Installing ModSecurity WAF with Nginx on Ubuntu

Building ModSecurity

Download and install the require dependencies:

```
sudo apt-get install -y git build-essential libpcre3 libpcre3-dev libssl-dev libtool autoconf apac
```

Checkout the latest source code for ModSecurity:

```
cd /usr/srcsudo git clone -b nginx_refactoring https://github.com/SpiderLabs/ModSecurity.git
```

Build ModSecurity:

```
cd ModSecurity
sudo ./autogen.sh
sudo ./configure --enable-standalone-module --disable-mlogcsudo make
```

Building Nginx

Download the latest source code for Nginx (versions can be checked here: https://nginx.org/en/download.html)

```
cd /usr/src
sudo wget http://nginx.org/download/nginx-1.15.5.tar.gzsudo tar xvzf nginx-1.15.5.tar.gz
```

Build Nginx:

```
cd nginx-1.15.5/sudo ./configure --user=www-data --group=www-data --add-
module=/usr/src/ModSecurity/nginx/modsecurity --with-http_ssl_module
sudo makesudo make install
```

Configuring Nginx

Modify the default Nginx user:

```
sudo sed -i "s/#user nobody;/user www-data www-data;/" /usr/local/nginx/conf/nginx.conf
```

Verify the installation by testing the default configuration file for errors:

```
/usr/local/nginx/sbin/nginx -t
```

Create a systemd unit file at /lib/systemd/system/nginx.service so the Nginx service will start at boot and can be controlled:

```
[Service]
Type=forkingExecStartPre=/usr/local/nginx/sbin/nginx -t -c /usr/local/nginx/conf/nginx.conf
ExecStart=/usr/local/nginx/sbin/nginx -c /usr/local/nginx/conf/nginx.conf
ExecReload=/usr/local/nginx/sbin/nginx -s reload
KillStop=/usr/local/nginx/sbin/nginx -s stop
KillMode=process
Restart=on-failure
RestartSec=42s
PrivateTmp=true
LimitNOFILE=200000
[Install]WantedBy=multi-user.target
```

The service can be controlled with the following commands:

```
sudo systemctl start nginx.service
sudo systemctl stop nginx.servicesudo systemctl restart nginx.service
```

Edit /usr/local/nginx/conf/nginx.conf to turn on ModSecurity and include the configuration file for each endpoint. For example, change:

```
location / {
    root   html;
    index  index.html index.htm;}
```

To:

```
location / {
    ModSecurityEnabled on;
    ModSecurityConfig modsec_includes.conf;
    root   html;
    index  index.html index.htm;}
```

Create the file /usr/local/nginx/conf/modsec_includes.conf and add:

```
include modsecurity.conf
include owasp-modsecurity-crs/crs-setup.confinclude owasp-modsecurity-crs/rules/REQUEST-900-EXCLUSION-RULES-BEFORE-CRS.confinclude owasp-modsecurity-crs/rules/REQUEST-901-INITIALIZATION.conf
include owasp-modsecurity-crs/rules/REQUEST-905-COMMON-EXCEPTIONS.confinclude owasp-modsecurity-crs/rules/REQUEST-910-IP-REPUTATION.confinclude owasp-modsecurity-crs/rules/REQUEST-911-METHOD-ENFORCEMENT.conf
include owasp-modsecurity-crs/rules/REQUEST-912-DOS-PROTECTION.confinclude owasp-modsecurity-crs/rules/REQUEST-913-SCANNER-DETECTION.confinclude owasp-modsecurity-crs/rules/REQUEST-920-PROTOCOL-ENFORCEMENT.confinclude owasp-modsecurity-crs/rules/REQUEST-921-PROTOCOL-ATTACK.conf
include owasp-modsecurity-crs/rules/REQUEST-930-APPLICATION-ATTACK-LFI.confinclude owasp-modsecurity-crs/rules/REQUEST-931-APPLICATION-ATTACK-RFI.confinclude owasp-modsecurity-crs/rules/REQUEST-932-APPLICATION-ATTACK-RCE.confinclude owasp-modsecurity-crs/rules/REQUEST-933-APPLICATION-ATTACK-PHP.confinclude owasp-modsecurity-crs/rules/REQUEST-941-APPLICATION-ATTACK-XSS.conf
include owasp-modsecurity-crs/rules/REQUEST-942-APPLICATION-ATTACK-SQLI.confinclude owasp-modsecurity-crs/rules/REQUEST-943-APPLICATION-ATTACK-SESSION-FIXATION.confinclude owasp-modsecurity-crs/rules/REQUEST-949-BLOCKING-EVALUATION.confinclude owasp-modsecurity-crs/rules/RESPONSE-950-DATA-LEAKAGES.confinclude owasp-modsecurity-crs/rules/RESPONSE-951-DATA-
```

```
LEAKAGES-SQL.conf
include owasp-modsecurity-crs/rules/RESPONSE-952-DATA-LEAKAGES-JAVA.confinclude owasp-
modsecurity-crs/rules/RESPONSE-953-DATA-LEAKAGES-PHP.confinclude owasp-modsecurity-
crs/rules/RESPONSE-954-DATA-LEAKAGES-IIS.confinclude owasp-modsecurity-crs/rules/RESPONSE-959-
BLOCKING-EVALUATION.conf
include owasp-modsecurity-crs/rules/RESPONSE-980-CORRELATION.conf
include owasp-modsecurity-crs/rules/RESPONSE-999-EXCLUSION-RULES-AFTER-CRS.conf
```

Import the ModSecurity configuration files:

```
sudo cp /usr/src/ModSecurity/modsecurity.conf-recommended /usr/local/nginx/conf/modsecurity.conf
sudo cp /usr/src/ModSecurity/unicode.mapping /usr/local/nginx/conf/
```

Enable the SecRuleEngine in modsecurity.conf:

```
sudo sed -i "s/SecRuleEngine DetectionOnly/SecRuleEngine On/" /usr/local/nginx/conf/modsecurity.co
```

Add the OWASP ModSecurity Core Rule Set:

```
cd /usr/local/nginx/confsudo git clone https://github.com/SpiderLabs/owasp-modsecurity-crs.git
sudo cd owasp-modsecurity-crs
sudo mv crs-setup.conf.example crs-setup.conf
sudo cd rulessudo mv REQUEST-900-EXCLUSION-RULES-BEFORE-CRS.conf.example REQUEST-900-EXCLUSION-
RULES-BEFORE-CRS.conf
sudo mv RESPONSE-999-EXCLUSION-RULES-AFTER-CRS.conf.example RESPONSE-999-EXCLUSION-RULES-AFTER-CRS
```

Restart the Nginx service for the configuration changes to take effect:

```
sudo systemctl restart nginx.service
```

Revision #17

Created Tue, Oct 23, 2018 9:59 AM by William Puckering

Updated Wed, Oct 24, 2018 9:35 AM by William Puckering