiquement *http://netfilter.filewatcher.org/unreliable-guides/NAT-HOWTO.html*

Reportez vous SVP à l' *IP Masquerade Resource* `http://ipmasq.cjb.net/` pour plus d'informations sur ces patches ou d'autres eventuels patches.

## 2.8 Configurations Requises pour IP Masquerade sous Linux 2.0.x

**** Reportez vous SVP à l' *IP Masquerade Resource*** `http://ipmasq.cjb.net/` **pour les informations les plus récentes****

- Tout ordinateur convenable. Reportez vous SVP à la section 7.1 () pour de plus amples détails.
- les sources du noyau 2.0.x disponibles ici : *http://www.kernel.org/*
  (La plupart des Linux modernes 7 () tels que RedHat 5.2 ont un noyau modulaire compilé avec toutes les options nécessaires à IP Masquerade. Dans de tels cas, il n'y a pas besoin de recompiler un nouveau noyau. Si vous UPGRADEZ votre noyau, il est possible que d'autres programmes puissent être requis et/ou mis à jour (ces programmes sont mentionnés plus loin dans ce HOWTO).
- Modules noyaux, de préférénce 2.1.85 ou mieux, disponible sur les sites suivants : *http://www.pi.se/blox/modutils/index.html* ou *ftp://ftp.ocs.com.au/pub/modutils/*

  (les modules-1.3.57 sont le minimum requis)
- Un réseau fonctionnant sous TCP/IP ou un LAN traité dans *Linux NET-3-4 HOWTO* `http://www.linuxdoc.org/HOWTO/NET3-4-HOWTO.html` et le *Network Administrator's Guide* `http://www.linuxdoc.org/LDP/nag/nag.html`
  Vous pouvez aussi regarder le document *TrinityOS* `http://www.ecst.csuchico.edu/~dranch/LINUX/`

index-linux.html#TrinityOS . TrinityOS est un guide très complet pour le réseau sous linux. Il traite de nombreux thèmes dont : l'IP MASQ, la sécurité, les DNS, DHCP, Sendmail, PPP, Diald, NFS, VPNs basés sur IPSEC et contient une section sur les performances. Il contient plus de 50 sections au total !

- La connexion de votre machine Linux à Internet est traitée dans : *Linux ISP Hookup HOWTO* http://www.linuxdoc.org/HOWTO/ISP-Hookup-HOWTO.html , *Linux PPP HOWTO* http://www.linuxdoc.org/HOWTO/PPP-HOWTO.html , *TrinityOS* http://www.ecst.csuchico.edu/~dranch/LINUX/index-linux.html#TrinityOS , *Linux DHCP mini-HOWTO* http://www.linuxdoc.org/HOWTO/mini/DHCP/index.html , *Linux Cable Modem mini-HOWTO* http://www.linuxdoc.org/HOWTO/Cable-Modem/index.html et *http://www.linuxdoc.org/HOWTO/mini/ADSL.html* http://www.linuxdoc.org/HOWTO/mini/ADSL.html
- Ipfwadm 2.3 ou supérieur est disponible ici : *ftp://ftp.xos.nl/pub/linux/ipfwadm/ipfwadm-2.3.tar.gz* Des informations supplémentaires sur les versions requises sont disponible ici : *Linux IPFWADM page* http://www.xos.nl/linux/ipfwadm/
  - Si vous voulez faire tourner IPCHAINS sur un noyau 2.0.38+, reportez vous ici : *Willy Tarreau's IPCHAINS enabler for 2.0.36* http://www-miaif.lip6.fr/willy/pub/linux-patches/ ou ici : *Rusty's IPCHAINS for 2.0.x kernels*
- Savoir configurer, compiler, et installer un nouveau noyau Linux comme expliqué ici : *Linux Kernel HOWTO* http://www.linuxdoc.org/HOWTO/Kernel-HOWTO.html
- Vous pouvez aussi appliquer plusieur patches optionnel à IP Masquerade pour lui 'apprendre' des fonctionnalités telles que :
  - port-forwarder ou re-diriger des ports TCP/IP : Avec ces utilitaires, vous pouvez faire fonctionner des programmes qui normalement ne fonctionne pas dernière un server MASQ. De plus, vous pouvez configurer le server MASQ de façon à permettre aux internautes de contacter des servers WWW, TELNET, SMTP, FTP (avec un patch), etc. internes. Reportez vous SVP à la section 6.7 () de ce HOWTO pour de plus amples informations. Voici une liste des différents patches IP Masquerade disponibles pour le noyau 2.0.x :
    - Steven Clarke's *IP PortForwarding (IPPORTFW)* - **RECOMMANDE**
    - *IP AutoForward* et *un mirroir* ftp://ftp.netis.com/pub/members/rlynch/ipautofw.tar.gz (IPAUTOFW) - *NON Recommande*
    - *REDIR* http://ipmasq.cjb.net/redir_0.7.orig.tar.gz pour TCP (REDIR) - NON Recommandé
    - *UDP redirector* (UDPRED) - NON Recommandé

    Solutions de PORTFW pour FTP: :
    - Si vous voulez utiliser le PORTFW pour renvoyer le traffic FTP vers un server FTP interne, vous aurez sans doute besoin de : *Fred Viles's FTP server patch via HTTP* ou *Fred Viles's FTP server patch via FTP* . De plus amples details sont disponibles à la section 6.7 () de ce HOWTO.

    forwarder des écrans X-Window :
    - *X-windows forwarding (DXCP)* ftp://sunsite.unc.edu/pub/Linux/X11/compress/dxpc-3.7.0.tar.gz

    module MASQ ICQ :
    - *Andrew Deryabin's ICQ MASQ module*

    forwarders PPTP (GRE) et VPNs SWAN (IPSEC) tunneling :

  * *John Hardin's VPN Masquerade forwarders* ou bien l'ancien patch uniquement pour le *PPTP* `http://ipmasq.cjb.net/ip_masq_pptp.patch.gz` .

Patches spécifiques aux jeux:

  * Glenn Lamb's *LooseUDP for 2.0.36+* `ftp://ftp.netcom.com/pub/mu/mumford/loose-udp-2.0.36.patch.gz` patch. Jetez un coup d'oeil à la *Page NAT* `http://www.alumni.caltech.edu/~dank/peer-nat.html` de Dan Kegel pour plus d'informations. De plus amples informations se trouvent à la section 6.3.1 ()et dans la section 7 ().

Reportez vous SVP à l' *IP Masquerade Resource* `http://ipmasq.cjb.net/` pour plus d'informations sur ces patches ou d'autres éventuels patches.

# 3 Installer IP Masquerade

**Si votre réseau privé contient quelqu'information vitale, vous devez soigneusement réflechir en terme de SECURITE avant d'utiliser IP Masquerade. Par defaut, IP MASQ devient une passerelle pour vous permettre d'acceder à Internet mais il peut aussi permettre à quelqu'un de s'introduire à partir d'Internet sur votre réseau interne.** Une fois que vous avez IP MASQ qui fonctionne, il est VIVEMENT recommandé d'utiliser un jeu de règles du sécurité largement supérieur, que nous appellerons STRONG (STRONG IPFWADM/IPCHAINS firewall ruleset en anglais) dans la suite. Vous pouvez vous reportez aux sections 6.4 () et 6.5 () plus loin dans le texte pour plus de détails.

## 3.1 Compiler un noyau avec les fonctionnalités d'IP Masquerade

**Si votre distribution Linux possède déjà toutes les fonctions nécessaires tels que :**

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

et que tous les modules relatif à MASQ y soient compilés (la plupart des noyaux modulaires vont avoir ce dont vous avez besoin), vous N'aurez PAS besoin de recompiler un noyau. Si vous ne savez pas si votre distribution Linux est pret pour MASQ, reportez vous à la section 7 (). Si vous ne faites pas confiance à cette liste, ou que votre distribution n'y est pas listée, essayez les tests suivants :

- logguez vous sur votre machine linux et lancez la commande suivante : "ls /proc/sys/net/ipv4".
- Regardez si les fichiers tels que "ip_forward", "ip_masq_debug", "ip_masq_udp_dloose"(optionnel), et "ip_always_defrag"(optionnel) existent.

Si oui, c'est que votre noyau est fin pret !

Si vous ne trouvez aucun des fichiers précités ou si votre distribution ne permet pas l'IP Masquerading par défaut, SUPPOSEZ QU'IL NE PERMET PAS l'utilisation de MASQ par defaut ! Dans ce cas, vous allez devoir compiler un noyau... mais ne vous inquietez pas, ce n'est pas difficile.

Que la compatibilité soit native ou pas sur votre distribution, la lecture de cette section est VIVEMENT recommandée parce qu'elle contient d'autres informations utiles.

### 3.1.1 Noyaux Linux 2.2.x

**Reportez vous à la section 2.5 ()pour les logiciels nécessaires, les patches, etc.**

- D'abord, vous aurez besoin des sources du noyau 2.2.x (de préférence la dernière version 2.2.16 ou mieux) NB #1: Les noyaux Linux 2.2.x inférieurs à 2.2.16 ont un trou de securité (TCP root exploit) et les versions inférieurs à 2.2.11 ont un bug de fragmentation dans IPCHAINS. Pour cette raison, les personnes qui utilisent des jeux de règles IPCHAINS très restrictives sont attaquables. Mettez à jour votre noyau vers une version corrigée.

  NB #2: Pendant les mises à jour des noyaux 2.2.x, les options de compilations n'ont cessé de changer. Ici nous allons vous montrer les réglages pour la version 2.2.15. Si vous utilisez une version antérieure du noyau, les dialogues peuvent paraître différents. Nous vous recommandons de faire la mise à jour vers la dernière version du noyau en raison des nouvelles fonctionnalités et de la stabilité accrue qu'elle procure.

- Ne soyez pas effrayé si c'est la première fois que vous compilez un noyau. En fait c'est plutôt facile et plusieurs URLs que vous trouverez dans la section 2.5 () traitent de ça.

- Decompressez les sources du noyau dans le répertoire `/usr/src/` en utilisant la commande `tar xvzf linux-2.2.x.tar.gz -C /usr/src` ou "x" représente la version de votre noyau 2.2. Une fois ceci terminé, vérifiez que le dossier ou le lien symbolique vers `/usr/src/linux/` existe bien.

- Appliquez tous les patches, optionnels ou non, au source de votre noyau. Pour le 2.2.1, IP Masq n'a pas pas besoin de patch pour fonctionner correctement. Des fonctionnalités telles que PPTP ou le forwarding de X-Window sont optionnels. Vous pouvez vous reporter à la section 2.5 () pour les URLs et à l' *IP Masquerade Resources* pour les informations les plus récentes et les liens vers les patches.

- Voici les options de compilations MINIMALES dont vous allez avoir besoin lors de la compilation de votre noyau. Vous aurez aussi besion de configurer votre noyau pour utiliser vos interfaces reseaux. Reportez vous au *Linux Kernel HOWTO* `http://www.linuxdoc.org/HOWTO/Kernel-HOWTO.html` et le fichier README qui se trouve dans le dossier des sources du noyau pour de plus amples informations sur la compilation du noyau.

  Répondez par ***YES ou NO*** aux questions suivantes. Toutes les options ne seront pas disponibles si votre noyau n'est pas patché convenablement comme décrit ci-dessous dans ce HOWTO :

```
* Prompt for development and/or incomplete code/drivers (CONFIG_EXPERIMENTAL) [Y/n/?]
  - YES: Bienque non requis par IP MASQ, cette option permet au noyau de creer les
  modules MASQ et d'activer l'option 'port forwarding'

-- Les options ne correspondant a MASQ sont omis --

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

-- Les options ne correspondant a MASQ sont omis --

* Networking support (CONFIG_NET) [Y/n/?]
  - YES: Active les capacites reseau

-- Les options ne correspondant a MASQ sont omis --

* Sysctl support (CONFIG_SYSCTL) [Y/n/?]
```

```
  - YES: vous donne le pouvoir d'activer/desactiver des options tels que le forwarding,
   les IP dynamiques, le LooseUDP, etc.

-- Les options ne correspondant a MASQ sont omis --

* Packet socket (CONFIG_PACKET) [Y/m/n/?]
  - YES: Bienque ca soit OPTIONNEL, il est recommande d'activer cette fontionnalite
  qui permet d'utiliser TCPDUMP pour
  debugguer les eventuels problems d'IP MASQ

* Kernel/User netlink socket (CONFIG_NETLINK) [Y/n/?]
  - YES: Bienque ca soit OPTIONNEL, cette fonctionnalite permet de creer des logs
  des problemes du firewall avance tel que le routage des messages etc.

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

-- Les options ne correspondant a MASQ sont omis --

* Network firewalls (CONFIG_FIREWALL) [Y/n/?]
  - YES: Permet de configurer le noyau avec l'utilitaire de firewall IPCHAINS

* Socket Filtering (CONFIG_FILTER) [Y/n/?]
  - OPTIONAL:  Bienque cette option n'ai rien a voir avec IPMASQ, si vous
  comptez installer un serveur DHCP sur votre reseau interne, vous AUREZ besoin
  de cette option.

* Unix domain sockets (CONFIG_UNIX) [Y/m/n/?]
  - YES: Active les mecanismes de sockets TCP/IP d'UNIX.

* TCP/IP networking (CONFIG_INET) [Y/n/?]
  - YES: Active les protocoles TCP/IP

-- Les options ne correspondant a MASQ sont omis --

* IP: advanced router (CONFIG_IP_ADVANCED_ROUTER) [Y/n/?]
  - YES: Permet de configurer les options avances de MASQ que nous verrons plus loins


* IP: policy routing (CONFIG_IP_MULTIPLE_TABLES) [N/y/?]
  - NO: Pas necessaire pour MASQ mais les utilisateurs qui ont besoin de fonctions avancees telles
  que le source address-based TCP/IP ou le routage par TOS doivent activer cette option.

* IP: equal cost multipath (CONFIG_IP_ROUTE_MULTIPATH) [N/y/?]
  - NO: Pas necessaire pour les fonctions usuelles de MASQ

* IP: use TOS value as routing key (CONFIG_IP_ROUTE_TOS) [N/y/?]
  - NO:  Pas necessaire pour les fonctions usuelles de MASQ

* IP: verbose route monitoring (CONFIG_IP_ROUTE_VERBOSE) [Y/n/?]
  - YES: Necessaire si vous voulez utiliser les codes de routage pour eliminer les paquets
  IP spoofes (vivement recommande) et si vous voulez les mettres dans les logs.

* IP: large routing tables (CONFIG_IP_ROUTE_LARGE_TABLES) [N/y/?]
  - NO:  Pas necessaire pour les fonctions usuelles de MASQ
```

```
* IP: kernel level autoconfiguration (CONFIG_IP_PNP) [N/y/?] ?
  - NO:  Pas necessaire pour les fonctions usuelles de MASQ

* IP: firewalling (CONFIG_IP_FIREWALL) [Y/n/?]
  - YES: Active les capacites de firewalling.

* IP: firewall packet netlink device (CONFIG_IP_FIREWALL_NETLINK) [Y/n/?]
  - OPTIONAL: Bienqu'OPTIONNELLE, cette fonction permet a IPCHAINS de copier quelques paquets
  vers l'utilitaire UserSpace pour des verifications supplementaires.

* IP: transparent proxy support (CONFIG_IP_TRANSPARENT_PROXY) [N/y/?]
  - NO:  Pas necessaire pour les fonctions usuelles de MASQ

* IP: masquerading (CONFIG_IP_MASQUERADE) [Y/n/?]
  - YES: Permet a IP Masquerade de readresser certains paquets TCP/IP specifiques de l'interieur
  vers l'exterieur

* IP: ICMP masquerading (CONFIG_IP_MASQUERADE_ICMP) [Y/n/?]
  - YES: Permet de masquerader les paquets ICMP de ping (dans tous les cas, les codes d'erreur
  d'ICMP sont MASQues). Cette fonction est importante pour regler les problemes de connexion.

* IP: masquerading special modules support (CONFIG_IP_MASQUERADE_MOD) [Y/n/?]
  - YES: Bienqu'OPTIONNELLE, cette option permet d'activer plus loin le port forwarding de
  TCP/IP qui permet aux ordinateurs exterieurs de ce connecter vers des machines MASQuees
  specifiques (donc internes).

* IP: ipautofw masq support (EXPERIMENTAL) (CONFIG_IP_MASQUERADE_IPAUTOFW) [N/y/m/?]
  - NO: IPautofw est une methode heritee du port forwardinf. C'est essentiellement du vieux
  code qui est reconnu pour avoir des problemes. NON recommande.

* IP: ipportfw masq support (EXPERIMENTAL) (CONFIG_IP_MASQUERADE_IPPORTFW) [Y/m/n/?]
  - YES: Active IPPORTFV qui permet a des ordinateurs externe se trouvant sur Internet de
  communiquer avec un ordinateur MASQue interne specifique. Cette fonctionnalite est typiquement
  utilisee pour acceder a des serveurs SMTP, TELNET et WWW. Le port forwarding pour le FTP aura
  besoin d'un patch supplementaire dont nous avons donne la description dans la FAQ de ce HOWTO.
  Des informations supplementaires sont disponibles dans la section Forwards de ce HOWTO.

* IP: ip fwmark masq-forwarding support (EXPERIMENTAL) (CONFIG_IP_MASQUERADE_MFW) [Y/m/n/?]
  - OPTIONAL: C'est une nouvelle methode pour faire du PORTFW. Avec elle, IPCHAINS peut marquer
  les paquets sur lesquels il faut faire du travail supplementaire. Avec l'utilitaire UserSpace,
  qui ressemble a IPMASQADM ou IPPORTFW, IPCHAINS pourra alors automatiquement readresser les paquets.
  Pour le moment, cette partie du code est moins testee que PORTFW mais reste neanmoins tres
  prometteur. Nous vous recommandons d'utiliser pour le l'instant IPMASQADM et IPPORTFW. Si vous
  avez des reflexions sur MFW, envoyez les moi par email SVP.

* IP: optimize as router not host (CONFIG_IP_ROUTER) [Y/n/?]
  - YES: Optimise le noyau pour le reseau bienque nous ne sachions pas si les gains de performance
  sont significatives ou pas.

* IP: tunneling (CONFIG_NET_IPIP) [N/y/m/?]
  - NO: OPTIONNEL pour le tunneling IPIP a traver IP Masq. Si vous avez besoin de fonctionnalites
  VPN/tunneling, il est recommande d'utiliser soit les tunnels GRE soit les tunnels IPSEC

* IP: GRE tunnels over IP (CONFIG_NET_IPGRE) [N/y/m/?]
```

```
  - NO: OPTIONNEL. Permet l'activation de tunnels GRE et PPTP a travers IP MASQ.

  -- Les options ne correspondant a MASQ sont omis --

* IP: TCP syncookie support (not enabled per default) (CONFIG_SYN_COOKIES) [Y/n/?]
  - YES: VIVEMENT recommande pour la securite TCP/IP de base.

  -- Les options ne correspondant a MASQ sont omis --

* IP: Allow large windows (not recommended if <16Mb of memory) * (CONFIG_SKB_LARGE) [Y/n/?]
  - YES: Ceci est recommande pour optimiser les fenetres TCP de Linux

  -- Les options ne correspondant a MASQ sont omis --

* Network device support (CONFIG_NETDEVICES) [Y/n/?]
  - YES: active la sous couche materielle du reseau sous Linux

  -- Les options ne correspondant a MASQ sont omis --

* Dummy net driver support (CONFIG_DUMMY) [M/n/y/?]
  - YES:  Bienqu'OPTIONNELLE, cette option peut aider pendant le debuggage

== N'oubliez pas d'activer les drivers de votre carte reseau !! ==

  -- Les options ne correspondant a MASQ sont omis --

== N'oubliez pas d'actiner la comptabiliter PPP/SLIP si vous voulez un modem RPC ou PPPoE/DSL !! ==

  -- Les options ne correspondant a MASQ sont omis --

* /proc filesystem support (CONFIG_PROC_FS) [Y/n/?]
  - YES: Necessaire pour activer le system de forwarding sous Linux
```

NB: Nous n'avons activé ici que les options nécessaires pour l'IP Masquerade. Vous devez sélectionner en plus les options spécifiques à votre installation.

- Apres la compilation du noyau, vous devrez compiler et installer les modules IP MASQ grâce aux commandes :

  ```
  make modules; make modules_install
  ```

- Vous devrez ensuite ajouter quelques lignes à votre fichier `/etc/rc.d/rc.local` pour charger automatiquement les modules IP Masquerades et activer IP MASQ après chaque redémarrage :

  ```
  .
  .
  .
  #rc.firewall script - Lance IPMASQ et le firewall
  /etc/rc.d/rc.firewall
  .
  .
  .
  ```

### 3.1.2 Noyaux Linux 2.0.x

**Reportez vous à la section 2.7 ()pour les logiciels nécessaires, les patches, etc.**

- D'abord, vous aurez besoin des sources du noyau 2.0.x (de préférence la dernière version 2.0.38 ou mieux)
- Ne soyez pas effrayé si c'est la première fois que vous compilez un noyau. En fait c'est plutôt facile et plusieurs URLs que vous trouverez dans la section 2.5 () traitent de ca.
- Décompressez les sources du noyau dans le repertoire `/usr/src/` en utilisant la commande `tar xvzf linux-2.2.x.tar.gz -C /usr/src` ou "x" représente la version de votre noyau 2.2. Une fois ceci terminé, vérifiez que le dossier ou le lien symbolique vers `/usr/src/linux/` existe.
- Appliquez tous les patches, optionnels ou non, au source de votre noyau. Des fonctionnalites telles que PPTP ou le forwarding de X-Window sont optionnels. Vous pouvez vous reporter à la section 2.7 () pour les URLs et à l' *IP Masquerade Resources* pour les informations les plus récentes et les liens vers les patches.
- Voici les options de compilations MINIMALES dont vous allez avoir besoin lors de la compilation de votre noyau. Vous aurez aussi besion de configurer votre noyau pour utiliser vos interfaces reseaux. Reportez vous au *Linux Kernel HOWTO* `http://www.linuxdoc.org/HOWTO/Kernel-HOWTO.html` et le fichier README qui se trouve dans le dossier des sources du noyau pour de plus amples informations sur la compilation du noyau.

  Repondez par ***YES ou NO*** aux questions suivantes. Toutes les options ne seront pas disponibles si votre noyau n'est pas patché convenablement comme décrit ci-dessus dans ce HOWTO :

```
* Prompt for development and/or incomplete code/drivers (CONFIG_EXPERIMENTAL) [Y/n/?]
  - YES: cette option permet selectionner les fonctionnalites IP Masquerade

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

* Networking support (CONFIG_NET) [Y/n/?]
  - YES: Active les capacites reseau

* Network firewalls (CONFIG_FIREWALL) [Y/n/?]
  - YES: Active l'utilitaire de firewall IPFWADM

* TCP/IP networking (CONFIG_INET)
  - YES: Active les protocoles TCP/IP

* IP: forwarding/gatewaying (CONFIG_IP_FORWARD)
  - YES: Permet le forwarding et le routage des paquets - Controle par IPFWADM

* IP: syn cookies (CONFIG_SYN_COOKIES) [Y/n/?]
  - YES: VIVEMENT recommande pour la securite TCP/IP de base.

* IP: firewalling (CONFIG_IP_FIREWALL) [Y/n/?]
  - YES: Active les capacites de firewalling.

* IP: firewall packet logging (CONFIG_IP_FIREWALL_VERBOSE) [Y/n/?]
  - YES: (OPTIONNEL mais VIVEMENT recommande): Permet de rapporter les chocs contre le firewall
```

```
* IP: masquerading (CONFIG_IP_MASQUERADE [Y/n/?]
  - YES: Permet a IP Masquerade de readresser certains paquets TCP/IP specifiques de l'interieur
  vers l'exterieur

* IP: ipautofw masquerade support (EXPERIMENTAL) (CONFIG_IP_MASQUERADE_IPAUTOFW) [Y/n/?]
  - NO:  IPautofw est une methode heritee du port forwardinf. C'est essentiellement du
  vieux code qui est reconnu pour avoir des problemes. NON recommande.

* IP: ipportfw masq support (EXPERIMENTAL) (CONFIG_IP_MASQUERADE_IPPORTFW) [Y/n/?]
  - YES: Cette option est DISPONIBLE UNIQUEMENT GRACE A UN PATCH pour les noyaux 2.0.x

     Cette option permet a des ordinateurs externe se trouvant sur Internet de communiquer
     avec un ordinateur MASQue interne specifique. Cette fonctionnalite est typiquement
     utilisee pour acceder a des serveurs SMTP, TELNET et WWW. Le port forwarding pour le
     FTP aura besoin d'un patch supplementaire dont nous avons donne la description dans
     la FAQ de ce HOWTO. Des informations supplementaires sont disponibles dans la section
     Forwards de ce HOWTO.

* IP: ICMP masquerading (CONFIG_IP_MASQUERADE_ICMP) [Y/n/?]
  - YES: Permet de masquerader les paquets ICMP. Bienqu'optionnelles, de nombreux programmes
  ne vont PAS fonctionnement correctement sans cette option.

* IP: loose UDP port managing (EXPERIMENTAL) (CONFIG_IP_MASQ_LOOSE_UDP) [Y/n/?]
  - YES: Cette option est DISPONIBLE UNIQUEMENT GRACE A UN PATCH pour les noyaux 2.0.x

      Avec cette option, des ordinateurs internes (ie MASQues) pourrons jouer au
      jeux compatibles NAT sur Internet. Des details supplementaires sont donnes
      dans la section FAQ de ce HOWTO.

* IP: always defragment (CONFIG_IP_ALWAYS_DEFRAG) [Y/n/?]
  - YES: Cette option optimise les connexions IP MASQ - VIVEMENT recommande

* IP: optimize as router not host (CONFIG_IP_ROUTER) [Y/n/?]
  - YES: Optimise le noyau pour le reseau

* IP: Drop source routed frames (CONFIG_IP_NOSR) [Y/n/?]
  - YES: HIGHLY recommended for basic network security

* Dummy net driver support (CONFIG_DUMMY) [M/n/y/?]
  - YES: VIVEMENT recommande pour la securite TCP/IP de base.

* /proc filesystem support (CONFIG_PROC_FS) [Y/n/?]
  - YES: Necessaire pour activer les capacites de forwarding de Linux
```

NB: Nous n'avons activé ici que les options nécessaire pour IP Masquerade. Vous devez sélectionner en plus les options spécifiques à votre installation.

- Après la compilation du noyau, vous devrez compiler et installer les modules IP MASQ grâce aux commandes :

  ```
  make modules; make modules_install
  ```

- Vous devrez ensuite ajouter quelques lignes à votre fichier `/etc/rc.d/rc.local` pour charger automatiquement les modules IP Masquerades et activer IP MASQ après chaque redémarrage :

```
.
.
.
#rc.firewall script - Lance IPMASQ et le firewall
/etc/rc.d/rc.firewall
.
.
.
```

### 3.1.3 Noyaux Linux 2.3.x / 2.4.x

**Les noyaux 2.3.x et 2.4.x ne sont PAS traités dans ce HOWTO pour le moment. Reportez vous à la section 2.6 () pour les URLs etc. jusqu'à ce que ces noyaux soient traités dans un nouveau HOWTO.**

## 3.2 Affecter des adresses IP privées au LAN interne

Puisque toutes les machines **INTERNES MASQées** ne devraient pas avoir d'adresses IP officielles, il doit exister une façon spécifique et reconnue d'affecter des adresses à ces machines sans entrer en conflit avec l'adresse IP de quelqu'un d'autre.

Tiré de la FAQ IP Masquerade originelle :

*RFC 1918* est un document officiel traitant des adresses IP qui doivent être utilisées pour des réseaux non-connectés ou "privés". Il y a 3 blocs de nombres mis de côtés exprès dans ce but.

```
Section 3: L'espace des Adresses Privees



L'Internet Assigned Numbers Authority (IANA) a reserve les trois blocs d'adresses IP suivants
pour les reseaux prives :

            10.0.0.0        -   10.255.255.255
            172.16.0.0      -   172.31.255.255
            192.168.0.0     -   192.168.255.255

Le premier bloc sera designe comme le "24-bit block", le second comme "20-bit block", et le
dernier comme "16-bit block". Remarquez que le premier bloc n'est rien d'autres qu'un simple reseau
de Classe A, alors que le second est un espace de 16 reseaux contigus de classe B, et le troisieme
est un blocs de 255 reseaux contigus de Classe C.
```

Je préfère utiliser le réseau 192.168.0.0 avec un masque de sous-réseau de classe C : 255.255.255.0 et ce HOWTO refléte cette préférence. Mais, chacun des réseaux privés ci-dessus est valide. Assurez vous simplement d'utiliser le bon masque de sous-réseau.

Donc, si vous utiliser le réseau de Classe C, vous devrez numéroter vos machines TCP/IP ainsi : 192.168.0.1, 192.168.0.2, 192.168.0.3, ..., 192.168.0.x

192.168.0.1 est habituellement la passerelle interne ou la machine MASQ Linux. Notez aussi ques les adresses 192.168.0.0 et 192.168.0.255 sont les adresses du réseau et de broadcast respectivement (et sont donc RESERVES). Evitez d'utiliser ces adresses sur vos machines sinon votre réseau risque de ne pas fonctionner correctement.

## 3.3 Politiques de configuration de l'IP FORWARDING

A partir d'ici, vous devrez avoir votre noyau et les autres packages nécessaires d'installés. Toutes les adresses IP du réseau, la passerelle, et le DNS devront aussi être configurés dans votre serveur Linux MASQ. Si vous ne savez pas configurer vos cartes réseau, reportez vous SVP aux HOWTOs listés dans les sections 2.7 () ou 2.5 ().

Maintenant, la seule chose qui reste à faire, c'est de configurer l'IP firewalling pour permettre le FORWARD et le MASQUERADE des paquets appropriés vers les machines appropriées :

> ** Ceci peut être accomplis de différentes façons. Les suggestions et les examples suivants ont fonctionné chez moi, mais vous aurez peut-être des besoins ou des idées différents.

> ** Cette section fournit seulement le MINIMUM de règles de firewall pour faire fonctionner l'IP Masquerade. Une fois que vous aurez testé IP MASQ (comme décrit plus loin dans ce HOWTO), reportez vous aux sections 6.4 () et 6.5 () pour des jeux de règles de firewalling plus sûres. Vous pouvez aussi lire en plus les manuels de IPFWADM (2.0.x) et/ou IPCHAINS(2.2.x) pour de plus amples détails.

### 3.3.1 Noyaux Linux 2.2.x

NB : **IPFWADM n'est plus un utilitaire de firewall** qui permette de manipuler les règles d'IP Masquerade pour les noyaux 2.1.x et 2.2.x. Ces nouveaux noyaux utilisent maintenant l'utilitaire IPCHAINS. Pour de plus amples détails sur les raisons de ce changement, vous pouvez vous reporter à la section 7 ().

