

Instalar web con NFS



Arquitectura:	3
Software:	5
Configuracion EC2:	6
Proxy.....	8
Referencias:.....	11

Arquitectura:

Servidores web EC2 Linux_01 Linux_02:

- Sistema operativo: Amazon Linux
- Almacenamiento: t2.micro
- VPC: Misma VPC en toda la arquitectura
- Configuración de red y seguridad

Subredes: Linux 01: zone us-east 1a

Linux 02: zone us-east 1b

Editar reglas de entrada [Información](#)

Las reglas de entrada controlan el tráfico entrante que puede llegar a la instancia.

ID de la regla del grupo de seguridad	Tipo Información	Protocolo Información	Intervalo de puertos Información	Origen Información	Descripción: opcional Información	
sgr-023816eea915435ec	HTTP	TCP	80	Person... <input type="text" value="0.0.0.0/0"/>		<input type="button" value="Eliminar"/>
<input type="button" value="Agregar regla"/>						

Servidor Proxy ProxyNFS:

- Sistema operativo: Ubuntu
- Almacenamiento: t2.micro
- VPC: Misma VPC en toda la arquitectura
- Configuración de red y seguridad

EC2 > Grupos de seguridad > sg-08318ef06255aded8 - Proxy > Editar reglas de entrada

Editar reglas de entrada [Información](#)

Las reglas de entrada controlan el tráfico entrante que puede llegar a la instancia.

ID de la regla del grupo de seguridad	Tipo Información	Protocolo Información	Intervalo de puertos Información	Origen Información	Descripción: opcional Información	
sgr-09a36e8c7021ab003	HTTP	TCP	80	Personaliz... <input type="text" value="0.0.0.0/0"/>		<input type="button" value="Eliminar"/>
<input type="button" value="Agregar regla"/>						

EFS: La creación de la EFS es simple, trabajara en la misma VPC y usaremos la estándar para abarcar mas zonas de disponibilidad.

Crear un sistema de archivos

×

Cree un sistema de archivos de EFS con la configuración recomendada por el servicio.
[Más información](#)

Nombre - opcional
 Asigne un nombre al sistema de archivos.

El nombre puede incluir letras, números y símbolos+ - = _ : / , con un máximo de 256 caracteres.

Virtual Private Cloud (VPC)
 Elija la VPC en la que desea que las instancias EC2 se conecten a su sistema de archivos. [Más información](#)

▼

predeterminado

Clase de almacenamiento [Más información](#)

☒ **Estándar**
 Almacenar datos de forma redundante en varias zonas de disponibilidad

☐ **Única zona**
 Almacenar datos de forma redundante en una única zona de disponibilidad

Seguridad: En este caso añadimos el grupo de seguridad para permitir solo al que pertenezca el acceso por el puerto 2049 que es usado para EFS

Editar reglas de entrada

información

Las reglas de entrada controlan el tráfico entrante que puede llegar a la instancia.

Reglas de entrada

información

ID de la regla del grupo de seguridad	Tipo	Protocolo	Intervalo de puertos	Origen	Descripción: opcional
	información	información	información	información	información
sg-0e125442be3cad8bc	NFS	TCP	2049	Personaliz... <input type="text" value="sg-06dac6f6982c700d5"/>	<input type="text"/>

Agregar regla

Eliminar

Cancelar

Previsualizar los cambios

Guardar reglas

Es importante acceder al apartado red de nuestro nfs y añadir nuestro grupo de seguridad a las disponibles zonas, en principio solo necesitaríamos las zonas “a” y “b” pero se lo asignamos a todas para posibles futuras expansiones.

Destinos de montaje

Un destino de montaje proporciona un punto de enlace NFSv4 en el que puede montar un sistema de archivos de Amazon EFS. Le recomendamos que cree un destino de montaje por zona de disponibilidad. [Más](#)

