

ANEXO 1 INSTALACIÓN WAFs OPEN SOURCE:

- **Instalación y configuración de WAF Modsecurity Apache**

Requerimientos necesarios para implementar WAF ModSecurity:

	ModSecurity
Sistema Operativo	Centos 7
Interfaz	Minimal, sin entorno gráfico
Interza Web para administrar y configurar	NO
Tamaño en disco duro	50 GB
Memoria RAM	2 GB
Procesadores	2
Paquete	httpd-2.4.6-90
Modulo	proxy_module
Paquete	mod_security-2.9.3
Modulo	security2_module
Paquete	mod_security-crs

Implementación del WAF ModSecurity:

La implementación del WAF Modsecurity requiere llevar acabo los siguientes pasos importantes para su instalación:

1. Instalar el sistema operativo centos 7 en una máquina virtual, una vez instalado actualizar el sistema operativo.
2. Instalar el paquete httpd.

```
[root@localhost ~]# yum install httpd
Complementos cargados:fastestmirror
Loading mirror speeds from cached hostfile
* base: mirror.epn.edu.ec
* extras: mirror.epn.edu.ec
* updates: mirror.epn.edu.ec
Resolviendo dependencias
```

3. Verificar que el módulo proxy_module se encuentre activado con el comando: httpd -M

```
[root@localhost ~]# httpd -M
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using localhost.localdomain. Set the 'ServerName' directive globally to suppress this message
Loaded Modules:
core_module (static)
so_module (static)
http_module (static)
access_compat_module (shared)
actions_module (shared)
alias_module (shared)
allowmethods_module (shared)
auth_basic_module (shared)
auth_digest_module (shared)
auth_anon_module (shared)
authn_core_module (shared)
authn_dbd_module (shared)
authn_dbm_module (shared)
authn_file_module (shared)
authn_socache_module (shared)
authz_core_module (shared)
authz_dbd_module (shared)
authz_dbm_module (shared)
authz_groupfile_module (shared)
authz_host_module (shared)
authz_owner_module (shared)
authz_user_module (shared)
autoindex_module (shared)
cache_module (shared)
```

Activar Windows
Ve a Configuración para activar Windows.

4. Configurar el proxy inverso, para ello es necesario configurar el siguiente fichero indicando la dirección IP del aplicativo web (192.168.1.41), donde se realizará la redirección de las consultas de los clientes.

```
192.168.1.40:22
About these Buttons  Unix Commands  Run Sample Script  Call Host from Host Directory  Purchase Licen
GNU nano 2.3.1 Fichero: /etc/httpd/conf.d/test.conf
<VirtualHost *:80>
    ServerName proxy-waf.com.ec
    ProxyPreserveHost On
    ProxyPass / http://192.168.1.41/
    ProxyPassReverse / http://192.168.1.41/
</VirtualHost>
```

5. Ejecutar los siguientes comandos para revisar que las configuraciones realizadas esten correctas:

```
apachectl configtest
```

```
apachectl graceful
```

```
[root@localhost ~]# apachectl configtest
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using localhost.localdomain. Set the 'ServerName' directive globally to suppress this message
Syntax OK
[root@localhost ~]# apachectl graceful
[root@localhost ~]#
```

- Reiniciar el servicio httpd y comprobar que se tenga acceso al aplicativo web vulnerable, apuntando la dirección IP o nombre de dominio del WAF.

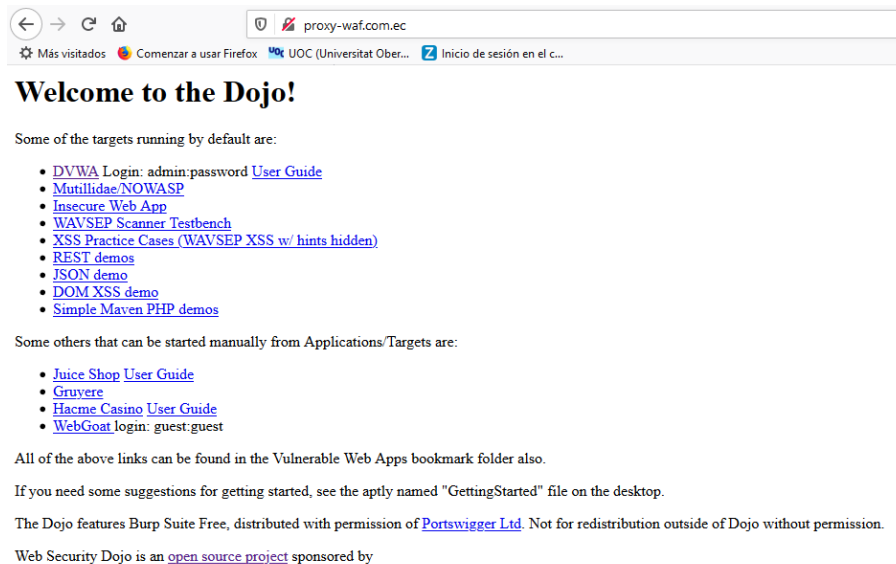
sudo systemctl restart httpd

```
[root@localhost ~]# systemctl restart httpd
[root@localhost ~]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: disabled)
   Active: active (running) since mié 2019-10-23 22:19:00 -05; 4s ago
     Docs: man:httpd(8)
           man:apachectl(8)
  Process: 2533 ExecStop=/bin/kill -WINCH ${MAINPID} (code=exited, status=0/SUCCESS)
 Main PID: 2538 (httpd)
   Status: "Processing requests..."
    CGroup: /system.slice/httpd.service
           └─2538 /usr/sbin/httpd -DFOREGROUND
             └─2539 /usr/sbin/httpd -DFOREGROUND
               └─2540 /usr/sbin/httpd -DFOREGROUND
                 └─2541 /usr/sbin/httpd -DFOREGROUND
                   └─2542 /usr/sbin/httpd -DFOREGROUND
                     └─2543 /usr/sbin/httpd -DFOREGROUND

oct 23 22:19:00 localhost.localdomain systemd[1]: Starting The Apache HTTP Server...
oct 23 22:19:00 localhost.localdomain httpd[2538]: AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using localhost.lo...s message
oct 23 22:19:00 localhost.localdomain systemd[1]: Started The Apache HTTP Server.
Hint: Some lines were ellipsized, use -l to show in full.
[root@localhost ~]#
```

Activar Windows
Ve a Configuración para activar Windows.

Comprobar el acceso en un navegador del cliente:



proxy-waf.com.ec

Más visitados Comenzar a usar Firefox UOC (Universitat Ober... Inicio de sesión en el c...

Welcome to the Dojo!

Some of the targets running by default are:

- [DVWA Login: admin:password User Guide](#)
- [Mutillidae/NOWASP](#)
- [Insecure Web App](#)
- [WAVSEP Scanner Testbench](#)
- [XSS Practice Cases \(WAVSEP XSS w/ hints hidden\)](#)
- [REST demos](#)
- [JSON demo](#)
- [DOM XSS demo](#)
- [Simple Maven PHP demos](#)

Some others that can be started manually from Applications/Targets are:

- [Juice Shop User Guide](#)
- [Gruyere](#)
- [Hacme Casino User Guide](#)
- [WebGoat login: guest:guest](#)

All of the above links can be found in the Vulnerable Web Apps bookmark folder also.

If you need some suggestions for getting started, see the aptly named "GettingStarted" file on the desktop.

The Dojo features Burp Suite Free, distributed with permission of [Portswigger Ltd](#). Not for redistribution outside of Dojo without permission.