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

```
#!/bin/sh
#
# rc.firewall - test IP Masquerade naif pour les noyaux 2.1.x et 2.2.x
#               avec IPCHAINS
#
# Charge les modules necessaires a IP MASQ
#
#   NB: Charger uniquement les modules IP MASQ dont vous avez besoin. Tous les modules
#   IP MASQ actuels sont montres ci-dessous mais sont commentes pour les empecher
#   de se charger.

# Necessaire pour le chargement initial des modules
#
/sbin/depmod -a

# Permet le masquerading correct des transfert de fichier par FTP avec la methode PORT
/sbin/modprobe ip_masq_ftp

# Permet le masquerading de RealAudio par UDP. Sans ce module,
#       RealAudio FONCTIONNERA mais en mode TCP. Ce qui peu causer une baisse
#       dans la qualite du son
#
#/sbin/modprobe ip_masq_raudio

# Permet le masquerading des transferts de fichier par DCC pour les IRC
```

```
#/sbin/modprobe ip_masq_irc


# Permet le masquerading de Quake et QuakeWorld par defaut. Ce module est
#   necessaire pour les utilisateurs multiples derriere un server Linux MASQ. Si vous voulez jouer
#   a Quake I, II, et III, utilisez le second exemple.
#
#   NB:  si vous rencontrez des ERREURs lors de chargement du module QUAKE, c'est que vous utilisez
#   un ancien noyau buggue. Mettez a jour votre noyau pour supprimer l'erreur.
#
#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 26000,27000,27910,27960


# Permet le masquerading du logiciel CuSeeme pour la video conference
#
#/sbin/modprobe ip_masq_cuseeme

# Permet le masquerading du logiciel VDO-live pour la video conference
#
#/sbin/modprobe ip_masq_vdolive


#CRITIQUE:  Active l'IP forwarding puisqu'il est desactive par defaut
#
#           Utilisateurs Redhat: vous pourrez essayer en changeant les options dans
#                          /etc/sysconfig/network de:
#
#                       FORWARD_IPV4=false
#                             a
#                       FORWARD_IPV4=true
#
echo "1" > /proc/sys/net/ipv4/ip_forward


#CRITIQUE:  Active automatiquement l'IP defragmenting puisqu'il est desactive par defaut
#           dans les noyaux 2.2.x. Ceci etait une option de compilation mais ca a change
#           depuis le noyau 2.2.12
#
echo "1" > /proc/sys/net/ipv4/ip_always_defrag


# Utilisateurs d'IP Dynamiques:
#
#   Si vous recevez votre adresse IP de maniere dynamique a partir d'un server SLIP, PPP,
#   ou DHCP, activez option suivante qui active le hacking (au bon sens du terme NDT) des
#   adresses IP dynamique dans IP MASQ, rendant ainsi les choses plus faciles pour les
#   programmes du type Diald.
#
#echo "1" > /proc/sys/net/ipv4/ip_dynaddr
```

```
# Active le patch LooseUDP dont certains jeux reseaux ont besoin
#
#  Si vous etes en train d'essayer de faire fonctionner un jeu sur Internet au travers votre
#  serveur MASQ, et vous l'avez configure le mieux que vous pouviez mais que ca fonctionne
#  toujours pas, essayez d'activer cette option (en supprimant le # en debut de ligne).
#  Cette option est desactivee par defaut pour eviter une probable vulnerabilite au port
#  scanning UDP en interne.
#
#echo "1" > /proc/sys/net/ipv4/ip_masq_udp_dloose


# MASQ timeouts
#
#  timeout de 2 heures pour les sessions TCP
#  timeout de  10 sec pour le traffic apres que le paquet TCP/IP "FIN" est recu
#  timeout de 160 sec pour le traffic UDP (Important pour les utilisateur d'ICQ MASQues)
#
/sbin/ipchains -M -S 7200 10 160


# DHCP:  Pour les personnes qui recoivent leur adresse IP externe par DHCP ou
#        BOOTP tels que les utilisateurs d'ADSL ou Cable, il est necessaire de lancer cette
#        commande avec celle du 'deny'. "nom_interface_cliente_bootp"
#        doit etre remplace par le nom de l'interface qui va recevoir l'adresse externe par
#        le serveur DHCP/BOOTP. C'est souvent quelquechose du style "eth0",
#        "eth1", etc.
#
#        Cet exemple est commante (desactive) ici :
#
#/sbin/ipchains -A input -j ACCEPT -i nom_interface_cliente_bootp -s 0/0 67 -d 0/0 68 -p udp

# Active l'IP forwarding et Masquerading simpliste
#
#  NB:    L'exemple suivant est donne pour le LAN interne 192.168.0.x avec un masque
#         de sous reseau de 255.255.255.0 soit un masque de sous reseau "24 bits"
#         connecte a Internet par l'interface eth0.
#
#         ** Changez les adreesse reseau et masque de sous reseau, et l'interface de
#         ** votre connexion a Internet de telle sorte qu'ils correspondent aux reglages de votre LAN
#
/sbin/ipchains -P forward DENY
/sbin/ipchains -A forward -i eth0 -s 192.168.0.0/24 -j MASQ
```

Une fois que vous aurez terminé de rédiger les règles de /etc/rc.d/rc.firewall, rendez ce dernier exécutable en tapant `chmod 700 /etc/rc.d/rc.firewall`

Maintenant que vos règles de firewall sont prêts, vous devez faire en sorte qu'ils soient actifs après chaque redémarrage. Vous pouvez soit décider de le lancer à la main à chaque fois (une vraie galère) ou bien de le rajouter dans vos scripts de boot. Nous vous montrons comment faire pour chacune des deux methodes ci-dessous :

- Pour les distribs RedHat ou derivées de la RedHat:
- Il y a deux manières de faire charger des trucs dans RedHat : /etc/rc.d/rc.local ou le mettre le script

d'init dans /etc/rc.d/init.d/. La première méthode est la plus simple. Tout ce que vous avez à faire c'est de rajouter cette ligne :

- echo "Chargement des regles de rc.firewall..." /etc/rc.d/rc.firewall

à la fin de votre /etc/rc.d/rc.local et c'est tout. Le problème de cette approche est que si vous tournez avec le jeu de règles STRONG du firewall, le firewall n'est pas actif avant les dernières phases du démarrage. La meilleure approche est d'avoir le firewall chargé juste après que le réseau est lancé. A partir de ce point, ce HOWTO ne traite que de l'approche /etc/rc.d/rc.local. Si vous voulez le système STRONG, je vous recommande de vous reportez à la section 10 de TrinityOS dont vous trouverez le lien à la fin de ce HOWTO.

- Slackware :
- Il y a deux manière de faire charger des trucs dans la Slackware: /etc/rc.d/rc.local ou modifier le fichier /etc/rc.d/rc.inet2. La première methode est la plus simple. Tout ce que vous avez à faire c'est de rajouter cette ligne :

  - echo "Chargement des regles de rc.firewall..."/etc/rc.d/rc.firewall

  à la fin de votre /etc/rc.d/rc.local et c'est tout. Le problème de cette aproche est que si vous tournez avec les jeux de règles STRONG du firewall, le firewall n'est pas actif avant les dernières phases du demarrage. La meilleure approche est d'avoir le firewall chargé juste après que le réseau est lancé. A partir de ce point, ce HOWTO ne traite que de l'approche /etc/rc.d/rc.local. Si vous voulez le système STRONG, je vous recommande de vous reporter à la section 10 de TrinityOS dont vous trouverez le lien à la fin de ce HOWTO.

**Remarques sur la manière dont les utilisateurs doivent s'y prendre s'il veulent modifier les règles de firewall que nous avons vues ci-dessus :**

Vous pouvez aussi activer l'IP Masquerading sur une base de cas par cas suivant la machine au lieu de la methode ci-dessus qui active le reseau TCP/IP entier. Par exemple, disons que je veux que seuls les machines 192.168.0.2 et 192.168.0.8 puissent accéder à Internet et AUCUN autre ordinateur interne. Je changerais les règles dans la section "Active l'IP forwarding et Masquerading simpliste" (voir ci-dessus) dans les règles qui se trouvent dans le fichier /etc/rc.d/rc.firewall.

```
#!/bin/sh
#
# Active l'IP forwarding et Masquerading simpliste
#
#  NB:    L'exemple suivant est donne pour l'activation de l'IP Masquerading pour les
#         machines 192.168.0.2 et 192.1680.0.8 avec un masque
#         de sous reseau de 255.255.255.0 soit un masque de sous reseau "24 bits"
#         connecte a Internet par l'interface eth0.
#
#         ** Changez les adreesse reseau et masque de sous reseau, et l'interface de
#         ** votre connexion a Internet de telle sorte qu'ils correspondent aux reglages de votre LAN
#
/sbin/ipchains -P forward DENY
/sbin/ipchains -A forward -i eth0 -s 192.168.0.2/32 -j MASQ
/sbin/ipchains -A forward -i eth0 -s 192.168.0.8/32 -j MASQ
```

**Erreurs courantes :**

Une erreur qui paraît courante pour les nouveaux utilisateurs d'IP Masq est de faire de la commande suivante :

/sbin/ipchains -P forward masquerade

la premiere commande.

Ne faites **PAS** du MASQUERADING votre politique par defaut. Sinon une personne qui peut manipuler ses tables de routage sera capable de s'infiltrer directement à travers votre passerelle, en l'utilisant pour masquerader sa propre identité !

Encore une fois, vous pouvez ajouter ces lignes dans votre fichier `/etc/rc.d/rc.firewall`, ou bien dans l'un de vos autres fichiers rc de votre convenance, ou bien le lancer manuellement à chaque fois que vous avez besoin de l'IP Masquerade.

Reportez vous SVP aux sections 6.4 () et 6.5 () pour un guide detaillé d'IPCHAINS et un exemple de règles STRONG pour IPCHAINS. Pour des détails supplémentaires sur l'utilisation d'IPCHAINS, vous pouvez vous reporter au site principal d'IPCHAINS *http://netfilter.filewatcher.org/ipchains/* ou au site *Linux IP CHAINS HOWTO Backup* .

### 3.3.2 Noyau Linux 2.0.x

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

```
# rc.firewall - Initial SIMPLE IP Masquerade setup for 2.0.x kernels using
#               IPFWADM
# rc.firewall - test IP Masquerade naif pour les noyaux 2.0.x
#               avec IPFWADM
#
# Charge les modules necessaires a IP MASQ
#
#   NB: Charger uniquement les modules IP MASQ dont vous avez besoin. Tous les modules IP MASQ
#       actuels sont montres ci-dessous mais sont commentes pour les empecher de se charger.
#
# Necessaire pour le chargement initial des modules
#
/sbin/depmod -a

# Permet le masquerading correct des transfert de fichier par FTP avec la methode PORT
/sbin/modprobe ip_masq_ftp

# Permet le masquerading de RealAudio par UDP. Sans ce module,
#       RealAudio FONCTIONNERA mais en mode TCP. Ce qui peu causer une baisse
#       dans la qualite du son
#
#/sbin/modprobe ip_masq_raudio

# Permet le masquerading des transferts de fichier par DCC pour les IRC
#/sbin/modprobe ip_masq_irc


# Permet le masquerading de Quake et QuakeWorld par defaut. Ce module est
#   necessaire pour les utilisateurs multiples derriere un server Linux MASQ. Si vous voulez jouer
#   a Quake I, II, et III, utilisez le second exemple.
#
```

```
#   NB:  si vous rencontrez des ERREURs lors de chargement du module QUAKE, c'est que vous utilisez
#   un ancien noyau buggue. Mettez a jour votre noyau pour supprimer l'erreur.
#
#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 26000,27000,27910,27960


# Permet le masquerading du logiciel CuSeeme pour la video conference
#
#/sbin/modprobe ip_masq_cuseeme

# Permet le masquerading du logiciel VDO-live pour la video conference
#
#/sbin/modprobe ip_masq_vdolive


#CRITIQUE:  Active l'IP forwarding puisqu'il est desactive par defaut
#
#           Utilisateurs Redhat: vous pourrez essayer en changeant les options dans
#                          /etc/sysconfig/network de:
#
#                       FORWARD_IPV4=false
#                             a
#                       FORWARD_IPV4=true
#
echo "1" > /proc/sys/net/ipv4/ip_forward


#CRITIQUE:  Active automatiquement l'IP defragmenting puisqu'il est desactive par defaut
#           dans les noyaux 2.2.x. Ceci etait une option de compilation mais ca a change
#           depuis le noyau 2.2.12
#
echo "1" > /proc/sys/net/ipv4/ip_always_defrag


# Utilisateurs d'IP Dynamiques:
#
#   Si vous recevez votre adresse IP de maniere dynamique a partir d'un server SLIP, PPP,
#   ou DHCP, activez option suivante qui active le hacking (au bon sens du terme NDT) des
#   adresses IP dynamique dans IP MASQ, rendant ainsi les choses plus faciles pour les
#   programmes du type Diald.
#
#echo "1" > /proc/sys/net/ipv4/ip_dynaddr


# MASQ timeouts
#
#   2 hrs timeout for TCP session timeouts
#  10 sec timeout for traffic after the TCP/IP "FIN" packet is received
#  160 sec timeout for UDP traffic (Important for MASQ'ed ICQ users)
#
/sbin/ipchains -M -S 7200 10 160
```

```
# DHCP:  For people who receive their external IP address from either DHCP or
#        BOOTP such as ADSL or Cablemodem users, it is necessary to use the
#        following before the deny command.  The "bootp_client_net_if_name"
#        should be replaced the name of the link that the DHCP/BOOTP server
#        will put an address on to?  This will be something like "eth0",
#        "eth1", etc.
#
#        This example is currently commented out.
#
#
#/sbin/ipchains -A input -j ACCEPT -i bootp_clients_net_if_name -s 0/0 67 -d 0/0 68 -p udp


# Active l'IP forwarding et Masquerading simpliste
#
#  NB:    L'exemple suivant est donne pour le LAN interne 192.168.0.x avec un masque
#         de sous reseau de 255.255.255.0 soit un masque de sous reseau "24 bits"
#         connecte a Internet par l'interface eth0.
#
#         ** Changez les adreesse reseau et masque de sous reseau, et l'interface de
#         ** votre connexion a Internet de telle sorte qu'ils correspondent aux reglages de votre LAN
#
/sbin/ipfwadm -F -p deny
/sbin/ipfwadm -F -a m -W eth0 -S 192.168.0.0/24 -D 0.0.0.0/0
```

Une fois que vous aurez terminé de rédiger les règles de /etc/rc.d/rc.firewall, rendez le exécutable en tapant `chmod 700 /etc/rc.d/rc.firewall`

Maintenant que vos règles de firewall sont prêts, vous devez faire en sorte qu'ils soient actifs après chaque redémarrage. Vous pouvez soit décider de le lancer à la main à chaque fois (une vraie galère) ou bien de le rajouter dans vos scripts de boot. Nous vous montrons comment faire pour chacune des deux methodes ci-dessous :

- Pour les distribs RedHat ou derivées de la RedHat:
- Il y a deux manières de faire charger des trucs dans RedHat : /etc/rc.d/rc.local ou le mettre le script d'init dans /etc/rc.d/init.d/. La première methode est la plus simple. Tout ce que vous avez à faire c'est de rajouter cette ligne :

  - echo "Chargement des regles de rc.firewall..." /etc/rc.d/rc.firewall

  à la fin de votre /etc/rc.d/rc.local et c'est tout. Le problème de cette approche est que si vous tournez avec les règles de STRONG firewall, le firewall n'est pas actif avant les dernières phases du démarrage. La meilleure approche est d'avoir le firewall chargé juste après que le reseau est lancé. A partir de ce point, ce HOWTO ne traite que de l'approche /etc/rc.d/rc.local. Si vous voulez le système STRONG, je vous recommande de vous reporter à la section 10 de TrinityOS dont vous trouverez le lien à la fin de ce HOWTO.

- Slackware :
- Il y a deux manières de faire charger des trucs dans la Slackware: /etc/rc.d/rc.local ou modifier le fichier /etc/rc.d/rc.inet2. La première methode est la plus simple. Tout ce que vous avez à faire c'est de rajouter cette ligne :

– echo "Chargement des regles de rc.firewall..."/etc/rc.d/rc.firewall

à la fin de votre /etc/rc.d/rc.local et c'est tout. Le problème de cette aproche est que si vous tournez avec les règles de STRONG firewall, le firewall n'est pas actif avant les dernières phases du démarrage. La meilleure approche est d'avoir le firewall chargé juste apres que le reseau est lancé. A partir de ce point, ce HOWTO ne traite que de l'approche /etc/rc.d/rc.local. Si vous voulez le système STRONG, je vous recommande de vous reporter à la section 10 de TrinityOS dont vous trouverez le lien à la fin de ce HOWTO.

**Remarques sur la manière dont les utilisateurs doivent s'y prendre s'il veulent modifier les règles de firewall que nous avons vues ci-dessus :**

Vous pouvez aussi activer l'IP Masquerading sur une base de cas par cas suivant la machine au lieu de la methode ci-dessus qui active le reseau TCP/IP entier. Par exemple, disons que je veux que seuls les machines 192.168.0.2 et 192.168.0.8 puisssent accéder à Internet et AUCUN autre ordinateur interne. Je changerais les règles dans la section "Active l'IP forwarding et Masquerading simpliste" (voir ci-dessus) dans les règles qui se trouvent dans le fichier /etc/rc.d/rc.firewall.

```
#!/bin/sh
#
# Active l'IP forwarding et Masquerading simpliste
#
#  NB:    L'exemple suivant est donne pour l'activation de l'IP Masquerading pour les
#         machines 192.168.0.2 et 192.1680.0.8 avec un masque
#         de sous reseau de 255.255.255.0 soit un masque de sous reseau "24 bits"
#         connecte a Internet par l'interface eth0.
#
#         ** Changez les adresse reseau et masque de sous reseau, et l'interface de
#         ** votre connexion a Internet de telle sorte qu'ils correspondent aux reglages de votre LAN
#
/sbin/ipfwadm -F -p deny
/sbin/ipfwadm -F -a m -W eth0 -S 192.168.0.2/32 -D 0.0.0.0/0
/sbin/ipfwadm -F -a m -W eth0 -S 192.168.0.8/32 -D 0.0.0.0/0
```

**Erreurs courrantes :**

Une erreur qui parait courante pour les nouveaux utilisateurs d'IP Masq est de faire de la commande suivante :

```
ipfwadm -F -p masquerade
```

la premiere commande.

Ne faites **PAS** du MASQUERADING votre politique par défaut. Sinon une personne qui peut manipuler ses tables de routage sera capable de s'infiltrer directement à travers votre passerelle, en l'utilisant pour masquerader sa propre identité !

Encore une fois, vous pouvez ajouter ces lignes dans votre fichier `/etc/rc.d/rc.firewall`, ou bien dans l'un de vos autres fichiers rc de votre convenance, ou bien le lancer manuellement à chaque fois que vous avez besoin de l'IP Masquerade.

Vous pouvez vous reporter aux sections 6.5 () et 6.4 ()pour un guide detaillé et des examples de règles STRONG pour IPCHAINS et IPFWADM.

# 4 Configurer les autres machines internes qui doivent être MASQuées

En plus des réglages d'adresses IP appropriés pour chaque machine MASQuée, vous devez régler pour chaque machine interne l'adresse IP de la passerelle (le serveur Linux MASQ) et les adresses des serveurs DNS. En général, ca découle de source. Vous entrez simplement l'adresse IP de votre serveur Linux (192.168.0.1 en général) dans le champ réservé à la passerelle.

Pour les Domain Name Service (Service de Nom de Domaine ou DNS en anglais), vous pouvez utiliser n'importe quel serveur DNS disponible. Le plus évident, c'est celui qu'utilise votre serveur Linux. Vous pouvez aussi ajouter n'importe quel domaine de recherche (facultatif).

Apres avoir reconfiguré correctement les machines internes MASQuées, n'oubliez pas de relancer leurs 'network services' (pour tenir compte des changements) ou bien de les redémarrer.

Les instructions suivantes supposent que vous utilisez un réseau de Classe C avec 192.168.0.1 comme IP pour votre serveur Linux MASQ. Rappelez vous aussi que les adresses sont des adresses 192.168.0.0 et 192.168.0.255 TCP/IP reservées.

Les plateformes ci-dessous ont été testées comme machines internes MASQuées. Voici juste un EXEMPLE de tous les systèmes d'exploitations compatibles :

- ordinateurs Apple Macintosh sous MacOS avec soit MacTCP soit Open Transport
- Commodore Amiga avect AmiTCP ou la pile AS225
- Stations Digital VAX 3520 et 3100 avec UCX (pile TCP/IP pour VMS)
- Digital Alpha/AXP sous Linux/Redhat
- IBM AIX sur un RS/6000
- IBM OS/2 (dont Warp v3)
- IBM OS400 sur un AS/400
- Linux 1.2.x, 1.3.x, 2.0.x, 2.1.x, 2.2.x
- Microsoft DOS (avec les packet drivers et le package NCSA Telnet, DOS Trumpet fonctionne partiellement)
- Microsoft Windows 3.1 (avec le package Netmanage Chameleon)
- Microsoft Windows For Workgroup 3.11 (avec le package TCP/IP)
- Microsoft Windows 95, OSR2, 98, 98se
- Microsoft Windows NT 3.51, 4.0, 2000 (workstation et server)
- Novell Netware 4.01 Server avec les services TCP/IP
- SCO Openserver (v3.2.4.2 et 5)
- Sun Solaris 2.51, 2.6, 7

## 4.1 Configuration de Microsoft Windows 95

NDT : je n'ai accès à AUCUN windows, je ne peux donc pas tester ce que je traduis. Merci de me faire parvenir les incorrections. Ceci reste valable pour toute la suite de ce document. Merci de ne pas m'en tenir rigueur.

1. Si vous n'avez pas encore installé votre carte réseau ou ses drivers, faites le maintenant. L'explication de ces étapes sort du cadre de ce document.
2. Allez dans *'Panneaux de Configuration'* –> *'Network'*.
3. Cliquez sur *Ajouter –> Protocole –> Manufacture: Microsoft –> Protocole: 'TCP/IP protocol'* Si vous ne l'avez pas encore.
4. Selectionnez le TCP/IP sur votre carte réseau de Windows95 et sélectionnez *'Proprietes'*. Maintenant allez sur *'IP Adresse IP'* et rentrez comme Adresse IP 192.168.0.x, (1 < x < 255), et comme Masque de sous-réseau 255.255.255.0
5. Maintenant sélectionnez *"Passerelle"* et ajoutez 192.168.0.1 comme passerelle dans *'Passerelle'* et cliquer sur "Ajouter".
6. Sous la languette *'Configuration DNS'*, mettez un nom pour votre machine et entrez le nom de domaine officiel. Si vous ne savez pas quel est votre nom de domaine, mettez celui de votre FAI. Maintenant, ajoutez tous les serveurs DNS que votre serveur Linux utilise (vous pouvez les trouver en général dans `/etc/resolv.conf`). En general, ces serveurs DNS sont ceux de votre FAI, bienque vous puissiez utiliser vos propres CACHING ou serveur Authoritative DNS sur votre seveur Linux MASQ. Vous pouvez aussi ajouter n'importe quel domaine de recherche (facultatif).
7. Laissez tous les autres réglages inchangés à moins que vous sachiez ce que vous faites.
8. Cliquez sur *'OK'* sur toutes les fenêtres de dialogues et rédemarrez votre ordinateur.
9. `Ping`uez le serveur Linux pour tester la connexion réseau: *'Executer'*, entrez: `ping 192.168.0.1` (Ceci est juste un test du LAN INTERNE, vous ne pouvez pas encore `ping`uer le monde exterieur.) Si vous ne recevez pas de réponses de vos PINGs, vérifiez votre configuration réseau.
10. Vous pouvez aussi créer un fichier `HOSTS` dans the C:\Windows et pouvoir ainsi utiliser des noms de machines de votre LAN sans avoir besoin de serveur DNS. Il y a un exemple appelé `HOSTS.SAM` dans le repertoire C:\windows.

## 4.2 Configuring Windows NT

NDT : je n'ai accès à AUCUN Windows, je ne peux donc pas tester ce que je traduis. Merci de me faire parvenir les incorrections. Ceci reste valable pour toute la suite de ce document. Merci de ne pas m'en tenir rigueur.

1. Si vous n'avez pas encore installé votre carte réseau ou ses drivers, faites le maintenant. L'explication de ces étapes sort du cadre de ce document.
2. Allez dans *'Panneaux de Configuration'* –> *'Network'*.
3. Cliquez sur *Ajouter –> Protocole –> Manufacture: Microsoft –> Protocole: 'TCP/IP protocol'* Si vous ne l'avez pas encore.
4. Dans la section *'Logiciels et Cartes Reseaux'*, sélectionnez le *'Protocle TCP/IP'* dans la boite de sélection *'Logiciels Reseaux Insatalles'*.

5. Dans *'Configuration TCP/IP'*, selectionnez la carte appropriee, i.e. `[1]Novell NE2000 Adapter`. Reglez ensuite l'adresse IP : 192.168.0.x (1 < x < 255), et le masque de sous-réseau : 255.255.255.0 et la Passerelle par Défaut à 192.168.0.1
6. N'activez aucune des options suivantes (à moins que vous sachiez exactement ce que vous faites) :
   - *'Configuration Automatique DHCP'* : A moins que vous ayiez un serveur DHCP sur votre réseau.
   - Rentrez n'importe quoi dans *'Sever WINS'* champs d'entrée : A moins que vous ayiez configuré un ou plusieurs serveurs WINS.
   - *Activer l'IP Forwarding* : A moins que vous routiez vers votre serveur NT et que vous sachiez vraiment -VRAIMENT- ce que vous faites PRECISEMENT.
7. Cliquez sur *'DNS'*, saisissez les informations que votre serveur Linux utilise (généralement dans /etc/resolv.conf) et cliquez sur *'OK'*quand vous avez fini.
8. Cliquez sur *'Advances'*, et DESACTIVEZ *'DNS pour la Résolution des Noms Windows'* et *'Activer la Recherche LMHOSTS'* à moins que vous sachiez ce que font ces options. Si vous voulez utiliser un fichier LMHOSTS, ils sont rangés dans C:\winnt\system32\drivers\etc.
9. Cliquez sur *'OK'* sur toutes les boites de dialogue et redémarrer votre ordinateur.
10. `Ping`uez le serveur Linux pour tester la connexion réseau: *'Executer'*, entrez: `ping 192.168.0.1` (Ceci est juste un test du LAN INTERNE, vous ne pouvez pas encore `ping`uer le monde extérieur.) Si vous ne recevez pas de réponses de vos PINGs, vérifiez votre configuration réseau.

## 4.3 Configuration de Windows for Workgroup 3.11

NDT : je n'ai accès à AUCUN Windows, je ne peux donc pas tester ce que je traduis. Merci de me faire parvenir les incorrections. Ceci reste valable pour toute la suite de ce document. Merci de ne pas m'en tenir rigueur.

1. Si vous n'avez pas encore installé votre carte réseau ou ses drivers, faites le maintenant. L'explication de ces étapes sort du cadre de ce document.
2. Installez le package TCP/IP 32b si vous ne l'avez pas encore fait.
3. Dans *'Main'/'Reglages Windows'/'Reglages Reseaux'*, cliquez sur *'Drivers'*.
4. Sélectionnez *'Microsoft TCP/IP-32 3.11b'* dans la section *'Drivers Reseaux'*, cliquez sur *'Regler'*.
5. Réglez ensuite l'adresse IP : 192.168.0.x (1 < x < 255), et le masque de sous-réseau : 255.255.255.0 et la Passerelle par Défaut a 192.168.0.1
6. N'activez aucune des options suivantes (à moins que vous sachiez exactement ce que vous faites) :
   - *'Configuration Automatique DHCP'* : A moins que vous ayiez un serveur DHCP sur votre réseau.
   - Rentrez n'importe quoi dans *'Sever WINS'* champs d'entree : A moins que vous ayiez configuré un ou plusieurs serveurs WINS.
7. Cliquez sur *'DNS'*, saisissez les informations que votre serveur Linux utilise (généralement dans /etc/resolv.conf) et cliquez sur *'OK'*quand vous avez fini.
8. Cliquez sur *'Avances'*, activez *'DNS pour la Resolution des Noms Windows'* et *'Activer la Recherche LMHOSTS'* que vous trouverez dans c:\windows.
9. Cliquez sur *'OK'* sur toutes les boites de dialogue et redémarrer votre ordinateur.

10. `Ping`uez le serveur Linux pour tester la connexion réseau: *'Executer'*, entrez: `ping 192.168.0.1` (Ceci est juste un test du LAN INTERNE, vous ne pouvez pas encore `ping`uer le monde extérieur.) Si vous ne recevez pas de réponses de vos PINGs, vérifiez votre configuration réseau.

## 4.4 Configuration des Systèmes Basés sur UNIX

1. Si vous n'avez pas encore installé votre carte réseau et recompilé votre noyau avec les drivers correspondants, faites le maintenant. L'explication de ces étapes sort du cadre de ce document.
2. Installez le réseau TCP/IP, tel que le package net-tools, si vous ne l'avez pas encore fait.
3. Changez *IPADDR* en 192.168.0.x ($1 < x < 255$), et changez ensuite *NETMASK* en 255.255.255.0, *GATEWAY* en 192.168.0.1, et *BROADCAST* en 192.168.0.255 Par exemple avec les Linux Redhat, vous pouvez modifier le fichier `/etc/sysconfig/network-scripts/ifcfg-eth0`, ou simplement le faire grâce au Tableau de Bord. Ces changement se font différemment sur les autres UNIXes tels que SunOS, BSDi, Slackware Linux, Solaris, SuSe, Debian, etc.). Reportez vous à la documentation de votre UNIX pour de plus amples informations.
4. Ajoutez votre DNS et votre domaine de recherche dans `/etc/resolv.conf` et suivant la version de votre UNIX, modifiez le fichier /etc/nsswitch.conf pour activer les DNS.
5. Vous pouvez aussi mettre à jour votre fichier `/etc/networks` suivant vos reglages.
6. Redémarrer les services consernés ou plus simplement, redémarrez votre machine.
7. Faites un `ping` : `ping 192.168.0.1` pour tester votre connexion avec la passerelle.
   (Ceci est juste un test du LAN INTERNE, vous ne pouvez pas encore `ping`uer le monde exterieur.) Si vous ne recevez pas de réponses de vos PINGs, vérifiez votre configuration réseau.

## 4.5 Configuration de DOS avec le package NCSA Telnet

1. Si vous n'avez pas encore installé votre carte réseau ou ses drivers, faites le maintenant. L'explication de ces étapes sort du cadre de ce document.
2. Chargez les drivers correspondants. Par exemple : pour la carte Ethernet NE2000 sur le port d'E/S 300 et d'IRQ 10, tappez `nwpd 0x60 10 0x300`
3. Créez un nouveau répertoire et dezippez le package NCSA Telnet : `pkunzip tel2308b.zip`
4. Utilisez un éditeur de texte pour ouvrir le fichier `config.tel`
5. Modifiez `myip=192.168.0.x` ($1 < x < 255$), et netmask=255.255.255.0
6. Dans cet exemple, vous devriez mettre : `hardware=packet, interrupt=10, ioaddr=60`
7. Vous devriez avoir au moins une machine comme passerelle, i.e. la machine Linux :

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

8. Entrez aussi les spécifications de votre serveur DNS :

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

   NB : remplacez les information ci-dessus par les informations DNS qu'utilisent le serveur Linux

9. Sauvegarder votre fichier `config.tel`
10. Faites un telnet vers le serveur linux pour tester la connexion reseau : `telnet 192.168.0.1`. Si vous ne recevez pas de réponse, vérifiez votre configuration réseau.

## 4.6 Configuration d'une machine tournant sous MacOS et MacTCP

1. Si vous n'avez pas encore installé votre carte réseau ou ses drivers, faites le maintenant. L'explication de ces étapes sort du cadre de ce document.
2. Ouvrez le *tableau de bord MacTCP*. Sélection les bons drivers réseaux (Ethernet, PAS EtherTalk) et cliquez sur le bouton *'More...'*.
3. Dans *'Obtain Address:'*, cliquez sur *'Manually'*.
4. Dans *'Adresse IP:'*, sélectionnez *class C* du menu déroulant. Ignorez le reste de cette boite de dialogue.
5. Remplissez les informations nécessaires dans *'Adresses Serveurs de Noms :'*.
6. Dans *'Adresse Passerelle :'*, entrez 192.168.0.1
7. Cliquez sur *'OK'* pour sauvegarder vos réglages. Dans la fenêtre principale de *MacTCP*, entrez l'adresse IP de votre Mac (192.168.0.x, 1 < x < 255) dans le champs *'Adresse IP :'*.
8. Fermez le *Tableau de Bord MacTCP*. Si un dialogue vous demande de redémarrer, faites le.
9. Vous pouvez aussi faire un ping vers le serveur Linux pour tester la connexion de votre réseau. Si vous avez le freeware *MacTCP Watcher*, cliquez sur le bouton *'Ping'*, et entrez l'adresse de votre serveur linux (192.168.0.1) dans le dialogue qui apparaît. (Ceci est juste un test du LAN INTERNE, vous ne pouvez pas encore `ping`uer le monde exterieur.) Si vous ne recevez pas de réponses de vos PINGs, vérifiez votre configuration réseau.
10. Vous pouvez eventuellement créer un fichier `Hosts` dans le Dossier Système pour pouvoir utiliser des noms de machines sur votre LAN. Ce fichier existe probablement déjà dans votre Dossier Système et devrait contenir quelques exemples (en commentaire donc désactives) que vous pouvez modifier suivant vos besoins.

## 4.7 Configuration d'une machine tournant sous MacOS et Open Transport

1. Si vous n'avez pas encore installé votre carte réseau ou ses drivers, faites le maintenant. L'explication de ces étapes sort du cadre de ce document.
2. Ouvrez le *Tableau de Bord TCP/IP* et choisissez *'Mode Utilisateur...'* dans le menu *Edit*. Verifiez que le mode en cours est au moins sur *'Avancé'* et cliquez sur le bouton *'OK'*.
3. Choisissez ensuite *'Configurations...'* dans le menu *Fichier*. Sélectionnez la configuration *'Par Defaut'* et cliquer sur *'Dupliquer...'*. Entrez 'IP Masq' (ou quelquechose que vous reconnaitrez comme une configuration spéciale) dans le dialogue de *'Dupliquer la Configuration'*, qui vous dit sans doute quelquechose du style *'Par Defaut copie'*. Cliquez ensuite sur le bouton *'OK'*, et finalement sur le bouton *'Selectionner'*
4. Selectionnez *'Ethernet'* dans le menu deroulant *'Connexion :'*.
5. Selectonnez l'article approprié dans le menu déroulant *'Configuration:'*. Si vous ne savez pas ce qu'il faut choisir, c'est que vous devez probablement choisir le même article celui de votre configuration *'Par Defaut'* et quittez. Moi j'utilise *'Manuellement'*.

