

# Bastionado de redes y sistemas

Tarea 6: Configuración de dispositivos  
y sistemas informáticos

## Índice

Estado del sistema previo a fail2ban.....	2
Instalación de fail2ban.....	5
Configuración de fail2ban.....	7
Ataque y reacción de fail2ban.....	10
Comprobación de direcciones IP baneadas.....	12
Desbaneo de direcciones IP.....	13
Envío de email.....	15
Protección de otros servicios con fail2ban.....	21
Bibliografía.....	27

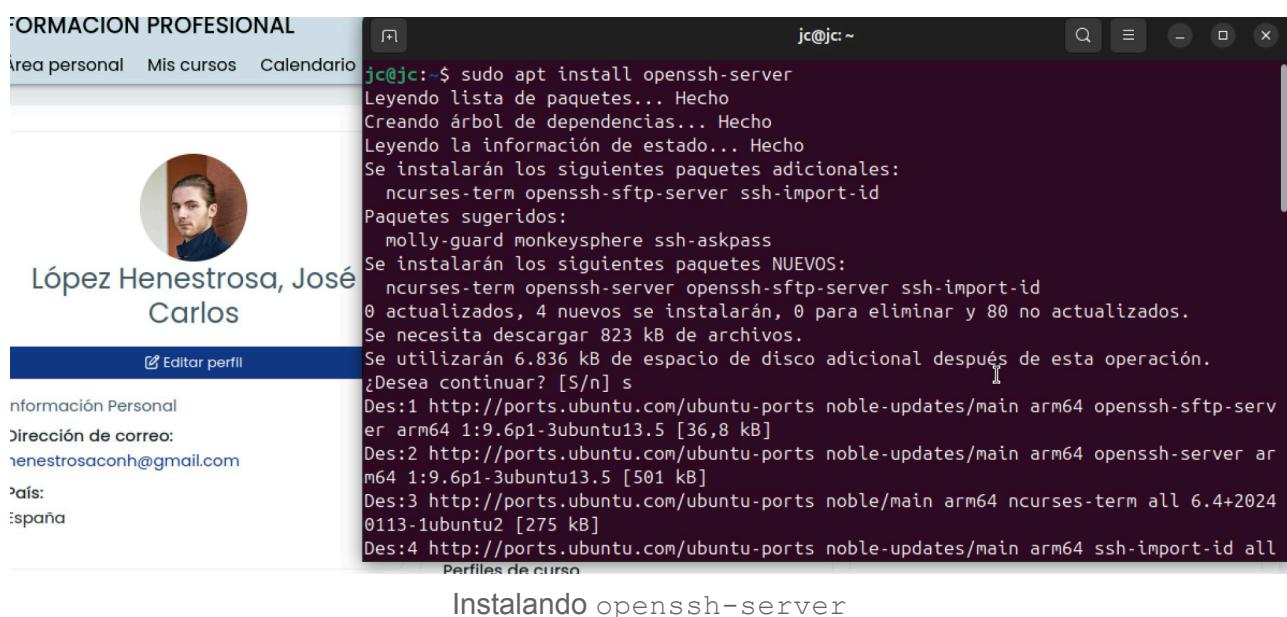
# Estado del sistema previo a fail2ban

Para entender la mejora que supone fail2ban en el sistema, simularemos un pequeño ataque por fuerza bruta a nuestro sistema. Para ello, debemos instalar un servidor SSH, por lo que actualizamos los paquetes con este comando:

```
sudo apt update && sudo apt upgrade
```

E instalamos el servidor SSH con este otro:

```
sudo apt install openssh-server
```



Instalando openssh-server

Para hallar la IP del servidor SSH en la red doméstica, ejecutamos:

```
hostname -I
```