Web Security Dojo is an [open source project](#) sponsored by



- Descargar el paquete modSecurity-2.9.3, de la siguiente página web: `wget https://www.modsecurity.org/tarball/2.9.3/modsecurity-2.9.3.tar.gz`

8. Descomprimir el archivo descargado, con el comando: `tar -xzf modsecurity-2.9.3.tar.gz`

```
[root@localhost ~]# tar -xzf modsecurity-2.9.3.tar.gz
[root@localhost ~]# ls -la
total 4240
dr-xr-x---.  3 root root    192 oct 24 14:44 .
dr-xr-xr-x. 17 root root    244 oct 24 11:50 ..
-rw-----.  1 root root   1343 oct 24 09:48 anaconda-ks.cfg
-rw-----.  1 root root     58 oct 24 11:41 .bash_history
-rw-r--r--.  1 root root     18 dic 28 2013 .bash_logout
-rw-r--r--.  1 root root    176 dic 28 2013 .bash_profile
-rw-r--r--.  1 root root    176 dic 28 2013 .bashrc
-rw-r--r--.  1 root root    100 dic 28 2013 .cshrc
drwx----- 14 root root   4096 dic  4 2018 modsecurity-2.9.3
-rw-r--r--.  1 root root 4307670 dic  4 2018 modsecurity-2.9.3.tar.gz
-rw-r--r--.  1 root root     129 dic 28 2013 .tcshrc
[root@localhost ~]#
```

9. Acceder a la carpeta descomprimida denominada “modsecurity-2.9.3”

10. Compilar el modulo modsecurity, con el comando: `./configure`

```
[root@localhost ~]# cd modsecurity-2.9.3
[root@localhost modsecurity-2.9.3]# ./configure
```

11. Instalar el modulo modsecurity, con el comando: `./install`

12. Realizar una copia del fichero de configuración, ejecutar:
`cp modsecurity.conf-recommended etc/httpd/conf.d/modsecurity.conf`

```
[root@localhost modsecurity-2.9.3]# cp modsecurity.conf-recommended /etc/httpd/conf.d/modsecurity.conf
```

13. Cargar el módulo de seguridad en el fichero de configuración del servicio httpd.

```
GNU nano 2.3.1                               Fichero: /etc/httpd/conf/httpd.conf
#
# Load config files in the "/etc/httpd/conf.d" directory, if any.
IncludeOptional conf.d/*.conf

LoadModule security2_module modules/mod_security2.so
<IfModule security2_module>
    Include conf.d/modsecurity.conf
</IfModule>
```

14. Instalar el paquete `mod_security_crs`:

```
[root@proxy-waf ~]# yum install mod_security_crs -y
Complementos cargados:fastestmirror
Loading mirror speeds from cached hostfile
 * base: mirror.esepoch.edu.ec
 * extras: mirror.esepoch.edu.ec
 * updates: mirror.esepoch.edu.ec
Resolviendo dependencias
--> Ejecutando prueba de transacción
---> Paquete mod_security_crs.noarch 0:2.2.9-1.el7 debe ser instalado
--> Resolución de dependencias finalizada

Dependencias resueltas

=====
Package                               Arquitectura                               Versión
=====
Instalando:
mod_security_crs                       noarch                                     2.2.9-1.el7

Resumen de la transacción
=====
Instalar 1 Paquete

Tamaño total de la descarga: 91 k
Tamaño instalado: 375 k
Downloading packages:
mod_security_crs-2.2.9-1.el7.noarch.rpm
Running transaction check
Running transaction test
```

15. Activar el módulo mod_security. Editar el fichero /etc/httpd/conf.d/mod_security.conf, buscar la directiva SecRuleEngine y configurar con los siguientes valores según nuestras necesidades:

on – Reglas Activadas

off – Reglas desactivadas

DetectionOnly – Solo intercepta y hace un log de la transacción

```
GNU nano 2.3.1                               Fichero: /etc/httpd/conf.d/mod_security.conf
<IfModule mod_security2.c>
# ModSecurity Core Rules Set configuration
  IncludeOptional modsecurity.d/*.conf
  IncludeOptional modsecurity.d/activated_rules/*.conf

# Default recommended configuration
SecRuleEngine On
SecRequestBodyAccess On
SecRule REQUEST_HEADERS:Content-Type "text/xml" \
  "id:'200000',phase:1,t:none,t:lowercase,pass,nolog,ctl:requestBodyProcessor=XML"
SecRequestBodyLimit 13107200
SecRequestBodyNoFilesLimit 131072
SecRequestBodyInMemoryLimit 131072
SecRequestBodyLimitAction Reject
SecRule REQBODY_ERROR "!@eq 0" \
  "id:'200001', phase:2,t:none,log,deny,status:400,msg:'Failed to parse request body.',logdata:'{reqbody_error_msg}',severity:2"
SecRule MULTIPART_STRICT_ERROR "!@eq 0" \
  "id:'200002',phase:2,t:none,log,deny,status:44,msg:'Multipart request body \
failed strict validation: \
PE %{REQBODY_PROCESSOR_ERROR}, \
BQ %{MULTIPART_BOUNDARY_QUOTED}, \
BW %{MULTIPART_BOUNDARY_WHITESPACE}, \
DB %{MULTIPART_DATA_BEFORE}, \
DA %{MULTIPART_DATA_AFTER}, \
HF %{MULTIPART_HEADER_FOLDING}, \
```

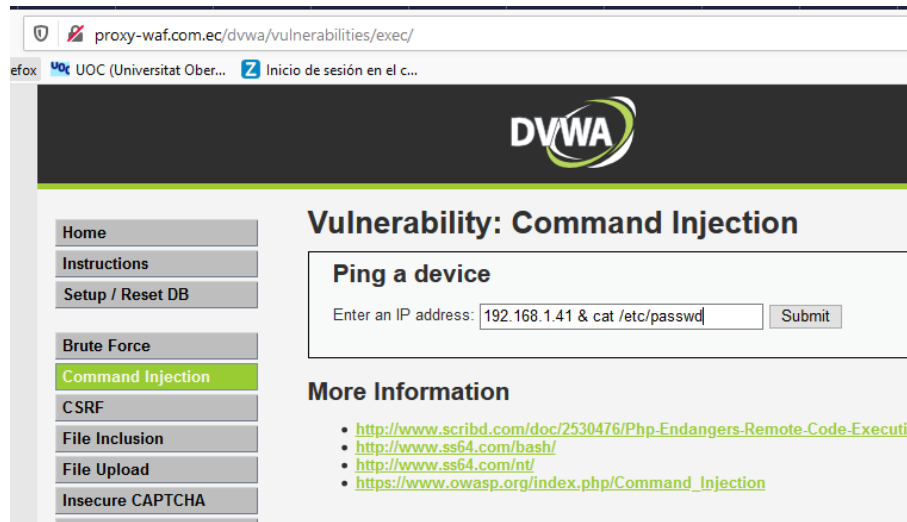
16. Comprobar que estén configuradas reglas del módulo mod_security_crs

```
[root@proxy-waf ~]# cd /etc/httpd/modsecurity.d/activated_rules/
[root@proxy-waf activated_rules]# ls
modsecurity_35_bad_robots.data          modsecurity_crs_23_request_limits.conf    modsecurity_crs_45_trojans.conf
modsecurity_35_scanners.data           modsecurity_crs_30_http_policy.conf       modsecurity_crs_47_common_exceptions.conf
modsecurity_40_generic_attacks.data    modsecurity_crs_35_bad_robots.conf        modsecurity_crs_48_local_exceptions.conf.example
modsecurity_50_outbound.data           modsecurity_crs_40_generic_attacks.conf    modsecurity_crs_49_inbound_blocking.conf
modsecurity_50_outbound_malware.data   modsecurity_crs_41_sql_injection_attacks.conf modsecurity_crs_50_outbound.conf
modsecurity_crs_20_protocol_violations.conf modsecurity_crs_41_xss_attacks.conf       modsecurity_crs_59_outbound_blocking.conf
modsecurity_crs_21_protocol_anomalies.conf modsecurity_crs_42_tight_security.conf     modsecurity_crs_60_correlation.conf
```

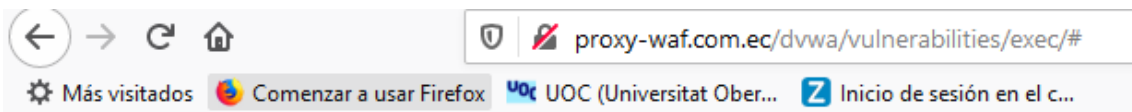
Activar Wi
Ve a Configuración

17. Reiniciar el servicio httpd con el comando `systemctl restart httpd`, y comprobar en los logs que se estén registrando en la consola del terminal correctamente los ataques, con el siguientes comando: `tail -f /var/log/httpd/error_log`.