6. Entrez l'adresse IP de votre Mac (192.168.0.x, 1 < x < 255) dans le champ *'Adresse IP :'*.
7. Entrez 255.255.255.0 dans le champ *'Masque sous-reseau :'*.
8. Entrez 192.168.0.1 dans le champ *'Adresse du Routeur :'*.
9. Entrez les Adresses IP de vos serveurs DNS dans le champ *'Adr. Serv. de Noms :'*.
10. Entrez votre nom domaine de recherche Internet (ex : 'microsoft.com') dans le champ *'Domaine de Départ'* en dessous de *'Domaine de recherche implicite :'*.
11. Les procédures qui suivent sont optionnels. Des valeurs incorrects peuvent causer un comportement erratique. Si vous n'êtes pas sûr de ce que vous faites, c'est sans doute plus sûr de les laisser vides, non-cochés et/ou non sélectionnés. Retirez les informations de ces champs si nécessaire. A ma connaissance, il n'est pas possible, au travers des dialogues de TCP/IP, de dire au système de ne pas utiliser un fichier "Hosts" précédement sélectionné. Si vous savez comment faire, faites moi en part. Cochez la case *'802.3'* si votre réseau a besoin des fenêtres de type 802.3.
12. Cliquez sur *'Options...'*et vérifiez que TCP/IP est actif. J'utilise l'option *'Charger uniquement au besoin'*. Si vous lancez et quittez souvent des applications TCP/IP sans redémarrer, vous trouverez sans doute, que décocher l'option *'Charger uniquement au besoin'*va empêcher/réduire les effets de la gestion de la mémoire sur vos machines. Quand cette option est décochée, les piles du protocole TCP/IP sont toujours chargées et disponibles. Quand cette option est cochée, les piles TCP/IP sont chargées automatiquement quand il le faut et sont ensuite déchargées (unloaded). C'est ce chargement/déchargement qui provoque la fragmentation de la mémoire de vos machines.
13. Vous pouvez aussi faire un ping vers le serveur Linux pour tester la connexion de votre réseau. Si vous avez le freeware *MacTCP Watcher*, cliquez sur le bouton *'Ping'*, et entrez l'adresse de votre serveur linux (192.168.0.1) dans le dialogue qui apparaît. (Ceci est juste un test du LAN INTERNE, vous ne pouvez pas encore `ping`uer le monde extérieur.) Si vous ne recevez pas de réponses de vos PINGs, vérifiez votre configuration réseau.
14. Vous pouvez eventuellement créer un fichier `Hosts` dans le Dossier Système pour pouvoir utiliser des noms de machines sur votre LAN. Ce fichier existe probablement déjà dans votre Dossier Système et devrait contenir quelques exemples (en commentaire donc désactivés) que vous pouvez modifier suivant vos besoins. Sinon, vous pouvez récupérer une copie d'une machine qui tourne sous MacTCP ou simplement créer le votre (il suit un sous format des fichiers UNIX `/etc/hosts`, décrit dans la RFC952). Une fois ce fichier créé, ouvrez le *Tableau de Bord TCP/IP*, cliquez sur bouton *'Choisir un fichier "Hosts"...'*, et choisissez le fichier `Hosts`.
15. Cliquez sur la case de fermeture ou choisissez *'Fermer'* ou *'Quitter'* dans le menu *Fichier*, et cliquez ensuite sur *'Sauvegarder'* pour enregistrer vos changements.
16. Les changements prennent effet immediatement mais un petit redémerrage ne ferait pas de mal non plus.

## 4.8 Configuration du réseau Novell sous DNS

1. Si vous n'avez pas encore installé votre carte réseau ou ses drivers, faites le maintenant. L'explication de ces étapes sort du cadre de ce document.
2. Téléchargez tcpip16.exe ici : *The Novell LanWorkPlace page* `ftp.novell.com/pub/updates/unixconn/lwp5`
3. `edit 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. `edit c:\nwclient\net.cfg`

   : (changer le lien du driver vers le votre i.e. NE2000)

```
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

Changez votre adresse IP dans le champs "ip_address" si dessus (192.168.0.x, 1 < x < 255)
et creez enfin le fichier c:\bin\resolv.cfg:

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

5. Maintenant modifiez les entrées "NAMESERVER" et remplacez less avec les adresses IP correctes pour votre serveur DNS local.

6. Faites un `ping` : `ping 192.168.0.1` pour tester votre connexion avec la passerelle.
(Ceci est juste un test du LAN INTERNE, vous ne pouvez pas encore `ping`uer le monde extérieur.) Si vous ne recevez pas de réponses de vos PINGs, vérifiez votre configuration réseau.

## 4.9 Configuration d'OS/2 Warp

NDT : je ne connais pas OS/2 et ne sais pas s'il existe une version francaise de ce système.

1. Si vous n'avez pas encore installé votre carte réseau et recompilé votre noyau avec les drivers correspondants, faites le maintenant. L'explication de ces étapes sort du cadre de ce document.
2. Installez le protocole TCP/IP si vous ne l'avez pas déjà fait.
3. Allez dans *Programs/TCP/IP (LAN) / TCP/IP* Settings
4. Dans le champ *'Network'* ajoutez votre Adresse TCP/IP (192.168.0.x) et réglez votre masque de sous-réseau (255.255.255.0)
5. Sous *'Routing'* cliquez sur *'Add'*. Comme *Type* mettez *'default'* et tappez l'Adresse IP de votre serveur Linux dans le champ *'Router Address'*. (192.168.0.1).
6. Mettez les adresses des serveurs DNS (Serveurs de Noms) qu'utilisent votre serveur Linux dans *'Hosts'*.
7. Fermez le tableau de bord TCP/IP. Répondez par oui aux questions qui suivent.
8. Redémarrez votre système
9. Vous pouvez faire un ping vers votre serveur Linux pour tester votre configuration réseau. Tappez `'ping 192.168.0.1'` dans la fenetre de prompt d'OS/2. Si vous recevez les packets ping, c'est que tout ce passe bien.

## 4.10 Configuration d'OS/400 sur un IBM AS/400

La configuration de TCP/IP sur OS/400 version V4R1M0 sur un AS/400 dépasse le cadre de ce document.

1) Pour pouvoir configurer toute tâche de communication sur votre AS/400, vous devez avoir le privilège spécial *IOSYSCFG (I/O System Configuration) dans votre profile utilisateur. Vous pouvez verifier les caractéristiques de votre profil utilisateur avec la commande DSPUSRPRF

2) Tappez la commande GO CFGTCP pour accéder au menu de configuration de TCP/IP.

3) Selectionnez Option 2 - Work with TCP/IP Routes.

4) Entrer un 1 dans le champs Opt pour ajouter une route * dans Route Destination (Route de Destination) tappez *DFTROUTE * dans Subnet Mask (Masque de sous reseau) tappez *NONE * dans Type of Service (Type de Service) tappez *NORMAL * dans Next Hop (Prochain saut) tappez l'adresse IP de votre passerelle (le serveur linux)

## 4.11 Configuration des autres Systèmes

La même logique devrait s'appliquer pour les réglages sur les autres plateformes. Consultez les sections précédentes. Si vous êtes intéréssés par la rédaction des méthodes pour les systèmes qui n'ont pas encore été traités, vous pouvez envoyer par email les instructions détaillés à *ambrose@writeme.com* et *dranch@trinnet.net* .

# 5 Tester IP Masquerade

Enfin, il est temps de faire un essai officiel de l'IP Masquerading après ce dur labeur. Si vous n'avez pas encore redémarré votre serveur Linux, faites le pour être sur que la machine démarre bien, exécute les scripts /etc/rc.d/rc.firewall etc. Ensuite, vérifiez que les connexions internes de votre LAN et les connexions de votre serveur Linux avec Internet fonctionnent bien.

Faites ces -10- tests pour être sur que les différents aspects de votre configuration MASQ fonctionnent correctement :

## 5.1 Tester les connexions locales

- **Première Etape : Tester les connexions locales entre les PC** A partir d'un ordinateur MASQué interne, essayez de pinguer sa propre adresse IP locale (i.e. *ping 192.168.0.10* ). Ce test va vérifier que TCP/IP fonctionne bien sur la machine locale. Presque tous les systèmes d'exploitation modernes ont une commande ping integrée. Si ce ping ne fonctionne pas, vérifiez que vous avez correctement configuré le PC MASQué comme décrit plus tôt dans la section 3.3.2 () de ce HOWTO. L'output devrait ressembler à ça (faire Control-C pour arrêter le ping) :

```
masq-client# ping 192.168.0.10
PING 192.168.0.10 (192.168.0.10): 56 data bytes
64 bytes from 192.168.0.10: icmp_seq=0 ttl=255 time=0.8 ms
64 bytes from 192.168.0.10: icmp_seq=1 ttl=255 time=0.4 ms
64 bytes from 192.168.0.10: icmp_seq=2 ttl=255 time=0.4 ms
64 bytes from 192.168.0.10: icmp_seq=3 ttl=255 time=0.5 ms

--- 192.168.0.10 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 0.4/0.5/0.8 ms
```

## 5.2 Tester les connexions internes du serveur Linux

- **Deuxième Etape : Tester les connexions internes du serveur Linux** Sur le serveur MASQ lui-même, pinguez l'adresse IP de l'interface réseau du serveur MASQ (i.e. *ping 192.168.0.1*). L'outpout devrait ressembler à ça (faire Control-C pour arreter le ping) :

```
masq-client# ping 192.168.0.1
PING 192.168.0.1 (192.168.0.1): 56 data bytes
64 bytes from 192.168.0.1: icmp_seq=0 ttl=255 time=0.8 ms
64 bytes from 192.168.0.1: icmp_seq=1 ttl=255 time=0.4 ms
64 bytes from 192.168.0.1: icmp_seq=2 ttl=255 time=0.4 ms
64 bytes from 192.168.0.1: icmp_seq=3 ttl=255 time=0.5 ms

--- 192.168.0.1 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 0.4/0.5/0.8 ms
```

## 5.3 Tester la Connection Externe du serveur Linux

- **Troisième Etape : Tester la Connection Externe du serveur Linux**

  Ensuite pinguez l'adresse IP de l'interface réseau externe (carte réseau connectée à Internet). Cette adresse peut être reçue par PPP, Ethernet, etc. C'est la connexion vers votre FAI. Si vous ne connaissez pas cette adresse IP, executez la commande Linux *"/sbin/ifconfig"* sur le serveur MASQ Linux. L'output devrait ressembler à ça (nous cherchons l'adresse IP de l'interface eth0) :

```
eth0      Link encap:Ethernet  HWaddr 00:08:C7:A4:CC:5B
          inet addr:12.13.14.15  Bcast:64.220.150.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:6108459 errors:0 dropped:0 overruns:0 frame:0
          TX packets:5422798 errors:8 dropped:0 overruns:0 carrier:8
          collisions:4675 txqueuelen:100
          Interrupt:11 Base address:0xfcf0
```

  Nous pouvons voir dans cet example que l'adresse IP externe est "12.13.14.15". Bon, maintenant que vous avez votre adresse IP après avoir lancé la commande "ifconfig", pinguez votre adresse IP externe. Nous aurons ainsi la confirmation que le serveur MASQ a bien toutes les connexions réseaux. L'outpout devrait ressembler à ça (faire Control-C pour arrêter le ping) :

```
masq-server# ping 12.13.14.15
PING 12.13.14.15 (12.13.14.15): 56 data bytes
64 bytes from 12.13.14.15: icmp_seq=0 ttl=255 time=0.8 ms
64 bytes from 12.13.14.15: icmp_seq=1 ttl=255 time=0.4 ms
64 bytes from 12.13.14.15: icmp_seq=2 ttl=255 time=0.4 ms
64 bytes from 12.13.14.15: icmp_seq=3 ttl=255 time=0.5 ms

--- 12.13.14.15 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 0.4/0.5/0.8 ms
```

  Si l'un de ces tests ne fonctionne pas, vous devez repartir en arrière et vérifier une seconde fois vos cablages, et vérifier que vos deux NICs du serveur MASQ sont vu dans "dmesg". Un exemple de l'output se trouve vers la fin (END) de la commande "dmesg" :

```
.
.
PPP: version 2.3.7 (demand dialling)
TCP compression code copyright 1989 Regents of the University of California
PPP line discipline registered.
3c59x.c:v0.99H 11/17/98 Donald Becker
http://cesdis.gsfc.nasa.gov/linux/drivers/
vortex.html
eth0: 3Com 3c905 Boomerang 100baseTx at 0xfe80,  00:60:08:a7:4e:0e, IRQ 9
  8K word-wide RAM 3:5 Rx:Tx split, autoselect/MII interface.
  MII transceiver found at address 24, status 786f.
  Enabling bus-master transmits and whole-frame receives.
eth1: 3Com 3c905 Boomerang 100baseTx at 0xfd80,  00:60:97:92:69:f8, IRQ 9
  8K word-wide RAM 3:5 Rx:Tx split, autoselect/MII interface.
```

```
    MII transceiver found at address 24, status 7849.
    Enabling bus-master transmits and whole-frame receives.
  Partition check:
   sda: sda1 sda2 < sda5 sda6 sda7 sda8 >
   sdb:
  .
  .
```

N'oubliez pas non plus de vérifier les configurations NIC de votre distrib Linux etc. suivant les recommandations qui se trouvent au début de ce HOWTO.

## 5.4 Tester les connexions locales des PC vers le serveur Linux

- **Quatrième Etape : Tester les connexions locales des PC vers le serveur Linux** Sur un ordinateur interne MASQué, essayez de pinguer l'adresse IP de la carte Ethernet interne du serveur de Masquerading, (i.e. *ping 192.168.0.1*). On va ainsi vérifier que le réseau interne et le routage sont corrects. L'outpout devrait ressembler à ça (faire Control-C pour arrêter le ping) :

```
masq-client# ping 192.168.0.1
PING 192.168.0.1 (192.168.0.1): 56 data bytes
64 bytes from 192.168.0.1: icmp_seq=0 ttl=255 time=0.8 ms
64 bytes from 192.168.0.1: icmp_seq=1 ttl=255 time=0.4 ms
64 bytes from 192.168.0.1: icmp_seq=2 ttl=255 time=0.4 ms
64 bytes from 192.168.0.1: icmp_seq=3 ttl=255 time=0.5 ms

--- 192.168.0.1 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 0.4/0.5/0.8 ms
```

Si le test échoue, vérifiez les connexions entre les cartes Ethernet sur le serveur MASQ et les ordinateurs MASQués. En général il y a une LED derrière chaque carte Ethernet et il y en aussi sur les hub/switch Ethernet (si vous en utiliser un). Si c'est ça le problème, vérifiez que l'ordinateur MASQué interne est configuré correctement comme indiqué dans la section 3.3.2 (). Si le client MASQué est correctement configuré, vérifiez une seconde fois votre cablage réseau et vérifiez que les LED s'allument bien de chaque côté des cables (carte Ethernet des clients et cartes Ethernet INTERNES du serveur Linux).

## 5.5 Tester le forwarding des paquets internes MASQ ICMP

- **Cinquième Etape : Tester le forwarding des paquets internes MASQ ICMP** A partir d'un ordinateur MASQué interne, pinguez l'adresse IP externe du serveur MASQ optenue à l'étape TROIS ci-dessus. Cette adresse est votre adresse PPP, Ethernet, etc. fournie par votre FAI. Ce ping va prouver que le masquerading fonctionne (spécifiquement ICMP Masquerading).

  Si ca ne marche pas, vérifiez d'abord que la "Passerelle par Defaut" du PC MASQué pointe vers l'adresse IP du serveur MASQ interne. Vérifiez aussi que le script /etc/rc.d/rc.firewall a fonctionné sans erreurs. Juste pour faire un test, lancer de nouveau le script /etc/rc.d/rc.firewall pour voir si tout se passe bien. Vérifiez aussi, puisque la plupart des noyaux le gérent par defaut, que vous avez activé l'"ICMP Masquerading" dans la configuration noyau et que vous avez aussi activé l'"IP Forwarding" dans votre script /etc/rc.d/rc.firewall.

  Si vous ne pouvez toujours pas faire fonctionner tout ça, jetez un coup d'oeil à l'output de cette commande sur votre serveur Linux MASQ :

- "*ifconfig*" : Vérifiez que l'interface de votre connexion Internet (ppp0, eth0, etc.) est UP (en route) et que vous avez la bonne adresse IP pour votre connexion à Internet. Un exemple d'output est donne a l'ETAPE TROIS ci-dessus.
- "*netstat -rn*" : Vérifiez que la passerelle par defaut (le .1 avec les adresses IP dans la colonne Gateway) est active. Par exemple, l'output devrait ressembler à ca :

```
masq-server# netstat -rn
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
192.168.0.1     0.0.0.0         255.255.255.255 UH        0 16384      0 eth1
12.13.14.15     0.0.0.0         255.255.255.255 UH        0 16384      0 eth0
12.13.14.0      0.0.0.0         255.255.255.0   U         0 0          0 eth0
192.168.0.0     0.0.0.0         255.255.255.0   U         0 0          0 eth1
127.0.0.0       0.0.0.0         255.0.0.0       U         0 16384      0 lo
0.0.0.0         12.13.14.1      0.0.0.0         UG        0 16384      0 eth0
```

  Vous avez remarqué que la DERNIERE ligne commençait par 0.0.0.0? Vous avez remarqué qu'il y a une adresse IP dans le champ "Gateway" ? Vous devriez mettre une adresse IP pour votre propre configuration dans ce champ.
- "*cat /proc/sys/net/ipv4/ip_forward*" : Vérifiez que vous avez un "1" qui montre que le forwarding sous Linux fonctionne
- Lancez la commande "*/sbin/ipchains -n -L*" pour les utilisateurs du 2.2.x ou "*/sbin/ipfwadm -F -l*" pour les utilisateurs du 2.0.x. Regardez aussi la section sur le FORWARDing pour vérifier que vous avez activé MASQ. Voici un exemple de l'output d'IPCHAINS pour les utilisateurs des règles naives rc.firewall :

```
.
.
Chain forward (policy REJECT):
target     prot opt     source               destination          ports
MASQ       all  ------  192.168.0.0/24       0.0.0.0/0            n/a
ACCEPT     all  ----l-  0.0.0.0/0            0.0.0.0/0            n/a
.
.
```

## 5.6 Tester le forwarding de paquets MASQ ICMP externes

- **Sixième Etape : Tester le forwarding de paquets MASQ ICMP externes** Essayer de pinguer à partir d'un ordinateur MASQué interne une IP static d'Internet (i.e. *ping 152.19.254.81* (c'est le - http://metalab.unc.edu - home of MetaLabs' Linux Archive). Si ca ne fonctionne pas, vérifiez de nouveau votre connexion à Internet. Si ca ne marche toujours pas, vérifier que vous utilisez bien les règles naives rc.firewall et que vous avez bien compilé l'ICMP Masquerading dans votre noyau linux. Vérifiez aussi que vos règles qui activent IP MASQ pointent vers la bonne interface EXTERNE.

## 5.7 Tester le fonctionnement de MASQ sans DNS

- **Septième Etape : Tester le fonctionnement de MASQ sans DNS** Maintenant essayez de vous connecter par Telnet à une adresse IP distante (i.e. *telnet 152.2.254.81* (metalab.unc.edu - NB : ca peut prendre du temps avant que vous ne voyiez apparaître le prompt de login parce que c'est un serveur

TRES chargé). Avez vous reçu le prompt du login après un certain laps de temps ? Si ça a marché, c'est que le TCP Masquerading fonctionne bien. Si ça n'a pas marché, essayez de faire un Telnet vers un autre serveur qui pourrait accepter les TELNET, comme 198.182.196.55 (www.linux.org). Si ca ne fonctionne toujours pas, vérifiez que vous utilisez bien les règles naives rc.firewall. Un exemple de ce que vous devriez voir est donne ici (pressez Control-D pour sortir de TELNET) :

```
masq-client# telnet 152.2.254.81
Trying 152.2.254.81...
Connected to 152.2.254.81.
Escape character is '^]'.


SunOS 5.7


******************** Welcome to MetaLab.unc.edu *******************

 To login to MetaLab as a user, connect to login.metalab.unc.edu.
           This machine allows no public telnet logins.

login: Connection closed by foreign host.
```

## 5.8 Tester le fonctionnement de MASQ avec DNS

- **Huitième Etape : Tester le fonctionnement de MASQ avec DNS** Maintenant essayez de faire un TELNET vers un HOSTNAME (i.e. *"telnet metalab.unc.edu"* (152.2.254.81). Si ça marche, c'est que le DNS fonctionne bien aussi. Si ça ne marche pas mais que l'etape SIX fonctionnait, vérifiez que vous avez entré des DNS valides dans votre ordinateur MASQué comme le montre la section 3.3.2 ().

## 5.9 Tester plus de fonctionnalités de MASQ avec DNS

- **Neuvième Etape : Tester plus de fonctionnalités de MASQ avec DNS** Comme dernier test, essayez de surfer sur quelques sites *'INTERNET'* WWW à partir des vos machines **MASQuées**, et regardez si ca fonctionne. Par exemple, accedez au site *Linux Documentation Project* . Si ça marche, vous pouvez être presque sûr que tout fonctionne BIEN ! Si certains sites ne fonctionnent pas bien la où les autres semblent fonctionner correctement, regardez les étapes suivantes pour essayer de trouvez les causes.

  Si vous voyez la page d'accueil de la The Linux Documentation Project, alors, **FELICITATIONS ! Ca marche !** Si ce site Web s'affiche correctement, alors tous les autres protocoles réseaux standards tels que PING, TELNET, SSH, et avec leurs modules IP MASQ respectifs chargés : FTP, Real Audio, IRC DCCs, Quake I/II/III, CuSeeme, VDOLive, etc. devraient aussi fonctionner correctement ! Si FTP, IRC, RealAudio, Quake I/II/III, etc. ne fonctionnent pas correctement, ou de manière peu performante, vérifiez que leurs modules Masquerading associés sont chargés en lancant la commande "lsmd" et vérifiez aussi que vous chargez les modules quand les ports ne sont pas les ports par défaut. Si vous ne voyez pas les modules dont vous avez besoin, vérifiez que le script /etc/rc.d/rc.firewall les charge bien (i.e. enlevez les caractères # pour un module IP MASQ donné).

### 5.10 S'il reste des problèmes de fonctionnement, performances etc.

- **Dixième Etape : S'il reste des problèmes de fonctionnement, performances etc.** Si votre système a passé avec succès tous les tests mais que quelques trucs genre le surf sur les sites WWW, le FTP, et d'autres types de traffic ne sont pas fiables, je vous recommande de lire l'entrée 7.14 () de la FAQ dans la Section 7. Il peut aussi y avoir d'autres éléments de la FAQ qui puissent vous aider autant que les nombreux utilisateurs qu'ils ont aidés par le passé.

# 6 Autres problèmes relatifs à IP Masquerade et à la compatibilité logicielle

## 6.1 Problèmes avec IP Masquerade

Certaines applications des protocoles TCP/IP ne fonctionnent pas actuellement avec l'IP Masquerading sous Linux parce que soit ils supposent des choses sur les numéros de port soit ils encodent sur les adresses TCP/IP et/ou les numéros de port dans leur flux de données. Ces derniers protocoles ont besoin de proxies ou de modules IP MASQ spécifiques pour fonctionner correctement dans le code de masquerading.

## 6.2 Services entrant

Par défaut, Linux IP Masquerading ne peut pas du tout prendre en charge les services entrant mais il y a quelques façons de les lui faire accepter.

Si vous n'avez pas besoin de beaucoup de securité, vous pouvez simplement forwarder ou rédiriger les ports IP. Il y a plusieurs façon de faire ça mais la plus stable est d'utiliser IPPORTFW. Pour plus d'informations, reportez vous à la section 6.7 ().

Si vous désirez avoir un certain niveau d'autorisation sur les connexions entrantes, vous aurez besoin de configurer soit des TCP-wrappers, soit Xinetd pour permettre la connexion d'adresses IP specifiques. Un bon endroit où trouver des utilitaires et de la documentation est le TIS Firewall Toolkit.

De plus amples détails sur la securité entrante peuvent être trouvés dans le document *TrinityOS* `http://www.ecst.csuchico.edu/~dranch/LINUX/index-linux.html#TrinityOS` et a l' *IP Masquerade Resource* `http://ipmasq.cjb.net` .

## 6.3 Compatibilité Logicielle et autres notes sur la configuration

> **** La *Linux Masquerade Application list* `http://www.tsmservices.com/masq` a une tonne d'informations au sujet des applications qui fonctionnent à travers l'IP Masquerading sous Linux. Ce site a récemment été pris en charge par Steve Grevemeyer qui l'a doté d'une base de données complète. C'est une source exceptionnelle !**

En général, toute application qui utilise le TCP et l'UDP standards devrait fonctionner. Si vous avez des suggestions, des indications etc. reportez vous à l' *IP Masquerade Resource* `http://ipmasq.cjb.net/` pour de plus amples détails.

### 6.3.1 Clients Réseaux qui -Fonctionnent- avec IP Masquerade

Clients Généraux:

**Archie**

toutes les plateformes compatibles, clients pour la recherche de fichiers (tous les clients archie ne sont pas compatibles)

**FTP**

toutes les plateformes compatibles, avec le module noyau *ip_masq_ftp.o* pour les connexions FTP actives.

**Gopher client**

toutes les plateformes compatibles

**HTTP**

toutes les plateformes compatibles, WWW surfing

**IRC**

toutes les plateformes compatibles, DCC est compatible via le module *ip_masq_irc.o*

**NNTP (USENET)**

toutes les plateformes compatibles, USENET news client

**PING**

toutes les plateformes compatibles, avec le module noyau ICMP Masquerading

**POP3**

toutes les plateformes compatibles, clients email

**SSH**

toutes les plateformes compatibles, TELNET/FTP clients sécurisés

**SMTP**

toutes les plateformes compatibles, serveurs d'email tels que Sendmail, Qmail, PostFix, etc.

**TELNET**

toutes les plateformes compatibles, session distante

**TRACEROUTE**

versions sous UNIX et Windows, quelques variantes pourraient ne pas fonctionner (NDT : fonctionne aussi sous MacOS)

**VRML**

Windows(peut-être toutes les plateformes compatibles), virtual reality surfing

**client WAIS**

toutes les plateformes compatibles

Clients Multimedia et Communication:

**Toutes les applications H.323**

- MS Netmeeting, Intel Internet Phone Beta , et autres applications H.323 - Il y a maintenant deux façons de faire fonctionner ces clients aux travers de connexions MASQuées :

Il y a un module BETA stable disponible sur le *MASQ WWW site* ou sur *http://www.coritel.it/projects/sofia/nat.html* `http://www.coritel.it/projects/sofia/nat.html` pour utiliser Microsoft Netmeeting v3.x sous les noyaux 2.2.x. Il y aussi un autre module sur le site Web de MASQ spécifique à Netmeeting 2.x pour les noyaux 2.0.x mais il n'est pas compatible avec Netmeeting v3.x.

Une autre solution, commerciale, et la passerelle *Equivalence's PhonePatch* `http://www.equival.com.au/phonepatch/index.html` H.323.

**Alpha Worlds**

Windows, Client-Serveur 3D programme de tchatche

**CU-SeeMe**

toutes les plateformes compatibles, avec le module *ip_masq_cuseeme* chargé, reportez vous SVP à la section 6.8.2 () pour de plus amples détails.

**ICQ**

Tous les clients sont compatibles. Requiert un noyau compilé avec la compatibilité IPPORTFW et ICQ configuré comme étant derrière un proxy NON-SOCKS. Une description complète de cette configuration est disponible à la section 6.9 ().

**Internet Phone 3.2**

Windows, communications audio Peer-to-peer, vous pouvez entrer en communication avec quelqu'un seulement si vous êtes l'appelant, vous ne pouvez être appelé sans un port forwarding spécifique. Reportez vous à la section 6.7 () pour de plus amples détails.

**Internet Wave Player**

Windows, streaming audio par Internet

**Powwow**

Windows, communications textuelles peer-to-peer, vous pouvez entrer en communication avec quelqu'un seulement si vous êtes l'appelant, vous ne pouvez être appelé sans un port forwarding spécifique. Reportez vous à la section 6.7 () pour de plus amples détails.

**Real Audio Player**

Windows, streaming audio par Internet, vous obtiendrez de meilleurs résultats avec le module UDP *ip_masq_raudio*

**True Speech Player 1.1b**

Windows, streaming audio par Internet

**VDOLive**

Windows, avec le patch *ip_masq_vdolive*

**Worlds Chat 0.9a**

Windows, Client-Serveur 3D programme de tchatche

Jeux - Reportez vous à la section 6.10 () pour de plus amples détails sur le patch LooseUDP

**Battle.net**

Fonctionne mais requiert les ports TCP 116 et 118 et les ports UDP 6112 IPPORTFWés vers la machine de jeu. reportez vous à la section 6.7 () pour de plus amples détails. Veuillez notez que les serveurs FSGS et Bnetd requièrent toujours IPPORTFW puisqu'ils n'ont pas été réécrits pour être compatibles NAT.

**BattleZone 1.4**

Fonctionne avec le patch LooseUDP et les nouveaux *.DLLs Activision* compatibles NAT.

**Dark Reign 1.4**

Fonctionne avec le patch LooseUDP ou requiert les ports TCP 116 et 118 et les ports UDP 6112 IPPORTFWés vers la machine de jeu. Reportez vous à la section 6.7 () pour de plus amples détails.

**Diablo**

Fonctionne avec le patch LooseUDP ou requiert les ports TCP 116 et 118 et les ports UDP 6112 IPPORTFWés vers la machine de jeu. Les nouvelles versions de Diablo n'utilisent que le port TCP 6112 et le port UDP 6112. Reportez vous à la section 6.7 () pour de plus amples détails.

**Heavy Gear 2**

Fonctionne avec le patch LooseUDP ou requiert les ports TCP 116 et 118 et les ports UDP 6112 IPPORTFWés vers la machine de jeu. Reportez vous à la section 6.7 () pour de plus amples détails.

**Quake I/II/III**

Fonctionne directement mais requiert le module *ip_masq_quake* s'il y a plus d'un joueur de Quake I/II/III derrière le serveur MASQ. Ce module n'est compatible qu'avec Quake I et QuakeWorld par défaut. Si vous voulez utiliser Quake II ou des ports non standard pour le server, reportez vous à la section d'installation des modules dans les jeux de règles 3.3.1 () et3.3 ().

**StarCraft**

Fonctionne avec le patch LooseUDP ou requiert les ports TCP et UDP 6112 IPPORTFWés vers la machine de jeu. Reportez vous à la section 6.7 () pour de plus amples détails.

**WorldCraft**

Fonctionne avec le patch LooseUDP

Autres Clients:

**package Linux net-acct**

Linux, package d'administration des comptes à distance

**NCSA Telnet 2.3.08**

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

**PC-anywhere pour Windows**

MS-Windows, Controle à distance un PC par TCP/IP, fonctionne uniquement si c'est un client. Ne fonctionne pas sans un port forwarding spécifique si c'est le serveur. Reportez vous à la section 6.7 () pour de plus amples details.

**Socket Watch**

utilise NTP - protocole d'horloge réseau

### 6.3.2 Clients qui ne sont pas entièrement compatibles avec IP MASQ :

**Intel Streaming Media Viewer Beta 1**

Impossible de se connecter au serveur

**Netscape CoolTalk**

Impossible de se connecter a l'hôte distant

**WebPhone**

Ne fonctionne pas (Il fait des suppositions erronnées au sujet des adresses).

## 6.4 Jeux de règles de d'IP Firewall (IPFWADM) plus résistants (Stronger)

Cette section fournit un guide plus detaillé d'utilisation de l'outil de firewall pour 2.0.x, IPFWADM. Reportez vous si dessous pour les règles d'IPCHAINS.

Voici un exemple d'un système de firewall/masquerade derrière une connexion PPP avec une adresse PPP statique (les instructions pour les connexions PPP dynamiques sont incluses mais désactivées). L'interface de confiance est 192.168.0.1 et l'adresse IP de l'interface PPP a été changée pour protéger le coupable :). J'ai listé chaque interface entrante et sortante pour détecter aussi bien les IP spoofings que les faux routage et/ou masquerading. Tout ce qui n'est pas explicitement permis est **INTERDIT** (euh... rejeté en fait). Si votre serveur IP MASQ ne fonctionne plus après avoir implémenté ce script rc.firewall, vérifiez que vous l'avez modifié pour votre propre configuration et contrôlez votre fichier SYSLOG /var/log/messages ou /var/adm/messages pour trouver d'éventuels erreurs de firewall.

Pour des exemples plus exhaustifs de règles Ôstrong' d'IPFWADM IP Masqueradé pour PPP, modem pour cable, etc., vous pouvez vous référer à *TrinityOS - Section 10* `http://www.ecst.csuchico.edu/~dranch/LINUX/index-linux.html#TrinityOS` et *GreatCircle's Firewall WWW page*

**NB:** Si vous avez une adresse TCP/IP assignée de facon dynamique par votre FAI (PPP, aDSL, Cable, etc.) vous **NE POUVEZ PAS CHARGER** ces règles Ôstrong' au moment du boot. Vous aurez soit à relancer le jeu de règles de ce firewall à CHAQUE FOIS que vous avez une nouvelle adresse IP soit faire un /etc/rc.d/rc.firewall plus intelligent. Pour faire ceci pour les utilisateurs de PPP, lisez attentitevement et enlever les marques de commentaires les lignes correspondantes dans la section Òrecuperation d'IP PPP dynamiqueÓ ci-dessous. Vous pouvez aussi trouver de plus amples détails dans le documentation *TrinityOS - Section 10* `http://www.ecst.csuchico.edu/~dranch/LINUX/index-linux.html#TrinityOS` sur les jeux de règles ÔStrong' et les adresses IP dynamiques.

**Veuillez aussi noter qu'il existe plusieurs utilitaires de création de Firewall qui possèdent des interfaces graphiques. Vous pouvez vous reporter à la section 7 () pour des détails complets.**

Enfin, si vous utilisez une adresse IP STATIQUE obtenue par PPP, changez la ligne "ppp_ip="votre.adresse.PPP.statique" par votre adresse.

——————————————————————————-

```
#!/bin/sh
#
# /etc/rc.d/rc.firewall: Un exemple de jeu de regles d'un firewall IPFWADM semi-STRONG IPFWADM
#

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

# on teste, attendre un peu puis effacer toutes les regles de firewall
# enlever les marques de commentaire des lignes qui suivent si vous voulez que
# le firewall se desactive automatiquement au bout de 10 mins.
# (sleep 600; \
# ipfwadm -I -f; \
# ipfwadm -I -p accept; \
```

```
# ipfwadm -O -f; \
# ipfwadm -O -p accept; \
# ipfwadm -F -f; \
# ipfwadm -F -p accept; \
# ) &

# Charge les modules necessaires a IP MASQ
#
#   NB: Charger uniquement les modules IP MASQ dont vous avez besoin. Tous les modules
#   IP MASQ actuels sont montres ci-dessous mais sont commentes pour les empecher de se charger.

