

# Guida All'istallazione di ModSecurity-crs on Nginx In container Docker

Lorenzo Mauriello

December 5, 2022

## Abstract

Guida completa all'istallazione di ModSecurity su Nginx, il flusso di operazioni è valido sui principali sistemi linux come Ubuntu e CentOS. Programmi necessari:

1. Ubuntu 20.04 lts (focal).
2. Docker 20.10.12.
3. ModSecurity-crs:nginx.
4. Modsecurity-crs master.

## Introduction

Prima di procedere bisogna con l'istallazione e la configurazione delle singole componenti bisogna scaricare le dipendenze necessarie:

```
1 $ sudo apt-get update
2 $ sudo apt-get upgrade
3 $ sudo apt-get install -y git
```

Il passaggio successivo all'istallazione delle dipendenze, possiamo installare Docker così da preparare l'ambiente di lavoro:

```
1 $ sudo apt install apt-transport-https ca-certificates curl software-properties-common
2 $ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
3 $ sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu focal stable"
4 $ sudo apt install docker-ce
```

*Attenzione* in questo modo abbiamo installato docker ma i comandi dovranno essere lacniati utilizzando *sudo*. Per ovviare al problema si possono sbloccare i privilegi al docker engine oppure avviare una sessione amministratore. P.S. possiamo verificare lo status di installazione di docker e la versione se lo desideriamo.

## ModSecurity-crs download and create container

Portiamoci nella */Home* e scarichiamo le nostre core rule set:

```
1 $ git clone https://github.com/coreruleset/coreruleset.git myCRS
2 $ cd myCRS
```

quando il download è completo e ci siamo spostati all'interno della nostra cartella possiamo modificare e configurare le nostre roules nella cartella roules come più vogliamo (Nel sito ufficiale sono riportati tutti gli esempi di utilizzo all'interno dei file **REQUEST -9xx - EXCLUTION -RULES**).

Effettuate queste operazioni siamo pronti per istanziare il nostro container Docker, quindi:

```
1 $ docker create --name modseccrs1 -p 80:80 -ti -e PARANOIA=4 -v /<absolute path to>/rules:/opt/owasp-crs/rules:ro --rm owasp/modsecurity-crs:nginx
```

Il comando appena eseguito crea un container permanente in pausa all'interno di docker, possiamo sempre in qualunque momento ricaricare le regole e riavviare nginx tramite il comando **docker exec -it <name container> nginx -s reload** (ci sarà un'appendice dedicata). A questo punto non ci resta che avviare il container:

```
1 $ docker start modseccrs1
```

Possiamo testare il funzionamento di nginx utilizzando un browser qualunque e scrivendo nella barra degli indirizzi: *http://localhost* (se nginx genera errore 502 vedere appendice errori). Per testare il funzionamento di ModSecurity proviamo ad effettuare un injection *http://localhost/?q=1 OR 1=1*, se tutto è andato bene dovremmo riscontrare un errore **403 forbidden**.

Per concludere, ogni modifica alle regole o alle caratteristiche di nginx possono essere effettuate accedendo al container tramite sessione ssh oppure tramite la creazione di una nuova immagine docker che sfrutta i DockerFile.

## Errori comuni e Consigli

**Errore 1** Potrebbe capitare che la configurazione standard di nginx sia sbagliata, ovvero non configurata adeguatamente per docker. L'errore 50x fa riferimento al così detto **resolver** che in questo caso viene configurato come *192.168.1.1*. In generale questo non dovrebbe generare alcun problema, ma alle volte potrebbero insorgere incompatibilità con la rete generata da docker engine; per risolvere basta configurare adeguatamente nginx modificando il file *default.conf* che nel container si trova in */etc/nginx/conf.d/default.conf* con il file in allegato. Nel caso il file allegato sia stato perso o non è reperibile al momento, riporto la sezione più importante:

```
1  listen 443 ssl;
2  listen      80;
3  listen  [::]:80;
4  server_name localhost;
5
6
7  #resolver 192.168.1.1 valid=5s;
8  set  $upstream http://localhost:80;
9
10 ssl_certificate /etc/nginx/conf/server.crt;
11 ssl_certificate_key /etc/nginx/conf/server.key;
12 ssl_ciphers
   ECDH+AESGCM:DH+AESGCM:ECDH+AES256:DH+AES256:ECDH+AES128:DH+AES:ECDH+3DES:DH+3DES:RSA+AESGCM:RSA+AES:
13 ssl_prefer_server_ciphers on;
14 ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
15 ssl_verify_client off;
```

La riga **7** va assolutamente commentata tramite il carattere '#', opzionalmente si possono aggiungere regole e migliorare la configurazione.

Per inserire il file all'interno del container basta avviare il comando:

```
1 $ sudo docker cp default.conf <name container>:/etc/nginx/conf.d
```

**È importante che il terminale sia aperto nella stessa cartella dove è stato lanciato il container e che anche il file sia nella stessa cartella.**

**Consiglio: come aggiornare le regole nel container** Come accennato nella guida per ottenere sicurezza diversa basta aggiornare le regole che sono state modificate, per far ciò si può procedere ricompilando l'intero container, variando quindi i livelli di "*paranoia*" e ricaricando i volumi delle regole, oppure con più semplicità cambiare i file di configurazione interni ad nginx e modsecurity, nel caso di interesse **variazione delle regole** basta eseguire il comando:

```
1 $ sudo docker cp REQUEST-9xx.conf <name container>:/opt/owasp-crs/rules
```

In modo da aggiornare i vecchi file con le nuove regole.