Linux WWW-HOWTO

Wayne Leister, *n3mtr@qis.net* ;
Version Française par Arnaud Launay, *asl@launay.org* v0.82, 19 Novembre 1997.

Ce document contient des informations sur la mise en place de services WWW sous Linux (à la fois serveur et client). Il n'est pas prévu pour être un manuel détaillé mais une vue d'ensemble et un bon pointeur vers des informations complémentaires.

Contents

1 Introduction

La plupart des gens sont passés sous Linux car ils cherchaient une plateforme véritablement *adaptée à l'Internet*. De plus, des institutions, des universités, des organisations à but non lucratif, et de petites entreprises désirent lancer des sites internets à peu de frais. C'est ici que le WWW-HOWTO intervient. Ce document explique comment configurer les clients et les serveurs pour la plus grande audience du net - *Le World Wide Web*.

Tous les prix indiqués dans ce document sont en dollars US. Ce document suppose que vous utilisez Linux sur une plateforme Intel. Les instructions et les produits disponibles peuvent varier de plateforme en platerforme. Il y a de nombreux liens pour charger les logiciels dans ce document. Utilisez autant que possible un site miroir pour charger plus rapidement et maintenir faible l'encombrement du serveur principal.

Le gouvernement US interdit aux compagnies US d'exporter de l'encryption de plus de 40 bits de long. Par la même, les compagnies US créeront deux versions de leurs logiciels. La version destinée au marché intérieur supportera 128 bits, et la version d'export ne supportera que 40 bits. Ceci s'applique aux butineurs web et aux serveurs supportant les transactions sécurisées. L'autre nom des transactions sécurisées est le Secure Socket Layer (SSL). Nous nous référerons à ces transactions comme SSL pour le reste du document.

1.1 Copyright

1.2 Echos

Tout écho est le bienvenu. Je ne clame pas être un expert. Une partie des informations provient de sites web mal écrits; il est très probable qu'il y ait des erreurs et des omissions. Cependant, vérifiez que vous avez la dernière version avant d'envoyer des corrections; ce pourrait être fixé dans la version suivante (voyez la section qui suit pour savoir où trouver la dernière version). Envoyez vos réactions à *n3mtr@qis.net* .

1.3 Nouvelles versions de ce document

Les nouvelles versions de ce document peuvent être récupérées au format texte sur Sunsite à `http://sunsite.unc.edu/pub/Linux/docs/HOWTO/WWW-HOWTO` et sur la plupart des sites mirroirs Linux. Vous pouvez voir la dernière version HTML sur le web à `http://sunsite.unc.edu/LDP/HOWTO/WWW-HOWTO.html` . Il y a aussi des versions HTML disponibles sur Sunsite sous forme d'archive tar.

2 Configurer un client WWW

Le chapitre qui suit est dédié à la configuration des navigateurs. Vous êtes libres de me contacter si votre navigateur favori n'est pas présent ici. Dans cette version du document seul un petit nombre de navigateurs ont leur propre section, mais je vais essayer de tous les inclure (tous ceux que je peux trouver) dans la section Survol. Dans le futur ces navigateurs auront chacun leur propre section.

L'information de survol est destiné à vous aider à vous décider en faveur d'un navigateur, et vous donne les informations principales sur chaque navigateur. La section détails est destiné à vous aider à installer, configurer, et maintenir chaque navigateur.

Personnellement, je préfère Netscape; c'est le seul navigateur qui offre les dernières nouveautés en HTML. Par exemple, les Frames, Java, Javascript, les feuilles de styles, les layers, et les transactions sécurisées. Rien n'est plus désagréable que de tenter de visiter un site web et de s'apercevoir que vous ne pouvez le voir parce que votre navigateur ne supporte pas quelque chose de nouveau.

Cependant j'utilise Lynx lorsque je n'ai pas envie de lancer le monstre X-Window/Netscape.

2.1 Survol

5 (Navigator/Communicator)

Netscape Navigator est le seul navigateur mentionné ici qui peut utiliser les dernières nouveautés HTML. Quelques unes de ces extensions sont le Java, le Javascript, les mises à jour automatiques, et les layers. Il est aussi capable de lire les news et le courrier. Mais c'est un gros mangeur de ressources; il prend beaucoup de temps processeur et de mémoire. Il utilise également un cache séparé pour tous les utilisateurs, ce faisant utilisant de l'espace disque. Netscape est un produit commercial. Les compagnies ont une période d'essai de 30 jours, mais il n'y a pas de limite pour les individus. Je voudrais cependant vous encourager à vous enregistrer pour supporter Netscape dans leurs efforts contre Microsoft (et qu'est ce que c'est que $40US). Mon sentiment est que, si Microsoft gagne, nous serons obligés d'utiliser MS Internet Explorer sur une plateforme Windows :(

3 (Lynx)

Lynx est le plus petit navigateur web. C'est le roi des navigateurs en mode texte. Il est gratuit et son code source est disponible sous les termes de la GNU public license. Il est en mode texte, mais dispose de nombreuses options.

Kfm

Kfm fait partie de K Desktop Environment (KDE). KDE est un système qui tourne au-dessus de X-Window. Il vous donne de nombreuses spécificités comme le copier - coller, les sons, une corbeille et une apparence unifiée. Kfm est le K File Manager, mais c'est aussi un navigateur web. Ne soyez pas étonné par son nom, pour un produit jeune il est très utilisable en tant que navigateur. Il supporte déjà les frames, les tableaux, les chargements par ftp, la navigation dans les archives tar, et beaucoup d'autres. La version actuelle de Kfm est la 1.39, et elle est gratuite. Kfm peut-être utilisé sans KDE, mais vous aurez besoin des librairies utilisées par KDE. Pour plus d'informations sur KDE et Kfm, visitez le site web de KDE à `http://www.kde.org` .

4 (Emacs)

Emacs est un programme qui fait tout. C'est un traitement de texte, un lecteur de news, un lecteur de courrier, et un navigateur web. Son chemin pour apprendre est très raide, car vous devez apprendre ce que font toutes les touches. La version X-Window est plus facile à utiliser, car la plupart des fonctions sont sur des menus. Un autre inconvénient est qu'il est principalement basé sur du texte. (il peut afficher des images si vous l'utilisez depuis X-Window). Il est également gratuit, et le code source est disponible sous la licence publique GNU.

NCSA Mosaic

Mosaic est un navigateur X-Window développé par le National Center for Supercomputing Applications (NCSA) de l'université de l'Illinois. Le NCSA a passé quatre ans sur ce projet et a maintenant bougé vers d'autres choses. La dernière version est la 2.6 qui a été lancé le 7 Juillet 1995. Le code source est disponible pour toute utilisation non-commerciale. *Spyglass Inc.* `http://www.spyglass.com`

possède les droits commerciaux sur Mosaic. C'est un navigateur X-Window solide, mais il lui manque les dernières nouveautés du HTML. Pour plus d'informations visitez la page NCSA Mosaic à `http://www.ncsa.uiuc.edu/SDG/Software/Mosaic/` . Ce logiciel peut-être chargé à partir de `ftp://ftp.ncsa.uiuc.edu/Mosaic/Unix/binaries/2.6/Mosaic-linux-2.6.Z` .

Arena

Arena était un navigateur X-Window conceptuel pour le W3C (World Wide Web Consortium) lorsqu'ils testaient le HTML 3.0. C'est pourquoi il supporte tous les standards HTML 3.0 comme les feuilles de style et les tableaux. Le développement a été effectué par Yggdrasil Computing, dans l'idée d'en faire un navigateur X-Window totalement libre. Mais le développement a été stoppé en février 1997 avec la version 0.3.11. Une partie seulement du standard HTML 3.2 a été implémentée. Le code source est disponible sous la license publique GNU. Pour plus d'informations voyez le site web à `http://www.yggdrasil.com/Products/Arena/` . Il peut être chargé de `ftp://ftp.yggdrasil.com/pub/dist/web/arena/` .