# Necessaire pour le chargement initial des modules
#
/sbin/depmod -a

# Permet le masquerading correct des transfert de fichier par FTP avec la methode PORT
#
/sbin/modprobe ip_masq_ftp

# Permet le masquerading de RealAudio par UDP. Sans ce module,
#       RealAudio FONCTIONNERA mais en mode TCP. Ce qui peu causer une baisse
#       dans la qualite du son
#
#/sbin/modprobe ip_masq_raudio

# Permet le masquerading des transferts de fichier par DCC pour les IRC
#
#/sbin/modprobe ip_masq_irc


# Permet le masquerading de Quake et QuakeWorld par defaut. Ce module est
#   necessaire pour les utilisateurs multiples dirriere un server Linux MASQ. Si vous voulez
#   jouer a Quake I, II, et III, utilisez le second exemple.
#
#   NB:  si vous rencontrez des ERREURs lors de chargement du module QUAKE, c'est que vous utilisez
#   un ancien noyau buggue. Mettez a jour votre noyau pour supprimer l'erreur.
#
#Quake I / QuakeWorld (ports 26000 and 27000)
#/sbin/modprobe ip_masq_quake
#
#Quake I/II/III / QuakeWorld (ports 26000, 27000, 27910, 27960)
#/sbin/modprobe ip_masq_quake 26000,27000,27910,27960


# Permet le masquerading du logiciel CuSeeme pour la video conference
#
#/sbin/modprobe ip_masq_cuseeme

# Permet le masquerading du logiciel VDO-live pour la video conference
#
```

```
#/sbin/modprobe ip_masq_vdolive


#CRITIQUE:  Active l'IP forwarding puisqu'il est desactive par defaut
#
#           Utilisateurs Redhat: vous pourrez essayer en changeant les options dans
#                          /etc/sysconfig/network de:
#
#                        FORWARD_IPV4=false
#                              to
#                        FORWARD_IPV4=true
#
echo "1" > /proc/sys/net/ipv4/ip_forward


#CRITIQUE:  Active automatiquement l'IP defragmenting puisqu'il est desactive par defaut
#           dans les noyaux 2.2.x. Ceci etait une option de compilation mais ca a change
#           depuis le noyau 2.2.12
#
echo "1" > /proc/sys/net/ipv4/ip_always_defrag


# Utilisateurs d'IP Dynamiques:
#
#   Si vous recevez votre adresse IP de maniere dynamique a partir d'un server SLIP, PPP, ou
#   DHCP, activez option suivante qui active le hacking (au bon sens du terme NDT) des
#   adresses IP dynamique dans IP MASQ, rendant ainsi les choses plus faciles pour les
#   programmes du type Diald.
#
#echo "1" > /proc/sys/net/ipv4/ip_dynaddr


# Specifiez ici votre adresse IP statique.
#
# Si vous avez une adresse IP DYNAMIQUE, vous devez faire trouver a ce jeu
# de regles votre adresse IP a chaque fois que vous avez une nouvelle. Dans ce but,
# activez le script d'une ligne qui suit.

#
#   utilisateurs de DHCP :
#   ----------------------
#   Si vous recevez votre adresse TCP/IP, **vous devez** activer les commandes
#   #ees sous la section PPP ET remplacer le mot "ppp0" par le nom de votre
#   de votre connexion Internet EXTERNE (eth0, eth1, etc). Notez aussi que le server
#   DHCP peut changer votre adresse IP. Pour resoudre ce probleme, les utilisateurs
#   doivent configurer leur client DHCP de sorte qu'il relance le jeu de regles du firewall
#   chaque fois que leur bail DHCP est renouvele.
#
#     NB #1:  Quelques clients DHCP comme l'ancienne version de "pump" (les nouvelles
#               versions ont ete corrigees) n'avaient pas la capacite de relancer
```

```
#               les scripts apres une renouvellement de bail. Pour cette raison, vous
#               aurez besoin de le remplacer par quelquechose du style "dhcpcd" ou "dhclient".
#
#     NB #2:  La syntaxe de "dhcpcd" a change dans les versions recentes.
#
#               Les anciennes version avaient une syntaxe du type:
#                         dhcpcd -c /etc/rc.d/rc.firewall eth0
#
#               Les versions plus recentes ont une syntaxe du type:
#                         dhcpcd eth0 /etc/rc.d/rc.firewall
#
#     NB #3:  Pour les utilisateurs de Pump, ajouter cette ligne de commande dans votre
#     fichier /etc/pump.conf:
#
#                   script /etc/rc.d/rc.firewall
#
#   utilisateurs de PPP :
#   ---------------------
#   Si vous n'etes pas deja au courant, le script /etc/ppp/ip-up est toujours lance quand
#   une connexion PPP arrive. A cause de ca, on peut demander au jeu de regles d'aller recuperer
#   la nouvelles adresse IP PPP et de mettre a jour notre jeu de regles du strong firewall.
#
#   Si le fichier /etc/ppp/ip-up existe deja, vous devez le modifier et ajouter une ligne
#   contenant "/etc/rc.d/rc.firewall" pres de la fin du fichier.
#
#   Si vous n'avez pas encore de script /etc/ppp/ip-up, vous devez creer le lien suivant
#   pour lancer le script /etc/rc.d/rc.firewall.
#
#       ln -s /etc/rc.d/rc.firewall /etc/ppp/ip-up
#
#   * Vous devez ensuite activer les commandes #ees si dessous *
#
#
# Utilisateurs de PPP et DHCP :
# -----------------------------
# Enlevez le # de la ligne si dessous et placez un # sur la ligne suivante.
#
#ppp_ip="`/sbin/ifconfig ppp0 | grep 'inet addr' | awk '{print $2}' | sed -e 's/.*://'`"
#
ppp_ip="your.static.PPP.address"


# MASQ timeouts
#
#  timeout de 2 heures pour les sessions TCP
#  timeout de  10 sec pour le traffic apres que le paquet TCP/IP "FIN" est recu
#  timeout de 160 sec pour le traffic UDP (Important pour les utilisateur d'ICQ MASQues)
#
/sbin/ipfwadm -M -s 7200 10 60
```

```
#############################################################################
# Entree (incoming), flush et politique par defaut de rejet. En fait la politique par defaut
# est inapplicable parce qu'il y une regle qui attrape tout, refuse et logue.
#
/sbin/ipfwadm -I -f
/sbin/ipfwadm -I -p reject

# interface locale, machines locales, on peut allez n'importe ou
#
/sbin/ipfwadm -I -a accept -V 192.168.0.1 -S 192.168.0.0/24 -D 0.0.0.0/0

# interface distance, pretendant etre une machine locale, IP spoofing, tire toi
#
/sbin/ipfwadm -I -a reject -V $ppp_ip -S 192.168.0.0/24 -D 0.0.0.0/0 -o

# interface distante, toute source, peut aller a l'adresse PPP permanante
#
/sbin/ipfwadm -I -a accept -V $ppp_ip -S 0.0.0.0/0 -D $ppp_ip/32

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

# regle qui attrape tout, refuse tout autre entree et le logue. Dommage qu'il n'y ait pas
# d'options pour le log sur cette politique mais ceci va faire le travail :
#
/sbin/ipfwadm -I -a reject -S 0.0.0.0/0 -D 0.0.0.0/0 -o


#############################################################################
# Sortie (outgoing), flush et politique par defaut de rejet. En fait la politique par defaut
# est inapplicable parce qu'il y une regle qui attrape tout, refuse et logue.
#
/sbin/ipfwadm -O -f
/sbin/ipfwadm -O -p reject

# interface locale, toute source allant vers le reseau 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

# sortie vers le reseau local d'une interface distance, routage bizarre, rejet
#
/sbin/ipfwadm -O -a reject -V $ppp_ip -S 0.0.0.0/0 -D 192.168.0.0/24 -o

# sortie du reseau local vers une interface distante, masquerading modifie, rejet
#
/sbin/ipfwadm -O -a reject -V $ppp_ip -S 192.168.0.0/24 -D 0.0.0.0/0 -o

# sortie du reseau local d'une interface distante, rejet
```

```
#
/sbin/ipfwadm -O -a reject -V $ppp_ip -S 0.0.0.0/0 -D 192.168.0.0/24 -o

# tout autre chose qui sort de l'interface distance est valide
#
/sbin/ipfwadm -O -a accept -V $ppp_ip -S $ppp_ip/32 -D 0.0.0.0/0

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

# regle qui attrape tout, refuse tout autre sortie et le logue. Dommage qu'il n'y ait pas
# d'options pour le log sur cette politique mais ceci va faire le travail :
#
/sbin/ipfwadm -O -a reject -S 0.0.0.0/0 -D 0.0.0.0/0 -o


#############################################################################
# Forwarding, flush et politique par defaut de rejet. En fait la politique par defaut
# est inapplicable parce qu'il y une regle qui attrape tout, refuse et logue.
#
/sbin/ipfwadm -F -f
/sbin/ipfwadm -F -p deny

# Masquerade a partir du reseau local sur l'interface locale vers n'importe ou.
#
/sbin/ipfwadm -F -a masquerade -W ppp0 -S 192.168.0.0/24 -D 0.0.0.0/0
#
# regle qui attrape tout, refuse tout autre forwarding et le logue. Dommage qu'il n'y ait pas
# d'options pour le log sur cette politique mais ceci va faire le travail :
#
/sbin/ipfwadm -F -a reject -S 0.0.0.0/0 -D 0.0.0.0/0 -o

#Fin du fichier.
```

Avec IPFWADM, vous pouvez bloquer le traffic vers un site particulier en utilisant les règles -I, -O ou -F. Souvenez vous que ces jeux de règles sont parcourus de début vers la fin et que "-a" dit a IPFWADM d'"ajouter" cette nouvelle règle au jeu de règles existant. Donc, en gardant ceci à l'esprit, toute restriction spécifique a besoin d'être ajoutée avant les règles globales. Par exemple :

avec la règle -I (input ):

Vraisemblablement la méthode la plus efficace et la plus rapide pour bloquer le traffic mais elle arrête seulement les machines MASQuées et NON la machine firewall elle-même. Biensûr, vous pourriez vouloir permettre cette combinaison :

Dans tous les cas, pour bloquer 204.50.10.13:

dans le jeu de règles de /etc/rc.d/rc.firewall :

```
... debut des regles -I ...

# rejette et logue l'interface locale, les machines locales allant a 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, allez n'importe ou est valide
#
/sbin/ipfwadm -I -a accept -V 192.168.0.1 -S 192.168.0.0/24 -D 0.0.0.0/0

... fin des regles -I ...
```

avec la règle -O (output ):

C'est la méthode la plus lente parce que les paquets passent par le masquerading d'abord et sont ensuite éliminés. Cependant, cette règle empêche même la machine firewall d'accéder à des sites interdits.

```
... debut des regles -O ...

# rejette et logue les transimissions 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 autre chose qui sort de 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 regles -O ...
```

avec la règle -F (forward ):

Sans doute plus lent pour bloquer le traffic que les règles -I (input). Ne bloque que les traffics des machines masqueradées (i.e. les machines internes). La machine firewall peut toujours atteindre le(s) site(s) interdit(s).

```
... debut des regles -F ...

# rejette et logue les transmissions de l'interface locale 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 du reseau local vers vers n'importe ou.
#
/sbin/ipfwadm -F -a masquerade -W ppp0 -S 192.168.0.0/24 -D 0.0.0.0/0

... fin des regles -F ...
```

Il n'y a pas besoin de règle spéciale pour permettres aux machines du réseau 192.168.0.0/24 d'aller à 204.50.11.0. Pourquoi ? Parce que c'est déjà traité dans la règle MASQ globale.

NB : Il y a plus d'une façon de coder les interfaces dans les règles si dessus. Par example au lieu de "-V 192.168.255.1", vous pouvez mettre "-W eth0", au lieu de "-V $ppp_ip", vous pouvez utiliser "-W ppp0". La méthode "-V" a été délaissée lors de la migration vers IPCHAINS mais pour les utilisateurs IPFWADM, c'est plus un choix personnel et de documentation plus qu'autre chose.

## 6.5 Règles de d'IP Firewall (IPCHAINS) plus résistants (Stronger)

Cette section fournit un guide plus détaillé sur l'utilisation de l'outil firewall des noyaux 2.2.X, IPCHAINS. Reportez vous ci-dessus pour les jeux de règles IPFAWDM.

Voici un exemple d'un système de firewall/masquerade derrière une connexion PPP avec une adresse PPP statique (les instructions pour les connexions PPP dynamiques sont incluses mais désactivées). L'interface de confiance est 192.168.0.1 et l'adresse IP de l'interface PPP a été changée pour protéger le coupable :). J'ai listé chaque interface entrante et sortante pour détecter aussi bien les IP spoofings que les faux routage et/ou masquerading. Tout ce qui n'est pas explicitement permis est **INTERDIT** (euh... rejeté en fait). Si votre serveur IP MASQ ne fonctionne plus après avoir implémenté ce script rc.firewall, vérifiez que vous l'avez modifié pour votre propre configuration et contrôlez votre fichier SYSLOG /var/log/messages ou /var/adm/messages pour trouver d'éventuels erreurs de firewall.

Pour des exemples plus exhaustifs de règles Ôstrong' d'IPFWADM IP Masqueradé pour PPP, modem pour cable, etc., vous pouvez vous référer à *TrinityOS - Section 10* `http://www.ecst.csuchico.edu/~dranch/LINUX/index-linux.html#TrinityOS` et *GreatCircle's Firewall WWW page*

**NB #1:** les noyaux Linux 2.2.x inférieurs à 2.2.16 ont un trou de sécurité dans la couche TCP (root exploit) et les versions inférieurs à 2.2.11 ont un bug de fragmentation dans IPCHAINS. En raison de cela, les personnes utilisant le jeu de règles 'strong IPCHAINS' sont vulnérables aux attaques. Veuillez donc faire la mise à jour de votre noyau vers une version corrigée.

**NB #2:** Si vous avez une adresse TCP/IP assignée de facon dynamique par votre FAI (PPP, aDSL, Cable, etc.) vous **NE POUVEZ PAS CHARGER** ces règles Ôstrong' au moment du boot. Vous aurez soit à relancer le jeu de règles de ce firewall à CHAQUE FOIS que vous avez une nouvelle adress IP soit faire un /etc/rc.d/rc.firewall plus intelligent. Pour faire ceci pour les utilisateurs de PPP, lisez attentitevement et enlever les marques de commentaires des lignes correspondantes dans la section Òrécuperation d'IP PPP dynamiqueÓ ci dessous. Vous pouvez aussi trouver de plus amples détails dans le documentation *TrinityOS - Section 10* `http://www.ecst.csuchico.edu/~dranch/LINUX/index-linux.html#TrinityOS` sur les jeux de régles ÔStrong' et les adresses IP dynamiques.

**Veuillez aussi noter qu'il existe plusieurs utilitaires de création de Firewall qui possèdent des interfaces graphiques. Vous pouvez vous reporter à la section 7 () pour des détail complets.**

Enfin, si vous utilisez une adresse IP STATIQUE obtenue par PPP, changer la ligne "ppp_ip="votre.adresse.PPP.statique" par votre adresse.

————————————————————————-

```
#!/bin/sh
#
# /etc/rc.d/rc.firewall: An example of a Semi-Strong IPCHAINS firewall ruleset.
#

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

# Charge les modules necessaires a IP MASQ
#
#   NB: Charger uniquement les modules IP MASQ dont vous avez besoin. Tous les modules
#   IP MASQ actuels sont montres ci-dessous mais sont commentes pour les empecher de
#   se charger.

# Necessaire pour le chargement initial des modules
```

```
#
/sbin/depmod -a

# Permet le masquerading correct des transfert de fichier par FTP avec la methode PORT
#
/sbin/modprobe ip_masq_ftp

# Permet le masquerading de RealAudio par UDP. Sans ce module,
#       RealAudio FONCTIONNERA mais en mode TCP. Ce qui peu causer une baisse
#       dans la qualite du son
#
/sbin/modprobe ip_masq_raudio

# Permet le masquerading des transferts de fichier par DCC pour les IRC
#
#/sbin/modprobe ip_masq_irc


# Permet le masquerading de Quake et QuakeWorld par defaut. Ce module est
#   necessaire pour les utilisateurs multiples dirriere un server Linux MASQ. Si vous voulez jouer
#   a Quake I, II, et III, utilisez le second exemple.
#
#   NB:  si vous rencontrez des ERREURs lors de chargement du module QUAKE, c'est que vous utilisez
#   un ancien noyau buggue. Mettez a jour votre noyau pour supprimer l'erreur.
#
#Quake I / QuakeWorld (ports 26000 and 27000)
#/sbin/modprobe ip_masq_quake
#
#Quake I/II/III / QuakeWorld (ports 26000, 27000, 27910, 27960)
#/sbin/modprobe ip_masq_quake 26000,27000,27910,27960


# Permet le masquerading du logiciel CuSeeme pour la video conference
#
#/sbin/modprobe ip_masq_cuseeme

# Permet le masquerading du logiciel VDO-live pour la video conference
#
#/sbin/modprobe ip_masq_vdolive


#CRITIQUE:  Active l'IP forwarding puisqu'il est desactive par defaut
#
#           Utilisateurs Redhat: vous pourrez essayer en changeant les options dans
#                          /etc/sysconfig/network de:
#
#                       FORWARD_IPV4=false
#                            to
#                       FORWARD_IPV4=true
#
```

```
echo "1" > /proc/sys/net/ipv4/ip_forward


#CRITIQUE: Active automatiquement l'IP defragmenting puisqu'il est desactive par defaut
#           dans les noyaux 2.2.x.
#
#           Ceci etait une option de compilation mais ca a change
#           depuis le noyau 2.2.12. Noter aussi que quelques distributions
#           ont enleve cette option de la table /proc. Cette cette entree n'est pas
#           presente dans votre /proc, ne vous inquietez pas.
#
echo "1" > /proc/sys/net/ipv4/ip_always_defrag


# Utilisateurs d'IP Dynamiques:
#
#   Si vous recevez votre adresse IP de maniere dynamique a partir d'un server SLIP, PPP,
#   ou DHCP, activez option suivante qui active le hacking (au bon sens du terme NDT) des
#   adresses IP dynamique dans IP MASQ, rendant ainsi les choses plus faciles pour les
#   programmes du type Diald.
#echo "1" > /proc/sys/net/ipv4/ip_dynaddr


# Active le patch LooseUDP dont certains jeux reseaux ont besoin
#
#  Si vous etes en train d'essayer de faire fonctionner un jeu sur Internet au travers votre
#  serveur MASQ, et vous l'avez configure le mieux que vous pouviez mais que ca fonctionne
#  toujours pas, essayez d'activer cette option (en supprimant le # en debut de ligne). Cette
#  option est desactivee par defaut pour eviter une probable vulnerabilite au port scanning
#  UDP en interne.
#
#echo "1" > /proc/sys/net/ipv4/ip_masq_udp_dloose


# Specifiez ici votre adresse IP statique.
#
# Si vous avez une adresse IP DYNAMIQUE :
# votre jeu de regles doit trouver votre adresse IP a chaque fois que vous avez une nouvelle.
# Dans ce but, activez le script d'une ligne qui suit. (Veuillez SVP noter que les differents
# apostrophes, guillemets etc. ont leur importance et sont distincts).
#
#
#   utilisateurs de DHCP :
#   ----------------------
#   Si vous recevez votre adresse TCP/IP, **vous devez** activer les commandes
#   #es sous la section PPP ET remplacerle mot "ppp0" par le nom de votre
#   de votre connetion Internet EXTERNE (eth0, eth1, etc). Notez aussi que le server
#   DHCP peut changer votre adress IP. Pour resoudre ce probleme, les utilisateurs
#   doivent configurer leur client DHCP de sorte qu'il relance le jeu de regles du firewall
#   chaque fois que leur bail DHCP est renouvele.
```

```
#
#     NB #1:  Quelques clients DHCP comme l'ancienne version de "pump" (les nouvelles
#               versions ont ete corrigees) n'avait pas la capacite de relancer
#               les scripts apres une renouvellement de bail. Pour cette raison, vous
#               aurez besoin de le remplacer par quelquechose du style "dhcpcd" ou "dhclient".
#
#     NB #2:  La syntaxe de "dhcpcd" a change dans les versions recentes.
#
#               Les anciennes version avaient une syntaxe du type:
#                         dhcpcd -c /etc/rc.d/rc.firewall eth0
#
#               Les versions plus recentes ont une syntaxe du type:
#                         dhcpcd eth0 /etc/rc.d/rc.firewall
#
#     NB #3:  Pour les utilisateurs de Pump, ajouter cette ligne de commande dans votre fichier
#    /etc/pump.conf:
#
#                   script /etc/rc.d/rc.firewall
#
#   utilisateurs de PPP :
#   ---------------------
#   Si vous n'etes pas deja au courant, le script /etc/ppp/ip-up est toujours lance quand
#   une connexion PPP arrive. A cause de ca, on peut demander au jeu de regles d'aller recuperer
#   la nouvelles adresse IP PPP et de mettre a jour notre jeu de regles du strong firewall.
#
#   Si le fichier /etc/ppp/ip-up existe deja, vous devez le modifier et ajouter une ligne
#   contenant "/etc/rc.d/rc.firewall" pres de la fin du fichier.
#
#   Si vous n'avez pas encore de script /etc/ppp/ip-up, vous devez creer le lien suivant
#   pour lancer le script /etc/rc.d/rc.firewall.
#
#      ln -s /etc/rc.d/rc.firewall /etc/ppp/ip-up
#
#   * Vous devez ensuite activer les commandes #ees si dessous *
#
#
# Utilisateurs de PPP et DHCP :
# -----------------------------
# Enlevez le # de la ligne si dessous et placez un # sur la ligne suivante.
#
#extip="`/sbin/ifconfig ppp0 | grep 'inet addr' | awk '{print $2}' | sed -e 's/.*://'`"

# Pour les utilisateurs de PPP avec une adresse IP statique:
#
extip="your.static.PPP.address"

# Tous les utilisateurs de PPP et DHCP doivent utiliser ceci pour corriger le nom de
# leur interface EXTERNE
extint="ppp0"
```

```
# Assigne l'IP interne
intint="eth0"
intnet="192.168.0.0/24"


# MASQ timeouts
#
#  timeout de 2 heures pour les sessions TCP
#  timeout de  10 sec pour le traffic apres que le paquet TCP/IP "FIN" est recu
#  timeout de 160 sec pour le traffic UDP (Important pour les utilisateur d'ICQ MASQues)
#
ipchains -M -S 7200 10 60

##############################################################################
# Entree (incoming), flush et politique par defaut de rejet. En fait la politique par defaut
# est inapplicable parce qu'il y une regle qui attrape tout, refuse et logue.
#
ipchains -F input
ipchains -P input REJECT

# interface locale, machines locales, on peut allez n'importe ou
#
ipchains -A input -i $intint -s $intnet -d 0.0.0.0/0 -j ACCEPT

# interface distance, pretendant etre une machine locale, IP spoofing, tire toi
#
ipchains -A input -i $extint -s $intnet -d 0.0.0.0/0 -l -j REJECT

# interface distante, toute source, peut aller a l'adresse PPP permanante
#
ipchains -A input -i $extint -s 0.0.0.0/0 -d $extip/32 -j ACCEPT

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

# regle qui attrape tout, refuse tout autre entree et le logue. Dommage qu'il n'y ait pas
# d'options pour le log sur cette politique mais ceci va faire le travail :
#
ipchains -A input -s 0.0.0.0/0 -d 0.0.0.0/0 -l -j REJECT

##############################################################################
# Sortie (outgoing), flush et politique par defaut de rejet. En fait la politique par defaut
# est inapplicable parce qu'il y une regle qui attrape tout, refuse et logue.
#
ipchains -F output
ipchains -P output REJECT

# interface locale, toute source allant vers le reseau local est valide
#
```

```
ipchains -A output -i $intint -s 0.0.0.0/0 -d $intnet -j ACCEPT

# sortie vers le reseau local d'une interface distance, routage bizarre, rejet
#
ipchains -A output -i $extint -s 0.0.0.0/0 -d $intnet -l -j REJECT

# sortie du reseau local vers une interface distante, masquerading modifie, rejet
#
ipchains -A output -i $extint -s $intnet -d 0.0.0.0/0 -l -j REJECT

# tout autre chose qui sort de l'interface distance est valide
#
ipchains -A output -i $extint -s $extip/32 -d 0.0.0.0/0 -j ACCEPT

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

# regle qui attrape tout, refuse tout autre sortie et le logue. Dommage qu'il n'y ait pas
# d'options pour le log sur cette politique mais ceci va faire le travail :
#
ipchains -A output -s 0.0.0.0/0 -d 0.0.0.0/0 -l -j REJECT

#############################################################################
# Forwarding, flush et politique par defaut de rejet. En fait la politique par defaut
# est inapplicable parce qu'il y une regle qui attrape tout, refuse et logue.
#
ipchains -F forward
ipchains -P forward DENY

# Masquerade a partir du reseau local sur l'interface locale vers n'importe ou.
#
ipchains -A forward -i $extint -s $intnet -d 0.0.0.0/0 -j MASQ
#
# regle qui attrape tout, refuse tout autre forwarding et le logue. Dommage qu'il n'y ait pas
# d'options pour le log sur cette politique mais ceci va faire le travail :
#
ipchains -A forward -s 0.0.0.0/0 -d 0.0.0.0/0 -l -j REJECT

#Fin du fichier.
```

Avec IPCHAINS, on peut bloquer le traffic vers un site particuler grâce aux règles "input", "output", et/ou "forward". Souvenez vous que les jeux de règles sont traitées de haut en bas et que "-A" dit a IPCHAINS de "coller" une nouvelle règle au jeu de règles existant. Donc, avec ça en tête, toute règle spécifique doit venir avant les règles globales. Par exemple :

Avec la règle "input" :

Vraisemblablement la méthode la plus efficace et la plus rapide pour bloquer le traffic mais elle arrête seulement les machines MASQuées et NON la machine firewall elle-même. Biensûr, vous pourriez vouloir permettre cette combinaison :

Dans tous les cas, pour bloquer 204.50.10.13:

dans le jeu de règles de /etc/rc.d/rc.firewall :

```
... debut des regles -I ...

# rejette et logue l'interface locale, les machines locales allant a 204.50.10.13
#
ipchains -A input -s 192.168.0.0/24 -d 204.50.10.13/32 -l -j REJECT


... fin des regles -I ...
```

avec la règle -O (output ):

C'est la méthode la plus lente parce qeu les paquets passent par le masquerading d'abord et sont ensuite éliminés. Cependant, cette règle empêche meme la machine firewall d'accéder à des sites interdits.

```
... debut des regles -O ...

# rejette et logue les transimissions sortantes vers 204.50.10.13
#
ipchains -A output -s $ppp_ip/32 -d 204.50.10.13/32 -l -j REJECT


# tout autre chose qui sort de l'interface distante est valide
#
ipchains -A output -s $ppp_ip/32 -d 0.0.0.0/0 -l -j ACCEPT

... fin des regles -O ...
```

avec la règle -F (forward ):

Sans doute plus lent pour bloquer le traffic que les règles -I (input). Ne bloque que les traffics des machines masqueradées (i.e. les machines internes). La machine firewall peut toujours atteindre le(s) site(s) interdit(s).

```
... debut des regles -F ...