Zona de disponibilidad	ID de la subred	Dirección IP	Grupos de seguridad	
us-east-1a	subnet-07a62a7401bc5001f	172.31.23.214	<div>Elegir grupos de seguridad ▼</div> <div>sg-08d32c8c44ab91d0d X SGEfs</div>	E
us-east-1b	subnet-05f4a82144043376c	172.31.40.33	<div>Elegir grupos de seguridad ▼</div> <div>sg-08d32c8c44ab91d0d X SGEfs</div>	E
us-east-1c	subnet-062e02edbc9bede8d	172.31.13.63	<div>Elegir grupos de seguridad ▼</div> <div>sg-08d32c8c44ab91d0d X SGEfs</div>	E
us-east-1d	subnet-0ce4a86ea58e13bb3	172.31.87.181	<div>Elegir grupos de seguridad ▼</div> <div>sg-08d32c8c44ab91d0d X SGEfs</div>	E
us-east-1e	subnet-07a5019cd11db5039	172.31.53.123	<div>Elegir grupos de seguridad ▼</div> <div>sg-08d32c8c44ab91d0d X SGEfs</div>	E
us-east-1f	subnet-08df3f71ef7b1ead3	172.31.74.123	<div>Elegir grupos de seguridad ▼</div> <div>sg-08d32c8c44ab91d0d X SGEfs</div>	E
<div>Agregar destino de montaje</div>				

Servidores web	172.31.18.248 172.31.36.4
Proxy	172.31.54.11

Software:

Servidores web:

- Httpd
- Nfs-utils
- Apache2

```
#!/bin/bash
yum update -y
yum install httpd -y
systemctl start httpd
systemctl enable httpd
yum -y install nfs-utils
```

Proxy:

- Apache2
- PHP

Datos de usuario asociados actualmente a esta instancia

```
#!/bin/bash
sudo apt update -y
sudo apt install apache2 -y
sudo apt install php -y
```

Configuracion EC2:

Cambiamos al directorio con `cd /var/www/html` y creamos la carpeta `efs-mount` por comodidad y introducimos el siguiente comando para montar nuestro nfs. Es importante cambiar la parte que pone `fs-ymuchos` números por la identificación de nuestro nfs `fs-00a4e3d4ee5204c0a`

```
sudo mount -t nfs -o
nfsvers=4.1,rsize=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport fs-
00a4e3d4ee5204c0a.efs.us-east-1.amazonaws.com:/ efs-mount
```

```
[root@ip-172-31-36-4 html]# sudo mount -t nfs -o nfsvers=4.1,rsize=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport fs-00a4e3d4ee5204c0a.efs.us-east-1.amazonaws.com:/ efs-mount
[root@ip-172-31-36-4 html]#
```

Comprobamos que ha sido montado con `df -h`

```
[root@ip-172-31-36-4 html]# sudo mount -t nfs -o nfsvers=4.1,rsize=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport fs-00a4e3d4ee5204c0a.efs.us-east-1.amazonaws.com:/ efs-mount
[root@ip-172-31-36-4 html]# df -h
```

Filesystem	Size	Used	Avail	Use%	Mounted on
devtmpfs	474M	0	474M	0%	/dev
tmpfs	483M	0	483M	0%	/dev/shm
tmpfs	483M	476K	482M	1%	/run
tmpfs	483M	0	483M	0%	/sys/fs/cgroup
/dev/xvda1	8.0G	1.6G	6.5G	20%	/
tmpfs	97M	0	97M	0%	/run/user/1000
tmpfs	97M	0	97M	0%	/run/user/0
fs-00a4e3d4ee5204c0a.efs.us-east-1.amazonaws.com:/	8.0E	0	8.0E	0%	/var/www/html/efs-mount

```
[root@ip-172-31-36-4 html]#
```

Dentro de la carpeta `efs-mount` usamos un `wget` y la url de la web en este caso la web que alija un clon de Netflix

```
[root@ip-172-31-36-4 efs-mount]# wget https://s3.eu-west-1.amazonaws.com/www.profesantos.cloud/Netflix.zip
--2023-01-19 13:05:23-- https://s3.eu-west-1.amazonaws.com/www.profesantos.cloud/Netflix.zip
Resolving s3.eu-west-1.amazonaws.com (s3.eu-west-1.amazonaws.com)... 52.218.112.155, 52.218.112.179, 52.92.34.144, ...
Connecting to s3.eu-west-1.amazonaws.com (s3.eu-west-1.amazonaws.com)|52.218.112.155|:443... connected.
HTTP request sent, awaiting response... 200 OK
length: 1993 (1.9K) [application/zip]
Saving to: 'Netflix.zip'

100%[=====>] 1,993 --.-K/s in 0s

2023-01-19 13:05:24 (32.7 MB/s) - 'Netflix.zip' saved [1993/1993]

[root@ip-172-31-36-4 efs-mount]#
```

