



# 27°

## Lab - AWS re/Start Malware de Firewall





# Protección contra Malware usando Netwrok AWS Firewall

A continuación, se muestra los objetivos del laboratorio:

- Actualizar un firewall de red de AWS
- Crear un grupo de reglas de firewall
- Verificar y probar que el acceso a los sitios maliciosos esté bloqueado

**Nota.** Un Malware es un software malicioso desarrollado por hacker, eg: virus, troyanos, spywares, adwares y ransomware. Mientras que los firewalls son como muros de seguridad físicos que se encuentran entre la red interna de la organización y cualquier red pública externa.

# Tarea 01



Empezaremos confirmando la accesibilidad a la instancia de EC2 llamada **TestInstance**. Esto mediante *Session Manager*. Una vez conectados, procedemos a descargar un Malware:

```
us-west-2.console.aws.amazon.com/systems-manager/session-manager/i-0359fae75d1b70674?region=us-west-2#

Session ID: 783c9c4b-e126-4728-9951-0a7c5c23e5f5-0aad4e9cc236606e4 Instance ID: i-0359fae75d1b70674 Terminate

sh-4.2$ cd ~
sh-4.2$ pwd
/home/ssm-user
sh-4.2$ wget http://malware.wicar.org/data/js_crypto_miner.html
--2024-01-31 06:24:50-- http://malware.wicar.org/data/js_crypto_miner.html
Resolving malware.wicar.org (malware.wicar.org)... 208.94.116.246, 2607:ff18:80:6::6a08
Connecting to malware.wicar.org (malware.wicar.org)|208.94.116.246|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 366 [text/html]
Saving to: 'js_crypto_miner.html'

100%[=====>] 366 --.-K/s in 0s

2024-01-31 06:24:50 (49.7 MB/s) - 'js_crypto_miner.html' saved [366/366]

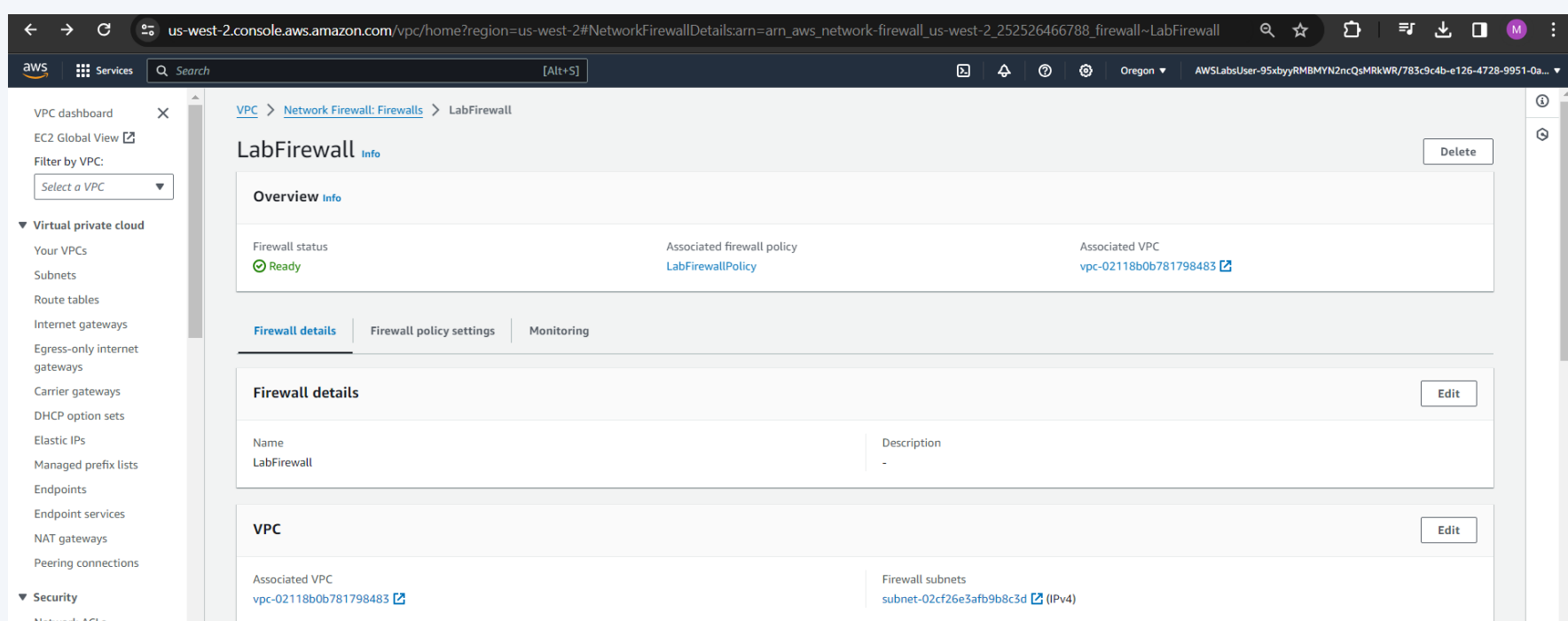
sh-4.2$ wget http://malware.wicar.org/data/java_jrel17_exec.html
--2024-01-31 06:25:01-- http://malware.wicar.org/data/java_jrel17_exec.html
Resolving malware.wicar.org (malware.wicar.org)... 208.94.116.246, 2607:ff18:80:6::6a08
Connecting to malware.wicar.org (malware.wicar.org)|208.94.116.246|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 129 [text/html]
Saving to: 'java_jrel17_exec.html'

100%[=====>] 129 --.-K/s in 0s

2024-01-31 06:25:01 (16.5 MB/s) - 'java_jrel17_exec.html' saved [129/129]

sh-4.2$ ls
java_jrel17_exec.html js_crypto_miner.html
sh-4.2$
```

