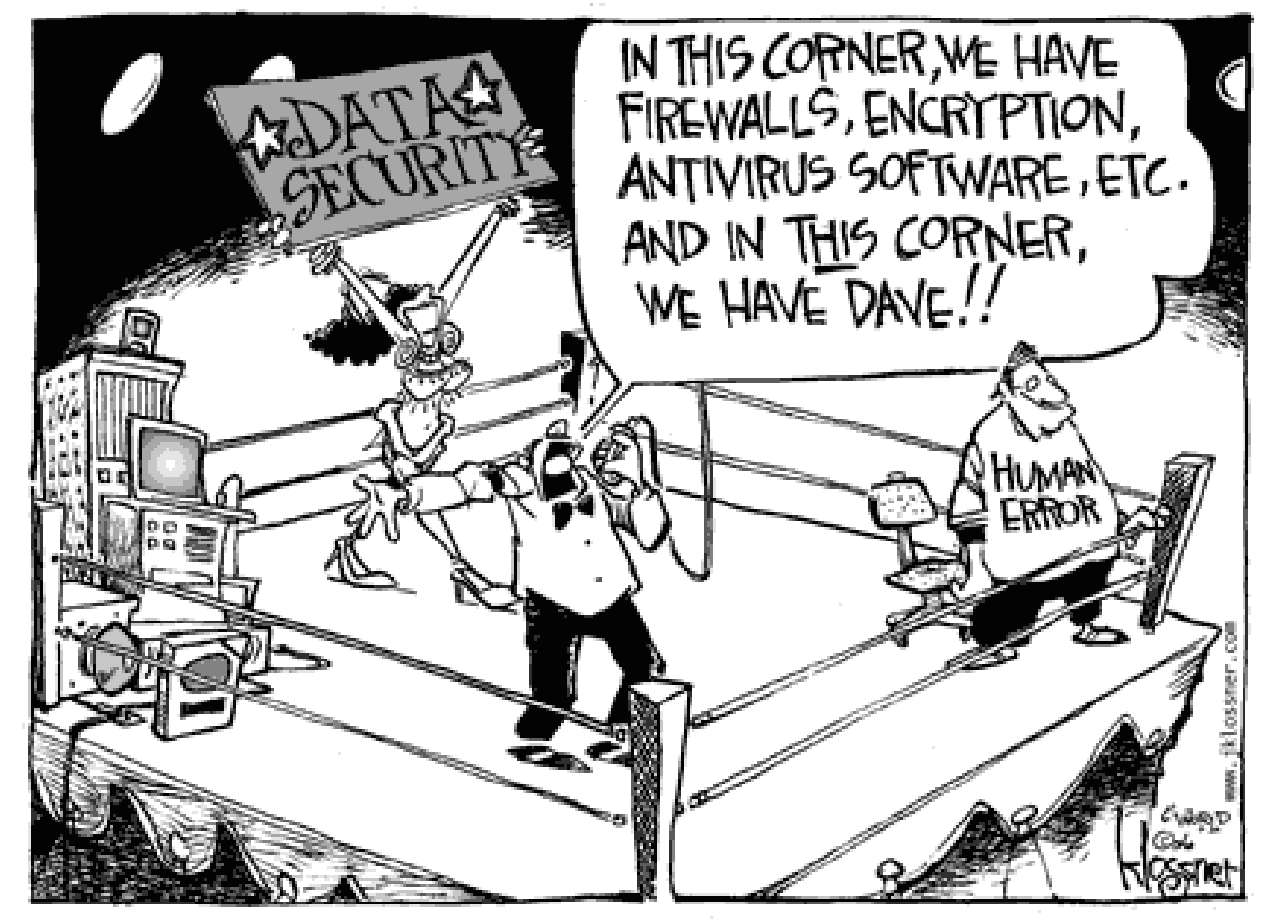
RAISIN/CEPAGE : séminaire web

Aurélien Dumez – Inria

ModSecurity, suPHP et PHP-FPM

Pourquoi ?


DATA SECURITY
IN THIS CORNER, WE HAVE FIREWALLS, ENCRYPTION, ANTIVIRUS SOFTWARE, ETC.
AND IN THIS CORNER, WE HAVE DAVE!!
HUMAN ERROR

ModSecurity

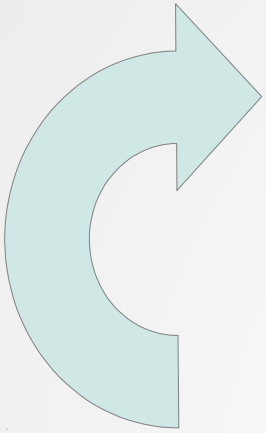
En résumé (TL;DR)

- WAF (not a dog) : Web Application Firewall
- Version actuelle : 2.8 (CentOS 7 : 2.7.3 – Debian 7 : 2.6.6)

- 2003 création pour Apache httpd
- Janvier 2013 support de IIS
- Mai 2013 support de nginx

- Efficace mais phase de paramétrage assez longue
- Gourmand en ressources (mémoire, CPU, disque)
- Empaqueté par la plupart des distributions GNU/Linux

Principe de base



- Intercepte les requêtes HTTP **et** les réponses
- Analyse ces éléments au moyen d'un ensemble de règles
- Effectue des actions en fonction du contenu des règles

⇒ Sans un ensemble complet et récent de règles, ModSecurity est inutile

Analyse

- 5 phases
  1. En-têtes de la requête
  2. Corps de la requête
  3. En-têtes de la réponse
  4. Corps de la réponse
  5. Écriture dans les journaux
- Chaque règle peut préciser à quelle phase elle intervient.