Comprobamos que se ha descargado y lo descomprimos

```
[root@ip-172-31-36-4 efs-mount]# ls
Netflix.zip
[root@ip-172-31-36-4 efs-mount]# unzip Netflix.zip
Archive: Netflix.zip
  inflating: index.html
  inflating: script.js
  inflating: style.css
[root@ip-172-31-36-4 efs-mount]#
```

Estas acciones hay que realizarlas en ambos Linux_EC

The image shows two terminal windows. The left window is on a host with IP 172-31-18-248 and shows the process of mounting an Amazon EFS file system. It includes commands like `mount.nfs`, `cd /var/www/html`, and `df -h` to check disk space. The right window is on a host with IP 172-31-36-4 and shows the process of downloading a file from the EFS mount using `wget` and then unzipping it with `unzip`.

```
root@ip-172-31-18-248:/var/www/html/efs-mount# ls -l,noreport fs-00a4e3d4ee5204c0a.efs.us-east-1.amazonaws.com/: efs-mount
mount.nfs: mount point efs-mount does not exist
ec2-user@ip-172-31-18-248 efs-mount]$ pwd
/var/www/html/efs-mount
ec2-user@ip-172-31-18-248 efs-mount]$ sudo su
root@ip-172-31-18-248 efs-mount]$ sudo mount -t nfs -o nfsvers=4.1,rsize=1048576,wsize=1048576,hard,timeo=600 fs-00a4e3d4ee5204c0a.efs.us-east-1.amazonaws.com/: efs-mount
mount.nfs: mount point efs-mount does not exist
root@ip-172-31-18-248 efs-mount]$ pwd
/var/www/html/efs-mount
root@ip-172-31-18-248 efs-mount]$ cd ..
root@ip-172-31-18-248 html]$ nfsvers=4.1,rsize=1048576,wsize=1048576,hard,timeo=600 fs-00a4e3d4ee5204c0a.efs.us-east-1.amazonaws.com/: No such file or directory
root@ip-172-31-18-248 html]$ sudo mount -t nfs -o nfsvers=4.1,rsize=1048576,wsize=1048576,hard,timeo=600 fs-00a4e3d4ee5204c0a.efs.us-east-1.amazonaws.com/: efs-mount
root@ip-172-31-18-248 html]$ df -h
Filesystem              Size  Used Avail Use% Mounted on
devtmpfs                474M   0  474M   0% /dev
tmpfs                   483M   0  483M   0% /dev/shm
tmpfs                   483M  476K  482M   1% /run
tmpfs                   483M   0  483M   0% /sys/fs/cgroup
/dev/xvda1              8.0G  1.6G  6.5G  20% /
tmpfs                   97M   0   97M   0% /run/user/1000
tmpfs                   97M   0   97M   0% /run/user/1000
fs-00a4e3d4ee5204c0a.efs.us-east-1.amazonaws.com/ 8.0E   0  8.0E   0% /var/www/html/efs-mount
root@ip-172-31-18-248 efs-mount]$ ls
index.html  Netflix.zip  script.js   style.css
root@ip-172-31-18-248 efs-mount]$

root@ip-172-31-36-4:/var/www/html/efs-mount# pwd
/var/www/html
[root@ip-172-31-36-4 html]$ cd efs-mount/
[root@ip-172-31-36-4 efs-mount]$ wget https://s3.eu-west-1.amazonaws.com/www.profesantostcloud/Netflix.zip
--2023-01-19 13:05:23-- https://s3.eu-west-1.amazonaws.com/www.profesantostcloud/Netflix.zip
Resolving s3.eu-west-1.amazonaws.com (s3.eu-west-1.amazonaws.com)... 52.218.112.155, 52.218.112.179, 52.92.34.144, ...
Connecting to s3.eu-west-1.amazonaws.com (s3.eu-west-1.amazonaws.com)|52.218.112.155|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1993 (1.9K) [application/zip]
Saving to: 'Netflix.zip'

100%[=====] 1,993
2023-01-19 13:05:24 (32.7 MB/s) - 'Netflix.zip' saved [1993/1993]

[root@ip-172-31-36-4 efs-mount]$ ls
Netflix.zip
[root@ip-172-31-36-4 efs-mount]$ unzip Netflix.zip
Archive: Netflix.zip
  inflating: index.html
  inflating: script.js
  inflating: style.css
root@ip-172-31-36-4 efs-mount]$ ls
index.html  Netflix.zip  script.js   style.css
root@ip-172-31-36-4 efs-mount]$
```

