Auteur	**David Parize**
Date de création	**30/06/2003**
Version	**1**
Date de dernière mise à jour	**30/06/03**

CYBERGATE -TP-APACHE_2.DOC

TABLE DES MATIERES

I. CONSEILS

- Le TP a été construit avec la version **d'Apache 1.3.27,** livrée sous format **rpm** (bien entendu, il faut toujours utiliser la dernière version, où les éventuelles failles de sécurité sont colmatées). Mais le fichier de configuration reste très semblable d'une version à l'autre, même si l'ordre des clauses change beaucoup !
- Pour les différents tests et réglages demandés, il est possible de travailler entièrement en ligne de commande (et même utiliser le navigateur ***lynx*** fonctionnant en mode texte.
- Faites une sauvegarde du fichier de configuration d'origine en le copiant sous le nom ***/etc/httpd/conf/httpd.old***
- A chaque modification du fichier de configuration, forcer la relecture par Apache de ce fichier par la commande ***killall -HUP httpd*** (au lieu de relancer le serveur par *service* ***httpd restart***)
- Dans ce qui suit, *serveur* désignera le nom de la machine hébergeant Apache. Si ce nom symbolique n'est pas interprété dans les URL, faute de DNS local, ou de résolution par */etc/hosts*, utilisez à la place une adresse IP du genre *http://192.1.1.(20+x)*, où x=1..10 (ou conformément au plan d'adressage de la salle)
- Certaines questions nécessiteront peut-être de se reporter au cours d'Apache

II. TP1 CONFIGURATION GLOBALE D'APACHE

Observer le contenu de /etc/httpd. On y trouve outre le répertoire de configuration, 2 liens symboliques qui pointent vers les répertoires

/var/log/httpd où Apache place ses journaux d'activités :

access_log pour les requêtes traitées (qu'elles soient réussies ou non)

error_log, pour noter les erreurs de fonctionnement du serveur

/usr/lib/apache où se trouvent les modules susceptibles d'être chargés par le noyau d'Apache.
En particulier, les modules qui permettent à Apache d'interpréter directement les langages Perl (libperl.so) et Php (libphp4.so)

Le répertoire /**etc/httpd/conf** contient outre ***httpd.conf, apache-mime.types*** (qui permet de renseigner le client sur le moyen d'afficher un fichier suivant son extension), et 2 répertoires

vhosts pour paramétrer les hotes virtuels

addon-modules qui contient des fichiers de configuration, en particulier php.conf

A. Modifications de paramètres d'exécution

Supposons que le serveur d'établissement soit hébergé par une machine aux ressources limitées, et que le nombre de requêtes qui lui sont adressées n'est jamais considérable.

Question 1. ***Il y a 10 serveurs WEB en exécution lors du démarrage d'Apache, et 150 au maximum simultanément. Comment le vérifiez-vous ?***

Reponse 1.

Question 2. ***Votre capacité mémoire est limitée. Ramenez ces nombres à 4 et à 25. Redémarrez Apache. Vérifiez avec une commande ps***

Reponse 2.

B. Modifications de paramètres d'accessibilité

Question 3. ***cas 1 : modifier l'accès au site par défaut (/var/www/html) pour autoriser l'accès de tous les hotes du sous-réseau 192.1.0. sauf à partir des 2 hotes qui vous sont voisins. Bien sur, bien vérifier.***

Reponse 3. ***DocumentRoot***

Question 4. ***cas 2 : interdire l'accès de tous les hotes sauf les machines du formateur et une machine voisine***

Reponse 4.

C. Suivi du journal des accès

- Passer la commande: ***tail -f /var/log/httpd/access_log > /dev/tty5 &***

Question 5. ***Expliquer la signification de cette commande :***

Reponse 5.

Noter le **PID** du processus qui vient d'être lancé en tâche de fond, puis observer le contenu de la console tty5

- Demander aux groupes voisins de jouer les rôles de clients WEB (avec un browser lynx, Konqueror ou Mozilla) et de passer des requêtes *http://serveur/*, en direction de votre serveur, et observer les lignes de access_log affichées "en direct"

Question 6. ***Interpréter les champs de chaque ligne, en particulier répérer les renseignements qui concernent le client et les codes de retour des diverses requetes (200 réussie, 404 "not found" ...)***

Reponse 6.