Dirección IP de la máquina virtual con el servidor SSH instalado

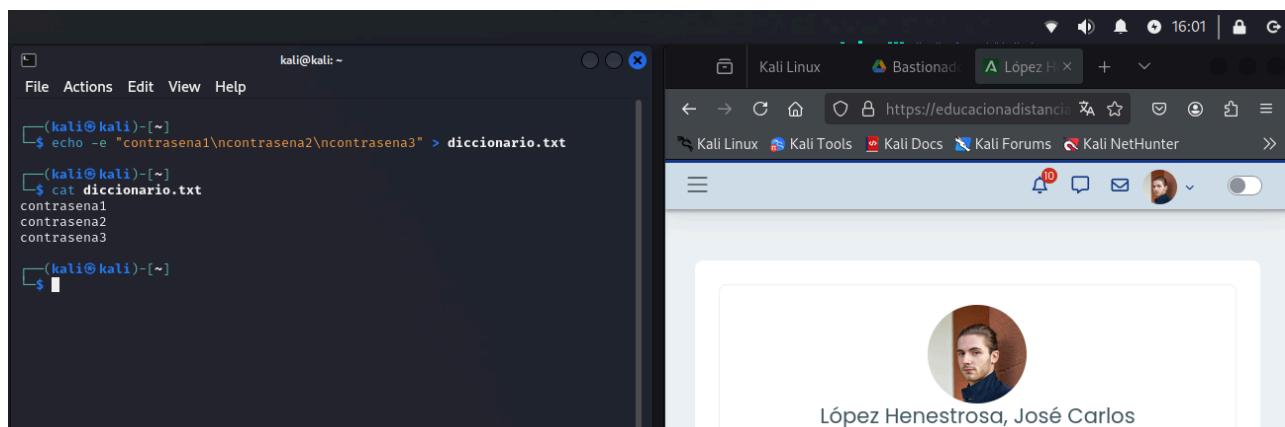
Ahora que tenemos el servidor SSH instalado y la IP de la máquina hallada, intentamos realizar el ataque desde una máquina externa conectada a la misma red que el servidor SSH. Para ello, ejecutaremos sistemática este comando:

```
ssh usuario@192.168.18.68
```

El usuario no es importante, ni la contraseña (que obviamente el atacante no conoce), ya que el objetivo del ataque es el servicio SSH en nuestra IP. Podemos usar la configuración menos restrictiva de SSH.

Para ejecutar el ataque, utilizaremos la herramienta `hydra` en Kali Linux, la cual viene instalada por defecto en dicha distribución. Vamos a usar un diccionario bastante simple para ejecutar el ataque, ya que sólo queremos comprobar cómo reacciona el servidor SSH ante los intentos de acceso. Para crearlo, ejecutaremos el siguiente comando:

```
echo -e "contraseña1\ncontraseña2\ncontraseña3" > diccionario.txt
```



Creando diccionario.txt

Cabe destacar que Kali también incluye algunos diccionarios por defecto, los cuales están almacenados en el directorio `/usr/share/wordlists/`. Entre ellos, incluye uno de los diccionarios de contraseñas más famoso, el cual es `rockyou.txt`.

Tras crear el diccionario, ejecutamos este comando para lanzar el ataque:

```
sudo hydra -l usuario -P diccionario.txt ssh://192.168.18.68
```

```

kali@kali:~$ sudo hydra -l usuario -P dictionario.txt ssh://192.168.18.68
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in
military or secret service organizations, or for illegal purposes (this is n
on-binding, these ** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-02-08 16:
07:05
[WARNING] Many SSH configurations limit the number of parallel tasks, it is r
ecommended to reduce the tasks: use -t 4
[DATA] max 3 tasks per 1 server, overall 3 tasks, 3 login tries (l:1/p:3), ~1
try per task
[DATA] attacking ssh://192.168.18.68:22/
1 of 1 target completed, 0 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-02-08 16:
07:09

```

Ejecución del ataque con `hydra` desde Kali Linux al servidor SSH de la máquina virtual

