

Actividades

Instancias que se utilizaron durante el trabajo:

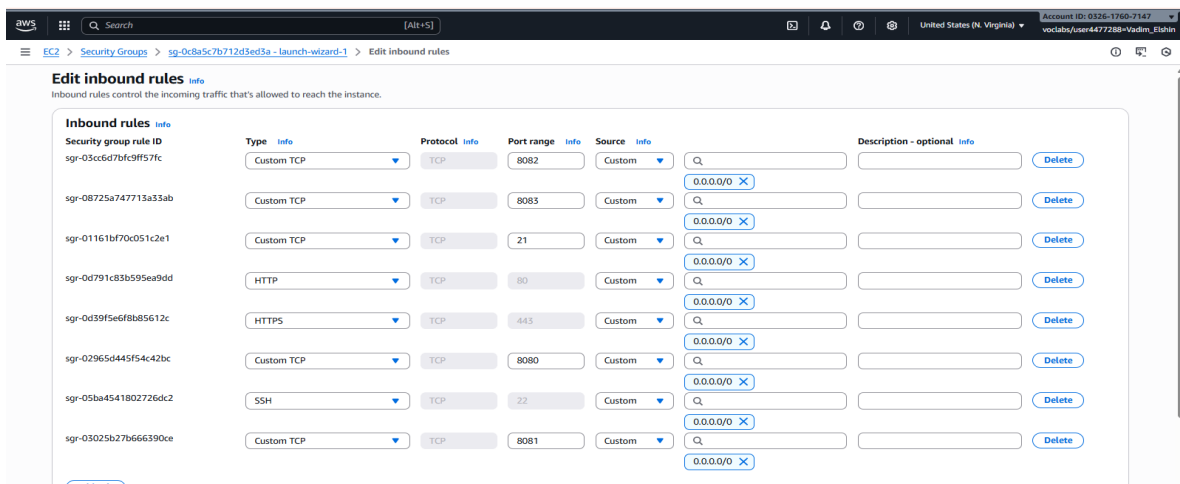


The screenshot shows the AWS Management Console 'Instances' page. It lists four EC2 instances, all in a 'Running' state. The instances are named 'WebServer #1', 'WebServer #2', 'Reverse-Proxy', and 'DEAW_Vadim'. They are all using the 't3.micro' instance type and are located in the 'us-east-1b' availability zone. The 'WebServer #1' and 'WebServer #2' instances have public IP addresses of 3.84.97.40 and 13.220.94.197 respectively. The 'Reverse-Proxy' instance has a public IP address of 3.80.44.162. The 'DEAW_Vadim' instance has a public IP address of 98.89.41.237.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4
WebServer #1	i-09d91df10f2cdd2c	Running	t3.micro	Initializing	View alarms +	us-east-1b	ec2-3-84-97-40.comput...	3.84.97.40
WebServer #2	i-0d3121ef395cae989	Running	t3.micro	Initializing	View alarms +	us-east-1b	ec2-13-220-94-197.co...	13.220.94.197
Reverse-Proxy	i-0ad9af53a99b581e8	Running	t3.micro	Initializing	View alarms +	us-east-1b	ec2-3-80-44-162.comp...	3.80.44.162
DEAW_Vadim	i-0120edbc3f0178952	Running	t3.micro	Initializing	View alarms +	us-east-1b	ec2-98-89-41-237.com...	98.89.41.237

Nota importante: Inicialmente, una instancia solo tiene permitido usar el puerto 22 (SSH). Para abrir otros puertos (ej. 80 - HTTP) es necesario configurar su grupo de seguridad.