Esto solo con fines académicos, después de ello, procederemos a inspeccionar el firewall de la red



# Tarea 01

---



Podemos ver que ya existe una política del firewall, la cual es la encargada de definir el comportamiento en una colección de grupos de reglas con estado (stateful) y sin estado (stateless), y otros ajustes. Asimismo, editamos las acciones sin estado predeterminadas:

The screenshot shows a dialog box titled "Stateless default actions" with a close button (X) in the top right corner. The dialog contains three sections:

- Fragmented packets:** Two radio button options. The first, "Use the same actions for all packets", is selected. The second is "Use different actions for full packets and fragmented packets".
- Rule action:** Three radio button options. The first, "Pass", is unselected. The second, "Drop", is unselected. The third, "Forward to stateful rule groups", is selected.
- Publish metrics - optional:** A sub-header followed by the text "Publish a custom Amazon CloudWatch metric to monitor the usage of your stateless rule groups." Below this is a checkbox labeled "Enable", which is currently unchecked.

At the bottom right of the dialog, there are two buttons: "Cancel" and "Save".

Ahora, procedemos a crear un grupo de reglas para el firewall, el cual te ayudará a bloquear el acceso a URLs maliciosos

# Tarea 01



## Esta es la configuración

VPC > Network Firewall: Rule groups > Create Network Firewall rule group

Step 1

Choose rule group type

Step 2

Describe rule group

Step 3

Configure rules

Step 4 - optional

Configure advanced settings

Step 5 - optional

Add tags

Step 6

Review and create

Choose rule group type [Info](#)

Network Firewall rule groups are either stateless or stateful. Stateless rule groups evaluate packets in isolation, while stateful rule groups evaluate them in the context of their traffic flow.

Rule group type

Rule group type

☒ Stateful rule group

Use stateful rule groups to inspect packets within the context of the traffic flow.

☐ Stateless rule group

Use stateless rule groups to inspect individual packets on their own, without the context of the traffic flow.

Rule group format

Suricata compatible rule string

Rule evaluation order [Info](#)

The way that your stateful rules are ordered for evaluation.

☐ Strict order - *recommended*

Rules are processed in the order that you define, starting with the first rule.

☒ Action order

Rules with a pass action are processed first, followed by drop, reject, and alert actions. This option was previously named **Default order**.

Describe rule group [Info](#)

Name and describe your rule group so you can easily identify it and distinguish it from other resources.

Rule group details

Name

Enter a name for the rule group that's unique within your stateful rule groups.