Amaya

Amaya est le navigateur X-Window conceptuel pour le W3C pour le HTML 3.2. Toutefois il supporte tous les standards HTML 3.2. Il supporte également quelques unes des nouveautés du HTML 4.0. Il supporte les tableaux, les formulaires, les cartes d'images du client, la publication, les images gifs, jpegs et png. C'est à la fois un navigateur et un outil d'apprentissage. La dernière version publique est la 1.0 beta. La version 1.1 beta est en tests internes et doit sortir bientôt. Pour plus d'informations voyez le site d'Amaya à `http://www.w3.org/Amaya/` . Il peut être chargé de `ftp://ftp.w3.org/pub/Amaya-LINUX-ELF-1.0b.tar.gz` .

Red Baron

Red Baron un navigateur X-Window conçu par Red Hat Software. Il est distribué avec la distribution officielle Red Hat Linux. Je n'ai pas pu avoir beaucoup plus d'informations sur lui, mais je sais qu'il supporte les frames, les formulaires et le SSL. Si vous utilisez Red Baron, aidez moi à remplir cette section. Pour plus d'informations visitez le site Red Hat à

`http://www.redhat.com` .

Chimera

Chimera est un navigateur X-Window basique. Il supporte quelques unes des spécificités du HTML 3.2. La dernière version est la 2.0 alpha 6 du 27 août 1997. Pour plus d'informations visitez le site de Chimera à `http://www.unlv.edu/chimera/` . Chimera peut être chargé sur

`ftp://ftp.cs.unlv.edu/pub/chimera-alpha/chimera-2.0a6.tar.gz` .

Qweb

Qweb est encore un autre navigateur X-Window basique. Il supporte les tableaux, les formulaires, et les cartes des images côté serveur. La dernière version est la 1.3. Pour plus d'informations visitez le site Qweb à

`http://sunsite.auc.dk/qweb/` . Les sources sont disponibles sur

`http://sunsite.auc.dk/qweb/qweb-1.3.tar.gz` . Les binaires sont disponibles en format Red Hat RPM à `http://sunsite.auc.dk/qweb/qweb-1.3-1.i386.rpm` .

Grail

Grail est un navigateur X-Window développé par la Corporation for National Research Initiatives (CNRI). Grail est entièrement écrit en Python, un langage interprété orienté objet. La dernière version est la 0.3 du 7 Mai 1997. Il supporte les formulaires, les marque-pages, l'historique, les frames, les tableaux, et beaucoup de choses du HTML 3.2.

Internet Explorer

Il y a des rumeurs, comme quoi Microsoft porterait Internet Explorer sur diverses plateformes Unix - peut-être Linux. Si c'est exact ils prennent le temps de le faire. Si vous connaissez quelque chose de plus sûr, envoyez moi un mail.

A mon humble avis, la plupart des logiciels ci-dessus sont inutilisables pour apprécier sérieusement le web. Je n'essaye pas de discréditer les auteurs, je sais qu'ils ont travaillé très dur sur ces projets. Imaginez simplement, si toutes ces personnes avaient travaillé en commun sur un seul projet, nous aurions peut-être un navigateur gratuit qui pourrait rivaliser avec Netscape et Internet Explorer.

A mon avis, sur tous ces navigateurs, Netscape et Lynx sont les meilleurs. Les suivants pourraient être Kfm, Emacs-W3 et Mosaic.

3 Lynx

Lynx est des plus petits (environ 600 ko d'exécutable) et des plus rapides des navigateurs disponibles. et probablement le browser web le plus rapide actuellement disponible. Il n'utilise pas autant de bande passante ni de ressources système car il affiche uniquement en mode texte. Il peut-être utilisé sur toute console, terminal ou xterm. Vous n'aurez pas besoin d'un *système X-Window* ni de mémoire supplémentaire pour faire tourner ce petit browser.

3.1 Où l'obtenir ?

Les deux distributions Red Hat et Slackware incluent Lynx. C'est pourquoi je ne vous ennuierai pas avec les détails sur la compilation et l'installation de Lynx.

La dernière version est la 2.8rel.2, et peut être obtenue sur `http://www.slcc.edu/lynx/fote/` ou sur tout serveur ftp miroir comme *ftp://ftp.lip6.fr dans /pub/linux/sunsite/apps/www/browsers/* .

Pour plus d'informations sur Lynx essayez l'une de ces url:

Lynx Links

`http://www.crl.com/~subir/lynx.html`

Lynx Pages

`http://lynx.browser.org`

Lynx Help Pages

`http://www.crl.com/~subir/lynx/lynx_help/lynx_help_main.html`

(les mêmes que celle données par lynx –help et ? dans lynx).

Note: les pages d'aides de Lynx ont récemment déménagé. Si vous avez une version ancienne de Lynx, vous devrez changer votre lynx.cfg (dans /usr/lib) pour pointer sur la nouvelle adresse (plus haut).

Je pense que la spécificité la plus importante de Lynx par rapport à tous les autres navigateurs web est sa capacité de récupération automatique des fichiers.N'importe qui peut écrire un script qui récupérera tout document, fichier ou n'importe quoi d'autre via des url *http*, *FTP*, *gopher*, *WAIS*, *NNTP* ou *file://* - et les sauver sur le disque. De plus, toute personne peut entrer des données dans les formulaires HTML en mode batch en redirigeant simplement l'entrée standard et en utilisant l'option *-post_ data*.

Pour en savoir plus les spécificités exceptionnelles de Lynx regardez juste les fichiers d'aide et les pages de man. Si vous utilisez une option spéciale de Lynx que vous désireriez voir ajouter à ce document, faîtes le moi savoir.

4 Emacs-W3

Il y a différentes versions d'Emacs. Les deux les plus populaires sont GNU Emacs et XEmacs. GNU Emacs vient de la Free Software Foundation, et est l'Emacs original. Il est principalement orienté vers les terminaux en mode texte, mais il peut tourner sous X-Window. XEmacs (au départ par Lucid Emacs) est une version qui tourne uniquement sous X-Window. Il dispose de nombreuses spécificités de type X-Window (de meilleurs menus, ...).

4.1 Où l'obtenir ?

Les deux distributions Red Hat et Slackware incluent GNU Emacs.

Le GNU emacs le plus récent est le 19.34. Il ne semble pas avoir de site web. Le site FTP original est à `ftp://ftp.gnu.ai.mit.edu/pub/gnu/` ; il y a un mirroir pour la France sur `ftp://ftp.lip6.fr/pug/gnu/` .

La dernière version de XEmacs est la 20.2. Le site FTP de XEmacs est à `ftp://ftp.xemacs.org/pub/xemacs` . Pour plus d'informations sur XEmacs allez voir leurs pages web à `http://www.xemacs.org` .

Les deux sont disponibles sur tous mirroir des archives Linux, par exemple à *ftp://ftp.lip6.fr/pub/linux/sunsite/apps/editors/emacs/*

5 Netscape Navigator/Communicator

5.1 Des versions et des options différentes.

Netscape Navigator est le roi des navigateurs WWW. Netscape Navigator peut quasiment tout faire. Mais d'un autre côté, c'est un des programmes les plus gourmand en mémoire et en ressources que j'ai pu voir.

Il y a 3 versions différentes de ce programme:

Netscape Navigator contient le navigateur web, netcaster (le client push) et un programme basique de courrier.

Netscape Communicator contient le navigateur web, un éditeur web, un programme avancé de courrier, un lecteur de news, netcaster (le client push), et un utilitaire pour les conférences de groupes.

Netscape Communicator Pro contient tout ce qu'a Communicator plus un calendrier groupé, l'émulation des terminaux IBM, et des options pour la gestion à distance (les administrateurs peuvent mettre à jour des milliers de copies de Netscape à partir de leur siège).

En addition à ces trois versions il y a deux autres options que vous pouvez choisir.

La première est l'installation complète ou basique. L'installation complète vous installe tout. L'installation de base vous en donne suffisamment pour commencer. Vous pouvez charger les composants supplémentaires lorsque vous en avez besoin (comme le support multimedia et netscaster). Ces composants peuvent être installés par l'utilitaire 'smart update' de Netscape (après l'installation allez dans 'help->software updates'). Actuellement, l'installation complète n'est pas disponible pour Linux.