Ejecutar en el navegador una prueba de inyección de comando:



Se obtiene el siguiente resultado:



Forbidden

You don't have permission to access /dvwa/vulnerabilities/exec/ on this server.

Y en los logs nos muestra el siguiente mensaje:

```
[Thu Oct 24 16:02:14.217763 2019] [:error] [pid 11030] [client 192.168.101.100:7444] [client 192.168.101.100] ModSecurity: Access denied with code 403 (phase 2). Pattern match "(?:\\\\\\\\b(?:\\\\\\\\.(?:ht(?:access|passwd|group)|www_?acl)|global\\\\\\\\.asa|http\\\\\\\\.conf|boot\\\\\\\\.ini)\\\\\\\\b\\\\\\\\/\\\\\\\\/etc\\\\\\\\/)" at ARGS:ip. [file "/etc/httpd/modsecurity.d/activated_rules/modsecurity_crs_40_generic_attacks.conf"] [line "205"] [id "950005"] [rev "3"] [msg "Remote File Access Attempt"] [data "Matched Data: /etc/ found within ARGS:ip: 192.168.1.41 & cat/etc/passwd"] [severity "CRITICAL"] [ver "OWASP CRS/2.2.9"] [maturity "9"] [accuracy "9"] [tag "OWASP_CRS/WEB_ATTACK/FILE_INJECTION"] [tag "WASC/33"] [tag "OWASP_TOP_10/A4"] [tag "PCI/6.5.4"] [hostname "proxy-waf.com.ec"] [uri "/dvwa/vulnerabilities/exec/"] [unique_id "XbTRVqCp7RsZP1QngyD@qAAAAT"], referer: http://proxy-waf.com.ec/dvwa/vulnerabilities/exec/
```

Activar Windows
Ve a Configuración para activar Windows

```
[Thu Oct 24 16:02:14.217763 2019] [:error] [pid 11030] [client
192.168.101.100:7444] [client 192.168.101.100] ModSecurity: Access denied
with code 403 (phase 2). Pattern match
"(?:\\b(?:\\.(?:ht(?:access|passwd|group)|www_?acl)|global|\\|asa|httpd|\\.con
f|boot|\\.ini)|\\|\\|\\|\\|Vetc|\\|\\|\\|\\|V)"
at ARGS:ip. [file
"/etc/httpd/modsecurity.d/activated_rules/modsecurity_crs_40_generic_at
tacks.conf"] [line "205"] [id "950005"] [rev "3"] [msg "Remote File Access
Attempt"] [data "Matched Data: /etc/ found within ARGS:ip: 192.168.1.41 &
cat/etc/passwd"] [severity "CRITICAL"] [ver "OWASP_CRS/2.2.9"] [maturity "9"]
[accuracy "9"] [tag "OWASP_CRS/WEB_ATTACK/FILE_INJECTION"] [tag
"WASCTC/WASC-33"] [tag "OWASP_TOP_10/A4"] [tag "PCI/6.5.4"] [hostname
"proxy-waf.com.ec"] [uri "/dvwa/vulnerabilities/exec/"] [unique_id
"XbIRVqCp7RsZP1QrgyD@qQAAAAI"], referer: http://proxy-
waf.com.ec/dvwa/vulnerabilities/exec/
```

Nos muestra el log, la fecha de acceso, la dirección IP del cliente, la regla que se esta aplicando, el id y la linea donde esta configurado la regla, etc.

- **Instalación y configuración de WAF Vulture**

Requerimientos necesarios para implementar WAF Vulture:

	Vulture
Sistema Operativo	FreeBSD 12.0-Release
Interfaz	Minimal, sin entorno gráfico
Interza Web para administrar y configurar	SI
Tamaño en disco duro	50 GB
Memoria RAM	2 GB
Procesadores	2
Paquete	apache2
Modulo	mod_secutiry

Modulo	mod_svm (Machine learning basado en Support Vector Machines)
Modulo	mod_defender
Paquete	mod_secutiry-crs

Implementación del WAF Vulture:

La implementación del WAF Vulture requiere llevar acabo los siguientes pasos importantes para su instalación:

1. Instalar el sistema operativo BreeBSD 12.0-Release en una máquina virtual. Se recomienda tener las siguientes particiones:

Swap	2 GB
Raíz /	6 GB
/home	2 GB
/var	el espacio necesario para almacenar logs

2. Una vez instalado el sistema operativo, actualizar el sistema operativo.
3. Descargar el script de instalación (contiene todos los prerequisites para instalar correctamente Vulture) del repositorio oficial: https://dl.vultureproject.org/vulture_installer_v22.sh. Una vez descargado damos permiso de ejecución con el comando `chmod +x vulture_installer_v22.sh`.
4. Ejecutamos el script con el siguiente comando:

```

root@vulture:~ # /bin/sh ./vulture_installer_v22.sh
Verifying archive integrity... 100% All good.
Uncompressing Vulture installer 100%
Updating FreeBSD repository catalogue...
FreeBSD repository is up to date.
All repositories are up to date.
Updating FreeBSD repository catalogue...
FreeBSD repository is up to date.
All repositories are up to date.
█

```


5. Una vez instalado todos los prerequisites, iniciar el proceso de arranque con el siguiente comando:

```
root@vulture:~ # /home/vlt-gui/env/bin/python2.7 /var/bootstrap/bootstrap.py
```

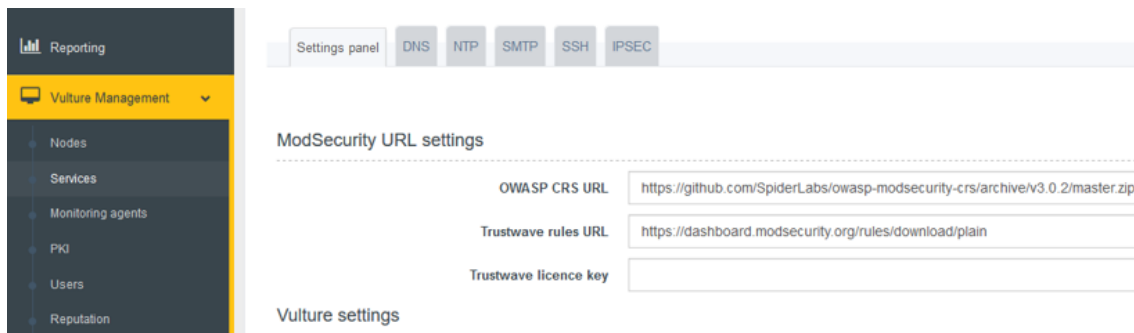
6. Continuar con el proceso de instalación, solicitará algunos datos como por ejemplo: configuración del teclado, configuración de red y proxy, crear un nuevo cluster, proporcionar una dirección de correo electrónico válida. Para continuar con el proceso de instalación, llegará al correo electrónico proporcionado una llave de registro, la cual debe ser insertada en la consola.

```
Please check your mailbox for the registration key.  
Please provide your Vulture registration key: [REDACTED]  
Validating registration key...  
Downloading Vulture LIBS...  
[+] Updating Vulture-LIBS...  
ELF ldconfig path: /lib /usr/lib /usr/lib/compat /usr/local/lib /usr/local/lib/i  
psec /usr/local/lib/mysql /usr/local/lib/perl5/5.38/mach/CORE  
32-bit compatibility ldconfig path: /usr/lib32  
[*] Update of Vulture-LIBS ended successfully  
Downloading Vulture...  
  
PKI Configuration  
Country: [REDACTED]
```