Paralelamente a estos ataques, monitorizamos en nuestro servidor cómo se manifiestan dichos intentos de log en el fichero que ya conocemos, `/var/log/auth.log`, con el siguiente comando:

```
tail -f /var/log/auth.log
```

```

jc@jc:~$ tail -f /var/log/auth.log
2025-02-08T16:07:05.746958+00:00 jc sshd[6217]: pam_unix(sshd:auth): check pass; user unknown
2025-02-08T16:07:05.747079+00:00 jc sshd[6217]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.18.55
2025-02-08T16:07:05.747185+00:00 jc sshd[6214]: pam_unix(sshd:auth): check pass; user unknown
2025-02-08T16:07:05.747703+00:00 jc sshd[6214]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.18.55
2025-02-08T16:07:07.547909+00:00 jc sshd[6215]: Failed password for invalid user usuario from 192.168.18.55 port 35654 ssh2
2025-02-08T16:07:07.556591+00:00 jc sshd[6217]: Failed password for invalid user usuario from 192.168.18.55 port 35656 ssh2
2025-02-08T16:07:07.556939+00:00 jc sshd[6214]: Failed password for invalid user usuario from 192.168.18.55 port 35652 ssh2
2025-02-08T16:07:09.371043+00:00 jc sshd[6217]: Connection closed by invalid user usuario 192.168.18.55 port 35656 [preauth]
2025-02-08T16:07:09.372748+00:00 jc sshd[6214]: Connection closed by invalid user usuario 192.168.18.55 port 35652 [preauth]
2025-02-08T16:07:09.374933+00:00 jc sshd[6215]: Connection closed by invalid user usuario 192.168.18.55 port 35654 [preauth]

```

Contenido del archivo `/var/log/auth.log` que refleja los intentos de login externos por SSH

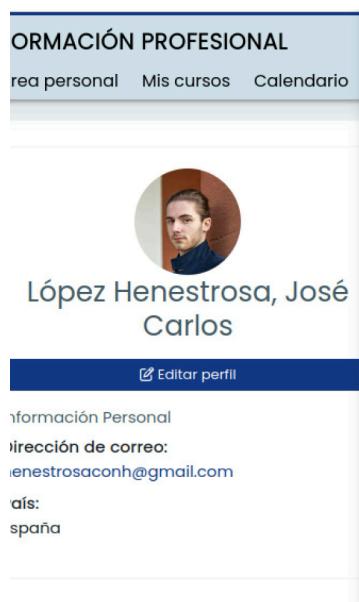
# Instalación de fail2ban

Procedemos a instalar `fail2ban` desde los repositorios debidamente actualizados de nuestro servidor. En caso de no haberlos actualizado en el apartado anterior, ejecutamos:

```
sudo apt update && sudo apt upgrade
```

Tras ello, procedemos a instalar `fail2ban`:

```
sudo apt install fail2ban
```



```
jc@jc:~$ sudo apt install fail2ban
[sudo] password for jc:
Sorry, try again.
[sudo] password for jc:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  python3-pyasyncore python3-pyinotify whois
Paquetes sugeridos:
  mailx monit sqlite3 python-pyinotify-doc
Se instalarán los siguientes paquetes NUEVOS:
  fail2ban python3-pyasyncore python3-pyinotify whois
0 actualizados, 4 nuevos se instalarán, 0 para eliminar y 80 no actualizados.
Se necesita descargar 494 kB de archivos.
Se utilizarán 2.654 kB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
Des:1 http://ports.ubuntu.com/ubuntu-ports noble/main arm64 python3-pyasyncore all 1.0.2-2 [10,1 kB]
Des:2 http://ports.ubuntu.com/ubuntu-ports noble-updates/universe arm64 fail2ban all 1.0.2-3ubuntu0.1 [409 kB]
Des:3 http://ports.ubuntu.com/ubuntu-ports noble/main arm64 python3-pyinotify all 0.9.6-2ubuntu1 [25,0 kB]
Des:4 http://ports.ubuntu.com/ubuntu-ports noble/main arm64 whois arm64 5.5.22 [50,3 kB]
```