La seconde option est l'import ou l'export. Si vous habitez les US ou le Canada vous pouvez choisir la version d'import. Celà vous donnera l'encryption plus solide de 128 bits pour les transactions sécurisées (SSL). La version d'export dispose seulement d'une encryption 40 bits, et est la seule version autorisée en dehors des US et du Canada.

La dernière version de Netscape Navigator/Communicator/Communicator Pro est la 4.03. Il y a deux versions différentes pour Linux, une pour l'ancienne série des noyaux 1.2 et une pour les nouveaux 2.0. Si vous n'avez pas de noyau 2.0 je vous suggère de le mettre à jour; il y a de nombreuses nouveautés dans le nouveau noyau.

Des beta version sont également disponibles. Si vous essayez une version beta, elles expireront généralement plus ou moins un mois après !

5.2 Où l'obtenir ?

Le meilleur moyen pour obtenir des logiciels de Netscape est d'aller le chercher par leur site web à `http://www.netscape.com/download/` . Ils ont des menus pour vous guider dans votre sélection. Lorsqu'on demande une version Linux, on référence par rapport au noyau (la plupart des gens doivent maintenant utiliser le 2.0). Si vous n'êtes pas sûr de la version du noyau que vous avez vous pouvez lancer 'cat /proc/version'. Chercher sur le site web est le seul moyen d'obtenir les versions d'import.

Si vous désirez une version export vous pouvez la charger directement des serveurs FTP de Netscape. Leurs serveurs FTP sont également plus à jour. Par exemple la première fois que j'ai écrit ceci, l'interface web n'avait pas encore la version non-beta 4.03 pour Linux, mais elle était sur le site FTP. Voici les liens vers les versions d'export pour Linux 2.0:

Netscape Navigator 4.03 est à `ftp://ftp.netscape.com/pub/communicator/4.03/shipping/english/unix/linux20/navigator_standalone/navigator-v403-export.x86-unknown-linux2.0.tar.gz`

Netscape Communicator 4.03 pour Linux 2.0 (noyau) est à `ftp://ftp.netscape.com/pub/`

communicator/4.03/shipping/english/unix/linux20/base_install/communicator-v403-export.
x86-unknown-linux2.0.tar.gz

Communicator Pro 4.03 pour Linux n'était pas disponible au moment où j'écris ces lignes.

Ces url changeront lorsque de nouvelles versions sortiront. Si ces liens ne fonctionnent pas vous pouvez les trouver en pêchant sur le site FTP ftp://ftp.netscape.com/pub/communicator/ .

Ces serveurs sont souvent très chargés. Il vaut mieux attendre hors des heures de bureau ou choisir un site mirroir. Soyez préparé à attendre, ces archives sont grosses. Navigator pèse environ 8 megs, et l'installation de base de Communicator est de 10 megs.

5.3 Installation

Cette section explique comment installer la version 4 de Netscape Navigator, Communicator, et Communicator Pro.

Tout d'abord décompactez l'archive dans un répertoire temporaire. Lancez ensuite le script `ns-install` (tapez `./ns-install`). Créez ensuite un lien symbolique du binaire `/usr/local/netscape/netscape` sur `/usr/local/bin/netscape` (tapez `ln -s /usr/local/netscape/netscape /usr/local/bin/netscape`). Enfin, attribuez à une variable d'environnement `$MOZILLA_HOME` pour tout le système la valeur `/usr/local/netscape`, de façon à ce que Netscape puisse trouvez ses fichiers. Si vous utilisez bash en tant que shell, éditez votre `/etc/profile` et ajoutez les lignes:

```
export MOZILLA_HOME="/usr/local/netscape"
```

Après l'avoir installé le logiciel peut se mettre automatiquement à jour avec smart update. Lancez juste Netscape en tant que super-utilisateur et allezz à help->software updates. Si vous avez simplement pris l'installation basique, vous pouvez également installer les composants Netscape de là.

Note: Ceci ne supprimera pas vos anciennes versions de Netscape, vous devez les supprimer manuellement en supprimant le binaire Netscape et le fichier des classes Java (pour la version 3).

6 Lancer un server WWW

Cette section contient des informations sur les différents serveurs http et les outils additionnels comme les langages de scripts pour les programmes CGI, etc. Il y a plusieurs douzaines de serveurs web, j'ai seulement couvert ceux qui sont pleinement fonctionnels. Comme certains sont des programmes commerciaux, je ne peux pas les tester. La plupart des informations de la section de présentation ont été récupérées sur divers sites web. S'il y a des informations incorrectes ou manquantes, veuillez me le faire savoir.

Pour une documentation plus technique des mécanismes du http, voyez les RFCs mentionnées dans le chapitre "Documents plus avancés" de ce HOWTO.

Je préfère utiliser le serveur Apache. Il a la plupart des options que vous avez jamais désiré et il est gratuit ! J'admets que cette section est fortement orientée vers Apache. J'ai décidé de concentrer mes efforts sur la section Apache plutôt que le négliger par rapport à tous les autres serveurs web. Je pourrais couvrir d'autres serveurs web dans le futur.

6.1 Survol

Cern httpd

Ce fut le premier serveur web. Il a été développé par le European Laboratory for Particle Physics (CERN). Le CERN httpd n'est plus supporté. Le serveur CERN httpd est connu pour avoir quelques bugs étranges, pour être lent et mangeur de ressources. La dernière version est la 3.0. Pour plus d'informations visitez la page mère du CERN httpd à `http://www.w3.org/Daemon/Status.html` . Il peut être chargé sur `ftp://sunsite.unc.edu/pub/Linux/apps/www/servers/httpd-3.0.term.tpz`

(non, ce n'est pas une erreur, l'extension est actuellement .tpz sur le site; ce devrait probablement être .tgz).

NCSA HTTPd

Le serveur NCSA HTTPd est le père d'Apache (le développement a donné naissance à deux serveurs différents). Toutefois les fichiers de configuration sont très similaires. le NCSA HTTPd est gratuit et le code source est disponible. Ce serveur n'est pas couvert par ce document, cependant la lecture de la section Apache peut vous donner quelques tuyaux. Le serveur NCSA fût populaire, mais la plupart des gens l'ont remplacé par Apache. Apache est un essai de remplacement du serveur NCSA (même fichiers de configuration), et il fixe plusieurs limites du serveur NCSA. Le serveur NCSA HTTPd compte pour 4.9% (en chute libre) de tous les serveurs web (source Septembre 1997, *Netcraft survey* `http://www.netcraft.com/survey/`). La dernière version est la 1.5.2a. Pour plus d'informations voyez le site du NCSA à `http://hoohoo.ncsa.uiuc.edu` .

7 (Apache)

Apache est le roi de tous les serveurs web. Apache et ses sources sont gratuits. Apache est modulaire, aussi il est facile d'y ajouter des caractéristiques. Apache est très flexible et dispose de très, très nombreuses caractéristiques. Apache et ses dérivés réalisent 44% de tous les domaines web (50% su vous comptez tous les dérivés). Il y a plus de 695.000 serveurs Apache actifs (source Septembre 1997, *Netcraft survey* `http://www.netcraft.com/survey/`).

La version officielle d'Apache ne contient pas le SSL, mais il y a deux dérivés qui l'incluent. Stronghold est un produit commercial qui est basé sur Apache. Il est vendu $995; une version plus économique est disponible pour $495 (basée sur une vieille version d'Apache). Stronghold est le numéro deux des serveurs sécurisés derrière Netscape (source

C2 net `http://www.c2.net/products/stronghold` et *Netcraft survey* `http://www.netcraft.com/survey/`). Pour plus d'informations voyez le site de Stronghold à `http://www.c2.net/products/stronghold/` . Il a été développé hors des US, il est donc disponible avec du SSL en 128 bits partout.

Apache-SSL est une implémentation gratuite de SSL, mais pas pour une utilisation commerciale dans les US (RSA a une license US sur la technologie SSL). Il peut être utilisé pour une utilisation non-commerciale aux US si vous le reliez avec la librairie gratuite RSAREF. Pour plus d'informations voyez le site à `http://www.algroup.co.uk/Apache-SSL/` .