StatefulRuleGroup

The name must have 1-128 characters. Valid characters: a-z, A-Z, 0-9 and - (hyphen). The name can't start or end with a hyphen, and it can't contain two consecutive hyphens.

Description - optional

This description appears when you view this rule group's details. It can help you quickly identify what your rule group is used for.

Enter rule group description

The description can have 0-256 characters.

Capacity [Info](#)

The number of rules you expect to have in this rule group during its lifetime. You can't change capacity after rule group creation, so leave room to grow.

100

The capacity must be greater than or equal to 1 and less than 30,000.

Cancel

Previous

Next

Configure rules [Info](#)

An AWS Network Firewall rule group is a reusable set of criteria for inspecting and handling network traffic.

▶ Rule variables - optional [Info](#)

Define IP sets and ports as variables. These variables can be used within this rule group for standard stateful rules and Suricata compatible rule strings.

▶ IP set references - optional [Info](#)

An IP set reference is a variable used in your rules that refers to a resource associated with a list of IPs or CIDRs.

Suricata compatible rule string [Info](#)

Suricata is an open source network IPS that includes a standard rule-based language for traffic inspection.

Suricata compatible rule string

drop http \$HOME\_NET any -> \$EXTERNAL\_NET 80 (msg:"MALWARE custom solution"; flow: to\_server,established; classtype:trojan-activity; sid:2002001; content:"/data/js\_crypto\_miner.html";http\_uri; rev:1;)

Copy rules

Cancel

Previous

Next

Las dos reglas de Suricata que se han añadido ahora **bloquean el tráfico** que coincide con las **URLs** http\_uri contents /data/js\_crypto\_miner.html y http\_uri contents /data/js\_crypto\_miner.html cuando el tráfico se inicia desde el LabVPC a la red pública.

< SWIPE ≡

# Tarea 01



Ahora, asignamos el grupo de reglas a un firewall de red

VPC > Network Firewall: Firewalls > LabFirewall > Add my own stateful rule groups

### Add unmanaged stateful rule groups [Info](#)

Select and add the stateful rule groups that you want in your firewall policy.

**i** A firewall policy can be associated with multiple firewalls. Modifying a firewall policy affects all firewalls that reference it.  
To use rule groups that are managed for you, see [AWS Partner Network \(APN\) integrations](#).

**Stateful rule group (1/1)** Create rule group Refresh

< 1 > Settings

| <input checked="" type="checkbox"/> | Name              |
|-------------------------------------|-------------------|
| <input checked="" type="checkbox"/> | StatefulRuleGroup |

Cancel Add stateful rule group

Finalmente, validamos la solución. Y vemos que al momento de intentar descargar una parte del malware, ha sido bloqueado por el firewall de la red

```
us-west-2.console.aws.amazon.com/systems-manager/session-manager/i-0359fae75d1b70674?region=us-west-2
Session ID: 783c9c4b-e126-4728-9951-0a7c5c23e5f5-040d528b3e9ed6cb5 Instance ID: i-0359fae75d1b70674 Terminate

sh-4.2$ wget http://malware.wicar.org/data/js_crypto_miner.html
--2024-01-31 06:56:48-- http://malware.wicar.org/data/js_crypto_miner.html
Resolving malware.wicar.org (malware.wicar.org)... 208.94.116.246, 2607:ff18:80:6::6a08
Connecting to malware.wicar.org (malware.wicar.org)|208.94.116.246|:80... connected.
HTTP request sent, awaiting response... ^C
sh-4.2$ wget http://malware.wicar.org/data/java_jre17_exec.html
--2024-01-31 06:58:01-- http://malware.wicar.org/data/java_jre17_exec.html
Resolving malware.wicar.org (malware.wicar.org)... 208.94.116.246, 2607:ff18:80:6::6a08
Connecting to malware.wicar.org (malware.wicar.org)|208.94.116.246|:80... connected.
HTTP request sent, awaiting response...
```

Y removemos estos archivos de prueba

```
sh-4.2$ cd ~
sh-4.2$ pwd
/home/ssm-user
sh-4.2$ rm java_jre17_exec.html js_crypto_miner.html
sh-4.2$ ls
sh-4.2$
```