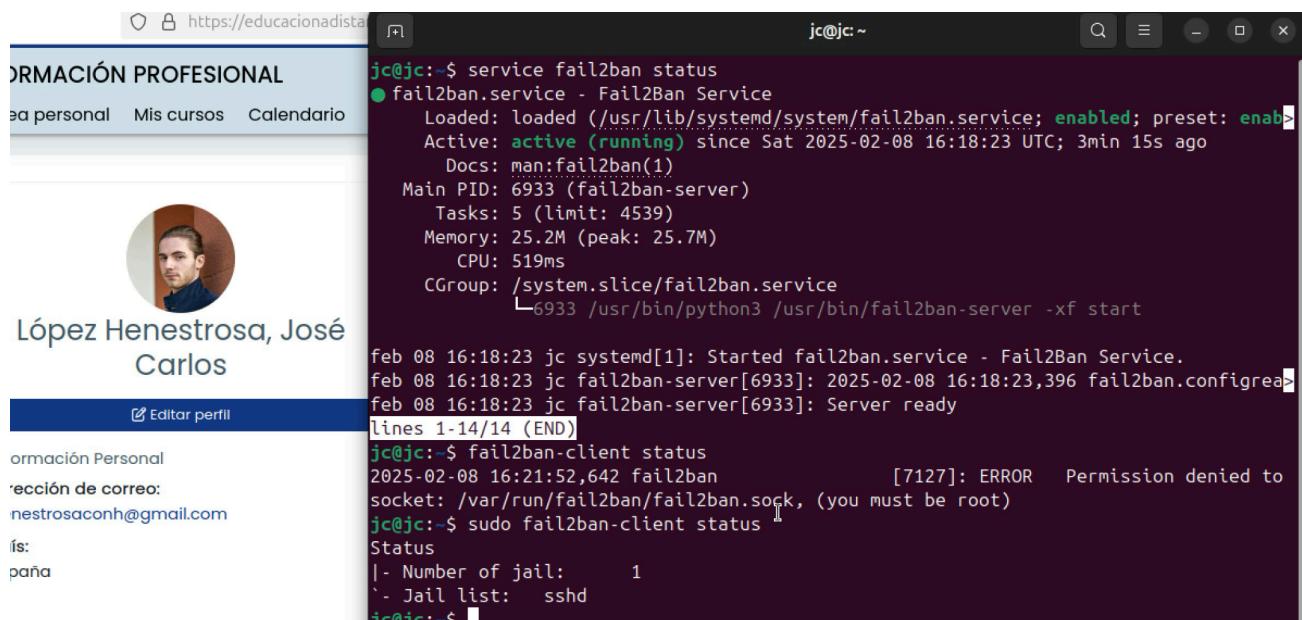
Instalando `fail2ban`

Ahora comprobamos que el servicio está instalado y funciona. Lo podemos comprobar tanto con

```
service fail2ban status
```

como con

```
fail2ban-client status
```



The screenshot shows a dual-pane interface. On the left, a web browser displays a user profile for 'López Henestrosa, José Carlos'. The profile includes a circular profile picture, the name, and a blue 'Editar perfil' button. Below the profile are sections for 'Información Personal', 'Dirección de correo:', and 'Sí':. On the right, a terminal window titled 'jc@jc: ~' shows command-line output. The user runs 'service fail2ban status' which shows the service is active and running. They then run 'fail2ban-client status' and 'sudo fail2ban-client status', both of which show a single jail named 'sshd'.

```
jc@jc:~$ service fail2ban status
● fail2ban.service - Fail2Ban Service
    Loaded: loaded (/usr/lib/systemd/system/fail2ban.service; enabled; preset: enabled)
    Active: active (running) since Sat 2025-02-08 16:18:23 UTC; 3min 15s ago
      Docs: man:fail2ban(1)
      Main PID: 6933 (fail2ban-server)
        Tasks: 5 (limit: 4539)
       Memory: 25.2M (peak: 25.7M)
         CPU: 519ms
        CGroup: /system.slice/fail2ban.service
                └─6933 /usr/bin/python3 /usr/bin/fail2ban-server -xf start

feb 08 16:18:23 jc systemd[1]: Started fail2ban.service - Fail2Ban Service.
feb 08 16:18:23 jc fail2ban-server[6933]: 2025-02-08 16:18:23,396 fail2ban.configread
feb 08 16:18:23 jc fail2ban-server[6933]: Server ready
lines 1-14/14 (END)

jc@jc:~$ fail2ban-client status
2025-02-08 16:21:52,642 fail2ban                         [7127]: ERROR  Permission denied to
socket: /var/run/fail2ban/fail2ban.sock, (you must be root)
jc@jc:~$ sudo fail2ban-client status
Status
|- Number of jail:      1
`- Jail list:   sshd
jc@jc:~$
```