D. La page d'accueil et les pages par défaut

Question 7. ***Quel est le fichier qui sert de page d'accueil et de test à l'installation ?***

Reponse 7.

Question 8. ***Comment charger cette page d'accueil du serveur Apache hébergé par votre propre machine ?***

Reponse 8.

Question 9. ***Pouvez-vous en faire autant avec les serveurs des autres groupes ?***

Reponse 9.

Question 10. ***Renommez index.bak la page d'accueil du serveur. Pouvez-vous l'obtenir comme précédemment ? Pourquoi ? Que faudrait-il faire ? (expliquez mais ne le faites pas)***

Reponse 10.

Question 11. ***Qu'obtenez-vous en passant l'URL : http://serveur/index.bak ? Expliquez pourquoi.***

Reponse 11.

Question 12. ***Créez rapidement une petite page HTML portant le nom accueil.html (et non pas welcome.html ;-)***
Vous voulez qu'elle devienne la page d'accueil de votre serveur. A noter que la page d'accueil initiale index.shtml doit toujours être présente dans /var/www/html
Rechercher avec grep la ligne de DirectoryIndex et la modifier convenablement.
Vérifier.

Reponse 12.

III. TP2 LES "PAGES PERSOS"

C'est très à la mode... tous vos collègues veulent publier leur page personnelle, voire gérer eux-mêmes un site. A la fois vous vous estimez investi d'une mission de service public, et vous ne désirez pas les renvoyer aux hébergeurs privés; mais comme "webmestre" du "site officiel" de votre établissement, vous ne voulez pas les gérer vous-mêmes...

Question 13. ***Vérifier bien la présence de la clause UserDir*** *public_html* ***dans le fichier de configuration. Rappelez ce qu'elle signifie.***

Reponse 13.

Question 14. ***Vérifier la présence ou modifier éventuellement pour avoir une directive :***

<Directory /home//public_html>*
order allow,deny
allow from all
</Directory>

Que faut-il alors faire pour permettre à vos utilisateurs de publier et de gérer eux-mêmes leurs "pages persos"?

Reponse 14.

Question 15. ***L'utilisateur*** **stagex (x=1 ..10)** ***crée lui-même le répertoire de son site web, et y place une page d'accueil*** accueil.html

Peut-il y accéder ? Si ce n'est pas le cas, cherchez à résoudre le problème en examinant les droits sur le chemin vers le fichier refusé. Etendre ***au minimum*** ces droits pour résoudre le problème
Conclusion : est-ce une bonne solution que de donner de tels droits à tous ?

Reponse 15.

Question 16. ***(*) toto vous demande un espace de publication Web, mais vous ne voulez pas lui créer de compte sur le serveur (évidemment il ne pourra pas mettre lui-même en ligne).***
Comment faites-vous pour le satisfaire ?

Reponse 16.

IV. TP3 PERMETTRE OU INTERDIRE L'AFFICHAGE DU CONTENU D'UN REPERTOIRE

Vous vous apercevez que le contenu d'un répertoire est affiché, lorsque le serveur ne trouve pas dans ce répertoire, l'une des pages par défaut (dont les noms sont listés par la clause ***DirectoryIndex*****)** n'est pas présente.
Vous voulez interdire l'affichage de ces fichiers (pas forcément tous publics) et générer un message d'erreur

Renommer ***index.bak*** la page d'accueil du serveur, et vérifier le comportement du serveur, pour tous les utilisateurs.
Rechercher les lignes comportant le mot-clé **indexes**, supprimer **indexes** sur le répertoire racine /

Question 17. ***Pouvez vous alors lister le répertoire racine du serveur Apache par http://serveur/ ?***

Reponse 17.