# rejette et logue les transmissions de l'interface locale PPP vers 204.50.10.13.
#
ipchains -A forward -i ppp0 -s 192.168.0.0/24 -d 204.50.10.13/32 -l -j REJECT


# Masquerade du reseau local vers vers n'importe ou.
#
ipchains -A forward -i ppp0 -s 192.168.0.0/24 -d 0.0.0.0/0 -j MASQ

... fin des regles -F ...
```

Il n'y a pas besoin de règle spéciale pour permettre aux machines du réseau 192.168.0.0/24 d'aller à 204.50.11.0. Pourquoi ? Parce que c'est déjà traité dans la règle MASQ globale.

NB : Contrairement à IPFWADM, IPCHAINS n'a seulement qu'une manière de coder le nom des interfaces. IPCHAINS utilise l'option "-i eth0" là ou IPFADM avait le "-W" pour le nom de l'interface et le "-V" pour l'adresse IP de l'interface.

## 6.6 IP Masquerader plusieurs réseaux internes

Masquerader plus d'un réseau interne est une tâche plutôt simple. Vous devez d'abord vérifier que tous vos réseaux fonctionnent correctement (internes et externes). Vous devez ensuite permettre au traffic de passer dans les interfaces internes et d'être MASQués vers Internet.

Ensuite, vous devez activer le Masquerading sur les interfaces INTERNES. Cet exemple utilise au total TROIS interfaces : eth0 est une connexion EXTERNE vers Internet, eth1 est le réseau 192.1680.0.0, et eth2 et le réseau 192.168.1.0. eth1 et eth2 vont tous deux être MASQués au travers de l'interface eth0. Dans votre jeu de règles rc.firewall, juste à côté de votre ligne activant MASQ, ajoutez ce qui suit :

- Noyaux 2.2.x avec IPCHAINS

```
#Active la communication entre les interfaces internes
/sbin/ipchains -A forward -i eth1 -d 192.168.0.0/24
/sbin/ipchains -A forward -i eth2 -d 192.168.1.0/24

#Permet aux interfaces internes de MASQuer vers 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
```

- noyaux 2.0.x avec IPFWADM

```
#Active la communication entre les interfaces internes
/sbin/ipfwadm -F -a accept -V 192.168.0.1 -D 192.168.1.0/24
/sbin/ipfwadm -F -a accept -V 192.168.1.1 -D 192.168.0.0/24

#Permet aux interfaces internes de MASQuer vers 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
```

Notez qu'il est CORRECT d'avoir spécifé "eth0" plusieurs fois dans les exemples ci-dessus. La raison est que le noyau Linux a besoin de savoir quel interface est utilisé por le traffic SORTANT. Puisque eth0 est la connexion Internet dans les exemples précédents, elle est listée pour chaque interface interne.

## 6.7 IP Masquerade et les connexions téléphoniques sur demande

1. Si vous voulez configurer votre réseau de manière à ce qu'il se connecte automatiquement par modem téléphonique à Internet, soit le package *Diald* de connexion sur demande soit les nouvelles versions de *PPPd* vous sernt d'une grande utilité. Diald est la solution recommandée en raison de sa configuration plus précise.
2. Pour configurer Diald, reportez vous a la page suivant : *Setting Up Diald for Linux Page* `http://home.pacific.net.sg/~harish/diald.config.html` ou à celle ci : *TrinityOS - Section 23* `http://www.ecst.csuchico.edu/~dranch/LINUX/index-linux.html#TrinityOS`
3. Une fois que Diald et IP Masq ont été configurés correctement, toute machine cliente MASQuée qui commence une session web, telnet ou ftp va provoquer la connexion dynamique par la machine Linux à Internet.

4. Un timeout va avoir lieu pour la première connexion. C'est inévitable si vous utilisez un modem analogique. Le temps d'établir la connexion du modem et la connexion PPP peuvent être suffisamment longs pour provoquer des timeouts dans votre programme client (browser WWW, etc.). Toutefois, ce n'est pas commun. Si ca vous arrive, essayer simplement de relancer votre requêtre Internet (disons, la page web) et ca devrait marcher. Vous pouvez aussi essayer de mettre l'option noyau suivant : *echo "1" > /proc/sys/net/ipv4/ip_ dynaddr* pour vous aider dans avec cette configuation initiale.

## 6.8 IPPORTFW, IPMASQADM, IPAUTOFW, REDIR, UDPRED, et d'autres outils de Port Forwarding

IPPORTFW, IPAUTOFW, REDIR, UDPRED, et les autres programmes sont des outils de port forwarding TCP et/ou UDP génériques pour Linux IP Masquerade. Ces outils sont typiquement utilisés avec ou en remplacement de modules IP MASQ spécifiques tels que ceux pour FTP, Quake, etc. Avec les port forwarders, vous pouvez rédiriger les connexions venant d'Internet vers une machine interne derrière le serveur IP MASQ, dont l'adresse est privée . Cette possibilité de forwarding inclus les protocoles réseaux tels que TELNET, WWW, SMTP, FTP (avec un patch special - regardez si dessous), ICQ, et bien d'autres.

NB : Si vous voulez juste faire un simple port forwarding sans IP Masquerading, vous aurez **TOUJOURS BESOIN** d'activer l'IP Masquerading dans votre noyau ET soit dans votre jeu de règles IPFWADM soit IPCHAINS pour être capable d'utiliser les outils de portforwarding de Linux

Alors pourquoi tous ces choix ? IPAUTOFW, REDIR, et UDPRED (toutes les URLs sont dans la section 2.7 ()) étaient les premiers outils disponibles pour les utilisateurs d'IP MASQ pour permettre cette fonctionnalié. Plus tard, quand Linux IP Masquerade a mûri, ces outils furent remplacés par IPPORTFW qui constitue une solution plus intelligente. A cause de la disponibilité d'outils nouveaux, il est *FORTEMENT DECONSEILLE* d'utiliser les outils tels que IPAUTOFW et REDIR parce qu'ils n'informent pas correctement le noyau de leur présence et peuvent dans les cas les plus extrêmes d'utilisation **CRASHer** votre serveur Linux. Notez aussi que la solution la plus recente est MFW. Son avantage principal est de permettre une intégration plus étroite avec l'outil IPCHAINS. Avec cette solution, vous utilisez un jeu de règles IPCHAINS pour "Marker" un paquet spécifique et créer ensuite une chaine différente pour faire ensuite le bon forwarding. Cette méthode n'est pas encore traitée dans ce HOWTO.

NB #2 : avec PORTFW sur les noyaux 2.2.x, *les machines internes* NE PEUVENT PAS utiliser la meme adresse IP PORTFWdé pour acceder une machine interne bienque ça marche très bien avec des machines externes sur Internet. Si c'est un probèleme pour vous, vous pouvez AUSSI implémenter l'outil portfw REDIR pour laisser des machines internes être redirigées vers un serveur interne. Une chose à noter est que le jeu de règles de l'imminent 2.7 () résoud ce problème. Si vous désirez avoir une explication technique sur les raisons du non fonctionnement du forwarding interne/externe, reportez vous SVP à la fin de la section PORTFW du noyau 2.2.x pour les notes de Juan.

NB #3 : Le forwarding du traffic de serveurs FTP vers un serveur FTP MASQué interne, connu sous le nom de **PORTFW FTP**, est maintenant compatible avec les noyaux 2.0.x et les noyaux 2.2.x . C'est possible soit en patchant le noyau Linux si la compatibilité n'est pas encore implémenté dans votre noyau ou bien un utilisant un programme de proxy FTP externe. Vous devriez aussi noter que le code du module noyau est toujours expérimental et que certaines personnes obtiennent de meilleurs résultats avec des sessions FTP ACTIVES par rapport aux connexions PASSIVES. Et, chose assez intéressante, d'autres personnes obtiennent exactement le contraire. Envoyez nous SVP vos résultats. De plus amples détails sont fournis dans les sections 2.2.x et 2.0.x en tant que solutions fournis pas les différents patches.

Avant de plonger dans l'installation de IPPORTFW pour 2.0.x ou de la version de 2.2.x de IPMASQADM avec le support IPPORTFW, veuillez noter qu'il peut y avoir des problèmes liés à la sécurité avec tout port forwarder. La raison est que ces outils créent un trou dans le firewall par paquet pour les port TCP/UDP forwardés. Bienque cela ne conduise à aucune menace pour votre machine Linux, ca pourrait être un problème pour la machine interne vers lequel ce traffic est forwardé. Ne vous inquietez pas non plus, voilà ce que Steven Clarke (l'auteur de IPPORTFW) a à dire sur ce sujet :

```
"Port Forwarding est appele seulement parmis les fonctions de masquerading il suit donc
les meme regles que IPFWADM/IPCHAINS. Masquerading est une extension de IP forwarding.
Toutefois, ipportfw ne vois les paquets que s'ils remplissent les conditions
d'entree et de masquerading du jeu de regles d'ipfwadm."
```

Maintenant que l'on a dit ceci, il est important d'avoir un jeu de règles de firewall 'strong'. Reportez vous SVP aux sections 6.4 () et 6.5 () pour de plus amples détails sur les jeux de règles 'strongs'.

Donc, pour installer l'IPPORTFW forwarding pour chacun des noyaux 2.0.x ou 2.2.x, vous devez recompiler le noyau Linux avec la compatibilité IPPORTFW.

- les utilisateurs de noyaux 2.2.x vont déjà avoir l'option noyau IPPORTFW de disponible via IPMASQADM
- les utilisateurs de noyaux 2.0.x vont avoir besoin d'appliquer un patch d'option noyau simple

### 6.8.1 IPMASQADM avec compatibilité IPPORTFW sur les noyaux 2.2.x

D'abord, vérifiez que vous avez les sources du noyau 2.2.x le plus récent dans /usr/src/linux. Si vous ne l'avez pas dejà fait, reportez vous SVP à la section 3.1 () pour de plus amples détails. Ensuite, téléchargez le programme "ipmasqadm.c" de la section 2.5 () dans le repertoire /usr/src.

Vous aurez ensuite à compiler le noyau 2.2.x comme expliqué dans la section 3.1 (). Vérifiez bien que vous dites YES à l'option IPPORTFW quand vous configurez votre noyau. Une fois que vous avez compilé le noyau et que vous avez rebooté, revenez à cette section.

Maintenant, compilez et installez l'outil IPMASQADM :

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

Ensuite, pour cet exemple, nous allons permettre à TOUT le traffic Internet WWW (port 80) arrivant à votre adresse Internet TCP/IP d'être forwardé vers une machine interne Masqueradée dont l'IP est 192.168.0.10.

PORTFW FTP : Comme mentionné precédemment, il y a deux solutions pour forwarder le traffic d'un server FTP vers une machine interne MASQuée. La première solution *EST* le module BETA *IP_MASQ_FTP* pour noyau 2.2.x pour PORT Forwarder les connexions FTP vers un serveur FTP interne MASQué. L'autre méthode est d'utiliser un programme de proxy FTP (l'URL se trouve dans la section 2.5 (). Vouz devriez aussi noter que le module noyau FTP permet aussi d'ajouter des PORTFW FTP supplémentaires à la volée sans avoir à relancer le module et ainsi planter les connexions FTP en cours. Vous trouverez de plus amples détails sur le site d'IPMASQ WWW à `http://ipmasq.cjb.net` . Il y a aussi des exemples et des informations sur les connexions FTP PORTFWés ci dessous dans la section du noyau 2.0.x.

**NB:** Une fois le port forward du port 80 activé, ce port ne pourra plus être utilisé par le serveur Linux IP Masquerade. Pour être plus précis, si vous avez un server WWW sur le server MASQ, un portfw va maintenant diriger tous les internautes vers les pages WWW INTERNES et non vers celles du serveur IPMASQ.

Dans tous les cas, pour activer le port forwarding, modifiez le jeu de règles /etc/rc.d/rc.firewall. Ajoutez les lignes suivant mais assurez vous de remplacer le mot "$extip" par votre adresse IP.

**NB:** Si vous avez une adresse IP DYNAMQUE que vous recevez par votre FAI (PPP, ADSL, Cablemodems, etc.), vous aurez BESOIN de rendre votre jeu de règles /etc/rc.d/rc.firewall plus intelligent. Pour ce faire, reportez vous SVP à la section 6.5 () ci-dessus ou à la section *TrinityOS - Section 10* `http://www.ecst.csuchico.edu/~dranch/LINUX/index-linux.html#TrinityOS` pour de plus amples détails sur les jeux de règles 'strong' et les adresses IP Dynamiques. Je vous donne une indication toutefois : /etc/ppp/ip-up pour les utilisateurs de PPP.

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

#echo "Activation de l'IPPORTFW sur le LAN externe..."
#
/usr/sbin/ipmasqadm portfw -f
/usr/sbin/ipmasqadm portfw -a -P tcp -L $extip 80 -R 192.168.0.10 80

--
```

C'est tout ! Relancez juste votre jeu de règles /etc/rc.d/rc.firewall et testez le !

Si vous recevez le message d'erreur "ipchains: setsockopt failed: Protocol not available", c'est que vous n'êtes pas en train d'utiliser le nouveau noyau. Verifiez que vous avez bien installé le nouveau noyau, reconfigurez votre boot loader (par exemple LILO), et ensuite, rebootez. Si vous êtes sûr que vous êtes sur le nouveau noyau, lancez la commande "ls /proc/net/ip_masq" et verifiez que le fichier "portfw" existe. Si non, vous devez avoir fait une erreur lors de la configuation de votre noyau. Essayez de nouveau.

Pour ceux qui veulent comprendre pourquoi PORTFW ne peut pas rediriger le traffic des interfaces externes et internes, voici un email de Juanjo qui l'explique mieux :

---

```
De Juanjo Ciarlante
--

>Si j'utilise :
>
> ipmasqadm portfw -a -P tcp  -L 1.2.3.4 80 -R 192.168.2.3 80
>
>Tout fonctionne tres bien a partir de l'exterieur mais les requetes internes pour la meme
>adresse 1.2.3.4 echouent. Y a t-il des chaines qui permettent a une machine sur le reseau local
>192.168.2.0 d'acceder a www.periapt.com sans utiliser de proxy ?

En fait, non.

D'habitude, je mets en place une regle ipmasqadm pour l'exterieur, *ET* un port
redirector pour l'interieur. Ceci fonctionne parce que ipmasqadm connecte avant que redir
ne recoive l'eventuelle connexion exterieur, _mais_ laisse les choses comme elles sont sinon
(gere par des regles APPROPRIEES).

Le vrai probleme "conceptuel" provient du fait que la VRAIE IP du client (peer) cible
est sur le meme reseau que le serveur cible

Le scenario d'un echec pour le "local masq" est :
```

```
   client: 192.168.2.100
   masq:   192.168.2.1
   serv:   192.168.2.10

1)client->server packet
 a) client:  192.168.2.100:1025  -> 192.168.2.1:80   [SYN]
 b) (masq):  192.168.2.100:1025  -> 192.168.2.10:80  [SYN]
            (et garde 192.168.2.1:61000 192.168.2.100:1025 apparentes)
 c) serv:    recoit le paquet masque (1b)

2)server->client packet
 a) serv:    192.168.2.10:80     -> 192.168.2.100:1025  [SYN,ACK]
 b) client:  192.168.2.100:1025  -> 192.168.2.10:80     [RST]

Maintenant prenez le temps de comparer (1a) avec (2a).
Vous voyez, le serveur a repondu DIRECTEMENT au client sans passer par
masq (ne laissant donc pas masq ANNULER la modification du paquet) parce que
c'est le MEME reseau, donc le client annule la connexion.

J'espere que cela aide.

Amicalement,
Juanjo
```

---

### 6.8.2 IPPORTFW sur noyaux 2.0.x

D'abord, vérifiez que vous avez les sources du noyau 2.0.x le plus récent dans /usr/src/linux. Si vous ne l'avez pas dejà fait, reportez vous SVP à la section 3.1 () pour de plus amples détails. Ensuite, téléchargez le programme "ipmasqadm.c" de la section 2.7 () dans le repertoire /usr/src.

Ensuite, si vous projetez de port forwarder le traffic FTP vers un serveur interne, vous allez devoir appliquer un **NOUVEAU** patch module additionnel, *IP_MASQ_FTP*, que vous trouverez à la section 2.7 (). De plus amples détails le concernant se trouvent plus loin dans cette section. Veuillez noter SVP que ce n'est pas le même patch que pour les noyaux 2.2.x donc quelques fonctionnalités telles que le dynamique FTP PORT n'est pas présent.

Maintenant, copiez le patch IPPORTFW (subs-patch-x.gz) dans le repertoire de Linux

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

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

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

Ok, il est temps de compiler le noyau comme indiqué à la section 3.1 (). Repondez YES à l'option IPPORTFW qui est maintenant disponible quand vous configurez le noyau. Une fois la compilation terminée, et après avoir rebooté, vous pouvez revenir à cette section.

Maintenant, avec votre nouveau noyau fraichment compilé, compilez et installer le programme "IPPORTFW"

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

Ensuite, pour cet exemple, nous allons permettre à TOUT le traffic Internet WWW (port 80) arrivant à votre adresse Internet TCP/IP d'être forwardé vers une machine interne Masqueradée dont l'IP est 192.168.0.10.

**NB:** Une fois le port forward du port 80 activé, ce port ne pourra plus être utilisé par le serveur Linux IP Masquerade. Pour être plus spécifique, si vous avez un server WWW sur le server MASQ, un portfw va maintenant diriger tous les internautes vers les pages WWW INTERNES et non vers celles du serveur IPMASQ. La seule solution à ce problème est de port fowarder un autre port, disons 8080, vers votre machine MASQ interne. Bienque cela fonctionne, tous les internautes devront coller un *:8080* à l'URL pour pouvoir contacter votre server WWW MASQué interne.

Dans tous les cas, pour activer le port forwarding, modifiez le jeu de règles /etc/rc.d/rc.firewall. Ajoutez les lignes suivant mais assurez vous de remplacer le mot "$extip" par votre adresse IP.

**NB:** Si vous avez une adresse IP DYNAMQUE que vous recevez par votre FAI (PPP, ADSL, Cablemodems, etc.), vous aurez BESOIN de rendre votre jeu de règles /etc/rc.d/rc.firewall plus intelligent. Pour ce faire, reportez vous SVP à la section 6.5 () ci dessus ou à la section *TrinityOS - Section 10* `http://www.ecst.csuchico.edu/~dranch/LINUX/index-linux.html#TrinityOS` pour de plus amples détails sur les jeux de règles 'strong' et les adresses IP Dynamiques. Je vous donne un peu indice toutefois : /etc/ppp/ip-up pour les utilisateurs de PPP.

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

        #echo "Activation de l'IPPORTFW sur le LAN externe..."
        #
        /usr/local/sbin/ipportfw -C
        /usr/local/sbin/ipportfw -A -t$extip/80 -R 192.168.0.10/80

    #  Veuillez notez SVP que le  PORTFWing du port 20 N'EST PAS necessaire pour les
    #  connexions ACTIVES puisque le serveur FTP interne va lancer cette connexion
    #  sur le port 20 et qu'il va donc etre correctement pris en charge par les mechanismes
    #  classiques de MASQ.
        --
```

C'est tout ! Relancez juste votre jeu de règles /etc/rc.d/rc.firewall et testez le !

Si vous recevez le message d'erreur "ipchains: setsockopt failed: Protocol not available", c'est que vous n'êtes pas en train d'utiliser le nouveau noyau. Verifiez que vous avez bien installé le nouveau noyau, reconfigurez votre boot loader (par exemple LILO), et ensuite, rebootez.

Port Forwarder des serveurs FTP :

Si vous projetez de port forwarder FTP vers une machine interne, les choses se compliquent. La raison en est que le module noyau *IP_MASQ_FTP* standard n'était pas écrit pour ça, meme si des utilisateurs nous ont dit que cela fonctionnait sans problème. Personnellement, sans le patch, j'ai entendu dire que les transferts de fichiers longs, qui excèdent 30 minutes vont échouer alors que d'autres personnes jurent que ça fonctionne sans problème. Quoiqu'il en soit, je vous recommande d'essayer cette instruction PORTFW avec le module STOCK ip_masq_ftp et de voir si ca fonctionne pour vous. Si ca marche pas, essayez d'utiliser le module ip_masq_ftp modifié.

Pour ceux qui ont besoin du module, Fred Viles a écrit un module IP_MASQ_FTP modifié pour faire en sorte que ca fonctionne. Si vous êtes curieux et que vous voulez savoir EXACTEMENT ce que sont les problèmes, téléchargez l'archive suivante parce que les documents de Fred sont très bien faits. Vous devez aussi comprendre que le patch est quelque peu expérimental et considerez le donc comme tel. Vous devez aussi noter que ce patch fonctionne UNIQUEMENT sur les noyaux 2.0.x puisqu'il y a un patch différent disponible pour les noyaux 2.2.x.

Donc, pour faire fonctionner le patch 2.0.x, vous avez besoin de :

- Appliquer en PREMIER le patch noyau IPPORTFW comme expliqué précédemment dans la section.
- Downloader le "msqsrv-patch-36" du serveur FTP de Fred Viles de la section 2.7 () et le mettre dans /usr/src/linux.
- Patcher le noyau avec ce nouveau code en lançant la commande "cat msqsrv-patch-36 | patch -p1"
- Ensuite, remplacer le module noyau *"ip_masq_ftp.c"* original par le nouveau fichier
  - 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, compiler et installer le noyau avec le nouveau code à la place.

Une fois que vous avez fait tout ça, modifiez votre jeu de règles /etc/rc.d/rc.firewall et ajoutez les lignes suivantes en prenant soin de remplacer "$extip" par votre propre adresse IP.

Cet exemple, comme ci-dessus, va permettre de renvoyer TOUT le traffic internet FTP (port 21) de votre connexion Internet TCP/IP vers la machine interne Masqueradée dont l'adresse IP est 192.168.0.10.

**NB:** Une fois le port forward du port 21 activé, ce port ne pourra plus être utilisé par le serveur Linux IP Masquerade. Pour être plus précis, si vous avez un server FTP sur le server MASQ, un portfw va maintenant diriger tous les internautes vers les pages FTP INTERNES et non vers celles du serveur IPMASQ.

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

        #echo "Activation de l'IPPORTFW sur le LAN externe..."
        #
        /usr/local/sbin/ipportfw -C
        /usr/local/sbin/ipportfw -A -t$extip/21 -R 192.168.0.10/21

        #NB :  Si vous allez utiliser plusieurs port locaux a PORTFWer
        #      vers plusieurs seveurs FTP internes (disons, 21, 2121, 2112,
        #      etc), vous devez configurer le module ip_masq_ftp pour qu'il
        #      ecoute ces ports. Pour ce faire, modifiez votre script
        #      /etc/rc.d/rc.firewall comme le montre ce HOWTO
        #      pour qu'il ressemble a ceci :
        #
        # /sbin/modprobe ip_masq_ftp ports=21,2121,2112
        #
        # Relancez le script /etc/rc.d/rc.firewall pour que les changements
        # prennent effet.

#Veuillez notez SVP que le  PORTFWing du port 20 N'EST PAS necessaire pour les
#  connexions ACTIVES puisque le serveur FTP interne va lancer cette connexion
#  sur le port 20 et qu'il va donc etre correctement pris en charge par les mechanismes
#  classiques de MASQ.
        --
```

C'est tout ! Relancez juste votre jeu de règles /etc/rc.d/rc.firewall et testez le !

Si vous recevez le message d'erreur "ipchains: setsockopt failed: Protocol not available", c'est que vous n'êtes pas en train d'utiliser le nouveau noyau. Verifiez que vous avez bien installé le nouveau noyau, reconfigurez votre boot loader (par exemple LILO), et ensuite, rebootez.

## 6.9 CU-SeeMe et Linux IP-Masquerade

Linux IP Masquerade est compatible avec CuSeeme via le module noyau *"ip_masq_cuseeme"*. Ce module noyau devrait être chargé par le script /etc/rc.d/rc.firewall. Une fois que le module "ip_masq_cuseeme" est installé, vous devriez être capables de recevoir et d'initier des connexions CuSeeme vers des réflecteurs distants et/ou des utilisateurs.

NB : Il est recommandé d'utiliser l'outil IPPORTFW au lieu du vieux IPAUTOFW pour utiliser CuSeeme.

Si vous avez besoin d'information explicites sur la configuration de CuSeeme, vous pouvez vous reporter à *Michael Owings's CuSeeMe page* `http://www.swampgas.com/vc/ipmcus.htm` pour un Mini-HOWTO ou *The IP Masquerade Resources* pour un mirroir de ce Mini-HOWTO.

## 6.10 Mirabilis ICQ

Il y a deux méthodes pour faire fonctionner ICQ derrière un serveur Linux MASQ. Une des solutions est d'utiliser le nouveau module ICQ Masq et l'autre solution est d'utiliser IPPORTFW.

Le modue ICQ a quelques avantages. Il permet une configuration simple pour plusieurs utilisateurs ICQ derrière un server MASQ. Il ne requiert pas non plus de changement dans les clients ICQ. Récemment, la version 2.2.x du module a été mis à jour pour permettre le transfert de fichier et le "chat" en temps-réél. Toutefois, la version 2.0.x du module n'est pas parfait. Toutefois, je pense maintenant que c'est la MEILLEURE méthode pour faire fonctionner ICQ avec IP Masq sous les noyaux 2.2.x.

Pour la configuration IPPORTFW, vous allez devoir faire quelques changements sur Linux et sur les clients ICQ mais toutes les fonctionnalités d'ICQ (messages, URLs, chat, transfert de fichier, etc.) fonctionnent.

Si vous êtes interessés par le module IP Masq ICQ pour les noyaux 2.2.x d'Andrew Deryabin's *djsf@usa.net* , vous pouvez vous reporter à la section 2.5 () pour plus de détails.

- D'abord, vous avez besoin d'un noyau linux avec IPPORTFW d'activé. Reportez vous à la section 6.7 () pour de plus amples détails.
  - Ensuite, vous avez besoin d'ajouter les lignes suivantes à votre fichier /etc/rc.d/rc.firewall file. Cet exemple suppose que votre IP externe est 10.1.2.3 et que votre machine interne ICQ MASQuées est 192.168.0.10 : L'exemple qui suit est pour les noyaux 2.0.x avec IPFWADM:

```
J'ai inclus deux exemples ici pour l'utilisateur : les deux fonctionnent 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 [ $port -le 2020 ]
  do
      /usr/local/sbin/ipportfw -A t10.1.2.3/$port -R 192.168.0.10/$port
      port=$((port+1))
  done
--
```

L'exemple suivant est pour les noyaux 2.2.x avec IPCHAINS:

```
J'ai inclus deux exemples ici pour l'utilisateur : les deux fonctionnent 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 -le 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 prêt, rechargez le jeu de règles en tapant "/etc/rc.d/rc.firewall" pour être sûr que tout va bien. Si vous recevez une erreur, c'est que soit vous n'avez pas un noyau qui supporte IPPORTFW soit que vous avez fait une erreur de frappe dans votre fichier rc.firewall.
  – Maintenant, dans les préférences d'ICQ, Preferences–>Connection, mettez "Behind a LAN" et "Behind a firewall or Proxy". Ensuite, cliquez sur "Firewall Settings" et mettez "I don't use a SOCK5 proxy". Notez aussi que l'on recommandait précédemment de changer la préférence d'ICQ "Firewall session timeouts" à "30" seconds MAIS de nombreux utilisateurs ont trouvé que ca le rendait non fiable. On a trouvé qu'ICQ est plus fiable avec son réglage de timeout par défaut (n'activez donc pas cet option d'ICQ) et changez plutôt le timeout de MASQ à 160 secondes. Vous pouvez changer ce timeout dans les jeu de règles 3.3.1 () et 3.3 (). Enfin, cliquez sur Next et configure ICQ "Use the following TCP listen ports.." pour mettre de "2000" à "2020". Maintenant, cliquez sur "done". Ensuite ICQ vous demandera de relancer ICQ pour que les changements prennent effet. Pour être tout a fait honnête, j'ai du REBOOTER la machine Windows9x pour faire marcher tout ca mais d'autres personnes disent que ca marche très bien autrement. Donc... essayez les deux manières.

- Notez aussi qu'un utilisateur m'a dit que portforwarder le port 4000 vers sa machine était ce qui fonctionnait le mieux. Il a dit que tout fonctionnait parfaitement (chat, transfert de fichier, etc.) SANS avoir besoin de reconfigurer ICQ et changer ses réglages par défaut. Vos résultats peuvent varier sur ce sujet mais j'ai pensé que ça pourrait vous paraître intéressant de savoir que cette configuration alternative peut exister.

## 6.11 Joueurs : Le patch LooseUDP

Le patch LooseUDP permet au jeux en réseau compatible NAT qui utilisent des connexion UDP de FONCTIONNER et d'avoir de bonnes performances derrière un serveur Linux IP Masquerade. Pour le moment, LooseUDP est disponible comme patch pour les noyaux 2.0.36+ mais se trouve par défaut dans les noyaux 2.2.3+ bien qu'il est DESACTIVE par DEFAUT dans 2.2.16+

Pour faire fonctionner LooseUDM sur un noyau 2.0.x, suivez les étapes suivantes :

- D'abord, verifiez que vous avez les sources du noyau 2.0.x le plus récent dans /usr/src/linux
- ABSOLUMENT NECESSAIRE pour v2.0.x : Downloadez et installez le patch IPPORTFW que vous trouverez dans la secion 2.7 () et comme décrit dans la section 6.7 () de ce HOWTO.
- Downloadez le patch LooseUDP à partir de la section 2.7 () Ensuite, placez le patch LooseUDP dans /usr/src/linux. Une fois que c'est fait, tappez :

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

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

Ensuite, suivant la version de votre "patch", vous allez voir 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 les texte "Hunk FAILED" UNE et UNE SEULE fois, au tout début de la processure de patching, ne vous inquiétez pas. Vous avez probablement un ancien fichier de patch (ce problème à été réparé) mais ca fonctionne quand même. Si ça échoue complètement, vérifiez que vous avez appliqué le patch IPPORTFW AVANT ce patch.

Une fois ce patch installé, reconfigurez votre noyau comme expliqué dans la section 3.1 () et verifiez bien que vous dites "Y" à l'option "IP: loose UDP port managing (EXPERIMENTAL) (CONFIG_IP_MASQ_LOOSE_UDP) [Y/n/?]".

Pour faire fonctionner LooseUDM sur un noyau 2.2.x, suivez les étapes suivantes :

- Dans le script /etc/rc.d/rc.firewall, allez à la fin du fichier et trouvez la section LooseUDP. Changez le "0" dans la ligne : `echo "0" > /proc/sys/net/ipv4/ip_masq_udp_dloose` en "1" et relancez votre jeu de règles rc.firewall. Un exemple de ceci est fourni dans les exemples des sections 3.3 ()et 6.5 ().

Une fois que vous avez relancé le nouveau noyau avec le LooseUDP activé, vous devriez être en conditions pour la plupart des jeux compatibles NAT. Nous vous fournissons quelques URL pour des patchs qui feront fonctionner des jeux tels que BattleZone ou d'autres jeux compatibles NAT. Vous pouvez vous reporter à la section 6.3.1 () pour de plus amples détails.

# 7 Frequently Asked Questions (Foire Aux Questions)

Si vous avez des suggestions utiles à la FAQ à faire, veuillez envoyez SVP un email en anglais à *dranch@trinnet.net* . Veuillez clairement rédiger la question et ca réponse (si vous l'avez). Merci !

## 7.1 Quelles distributions sont fournis directement avec IP Masquerading ?

Si votre distribution Linux ne n'est pas fourni directement avec IP MASQ, ne vous inquiétez pas. Tout ce que vous avez à faire est de recompiler le noyau comme expliqué précédement dans ce HOWTO

NB : Si vous pouvez nous aider à remplir ce tableau, envoyez SVP un email à *ambrose@writeme.com* ou à *dranch@trinnet.net* .

- Caldera < v1.2 : NON - ?
- Caldera v1.3 : OUI - 2.0.35 based
- Caldera v2.2 : OUI - 2.2.5 based
- Caldera eServer v2.3 : OUI - ? based
- Debian v1.3 : NON - ?
- Debian v2.0 : NON - ?

- Debian v2.1 : OUI - 2.2.1 based
- Debian v2.2 : OUI - 2.2.15 based
- DLX Linux v? : ? - ?
- DOS Linux v? : ? - ?
- FloppyFW v1.0.2 : ? - ?
- Hal91 Linux v? : ? - ?
- Linux Mandrake v5.3 : OUI - ?
- Linux Mandrake v6.0 : OUI - 2.2.5 based
- Linux PPC vR4 : NON - ?
- Linux Pro v? : ? - ?
- LinuxWare v? : ? - ?
- Mandrake v6.0 : OUI - ?
- Mandrake v6.1 : OUI - ?
- Mandrake v7.0 : OUI - 2.2.14
- Mandrake v7.1 : OUI - 2.2.15
- Mandrake v7.2 : OUI - 2.2.17
- MkLinux v? : ? - ?
- MuLinux v3rl : OUI - ?
- Redhat < v4.x : NON - ?
- Redhat v5.0 : OUI - ?
- Redhat v5.1 : OUI - 2.0.34 based
- Redhat v5.2 : OUI - 2.0.36 based
- Redhat v6.0 : OUI - 2.2.5 based
- Redhat v6.1 : OUI - 2.2.12 based
- Redhat v6.2 : OUI - 2.2.14 based
- Redhat v7.0 : OUI - 2.2.16 based
- Slackware v3.0 : ? - ?
- Slackware v3.1 : ? - ?
- Slackware v3.2 : ? - ?
- Slackware v3.3 : ? - 2.0.34 based
- Slackware v3.4 : ? - ?
- Slackware v3.5 : ? - ?

- Slackware v3.6 : ? - ?
- Slackware v3.9 : ? - 2.0.37pre10 based
- Slackware v4.0 : ? - ?
- Slackware v7.0 : OUI - 2.2.13 based
- Slackware v7.1 : OUI - 2.2.16 based
- Stampede Linux v? : ? - ?
- SuSE v5.2 : OUI - 2.0.32 base
- SuSE v5.3 : OUI - ?
- SuSE v6.0 : OUI - 2.0.36 based
- SuSE v6.1 : OUI - 2.2.5 based
- SuSE v6.3 : OUI - 2.2.13 based
- Tomsrbt Linux v? : ? - ?
- TurboLinux Lite v4.0 : OUI - ?
- TurboLinux v6.0 : OUI - 2.2.12 based
- TriLinux v? : ? - ?
- Yggdrasil Linux v? : ? - ?

## 7.2 Quelles sont la configuration matérielle minimale requise et les limitations d'IP Masquerade? Les performances sont-elles bonnes ?

Un 486/66 avec 16MO de RAM été bien plus que suffisant pour remplir les 1.54Mb/s d'une connexion T1 à 100% ! MASQ a aussi déjà bien tourné sur des 386SX-16 avec 8MO de RAM. Vous devez cependant noter que Linux IP Masquerade commence à faire des dechets avec plus de 500 entrées MASQ.

La seule application que je connaisse qui puisse temporairement casser Linux IP Masquerade est GameSpy. Pourquoi ? Quand il actualise ses listes, il crée des dizaines de milliers de connexions rapides pendant une TRES courte période. Jusqu'au timeout de ces sessions, les tables de MASQ sont pleines ("FULL"). Reportez vous à la section 7.20 () pour de plus amples détails.

Pendant qu'on y est :

Il y a une limite the 4096 connexions concurrentes chacune pour TCP & UDP. Cette limite peut être changée en bidouillant les valeurs de */usr/src/linux/net/ipv4/ip_masq.h*, une limite supérieure de 32 000 devrait être OK. Si vous voulez changer cette limite, vous devez changer les valeurs de PORT_MASQ_BEGIN & PORT_MASQ_END pour avoir une taille correctemment dimensionnée au dessus de 32K et en dessous de 64K.

## 7.3 Quand je lance la commande rc.firewall, je reçois des erreurs "command not found". Pourquoi ?

Comment avez vous mis le fichier rc.firewall sur votre machine ? L'avez vous copié-collé dans une fenêtre de TELNET, ou envoyé par FTP à partir d'une machine Windows/DOS etc. ? Essayez ça ... logguez vous sur la machine Linux et lancer "vim -b /etc/rc.d/rc.firewall" et regarder si vos lignes finissent pas ^M. Si oui, effacer les ^M et essayer de nouveau.

## 7.4 J'ai vérifié toutes mes configurations, et j'arrive toujours pas à faire fonctionner IP Masquerade. Que dois-je faire ?

- Restez calme. Prenez une tasse de thé, de café, de boisson etc. et reposez vous. Une fois que votre esprit est reposé, essayez les suggestions ci-dessous. Mettre en place Linux IP Masquerading n'est PAS difficile mais il a plusieurs concepts que vous devez comprendre.
- Encore une fois, suivez les étapes de la section 5 (). 99% des utilisateurs qui utilisent Masquerade pour la première fois ne sont même pas allé regarder là-bas.
- Verifier les *IP Masquerade Mailing List Archives* `http://www.indyramp.com/lists/masq/` , sans doute que votre problème est courant et que vous pourrez trouver une solution avec une simple recherche dans l'archive.
- Vérifiez aussi le document *TrinityOS* `http://www.ecst.csuchico.edu/~dranch/LINUX/index-linux.html#TrinityOS` . Il couvre l'IP Masquerading pour les noyaux 2.0.x et 2.2.x et beaucoup d'autres sujets dont PPPd, DialD, DHCP, DNS, Sendmail, etc.
- Vérifiez bien que vous n'êtes pas ROUTED ou GATES. Pour ce faire, lancez "ps aux | grep -e routed -e gated".
- Envoyez votre question sur l'IP Masquerade Mailing List (regardez la suite de la section FAQ pour les détails). Utilisez cette solution seulement si vous ne pouvez pas trouver de solution dans l'IP Masquerading Archive. Vérifiez bien que vous envoyez toutes les informations demandées dans la section 5 () dans votre email!!
- Postez votre question dans un newsgroup NNTP Linux correspondant.
- Envoyez un email à *ambrose@writeme.com* et *dranch@trinnet.net* . Vous avez plus de chance de recevoir une réponse de la mailing list IP Masquerading que de l'un d'entre nous.
- Verifiez vos configurations de nouveau :-)

## 7.5 Comment puis-je m'inscrire ou consulter les mailing lists d'IP Masquerade et/ou IP Masqurade Developers et les archives ?

Il y a deux manières de s'inscrire aux deux mailing lists de Linux IP Masquerading. La première façon est d'envoyer un email à *masq-request@indyramp.com* . Pour s'inscrire à la mailing list de Linux IP Masquerading Developers, envoyez un email à *masq-dev-request@indyramp.com* . Reportez vous SVP à la boulette si dessous pour plus de détails.

- S'abonner par email: Mettez "subscribe" soit dans le champs 'subject' soit dans le corps du message de l'email. Si vous voulez vous abonner à la version Digest (tous les emails de la liste donnée vous sont envoyés dans un seul gros email), de l'une des mailing list (la principale MASQ ou la liste MASQ-DEV), mettez les mots "subscribe digest" à la place, soit dans le champs 'subject' soit dans le corps de l'email. Une fois que le serveur reçoit votre requête, il va vous abonner à la liste que vous avez demandé et vous fournir un MOT DE PASSE. Sauvegardez ce mot de passe parce que vous en aurez besoin plus tard, pour terminer votre abonnement ou changer vos options

La seconde méthode et d'utiliser un browser WWW et de vous inscrire via le formulaire qui se trouve à l'URL *http://www.indyramp.com/masq-list/* pour la liste principale MASQ ou *http://www.indyramp.com/masq-dev-list/* pour la liste MASQ-DEV.

Une fois abonné, vous recevrez des emails de la liste à laquelle vous vous êtes abonné. Notez aussi que les utilisateurs abonnés et NON-abonnés peuvent acceder aux archives des deux listes. Pour ce faire, veuillez SVP vous reporter aux URLs WWW si dessus pour plus de détails.

Enfin, veuillez noter que vous pouvez envoyer des emails à la liste MASQ uniquement à partir de votre compte/adresse avec lesquels vous vous êtes abonné.

Si vous avez des problèmes avec les mailing lists, ou l'archive de la mailing liste, veuillez contacter SVP *Robert Novak* .

## 7.6 En quoi IP Masquerade est différent des Proxy ou des services NAT ?

