More secure than 'not-at-all' mass Virtual Hosting with apache

26 juin 2009 - VVT 2009

Geoffroy Desvernay

Centrale Marseille

Un site pour chacun

Contexte: donner accès à chacun (personnels, enseignants, chercheurs, élèves, associations, services, congrès, ...) la possibilité de gérer son site web.

Préoccupations usuelles de l'ASR (et du RSSI...):

- Eviter les usurpations/vols de données (ex: mots de passe)
- Ne pas déployer 1 serveur/utilisateur...
- Eviter l'usine à gaz si on ne la maîtrise pas entièrement
- Ne pas devoir passer trop de temps pour les MAJ, etc.
- Savoir évaluer/limiter la charge de chaque site (CPU, réseau, RAM, ...)

Notre cahier des charges:

- Proposer une 'jolie' URL du type:
http://prenom.nom.perso.univ.fr (comme à l'ESIL...)
- Donner accès aux languages 'CGI' les plus courants (php/perl/python/sh/...)
- Ne pas trop brider (pour éviter d'avoir à gérer des exceptions, et/ou maintenir un autre système d'hébergement, entre autres)
- Ne pas trop multiplier les machines (physiques ET virtuelles)
  → gagner du temps pour les inévitables MAJ de sécurité php...

Le (vieux) problème du «mutualisé»

« attaque »:

```
mechant@machine$ cat ~gentil/html/conf.php
<?php
// informations confidentielles !
// memo: penser à ne pas les donner
$conf['db_user']='mylogin';
$conf['db_pass']='mYp455w0rd';
$conf['db_db']='mybase';
?>
```

Vulnerabitité: umask 022

```
root@machine$ ls -al ~gentil/html/conf.php
-rwxr-xr-x 1 gentil eleves    164 25 jui 14:48 conf.php
```

« solution »:

```
root@machine$ chown :80 /home/*/html
root@machine$ chmod 2750 /home/*/html
```

Le vieux problème du «mutualisé»

Attaque II:

```
system('cp -r ../../voisin/html ~/htmlduvoisin');
```

Ou encore…

```
fopen('../../voisin/html/db.inc.php');
```

disable_functions = show_source, system, shell_exec, passthru, phpinfo, popen, proc_open, exec, eval, parse_ini_file, fopen, …

Et pour le perl/shell/python/c/… ?

Attaque III:

```
<pre><?php
require('../../voisin/html/db.inc.php');
?></pre>
```

open_basedir = /home/luietpasunautre

→ un fichier php.ini par utilisateur ?
Et les modules pear ?
Et les autres langages ?
Et les liens symboliques ?

Centrale Marseille

Des solutions existent…

•**metuxmpm**: basé sur 'perchild', supporte les threads (pas comme mod_php…)

•**mpm-peruser**: performant, un process/utilisateur lancé par le « multiplexeur » au besoin, mais dernière mise à jour en 2007 (apache 2.2.3) et rien depuis… le logiciel parfait ?

•**mpm-itk**: moins performant, mais maintenu.

•**mod_vhost-ldap**: permet de déplacer la config des vhosts dans un serveur LDAP. (schema spécifique) – pas de support pour mpm-itk

•**mod_vhs**: permet de définir des virtualhosts depuis des requêtes SQL/LDAP
+ usage des mod_ldap et/ou mod_dbd d'apache (cache compris)

•**suexec**: livré en standard avec apache, peu souple.

Centrale Marseille

Notre choix: deux solutions différentes

- 1 serveur 'perso' pour tous avec **proxy/rewrite+suexec/CGI**
  → performance peu cruciale
  → seule méthode stable sans aucune (re)configuration manuelle

- 1 serveur 'administratif' avec **mod-itk** + script 'ldap2apache_jail.pl'
  Sites 'institutionnels'

- 1 serveur 'guests' avec **mod-itk** + ldap2apache_jail.pl

La création d'un site web « non perso » passe donc par:
→ 1 compte LDAP
→ 1 enregistrement DNS (<= .centrale-marseille.fr)
→ le script tourne la nuit (ou manuel si urgence)

La création d'un site 'perso':
→ comptes LDAP standards '&(eduPersonAffiliation=member)'
→ script nocturne 'mails2login.pl'

Centrale Marseille

Serveur 'persos': little picture

```
GET /vvt2009 HTTP/1.1
Host: geoffroy.desvernay.perso.ec-m.fr

HTTP/1.x 302 OK
Location: http://geoffroy.desvernay.perso.ec-m.fr/vvt2009/
```

mod_proxy
(+mod_security?)