Créer un sous-répertoire nommé webftp à la racine du web **/var/www/html**
Permettre (exclusivement) à ce répertoire d'etre listé à l'aide d'un conteneur du genre :

<Directory /var/www/html/webftp>
Options Indexes
order allow,deny

allow from all
</Directory>

Question 18. ***Vérifier alors que l'affichage est bien interdit ailleurs***

Reponse 18.

V. TP4 PROTECTIONS DES ACCES ET WEBS PRIVES

Principe

La clause de base est : **AccessFileName** .htaccess (en ligne 248)

Les directives contenues dans ce fichier sont systématiquement interprétées avant d'autoriser l'accès au répertoire qui le contient.

Voici les directives usuelles et leur signification

- ***AuthType basic***, type d'authentification communément adopté, fait hélas circuler les mots de passe en clair
- ***AuthName texte***, affichera le texte comme invite dans la boite de dialogue
- ***AuthUserFile chemin/fichier***, précise le fichier qui contient les comptes et mots de passe des utilisateurs ayant droit d'accès
- ***Require valid-user*** liste tous noms, ou seulement les comptes énumérés dans la liste, auront accès,

Question 19.

On demande ici de protéger l'accès au sous-site privé de l'établissement, supposé etre situé dans le sous-répertoire /var/www/html/prive***.***
Il ne devra etre accessible qu'à un ensemble limité de comptes Apache (et non Linux) à créer.
La première requête adressée à ce répertoire protégé provoquera l'affichage d'une boite de dialogue par laquelle l'utilisateur devra s'authentifier (nom et mot de passe).

Reponse 19.

Créer le répertoire /**var/www/html/prive**, y placer quelques pages HTML.
Tester leur accessibilité pour tous. Sinon penser à modifier les permissions Linux sur ces fichiers.

Créer dans ce répertoire à protéger le fichier **.htaccess**. Voici une écriture standard :

AuthType Basic
AuthUserFile /etc/httpd/conf/users
AuthGroupFile /dev/null
AuthName "Acces prive"

<limit GET>
require valid-user
</limit>

Question 20. ***.Dans ces conditions où se trouvera le fichier d'authentification ?***

Reponse 20.

Question 21. ***Créer quelques comptes Apache avec la commande** htpasswd **puis examiner le fichier ainsi créé.***

Reponse 21.

Question 22. ***Tester. Pourquoi la protection ne semble t-elle pas fonctionner ?***

Rechercher dans le fichier de configuration d'Apache la ligne contenant **AllowOverride** et précédée de ***<Directory />*** qui fixe des directives par défaut. La valeur de ce paramètre ne doit pas etre **NONE** (dans ce cas la prise en compte de ***.htaccess*** est désactivée). Changer cette valeur et mettre **all**
Retester avec succès ! N'oubliez pas de relancer le navigateur quand on change de compte.

Reponse 22.

Question 23. ***(*) Procéder maintenant à la protection des pages personnelles de toto pour les amis de toto, un autre groupe d'utilisateurs.***

Reponse 23.

(*) Les fichiers .***htaccess*** doivent être lus, si bien qu'il est possible de connaitre l'emplacement du fichier d'authentification.
Pour interdire à tous l'accès à ces fichiers par http, ajouter :

```
<Files ~ "^\.ht">
   Order allow,deny
   Deny from all
</Files>
```

VI. TP5 DEFINIR DES ALIAS

Il s'agit de donner des noms virtuels à des répertoires qui ne se trouvent pas dans l'arborescence usuelle ***/var/www/html/*** (Apache est configuré ainsi sur la plupart des distributions de GNU/Linux)

Repérer les lignes où il est question du mot **alias** avec une commande **grep**
Cette déclaration d'alias est déjà écrite : **Alias /doc /usr/share/doc**
Tester un accés sur une station par **http://serveur/doc**, en remplaçant serveur par le nom attribué au serveur, ou son adresse IP.

Question 24. ***L'accès n'est-il pas "Forbidden" ?***

Reponse 24.

Question 25. ***Normalement, vous avez auparavant interdit totalement l'affichage des répertoires sauf sur le répertoire webftp. Or il est bien pratique de "naviguer" dans les docs html. Faites le nécessaire pour permettre cette navigation. Vérifier***

Reponse 25.