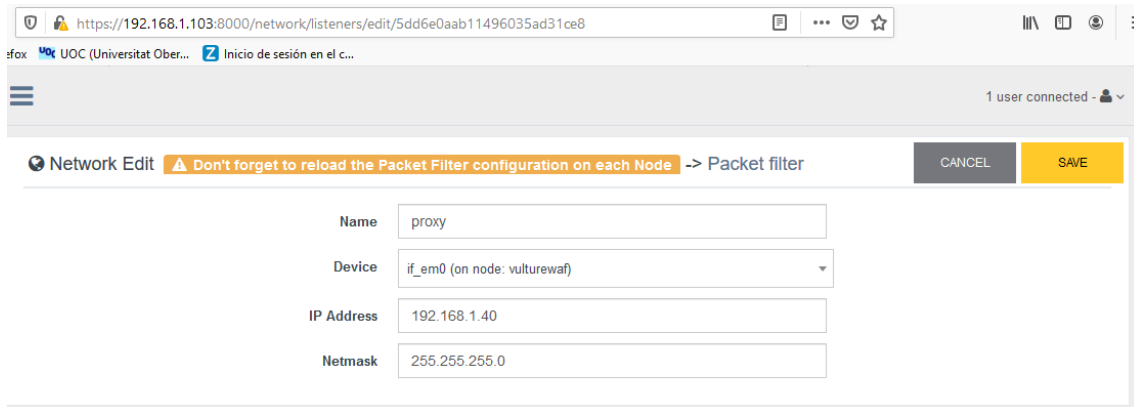
7. Seguir con la instalación, indicar las iniciales del país origen. El sistema mostrará un usuario denominado vlt-adm con su respectiva clave para poder acceder por SSH al sistema

```
vlt-adm SSH account has been created with the following password: [REDACTED]  
Don't forget to change it after installation.  
[REDACTED]
```

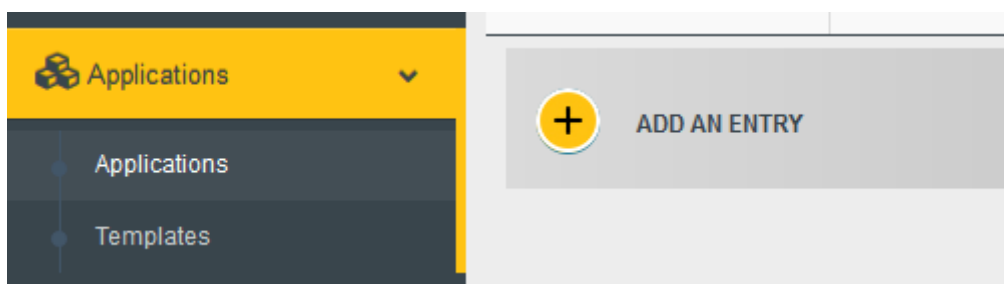
8. Si no se tiene ningún error durante el proceso de instalación, el WAF está instalado correctamente y nos muestra la siguiente pantalla:



- Una vez importado las reglas OWASP, proceder a configurar el WAF para que trabaje en modo proxy inverso, para ello dar clic en la opción Network – Listener. Clic en agregar, nos aparecerá las siguientes opciones: nombre, device (tarjeta de red que están disponibles), Ip adres (Dirección Ip del Listener para el proxy inverso), Netmask (Máscara de red de la dirección IP).



- Una vez creado la dirección IP de escucha del proxy inverso, proceder a crear una nueva entrada en la opción de Aplicaciones.



- Llenar algunos parámetros en la pestaña de Backend como por ejemplo: El template, tipo de aplicación, la dirección IP del aplicativo web.

https://192.168.1.103:8000/application/edit/5dd6e117b11496035ad31d01

UOC (Universitat Ober... Inicio de sesión en el c...

Application backend settings

Portal template: My Template

Application type: Plain Text Web Application

Private URI: http://192.168.1.41/

Timeout: 60

Preserve incoming Host Header:

Disable connection reuse:

Send KEEP_ALIVE messages to backend:

TTL of inactive connection: 30

Advanced settings

Send Proxy HTTP headers to backend:

Override backend's HTTP errors:

Rewrite Cookie Path:

Accept Web-Socket Protocol Upgrade:

15. En la pestaña Network, seleccionar la dirección ip de escucha y colocar el puerto TCP local, para que pueda operar en proxy inverso.

Application -> Edit 1 user connected

Change settings for application "Proxy" CANCEL SAVE

Internet Backend Network Security Logs Request Headers Response Headers Content Rewriting Authentication SSO Forward

Advanced SSO Forward Custom configuration

IP Address	Local TCP port	Translated Public TCP port	SSL Profile	Action
vulturewaf - 192.168.1.40	80			

+ ADD AN ENTRY

16. En la pestaña Security, agregar las reglas de OWASP. O todas las reglas necesarias para proteger el aplicativo web.

WAF Policy settings

Protect application with WAF profile

Enable learning mode

WAF Ruleset settings

Choose the Rules Set to use

URL specific Rules Set to use

ADD AN ENTRY

17. Seleccionar el tipo de perfil y el nivel en la pestaña de logs.

Application -> Edit 1 user connected

Change settings for application "Proxy"

Internet Backend Network Security **Logs** Request Headers Response Headers Content Rewriting Authentication SSO Forward

Advanced SSO Forward Custom configuration

Log Profile

Log Level

18. Guardar los cambios realizados, dar clic en iniciar aplicación. Si toda esta configurado correctamente, pasará el estado de la aplicación creada de **Stopped** a **Started**.

Firefox UOC (Universitat Ober... Inicio de sesión en el c...

Applications 1 user connected

Success
Listener successfully started

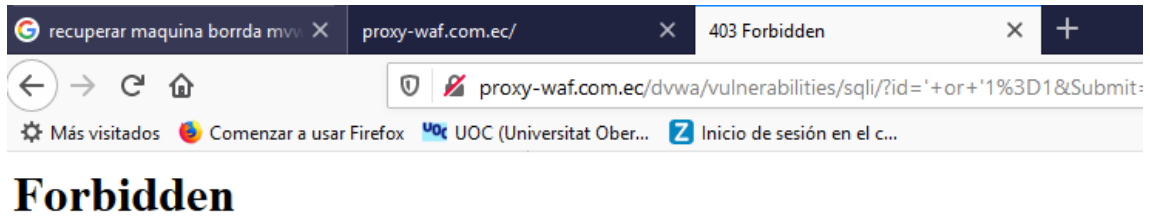
vulturewaf

Address	Port	Status	Action
192.168.1.40	80	Started	<input type="button" value=" "/> <input type="button" value="⬇"/>

Application	Enabled	Model	Tags	Authentication	Listener	Public URL	Private URL	Action
Proxy	✓	✗	None	✗	192.168.1.40:80	http://proxy-waf.com.ec/	http://192.168.1.41/	<input type="button" value="📄"/> <input type="button" value="🗑"/>

ADD AN ENTRY

19. Realizar algún ataque al WAF, y nos muestra la siguiente pantalla como resultado:



20. Proceder a revisar los logs que se generan, en el sistema WAF.

```
[Thu Nov 21 21:52:10.406021 2019] [:error] [pid 18803:tid 34385541120] [client 192.168.1.148:4260] [client 192.168.1.148] ModSecurity: Access denied with code 403 (phase 2). Operator GE matched 18 at TX:inbound_anomaly_score. [file "/home/vlt-sys/Engine/conf/192.168.1.40-80.conf"] used on line 710 of "/home/vlt-sys/Engine/conf/192.168.1.40-80.conf" [line "14"] [id "91"] [msg "Inbound anomaly score 49 exceeded threshold 18"] [hostname "proxy-waf.com.ec"] [uri "/dvwa/vulnerabilities/sqli/"] [unique_id "XdcHCof0seB240y2d5YtHgAAABg"], referer: http://proxy-waf.com.ec/dvwa/vulnerabilities/sqli/
[Thu Nov 21 21:52:10.406564 2019] [:error] [pid 18803:tid 34385541120] [client 192.168.1.148:4260] [client 192.168.1.148] ModSecurity: Warning. Operator GE matched 18 at TX:inbound_anomaly_score. [file "/home/vlt-sys/Engine/conf/modsec/5dd6e95bb11496027004b5a5.rules"] [line "73"] [id "980130"] [msg "Inbound Anomaly Score Exceeded (Total Inbound Score: 49 - SQLI=28, XSS=, RFI=, LFI=, RCE=, PHPI=, HTTP=, SSS=): "] [tag "event-correlation"] [hostname "proxy-waf.com.ec"] [uri "/dvwa/vulnerabilities/sqli/"] [unique_id "XdcHCof0seB240y2d5YtHgAAABg"], referer: http://proxy-waf.com.ec/dvwa/vulnerabilities/sqli/
```