Comprobación de que fail2ban está activo en el sistema

# Configuración de fail2ban

---

Dado que fail2ban es un programa que tiene bastante trayectoria, las posibilidades que ofrece son muy amplias. Realizaremos una configuración base para que veamos su funcionamiento ante un ataque por fuerza bruta.

fail2ban utiliza lo que llama *jails* (cárcel), las cuales son reglas o filtros que supervisan los registros del sistema en busca de intentos de acceso no autorizados y aplican bloqueos si detectan actividad sospechosa. Estas se utilizan para configurar los diferentes servicios que ofrece fail2ban. Nosotros nos centraremos principalmente en un *jail* para SSH.

Para ello, primero hacemos una copia del fichero inicial de configuración:

```
cp -p /etc/fail2ban/jail.conf /etc/fail2ban/jail.local
```

Ahora trabajaremos con el fichero `jail.local`, por lo que lo editamos con el siguiente comando:

```
sudo vim /etc/fail2ban/jail.local
```

En la sección `[DEFAULT]` indicaremos las siguientes directivas:

```
[DEFAULT]
# Comando personalizado que se ejecutará antes de bloquear una dirección IP.
# En este caso, no se ejecuta ninguno.
ignorecommand =

# Tiempo que una IP permanecerá baneada.
bantime = 240s

# Tiempo en el que deben ocurrir los intentos fallidos para activar el baneo.
findtime = 5m

# Número máximo de intentos fallidos dentro del findtime antes de banear
maxretry = 3
```

The screenshot shows a dual-pane interface. On the left, a web browser window displays a user profile for 'López Henestrosa, José Carlos'. The profile includes a circular profile picture, the name, and a 'Editar perfil' button. Below the profile, there's some text about personal information and email. On the right, a terminal window titled 'jc@jc: ~' shows the contents of the '/etc/fail2ban/jail.local' file. The file contains several sections of commented-out configuration code related to fail2ban's jail mechanism.

```

# The DEFAULT allows a global definition of the options. They can be overridden
# in each jail afterwards.

[DEFAULT]

# External command that will take an tagged arguments to ignore, e.g. <ip>,
# and return true if the IP is to be ignored. False otherwise.
#
# ignorecommand = /path/to/command <ip>
ignorecommand =

# "bantime" is the number of seconds that a host is banned.
bantime = 240s

# A host is banned if it has generated "maxretry" during the last "findtime"
# seconds.
findtime = 5m

# "maxretry" is the number of failures before a host get banned.
maxretry = 3

# "maxmatches" is the number of matches stored in ticket (resolvable via tag
<matches> in actions).
maxmatches = %(maxretry)s

```

Configuración de la sección [DEFAULT] del archivo /etc/fail2ban/jail.local

A continuación, buscaremos el *jail* de SSH en ese mismo fichero, en la sección [sshd], y pondremos las siguientes directivas:

```

[sshd]
# Como su valor es true, activa la jail para SSH
enabled = true

# Protege el puerto 22 (o el configurado para SSH)
port = ssh

# Usa el filtro sshd.conf para detectar accesos fallidos
filter = sshd

# Busca intentos fallidos en este archivo
logpath = /var/log/auth.log

# Bloquea una IP tras 3 intentos fallidos
maxretry = 3

# Método usado para leer logs (systemd, polling, etc.).
backend = %(sshd_backend)s

```

The screenshot shows a Linux desktop interface. On the left, there's a sidebar with a user profile picture, name (López Henestrosa, José Carlos), and a 'Editar perfil' button. Below that are sections for 'Información Personal' (with placeholder text) and 'Ección de correo:' (nestrosa@conh@gmail.com). On the right is a terminal window titled 'jc@jc: ~'. It displays the configuration of the [sshd] section in the /etc/fail2ban/jail.local file. The configuration includes parameters like mode (normal), enabled (true), port (ssh), filter (sshd), logpath (%(sshd\_log)s), maxretry (3), and backend (%(sshd\_backend)s). The terminal status bar shows '271,0-1' and '27%'. The overall theme is dark.

```
#  
# SSH servers  
#  
[sshd]  
:  
# To use more aggressive sshd modes set filter parameter "mode" in jail.local  
:  
# normal (default), ddos, extra or aggressive (combines all).  
# See "tests/files/logs/sshd" or "filter.d/sshd.conf" for usage example and d  
etails.  
#mode    = normal  
enabled  = true  
port     = ssh  
filter   = sshd  
logpath  = %(sshd_log)s  
maxretry = 3  
backend  = %(sshd_backend)s
```