Grupo de seguridad de todas las instancias:

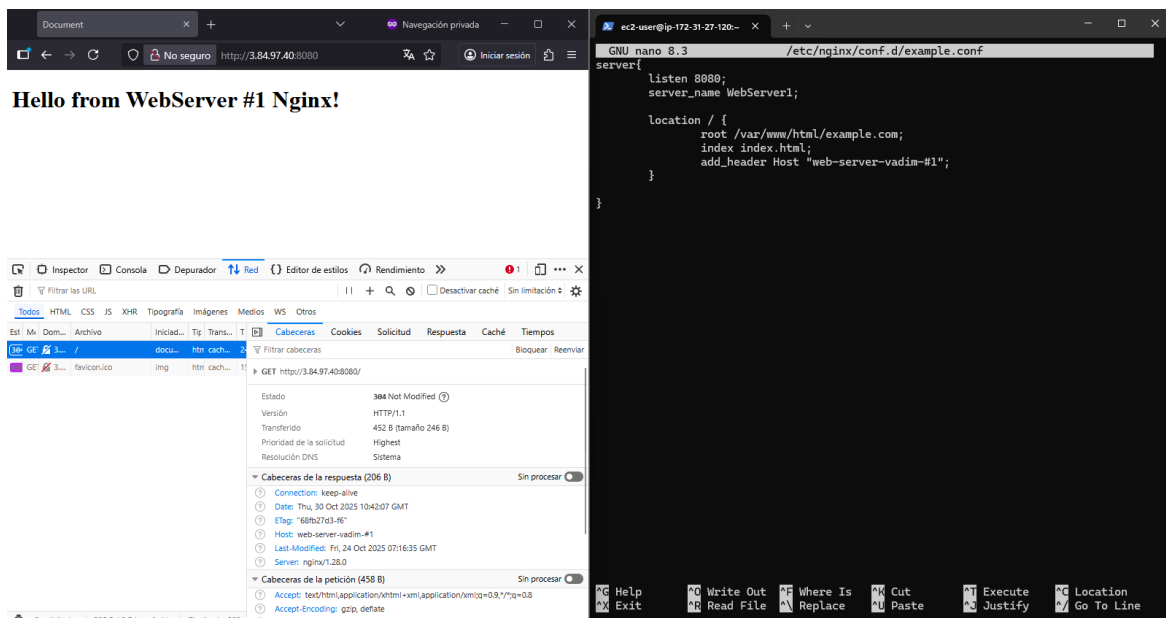


The screenshot shows the AWS Management Console 'Edit inbound rules' page for a security group. It lists several inbound rules for the security group 'sg-0c8a5c7b712d3ed3a'. The rules are configured for various ports and protocols, including TCP, HTTP, HTTPS, and SSH. The rules are as follows:

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sg-0c8a5c7b712d3ed3a	Custom TCP	TCP	8082	Custom	
sg-08725a747713a33ab	Custom TCP	TCP	8083	Custom	
sg-01161bf70d051c2e1	Custom TCP	TCP	21	Custom	
sg-0d791c83b595ea9dd	HTTP	TCP	80	Custom	
sg-0d39f5e6f8b5612c	HTTPS	TCP	443	Custom	
sg-02965d445f54c42bc	Custom TCP	TCP	8080	Custom	
sg-05ba4541802726dc2	SSH	TCP	22	Custom	
sg-03025b27b666390ce	Custom TCP	TCP	8081	Custom	

P2.1 Instalación y configuración de Nginx

A diferencia de Linux Debian, en Amazon Linux los hosts virtuales de nginx se pueden configurar dentro de una única carpeta "/etc/nginx/conf.d/".