- **Instalación y configuración de WAF Vulture**

Requerimientos necesarios para implementar WAF Waf2Py:

	Waf2Py
Sistema Operativo	Debian 9.3
Interfaz	Minimal, sin entorno gráfico
Interza Web para administrar y configurar	Si

Tamaño en disco duro	50 GB
Memoria RAM	2 GB
Procesadores	2
Paquete	apache2
Modulo	mod_security v3
Modulo	Web2Py 2.17.2
Modulo	Modsecurity Nginx connector
Paquete	mod_security-crs
Paquete	openresty-1.13.6.2

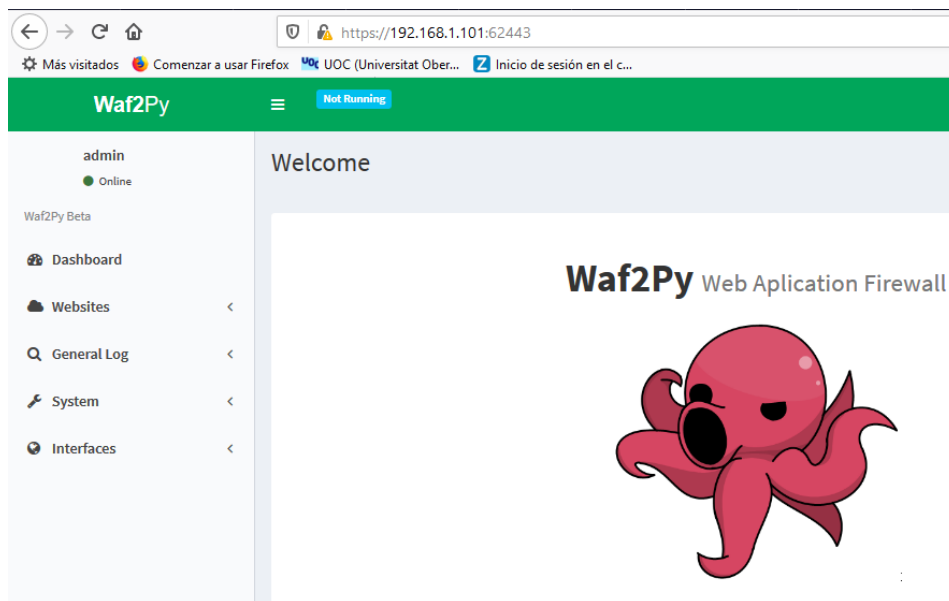
Implementación del WAF Waf2Py:

La implementación del WAF Waf2Py requiere llevar a cabo los siguientes pasos importantes para su instalación:

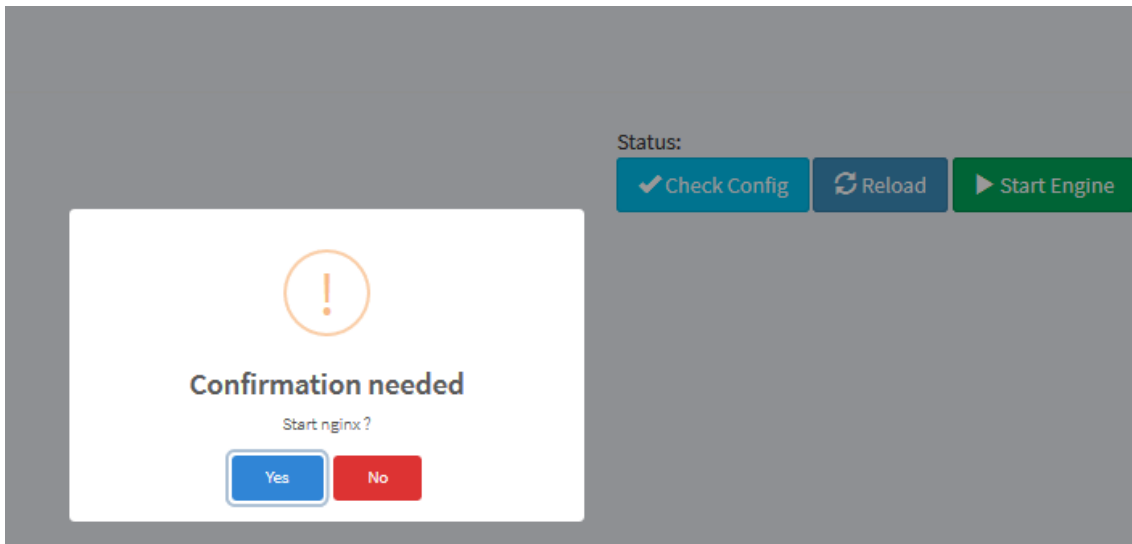
1. Instalar el sistema operativo Debian 9.3 en una máquina virtual.
2. Una vez instalado el sistema operativo, actualizar el sistema operativo.
3. Descargar el script de instalación (contiene todos los prerequisites para instalar correctamente Waf2Py) del repositorio oficial: <https://github.com/ITSec-Chile/Waf2Py>. Una vez descargado ingresar a la carpeta denominada Waf2Py, dar permiso de ejecución con el comando `chmod +x waf2py_installer.sh`
4. Ejecutar el script con el siguiente comando: `./waf2py_installer.sh`
5. Solicitará llenar algunos datos como: país, provincia, ciudad, nombre empresa, correo electrónico, clave del usuario admin para ingresar al sistema por interfaz web.

```
Country Name (2 letter code) [AU]:EC
State or Province Name (full name) [Some-State]:IMBABURA
Locality Name (eg, city) []:IBARRA
Organization Name (eg, company) [Internet Widgits Pty Ltd]:UOC
Organizational Unit Name (eg, section) []:TICS
Common Name (e.g. server FQDN or YOUR name) []:
Email Address []:gabrielcime@live.com
```

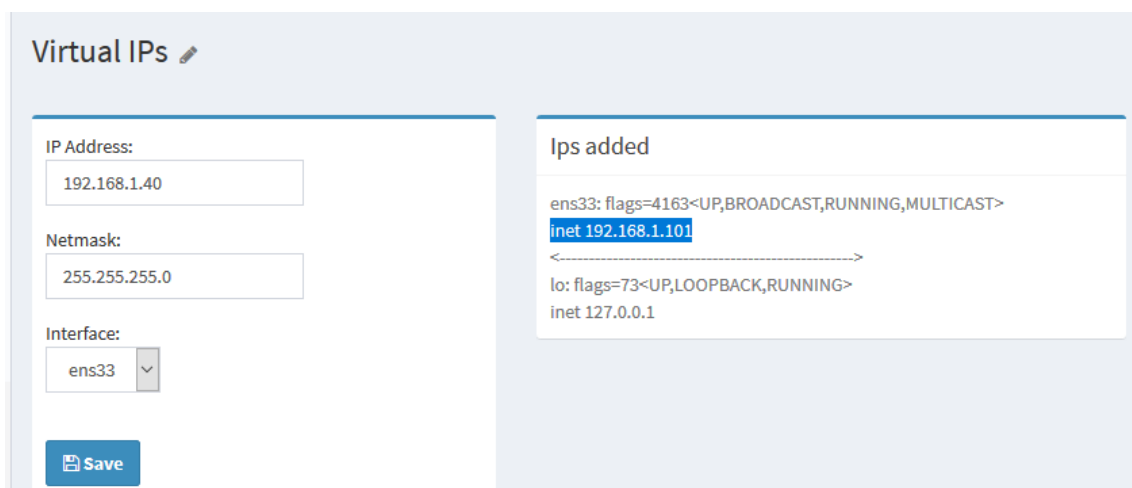
6. Una vez descargado e instalado el WAF correctamente, se podrá ingresar al sistema WAF por un navegador web, colocando la dirección Ip seguido del puerto 62443



7. Dar clic en la pestaña "Manage Engine", y seleccionar la opción "Start Engine", para inicializar el servicio Nginx.



8. Crear una ip virtual, para que le WAF trabaje como proxy inverso: agregar la dirección IP del proxy, la máscara de red correspondiente a la dirección IP, y seleccionar la interfaz.