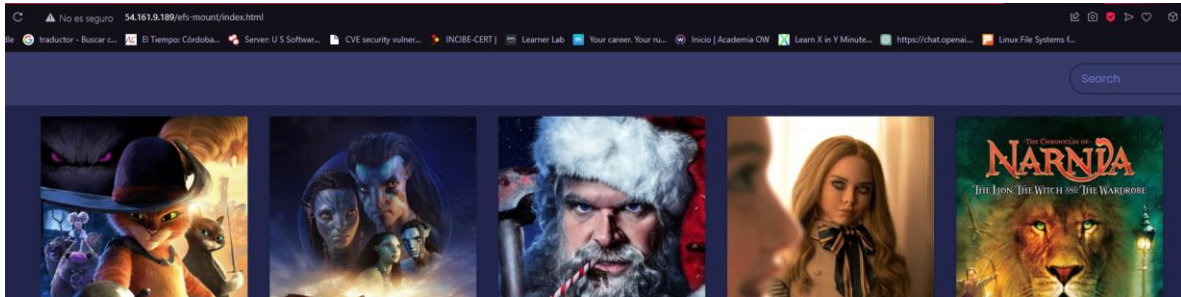
Comprobamos el contenido de index.html

```
root@ip-172-31-18-248:/var/www/html#
[root@ip-172-31-18-248 efs-mount]# [root@ip-172-31-18-248 efs-mount]# cd ..
[root@ip-172-31-18-248 html]# curl localhost/efs-mount/index.html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <link rel="stylesheet" href="style.css" />
    <title>Movie App</title>
  </head>
  <body>
    <header>
      <form id="form">
        <input type="text" id="search" class="search" placeholder="Search">
      </form>
    </header>

    <main id="main"></main>

    <script src="script.js"></script>
  </body>
</html>
[root@ip-172-31-18-248 html]#
```

Si accedemos desde la ip publica/efs-mount/index.html veremos que se visualiza nuestra web



Accedemos a /etc/httpd/cnf/httpd.conf y lo editamos con nano hasta la siguiente línea y añadimos /efs-mount en ambas dos maquinas

```
GNU nano 2.9.8 /etc/httpd/cnf/httpd.conf

</Directory>

#
# Note that from this point forward you must specifically allow
# particular features to be enabled - so if something's not working as
# you might expect, make sure that you have specifically enabled it
# below.
#
#
# DocumentRoot: The directory out of which you will serve your
# documents. By default, all requests are taken from this directory, but
# symbolic links and aliases may be used to point to other locations.
#
DocumentRoot "/var/www/html/efs-mount"

#
# Relax access to content within /var/www.
#
<Directory "/var/www">
    AllowOverride None
    # Allow open access:
    Require all granted
</Directory>
```

Importante no olvidar resetear el servicio

```
# systemctl restart httpd
#
```

Proxy

Para mejorar esta infraestructura añadimos un servidor proxy que balanceara entre nuestras maquinas.