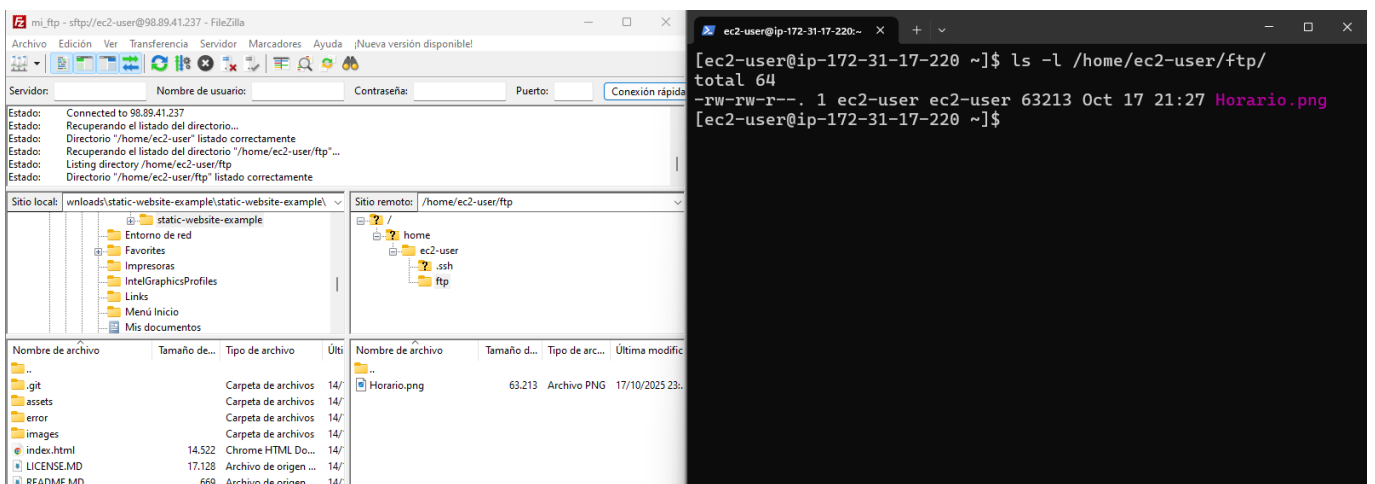
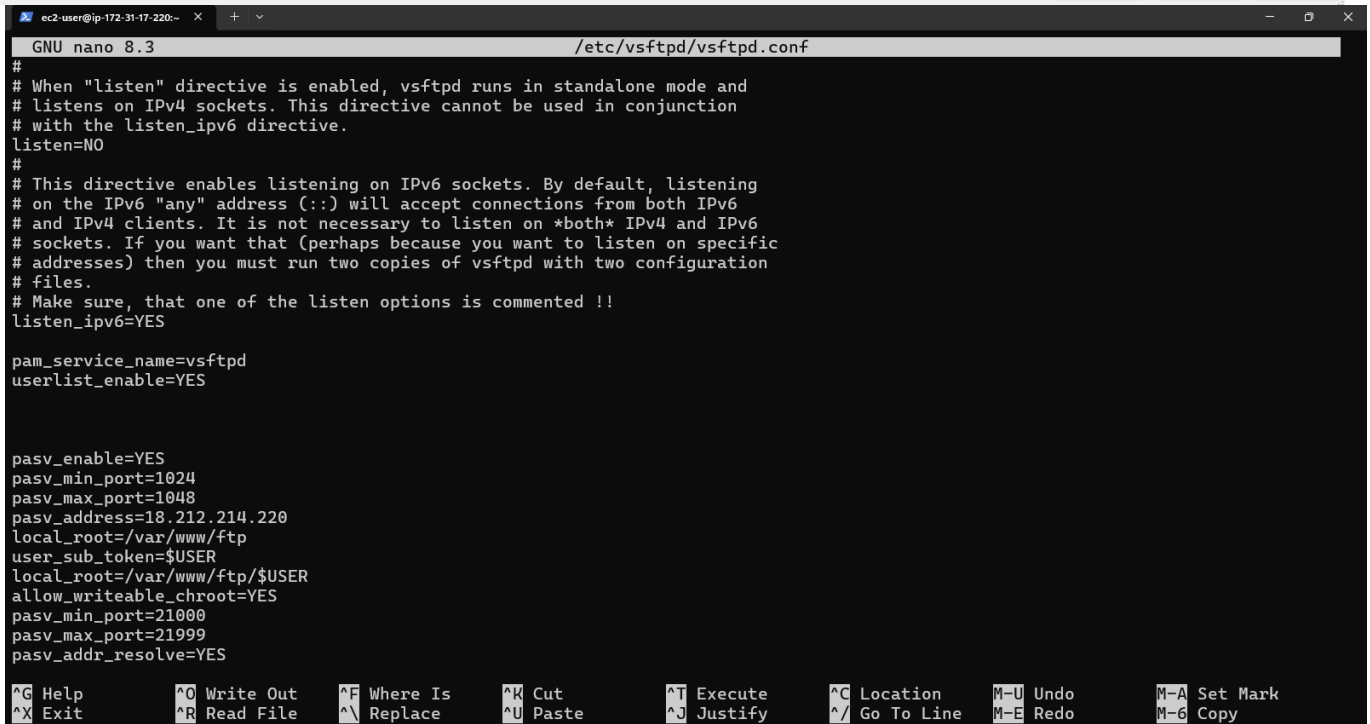
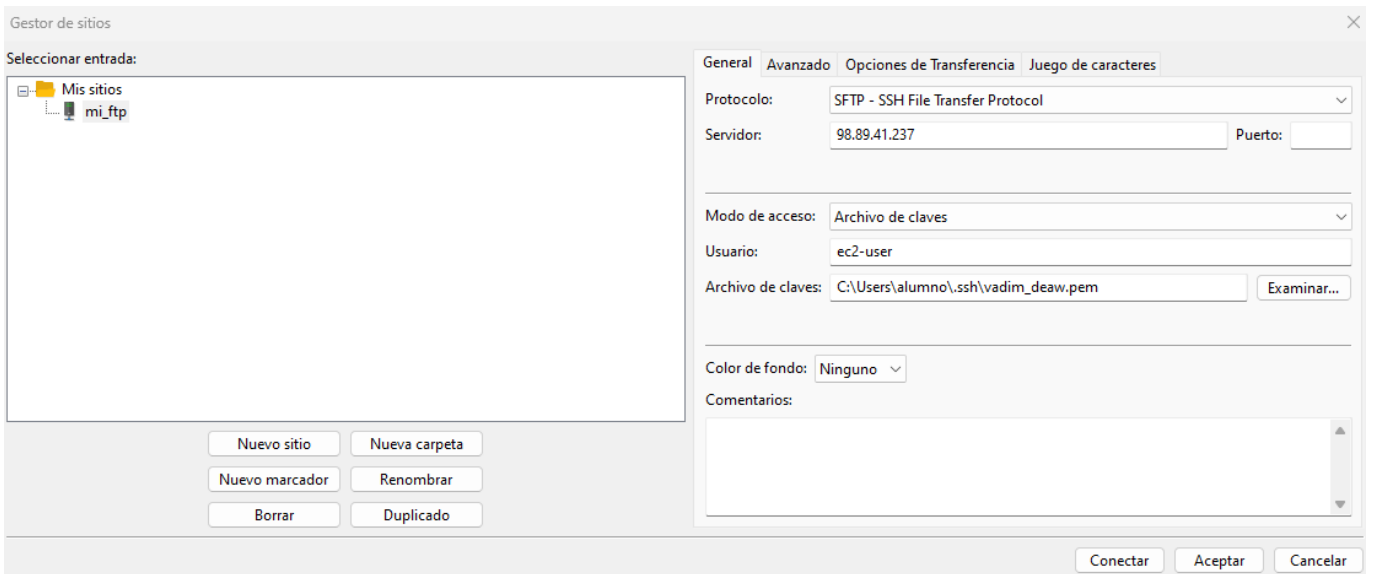
The screenshot shows a web browser window displaying 'Hello from WebServer #1 Nginx!' and a terminal window showing the Nginx configuration