```
Proxy: les serveurs Proxy sont disponibles pour : Win95, NT, Linux, Solaris, etc.

                Avantages:      + (1) seule adresse IP; pas cher
                                + Peu optionnellement utiliser une cache pour de meilleurs
                                  performances (WWW, etc.)

                Inconvenients:  - Toutes les applications derriere un serveur
                                          proxy doivent etre COMPATIBLES avec les
                                          services proxy (SOCKS) et etre CONFIGURES
                                          pour utiliser le serveur Proxy
                                        - Fout en l'air les compteurs WWW et les stastiques WWW

         Un serveur proxy utilise seulement (1) une adresse IP publique, comme IP MASQ, et se
         comporte comme un traducteur vers les clients du LAN prive (browser WWW, etc.)
         Ce serveur proxy recoit les requetes tels que TELNET, FTP, WWW, etc. du reseau prive
         sur une interface. Il va ensuite initialiser ces requetes comme si c'etait quelqu'un sur
         la machine elle-meme qui les faisait. Une fois que le serveur distant sur Internet renvoie
         les informations demandees, il va retraduire les adresses TCP/IP vers les machines
         internes et envoyer le traffic vers la machine interne qui avait fait la demande. C'est
         la raison pour laquelle il est appele serveur PROXY.

                NB :   TOUTE application que vous pourriez vouloir utiliser sur les
                       machines internes *DOIT* avoir la compatibilite avec les serveurs
                       proxy, comme Netscape et quelques applications parmi les meilleurs
                       clients TELNET et FTP. Tout client qui n'est pas compatible avec
                       les serveurs proxy ne fonctionnera pas.

         Une autre chose bien a propos des serveurs proxy est que quelques uns d'entre eux
         peuvent aussi servir a faire du cache (antememoire) (Squid pour WWW). Imaginez alors
         vous avez 50 machines 'proxies' qui vont charger Nestcape en meme temps. S'ils sont
         que installes avec la page de garde par defaut, vous allez avoir 50 copies de la
         Netscape qui vont arriver a travers le WAN pour chaque ordinateur. Avec un proxy a
         meme page Web antememoire, seule une copie serait telechargee par le serveur proxy et ensui
         les machines proxies recevraient la page a partir de l'antememoire. Cela va non
         seulement economiser de la bande passante sur la connexion Internet, mais en plus ca va etr
         BEAUCOUP BEAUCOUP plus rapide pour les machines internes proxies.
```

```
MASQ:    IP Masq est disponible sur Linux et quelques routeurs ISDN tels que
 ou      le Zytel Prestige128, Cisco 770, les routeurs ISDN NetGear, etc.
1:Many
 NAT
                Avantages:      + (1) seule adresse IP; pas cher
                                + N'a pas besoin de compatibilite speciale des applications
                                + Utilise un firewall logiciel donc votre reseau peu devenir
                                plus sur

                Inconvenients:  - Requiert une machine Linux ou un routeur ISDN special
                                (meme si d'autres produits pourraient l'avoir...)
                                - Le traffic entrant ne peut acceder au LAN interne sans que
                                le LAN interne soit initiateur du traffic ou qu'il y ait un
                                logiciel specifique pour le port forwarding d'installe.
                                Beaucoup de serveurs NAT NE PEUVENT PAS fournir cette
                                fonctionnalite
                                - Des protocoles speciaux doivent etre traites de maniere speciale
                                par les redirecteurs de firewall, etc. Linux est completement
                                compatible avec ceux-ci (FTP, IRC, etc.) mais beaucoup de routeurs
                                NE le SONT PAS (NetGear DOES).


         Masq ou 1:Many NAT est similaire a un serveur proxy parce que le serveur va faire une
         translation d'adresse IP et faire croire au serveur distant (par exemple le serveur WWW)
         que c'est le serveur MASQ qui a fait la requete et non la machine interne.

         Une difference majeure entre un serveur MASQ et un serveur PROXY est que les serveurs
         MASQ n'ont pas du tout besoin de changement de configuration des machines clientes.
         Il suffit de les configurer pour qu'elles utilisent la machine linux en tant que leur
         passerelle par defaut et tout fonctionne correctement. Vous AUREZ besoin d'installer
          des modules linux speciaux pour faire fonctionner des trucs genre RealAudio, FTP, etc. !

         De plus, de nombreuses personnes utilisent IP MASQ pour le TELNET? FTP, etc. *ET*
         mettent en place un serveur proxy avec cache sur la meme machine Linux pour le traffic
         WWW afin d'obtenir de meilleurs performances.


NAT:     des serveurs NAT sont disponibles sur Windows 95/NT, Linux, Solaris, et quelques
         un des meilleurs routeurs ISDN (pas chez Ascend)

                Avantages:      + Tres configurables
                                + Pas de logiciel special requis

                Inconvenients:  - Necessite un sous-reseau de votre FAI (cher)

         Translation d'Adresse Reseau est le nom d'une boite qui aurait un groupe d'adresses
         IP valides qu'il peut utiliser sur l'interface Internet. Quand sur le reseau interne,
         une machine veut acceder a Internet, il associe une des adresses IP VALIDES disponibles
         de son interface Internet a l'adresse IP PRIVEE qui a fait la demande. Ensuite, tout
         le traffic est retranscrit de l'adresse IP public du NAT vers son adresse IP privee.
```

```
Lorsque l'adresse NAT PUBLIQUE devient inactif pour une certaine periode predeterminee,
l'adresse IP PUBLIQUE est rangee de nouveau dans le groupe d'adresses NAT publiques

Le principal probleme de NAT est que, une fois que toutes les adresses IP publiques
disponibles sont utilisees, tout utilisateur prive qui demande un service Internet doit
attendre qu'une adresse publique NAT se libere.
```

Pour une description très bien faite et très complète des différentes formes de NAT, veuillez SVP vous reporter à :

- http://www.suse.de/~mha/linux-ip-nat/diplom/nat.html

Voici un autre bon site pour apprendre des choses sur NAT, bien que beaucoup d'URLs sont anciennes, elles sont toujours valables :

- http://www.linas.org/linux/load.html

Voici un très bon URL pour apprendre des choses sur les autres solutions NAT pour Linux mais aussi pour les autres plateformes :

- http://www.uq.net.au/~zzdmacka/the-nat-page/

## 7.7 Existe-t-il des outils de création/gestion de firewall avec interface graphique ?

Oui ! Ils ont différentes interfaces, complexités, etc. mais ils sont très bien bienque la plupart soit exclusivement pour l'outil IPFWADM. Voici une courte liste des outils disponibles, dans l'ordre alphabétique. Si vous en connaissez d'autres ou vous savez lesquels sont bien/mauvais/immondes, envoyez SVP un email à David

- Le *IPFWADM Dot file generator* de John Hardin - Une version IPCHAINS est en cours de developpement.
- *fBuilder* http://www.innertek.com de Sonny Parlin : De l'auteur de FWCONFIG, cette nouvelle solution est entièrement basée sur le WWW et offre des options de redondance, etc. pour IPCHAINS et NetFilter.
- *Mason* http://www.pobox.com/~wstearns/mason/ de William Stearns - Un système pour créer des jeux de règles à la volée

## 7.8 IP Masquerade fonctionne-t-il avec des adresses IP alouées dynamiquement ?

**Oui**, ça fonctionne, avec les IP dynamique, assignée par votre FAI via un serveur PPP ou DHCP/BOOTp. Bien sur les IP statiques fonctionnent aussi. Toutefois, si vous voulez implémenter un jeu de règle IPFWADM/IPCHAINS 'strong' ou utiliser un port forwarder, votre jeu de règles devra être réexecuté à chaque fois que votre IP change. Reportez vous SVP au debut de la section *TrinityOS - Section 10* http://www.ecst.csuchico.edu/~dranch/LINUX/index-linux.html#TrinityOS pour plus d'aide sur les jeux de règles 'strong' du firewall et les adressses IP dynamiques.

## 7.9 Puis-je utiliser un modem par cable (soit bidirectionnel, soit avec un modem pour le retour), une connexion DSL, un lien satellite, etc. pour me connecter à internet et utiliser IP Masquerade ?

**OUI**, tant que Linux est compatible avec l'interface réseau, ça devrait fonctionner. Si vous recevez une adresse IP dynamique, reportez vous SVP à l'URL de la partie "IP Masquerade fonctionne-t-il avec des adresses IP alouées dynamiquement" dans l'article de la FAQ si dessus.

## 7.10 Puis-je utiliser Diald ou la fonction Dial-on-Demand de PPPd avec IP MASQ?

Bien sûr ! IP Masquerading est totalement transparent pour Diald ou PPP. La seule chose qui pourrait poser problème est l'utilisation d'un jeu de règle 'strong' avec un adresse IP dynamique. Reportez vous à l'article de la FAQ " IP Masquerade fonctionne-t-il avec des adresses IP alouées dynamiquement" ci-dessus pour de plus amples détails.

## 7.11 Quels applications sont compatibles avec IP Masquerade?

C'est difficile de garder une liste de toutes les "applications qui fonctionnent". Cependant, la plupart des applications Internet classiques sont compatibles (les browser WWW (Netscape, MSIE, etc. FTP (tels que WS_FTP), TELNET, SSH, RealAudio, POP3 (email entrant - Pine, Eudora, Outlook), SMTP (email sortant), etc.) Une liste assez complète de clients compatibles MASQ peut être trouvé à la section 6.3 () de ce HOWTO.

Les applications impliquant des protocoles plus compliqués ou des méthodes de connexion spéciales telles que la video conferencing ont besoin d'outils d'aide spéciaux.

Pour de plus amples détails, veuillez vous reporter SVP à la page *Linux IP masquerading Applications* `http://www.tsmservices.com/masq`

## 7.12 Comment puis-je faire fonctionner IP Masquerade sur Redhat, Debian, Slackware, etc.?

Peu importe quelle distribution Linux vous utilisez, les procédures pour configurer IP Masquerade mentionnées dans ce HOWTO devraient fonctionner. Quelques distributions peuvent avoir une interface graphique ou des fichiers de configurations speciaux qui peuvent rendre la configuration plus simple. Nous faisons le mieux que nous pouvons pour écrire ce HOWTO dans le cas le plus général possible.

## 7.13 Les connexions TELNET semblent s'interrompre si je ne les utilise pas souvent. Pourquoi ça ?

Par défaut, IP Masq, règles ses timers pour les sessions TCP, TCP FIN, et les traffics UDP à 15 minutes. Il est recommandé d'utiliser les réglages suivant (comme indiqué précédemment dans ce HOWTO au niveau du jeu de règles /etc/rc.d/rc.firewall ) pour la plupart des utilisateurs :

Linux 2.0.x avec IPFWADM:

```
#timeouts de MASQ
#
#   timeout de 2heures pour les sessions TCP
```

```
#   timeout de 10secondes pour le traffic avoir recu le paquet TCP/IP "FIN"
#   timeout de 60secondes pour le traffic UDP (les utilisateurs d'ICQ MASQ'ues
#   doivent active un timeout firewall de 30 secondes dans ICQ lui-meme)
#
/sbin/ipfwadm -M -s 7200 10 60
```

Linux 2.2.x avec IPCHAINS:

```
#timeouts de MASQ
#
#   timeout de 2heures pour les sessions TCP
#   timeout de 10secondes pour le traffic avoir recu le paquet TCP/IP "FIN"
#   timeout de 60secondes pour le traffic UDP (les utilisateurs d'ICQ MASQ'ues
#   doivent active un timeout firewall de 30 secondes dans ICQ lui-meme)
#
/ipchains -M -S 7200 10 60
```

## 7.14 Quand je me connecte une première fois à Internet, rien de fonctionne. Si j'essaie de nouveau, tout fonctionne correctement. Pourquoi ?

La raison est que vous avez une IP dynamique et que quand vous vous connectez à Internet, IP Masquerade ne connait pas votre IP. Il y a une solution à ce problème. Dans votre jeu de règles /etc/rc.d/rc.firewall, ajoutez ceci :

```
# Utilisateur d'IP dynamique :
#
#   Si vous recevez votre IP dynamiquement par SLIP, PPP ou DHCP, activez cette option
#   Ceci permet le hacking des IP dynamiques dans IP MASQ, rendant la vie
#   avec Diald et les programmes similaires plus simple.
#
echo "1" > /proc/sys/net/ipv4/ip_dynaddr
```

## 7.15 ( MTU ) - IP MASQ semble fonctionner correctement mais certain sites ne fonctionnent pas. D'habitude, ça arrive avec le FTP et le WWW.

Il y a deux raisons possibles à ce problème. La première est très COURANTE, et la seconde est très RARE

- Il y a un BUG discutable dans le code de Masquerade des noyaux 2.0.38 2.2.9+. Quelques utilisateurs pointent du doigt le fait que IPMASQ pourrait avoir des problèmes avec les paquets qui ont un le bit DF ou "Don't Fragment" activé. En gros, quand une machine Linux se connecte à Internet avec un MTU inférieur à 1500, certains paquets vont avoir le flag DF activé. Bienque changer le MTU 1500 sur le serveur linux va sembler résoudre le problème, le bug probable va toujours être là. On croit qu'il se passe la chose suivante : le code MASQ ne réécrit pas correctement les paquets de retour ICMP avec le code ICMP 3 Sub 4 à l'ordinateur MASQué qui est à l'origine du flux. En raison de cela, les paquets sont éliminés.

  D'autres utilisateurs pointent du doigt les administrateurs des sites distants qui posent problème (typiquement les sites utilisant le SSL, etc.) et disent qu'en raison du filtrage de TOUTES LES FORMES de messages ICMP (dont les Type4 - Fragmentation Needed) en cause de la paranoia de la sécurité, ils fracturent les aspects fondamentaux du protocole TCP/IP.

Les deux arguments ont des aspects valides et les partisans des deux camps continuent à débattre sur ce sujet chaque jour. Si vous êtes un programmeur réseau et que vous pensez pouvoir soit résoudre ce problème soit deviner son origine... ESSAYEZ SVP ! Pour de plus amples détails reportez vous à la liste suivante *MTU Thread from the Linux-Kernel* .

Pas d'inquiétudes toutefois. Une manière efficace à 100% de résoudre le problème est de changer le MTU de votre liaison Internet à 1500. Maintenant quelques utilisateurs vont grogner contre ceci parce que ca peut nuire à la latence de quelques programmes spécifiques tels que TELNET ou les jeux mais son impact n'est que très faible. D'un autre côté, la plupart des connexions HTTP et FTP vont s'ACCELERER !

[ – Si vous avez une connexion PPPoE pour votre DSL/Cablemodem ou si vous décidez de ne pas changer votre MTU à 1500, regardez si dessous pour une autre solution. – ]

Pour réparer ca, regardez d'abord le MTU de votre connexion Internet. Pour ce faire, lancer /bin/ifconfig". Maintenant regardez les lignes correspondants à votre connexion Internet et chercher le MTU. Il FAUT le fixer à 1500. Généralement, les connexions Ethernet vont l'avoir par défaut mais les lignes série PPP vont avoir par défaut 576.

### 7.15.1 Changer le MTU d'une ligne PPP :

- Pour régler le problème de MTU sur une ligne PPP, modifiez votre fichier /etc/ppp/options, et vers le début, ajoutez le texte suivant sur deux lignes séparées : "mtu 1500" et "mru 1500". Sauvegardez les changements et redemarrez PPP. Maintenant vérifiez comme indiqué précédemment que le lien PPP a un MTU et un MRU corrects.
- Pour régler le problème de MTU sur une ligne Ethernet vers votre connexion DSL ou CableModem relié par un routeur ou un pont, vous devez modifier le bon script de démarrage réseau pour votre distribution Linux. Reportez vous SVP au document *TrinityOS - Section 16* pour l'optimisation du réseau.

### 7.15.2 Anciennes interfaces series UNIX :

- Enfin, bienque ceci n'est pas un problème courant, quelques personnes ont trouvé la solution suivante. Les utilisateurs de PPP, vérifiez sur quel port votre PPPd se connecte. Est-ce un port /dev/cua* ou un port /dev/ttyS* ? Le type cua est un ANCIEN et il perturbe certains trucs de manières étranges.

### 7.15.3 Utilisateurs de PPPoE :

Pour les utilisateurs de PPPoE (qui a un MTU maximal de 1490) ou pour ceux qui décident de ne pas utiliser un MTU de 1500, tout n'est pas perdu. Si vous reconfigurez TOUTES les machines MASQuées de façon à ce qu'elles utilisent le MEME MTU que celui de votre connexion externe à Internet, tout devrait fonctionner correctement. A noter cependant que certains FAI par PPPoE peuvent exiger un MTU de 1460 pour une connectivité correcte.

Comment vous pouvez faire cela ? Suivez ces quelques étapes pour votre système d'exploitation.

L'exemple suivant montre la configuration d'un MTU de 1490 pour une connexion PPPoE utilisée par certains utilisateurs de DSL ou de Cablemodems. Il est recommandé d'utiliser les valeurs les PLUS HAUTES possibles pour toutes les connexions dont le débit est supérieur ou égal à 128kb/s.

La seule raison d'utiliser un MTU plus petit est la latence au depend du débit. Reportez vous SVP à :

*http://www.ecst.csuchico.edu/˜dranch/PPP/ppp-performance.html#mtu*

pour de plus amples détails sur ce sujet.

*** Si vous avez REUSSI, ECHOUE, ou que vous avez la procédure à suivre pour d'autres systèmes d'exploitations, *** envoyer SVP une email à DAVID Ranch. Merci !

#### 7.15.4 Linux:

```
1. Le reglage du MTU peut changer d'une distribution a l'autre.

   Pour Redhat : Vous avez besoin de modifier les differentes declarations "ifconfig'
                 dans /sbin/ifup script

   Pour Slackware : Vous avez besoin de modifier les differentes declarations "ifconfig'
                    dans /etc/rc.d/rc1.inet

2. Voici un bon exemple qui marche avec toutes les distributions, modifiez le fichier
   /etc/rc.d/rc.local et mettez ceci a la FIN du fichier :

        echo "Changement du MTU de l'interface ETH0"
        /sbin/ifconfig eth0 mtu 1490

     Remplacez "eth0" par le nom de l'interface de votre machine qui est connectee a Internet.

3. Pour les options avancees telles que "TCP Receive Windows", des exemples detailles
   sur la maniere de modifier les scripts de reseau sur les differentes distributions
   Linux, etc. reportez vous SVP au Chapitre 16 de
   http://www.ecst.csuchico.edu/~dranch/LINUX/index-linux.html#trinityos
```

#### 7.15.5 MS Windows 95:

```
1. TOUT changement dans la base de registre est risquee mais avec une copie de sauvegarde,
   vous devriez etre en securite. Continuez en CONNAISSANCE DE CAUSE.

2. Allez dans Start-->Run-->RegEdit

3. Vous devriez faire une copie de sauvegarde de votre Base De Registre avant de faire quoi
   que ca soit.
   Pour ce faire, copiez les fichiers "user.dat" et "system.dat" du repertoire \WINDOWS
   et mettez les en lieu sur. Notez aussi, que la methode mentionnee
   precedemment utilisant "Regedit: Registry-->Export Registry File-->Save a copy of
   your registry" ne fait que de la FUSION de la Base de Registre et NON PAS son remplacement.

4. Cherchez dans chaque cle de la base de registre qui fini par "n" (c-a-d 0007)
   qui a une entree appelee "IPAddress" qui a votre adresse IP.
   Sous cette cle, ajoutez le texte suivant :

   Tire de http://support.microsoft.com/support/kb/articles/q158/4/74.asp

     [Hkey_Local_Machine\System\CurrentControlset\Services\Class\NetTrans\000n]

         type=DWORD
         name="MaxMTU"          (NE PAS inclure les guillements)
```

```
      value=1490 (Decimal)    (NE PAS inclure le texte "(Decimal)")

      type=DWORD
      name="MaxMSS"           (NE PAS inclure les guillements)
      value=1450 (Decimal)    (NE PAS inclure le texte "(Decimal)")


5. Vous pouvez aussi changer le "TCP Receive Window" qui augmente parfois
   les performances reseau CONSIDERABLEMENT. Si vous remarquez que votre debit
   a DIMINUE, REMETTEZ les anciens reglages et redemarrez.

     [HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\VxD\MSTCP]
        type=DWORD
        name="DefaultRcvWindow"   (NE PAS inclure les guillements)
        value=32768 (Decimal)     (NE PAS inclure le texte "(Decimal)")

        type=DWORD
        name="DefaultTTL"         (NE PAS inclure les guillements)
        value=128 (Decimal)       (NE PAS inclure le texte "(Decimal)")


6. Rebootez pour que les changements soit pris en compte.
```

---

### 7.15.6 MS Windows 98:

```
1. TOUT changement dans la base de registre est risquee mais avec une copie de sauvegarde,
   vous devriez etre en securite. Continuez en CONNAISSANCE DE CAUSE.

2. Allez dans Start-->Run-->RegEdit

3. Vous devriez faire une copie de sauvegarde de votre Base De Registre avant de faire quoi que
   ca soit. Pour ce faire, copiez les fichiers "user.dat" et "system.dat" du repertoire
   \WINDOWS et mettez les en lieu sur. Notez aussi, que la methode mentionnee
   precedemment utilisant "Regedit: Registry-->Export Registry File-->Save a copy of your
   registry" ne fait que de la FUSION de la Base de Registre et NON PAS son remplacement.

4. Cherchez dans chaque cle de la base de registre qui fini par "n" (c-a-d 0007)
   qui a une entree appelee "IPAddress" qui a votre adresse IP. Sous cette cle,
   ajoutez le texte suivant :

   Tire de http://support.microsoft.com/support/kb/articles/q158/4/74.asp

     [Hkey_Local_Machine\System\CurrentControlset\Services\Class\NetTrans\000n]

       type=STRING
       name="MaxMTU"            (NE PAS inclure les guillements)
       value=1490 (Decimal)     (NE PAS inclure le texte "(Decimal)")


5. Vous pouvez aussi changer le "TCP Receive Window" qui augmente parfois
```

```
   les performances reseau CONSIDERABLEMENT. Si vous remarquez que votre debit
   a DIMINUE, REMETTEZ les anciens reglages et redemarrez.

     [HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\VxD\MSTCP]
        type=DWORD
        name="DefaultRcvWindow"   (NE PAS inclure les guillements)
        value=32768 (Decimal)     (NE PAS inclure le texte "(Decimal)")

        type=DWORD
        name="DefaultTTL"         (NE PAS inclure les guillements)
        value=128 (Decimal)       (NE PAS inclure le texte "(Decimal)")


6. Rebootez pour que les changements soit pris en compte.
```

---

### 7.15.7 MS Windows NT 4.x

```
1. TOUT changement dans la base de registre est risquee mais avec une copie de sauvegarde,
   vous devriez etre en securite. Continuez en CONNAISSANCE DE CAUSE.

2. Allez dans Start-->Run-->RegEdit


3. Registry-->Export Registry File-->Sauvegardez une copie de votre base de registre dans un
   endroit sur

4. Creer les cles suivantes dans les deux Bases de Registre possible.
   Des entrees multiples correspondent a differentes connexions reseaux tels
   que PPP, Ethernet NICs, VPNs PPTP, etc.

   http://support.microsoft.com/support/kb/articles/Q102/9/73.asp?LN=EN-US&SD=gn&FR=0


   [HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Parameters\Tcpip]
                    and
   [HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\<Adapter-name>\Parameters\Tcpip]

      Remplacez "<Adapter-Name>" par le nom du lien de l'interface de votre LAN connectee
      a Internet

         type=DWORD
         name="MTU"                (NE PAS inclure les guillements)
         value=1490 (Decimal)      (NE PAS inclure le texte "(Decimal)")

       (NE PAS inclure les guillements)


 *** Si vous savez aussi comment changer MSS, la taille de la fenetre TCP, et les
 *** parametres TTL dans NT 4.x, envoyez SVP un  email a dranch@trinnet.net parce que
 *** j'adorerais les ajouter a ce HOWTO.
```

```
5. Rebootez pour que les changements soit pris en compte.
```

#### 7.15.8 MS Windows 2000

```
1. TOUT changement dans la base de registre est risquee mais avec une copie de sauvegarde,
   vous devriez etre en securite. Continuez en CONNAISSANCE DE CAUSE.

2. Allez dans Start-->Run-->RegEdit


3. Registry-->Export Registry File-->Sauvegardez une copie de votre base de registre
   dans un endroit sur

4. Naviguez jusqu'a la cle :

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\<ID for Adapter>

   Chaque ID Adapter a une cle par default pour le DNS, l'adresse TCP/IP, la passerelle
   par defaut, le masque de sous reseau, etc. Trouvez la cle qui correspond a
   votre carte reseau.

5.  Creez l'entree suivante :

      type=DWORD
      name="MTU"                                (NE PAS inclure les guillements)
      value=1490 (Decimal)      (NE PAS inclure le texte "(Decimal)")

http://support.microsoft.com/support/kb/articles/Q120/6/42.asp?LN=EN-US&SD=gn&FR=0

 *** Si vous savez aussi comment changer MSS, la taille de la fenetre TCP, et les
 *** parametres TTL dans NT 4.x, envoyez SVP un  email a dranch@trinnet.net parce que
 *** j'adorerais les ajouter a ce HOWTO.

5. Rebootez pour que les changements soit pris en compte.
```

Comme déclaré si dessus, si vous savez comment effectuer ces changement pour d'autres OS tels que OS/2, MacOS, etc. envoyez SVP un email *David Ranch* pour qu'ils puissent être inclus dans ce HOWTO.

## 7.16 les clients FTP MASQués ne fonctionnent pas.

Vérifiez si le module "ip_masq_ftp" est chargé. Pour ce faire, logguez vous sur le serveur MASQ et tapez la commande "/sbin/lsmod". Si vous ne voyez pas "ip_masq_ftp" chargé, verifiez que vous avez bien suivi les recommandations du /etc/rc.d/rc.firewall BASIQUE que vous trouverez à la section 3.2 (). Si vous implémentez votre propre jeu de règles, faites en sorte d'inclure la plupart des exemples de ce HOWTO ou vous aurez encore de nombreux problèmes.

## 7.17 l'IP Masquerading semble lent

Il peut y avoir deux raisons à cela :

- Peut être attendez vous plus de votre ligne modem que ce qui est possible. Faisons le calcul pour une connexion par modem 56k typique :

  1. modem 56k = 56,000 bits par seconde.
  2. Vous n'avez PAS vraiment un modem 56k mais un modem 52k à cause des limitations US FCC (NDT : cette restriction est valable uniquement aux Etats-Unis, bien sûr. Mais la suite fourni une méthode de calcul transposable à une veritable ligne 56k).
  3. Vous n'aurez JAMAIS 52k, la meilleure connexion que j'ai réussi à avoir était 48k
  4. 48,000 bits par seconde vaut 4,800 OCTETS par seconde (8 bits pour un octet +2 bits pour les bits START et STOP RS-232 de la laison série)
  5. Avec un MTU de 1500, vous aurez (3.2) paquets en une seconde. Puisque ceci implique la fragmentation, vous aurez besoin de l'arrondir à l'entier INFERIEUR (3) paquets par seconde.
  6. De même, avec un MTU de 1500, il y a 3.2 x 40 octets d'entête TCP/IP (8%)
  7. Donc, le MEILLEUR débit que vous pourrez espérer avoir est de 4.68ko/s sans compression. Avec compression, soit-elle une compression matérielle v.42bis, MNP5, ou MS/Stac, on peut atteindre des nombres impressionnants avec des matériaux hautement compressibles tels que les fichiers TEXT mais en fait elle peut aussi ralentir les choses si on transfets des fichiers pre-compressés tels que des ZIPs, MP3s, etc.

- Connexions reliées par Ethernet (DSL, Cablemodem, LANs, etc.)

  – Verifiez que vous n'avez pas un réseau INTERNE et un réseau EXTERNE tournant sur la même carte réseau avec la fonction "IP Alias". Si c'est ce que vous *FAITES*, ca peut fonctionner mais ca peut être excessivement lent à cause du grand nombres de collisions, de l'utilisation d'IRQ, etc. Il est VIVEMENT RECOMMANDE d'avoir une autre carte réseau pour que les réseaux interne et externe aient leur propre interface. Vérifiez que vous avez les bons réglages Ethernet pour la VITESSE et le DUPLEX.

    ∗ Quelques cartes Ethernet 10Mb/s et la plupart des cartes 100Mb/s sont compatibles avec les connexions FULL Duplex. Les connexions directes de la carte Ethernet vers, disons, un modem DSL (sans hubs entre) *PEU* être réglé sur FULL DUPLEX mais seulement si le modem DSL le permet. Vous devriez aussi vérifier que vos cables Ethernet avec les 8 fils, sont de bonne qualité.

    ∗ Les réseaux internes qui utilisent des HUBs -ne peuvent pas- utiliser le Full Duplex. Vous avez besoin soit d'un *SWITCH* Ethernet 10Mb/s ou 100Mb/s pour pouvoir le faire.

    ∗ Les négociation automatiques sur la vitesse 10/100Mb/s et sur le DUPLEX Full/Half sur les cartes Ethernet peuvent faire des ravages sur un réseau. Je recommande de coder en dur la vitesse et le duplex dans vos cartes réseaux si possible. Vous pouvez le faire directement via les modules Linux NIC mais ce n'est pas possible directement sur les noyaux monolithiques. Vous aurez soit besoin d'utiliser les utilitaires MII de 8.1 () ou bien de coder en dur dans le source du noyau.

- Optimisez votre MTU et réglez la fenêtre glissante de TCP à 8192 au moins

  – Bienque ceci soit COMPLETEMENT en dehors du cadre de ce document, ca aide UN PEU sur TOUTE liaison réseau que vous avez, que ça soit interne ou externe par liaison PPP, Ethernet, TokenRing etc. Pour de plus amples détails, ce sujet est brièvement abordé dans la section 7.14 () si dessus. Pour avoir encore plus de détails, vous pouvez vous reporter à la section Network Optimization de *TrinityOS - Section 16* .

- Utilisation de modems séries avec PPP

- Si vous avez un modem externe, verifiez que vous avez un bon cable serie. De plus, beaucoup de PC on des cables foireux reliant le port série de la carte mère ou la carte d'E/S vers la connexion port série. Si vous êtes dans l'une de ces situations, faites en sorte d'avoir de bonnes conditions. Personnellement, je mets des enrobages de ferrite (ces anneaux de metal gris-noir) autour de TOUTES mes nappes.
- Vérifiez que votre MTU est réglé à 1500 comme décrit précédemment dans la section de la FAQ de ce HOWTO.
- Vérifiez que l'UART de votre port série est au moins à 16550A. Lancez "dmesg | more" pour ce faire.
- Régler l'IRQ-Tune de vos ports séries.
  - Sur la majorité du matériel PC, l'utilisation de l'outil de Craig Estey *IRQTUNE* `http://www.best.com/~cae/irqtune/` augmente de manière significative les performances des ports séries, donc les connexions SLIP and PPP connexions.
- Vérifiez que le port série de votre connexion PPP est au moins à 115200 bauds (ou plus si et votre modem et votre port série peuvent le supporter... aussi connu sous le nom d'adaptateur terminal ISDN)
  - noyaux 2.0.x : les noyaux 2.0.x sont plutôt bizarres parce que vous ne pouvez pas directement leur dire de régler les ports série à 115200. Donc, dans l'un de vos scripts de démarrage, comme les fichiers /etc/rc.d/rc.local ou /etc/rc.d/rc.serial, executez le commande suivante pour un modem sur le port COM2 :
    - setserial /dev/ttyS1 spd_vhi
    - dans votre script PPPd, modifiez la ligne d'execution de pppd de manière à y inclure la vitesse "38400" à l'aide du manuel de pppd.
  - noyaux 2.2.x : contrairement aux noyaux 2.0.x, les noyaux 2.1.x et 2.2.x n'ont pas ce problème de "spd_vhi".
    - Donc, dans le script de PPPd, modifiez la ligne d'execution de pppd de manière à y inclure la vitesse "115200" à l'aide du manuel de pppd.

- Tous types d'interface :

## 7.18 IP Masquerading avec PORTFWing semble s'arrêter quand ma ligne est inactive pendant de longs périodes

Si vous avez une ligne DSL ou par Cablemodem, ce comportement est malheureusement très commun. Ce qui se passe c'est que votre FAI met votre connexion dans une file de priorité très faible pour mieux servir les connexions qui ne sont pas inactives. Le problème est que la connexion de quelques utilisateurs finaux va effectivement être COUPEE jusqu'à ce que le traffic de la connexion de l'utilisateur reveille le matériel du FAI.

- 
- Certaines installations DSL peuvent DECONNECTER une connexion inactive et vérifier l'activité seulement une fois toutes les 30 secondes environ.
- Certains réglages de Cablemodem peuvent mettre une connexion inactive dans une file d'attente de priorité faible et vérifier l'activité seulement une fois toutes les quelques minutes.

Qu'est-ce que je recommande de faire ? Faites un ping vers votre passerelle par défaut toutes les 30 secondes. Pour se faire, modifiez le fichier /etc/rc.d/rc.local et ajoutez la ligne suivante à la fin du fichier :