Question 26. ***Poursuivre le paramétrage particulier de ce répertoire, en accordant les permissions d'accès au "site" doc, localement et uniquement à partir de stations voisines choisies. Attention ! bien choisir la clause order. Vérifier.***

<Directory /usr/share/doc>
order .....
deny from all
 # permission d'accès local
allow from localhost, 127.0.0.1
 # permission d'accès de stations voisines
allow from .....
</Directory>

Reponse 26.

VII. TP6 INSTALLER DES SERVEURS VIRTUELS

Exemple de paramétrage

Se servir de l'exemple ci-dessous pour mettre en oeuvre un serveur virtuel sur votre serveur

```
# déclarer les nouveaux noms de machines dans le DNS local ou à défaut dans les fichiers hosts du
serveur # et des stations autorisée
127.0.0.1        localhost        localhost
192.1.1.10   info1.poste2.fr      poste2
192.1.1.10   www.poste2.fr       www

# paramétrage des différents répertoires
#############################################

# Protection maximale de la racine de l'hote du serveur
<Directory />
   Options FollowSymLinks
   AllowOverride None
</Directory>

# Paramétrage du site web usuel et par défaut  accessible par http://pc1/
################################################

<Directory "/var/www/html">
#################################
# Options possibles :"All", ou une combinaison de Indexes,Includes,FollowSymLinks,ExecCGI,MultiViews
Options Indexes FollowSymLinks

# Pour empecher l'action "outrepassante" des fichiers .htaccess dans les répertoires
# les paramètres possibles sont All, ou une combinaison qcq de Options,FileInfo,AuthConfig,Limit
AllowOverride None

# Pour controler les permissions d'accès au serveur, pour des droits restrictifs interdire D'ABORD de
partout, # puis ENSUITE accorder à des machines particulières
order deny,allow
deny from all
allow from localhost 192.1.0.0/255.255.255.0
</Directory>

# Paramétrage d'un site web virtuel qui sera accessible par http://www/
```

```
<Directory "/formapache">
#################################
   Options Indexes FollowSymLinks
   order deny,allow
   deny from all
   allow from localhost 192.1.1.x
</Directory>

# paramétrage des différents répertoires
#############################################

# Hotes virtuels nommés
########################
# le numéro ip de la machine
NameVirtualHost 192.1.1.x

# Le premier paragraphe décrit le site par défaut qui pointe vers DocumentRoot
```

 # description du serveur par defaut
#####################################
<VirtualHost 192.1.1.10>
DocumentRoot /var/www/html
ServerName info1.poste2.fr
</VirtualHost>

 # serveur virtuel pointant dans une autre partition, droit d'accès a réserver
###
<VirtualHost 192.1.1.x>
DocumentRoot /home/www/
ServerName www
</VirtualHost>

VIII. TP7 MISE A JOUR DES SITES WEB

Il s'agit de mettre en oeuvre plusieurs façons de gérer le site WEB à distance, et en particulier pouvoir mettre en ligne les pages.
Créer un utilisateur "***webmestre***" appelé **webadmin**, et donner la propriété de groupe au groupe **webadmin**, sur **/var/www/html**

1. Par FTP : Paramétrer la connexion au serveur sous le compte *webadmin* dans un client graphique comme **gftp**
2. Pour la mise en oeuvre d'un accès par un partage Samba à partir d'une station Windows voir le TP *tp-samba1.html*

IX. TP8 GESTION DISTANTE D'APACHE AVEC WEBMIN

Contexte : normalement vous avez précédemment installé une mise à jour de WEBMIN, puis créé un gestionnaire admin/admin.
Faites une sauvegarde du fichier de configuration comme */etc/httpd/conf/httpd.old*

1. Connectez vous comme root à webmin avec le protocole https sur le port 10000, et vérifiez que admin a bien accès au module d'administration d'Apache. Sinon accordez lui ce droit.
2. Reconnectez vous comme *admin*, parcourer les différents réglages de "configuration globale"