9. Dar clic en la pestaña "New Website", colocar un nombre a la aplicación y el url del aplicativo.

Add a new application +

Add a new Application

Application Name

Application url

 Save

10. Dar clic en la opción “Websites”, y luego clic en “Websites Disabled”.

Waf2Py Running but not Listening admin






admin Online

Waf2Py Beta

- Dashboard
- Websites
- General Log
- System
- Interfaces

Websites Home > Websites

Websites Running (0) Websites Disabled (1)

Name	Application	Listens to	Real web server	Logs	Status	Mode	Actions
tests	proxy-waf.com.ec	None			Disabled	Defend	   

11. Configurar los datos correspondientes, como la dirección IP virtual, la dirección IP del aplicativo Web.

Waf2Py Running but not Listening admin






admin Online

Waf2Py Beta

- Dashboard
- Websites
- Websites Running
- New Website

Websites Home > Websites

Websites Running (0) Websites Disabled (1)

Name	Application	Listens to	Real web server	Logs	Status	Mode	Actions
tests	proxy-waf.com.ec	192.168.1.40	192.168.1.41		Disabled	Defend	   

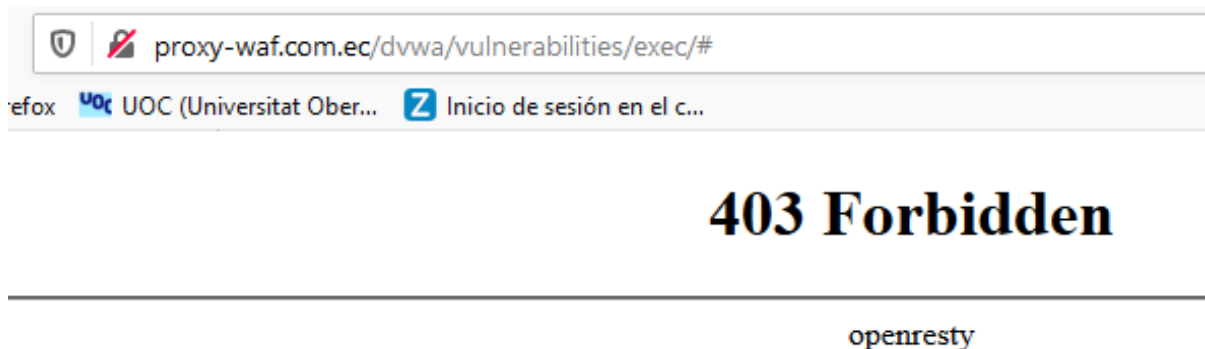
12. Guardar los cambios y dar clic en la opción iniciar, cambiará el estado de la aplicación de **Disabled** a **Enabled**.

Websites 🌩

Websites Running (1) Websites Disabled (0)

Name	Application	Listens to	Real web server	Logs	Status	Mode
tests	proxy-waf.com.ec	192.168.1.40	192.168.1.41:80	🔍	Enabled	Defend ⌵

13. Realizar algún ataque al WAF, y nos muestra la siguiente pantalla como resultado:



14. Para ver el monitoreo de los logs, dar clic en la pestaña de logs del aplicativo creado.

proxy-waf.com.ec Logs Home > proxy-waf.com.ec Logs

Summary Attacks Manual Exclusion Rules Excluded Download Logs Access Logs **Error Logs** Debug Logs

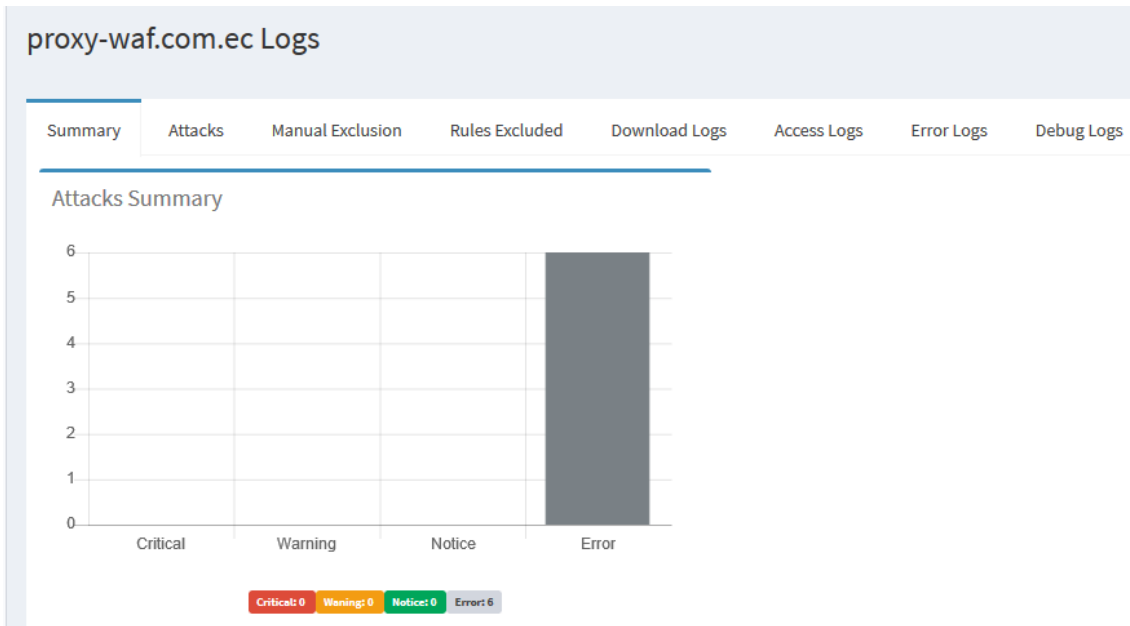
Error Logs

Showing last 300 logs 📄

Show entries Search:

#	Logs
1	2019/11/19 21:12:34 [error] 9443#0: *134 [client 192.168.1.148] ModSecurity: Access denied with code 403 (phase 2). [file "/opt/waf/nginx/etc/modsec_rules/proxy-waf.com.ec/enabled_rules/REQUEST-942-APPLICATION-ATTACK-SQLI.conf"] [line "45"] [id "942100"] [rev ""] [msg ""] [data ""] [severity "0"] [ver "OWASP_CRS/3.2.0"] [maturity "0"] [accuracy "0"] [tag "application-multi"] [tag "language-multi"] [tag "platform-multi"] [tag "attack-sqli"] [tag "paranoia-level/1"] [tag "OWASP_CRS"] [tag "OWASP_CRS/WEB_ATTACK/SQL_INJECTION"] [tag "WASCTC/WASC-19"] [tag "OWASP_TOP_10/A1"] [tag "OWASP_AppSensor/CIE1"] [tag "PCI/6.5.2"] [hostname "192.168.1.148"] [uri "/dvwa/vulnerabilities/sqli/"] [unique_id "15742159545.017382"] [ref ""], client: 192.168.1.148, server: proxy-waf.com.ec, request: "GET /dvwa/vulnerabilities/sqli/?id=%27+or+%271%3D1&Submit=Submit HTTP/1.1", host: "proxy-waf.com.ec", referer: "http://proxy-waf.com.ec/dvwa/vulnerabilities/sqli/"

15. Este WAF tiene la opción de ver gráficamente los logs que se generan:



- **Instalación y configuración de WAF Nginx**

Requerimientos necesarios para implementar WAF ModSecurity para NGINX + OWASP ModSecurity CRS (Core Rule Set):

	NGINX ModSecurity
Sistema Operativo	Centos 7
Interfaz	Minimal, sin entorno gráfico
Interza Web para administrar y configurar	NO
Tamaño en disco duro	50 GB
Memoria RAM	2 GB
Procesadores	2
Paquete	nginx-1.15.12
Modulo	mod_security

Modulo	owasp-modsecurity-crs
--------	-----------------------

Implementación del WAF ModSecurity para NGINX + OWASP ModSecurity CRS :