Lo primero que debemos hacer es habilitar los modulos del proxy con los siguientes comandos:


```

ubuntu@ip-172-31-54-11:~$ sudo a2enmod proxy
Enabling module proxy.
To activate the new configuration, you need to run:
  systemctl restart apache2
ubuntu@ip-172-31-54-11:~$ sudo a2enmod proxy_http
Considering dependency proxy for proxy_http:
Module proxy already enabled
Enabling module proxy_http.
To activate the new configuration, you need to run:
  systemctl restart apache2
ubuntu@ip-172-31-54-11:~$ sudo a2enmod proxy_ajp
Considering dependency proxy for proxy_ajp:
Module proxy already enabled
Enabling module proxy_ajp.
To activate the new configuration, you need to run:
  systemctl restart apache2
ubuntu@ip-172-31-54-11:~$ sudo a2enmod rewrite
Enabling module rewrite.
To activate the new configuration, you need to run:
  systemctl restart apache2
ubuntu@ip-172-31-54-11:~$ sudo a2enmod deflate
Considering dependency filter for deflate:
Module filter already enabled
Module deflate already enabled
ubuntu@ip-172-31-54-11:~$ sudo a2enmod headers
Enabling module headers.
To activate the new configuration, you need to run:
  systemctl restart apache2
ubuntu@ip-172-31-54-11:~$ sudo a2enmod proxy_balancer
Considering dependency proxy for proxy_balancer:
Module proxy already enabled
Considering dependency alias for proxy_balancer:
Module alias already enabled
Considering dependency slotmem_shm for proxy_balancer:
Enabling module slotmem_shm.
Enabling module proxy_balancer.
To activate the new configuration, you need to run:
  systemctl restart apache2
ubuntu@ip-172-31-54-11:~$ sudo a2enmod proxy_connect
Considering dependency proxy for proxy_connect:
Module proxy already enabled
Enabling module proxy_connect.
To activate the new configuration, you need to run:
  systemctl restart apache2
ubuntu@ip-172-31-54-11:~$ sudo a2enmod proxy_html
Considering dependency proxy for proxy_html:
Module proxy already enabled
Considering dependency xml2enc for proxy_html:
Enabling module xml2enc.