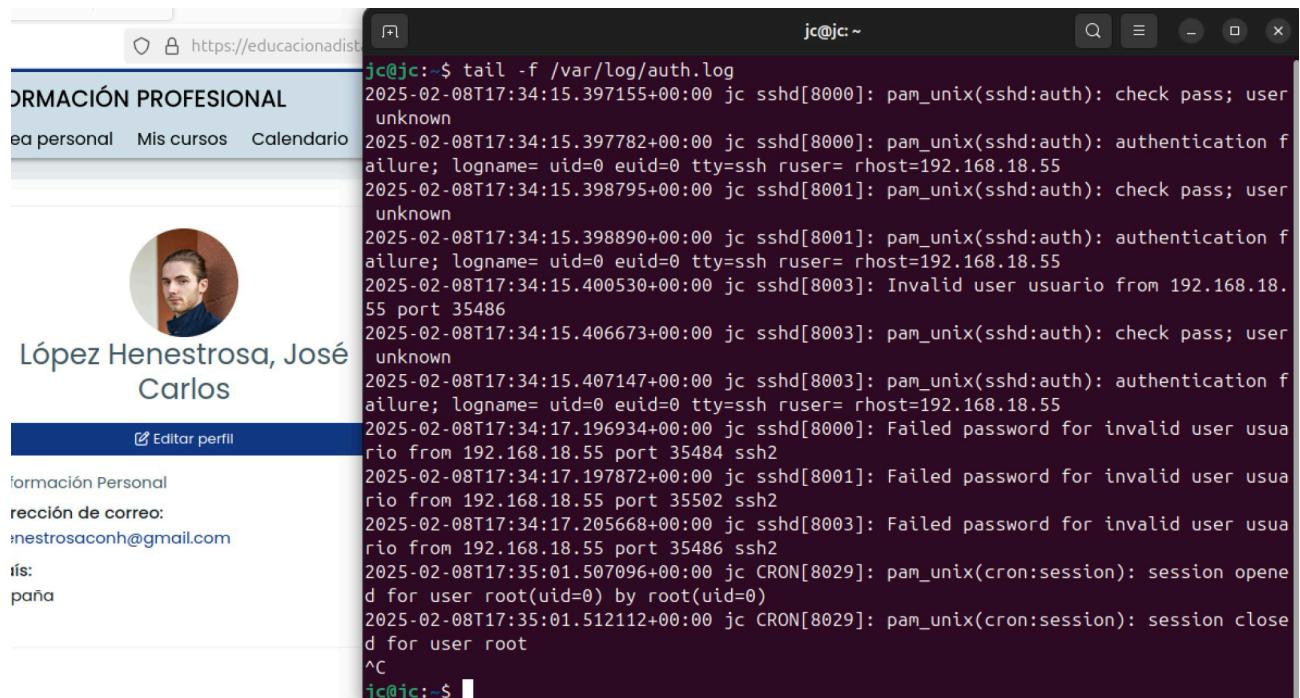
Configuración de la sección [sshd] del archivo /etc/fail2ban/jail.local

Una vez configurado el archivo, lo guardamos y reiniciamos el servicio de fail2ban con este comando:

```
sudo service restart fail2ban
```