Netscape Fast Track Server

Fast Track a été développé par Netscape, mais la version Linux est mise sur le marché par Caldera. Le site de Caldera le liste en tant que "Fast Track for OpenLinux". Je ne suis pas sûr qu'il tourne seulement sous Caldera OpenLinux ou si toute distribution Linux peut le faire (écrivez moi si vous connaissez la réponse). Les serveurs Netscape comptent pour 11.5% (en chute libre) de tous les serveurs web (source Septembre 1997 `http://www.netcraft.com/survey/`). Le serveur est vendu $295. Il est également inclus avec la distribution Caldera OpenLinux Standard qui est vendue $399 (version éducation: $199.50). Les pages web parlent d'une interface d'administration simple et pratique et d'une configuration rapide en 10 minutes. Le serveur supporte le SSL en 40 bits. Pour obtenir les 128 bits SSL vous aurez besoin du Netscape Enterprise Server. Malheureusement il n'est pas disponible pour Linux :(La dernière version disponible pour Linux est la 2.0 (la version 3 est en beta, mais elle n'est pas encore disponible pour Linux). Pour l'acheter, allez sur le site web de Caldera à http://www.caldera.com/products/netscape/netscape.html . Pour plus d'informations voyez la page Fast Track à http://www.netscape.com/comprod/server_central/product/fast_track/

WN

WN dispose de nombreuses caractéristiques qui le rende attractif. Tout d'abord il est plus petit que les serveurs CERN, NCSA HTTPd, Apache. Il dispose également de nombreuses options intégrées qui nécessiteraient sinon des CGI. Par exemple la recherche sur le site, des intégrés du côté du serveur. Il peut également décompresser/compresser des fichiers en transparence avec son option de filtrage. Il peut également récupérer une partie seulement d'un fichier avec son option d'échelle. Il est distribué sous licence publique GNU. La version actuelle est la 1.18.3. Pour plus d'informations voyez le site de WN à http://hopf.math.nwu.edu/ .

AOLserver

AOLserver est développé par America Online. Je dois admettre que j'ai été surptis par les options offertes par un serveur web venant de chez AOL. En addition aux options standard, il supporte la connectivité des bases de données. Les pages peuvent interroger une base de données par les commandes Structured Query Language (SQL). La base de données est accessible au travers du Open Database Connectivity (ODBC). Il dispose également d'un moteur de recherche et des scripts TCL. Si celà ne vous suffit pas, vous pouvez ajouter votre module par la Application Programming Interface (API), en C. J'ai même oublié de mentionner le support pour 40 bits SSL. Et vous obtenez tout ceci gratuitement ! Pour plus d'informations voyez le site du AOLserver à http://www.aolserver.com/server/ .

Zeus Server

Zeus Server a été développé par Zeus Technology. Ils se réclament comme ayant le serveur web le plus rapide (d'après la batterie de tests WebSpec96). Le serveur peut être configuré et contrôlé par un navigateur web ! Cela limite l'encombrement du processeur et de la mémoire pour les scripts CGI, et il s'exécute dans un environnement sécurisé (quelle que soit la signification de cette expression). Il supporte également les serveurs virtuels sans limitation. Il est vendu $999 pour la version standard. Si vous désirez le serveur sécurisé (SSL) le prix grimpe à $1699. Ils sont basé hors des US, la technologie 128 bits SSL est donc disponible partout. Pour plus d'informations voyez le site de Zeus Technology à http://www.zeus.co.uk . Le site web US se trouve sur

http://www.zeus.com . Je vous préviens qu'ils sont trop sûrs d'eux à propos de leur serveur le plus rapide. Mais ils n'apparaissent même pas dans le top des serveurs web de la Netcraft Surveys.

CL-HTTP

CL-HTTP est l'abrégé de Common Lisp Hypermedia Server. Si vous êtes un programmeur Lisp ce serveur est pour vous. Vous pouvez écrire vos scripts CGI en Lisp. Il a une fonction de configuration basée sur le web. Il supporte également toutes les options standards des serveurs. CL-HTTP est gratuit et le code source est disponible. Pour plus d'informations voyez le site web de CL-HTTP à http://www.ai.mit.edu/projects/iiip/doc/cl-http/home-page.html (ne pouvaient-ils rendre cette url plus longue ?).

Si vous avez un dessein commercial (site web d'une compagnie, ou ISP), je vous recommanderais fortement l'utilisation d'Apache. Si vous regardez plus à une configuration simple qu'à des options avancés, alors le Zeus Server est fait pour vous. J'ai également entendu dire que le Netscape Server est facile à configurer. Si vous avez une utilisation interne, vous pouvez être un peu plus flexible. Mais à moins que l'un d'entre eux dispose d'une option que vous devez utiliser, je vous recommanderai néanmoins l'utilisation de l'un des trois pré-cités.

Il s'agit seulement d'une liste partielle de tous les serveurs disponibles. Pour une liste plus complète visitez Netcraft à http://www.netcraft.com/survey/servers.html ou Web Compare à http://webcompare.internet.com .

7 Apache

La version actuelle d'Apache est la 1.2.4. La version 1.3 est en test. Le site principal d'Apache est à `http://www.apache.org/` . Une autre bonne source d'information est Apacheweek à `http://www.apacheweek.com/` . La documentation d'Apache est bonne, je ne rentrerai donc pas dans les détails de la configuration d'apache. La documentaion est sur le site web et également incluse dans les sources (au format HTML). Il y a également des fichiers textes inclus dans les sources, mais la version HTML est meilleure. La documentation doit devenir bien meilleure depuis que le Apache Documentation Project a été lancé. Pour l'instant la plupart des documents sont écrits par les développeurs. Sans vouloir discréditer les développeurs, ils sont un peu difficiles à comprendre si vous ne connaissez pas la terminologie.

7.1 Où l'obtenir ?

Apache est inclus dans les distributions Red Hat, Slackware, et OpenLinux. Cependant il se peut que ce ne soit pas la dernière version, ce sont des binaires très fiables. La mauvaise nouvelle est que vous aurez à vivre avec leurs choix de répertoires (qui sont totalement différents les uns des autres et de ceux par défaut d'Apache).

Le source est disponible sur le site web d'Apache à `http://www.apache.org/dist/` . Les binaires sont également disponibles au même endroit. Vous pouvez aussi obtenir les binaires de sunsite à `ftp://sunsite.unc.edu/pub/Linux/apps/www/servers/` et pour la France sur `ftp://ftp.lip6.fr/pub/linux/sunsite/apps/www/servers/` . Et pour ceux d'entre vous qui utilisent une Red Hat le dernier fichier RPM se trouve généralement dans le répertoire des contributions à `ftp://ftp.redhat.com/pub/contrib/i386/` .

Si votre serveur doit desservir un dessein commercial, il est hautement recommandé que vous obteniez les sources à partir du site web d'Apache et de le compiler vous même. L'autre option est d'utiliser un binaire qui provient d'une distribution majeure, par exemple Slackware, Red Hat, ou OpenLinux. La principale raison en est la sécurité. Un binaire inconnu peut avoir une "sortie cachée" pour les hackers, ou une correction instable peut crasher votre système. Ceci vous donnera également plus de contrôle sur les modules compilés, et vous autorisera à changer les répertoires par défaut. Il n'est pas difficile de compiler Apache, et vous ne serez pas un réel utilisateur de Linux tant que vous ne compilerez pas vos programmes ;)

7.2 Compiler et Installer

Tout d'abord décompactez l'archive dans un répertoire temporaire. Ensuite placez vous dans le répertoire src. Editez alors le fichier Configuration si vous désirez ajouter des modules spéciaux. Les modules les plus communément utilisés sont déjà inclus. Il n'y a pas besoin de changer les règles ou le makefile pour Linux. Lancez ensuite le script shell Configure (`./Configure`). Vérifiez qu'il vous dit que vous utilisez une plateforme Linux et gcc en tant que compilateur. Ensuite vous pouvez éditer le fichier httpd.h pour changer les répertoires par défaut. L'emplacement du serveur (où sont conservés les fichiers de configuration) par défaut est `/usr/local/etc/httpd/`, mais vous pouvez le changer pour juste `/etc/httpd/`. Et le répertoire principal du serveur (où sont conservées les pages HTML) par défaut est `/usr/local/etc/httpd/htdocs/`, mais j'apprécie le répertoire `/home/httpd/html` (le défaut de Red Hat pour Apache). Si vous devez utiliser su-exec (voir les options spéciales plus bas) vous pouvez également désirer changer le répertoire. Le répertoire principal peut également être changé à partir des fichiers de config. Mais il est également bon de le compiler, juste au cas où Apache ne puisse trouvez ou lire les fichiers de configuration. Tout le reste doit être changé à partir des fichiers de configuration. Lancez enfin make pour compiler Apache.