file. The browser window shows the URL 'http://3.84.97.40:8080' and the response 'Hello from WebServer #1 Nginx!'. The terminal window shows the Nginx configuration file content:

```
GNU nano 8.3 /etc/nginx/conf.d/example.conf
server{
    listen 8080;
    server_name WebServer1;

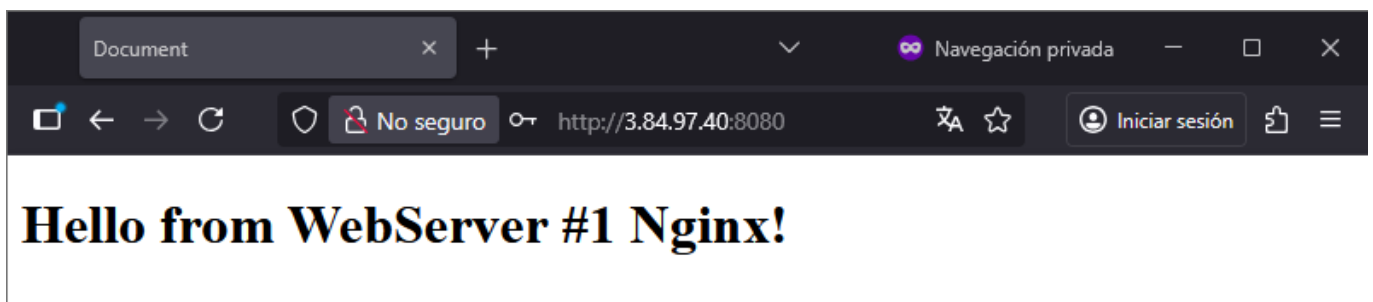
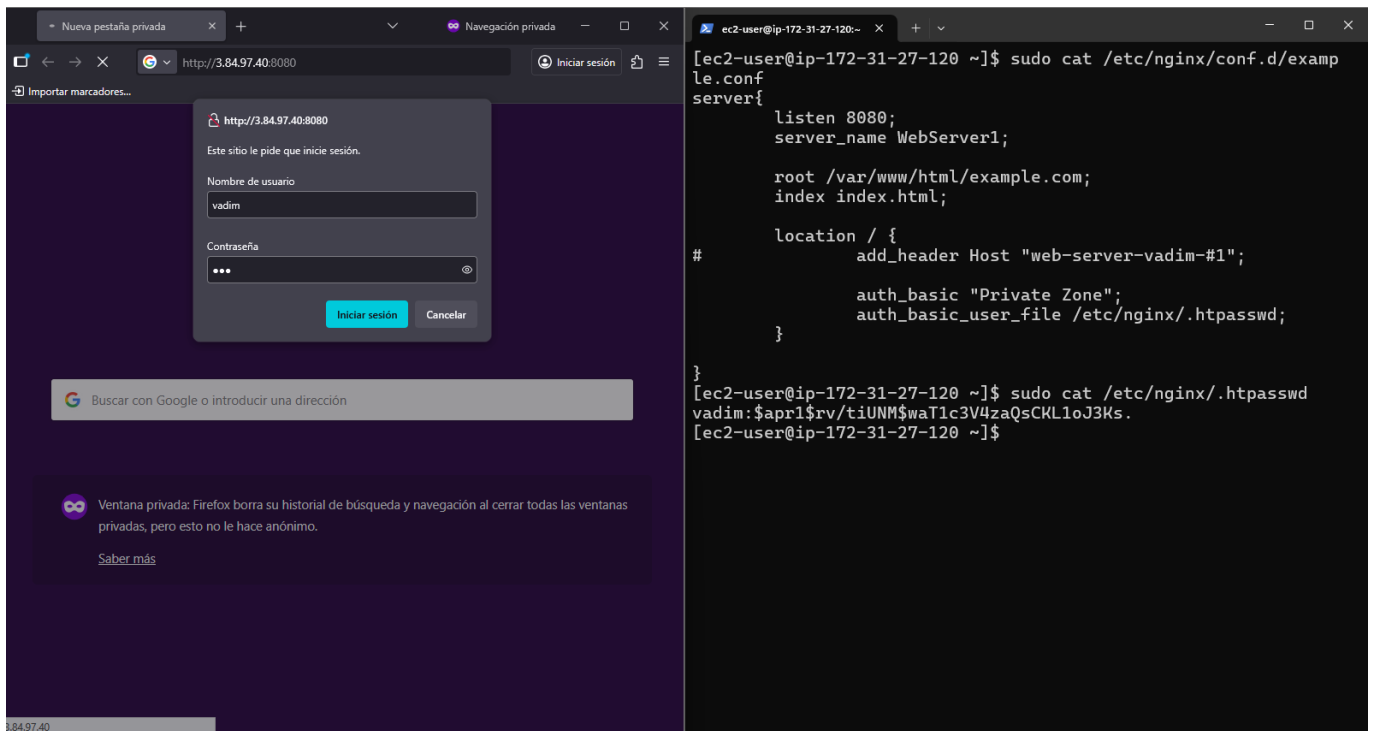
    location / {
        root /var/www/html/example.com;
        index index.html;
        add_header Host "web-server-vadim-#1";
    }
}
```