# Ataque y reacción de fail2ban

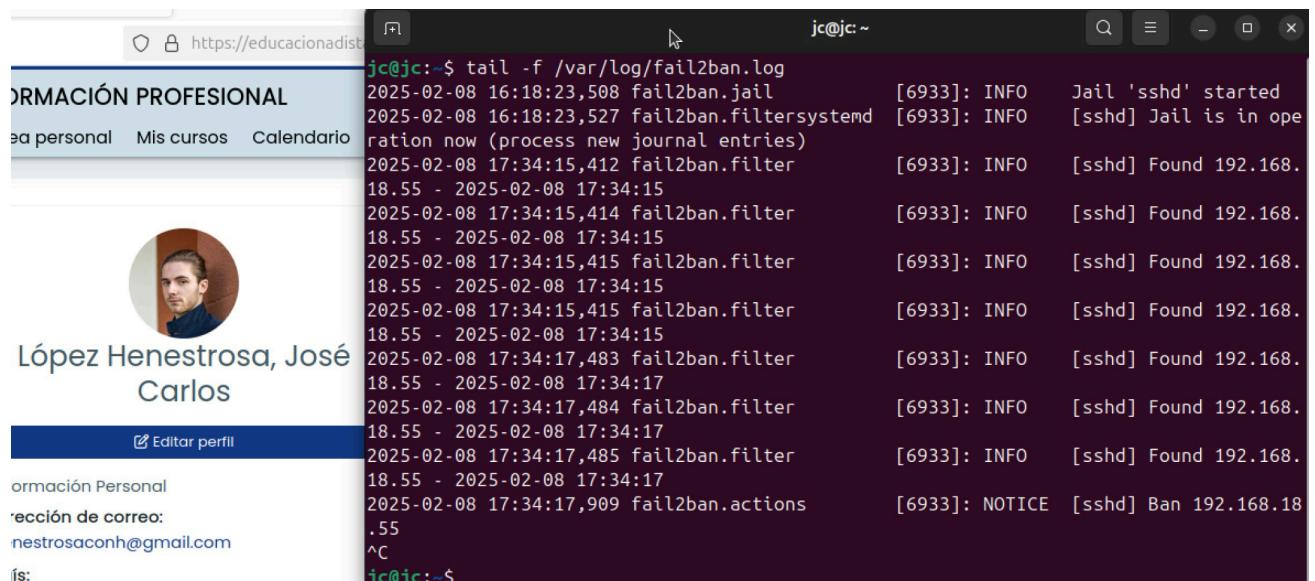
Ahora repetiremos el mismo ataque que al principio de la práctica. No obstante, a diferencia del apartado Estado del sistema previo a fail2ban, ahora prestaremos atención a dos ficheros de log en lugar de a uno: `/var/log/auth.log` / `/var/log/fail2ban.log`.



The screenshot shows a web browser window with a URL of `https://educacionadistancia.es`. On the left, there's a sidebar with a profile picture of a man, the name "López Henestrosa, José Carlos", and a "Editar perfil" button. The main content area shows a "FORMACIÓN PROFESIONAL" section with links for "Carrera personal", "Mis cursos", and "Calendario". On the right, a terminal window titled "jc@jc: ~" displays the command `tail -f /var/log/auth.log`. The log output shows multiple failed SSH login attempts from the IP address 192.168.18.55, with timestamps ranging from February 8, 2025, at 17:34 to 17:35. It also shows entries for CRON tasks and session openings.

```
jc@jc:~$ tail -f /var/log/auth.log
2025-02-08T17:34:15.397155+00:00 jc sshd[8000]: pam_unix(sshd:auth): check pass; user unknown
2025-02-08T17:34:15.397782+00:00 jc sshd[8000]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.18.55
2025-02-08T17:34:15.398795+00:00 jc sshd[8001]: pam_unix(sshd:auth): check pass; user unknown
2025-02-08T17:34:15.398890+00:00 jc sshd[8001]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.18.55
2025-02-08T17:34:15.400530+00:00 jc sshd[8003]: Invalid user usuario from 192.168.18.55 port 35486
2025-02-08T17:34:15.406673+00:00 jc sshd[8003]: pam_unix(sshd:auth): check pass; user unknown
2025-02-08T17:34:15.407147+00:00 jc sshd[8003]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.18.55
2025-02-08T17:34:17.196934+00:00 jc sshd[8000]: Failed password for invalid user usuario from 192.168.18.55 port 35484 ssh2
2025-02-08T17:34:17.197872+00:00 jc sshd[8001]: Failed password for invalid user usuario from 192.168.18.55 port 35502 ssh2
2025-02-08T17:34:17.205668+00:00 jc sshd[8003]: Failed password for invalid user usuario from 192.168.18.55 port 35486 ssh2
2025-02-08T17:35:01.507096+00:00 jc CRON[8029]: pam_unix(cron:session): session opened for user root(uid=0) by root(uid=0)
2025-02-08T17:35:01.512112+00:00 jc CRON[8029]: pam_unix(cron:session): session closed for user root
^C
jc@jc:~$
```