Si vous avez des problèmes au sujet de fichiers include manquants, vérifiez les points suivants. Vérifiez que vous avez les entêtes du noyau (fichiers include) installés pour votre version du noyau. Vérifiez également que vous avez ces liens symboliques installés:

```
/usr/include/linux doit être un lien sur /usr/src/linux/include/linux
/usr/include/asm doit être un lien sur /usr/src/linux/include/asm
/usr/src/linux doit être un lien sur les sources de Linux (ex.linux-2.0.30)
```

Les liens peuvent être créés avec `ln -s`, celà fonctionnera comme la commande cp à part qu'il fera un lien ((`ln -s SOURCE DESTINATION`).

Lorsque make a terminé il doit y avoir un exécutable nommé httpd dans le répertoire. Il est nécessaire de le déplacer dans un répertoire bin. `/usr/sbin` ou `/usr/local/sbin` seraient de bons choix.

Copiez les sous-répertoires conf, logs, et icons des sources vers l'emplacement du serveur. Renommez ensuite trois des fichiers du sous-répertoire conf pour vous débarasser de l'extension `-dist` ((ex. `httpd.conf-dist` devient `httpd.conf`).

Il y a aussi divers programmes de support qui sont inclus avec Apache. Ils sont dans le répertoire `support` et doivent être compilés et installés séparément/ La plupart d'entre eux peuvent être créés en utilisant le makefile de leur répertoire (ce qui est fait lorsque vous lancez le script principal `Configure`). Vous n'avez besoin d'aucun d'entre eux pour utiliser Apache, mais certains rendent le travail des administrateurs plus simple.

7.3 Configurer

Maintenant vous devez avoir quatre fichiers dans votre sous-répertoire `conf` de l'emplacement du serveur. Le fichier `httpd.conf` met en place le daemon du serveur (numéro de port, utilisateur, etc). Le `srm.conf` donne l'arborescence pour les documents principaux, les actions spéciales, etc. Le `access.conf` donne les cas basiques pour l'accès. Finalement, le `mime.types` dit au serveur que type mime il doit envoyer au navigateur pour quelle extension.

Les fichiers de configuration sont assez bien documentés (ils sont pleins de commentaires), tant que vous comprenez le langage. Vous devriez les lire attentivement avant de mettre votre serveur en marche. Chaque option de configuration est couvert dans la documentation Apache.

Le fichier `mime.types` n'est pas réellement un fichier de configuration. Il est utilisé par le serveur pour traduire les extensions des fichiers en types mime à envoyer au navigateur. La plupart des types mime communs sont déjà dans le fichier. La majorité des gens ne devrait pas éditer ce fichier. Au cours du temps, plus de types mime seront ajoutés pour supporter de nouveaux programmes. La meilleure chose à faire est de prendre un nouveau fichier mime.types (et peut-être une nouvelle version du serveur) à ce moment là.

Souvenez vous toujours que lorsque vous changez les fichiers de configuration vous devrez relancer Apache ou lui envoyer le signal SIGHUP avec `kill` pour que les changements prennent effet. Vérifiez que vous envoyez le signal au process parent et non aux process enfants. Le parent a généralement le chiffre id le plus faible. L'id du process du parent est également dans le fichier `httpd.pid` du répertoire log. Si vous envoyez accidentellement le signal à un des process enfants, le process stoppera et le parent le relancera.

Je ne vous conduirai pas le long des chemins de la configuration d'Apache. A la place je parlerai des problèmes spécifiques, des choix à faire, et des options spéciales.

Je recommande chaudement que tous les utilisateurs lisent les trucs sur la sécurité dans la documentation Apache. Elle est également disponible sur le site web d'Apache à `http://www.apache.org/docs/mics/security_tips.html` .

7.4 Héberger des serveurs web virtuels

L'hébergement virtuel existe lorsqu'un seul ordinateur dispose de plus d'un nom de domaine. L'ancienne méthode était d'avoir une adresse IP pour chaque site virtuel. La nouvelle méthode utilise uniquement une adresse IP, mais celà ne fonctionne pas correctement avec les navigateurs qui ne supportent pas le HTTP 1.1.

Ma recommandation pour le commerce est de conserver un hébergement virtuel basé sur plusieurs IP jusqu'à ce que la majorité des gens disposent de navigateurs supportant HTTP 1.1 (donnez leur un an ou deux). Ceci vous donne également une illusion plus complète de l'hébergement virtuel. Alors que les deux méthodes peuvent vous donner des possibilités de courrier virtuel (est ce que quelqu'un peut confirmer?), seul l'hébergement virtuel basé IP peut également vous donner un serveur FTP virtuel.

S'il s'agit d'un club ou d'une page personnelle, vous pouvez considérer l'hébergement virtuel par IP liée. Ce devrait être moins cher que l'hébergement basé IP et vous sauverez de précieuses addresses IP.

Vous pouvez également mélanger les hébergements virtuels par IP fixes et par IP partagées sur le même serveur. Pour plus d'information sur l'hébergement virtuel voyez Apacheweek sur `http://www.apacheweek.com/features/vhost` .

7.4.1 Hébergement virtuel basé IP

Pour cette méthode chaque site virtuel dispose de sa propre adresse IP. En déterminant l'adresse IP à laquelle la requête était envoyée, Apache et d'autres programmes peuvent dire quel domaine desservir. C'est un incroyable gain d'adresses IP. Prenez par exemple les serveurs où est conservé mon domaine virtuel. Ils ont plus de 35.000 comptes virtuels, ce qui signifie 35.000 adresses IP. Pourtant je crois qu'aux derniers comptes ils avaient moins de 50 serveurs réels.

La configuration se fait en deux parties. La première consiste à obtenir de Linux l'acceptation de plus d'une adresse IP. La seconde est de configurer apache pour servir les sites virtuels.

La première étape dans la configuration pour l'acceptation par Linux de multiples adresses IP est de créer un nouveau noyau. Ceci fonctionne mieux avec la série des noyaux 2.0 (ou supérieure). Vous devrez inclure le support pour "IP networking" et "IP aliasing". Si vous avez besoin d'aide pour la compilation du noyau voyez le *kernel howto* `http://sunsite.unc.edu/LDP/HOWTO/Kernel-HOWTO.html` , ou sa version française sur `http://www.freenix.org/unix/linux/HOWTO/Kernel-HOWTO.html` .

Vous devrez ensuite configurer chaque interface au lancement. Si vous utilisez la distribution Red Hat akirs ceci peut être fait à partir du panneau de contrôle. Lancez X-Window en tant que super-utilisateur, vous devriez voir un panneau de contrôle. Double-cliquez sur "network configuration" (configuration réseau). Allez ensuite sur le panneau des interfaces et choisissez votre carte réseau. Ensuite cliquez sur alias en bas de l'écran. Rentrez les informations et cliquez sur "done". Ceci devra être fait pour chaque site virtuel / adresse IP.

Si vous utilisez une autre distribution vous pouvez avoir à le faire manuellement. Vous pouvez simplement mettre les commandes dans le fichier `rc.local` de `/etc/rc.d` (en réalité vous devriez les mettre avec toutes celles concernant le réseau). Vous devrez avoir une commande `ifconfig` et `route` pour chaque périphérique. Les adresses en alias sont divisées sous le périphérique principal. Par exemple eth0 aurait les alias eth0:0, eth0:1, eth0:2, etc. Voici un exemple de configuration d'un alias:

```
ifconfig eth0:0 192.168.1.57
route add -host 192.168.1.57 dev eth0:0
```

Vous pouvez également ajouter une adresse de broadcast et un netmask à la commande ifconfig. Si vous avez beaucoup d'alias vous pouvez vouloir faire une boucle pour le rendre plus simple. Pour plus d'informations voyez le

IP alias mini howto `http://sunsite.unc.edu/LDP/HOWTO/mini/IP-Alias.html` ou sa version française à `http://www.freenix.org/unix/linux/HOWTO/mini/IP-Alias.html` .