ProxyPass (RewriteRule [P])

ProxyPassReverse http://perso.ec-m.fr/~${LOGIN} interpolate

```
GET /~dgeo/vvt2009 HTTP/1.1
Host: perso.ec-m.fr

HTTP/1.x 302 OK
Location: http://perso.ec-m.fr/~dgeo/vvt2009/
```

Suexec
+userdir

→ Script: suexec/CGI

→ Statique: HTTP/1.x 200 OK

http://dgeo.perso.ec-m.fr/vvt2009/

Centrale Marseille

Le proxy

- On peut utiliser mod_security sur ce 'frontal'
- Ça n'est qu'un proxy (pas de contenu, ou seulement du statique: erreurs, etc.)
- On **peut** «cacher» le vrai serveur, transformer/filtrer certaines pages d'erreur, etc.
- On peut activer/desactiver des sites avec la RewriteMap

```
RewriteMap mails2login txt:/etc/apache22/mail2uid.map

RewriteCond  %{SERVER_NAME}  ^[-a-z\.0-9]+\.perso\.ec-m\.fr$ [NC]
RewriteRule  ^(.+)  ${lowercase:%{SERVER_NAME}}$1 [C]
RewriteRule  ^([-a-z\.0-9]+)\.perso[^/]*/(.*)$ http://perso.ec-
m.fr/~${mails2login:$1}/$2 [L,P,E=LOGIN:${mails2login:$1}]
```


Centrale Marseille

suexec+php: patch ou binfmt-support

suexec+CGI: #!/usr/bin/perl OK...

Pour php:

- Linux: chmod +x + binfmt-support:

```
(debian): update-binfmts --install PHP /usr/bin/php5-cgi --extension php;
```

- Autres: Patch suexec

```
+    if (strnstr(cmd, ".php",-4)) {
+        execl("/usr/local/bin/php-wrapper", "php-wrapper", cmd, NULL);
+    } else {
     execv(cmd, &argv[3]);
```

Ce patch présuppose l'existence d'un exécutable (script par exemple) 'php-wrapper' qui permettra d'introduire une certaine souplesse dans l'appel du binaire php (autorise 128M de ram pour untel, placer une directive 'open_basedir' personnalisée, ...

Avantages/Inconvénients de la «solution» suexec

- Suexec supporte mod_userdir nativement
- Le problème des URL's est réglé par mod_rewrite (comme mass virtual hosting par "La firme"), plus mod_proxy. => qui a dit lourd?
- configuration « une fois pour toutes »
- Filtrage des sites 'actifs' via une RewriteMap
- Droits sur les fichiers:
  - Les scripts ***doivent*** être protégés
  - les fichiers statiques (html, css, images, etc.) **doivent** rester lisibles par tous…
  - l'utilisateur moyen ne sait pas utiliser chmod…
- suexec et php: binfmt-support ou patch
- Performance de PHP/CGI => fastcgi ?

Pour les sites non 'perso': mpm-itk

Le serveur entier emprunte l'identité de l'utilisateur, ce qui simplifie la gestion (droits sur les fichiers, accounting, ...)

On peut gérer plus finement la priorité des uns et des autres, et utiliser les limites du système (limits.conf, login.conf, etc.)

Inconvénient: un fichier de conf par utilisateur...

=> Les configs sont générées par le script ldap2apache.pl

=> idéal pour des comptes/sites type services (rh, cri, ...)

=> installation/configuration **beaucoup** plus simple

Gestion des logs…

- Syslog vers un serveur central (syslog-ng) pour statistiques, traces et analyses

httpd.conf:

```
CustomLog  "|/usr/bin/logger -h syslog.nettest.egim -p local3.info"
```

syslog-ng.conf:

```
template www_log_template { template("$MSG\n"); template-escape(no); };
destination www_persos { file("/var/log/www/persos.log" template(www_log_template));};
log { source(servint); filter(fh_perso); filter(f_local3); filter(f_is_info); destination(www_persos); };
```

- Awstats (et autres) sur ce serveur, permettant un accès aux statistiques par chacun.
  → le script ldap2apache.pl génère également des configs awstats, sur le serveur syslog.
  → awstats génère des pages html statiques la nuit, et met ses bases à jour toutes les 1/2h

L'avenir...

mod_vhs+mpm-itk

Un patch existe pour mod_vhs 1.0.32
(pour l'instant dans la liste mod_vhs@lists.oav.net)

Prévu pour la v1.1 (1.2?) de mod_vhs.

La solution idéale pour alléger les 'perso' ?

Des questions ?

...Bon appétit !

Centrale Marseille