```

Por ultimo accedemos al archivo de configuración en /etc/apache2/sites-enabled/000-default.conf y añadimos las siguientes líneas para el balanceador de carga con las ip de las ec2 en los balancer-member. No olvidar cambiar el documentroot para una url mas directa.

```

GNU nano 6.2 /etc/apache2/sites-enabled/000-default.conf
VirtualHost *:80
    # The ServerName directive sets the request scheme, hostname and port that
    # the server uses to identify itself. This is used when creating
    # redirection URLs. In the context of virtual hosts, the ServerName
    # specifies what hostname must appear in the request's Host: header to
    # match this virtual host. For the default virtual host (this file) this
    # value is not decisive as it is used as a last resort host regardless.
    # However, you must set it for any further virtual host explicitly.
    #ServerName www.example.com

    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html/efs-mount_

    ProxyPass /balancer-manager !
    <Proxy balancer://mycluster>
        #server1
        BalancerMember http://172.31.10.248/ loadfactor=3 timeout=1
        #server2
        BalancerMember http://172.31.36.4/ loadfactor=2
    </Proxy>

    ProxyPass "/" "balancer://mycluster/"
    ProxyPassReverse "/" "balancer://mycluster/"

    <Location /balancer-manager>
        SetHandler balancer-manager
        Order Deny,Allow
        Allow from all
    </Location>

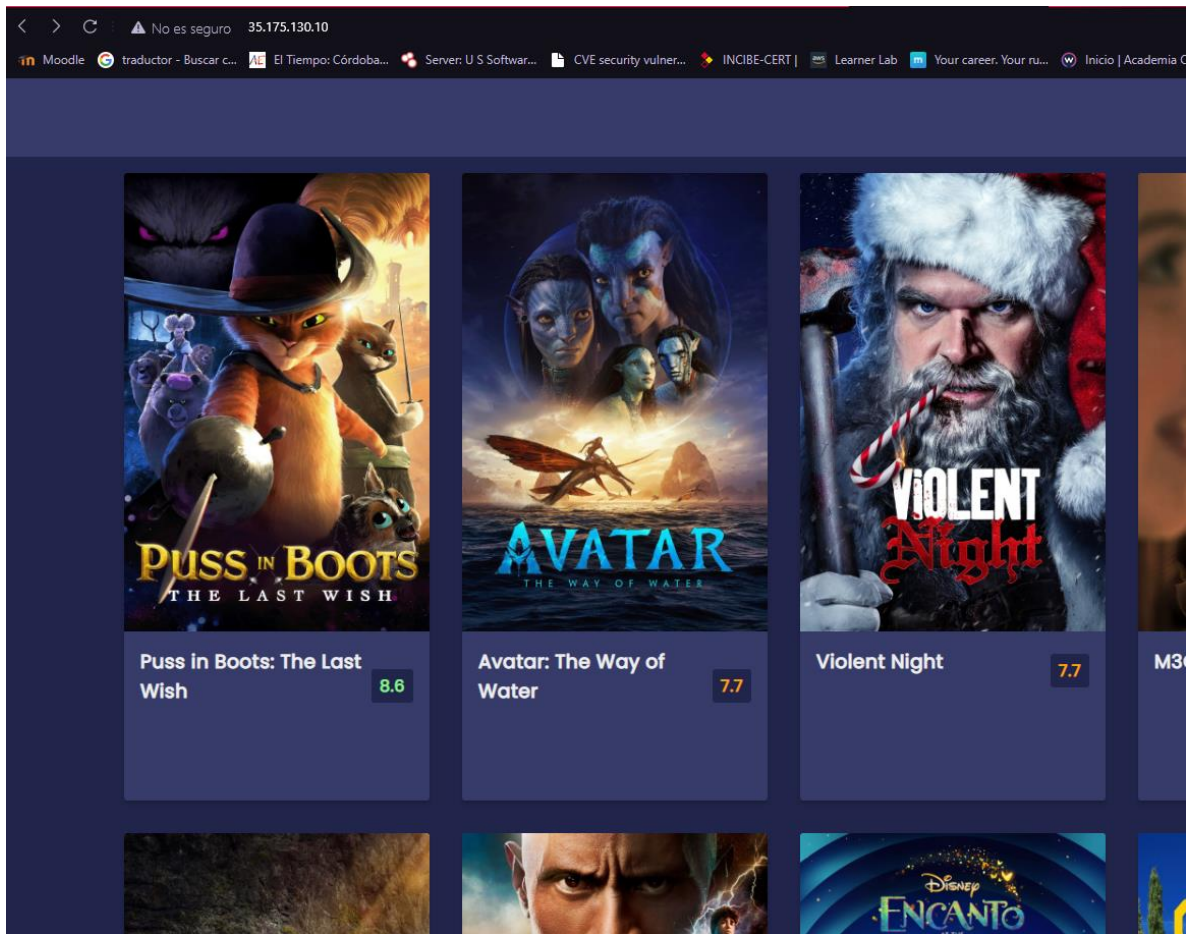
    # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
    # error, crit, alert, emerg.
    # It is also possible to configure the loglevel for particular
    # modules, e.g.
    #LogLevel info ssl:warn

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

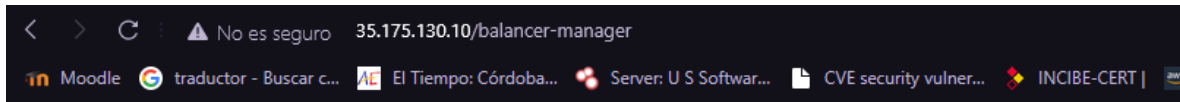
    # For most configuration files from conf-available/, which are
    # enabled or disabled at a global level, it is possible to
    # include a line for only one particular virtual host. For example the
    # following line enables the CGI configuration for this host only
    # after it has been globally disabled with "a2disconf".
    #Include conf-available/serve-cgi-bin.conf
VirtualHost

```

Comprobamos el resultado:



Y entramos en el balanceador para comprobar si esta funcionando:



Load Balancer Manager for 35.175.130.10

Server Version: Apache/2.4.52 (Ubuntu)
 Server Built: 2022-09-30T04:09:50
 Balancer changes will NOT be persisted on restart.
 Balancers are inherited from main server.
 ProxyPass settings are inherited from main server.

LoadBalancer Status for [balancer://mycluster](#) [pf0cf7f1c_mycluster]

MaxMembers	StickySession	DisableFailover	Timeout	FailoverAttempts	Method	Path	Active
2 [2 Used]	(None)	Off	0	1	byrequests	/	Yes

Worker URL	Route	RouteRedir	Factor	Set	Status	Elected	Busy	Load	To	From
http://172.31.18.248/			3.00	0	Init Ok	2	0	0	1.1K	2.2K
http://172.31.36.4/			2.00	0	Init Ok	1	0	0	435	1.7K

Apache/2.4.52 (Ubuntu) Server at 35.175.130.10 Port 80

Referencias:

<https://github.com/EduAlbanoCarmona/IAW-Arquitectura-3-niveles>