FreeBSD Administration - Support #562

Install mod_evasive for Apache 2.4 on FreeBSD

02/13/2015 10:39 PM - Daniel Curtis

Status:	Closed	**Start date:**	02/13/2015
Priority:	Normal	**Due date:**	
Assignee:	Daniel Curtis	**% Done:**	100%
Category:	Web Server	**Estimated time:**	1.00 hour
Target version:	FreeBSD 9	**Spent time:**	1.50 hour

Description

This is a simple guide for installing and configuring mod_evasive for Apache 2.4 on FreeBSD 9.2.

- Update the system and ports tree:

```
pkg update && pkg upgrade
portsnap fetch extract
```

- Install git:

```
pkg install git
```

Install mod_evasive

- Edit the mod_evasive Makefile:

```
cd /usr/ports/www/mod_evasive
vi Makefile
```

  - And change the line **USE_APACHE=22** to:

```
USE_APACHE=     24
```

- Begin compilation:

```
make install clean
```

  - Currently the port will fail with output similar to the following:

```
mod_evasive20.c: In function 'access_checker':
mod_evasive20.c:142: error: 'conn_rec' has no member named 'remote_ip'
mod_evasive20.c:146: error: 'conn_rec' has no member named 'remote_ip'
mod_evasive20.c:158: error: 'conn_rec' has no member named 'remote_ip'
mod_evasive20.c:165: error: 'conn_rec' has no member named 'remote_ip'
mod_evasive20.c:180: error: 'conn_rec' has no member named 'remote_ip'
mod_evasive20.c:187: error: 'conn_rec' has no member named 'remote_ip'
mod_evasive20.c:208: error: 'conn_rec' has no member named 'remote_ip'
mod_evasive20.c:212: warning: implicit declaration of function 'getpid'
mod_evasive20.c:215: error: 'conn_rec' has no member named 'remote_ip'
mod_evasive20.c:221: error: 'conn_rec' has no member named 'remote_ip'
mod_evasive20.c:222: error: 'conn_rec' has no member named 'remote_ip'
mod_evasive20.c:228: error: 'conn_rec' has no member named 'remote_ip'
```

```
apxs:Error: Command failed with rc=65536
.
*** [do-build] Error codhttp://xmodulo.com/harden-apache-web-server-mod_security-mod_evasive-centos.htmle 1

Stop in /usr/ports/www/mod_evasive.
```

- Fix the working mod_evasive source code:

```
sed -i '' -e 's/remote_ip/client_ip/g' work/mod_evasive/mod_evasive20.c
```

- Then finish installing mod_evasive:

```
make install clean
```

- Create the mod_evasive config file:

```
vi /usr/local/etc/apache24/modules.d/010_mod_evasive.conf
```

  - And add the following:

```
LoadModule evasive20_module   libexec/apache24/mod_evasive20.so

<IfModule evasive20_module>
#increases size of hash table. Good, but uses more RAM.
DOSHashTableSize    3097
#Interval, in seconds, of the page interval.
DOSPageInterval     1
#Interval, in seconds, of the site interval.
DOSSiteInterval     1
#period, in seconds, a client is blocked.  The counter is reset to 0 with every access within this interval.
DOSBlockingPeriod   10
#threshold of requests per page, per page interval.  If hit == block.
DOSPageCount        2
#threshold of requests for any object by the same ip, on the same listener, per site interval.
DOSSiteCount        50
#locking mechanism prevents repeated calls.  email can be sent when host is blocked (leverages the following by default "/bin/mail -t %s")
DOSEmailNotify      admin@example.com
#locking mechanism prevents repeated calls.  A command can be executed when a host is blocked.  %s is the host IP.
#DOSSystemCommand    "su - someuser -c '/sbin/... %s ...'"
#DOSLogDir           "/var/lock/mod_evasive"
#whitelist an IP., leverage wildcards, not CIDR, like 127.0.0.*
#DOSWhiteList 127.0.0.1
</IfModule>
```

- Restart apache24 to enable mod_evasive

```
service apache24 restart
```

- Now check to see that the module loaded correctly:

```
apachectl -M
```

- *Truncated output*

```
Loaded Modules:
 ...
 evasive20_module (shared)
```

Testing mod_evasive

Using Perl

- On a remote machine, create test-evasive.pl:

```
vi test-evasive
```

  - And add the following:

```
#!/usr/bin/perl
# test-evasive.pl: small script to test mod_evasive's effectiveness

use IO::Socket;
use strict;

for(0..100) {
  my($response);
  my($SOCKET) = new IO::Socket::INET( Proto   => "tcp",
                                      PeerAddr=> "www.example.com:80");
  if (! defined $SOCKET) { die $!; }
  print $SOCKET "GET /?$_ HTTP/1.0nn";
  $response = <$SOCKET>;
  print $response;
  close($SOCKET);
}
```

  NOTE: Change the **PeerAddr** to the URL to be tested.

- Once the file is saved, run it:

```
perl test-evasive.pl
```

Using Apache Bench

- Apache server benchmarking tool.

```
ab -n1000 -c1000 http://www.example.com/index.php
```

  - -n: Number of requests to perform for the benchmarking session.
  - -c: Number of multiple requests to perform at a time.

Resources

- http://xmodulo.com/harden-apache-web-server-mod_security-mod_evasive-centos.html

Related issues:

Related to FreeBSD Administration - Support #560: Hardening Apache 2.4 & PHP ...	**Closed**	**02/13/2015**

History

#1 - 02/14/2015 10:57 AM - Daniel Curtis

- Target version set to FreeBSD 9

#2 - 02/14/2015 10:57 AM - Daniel Curtis

- Category set to 1

#3 - 02/14/2015 12:11 PM - Daniel Curtis

- Category set to Web Server

#4 - 03/11/2015 06:22 AM - Daniel Curtis

- Subject changed from Installing mod_evasive for Apache 2.4 on FreeBSD to Install mod_evasive for Apache 2.4 on FreeBSD

- Description updated

- Status changed from New to In Progress

- % Done changed from 0 to 80

#5 - 03/11/2015 07:27 AM - Daniel Curtis

- Status changed from In Progress to Resolved

- % Done changed from 80 to 100

#6 - 03/17/2015 07:54 PM - Daniel Curtis

- Description updated

#7 - 03/17/2015 08:00 PM - Daniel Curtis

- Description updated

#8 - 03/17/2015 08:00 PM - Daniel Curtis

- Description updated

#9 - 03/18/2015 09:39 AM - Daniel Curtis

- Status changed from Resolved to Closed

#10 - 04/11/2015 01:18 PM - Daniel Curtis

- Related to Support #560: Hardening Apache 2.4 & PHP 5 on FreeBSD added