Vous devrez ensuite configurer votre domain name server (DNS) pour desservir ces nouveaux domaines. Et si vous ne possédez pas déjà les noms de domaine, vous devrez contacter l' *Internic* `http://www.internic.net` pour enregistrer les noms de domaines. Voyez le DNS-howto pour plus d'informations sur la configuration de votre DNS.

Dernièrement vous devrez configurer Apache de manière à desservir correctement le domaine virtuel. Ceci se fait dans le fichier de configuration `httpd.conf` près de la fin. Ils vous donnent un exemple sur la façon de procéder. Toutes les commandes spécifiques au site virtuel sont placés entre les marqueurs `virtualhost`. Vous pouvez mettre à peu près n'importe quelle commande ici. Généralement vous devrez placer différents répertoires pour le serveur, le répertoire script, et les fichiers de log. Vous pouvez avoir un nombre quasi-illimité de sites virtuels en ajoutant plus de marqueurs `virtualhost`.

Dans de rares cas vous pouvez avoir à lancer des serveurs séparés si une directive est nécessaire pour un site virtuel, mais n'est pas accepté dans les marqueurs du site virtuel. Ceci se fait en utilisant la directive bindaddress. Chaque serveur aura un nom des fichiers de configuration différents. Chaque serveur répondra seulement à une seule adresse IP, spécifiée par la directive bindaddress. C'est un gain incroyable de ressources système.

7.4.2 Hébergement virtuel par IP partagée

C'est une nouvelle méthode pour l'hébergement virtuel. Elle utilise une adresse IP simple, tout en conservant les adresses IP pour les vrais machines (et pas les virtuelles). Dans le même exemple utilisé plus haut ces 30.000 sites virtuels utiliseraient seulement 50 adresses IP (une pour chaque machine). Ceci est fait en utilisant le nouveau protocole HTTP 1.1. Le navigateur dit au serveur quel site il désire lorsqu'il envoie la demande. Le problème est que les navigateurs qui ne supportent pas HTTP 1.1 obtiendront la page principale des serveurs, qui peut être configurée pour donner la liste des sites virtuels disponibles. Ceci ruine la totale illusion de l'hébergement virtuel. L'illusion que vous avez votre propre serveur.

La configuration est bien plus simple que pour l'hébergement virtuel basé sur IP. Vous aurez toujours besoin d'obtenir votre domaine à partir de l'Internic et de configurer votre DNS. Cette fois-ci le DNS pointe sur la même adresse IP que le domaine original. Ensuite Apache se configure comme précédemment. Puisque vous utilisez la même adresse IP dans les marqueurs virtualhost, Apache sait que vous désirez l'hébergement virtuel par IP partagée.

Il y a différentes choses à faire pour les anciens navigateurs. Je vais vous expliquer la meilleure. Tout d'abord vous devrez créer vos pages principales pour un serveur virtuel (soit basé IP, soit par IP partagée). Ceci libère la page principale pour un lien listant tous vos sites virtuels. Ensuite vous devrez créer une sortie cachée pour les anciens navigateurs. On le réalise en utilisant la directive `ServerPath` pour chaque site virtuel de la directive `virtualhost`. Par exemple en ajoutant `ServerPath /monsite/` à www.monsite.com, les anciens navigateurs pourront accéder au site par www.monsite.com/monsite/. Ensuite mettez dans la page par défaut du serveur principal un message qui les incitera poliment à obtenir un nouveau navigateur, et listera les liens sur toutes les sorties cachées des sites qye vous hébergez sur cette machine. Lorsqu'un ancien navigateur accèdera au site ils seront renvoyés à la page principale, et auront un lien sur la page correcte. Les nouveaux navigateurs ne verront jamais la page principale et iront directement sur les sites virtuels. Vous devez vous rappeler de conservez tous vos liens relatifs entre les sites web, car les pages seront demandées à partir de deux URL différentes (www.monsite.com et www.monsite.com/monsite/).

J'espère que je ne vous ai pas perdu ici, mais ce n'est pas d'une compréhension simple. Après tout, vous devriez peut-être considérer l'hébergement basé IP. Une explication très similaire se trouve sur le site web d'apache à `http://www.apache.org/manual/host.html` .

Si quelqu'un dispose d'un pointeur sympathique pour l'hébergement par IP partagée, j'aimerais le connaître. Il serait agréable de connaître le pourcentage de navigateurs qui supportent le HTTP 1.1, et d'avoir une liste des navigateurs et des versions qui supportent HTTP 1.1.

7.5 Scripts CGI

Il existe deux méthodes différentes pour donner à vos utilisateurs la possibilité d'utiliser des scripts CGI. La première est de déclarer tout fichier se terminant par `.cgi` comme script CGI. La seconde est de créer des répertoires de scripts (généralement nommés `cgi-bin`). Vous pouvez également utiliser les deux méthodes. Quelque soit la méthode utilisée vos scripts doivent être exécutable par n'importe qui (`chmod 711`). En donnant à vos utilisateurs l'accès aux scripts vous créez un gros risque de sécurité. Faîtes proprement votre travail afin de minimiser les risques concernant la sécurité.

Je préfère la première méthode, spécialement pour les scripts complexes. Ceci vous autorisera à mettre les scripts dans n'importe quel répertoire. J'aime mettre mes scripts au même endroit que les pages web qui s'en servent. Pour les sites avec beaucoup de script, ceci est beaucoup mieux que d'avoir un répertoire plein de scripts. La configuration est simple. Tout d'abord supprimez le commentaire du marqueur `.cgi` à la fin du fichier `srm.conf`. Ensuite vérifiez que tous vos répertoires ont les marqueurs `option ExecCGI` ou `All` dans le fichier `access.conf`.

Créer un répertoire de scripts est considéré comme plus sûr. Pour créer un répertoire de scripts vous utilisez la directive ScriptAlias dans le fichier `srm.conf`. Le premier argument est l'Alias, et le second le répertoire réel. Par exemple `ScriptAlias /cgi-bin/ /usr/httpd/cgi-bin/` rend le répertoire `/usr/httpd/cgi-bin` capable d'exécuter les scripts. Ce répertoire sera utilisé même si quelqu'un demande le répertoire `/cgi-bin/`. Pour des raisons de sécurité vous devez également changer les propriétés du répertoire pour `Options none, AllowOveride none` dans le fichier `access.conf` (supprimer simplement les commentaires de l'exemple donné). Egalement ne placez pas vos répertoires de scripts en tant que sous répertoire de vos répertoires de pages web. Par exemple si vous servez les pages à partir de `/home/httpd/html/`, ne créez pas le répertoire de scripts en tant que `/home/httpd/html/cgi-bin`; au lieu de ça mettez le dans `/home/httpd/cgi-bin`.

Si vous désirez que vos utilisateurs disposent de leurs propres répertoires de scripts vous pouvez utiliser de multiples commandes `ScriptAlias`. Les sites virtuels doivent avoir cette commande `ScriptAlias` dans leurs directives `virtualhost`. Est ce que quelqu'un connaît un moyen simple pour autoriser les utilisateurs à avoir leur propre répertoire cgi-bin sans utiliser des commandes individuelles ScriptAlias ?

7.6 Répertoires Web des Utilisateurs

Il y a deux différentes méthodes pour s'occuper des répertoires web des utilisateurs. La première est d'avoir un sous-répertoire dans les répertoires des utilisateurs (généralement `public_html`). La seconde est d'avoir une aborescence entièrement différente pour les répertoires web. Pour ces deux méthodes vérifiez les options d'accès aux répertoires dans le fichier `access.conf`.