La implementación del WAF requiere llevar a cabo los siguientes pasos importantes para su instalación:

1. Instalar el sistema operativo centos 7 en una máquina virtual, una vez instalado actualizar el sistema operativo.
2. Instalar las siguientes dependencias:

```
yum groupinstall -y "Development Tools"
```

```
yum install -y httpd httpd-devel pcre pcre-devel libxml2 libxml2-devel curl curl-devel openssl openssl-devel
```

3. Descargar el ModSecurity para Nginx.

```
cd /usr/src
```

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

4. Compilar el ModSecurity, ejecutar los siguientes comandos:

```
cd ModSecurity
```

```
sed -i '/AC_PROG_CC/a\AM_PROG_CC_C_O' configure.ac
```

```
sed -i '1 \iAUTOMAKE_OPTIONS = subdir-objects' Makefile.am
```

```
./autogen.sh
```

```
./configure --enable-standalone-module --disable-mlogc
```

```
make
```

5. Una vez instalado ModSecurity para Nginx, descargar y descomprimir el paquete nginx-1.15.12

```
cd /usr/src
```

```
wget https://nginx.org/download/nginx-1.10.3.tar.gz
```

```
tar -zxvf nginx-1.15.12.tar.gz && rm -f nginx-1.15.12.tar.gz
```

6. Agregar un grupo denominado nginx, y agregar el usuario nginx al grupo nginx.

```
groupadd -r nginx
```

```
useradd -r -g nginx -s /sbin/nologin -M nginx
```

```
[root@localhost src]# groupadd -r nginx
[root@localhost src]# useradd -r -g nginx -s /sbin/nologin -M nginx
[root@localhost src]# _
```

7. Compilar Nginx, habilitando los módulos ModSecurity y SSL.

```
cd nginx-1.10.3/
```

```
./configure --user=nginx --group=nginx --add-  
module=/usr/src/ModSecurity/nginx/modsecurity --with-http_ssl_module
```

```
make
```

```
make install
```

8. Modificar del fichero de configuración de nginx “/usr/local/nginx/conf/nginx.conf”, el usuario por default de Nginx.

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

```
[root@localhost nginx-1.15.12]# sed -i "s/#user nobody;/user nginx nginx;/" /usr/local/nginx/conf/nginx.conf
[root@localhost nginx-1.15.12]# _
```

9. Comprobar con un test, si las configuraciones realizadas no tienen ningún error, ejecutar el siguiente comando /usr/local/nginx/sbin/nginx -t, y se debe obtener el siguiente resultado:

```
[root@localhost nginx]# /usr/local/nginx/sbin/nginx -t
nginx: the configuration file /usr/local/nginx/conf/nginx.conf syntax is ok
nginx: configuration file /usr/local/nginx/conf/nginx.conf test is successful
[root@localhost nginx]# _
```

10. Configurar un archivo systemd para Nginx, para ello se crea un fichero denominado nginx.service con el siguiente contenido:

```
GNU nano 2.3.1 Fichero: /lib/systemd/system/nginx.service
[Service]
Type=forking
ExecStartPre=/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
```

11. Guardar el fichero, y ya se podrá detener, empezar, o restaurar el demonio nginx.service:

```
systemctl start nginx.service
systemctl stop nginx.service
systemctl restart nginx.service
systemctl status nginx.service
```

12. Configurar Nginx en modo proxy inverso y habilitar los módulos ModSecurity, abrir el archivo de configuración de nginx localizado en la siguiente ruta /usr/local/nginx/conf/nginx.conf.

Agregar las siguientes líneas:

ModSecurityEnabled on; = habilita el modulo ModSecurity.

MosSecurityConfig modsec_includes.conf; = fichero donde están los directorios de las reglas de configuración.

Proxy_pass <http://192.168.1.41>; = Dirección Ip del servidor web.

```
GNU nano 2.3.1 Fichero: /usr/local/nginx/conf/nginx.conf

server {
    listen      80;
    server_name localhost;

    #charset koi8-r;

    #access_log logs/host.access.log main;

    location / {
        ModSecurityEnabled on;
        ModSecurityConfig modsec_includes.conf;
        #proxy_pass http://localhost:8011;
        proxy_pass http://192.168.1.41;
        #proxy_read_timeout 180s;
        root    html;
        index  index.html index.htm;
    }
}
```

13. Crear un archivo denominado modsec_includes.conf, en la ruta /usr/local/nginx/conf/, con el siguiente contenido:

```
GNU nano 2.3.1 Fichero: /usr/local/nginx/conf/modsec_includes.conf

include modsecurity.conf
include owasp-modsecurity-crs/crs-setup.conf
include owasp-modsecurity-crs/rules/*.conf
```

14. Realizar una copia de 2 ficheros denominados modsecurity.conf-recommended y unicode.mapping, en la siguiente ruta: /usr/local/nginx/conf/

```
[root@localhost nginx]# cp /usr/src/ModSecurity/modsecurity.conf-recommended /usr/local/nginx/conf/modsecurity.conf
[root@localhost nginx]# cp /usr/src/ModSecurity/unicode.mapping /usr/local/nginx/conf/
[root@localhost nginx]# _
```

15. Editar el fichero /usr/local/nginx/conf/modsecurity.conf y cambiar el parámetro SecRuleEngine DetectionOnly por SecRuleEngine On.

```
GNU nano 2.3.1 Fichero: /usr/local/nginx/conf/modsecurity.conf

# -- Rule engine initialization -----

# Enable ModSecurity, attaching it to every transaction. Use detection
# only to start with, because that minimises the chances of post-installation
# disruption.
#
SecRuleEngine On
```


16. Agregar las reglas OWASP ModSecurity CRS (Core Rule Set). Ejecutar los siguientes comandos:

```
cd /usr/local/nginx/conf
```

```
git clone https://github.com/SpiderLabs/owasp-modsecurity-crs.git
```

```
[root@localhost nginx]# cd /usr/local/nginx/conf
[root@localhost conf]# git clone https://github.com/SpiderLabs/owasp-modsecurity-crs.git
Cloning into 'owasp-modsecurity-crs'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 9947 (delta 0), reused 1 (delta 0), pack-reused 9943
Receiving objects: 100% (9947/9947), 3.18 MiB | 453.00 KiB/s, done.
Resolving deltas: 100% (7304/7304), done.
[root@localhost conf]#
```

```
cd owasp-modsecurity-crs
```

```
mv crs-setup.conf.example crs-setup.conf
```

```
[root@localhost conf]# cd owasp-modsecurity-crs/
[root@localhost owasp-modsecurity-crs]# ls
CONTRIBUTING.md CONTRIBUTORS.md crs-setup.conf.example CHANGES documentation INSTALL KNOWN_BUGS LICENSE README.md rules SECURITY.md util
[root@localhost owasp-modsecurity-crs]# mv crs-setup.conf.example crs-setup.conf
[root@localhost owasp-modsecurity-crs]# ls
CONTRIBUTING.md CONTRIBUTORS.md crs-setup.conf CHANGES documentation INSTALL KNOWN_BUGS LICENSE README.md rules SECURITY.md util
```

```
cd rules
```

```
[root@localhost owasp-modsecurity-crs]# cd rules/
[root@localhost rules]# ls
crawlers-user-agents.data REQUEST-900-EXCLUSION-RULES-BEFORE-CRS.conf.example RESPONSE-950-DATA-LEAKAGES.conf
iis-errors.data REQUEST-905-COMMON-EXCEPTIONS.conf RESPONSE-951-DATA-LEAKAGES-SQL.conf
java-classes.data REQUEST-910-IP-REPUTATION.conf RESPONSE-952-DATA-LEAKAGES-JAVA.conf
java-code-leakages.data REQUEST-911-METHOD-ENFORCEMENT.conf RESPONSE-953-DATA-LEAKAGES-PHP.conf
java-errors.data REQUEST-912-DOS-PROTECTION.conf RESPONSE-954-DATA-LEAKAGES-IIS.conf
lfi-os-files.data REQUEST-913-SCANNER-DETECTION.conf RESPONSE-959-BLOCKING-EVALUATION.conf
php-config-directives.data REQUEST-920-PROTOCOL-ENFORCEMENT.conf RESPONSE-980-CORRELATION.conf
php-errors.data REQUEST-921-PROTOCOL-ATTACK.conf RESPONSE-999-EXCLUSION-RULES-AFTER-CRS.conf.example
php-function-names-933150.data REQUEST-930-APPLICATION-ATTACK-LFI.conf restricted-files.data
php-function-names-933151.data REQUEST-931-APPLICATION-ATTACK-RFI.conf restricted-upload.data
php-variables.data REQUEST-932-APPLICATION-ATTACK-RCE.conf scanners-headers.data
REQUEST-900-EXCLUSION-RULES-BEFORE-CRS.conf.example REQUEST-933-APPLICATION-ATTACK-PHP.conf scanners-urls.data
REQUEST-901-INITIALIZATION.conf REQUEST-934-APPLICATION-ATTACK-NODEJS.conf scanners-user-agents.data
REQUEST-903-9001-DRUPAL-EXCLUSION-RULES.conf REQUEST-941-APPLICATION-ATTACK-XSS.conf scripting-user-agents.data
REQUEST-903-9002-WORDPRESS-EXCLUSION-RULES.conf REQUEST-942-APPLICATION-ATTACK-SQLI.conf sql-errors.data
REQUEST-903-9003-NEXTCLOUD-EXCLUSION-RULES.conf REQUEST-943-APPLICATION-ATTACK-SESSION-FIXATION.conf unix-shell.data
REQUEST-903-9004-DOKUWIKI-EXCLUSION-RULES.conf REQUEST-944-APPLICATION-ATTACK-JAVA.conf windows-powershell-commands.data
REQUEST-903-9005-CPANEL-EXCLUSION-RULES.conf REQUEST-949-BLOCKING-EVALUATION.conf
```

```
mv REQUEST-900-EXCLUSION-RULES-BEFORE-CRS.conf.example
REQUEST-900-EXCLUSION-RULES-BEFORE-CRS.conf
```

```
mv RESPONSE-999-EXCLUSION-RULES-AFTER-CRS.conf.example
RESPONSE-999-EXCLUSION-RULES-AFTER-CRS.conf
```