P2.1 Instalación y configuración de FTP

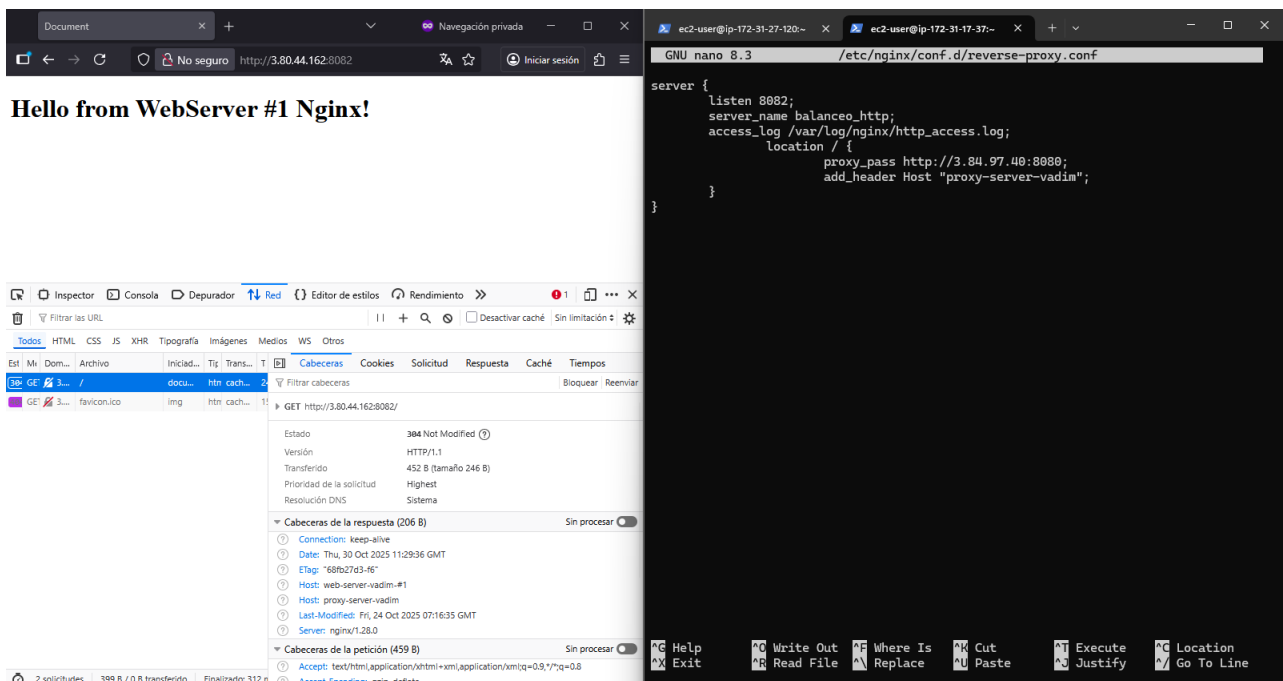
Durante la creación de la instancia EC2, ya había generado mis claves. Posteriormente, las utilicé para conectarme a mi instancia mediante FileZilla.



P2.3 Autenticación en Nginx



P2.4 Proxy inverso con Nginx



P2.5 Balanceo de carga con Nginx

The screenshot shows a web browser window on the left and a terminal window on the right. The browser window displays the text "Hello from WebServer #2 Nginx!". The terminal window shows the Nginx configuration file `/etc/nginx/conf.d/reverse-proxy.conf` in nano 8.3. The configuration includes an upstream block for `backend_hosts` with two servers: `3.84.97.40:8080` and `13.220.94.197:8081`. The server block listens on `8082` and proxies requests to `http://backend_hosts`, adding a `Host` header.

Browser Window:

- Address bar: `http://3.80.44.162:8082`
- Page content: **Hello from WebServer #2 Nginx!**
- Inspector: Network tab selected. Request: `GET http://3.80.44.162:8082/`. Status: **200 OK**. Headers: `Accept-Ranges: bytes`, `Connection: keep-alive`, `Content-Length: 241`, `Content-Type: text/html`, `Date: Thu, 30 Oct 2025 11:37:40 GMT`, `Etag: "69034db7-11"`, `Host: web-server-vadim-#2`, `Host: proxy-server-balanceo-vadim`, `Last-Modified: Thu, 30 Oct 2025 11:36:23 GMT`, `Server: nginx/1.28.0`.