La première méthode est configurée par défaut dans apache. Lorsqu'une demande pour `/~bob/` arrive, Apache regarde dans le répertoire `public_html` du répertoire principal de bob. Vous pouvez changer le répertoire avec la directive `UserDir` dans le fichier `srm.conf`. Ce répertoire doit lisible et exécutable par tout le monde. Cette méthode crée un risque pour la sécurité car Apache le répertoire principal eds utilisateurs doit être exécutable par toute personne afin qu'Apache puisse y accéder.

La seconde méthode est facile à configurer. Vous devez juste changer la directive `UserDir` dans le fichier `srm.conf`. Il y a beaucoup de formats différents; vous pouvez voir la documentation d'Apache pour clarification. Si vous que chaque utilisateur dispose de son propre répertoire sous `/home/httpd/`, vous utiliserez `UserDir /home/httpd`. Ensuite lorsqu'une demande arrivera pour `/~bob/` Apache la traduira pour `/home/httpd/bob/`. Ou si vous avez un sous-répertoire dans le répertoire de bob vous pouvez utiliser `UserDir /home/httpd/*/html`. Ceci traduira en `/home/httpd/bob/html/` et vous autorisera également à avoir un répertoire de script (par exemple `/home/httpd/bob/cgi-bin/`).

7.7 Mode démon contre mode Inetd

Il y a deux méthodes par lesquelles apache peut tourner. L'une est un démon qui tourne tout le temps (Apache appelle ceci standalone). La seconde est celle du super-serveur inetd.

Le mode démon est de loin supérieur au mode inetd. Apache est configuré pour le mode démon par défaut. La seule raison d'utiliser le mode d'inetd est pour les applications très peu utilisées, comme les tests de scripts en interne, l'Intranet d'une petite compagnie, etc. Le mode inetd économisera de la mémoire car apache ne sera chargé que lorsqu'il sera demandé. Seul le démon inetd restera en mémoire.

Si vous n'utilisez pas très souvent apache vous pouvez juste vouloir le conserver en mode démon et le lancer lorsque vous en avez besoin. Ensuite vous pouvez le supprimer lorsque vous avez terminé (soyez sûr de bien supprimer le processus parent et non pas un des enfants).

Pour configurer le mode inetd vous devrez éditer quelques fichiers. Tout d'abord `/etc/services` regardez si http est déjà présent. S'il n'y est pas alors ajoutez ceci:

```
http    80/tcp
```

Le placer juste après 79 (finger) serait un bon endroit. Ensuite vous devez éditer le fichier `/etc/inetd.conf` et ajouter la ligne pour Apache:

```
http    stream  tcp     nowait  root    /usr/sbin/httpd httpd
```

Changez le chemin si vous avez Apache à un autre endroit. et le second httpd n'est pas une erreur; le démon inetd en a besoin. Si vous n'utilisez généralement pas le démon inetd, vous pouvez vouloir commenter toutes les autres lignes du fichier afin de ne pas activer les autres services (FTP, finger, telnet, et beaucoup d'autres choses qui sont généralement lancées par ce démon).

Si le démon inetd est déjà lancé (`inetd`), alors vous devez lui envoyer le signal SIGHUP (par kill; voyez la page de manuel de kill pour plus d'infos) ou relancer l'ordinateur pour que les changements soient effectifs. Si vous n'avez pas lancé `inetd` alors vous pouvez le lancer manuellement. Vous devez également l'ajouter à vos fichiers d'initialisation afin qu'il soit chargé au lancement (le fichier `rc.local` serait un bon choix).

7.8 Autoriser les commandes put et delete

Les nouveaux outils de publication web supportent cette nouvelles méthodes d'envoi des pages web par http (à la place de FTP). Certains de ces produits ne supportent même plus le FTP! Apache le supporte, mais il manque d'un script pour se charger des requêtes. Le script peut être un gros trou de sécurité, soyez certain de ce que vous faîtes avant de tenter d'en écrire ou d'en installer un.

Si quelqu'un connaît un script qui fonctionne faîtes le moi savoir et j'inclurai l'adresse ici.

Pour plus d'informations voyez l'article d'Apacheweek à `http://www.apacheweek.com/features/put` .

7.9 Authentification de l'Utilisateur / Contrôle des Accès

Il s'agit de l'une de mes options préférées. Elle vous autorise à protéger un répertoire ou un fichier par un mot de passe sans utiliser de scripts CGI. Il vous autorise également à interdire ou à autoriser l'accès sur la base de l'adresse IP ou du nom de domainde du client. C'est une spécificité très intéressante pour laisser les crétins hors de votre messagerie et des vos livres d'or (vous avez l'IP ou le nom de domaine à partir de vos fichiers de log).

Pour autoriser l'authentification de l'utilisateur le répertoire doit avoir `AllowOverrides AuthConfig` dans le fichier `access.conf`. Pour autoriser le contrôle d'accès (par domaine ou adresse IP) AllowOverrides Limit doit être mis pour ce répertoire.

La configuration du répertoire oblige le placement d'un fichier `.htaccess` dans le répertoire. Pour l'authentification de l'utilisateur il est également utilisé avec un `.htpasswd` et optionnellement un fichier `.htgroup`. Ces fichiers peuvent être partagées pour de multiples fichiers `.htaccess` si vous le désirez.

Pour des raisons de sécurité je recommande que chacun utilise ces directives dans leur fichier access.conf:

```
<files ~ "/\.ht">
order deny,allow
deny from all
</files>
```

Si vous n'êtes pas l'administrateur du système vous pouvez également l'ajouter dans votre fichier .htaccess si AllowOverride Limit est configuré pour votre répertoire. Cette directive interdira à quiconque de regarder dans vos fichiers de contrôle d'accès (.htaccess, .htpasswd, etc).

Il y a de nombreux types de fichiers et d'options qui peuvent être utilisés avec le contrôle d'accès. Toutefois il n'est pas de la compétence de ce document de décrire ces fichiers. Pour les informations sur la configuration de l'authentification des utilisateurs voyez l'article d'Apacheweek à `http://www.apacheweek.com/features/userauth` ou les pages de la NCSA à

`http://hoohoo.ncsa.uiuc.edu/docs-1.5/tutorials/user.html` .

7.10 su-exec

su-exec lance les scripts CGI en tant qu'utilisateur du propriétaire. Normalement ils sont lancés en tant qu'utilisateur du serveur web (généralement nobody). Ceci autorise les utilisateurs à accéder à leurs propres fichiers CGI sans les rendre autorisés en écriture (un trou de sécurité). Mais si vous ne faîtes pas attention vous pouvez un trou encore plus gros en utilisant le code su-exec. Celui-ci effectue des contrôles de sécurité avant d'exécuter les scripts, mais si vous le configurez de manière incorrecte vous aurez un trou de sécurité.

Le code su-exec n'est pas pour les amateurs. Ne l'utilisez pas si vous ne savez pas ce que vous faîtes. Vous pouvez terminer par un gros problème de sécurité où vos utilisateurs peuvent obtenir des accès super-utilisateurs pour votre système. Ne modifiez pas le code quelqu'en soit la raison. Lisez attentivement toute la documentation. Le code su-exec est intentionnellement difficile à configurer afin d'éviter l'utilisation par des amateurs (tout doit être fait à la main, il n'y a pas de script d'installation).

Le code su-exec se trouve dans le répertoire `support` des sources. Tout d'abord vous devez éditer le fichier `suexec.h` pour votre système. Ensuite vous devez compiler le code su-exec avec cette commande:

```
gcc suexec.c -o suexec
```

Copiez ensuite l'exécutable suexec dans le répertoire approprié. Apache le place par défaut dans `/usr/local/etc/httpd/sbin/`. Ceci peut être changé en éditant le fichier `httpd.h` dans les sources d'Apache et en recompilant Apache. Apache regardera seulement dans ce répertoire, Il ne cherchera pas ailleurs. Ensuite le fichier doit être changé pour la possession par le super-utilisateur (`chown root suexec`) et les permissions doivent être changées (`chmod 4711 suexec`). Enfin relancez Apache, il doit vous donner un message sur la console indiquant que su-exec est utilisé.