Étude d'une règle

SecRule ARGS "<script>" t:lowercase log,deny,status:403

Étude d'une règle

SecRule ARGS "<script>" t:lowercase log,deny,status:403

- Directives
  - SecRuleEngine : On / Off / DetectionOnly
  - SecRule : définition règle d'analyse (action facultative)

Étude d'une règle

SecRule ARGS "<script>" t:lowercase log,deny,status:403

- Variables
  - ARGS : tous les paramètres contenus dans la requête
  - ARGS:p : seulement le paramètre p
  - REMOTE_ADDR : @IP du client émettant la requête

Étude d'une règle

SecRule ARGS "<script>" t:lowercase log,deny,status:403

- Opérateur
  - Une simple chaîne à trouver dans la requête (le cas ici)
  - Une expression rationnelle (motif)
  - Des opérateur prédéfinis spécialisés
    - Pour faire des comparaisons numériques : @eq, @gt...
    - Valider du XML : @validateDTD
    - Invoquer un outil externe : @inspectFile

Étude d'une règle

SecRule ARGS "<script>" t:lowercase log,deny,status:403

- Transformations
  - Appliquées avant l'analyse
  - Pour la simplifier :
    - Lowercase (pour ne pas rater <ScRiPt>, par exemple)
    - NormalisePath (simplifier .//toto/.//titi/./)
    - CompressWhitespace (0x20, \f, \t, \n, \r, \v, 0xa0 → 0x20 et suppression des espaces consécutifs)

Étude d'une règle

SecRule ARGS "<script>" t:lowercase log,deny,status:403

- Action
  - log : enregistre l'événement dans les journaux système
  - deny : arrête l'analyse et bloque la requête
  - status : renvoie au client une erreur HTTP (ici : 403 - forbidden)

A vous !

```
SecRule ARGS "(?i)(<script[^>]*>[\s\S]*?<\/script[^>]*>|
<script[^>]*>[\s\S]*?<\/script[[\s\S]]*[\s\S]|<script[^>]*>[\s\S]*?
<\/script[\s]*[\s]|<script[^>]*>[\s\S]*?<\/script|
<script[^>]*>[\s\S]*?)"
"id:'973336',phase:2,rev:'1',ver:'OWASP_CRS/2.2.9',maturity:'1',accuracy
:'8',t:none,t:urlDecodeUni,t:htmlEntityDecode,t:jsDecode,t:cssDecode,log
,capture,msg:'XSS Filter - Category 1: Script Tag
Vector',tag:'OWASP_CRS/WEB_ATTACK/XSS',tag:'WASCTC/WASC-
8',tag:'WASCTC/WASC-
22',tag:'OWASP_TOP_10/A2',tag:'OWASP_AppSensor/IE1',tag:'PCI/6.5.1',logd
ata:'Matched Data: %{TX.0} found within %{MATCHED_VAR_NAME}: %
{MATCHED_VAR}',severity:'2',setvar:'tx.msg=%
{rule.msg}',setvar:tx.xss_score=+%
{tx.critical_anomaly_score},setvar:tx.anomaly_score=+%
{tx.critical_anomaly_score},setvar:tx.%{rule.id}-
OWASP_CRS/WEB_ATTACK/XSS-%{matched_var_name}=%{tx.0}"
```

LV1 : les règles de ModSec

- La règle précédente était très simple
- Les menaces sont nombreuses et de nouvelles apparaissent quotidiennement

Comment alimenter efficacement en règles ModSecurity ?

OWASP ModSecurity Core Rule Set Project

LV2 : les logs de ModSec (1/2)

```
--5f3acc73-A-- [Audit Log Header]
[26/Jul/2011:11:45:49 +0700] Ti5GfNJW71wAAC63D4YAAABU 1.2.3.4 12446 5.6.7.8 80

--5f3acc73-B-- [Request Headers]
POST /newthread.php?do=postthread&f=8 HTTP/1.1
Host: www.example.com
User-Agent: Mozilla/5.0 (Windows NT 5.1; rv:2.0.1) Gecko/20100101 Firefox/4.0.1
[...snip...]
Content-Length: 24390

-5f3acc73-C-- [Request Body]
subject=some subject&message=link to a site thats banned such as http://www.example.com
```