```
ping -i 30 100.200.212.121 > /dev/null &
```

Remplacez 100.200.212.121 par votre routeur par défaut (routeur en amont).

## 7.19 Maintenant que j'ai l'IP Masquerading qui fonctionne, j'ai plein de sortes de messages d'erreurs et d'avertissements bizarres dans les fichiers log SYSLOG. Comment faut-il lire les erreurs du firewall IPFWADM/IPCHAINS ?

Il y a sans doute deux choses que vous allez couramment voir :

- **MASQ: Failed TCP Checksum error:** Vous verrez cette erreur quand un paquet arrivant d'Internet est corrompu dans la partie donnée mais que le reste "semble" bon. Quand la machine Linux reçcoit le paquet, il va calculer le CRC du paquet et determiner s'il est corrompu ou pas. Sur la plupart des machines tournant sur des OS tels que Microsoft Windows, il vont simplement ignorer le paquet mais Linux IP MASQ le met dans son compte-rendu. Si vous en recevez beaucoup sur une ligne PPP, regardez d'abord l'entrée de la FAQ correspondant à "MASQ est lent".
- Si toutes ces astuces ne vous aident pas, essayer en ajoutant la ligne "-vj" dans votre fichier /etc/ppp/options et relancer PPPd.
- **Hits sur le firewall** : Quand vous êtes sur Internet avec un firewall décent, vous êtes surpris du nombre de personnes qui essayer d'entrer dans votre machine Linux ! Donc que signifient tous ces logs du firewall ? Tiré du document :

  *TrinityOS - Section 10* `http://www.ecst.csuchico.edu/~dranch/LINUX/index-linux.html#TrinityOS` :

```
        Dans le jeu de regles si dessous, chaque ligne avec soit un DENY soit un REJECT
        contient aussi un "-o" pour LOGguer ces hits firewall dans le fichier de messages
        SYSLOG qui se trouve dans :

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

        Si vous regardez l'un de ces logs de firewall, vous devriez voir quelquechose du type :

        ---------------------------------------------------------------------
        IPFWADM:
        Feb 23 07:37:01 Roadrunner kernel: IP fw-in rej eth0 TCP 12.75.147.174:1633 100.200.0.2
        L=44 S=0x00 I=54054 F=0x0040 T=254

        IPCHAINS:
        Packet log: input DENY eth0 PROTO=17 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 dans une seule ligne. Regardons cet exemple,
  reportez vous au hit sur le firewall en lisant ceci. Veuillez noter que cet
  exemple est pour IPFWADM mais il est DIRECTEMENT lisible pour les utilisateurs
  d'IPCHAINS.
```

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

- Ce "hit" firewall est arrive a ce moment : "Feb 23 07:37:01"

- Ce hit etait sur l'ordinateur "RoadRunner".

- Ce hit etait sur le protocole "IP" ou TCP/IP

- Ce hit est entre dans le firewall ("fw-in")
        * D'autres log peuvent dirent "fw-out" pour sortant or "fw-fwd" pour FORWARD

- Ce hit etait encore "rejETE".
        * D'autres logs peuvent dire "deny" ou "accept"

- Ce his etait sur l'interface "eth0" (liaison Internet)

- Ce hit etait un paquet "TCP"

- Ce hit est venu de l'adresse IP "12.75.147.174" et du port "1633".

- Ce hit etait a destination de "100.200.0.212" et de son port "23" ou TELNET.
        * Si vous ne savez pas que le port 23 est pour TELNET, regardez votre fichier
                 /etc/services pour voir par quoi les autres ports sont utilises.

- Ce paquet avait une longueur de "44" octets

- Ce paquet n'avez PAS de "Type of Service" (TOS) fixe
        --Ne vous inquietez pas si vous ne comprenez pas ca...pas besoin de le savoir
        * utilisateurs d'ipchains, divisez le par 4 pour avoir le Type of Service

- Ce paquet avet le numero "IP ID" "18"
        --Ne vous inquietez pas si vous ne comprenez pas ca... pas besoin de le savoir

- Ce paquet avait un fragment offset de 16bits dont tout flag de paquet TCP/IP
  de "0x0000"
--Ne vous inquietez pas si vous ne comprenez pas ca... pas besoin de le savoir
  * une valeur qui commence par "0x2..." ou "0x3..." signifie que le bit
   "More Frangments" etait active donc de nouveaux paquets fragmentes
    vont arriver pour completer ce GROS paquet.
  * une valeur qui commence par "0x4..." ou "0x5..." signifie que le bit
   "Don't Fragment" est active.
  * Toute autre valeur est l'offset du Fragment (divise par 8) qui doit etre
   utilise plus tard pour reconstituer le GROS paquet original.

- Ce paquet a un TimeToLive (TTL) de 20.
  * Chaque saut sur Internet soustrait (1) a ce nombre. Normalement,
  les paquets sont emis avec un TTL de (255) et si ce nombre atteind (0),
  ca signifie qu'il est realiste de considerer que ce paquet est perdu et il
  va donc etre efface.
```

## 7.20 Puis-je configurer IP MASQ de façon à permettre aux Internautes de contacter directement un server interne MASQué ?

Oui ! Avec IPPORTFW, vous pouvez permettre TOUT ou seulement une partie des Internautes de contacter TOUTE machine interne MASQuée. **Ce sujet est entièrement traité dans la section 6.7 () de ce HOWTO.**

## 7.21 Je reçois des "kernel: ip_masq_new(proto=UDP): no free ports." dans mon fichier SYSLOG. Que se passe-t-il ?

Une des machines MASQuée interne crée un nombre anormalement grand de paquets destinés à Internet. Comme le serveur IP Masq construit une table MASQ et forward ces paquets vers Internet, la table se remplit rapidement. Une fois que la table est pleine, elle va générer des erreurs.

La seule application que je connaisse qui puisse temporairement casser Linux IP Masquerade est GameSpy. Pourquoi ? Quand il actualise ses listes, il crée des dizaines de milliers de connexions rapides pendant une TRES courte période. Jusqu'au timeout de ces sessions, les tables de MASQ sont pleines ("FULL"). Reportez vous à la section 7.20 () pour de plus amples détails.

Donc que pouvez-vous faire contre ça ? Pratiquement, n'utilisez pas de programmes qui génèrent ce genre de choses. Si vous recevez ce genre d'erreurs dans vos logs, trouvez le et arrêtez de l'utiliser. Si vous aimez vraiment GameSpy, ne faites pas beaucoup de reactualisation de serveurs. De toute façon, une fois que vous arrêtez le programme MASQué, cette erreur MASQ va disparaître puisque ces connexions vont faire des 'timeouts' dans les tables de MASQ.

## 7.22 Je reçois "ipfwadm: setsockopt failed: Protocol not available" quand j'essaie d'utiliser IPPORTFW!

Si vous recevez le message "ipfwadm: setsockopt failed: Protocol not available", c'est que vous n'êtes pas sous le nouveau noyau. Verifiez que vous l'avez bien installé, relancer votre BootLoader (LILO), et redemarrez.

Reportez vous SVP à la fin de la section 6.7 () pour de plus amples détails.

## 7.23 ( SAMBA ) - Les clients de partage de fichiers et d'imprimantes, et de domaine, de Microsoft ne fonctionnent pas à travers IP Masq ! Pour être correctement compatible avec le protocle SMP de Microsoft, un module IP Masq doit être écrit mais il y a trois moyens viables de le contourner. Pour plus de détails, reportez vous SVP à : *this Microsoft KnowledgeBase article* .

Le premier moyen de contournement est de configurer IPPORTFW de la section 6.7 () et de portforwarder les ports 137, 138 et 139 vers les IP de la machine interne Windows. Bienque cette solution fonctionne, elle ne peut marcher que pour UNE machine interne.

La seconde solution est d'installer et de configurer *Samba* sur le serveur Linux MASQ. Avec Samba de lancé, vous pouvez mapper vos partages de Fichier et d'Imprimantes Windows sur le serveur Samba. Vous pouvez ensuite monter ces nouveaux partages SMB vers tout vos clients externes. La configuration de Samba est entièrement traitée dans un HOWTO que vous pourrez trouver sur le site du Linux Documentation Project et dans le document TrinityOS.

La troisième solution est de configurer un VPN (virtual private network ou réseau privé virtuel) entre les deux machines Windows ou entre les deux réseaux. Ceci peut être réalisé via PPTP ou via les solutions VPN IPSEC. Il y a un patch 7.31 () pour linux et aussi un implémentation complète de IPSEC disponibles pour les deux noyaux 2.0.x et 2.2.x. Cette solution est sans doute la plus stable et la plus sécurisée des trois.

Toutes ces solutions de sont PAS traitées dans ce HOWTO. Je vous recommande de regarder la documentation de TrinityOS pour l'aide sur IPSEC et la page PPTP de John Hardin pour de plus amples informations.

*Veuillez aussi comprendre SVP que le protocole SMB de Microsoft est TRES peu sûr. C'est pour cela que d'avoir des traffics de partage de fichiers ou d'imprimante, et de domaine windows de Microsoft sur Internet sans encryption est une TRES MAUVAISE idée.*

## 7.24 ( IDENT ) - IRC ne fonctionne pas correctement pour les utilisateurs MASQués. Pourquoi?

La raison la plus courante est que les serveurs IDENT ou "Identity" de la plupart des distributions Linux ne peuvent pas travailler avec des liens IP Masqueradés. Pas d'inquiétudes toutefois, il existe des IDENTs qui fonctionnent.

L'installatin de ce logiciel sort du cadre de ce HOWTO mais chaque utilitaire a sa propre documentation. Voici quelques un des URLS :

- *Oident* `http://freshmeat.net/projects/oidentd/homepage/` est le serveur IDENT favoris des utilisateurs de MASQ.
- *Mident* `ftp://ftp.code.org/pub/linux/midentd/` est un serveur IDENT qui a du succès.
- *Sident* `http://insecurity.net/sidentd.gz`
- *Autres Idents* `ftp://sunsite.unc.edu/pub/Linux/system/network/daemons/`

Veuillez noter que certains serveur IRC ne permettront toujours pas des connexions multiples à partir de la même machine (comprendre ici même IP), même s'ils recupèrent les infos Ident et que les utilisateurs sont différents. Vous pouvez vous plaindre à l'administrateur système du serveur distant :-)

## 7.25 ( DCC ) - mIRC ne marche pas avec les DCC Sends

Ceci est un problème de configuration de votre version de mIRC. Pour le résoudre, deconnecter mIRC de votre serveur IRC. Maintenant dans mIRC, allez dans File –> Setup et cliquez sur la languette "IRC servers". Verifiez que le port est réglé sur 6667. Si vous avez besoin d'autres ports, reportez vous ci dessous. Ensuite, allez dans File –> Setup –> Local Info et effacer les champs Local Host et IP Address. Maintenant cochez les cases de "LOCAL HOST" et "IP address" (IP address peut être coché et désactivé). Ensuite, dans "Lookup Method", réglez sur "normal". Ca ne marchera PAS si "server" est sélectionné. C'est tout. Essayez de vous connecter au serveur IRC de nouveau.

Si vous avez besoin de ports pour le serveur IRC différents de 6667, (par exemple 6969) vous devez modifier votre fichier /etc/rc.d/rc.firewall là où vous chargez le module MASQ IRC. Modifiez ce fichier et la ligne contenant "modprobe ip_masq_irc" et ajoutez cette ligne : "ports=6667,6969". Vous pouvez ajoutez autant de ports que vous voulez, séparés par des virgules.

Enfin, fermez tous les clients IRC lancé sur les machines MASQuées et redémarrez le module MASQ IRC :

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

## 7.26 ( IP Aliasing ) - IP Masquerade peut-il fonctionner avec UNE seule carte Ethernet ?

**Oui et non**. Avec la fonctionnalité "IP Alias" du noyau, les utilisateurs peuvent régler plusieurs interfaces aliasées tels que eth0:1, eth0:2, etc mais il N'est PAS recommandé d'utiliser ces interfaces aliasées pour l'IP Masquerading. Pourquoi ? Fournir un firewall sûr devient très difficile avec une seule carte réseau. En plus, vous allez trouver une quantité anormale d'erreurs sur cette liaison puisque les paquets entrant vont être envoyés presque simultanément vers l'extérieur. A cause de tout celà, et du fait qu'une carte réseau coute moins de 150F, je vous recommande vivement d'en acheter une pour chaque segment de réseau MASQué.

Les utilisateurs devraient aussi comprendre que IP Masquerading ne fonctionne que sur des interfaces physiques telles que eth0, eth1, etc. MASQuer une interface aliasée telles que "eth0:1, eth1:1, etc" NE fonctionnera PAS. En d'autres termes, ce qui suit ne fonctionnera 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 voulez toujours utiliser des interfaces aliasées, vous devez activer la fonction "IP Alias" du noyau. Vous devrez recompiler et redemarrer. Une fois sous le nouveau noyau, vous devez configurer Linux de manière à ce qu'il utilise la nouvelle interface (i.e. /dev/eth0:1, etc.). Ensuite, vous pouvez la considerer comme une interface Ethernet normale avec toutefois quelques restrictions comme celui ci-dessus.

## 7.27 ( MULTI-LAN ) - J'ai deux LANs MASQués mais je ne peux pas communiquer de l'un vers l'autre !

Reportez vous SVP à la section 6.5 () pour les détails complets.

## 7.28 ( FACONNAGE ) - Je voudrais être capable de limiter la vitesse de certains types spécifiques de traffic

Ce sujet n'a vraiment rien à voir avec IPMASQ et concerne ce qui touche au façonnage du traffic et de la limitation des taux de Linux. Reportez vous SVP au fichier /usr/src/linux/Documentation/networking/shaper.txt de vos sources locales du noyau pour de plus amples détails.

Vous trouverez aussi plus d'informations sur ce sujet et plusieurs URLs dans la section 2.5 () d'IPROUTE2.

## 7.29 ( COMPATIBILITE ) - J'ai besoin de faire de la comptabilité sur les personnes qui utilisent le réseau

Bienque ca n'est pas grand chose à voir avec IPMASQ, voici quelques idées. Si vous connaissez de meilleures solutions, envoyez SVP un email à l'auteur de ce HOWTO pour qu'il puisse l'inclure dedans.

- Idée #1: Disons que vous voulez logguer TOUT le traffic WWW sortant vers Internet. Vous pouvez créer une règle pour le firewall qui ACCEPTE le traffic sur le PORT 80 avec le bit SYN activé et qui le LOGGUE. Maintenant rappelez vous, ceci peut créer de TRES gros fichiers log.
- Idée #2: Vous pouvez lancer la commande "ipchains -L -M" une fois par seconde et logguer toutes les entrées. Vous pouvez ensuite écrire un programme qui combine toutes ces informations dans un seul fichier plus gros.

## 7.30 ( IPs MULTIPLEs ) - J'ai plusieurs adresses IP EXTERNES que je veux PORTFWer vers plusieurs machines internes. Comment je peux faire ça ?

Vous NE POUVEZ PAS. MASQ est un NAT 1:Many (1 vers plusieurs) et n'est pas le bon outil pour faire ca. Vous cherchez une solution NAT Many:Many qui est une installation NAT traditionnelle. Jetez un coup de d'oeil sur l'entrée 7.27 () de la FAQ pour de plus amples détails sur l'outil IPROUTE2 qui fera ce dont vous avez besoin.

Pour les personnes ici qui compte activer plusieurs adresses IP sur une seule interface réseau avec "IP Alias" et ensuite PORTFWer TOUT les ports (0-65535) et utiliser IPROUTE2 pour entretenir les bonnes correspondances des IP source/destination : ça a été réalisé AVEC SUCCES sur les noyaux 2.0.x et avec moins de réussite sur les noyaux 2.2.x. Sans considération du succès, ce n'est pas la bonne façon de faire ça et ce n'est pas une configuraion MASQ compatible. Jetez un coup d'oeil sur IPROUTE2 SVP... c'est la bonne manière de faire du vrai NAT.

Autre chose à noter aussi :

Si vous avez une connexion DSL ou Cablemdem routée (pas PPPoE), les choses se compliquent un peu plus parce que votre installation n'est pas routée. Pas d'inquiétudes cependant, reportez vous au document "Bridge+Firewall, Linux Bridge+Firewall Mini-HOWTO" sur LDP. Vous y apprendrez à faire reconnaitre à votre machine Linux plusieurs adresses IP sur une seule interface !

## 7.31 J'essaie d'utiliser la commande NETSTAT pour me montrer mes connexions Masqueradées mais ca marche pas

Il peut y avoir un problème avec le programme "netstat" sur les distribs basées sur Linux 2.0.x. Après le redemarrage de Linux, la commande "netstat -M" fonctionne bien mais apres qu'un ordinateur MASQué lance plusieurs fois des traffics ICMP avec succès, tels que ping, traceroute, etc., vous obtiendrez peut-être quelquechose du style :

```
masq_info.c: Internal Error ‘ip_masquerade unknown type’.
```

La manière de détourner ce problème est de lancer la commande "/sbin/ipfwadm -M -l". Vous remarquerez aussi qu'après les timeouts des entrées masquerade ICMP, "netstat" fonctionne de nouveau.

## 7.32 ( VPNs ) - Je voudrais faire fonctionner Microsoft PPTP (tunnels GRE) et/ou les tunnels IPSEC (Linux SWAN) à travers IP MASQ

C'EST possible. Cependant c'est quelque peu hors de la portée de ce document, vous pouvez vous reporter à la page de John Hardin *PPTP Masq* pour tous les détails.

## 7.33 Je veux faire fonctionner le jeu réseau XYZ à travers IP MASQ mais ca fonctionne pas. A l'aide !

D'abord, allez ici : *Steve Grevemeyer's MASQ Applications page* `http://www.tsmservices.com/masq` . Si vous ne trouvez pas de solution là-bas, essayer de patcher le noyau Linux avec le patch *LooseUDP* `ftp://ftp.netcom.com/pub/mu/mumford/loose-udp-2.0.36.patch.gz` de Glenn Lamb, qui est traitée dans la section 6.10 () ci-dessus. Vous pouvez aussi regarder la *NAT Page* `http://www.alumni.caltech.edu/~dank/peer-nat.html` de Dan Kegel pour plus d'informations.

Si vous avez les aptitudes technique pour utiliser "tcpdump" et sniffer votre réseau, essayer de trouver quels protocoles et quels numéros de port votre jeu XYZ utilise. Avec ces informations en main, abonnez vous a la *IP Masq email list* et envoyer vos résultats pour obtenir de l'aide.

## 7.34 IP MASQ fonctionne bien pendant un certain temps puis s'arrête de marcher. Un redémarrage semble résoudre ce problème pour un certain temps. Pourquoi ?

Je parie que vous utilisez IPAUTOFW et/ou vous l'avez compilé dans le noyau hein ?? C'est un problème reconnu de IPAUTOFW. Il est recommandé de NE PAS configurer IPAUTOFW dans le noyau Linux et d'utiliser IPPORTFW à la place. Ceci est traité en detail dans la section 6.7 ().

## 7.35 Les ordinateurs internes MASQués ne peuvent pas envoyer d'email SMTP ou POP-3 !

Bienque ceci ne soit pas un problème dû au Masquerading, beaucoup de personnes l'ont mentionné.

SMTP: Le problème est que vous utilisez probablement votre machine linux comme un serveur de relais SMTP et vous recevez l'erreur suivante :

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

Les versions récentes de Sendmail et d'autres Mail Transfer Agents (MTAs) désactivent le relaying par défaut (c'est une bonne chose). Donc pour résoudre le problème, faites ceci :

- Sendmail: activez le relaying spécifique pour vos machines internes MASQuées en modifiant le fichier /etc/sendmail.cw et en ajoutant le hostname et le nom de domaine de vos machines internes MASQuées. Vous devriez aussi vérifier que votre fichier /etc/hosts a l'adresse IP et le Fully Qualified Domain Name (FQDN) écrit dedans. Une fois que vous avez fait ça, vous devez relancer Sendmail pour qu'il relise ses fichiers de configuration. Ceci est traité dans *TrinityOS - Section 25* `http://www.ecst.csuchico.edu/~dranch/LINUX/index-linux.html#TrinityOS`

POP-3: Certains utilisateurs configurent leur ordinateurs internes MASQués de manière à ce que leurs clients POP-3 se connectent sur un serveur SMTP externe. Bienque que cela soit correct, de nombreux serveurs SMTP vont essayer d'identifier (IDENT) votre connexion sur le port 113. Il est très probable que votre problème vienne du fait que votre politique par défaut pour Masquerade soir DENY (refuse). C'est mal. Changez-le en REJECT (rejette) et relancer votre jeu de règles rc.firewall.

## 7.36 ( IPROUTE2 ) - J'ai besoin que différents réseaux internes MASQués puissent sortir sur différentes adresses IP externes

Disons que vous avez l'installation suivant : Vous avez plusieurs réseaux internes et aussi plusieurs adresses IP externes et/ou réseaux. Ce que vous voulez faire c'est que le LAN#1 n'utilise que l'IP externe IP#1 et vous voulez aussi que le LAN#2 utilise l'IP externe IP#2.

LAN interne ———-> IP officielle

LAN #1 IP externe #1 192.168.1.x –> 123.123.123.11

LAN #2 IP externe #2 192.168.2.x –> 123.123.123.12

En gros, ce que nous avons décrit ici est un routage, PAS seulement sur l'adresse de destination (routage IP usuel) mais aussi un routage basé sur l'adresse SOURCE. Ceci est appelé "routage basé sur une politique" ("policy-based routing") ou "routage par source" ("source routing"). Cette fonctionnalité n'est PAS disponible dans les noyaux 2.0.x, mais l'*EST* pour les noyaux 2.2.x via le package IPROUTE2, et est implémenté dans le nouveau noyau 2.4.x avec IPTABLES.

Vous devez tout d'abord comprendre que IPFWADM et IPCHAINS ne rentrent en action qu'*APRES* que le moment où le système de routage a décidé de l'endroit où il va envoyer un paquet donné. Cet énoncé est très important est devrait être estampé avec de grosses lettres rouges sur toute documentation sur IPFWADM/IPCHAINS/IPMASQ. C'est pourquoi les utilisateurs DOIVENT installer leur routage d'abord et commencer à ajouter IPFWADM/IPCHAINS et/ou des fonctions Masq.

Dans l'exemple précédent, vous devez dire au système de routage de diriger les paquets en provenance de 192.168.1.x via 123.123.1233.11 et les paquets en provenance de 192.168.2.x via 123.123.123.12. C'est la partie difficile du travail, ajouter Masq par dessus un routage correct est facile.

Pour faire ce routage élégant, vous utiliserez IPROUTE2. Comme cette fonction n'a rien à voir avec IPMASQ, ce HOWTO ne le traite pas en détail. Referez vous SVP à 2.5 () pour des URL et une documentation sur ce sujet.

Les commandes sont les même que les commandes "iprule" et "iproute" (je préfére le premier puisqu'il est plus facile de le chercher). Les commandes ci-dessous ne sont pas testées, si elles ne fonctionnent pas, veuillez contacter l'auteur de IPROUTE2... pas David Ranch ou qui que ce soit dans la mailing list de Masq puisque ça n'a RIEN avoir avec IP Masquerading.

Les toutes premières commandes ont seulement besoin d'être lancé une fois au démarrage, disons dans le fichier /etc/rc.d/rc.local

```
# Permets aux LANs internes de communiquer entre eux, pas de masq.
  /sbin/iprule add from 192.168.0.0/16 to 192.168.0.0/16 table main pref 100
# Tout autre traffic de 192.168.1.x est externe, pris en charge par la table 101
  /sbin/iprule add from 192.168.1.0/24 to 0/0 table 101 pref 102
# Tout autre traffic de 192.168.2.x est externe, pris en charge par la table 102
  /sbin/iprule add from 192.168.2.0/24 to 0/0 table 102 pref 102

Ces commandes ont besoin d'etre testees quand eth0 est configure, peut-etre dans
/etc/sysconfig/network-scripts/ifup-post (systemes RedHat). Lancez les une fois
a la main pour etre sur qu'ils fonctionnent.

# la table 101 force tous les paquets qui lui sont assigne a sortir via 123.123.123.11
  /sbin/iproute add table 101 via 62123.123.123.11
# la table 102 force tous les paquets qui lui sont assigne a sortir via 123.123.123.12
  /sbin/iproute add table 102 via 62123.123.123.12

A partir de la, vous devriez voir que les paquets provenant de 192.168.1.x partant
vers le monde exterieur sont routes; via 123.123.123.11, et les paquets de
192.168.2.x sont route via 123.123.123.12.

Une fois que le routage est correct, vous pouvez ajouter les regles IPFWADM et IPCHAINS.
Les exemples suivants sont pour IPCHAINS :

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

Si tout ce goupille bien, le code de masq va voir les paquets routes via
123.123.123.11 et 123.123.123.12 et va utiliser ces adresses comme adresse source masq.
```

## 7.37 Pourquoi les nouveaux noyaux 2.1.x et 2.2.x utilisent IPCHAINS au lieu de IPFWADM ?

IPCHAINS possède les fonctions suivantes que IPFWADM ne possède pas :

- "Quality of Service" (compatibilité QoS)
- Un système de chaines en forme d'ARBRE, contre un système LINEAIRE pour IPFWADM (ce qui permet de faire des trucs du genre : "si c'est ppp0, saute vers cette chaine (qui contient son propre jeu de règles)")
- IPCHAINS est plus flexible pour la configuration. Par exemple, il a la commande "replace" (remplace) en plus de "insert" et "add" (insere et ajoute). Vous pouvez aussi avoir des règles négatives (par exemple : "ignore tous les paquets venant de l'extérieur qui ne viennent pas de mon IP enregistré" pour que vous ne puissiez pas être la source d'attaques spoofés).
- IPCHAINS peut filtrer tout protrocole IP explicitement, pas seulement TCP, UDP, ICMP

## 7.38 Je viens de faire la mise à jour vers le noyau 2.2.x, pourquoi IP Masquerade ne fonctionne pas ?

Il y a plusieurs choses que vous devez vérifier, si on considère que votre machine Linux IP Masq est bien connectée à Internet et à votre LAN :

- Verifiez que vous avez les fonctions nécessaires et les modules compilés et chargés. Reportez vous aux sections précédentes pour les details.
- Vérifiez `/usr/src/linux/Documentation/Changes` et assurez vous que vous avez la configuration minimale requise pour les outils réseaux d'installés.
- Assurez vous d'avoir bien suivi tous les tests de la section 5 () de ce HOWTO.
- Vous devriez utiliser *ipchains* `http://netfilter.filewatcher.org/ipchains/` pour manipuler IP Masq et les règles de firewalling.
- Les port forwarders standards IPAUTOFW et IPPORTFW ont été remplacés par *IPMASQADM* `http://juanjox.kernelnotes.org/` . Vous aurez besoin d'appliquer ces patches au noyau, de le recompiler, compiler le nouvel outil IPMASQADM et ensuite de convertir vos anciens jeux de règles de firewall IPAUTOFW/IPPORTFW avec la nouvelle syntaxe. Cette partie est entièrement traitée dans la section 6.7 ().
- Recommencer à vérifier toute l'installation et la configuration ! Souvent, c'est juste une erreur typographiqe ou une simple erreur que vous cherchez.

## 7.39 Je viens de faire la mise à jour vers le noyau 2.0.38+, pourquoi IP Masquerade ne fonctionne pas ?

Il y a plusieurs choses que vous devez vérifiant, si on considère que votre machine Linux IP Masq est bien connectée a Internet et à votre LAN :

- Verifiez que vous avez les fonctions nécessaires et les modules compilés et chargés. Reportez vous SVP aux sections précédentes pour les details.

- Vérifiez `/usr/src/linux/Documentation/Changes` et assurez vous que vous avez la configuration minimale requise pour les outils réseaux installés.
- 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 IP Masq et les règles de firewalling. Si vous voulez utiliser IPCHAINS, vous aurez besoin d'appliquer un patch aux noyaux 2.0.x.
- Recommencez à vérifier toute l'installation et la configuration ! Souvent, c'est juste une erreur de typographie ou une erreur toute simple que vous cherchez.

## 7.40 J'ai besoin d'aide sur les connexions EQL et IP Masq

EQL n'a rien à faire avec IP Masq bienqu'ils soient souvent combinés sur les machines Linux. C'est pourquoi, je vous recommande de voir la nouvelle version de *Robert Novak's EQL HOWTO* pour vos besoins sur EQL.

## 7.41 J'arrive pas faire fonctionner IP Masquerade ! Quelles options ai-je pour les Plateformes Windows ?

Vous voulez abandonner une solution gratuite, sûre, haute performance qui fonctionne avec un minimum de ressources matérielles pour quelquechose qui a besoin de plus de matériel, avec des performances inférieures et moins sûr ? (AMHO. Et oui, j'ai des expériences grandeur nature de ces choses là ;-)

Okay, c'est votre choix. Si vous voulez une solution NAT et/ou proxy Windows, voici une liste convenable. Je n'ai pas de préférence pour ces outils puisque je ne m'en suis jamais servi.

- Firesock (par les créateurs de Trumpet Winsock)
  - Fait aussi Proxy
  - *http://www.trumpet.com.au*
- Iproute
  - programme DOS créé pour fonctionner sur des 286+
  - a besion d'une autre machine telle que Linux MASQ
  - *http://www.mischler.com/iproute/*
- Microsoft Proxy
  - Necessite Windows NT Server
  - Très cher
  - *http://www.microsoft.com*
- NAT32
  - Compatible avec Windows 95/98/NT
  - *http://www.nat32.com*
  - Environ $25 pour Win9x et $47 pour WinNT
- SyGate
  - *http://www.sygate.com*
- Wingate

  - Fait Proxy
  - Coûte environ $30 pour 2-3 IPs
  - *http://www.wingate.com*

- Winroute
  - Fait NAT
  - *http://www.winroute.cz/en/*

Enfin faites une recherche sur le web sur "MS Proxy Server", "Wingate", "WinProxy", ou allez sur *www.winfiles.com* `http://www.winfiles.com` . Et ne dites surtout a personnes que c'est nous qui vous envoyons.

## 7.42 Je voudrais aider à developper IP Masquerade. Que puis-je faire ?

Abonnez vous à la mailing list Linux IP Masquerading DEVELOPERS et demander aux développeurs sur quoi vous pouvez aider. Pour plus de détails sur comment s'abonner aux mailing lists, regardez la section 7.4 () de la FAQ.

SVP NE posez PAS de questions non relatifs au développement d'IP Masquerade là-bas !!!!

## 7.43 Où puis-je trouver plus d'informations sur IP Masquerade?

Vous pouvez trouvez plus d'informations sur IP Masquerade ici : *Linux IP Masquerade Resource* `http://ipmasq.cjb.net/` , site dont s'occupe David Ranch.

Vous pouvez aussi trouver des informations sur *Dranch's Linux page* `http://www.ecst.csuchico.edu/~dranch/LINUX/index-linux.html` , où se trouvent les documents de TrinityOS et d'autres documents sur Linux.

Vous pouvez aussi trouver des informations sur *The Semi-Original Linux IP Masquerading Web Site* `http://www.indyramp.com/masq/` entretenu par Indyramp Consulting, qui fournit aussi les mailing lists IP Masq.

Enfin, vous pouvez trouver des réponses aux questions spécifiques dans les archives des mailing lists IP MASQ et IP MASQ DEV. Reportez vous à la FAQ 7.4 () pour de plus amples détails.