Les scripts CGI doivent être mis en exécutable par tous comme d'habitude. Ils seront automatiquement lancés par le possesseur du script CGI. Si vous changez les permissions les scripts CGI ne fonctionneront pas. Si le répertoire ou le fichier est en écriture par tous ou par un groupe le script ne fonctionnera pas. Les scripts possédés par les utilisateurs système ne doivent pas être lancés (root, bin, etc.). Pour les autres conditions de sécurité qui doivent être remplies voyez la documentation de su-exec. Si vous avez des problèmes voyez le fichier de log de su-exec nommé `cgi.log`.

Su-exec ne fonctionne pas si vous lancez Apache par inetd, il fonctionne seulement en mode démon. Ceci sera fixé dans la prochaine version car il n'y aura pas de mode inetd. Si vous aimez jouer avec le code source, vous pouvez éditer le fichier http_main.c. Vous pouvez rire de la ligne où Apache annonce qu'il utilise le su-exec wrapper (ceci est faussement marqué sur l'avant de toute sortie).

Lisez attentivement la documentation d'Apache sur su-exec. Elle est inclue dans les sources et est disponible sur le site web d'Apache à `http://www.apache.org/docs/suexec.html` .

7.11 Imagemaps

Apache peut gérer des cartes d'images du côté du serveur. Ce que l'on appelle "Imagemaps" sont les images des pages web qui envoient les utilisateurs à divers emplacements, dépendant de l'endroit où ils cliquent. Pour utiliser les imagemaps vérifiez que le module imagemap est installé (c'est un des modules par défaut). Ensuite vous devez supprimer le commentaire de la ligne `.map` à la fin du fichier `srm.conf`. Maintenant tous les fichiers se terminant en `.map` seront des fichiers d'imagemap. Les fichiers imagemap délimitent différentes aires sur l'image renvoyant vers des liens séparés. Apache utilise des fichiers d'aires au format standard NCSA. Voici un exemple utilisant un fichier d'aire dns une page web:

```
<a href="/map/mapfile.map">
<img src="picture.gif" ISMAP>
</a>
```

Dans cette exemple `mapfile.map` est le fichier d'aires, et `picture.gif` est l'image cliquable.

Il y a de nombreux programmes qui peuvent générer des fichiers d'aires compatibles NCSA ou vous pouvez les créer vous-même. Pour une discussion plus détaillée sur les imagemaps et les fichiers d'aires voyez l'article d'Apacheweek à `http://www.apacheweek.com/features/imagemaps` .

7.12 SSI/XSSI

Les Server Side Includes (SSI) ajoutent un contenu dynamique à des pages web qui sinon seraient statiques. Les en-têtes sont ajoutés dans les pages web en tant que commentaires. Le serveur web les exécute ensuite et passe les résultats au serveur web. SSI peut ajouter des en-têtes et des notes de pieds aux documents, ajouter la date à laquelle le document a été modifié pour la dernière fois, exécuter une commande système ou un script CGI. avec le tout nouveau eXtended Server Side Includes (XSSI) vous pouvez faire bien plus. XSSI ajoute les variables et les instructions de contrôle du flux (if, else, etc). C'est quasiment comme avoir un langage de programmation avec lequel on peut travailler.

L'analyse syntaxique de tous les fichiers pour les commandes SSI utiliserait beaucoup de ressources système. Cependant vous devez distinguer les fichiers HTML normaux de ceux qui contiennent les commandes SSI. Ceci se fait généralement en changeant l'extension des fichiers HTML utilisant SSI. Généralement on utilise l'extension `.shtml`.

Pour faire fonctionner SSI/XSSI vérifiez tout d'abord que le module des en-têtes est installé. Editez ensuite `srm.conf` et supprimez les commentaires des directives `AddType` et `AddHandler` pour les fichiers `.shtml`. Finalement vous devez configurer `Options Includes` pout tous les répertoires où vous désirez lancer des fichiers SSI/XSSI. Ceci se fait dans le fichier `access.conf`. Maintenant tous les fichiers avec l'extension `.shtml` seront analysés pour les commandes SSI/XSSI.

Un autre moyen de faire fonctionner les en-têtes est d'utiliser la directive `XBitHack`. Si vous la mettez en marche il regardera si le fichier est exécutable par l'utilisateur. Si il l'est et que `Options Includes` est autorisé pour le répertoire, alors le fichier sera traité comme un fichier SSI. Ceci fonctionne seulement pour les fichiers dont le type mime est text/html (fichiers `.html .htm`). Ceci n'est pas la méthode préférée.

Il y a un risque pour la sécurité en autorisant SSI à exécuter des commandes systèmes et des scripts CGI. Toutefois il est possible de bloquer cette possibilité avec la directire `Option IncludesNOEXEC` au lieu de Option Includes dans le fichier `access.conf`. Toutes les autres commandes SSI fonctionneront encore.

Pour plus d'informations voyez la documentation d'Apache mod_includes qui se trouve dans les sources. Il est également disponible sur le site web à `http://www.apache.org/docs/mod/mod_include.html` .

Pour une discussion plus détaillée de l'implémentation SSI/XSSI voyez l'article d'Apacheweek à `http://www.apacheweek.com/features/ssi` .

Pour plus d'informations sur les commandes SSI voyez la documentation de la NCSA à `http://hoohoo.ncsa.uiuc.edu/docs/tutorials/includes.html` .

Pour plus d'informations sur les commandes XSSI allez sur `ftp://pageplus.com/pub/hsf/xssi/xssi-1.1.html` .

7.13 Système modulaire

Apache peut être étendu pour supporter quasiment tout avec les modules. Il y a beaucoup de modules qui existent déjà. Seul les modules d'intérêt général sont inclus avec Apache. Pour les liens vers les modules existants allez voir le

Apache Module Registry à

`http://www.zyzzyva.com/module_registry/` .

Pour les informations sur la programmation des modules voyez

`http://www.zyzzyva.com/module_registry/reference/`

8 Ajouts au serveur web

Désolé, cette section n'a pas encore été écrite.

A venir bientôt: mSQL, PHP/FI, cgiwrap, Fast-cgi, MS frontpage extentions, et beaucoup d'autres.

9 FAQ (Foire Aux Questions)

Il n'y a pas de question fréquemment posées - pour l'instant...

10 Documents plus avancés

10.1 Livres de chez O'Reilly & Associates

A mon avis O'Reilly & Associates éditent les meilleures références techniques de la planète. Ils concentrent leurs efforts sur des sujets comme l'Internet, Unix et la programmation. Ils commencent tout doucement avec beaucoup d'exemples et lorsque vous terminez le livre vous êtes un expert. Je pense que vous pouvez arrêter même si vous avez seulement lu la moitié du livre. Ils ajoutent également un peu d'humour à des sujets qui sinon seraient lassants.

Ils disposent d'excellents livres sur HTML, PERL, la programmation CGI, Java, JavaScript, C/C++, Sendmail, Linux et beaucoup d'autres. De plus les sujets changeant rapidement (comme le HTML) sont mis à jour et révisés tout les six mois environ. Visitez donc le site web d' *O'Reilly & Associates* `http://www.ora.com/` ou arrêtez vous chez votre libraire local pour plus d'informations.

Et rappellez vous que s'il n'est pas marqué O'Reilly & Associates sur la couverture, c'est probablement quelqu'un d'autre qui l'a écrit.

10.2 Internet Request For Comments (RFC)

- RFC1866 written by T. Berners-Lee and D. Connolly, "Hypertext Markup Language - 2.0", 11/03/1995
- RFC1867 writtenm by E. Nebel and L. Masinter, "Form-based File Upload in HTML", 11/07/1995
- RFC1942 written by D. Raggett, "HTML Tables", 05/15/1996
- RFC1945 by T. Berners-Lee, R. Fielding, H. Nielsen, "Hypertext Transfer Protocol – HTTP/1.0", 05/17/1996.
- RFC1630 by T. Berners-Lee, "Universal Resource Identifiers in WWW: A Unifying Syntax for the Expression of Names and Addresses of Objects on the Network as used in the World-Wide Web", 06/09/1994
- RFC1959 by T. Howes, M. Smith, "An LDAP URL Format", 06/19/1996