LV2 : les logs de ModSec (2/2)

```
--5f3acc73-F-- [Response Headers]
HTTP/1.1 403 Forbidden
[..snip...]

--5f3acc73-H-- [Audit Log Trailer]
Message: [file "/etc/httpd/modsecurity.d/30_asl_antispam.conf"] [line "52"]
 [id "300001"] [rev "23"] [msg "Atomicorp.com WAF Rules: Blacklist Spam Domain"]
[data ""] [severity "CRITICAL"] Access denied with code 403 (phase 2).
Matched phrase "www.example.com" at ARGS:message.
Action: Intercepted (phase 2)
Apache-Handler: php5-script
[...snip...]
```

Au quotidien

- Avalanche de faux positifs
  - Avec les CMS et autres applis web et leurs greffons
  - Nécessite des exceptions : `SecRuleRemoveById`
- Tentation à combattre : `SecRuleEngine Off`
- Ménager les utilisateurs
  - Générer une liste d'exceptions dans un premier temps : `SecRuleEngine DetectionOnly`
  - Informer et être réactif en cas de faux positif
  - S'appuyer sur l'expérience de la communauté pour écrire les listes d'exceptions (en particulier pour les CMS)

Pour en savoir plus

- Site du projet
  - https://www.modsecurity.org/
- OWASP ModSecurity Core Rule Set Project
  - https://www.owasp.org/
- ModSecurity Handbook :
  - https://www.feistyduck.com/

suPHP

Question de droits

chmod -R 0777 /var/www/

- Problème
  - Par défaut, httpd exécute les scripts PHP avec l'identité de son propre utilisateur (www-data, apache...)
- Conséquences
  - Incompréhension des droits précis à donner à l'arborescence du DocumentRoot
  - mod_php : une faille dans l'interpréteur PHP peut compromettre le bon fonctionnement du serveur httpd et réciproquement
  - En environnement web partagé, une configuration trop basique peut permettre des altérations de fichiers entre applications

su(per)PHP

- Module Apache httpd (mod_suphp) et binaire setuid root (suphp)
- Configuration simple dans un fichier de type INI
- Empaqueté par la plupart des distributions GNU/Linux
- Exécute les scripts PHP avec des droits différents (souvent ceux du propriétaire des fichiers)
- Par défaut, refuse d'exécuter les scripts avec des UID/GID inférieurs à 100 (comptes "systèmes")

suPHP

- Trois modes d'exécution
  - **Owner** : le script est exécuté avec les UID/GID de son propriétaire
  - **Force** : le script est exécuté avec les UID/GID précisés dans la conf apache
  - **Paranoid** (défaut) : le script est exécuté avec les UID/GID de son propriétaire uniquement s'ils correspondent à ceux déclarés dans la conf apache

suPHP

- Contrôle des droits avant exécution
  - allow_file_group_writeable (true/**false**)
    - Pas d'exécution si les droits du script sont : g+w
  - allow_file_others_writeable (true/**false**)
    - Pas d'exécution si les droits du script sont : o+w
  - allow_directory_group_writeable (true/**false**)
    - Pas d'exécution si le répertoire . a pour droits : g+w
  - allow_directory_others_writeable (true/**false**)
    - Pas d'exécution si le répertoire . a pour droits : o+w

Inconvénients

- Binaire suphp setuid root
- Exécution des scripts en mode CGI : perfs
- Dernière mise à jour ancienne (juin 2013)

Pour en savoir plus

- Site du projet
  - http://www.suphp.org/

PHP-FPM

En résumé (TL;DR)

- Interpréteur PHP communiquant par FastCGI
  - FPM : FastCGI Process Manager
- Indépendant du serveur HTTP (protocole FastCGI)
- Intégré à PHP depuis la version 5.3.3 (2010)
- Workers sous différents UID/GID et php.ini différents
- "chroot"able
- Rend l'utilisation de suPHP inutile
- Empaqueté par la plupart des distributions GNU/Linux

Internals

1 pool = ensemble de processus tournant dans la même configuration (pour servir une appli web, par exemple)

- Un processus "master" (conf : php-fpm.conf)
  - Contrôle des pools et création des "workers"
- N processus "workers"
  - Écoute des requêtes du serveur web (sockets UNIX ou TCP)
  - Exécution des scripts PHP et retour du résultat au serveur web

Master

Exemple de configuration

```
[global]
Pid = /var/run/php5-fpm.pid
error_log = /var/log/php5-fpm.log
Include = /etc/php5/fpm/pool.d/*.conf
```

Pool

Exemple de configuration

```
[www]
user = www-data
group = www-data
listen = 127.0.0.1:9000
listen.allowed_clients = 127.0.0.1
pm = dynamic
pm.max_children = 5
pm.start_servers = 2
pm.min_spare_servers = 2
pm.max_spare_servers = 3
chdir = /
```

Pour en savoir plus

- Page historique du projet
  - http://php-fpm.org/
- Dans la documentation du projet PHP
  - http://php.net/
- A propos de FastCGI
  - http://www.fastcgi.com/

Merci (encore) !