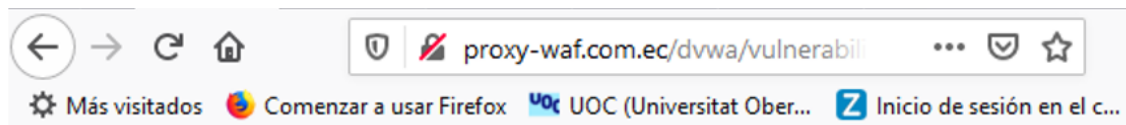
```
[root@localhost rules]# mv REQUEST-900-EXCLUSION-RULES-BEFORE-CRS.conf.example REQUEST-900-EXCLUSION-RULES-BEFORE-CRS.conf
[root@localhost rules]# mv RESPONSE-999-EXCLUSION-RULES-AFTER-CRS.conf.example RESPONSE-999-EXCLUSION-RULES-AFTER-CRS.conf
[root@localhost rules]#
```

17. Reiniciar el servicio nginx con el comando `systemctl restart nginx`, y comprobar en los logs que se estén registrando correctamente en la consola del terminal los ataques con el siguiente comando: `tail -f /usr/local/nginx/logs/error_log`.

```
[root@localhost rules]# systemctl restart nginx.service
[root@localhost rules]# systemctl status nginx.service
● nginx.service
   Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; vendor preset: disabled)
   Active: active (running) since vie 2019-11-22 16:04:17 -05; 5s ago
     Process: 28772 ExecStart=/usr/local/nginx/sbin/nginx -c /usr/local/nginx/conf/nginx.conf (code=exited, status=0/SUCCESS)
     Process: 28769 ExecStartPre=/usr/local/nginx/sbin/nginx -t -c /usr/local/nginx/conf/nginx.conf (code=exited, status=0/SUCCESS)
    Main PID: 28773 (nginx)
      CGroup: /system.slice/nginx.service
              └─28773 nginx: master process /usr/local/nginx/sbin/nginx -c /usr/local/nginx/conf/nginx.conf
                  └─28774 nginx: worker process

nov 22 16:04:16 localhost.localdomain systemd[1]: Starting nginx.service...
nov 22 16:04:17 localhost.localdomain nginx[28769]: nginx: the configuration file /usr/local/nginx/conf/nginx.conf syntax is ok
nov 22 16:04:17 localhost.localdomain nginx[28769]: nginx: configuration file /usr/local/nginx/conf/nginx.conf test is successful
nov 22 16:04:17 localhost.localdomain systemd[1]: Started nginx.service.
[root@localhost rules]#
```

18. Ejecutar en el navegador una prueba de inyección de comando, y se obtiene el siguiente resultado:



403 Forbidden

nginx/1.15.12

19. En los logs nos muestra el siguiente mensaje:

```
2019/11/22 23:35:12 [error] 2394#0: [client 192.168.1.148]
ModSecurity: Access denied with code 403 (phase 2). detected SQLi
using libinjection with fingerprint 's&s' [file
"/usr/local/nginx/conf/owasp-modsecurity-crs/rules/REQUEST-942-
APPLICATION-ATTACK-SQLI.conf"] [line "68"] [id "942100"] [msg
"SQL Injection Attack Detected via libinjection"] [data "Matched Data:
s&s found within ARGS:id: ' or '1=1" ] [severity "CRITICAL"] [ver
"OWASP_CRS/3.2.0"] [tag "application-multi"] [tag "language-multi"]
[tag "platform-multi"] [tag "attack-sqli"] [tag "paranoia-level/1"] [tag
"OWASP_CRS"] [tag
```

"OWASP_CRS/WEB_ATTACK/SQL_INJECTION" [tag
"WASCTC/WASC-19"] [tag "OWASP_TOP_10/A1"] [tag
"OWASP_AppSensor/CIE1"] [tag "PCI/6.5.2"] [hostname ""] [uri
"/dvwa/vulnerabilities/sqli/"] [unique_id
"AcAcAcA9AcOcAcAfA7WeAcAc"]

```
2019/11/22 23:35:12 [error] 2394#0: [client 192.168.1.148] ModSecurity: Access denied with code 403 (phase 2). detected SQLi using libinjection with fingerprint 's&s' [file "/usr/local/nginx/conf/owasp-modsecurity-crs/rules/REQUEST-942-APPLICATION-ATTACK-SQLI.conf"] [line "68"] [id "942100"] [msg "SQL Injection Attack Detected via libinjection"] [data "Matched Data: s&s found within ARGS:id: ' or '1=1'"] [severity "CRITICAL"] [ver "OWASP_CRS/3.2.0"] [tag "application-multi"] [tag "language-multi"] [tag "platform-multi"] [tag "attack-sqli"] [tag "paranoia-level/1"] [tag "OWASP_CRS"] [tag "OWASP_CRS/WEB_ATTACK/SQL_INJECTION"] [tag "WASCTC/WASC-19"] [tag "OWASP_TOP_10/A1"] [tag "OWASP_AppSensor/CIE1"] [tag "PCI/6.5.2"] [hostname "" ] [uri "/dvwa/vulnerabilities/sqli/"] [unique_id "AcAcAcA9AcOcAcAfA7WeAcAc"]
2019/11/22 23:35:12 [error] 2394#0: [client 192.168.1.148] ModSecurity: Audit log: Failed to lock global mutex: Permission denied [hostname "" ] [uri "/dvwa/vulnerabilities/sqli/"] [unique_id "AcAcAcA9AcOcAcAfA7WeAcAc"]
2019/11/22 23:35:12 [error] 2394#0: [client 192.168.1.148] ModSecurity: Audit log: Failed to unlock global mutex: Permission denied [hostname "" ] [uri "/dvwa/vulnerabilities/sqli/"] [unique_id "AcAcAcA9AcOcAcAfA7WeAcAc"]
```

Activar Windows
Ve a Configuración para activar Windows.