## 7.44 Je veux traduire ce HOWTO dans une autre langue, que dois-je faire ?

Assurez vous que la langue dans laquelle vous voulez traduire n'est pas déjà traitée par quelqu'un d'autre. Mais la plupart des HOWTOs traduits sont VIEUX et ont besoin d'être mis à jour. Une liste des HOWTO traduits est disponible ici :

*Linux IP Masquerade Resource* `http://ipmasq.cjb.net/` .

Si une copie de la version **en cours** de l'IP MASQ HOWTO n'existe pas dans la langue que vous proposez, téléchargez SVP la version la plus récente du code SGML de l'IP-MASQ HOWTO ici : *Linux IP Masquerade Resource* `http://ipmasq.cjb.net/` . De là, continuez votre travail tout en produisant du bon code SGML. Pour plus d'aide sur le SGML, vous pouvez voir *www.sgmltools.org* `http://www.sgmltools.org`

### 7.45 Ce HOWTO semble périmé, continuez vous à le mettre à jour ? Pouvez vous inclure plus d'information sur ... ? Comptez vous le rendre meilleur ?

Oui, ce HOWTO est toujours mis à jour. Par le passé, j'ai été coupable d'être trop occupé avec deux emplois et ne pas avoir assez de temps pour travailler dessus, mes excuses. A partir de v1.50, David Ranch a commencé à réaménager ce document et le maintenir à jour.

Si vous pensez qu'un sujet devrait être ajouté à ce HOWTO, envoyez SVP un email à *ambrose@writeme.com* and *dranch@trinnet.net* . Ca serait encore mieux si vous pouviez fournir ces informations. Nous inclurons alors ces informations dans ce HOWTO si nous les trouvons appropriées et quand nous les aurons testées. Merci beaucoup pour vos contributions !

Nous avons beaucoup de nouvelles idées et de plans pour améliorer ce HOWTO, tels que des études de cas qui vont traiter différentes installations réseaux impliquant de IP Masquerade, plus de sécurité via des jeux de règles IPFWADM/IPCHAINS 'strong', plus d'entrées dans la FAQ, etc. Si vous pensez pouvoir nous aider, SVP faites le ! Merci.

### 7.46 Je viens de faire marcher IP Masquerade, c'est super ! Je veux vous remercier les gars, que puis-je faire ?

- Pouvez vous traduire la version la plus récente de ce HOWTO dans une autre langue ?
- Remerciez les developpeurs et évaluez le temps qu'ils y ont passé et les efforts qu'il ont faits.
- Joignez vous à la mailing list IP Masquerade et aidez les nouveaux utilisateurs de MASQ.
- Envoyez nous un email et dites nous à quel point vous êtes heureux.
- Présentez Linux à d'autres personnes et aidez les quand ils ont des problèmes.

# 8 Divers

## 8.1 Sources Utiles

NDT : Toutes ces sources sont bien entendu anglophone et j'ignore s'il existe des versions traduites pour les documents mentionnés.

- *IP Masquerade Resource page* `http://ipmasq.cjb.net/` Regroupe toutes les informations nécessaires pour installer IP Masquerade sur les noyaux 2.0.x, 2.2.x, et même le vieux 1.2 !
- *Juan Jose Ciarlante's WWW site* `http://juanjox.kernelnotes.org` Qui est la personne qui s'occupe actuellement de la maintenance de Linux IP Masquerade.
- *IP Masquerade mailing list Archives* `http://www.indyramp.com/lists/masq` contient les messages envoyés récemment aux mailing lists.
- *David Ranch's Linux page including the TrinityOS Linux document and current versions of the IP-MASQ-HOWTO.* `http://www.ecst.csuchico.edu/~dranch/LINUX/index-linux.html` . Ses sujets tels que IP MASQ, jeux de règles IPFWADM/IPCHAINS 'strong', PPP, Diald, Cablemodems, DNS, Sendmail, Samba, NFS, Security, etc. y sont traités.

- La page des applications IP Masquerading *(IP Masquerading Applications page)* `http://www.tsmservices.com/masq` : Une liste exhaustive des applications qui fonctionnent ou qui peuvent être accordées de façon à ce qu'ils fonctionnent à travers un serveur Linux IP masquerading.
- Pour les personnes qui installent IP Masq sur MkLinux, envoyez un email à Taro Fukunaga : *tarozax@earthlink.net* pour qu'il vous envoie une version de son court HOWTO pour MkLinux.
- *IP masquerade FAQ* `http://www.indyramp.com/masq/ip_masquerade.txt` quelques informations d'ordre général
- *http://netfilter.filewatcher.org/ipchains/* La doc de Paul Russel et peut-être une ancienne sauvegarde ici : *Linux IPCHAINS HOWTO* . Ce HOWTO contient beaucoup d'informations sur l'utilisation d'IPCHAINS, de même que les sources et les fichiers binaires de l'outil ipchains.
- *X/OS Ipfwadm page* `http://www.xos.nl/linux/ipfwadm/` contiens les sources, les binaires, la documentation, et d'autres informations au sujet de paquet `ipfwadm`
- Allez voir la *GreatCircle's Firewall mailing list* : Une excellente source pour les jeux de règles 'strong' pour le firewall.
- Le *LDP Network Administrator's Guide* `http://www.linuxdoc.org/LDP/nag/nag.html` est un MUST pour l'administrateur Linux débutant essayant d'installer un réseau.
- Le *Linux NET-3-4 HOWTO* `http://www.linuxdoc.org/HOWTO/NET3-4-HOWTO.html` est aussi un autre document exhaustif sur la manière d'installer et de configurer un réseau Linux.
- Les *Linux ISP Hookup HOWTO* `http://www.linuxdoc.org/HOWTO/ISP-Hookup-HOWTO.html` et *Linux PPP HOWTO* `http://www.linuxdoc.org/HOWTO/PPP-HOWTO.html` vous fournissent les renseignements nécessaires sur les démarches à suivre pour connecter votre machine Linux à Internet.
- Le *Linux Ethernet-Howto* `http://www.linuxdoc.org/HOWTO/Ethernet-HOWTO.html` est une bonne source d'information pour installer un LAN Ethernet.
- *Donald Becker's NIC drivers and Support Utils* `http://cesdis.gsfc.nasa.gov/linux/drivers/`
- Vous pourriez aussi peut-être être intéressé par le *Linux Firewalling and Proxy Server HOWTO* `http://www.linuxdoc.org/HOWTO/Firewall-HOWTO.html`
- Le *Linux Kernel HOWTO* `http://www.linuxdoc.org/HOWTO/Kernel-HOWTO.html` vous guidera à travers le processus de compilation du noyau.
- D'autres *Linux HOWTOs* `http://www.linuxdoc.org/HOWTO/HOWTO-INDEX/howtos.html` tels que Kernel HOWTO
- Faire des 'Post' sur le newsgroup USENET : *comp.os.linux.networking*

## 8.2 Sources Linux IP Masquerade

La *Linux IP Masquerade Resource* `http://ipmasq.cjb.net/` est un site web dédié à Linux IP Masquerade dont s'occupe aussi Ambrose Au. Il a les dernières informations relatif à IP Masquerade et toute autre information qui pourrait ne pas être inclus dans ce HOWTO.

Vous pouvez trouver La Linux IP Masquerade Resource aux endroits suivant :

- *http://ipmasq.cjb.net/* , Site Primaire, redirigé vers *http://ipmasq.cjb.net/*
- *http://ipmasq2.cjb.net/* , Site Secondaire, redirigé vers *http://www.geocities.com/SiliconValley/Heights/2288/*

## 8.3 Merci aux personnes suivantes :

Par ordre alphabétique :

- Gabriel Beitler, gabrielb@voicenet.com
  pour avoir fourni la section 3.3.8 (installation de Novell)
- Juan Jose Ciarlante, irriga@impsat1.com.ar
  pour ses contributions à son outil de port forwarding pour IPMASQADM, son travail sur les sources des noyaux 2.1.x et 2.2.x, le patch LooseUDP original, etc.
- Steven Clarke, steven@monmouth.demon.co.uk
  pour sa contribution : l'outil IP de port forwarding IPPORTFW
- Andrew Deryabin, djsf@usa.net
  pour sa contribution : le module ICQ MASQ
- Ed Doolittle, dolittle@math.toronto.edu
  pour sa suggestion : l'option `-V` dans la commande `ipfwadm` pour une sécurité accrue
- Matthew Driver, mdriver@cfmeu.asn.au
  pour son importante aide à ce HOWTO, et avoir fourni la section 3.3.1 (configuration de Windows 95)
- Ken Eves, ken@eves.com
  pour la FAQ qui fourni des informations inestimables à ce HOWTO
- John Hardin, jhardin@wolfenet.com
  pour ses outils de forwarding de PPTP et de IPSEC
- Glenn Lamb, mumford@netcom.com
  pour le patch LooseUDP
- Ed. Lott, edlott@neosoft.com
  pour une longue liste de systèmes et de logiciels testés
- Nigel Metheringham, Nigel.Metheringham@theplanet.net
  pour ses versions des HOWTO sur l'IP Packet Filtering et l'IP Masquerading, qui font de ce HOWTO un document meilleur et plus technique
  sections 4.1, 4.2, et d'autres
- Keith Owens, kaos@ocs.com.au
  pour son excellent guide sur ipfwadm section 4.2
  et sa correction de l'option `ipfwadm -deny` qui évite un trou de sécurité, et clarifie le status du `ping` avec IP Masquerade
- Michael Owings, mikey@swampgas.com
  pour sa section sur CU-SeeMe et son Linux IP-Masquerade Teeny How-To
- Rob Pelkey, rpelkey@abacus.bates.edu
  pour les sections 3.3.6 et 3.3.7 (configuration de MacTCP et d'Open Transport)
- Harish Pillay, h.pillay@ieee.org
  pour la section 4.5 (connexion avec Diald)
- Mark Purcell, purcell@rmcs.cranfield.ac.uk
  pour la section 4.6 (IPautofw)

- David Ranch, dranch@trinnet.net
  aide à mettre à jour et à entretenir ce HOWTO et la Linux IP Masquerade Resource Page, le document TrinityOS , ..., trop de choses pour être listés ici :-)

- Paul Russell, rusty@linuxcare.com.au
  pour tout son travail sur IP CHAINS, les patches noyau IP Masquerade, etc

- Ueli Rutishauser, rutish@ibm.net
  pour la section 3.3.9 (configuration d'OS/2 Warp)

- Steve Grevemeyer, grevemes@tsmservices.com
  pour avoir repris la page IP Masq Applications à Lee Nevo et l'avoir mise à jour en une backend de BD

- Fred Viles, fv@episupport.com& nl;pour ses patches pour un port forwarding correct de FTP

- John B. (Brent) Williams, forerunner@mercury.net
  pour la section 3.3.7 (configuraion d'Open Transport)

- Enrique Pessoa Xavier, enrique@labma.ufrj.br
  pour sa suggestion de configuraiton pour BOOTp

- Toutes les personnes de la mailing list d'IP-MASQ, masq@tiffany.indyramp.com
  pour leur aide et leur soutien aux nouveaux utilisateurs de Linux MASQ

- Les autres développeurs de code et de documentations d'IP Masquerade pour cette exceptionne fonction

  - 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 ceux que j'aurais pu oubliés de mentionner ici (SVP faites le moi savoir)

- Tous les utilisateurs qui ont envoyé des feedback ou des suggestions à la mailing list, surtout ceux qui ont signalé des erreurs dans ce document ou les clients qui sont compatibles ou pas.

- Nous nous excusons si nous avons omis des noms importants, pas encore inclus des informations que certains utilisateurs nous ont envoyé, etc. Il y a de nombreuses suggestions et idées qui nous sont envoyés mais il n'y pas assez de temps pour vérifier et intégrer ces changements. David Ranch essaie continuellement de faire de son mieux pour intégrer ces informations, que l'on m'envoie, dans ce HOWTO. Je vous remerci de votre effort, et j'espère que vous comprendrez notre situtation.

## 8.4 Reference

- IP masquerade FAQ original par Ken Eves
- archive de l'IP masquerade mailing list par Indyramp Consulting
- Site WWWW IP Masquerade par Ambrose Au
- page Ipfwadm par X/OS
- Linux HOWTOs Linux sur les réseaux
- Certains sujets traités dans TrinityOS par David Ranch

## 8.5 Changes

- TO do - HOWTO:
  - Add the scripted IPMASQADM example to the Forwarders section. Also confirm the syntax.
  - Add a little section on having multiple subnets behind a MASQ server
  - Confirm the IPCHAINS ruleset and make sure it is consistant with the IPFWADM ruleset

  TO DO - WWW page:

  - Update all PPTP urls from lowrent to ftp://ftp.rubyriver.com/pub/jhardin/masquerade/ip_masq_vpn.html
  - Update the PPTP patch on the masq site
  - Update the portfw FTP patch

  Changes from 1.90 to 1.95 - 11/14/00

  - Added a quick upfront notice in the intro that running a SINGLE NIC in MASQ mutliple ethernet segments is NOT recommended and linked to the relivant FAQ entry. Thanks to Daniel Chudnov for helping the HOWTO be more clear.
  - Added a pointer in the Intro section to the FAQ section for users looking for how MASQ is different from NAT and Proxy services.
  - Reordered the Kernel requirements sections to be 2.2.x, 2.4.x, 2.0.x
  - Expanded the kernel testing in Section 3 to see if a given kernel already supports MASQ or not.
  - Reversed the order of the displayed simple MASQ ruleset examples (2.2.x and 2.0.x)
  - Cleaned up some formatting issues in the 2.0.x and 2.2.x rc.firewall files
  - Noted in the 2.2.x rc.firewall that the defrag option is gone in some distro's proc (Debian, TurboLinux, etc)
  - Added a NOTE #3 to the rc.firewall scripts to include instructions for Pump. Thanks to Ross Johnson for this one.
  - Cleaned up the simple MASQ ruleset examples for both the 2.2.x and 2.2.x kernels
  - Updated the simple and stronger IPCHAINS and IPFWADM rulesets to include the external interface names (IPCHAINS is -i; IPFWADM is -W) to avoid some internal traffic MASQing issues.
  - Vastly expanded the Section 5 (testing) with even more testing steps with added complete examples of what the output of the testing commands should look like.
  - Moved the H.323 application documentation from NOT supported to Supported. :)
  - Reordered the Multiple LAN section examples (2.2.x then 2.0.x)
  - Made some additional clarifications to the Multiple LAN examples Fixed a critical typo with multiple NIC MASQing where the network examples had the specified networks reversed. Thanks to Matt Goheen for catching this.
  - Added a little intro to MFW in the PORTFW section.
  - Reveresed the 2.0.x and 2.2.x sections for PORTFW
  - Updated the news regarding PORTFWing FTP traffic for 2.2.x kernels

```
NOTE:  At this time, there *IS* a BETA level IP_MASQ_FTP module
       for PORT Forwarding FTP connections 2.2.x kernels which also supports
       adding additional PORTFW FTP ports on the fly without the requirement
       of unloading and reloaded the IP_MASQ_FTP module and thus breaking any
       existing FTP transfers.
```

- Added a top level note about PORTFWed FTP support
- Added a noted to the 2.0.x PORTFW'ed FTP example why users DON'T need to PORTFW port 20.
- Updated the PORTFW section to also mention that users can use FTP proxy applications like the one from SuSe to support PORTFWed FTP-like functionality. Thanks to Stephen Graham for this one.
- Updated the example for how to enable PORTFWed FTP to also include required configurations to how the ip_masq_ftp module is loaded for users who use multiple PORTs to contact multiple internal FTP servers. Thanks to Bob Britton for reminding me about this one.
- Added a FAQ entry for users who have embedded ^Ms in their rc.firewall file
- Expanded the FAQ entry talking about how MASQ is different from NAT and Proxy to include some informative URLs.
- Updated the explanation of the MASQ MTU issue and describe the two main explanations of the issue.
- Clarified that per the RFC, PPPoE should only require an MTU of 1490 though some ISPs require a setting of 1460. Because of this, I have updated the example to show an MTU of 1490.
- Broke out the Windows 9x sections into Win95 and Win98 as they use different settings (DWORD vs. STRING). I also updated the sections to be more clear and the Registry backup methods have been updated.
- Fixed a typo where the NT 4.0 Registry entries were backwards (Tcpip/Parameters vs. Parameters/Tcpip).
- Fixed an issue where the WinNT entry should have been a DWORD and not a STRING.
- A serious thanks goes out to Geoff Mottram for his various PPPoE and various Windows Registry entry fixes.
- Added an explicit URL for Oident in the IRC FAQ entry
- Updated the FAQ section regarding some broken "netstat" versions
- Added new FAQ sections for MASQ accounting ideas and traffic shaping
- Expanded the IPROUTE2 FAQ entry on what Policy-routing is.
- Moved the IPROUTE2 URLs to the 2.2.x Kernel requirements section and also added a few more URLs as well.
- Corrected the "intnet" variable in the stronger IPCHAINS ruleset to reflect the 192.168.0.0 network to be consistent with the rest of the example. Thanks to Ross Johnson for this one.
- Added a new FAQ section for people asking about forwarding problems between multiple internal MASQed LANs.
- Added a new FAQ section about users wanting to PORTFW all ports from multiple external IP addresses to internal ones. I also touched on people trying to PORTFW all ports on multiple IP ALIASed interfaces and also noted the Bridge+Firewall HOWTO for DSL and Cablemodem users who have multiple IPs in a non-routed environment.
- Added Mandrake 7.1, Mandrake 7.2, and Slackware 7.1 to the supported list
- Added Redhat 7.0 to the MASQ supported distros. Thanks to Eugene Goldstein for this one.
- Fixed a mathematical error in the "Maximum Throughput" calculation in the FAQ section. Thanks to Joe White @ ip255@msn.com for this one.
- Fixed the fact that the Windows9x MTU changes are a STRING change and not a DWORD change to the registry. Thanks to jmoore@sober.com for this one.

- Updated the comments in the 2.0.x rc.firewall script to note that the ip_defrag option is for both 2.0 and 2.2 kernels. Thanks to pumilia@est.it for this clarification.

Changes from 1.85 to 1.90 - 07/03/00

- Updated the URL for TrinityOS to reflect its new layout
- Caught a typo in the IPCHAINS rulesets where I was setting "ip_ip_always_defrag" instead of "ip_always_defrag"
- The URL to Taro Fukunaga was invaild since it was using "mail:" instead of "mailto:"
- Added some clarification to the "Masqing multiple internal interfaces" where some people didn't understand why eth0 was referenced multiple times.
- Fixed another "space after the EXTIP variable" bug in the stronger IPCHAINS section. I guess I missed one.
- In Test #7 of Section 5, I referred users to go back to step #4. Thats should have been step #6.
- Updated the kernel versions that came with SuSe 5.2 and 6.0
- Fixed a typo (or vs. of) in Section 7.2
- Added Item #9 to the Testing MASQ section to refer users who are still haing MASQ problems to read the MTU entry in the FAQ
- Improved the itemization in Section 5
- Updated the IPCHAINS syntax to show the MASQ/FORWARD table. Before, it was valid to run "ipchains -F -L" but now only "ipchains -M -L" works.
- Updated the LooseUDP documentation to reflect the new LooseUDP behavior in 2.2.16+ kernels. Before, it was always enabled, now, it defaults to OFF due to a possible MASQed UDP port scanning vunerability. I have updated the BASIC and SEMI-STRONG IPCHAINS rulesets to reflect this option.
- Updated the recommended 2.2.x kernel to be 2.2.16+ since there is a TCP root exploit vunerability in all lesser versions.
- Added Redhat 6.2 to the MASQ supported list
- Updated the link for Sonny Parlin's FWCONFIG to now point to fBuilder.
- Updated the various example IP addresses from 111.222.333.444 to be 111.222.121.212 to be within a valid IP address range
- Updated the URL for the BETA H.323 MASQ module
- Finally updated the MTU FAQ section to help out PPPoE DSL and Cablemodem users. Basically, the 7.14 () section now reflects that users can also change the MTU settings of all of their INTERNAL machines to solve the dreaded MASQ MTU issue.
- Added a clarification to the PORTFW section that PORTFWed connections that work for EXTERNAL clients will not work for INTERNAL clients. If you also need INTERNAL portfw, you will need to also impliment the REDIR tool as well. I also noted that this issue is fixed in the 2.4.x kernels with Netfilter.
- I also added a technical explanation from Juanjo to the end of the PORTFW section to why this senario doesn't work properly.
- Updated all of the IPCHAINS URLs to point to Paul Rusty's new site at http://netfilter.filewatcher.org/ipchains/
- Updated Paul Rustys email address
- Added a new FAQ section for users whose connections remain idle for a long time and their PORTFWed connection no longer work.

- Updated all the URLs to the LDP that pointed to metalab.unc.edu to the new site of linuxdoc.org
- Updated the Netfilter URLs to point to renamed HOWTOs, etc.
- I also updated the status of the 2.4.x support to note that I *will* add full Netfilter support to this HOWTO and if the time comes, then split that support off into a different HOWTO.
- Updated the 2.4.x Requirements section to reflect how NetFilter has changed compared to IPFWADM and IPCHAINS and gave a PROs/CONs list of new features and changes to old behaviors.
- Added a TCP/IP math example to the "My MASQ connection is slow" FAQ entry to better explain what a user should expect performance wise.
- Updated the HOWTO to reflect that newer versions of the "pump" DHCP client now can run scripts upon bringup, lease renew, etc.
- Updated the PORTFWing of FTP to reflect that several users say they can successfully forward FTP traffic to internal machines without the need of a special ip_masq_ftp module. I have made the HOWTO reflect that users should try it without the modified module first and then move to the patch if required.

Changes from 1.82 to 1.85 - 05/29/00

- Ambrose Au's name has been taken off the title page as David Ranch has been the primary maintainer for the HOWTO for over a year. Ambrose will still be involved with the WWW site though.
- Deleted a stray SPACE in section 6.4
- Re-ordered the compatible MASQ'ed OS section and added instructions for setting up a AS/400 system running on OS/400. Thanks to jaco@libero.it for the notes.
- Added an additional PORFW-FTP patch URL for FTP access if HTTP access fails.
- Updated the kernel versions for Redhat 5.1 & 6.1 in the FAQ
- Added FloppyFW to the list of MASQ-enabled Linux distros
- Fixed an issue in the Stronger IPFWADM rule set where there were spaces between "ppp_ip" and the "=".
- In the kernel compiling section for 2.2.x kernels, I removed the reference to enable "CONFIG_IP_ALWAYS_DEFRAG". This option was removed from the compiling section and enabled by default with MASQ enabled in 2.2.12.
- Because of the above change in the kernel behavior, I have added the enabling of ip_always_defrag to all the rc.firewall examples.
- Updated the status of support for H.323. There is now ALPHA versions of modules to support H.323 on both 2.0.x and 2.2.x kernels.
- Added Debian v2.2 to the supported MASQ distributions list
- Fixed a long standing issue where the section that covered explicit filtering of IP addresses for IPCHAINS had old IPFWADM syntax. I've also cleaned this section up a little and made it a little more understandable.
- Doh! Added Juan Ciarlante's URL to the important MASQ resources section. Man.. you guys need to make me more honest than this!!
- Updated the HOWTO to reflect kernels 2.0.38 and 2.2.15
- Rerversed the order shown to compile kernels to show 2.2.x kernels first as 2.0.x is getting pretty old.

– Updated the 2.2.x kernel compiling section to reflect the changed options for the latter 2.2.x kernels.
– Added a a possible solution for people that fail to get past MASQ test #5.

Changes from 1.81 to 1.82 - 01/22/00

– Added a missing subsection for /proc/sys/net/ipv4/ip_dynaddr in the stronger IPCHAINS ruleset. Section 6.5
– Changed the IP Masq support for Debian 2.1 to OUI
– Reorganized and updated the "Masq is slow" FAQ section to include fixing Ethernet speed and duplex issues.
– Added a link to Donald Becker's MII utilities for Ethernet NIC cards
– Added a missing ")" for the 2.2.x section (previously fixed it only for the 2.0.x version) to the ICQ portfw script and changed the evaluation from -lt to -le
– Added Caldera eServer v2.3 to the MASQ supported list
– Added Mandrake 6.0, 6.1, 7.0 to the MASQ supported list
– Added Slackware v7.0 to the MASQ supported list
– Added Redhat 6.1 to the MASQ supported list
– Added TurboLinux 4.0 Lite to the MASQ supported list
– Added SuSe 6.3 to the MASQ supported list
– Updated the recommended stable 2.2.x kernel to be anything newer than 2.2.11
– In section 3.3, the HOWTO forgot how to tell the user how to load the /etc/rc.d/rc.firewall upon each reboot. This has now been covered for Redhat (and Redhat-based distros) and Slackware.
– Added clarification in the Windows WFWG v3.x and NT setup sections why users should NOT configure the DHCP, WINS, and Forwarding options.
– Added a FAQ section on how to fix FTP problems with MASQed machines.
– Fixed a typo in the Stronger firewall rulesets. The "extip" variabl cannot have the SPACE between the variable name and the "=" sign. Thanks to johnh@mdscomp.com for the sharp eye.
– Updated the compatibly section: Mandrake 7.0 is based on 2.2.14 and TurboLinux v6.0 runs 2.2.12

Changes from 1.80 to 1.81 - 01/09/00

– Updated the ICQ section to reflect that the new ICQ Masq module supports file transfer and real-time chat. The 2.0.x module still has those limitations.
– Updated Steven E. Grevemeyer's email address. He is the maintainer of the IP Masq Applications page.
– Fixed a few lines that were missing the work AREN'T for the "setsockopt" errors.
– Updated a error the strong IPCHAINS ruleset where it was using the variable name "ppp_ip" instead of "extip".
– Fixed a "." vs a "?" typo in section 3.3.1 in the DHCP comment section.
– Added a missing ")" to the ICQ portfw script and changed the evaluation from -lt to -le
– Updated the Quake Module syntax to NOT use the "ports=" verbage

Changes from 1.79 to 1.80 - 12/26/99

– Fixed a space typo when setting the "ppp_ip" address.

- Fixed a typo in the simple IPCHAINS ruleset. "deny" to "DENY"
- Updated the URLs for Bjorn's "modutils" for Linux
- Added verbage about NetFilter and IPTables and gave URLs until it is added to this HOWTO or a different HOWTO.
- Updated the simple /etc/rc.d/rc.firewall examples to notify users about the old Quake module bug.
- Updated the STRONG IPFWADM /etc/rc.d/rc.firewall to clarify users about dynamic IP addresses (PPP & DHCP), newer DHCPCD syntax, and the old Quake module bug.
- Updated the STRONG IPCHAINS /etc/rc.d/rc.firewall to ADD a missing section on dynamic IP addresses (PPP & DHCP) and the old Quake module bug.
- Added a note in the "Applications that DO NOT work" section that there IS a beta module for Microsoft NetMeeting (H.323 based) v2.x on 2.0.x kernels. There is NON versions available for Netmeeting 3.x and/or 2.2.x kernels as of yet.

Changes from 1.78 to 1.79 - 10/21/99

- Updated the HOWTO name to reflect that it isn't a MINI anymore!

Changes from 1.77 to 1.78 - 8/24/99

- Fixed a typeo in "Section 6.6 - Multiple Internal Networks" where the -a policy was ommited.
- Deleted the 2.2.x kernel configure option "Drop source routed frames" since it is now enabled by default and the kernel compile option was removed.
- Updated the 2.2.x and all other IPCHAINS sections to notify users of the IPCHAINS fragmentation bug.
- Updated all the URLs point at Lee Nevo's old IP Masq Applications page to Seg's new page.

Changes from 1.76 to 1.77 - 7/26/99

- Fixed a typo in the Port fowarding section that used "ipmasqadm ipportfw -C" instead of "ipmasqadm portfw -f"

Changes from 1.75 to 1.76 - 7/19/99

- Updated the "ipfwadm: setsockopt failed: Protocol not available" message in the FAQ to be more clear instead of making the user hunt for the answer in the Forwarders section.
- Fixed incorrect syntax in section 6.7 for IPMASQADM and "portfw"

Changes from 1.72 to 1.75 - 6/19/99

- Fixed the quake module port setup order for the weak IPFWADM & IPCHAINS ruleset and the strong IPFWADM ruleset as well.
- Added a user report about port forwarding ICQ 4000 directly in and using ICQ's default settings WITHOUT enabling the "Non-Sock" proxy setup.
- Updated the URLs for the IPMASQADM tool
- Added references to Taro Fukunaga, tarozax@earthlink.net for his MkLinux port of the HOWTO
- Updated the blurb about Sonny Parlin's FWCONFIG tool to note new IPCHAINS support
- Noted that Fred Vile's patch for portfw'ed FTP access is ONLY available for the 2.0.x kernels
- Updated the 2.2.x kernel step with a few clarifications on the Experiemental tag
- Added Glen Lamb's name to the credits for the LooseUDP patch

- Added a clarification on installing the LooseUDP patch that it should use "cat" for non-compressed patches.
- Fixed a typo in the IPAUTO FAQ section
- I had the DHCP client port numbers reversed for the IPFWADM and IPCHAINS rulesets. The order I had was if your Linux server was a DHCP SERVER.
- Added explicit /sbin path to all weak and strong ruleset examples.
- Made some clarifications in the strong IPFWADM section regarding Dynamic IP addresses for PPP and DHCP users. I also noted that the strong rulesets should be re-run when PPP comes up or when a DHCP lease is renewed.
- Added reference in the 2.2.x requirements, updated the ICQ FAQ section, and added Andrew Deryabin to credits section for his ICQ MASQ module.
- Added some clarifcation in the FAQ section why the 2.1.x and 2.2.x kernels went to IPCHAINS.
- Added a little FAQ section on Microsoft File/Print/Domain services (Samba) through a MASQ server. I also added a URL to a Microsoft Knowledge base document pour de plus amples détails.
- Added clarification in the FAQ section that NON Debian distribution supports IP masq out of the box.
- Updated the supported MASQ distributions in the FAQ section.
- Added to the Aliased NIC section of the FAQ that you CANNOT masq out of an aliased interface.
- Wow.. never caught this before but the "ppp-ip" variable in the strong ruleset section is an invalid variable name! It has been renamed to "ppp_ip"
- In both the IPFWADM and IPCHAINS simple ruleset setup areas, I had a commented out section on enabling DHCP traffic. Problem is, it was below the final reject line! Doh! I moved both up a section.
- In the simple IPCHAINS setup, the #ed out line for DHCP users, I was using the IPFWADM "-W" command instead of IPCHAINS's "-i" parameter.
- Added a little blurb to the Forwarders section the resolution to the famous "ipfwadm: setsockopt failed: Protocol not available" error. This also includes a little /proc test to let people confirm if IPPORTFW is enabled in the kernel. I also added this error to a FAQ section for simple searching.
- Added a Strong IPCHAINS ruleset to the HOWTO
- Added a FAQ section explaining the "kernel: ip_masq_new(proto=UDP): no free ports." error.
- Added an example of scripting IPMASQADM PORTFW rules
- Updated a few of the Linux Documentation Project (LDP) URLs
- Added Quake III support in the module loading sections of all the rc.firewall rulesets.
- Fixed the IPMASQADM forwards for ICQ

- 1.72 - 4/14/99 - Dranch: Added a large list of Windows NAT/Proxy alternatives with rough pricing and URLs to the FAQ.
- 1.71 - 4/13/99 - Dranch: Added IPCHAINS setups for multiple internal MASQed networks. Changed the ICQ setup to use ICQ's default 60 second timeout and change IPFWADM/IPCHAINS timeout to 160 seconds. Updated the MASQ and MASQ-DEV email list and archive subscription instructions.
- 1.70 - 3/30/99 - Dranch: Added two new FAQ sections that cover SMTP/POP-3 timeout problems and how to masquerade multiple internal networks out different external IP addresses with IPROUTE2.

- 1.65 - 3/29/99 - Dranch: Typo fixes, clarifications of required 2.2.x kernel options, added dynamic PPP IP address support to the strong firewall section, additional quake II module ports, noted that the LooseUDP patch is built into later 2.2.x kernels and its from Glenn Lamb and not Dan Kegel, added more game info in the compatibility section.
- 1.62 - Dranch: Make the final first-draft changes to the doc and now announce it the the MASQ email list.
- 1.61 - Dranch: Make editorial changes, cleaned things up and fixed some errors in the Windows95 and NT setups.
- 1.58 - Dranch: Addition of the port forwarding sections; LooseUDP setup; Ident servers for IRC users, how to read firewall logs, deleted the CuSeeme Mini-HOWTO since it is rarely used.
- 1.55 - Dranch: Complete overhaul, feature and FAQ addition, and editing sweep of the v1.50 HOWTO. Completed the 2.2.x kernel and IPCHAINS configurations. Did a conversion from IPAUTOFW to IPPORTFW for the examples that applied. Added many URLs to various other documentation and utility sites. There are so many changes.. I hope everyone likes it. Final publishing of this new rev of the HOWTO to the LDP project won't happen until the doc is looked over and approved by the IP MASQ email list (then v2.00).
- 1.50 - Ambrose: A serious update to the HOWTO and the initial addition of the 2.2.0 and IPCHAINS configurations.
- 1.20 - Ambrose: One of the more recent HOWTO versions that solely dealt with $<$ 2.0.x kernels and IPFWADM.