Terminal Window:

```
GNU nano 8.3 /etc/nginx/conf.d/reverse-proxy.conf
upstream backend_hosts{
    random;
    server 3.84.97.40:8080;
    server 13.220.94.197:8081;
}

server {
    listen 8082;
    server_name balanceo_http;
    access_log /var/log/nginx/http_access.log;
    location / {
        proxy_pass http://backend_hosts;
        add_header Host "proxy-server-balanceo-vadim";
    }
}
```

The screenshot shows a web browser window on the left and a terminal window on the right. The browser window displays the text "Hello from WebServer #1 Nginx!". The terminal window shows the same Nginx configuration file as above. The browser's network inspector shows a different request and response.

Browser Window:

- Address bar: `http://3.80.44.162:8082`
- Page content: **Hello from WebServer #1 Nginx!**
- Inspector: Network tab selected. Request: `GET http://3.80.44.162:8082/`. Status: **304 Not Modified**. Headers: `Connection: keep-alive`, `Date: Thu, 30 Oct 2025 11:37:16 GMT`, `Etag: "68fb27d3-46"`, `Host: web-server-vadim-#1`, `Host: proxy-server-balanceo-vadim`, `Last-Modified: Fri, 24 Oct 2025 07:16:35 GMT`, `Server: nginx/1.28.0`. Request headers: `Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8`, `Accept-Encoding: gzip, deflate`.

Terminal Window:

```
GNU nano 8.3 /etc/nginx/conf.d/reverse-proxy.conf
upstream backend_hosts{
    random;
    server 3.84.97.40:8080;
    server 13.220.94.197:8081;
}

server {
    listen 8082;
    server_name balanceo_http;
    access_log /var/log/nginx/http_access.log;
    location / {
        proxy_pass http://backend_hosts;
        add_header Host "proxy-server-balanceo-vadim";
    }
}
```

P2.6 Proxy inverso y balanceo de carga con SSL en NGINX

```
GNU nano 8.3 /etc/nginx/conf.d/reverse-proxy.conf
upstream backend_hosts{
    random;
    server 3.84.97.40:8080;
    server 13.220.94.197:8081;
}

server{
    listen 443 ssl;
    ssl_certificate /etc/nginx/ssl/server.crt;
    ssl_certificate_key /etc/nginx/ssl/server.key;
    #
    # ssl_protocols TLSv1.3;
    # ssl_ciphers ECDH+AESGCM:DH+AESGCM:ECDH+AES256:DH+AES256:ECDH+AES128:DH+AES:
    # ECDH+3DES:DH+3DES:RSA+AESGCM:RSA+AES:RSA+3DES:
    server_name balanceo_https;

    access_log /var/log/nginx/https_access.log;

    location / {
        proxy_pass http://backend_hosts;
    }
}

server {
    listen 8080;
    server_name balanceo_http;
    access_log /var/log/nginx/http_access.log;
    return 301 https://3.80.44.162:443$request_uri;
}
```



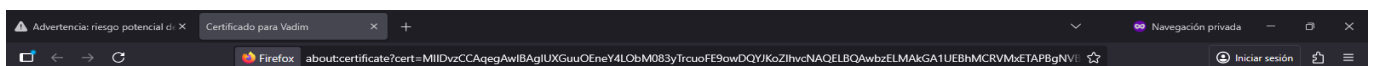
Advertencia: riesgo potencial de seguridad a continuación

Firefox ha detectado una posible amenaza de seguridad y no ha cargado **3.80.44.162**. Si visita este sitio, los atacantes podrían intentar robar información como sus contraseñas, correos electrónicos o detalles de su tarjeta de crédito.

[Más información...](#)

[Retroceder \(recomendado\)](#)

[Avanzado...](#)



Certificado

Vadlim	
Nombre del asunto	
Pais	ES
Estado/Provincia	Alicante
Localidad	Alicante
Organización	IES Doctor Balmis
Unidad organizativa	2 DAW
Nombre común	Vadlim
Nombre del emisor	
Pais	ES
Estado/Provincia	Alicante
Localidad	Alicante
Organización	IES Doctor Balmis
Unidad organizativa	2 DAW
Nombre común	Vadlim