Contenido del archivo `/var/log/auth.log` que refleja los intentos de login externos por SSH



The screenshot shows a web browser window with a URL of `https://educacionadistancia.es`. On the left, there's a sidebar with a profile picture of a man, the name "López Henestrosa, José Carlos", and a "Editar perfil" button. The main content area shows a "FORMACIÓN PROFESIONAL" section with links for "Carrera personal", "Mis cursos", and "Calendario". On the right, a terminal window titled "jc@jc: ~" displays the command `tail -f /var/log/fail2ban.log`. The log output shows activity from the IP address 192.168.18.55, including entries for fail2ban.jail starting, fail2ban.filtersystemd, fail2ban.filter rules being triggered, and fail2ban.actions banning the IP.

```
jc@jc:~$ tail -f /var/log/fail2ban.log
2025-02-08 16:18:23,508 fail2ban.jail [6933]: INFO Jail 'sshd' started
2025-02-08 16:18:23,527 fail2ban.filtersystemd [6933]: INFO [sshd] Jail is in operation now (process new journal entries)
2025-02-08 17:34:15,412 fail2ban.filter [6933]: INFO [sshd] Found 192.168.18.55 - 2025-02-08 17:34:15
2025-02-08 17:34:15,414 fail2ban.filter [6933]: INFO [sshd] Found 192.168.18.55 - 2025-02-08 17:34:15
2025-02-08 17:34:15,415 fail2ban.filter [6933]: INFO [sshd] Found 192.168.18.55 - 2025-02-08 17:34:15
2025-02-08 17:34:15,415 fail2ban.filter [6933]: INFO [sshd] Found 192.168.18.55 - 2025-02-08 17:34:15
2025-02-08 17:34:17,483 fail2ban.filter [6933]: INFO [sshd] Found 192.168.18.55 - 2025-02-08 17:34:17
2025-02-08 17:34:17,484 fail2ban.filter [6933]: INFO [sshd] Found 192.168.18.55 - 2025-02-08 17:34:17
2025-02-08 17:34:17,485 fail2ban.filter [6933]: INFO [sshd] Found 192.168.18.55 - 2025-02-08 17:34:17
2025-02-08 17:34:17,909 fail2ban.actions [6933]: NOTICE [sshd] Ban 192.168.18.55
^C
jc@jc:~$
```

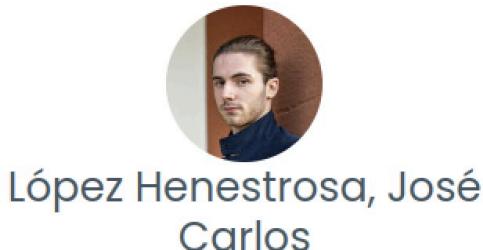
Contenido del archivo `/var/log/fail2ban.log` que refleja la actividad recibida de la IP de la máquina desde la que se han lanzado los ataques y su posterior baneo

Como podemos apreciar, una vez que se cumplen las condiciones marcadas en el fichero de configuración `/etc/fail2ban/jail.local`, se bloqueará la IP desde la que se está realizando el ataque durante el tiempo que le indiquemos.

# Comprobación de direcciones IP baneadas

En continuación con el apartado anterior, comprobaremos la lista de las direcciones IP baneadas en el momento de ejecutar el siguiente comando (recuerda que el baneo es temporal):

```
sudo fail2ban-client get sshd banip
```



```
jc@jc:~$ sudo fail2ban-client get sshd banip
[sudo] password for jc:
192.168.18.55
jc@jc:~$
```

Lista de direcciones IP baneadas

Por último, podremos comprobar que el método de baneo es el de incorporar una nueva regla al firewall `iptables`, donde se indica que la IP en cuestión es rechazada. Lo podremos ver con el comando:

```
sudo iptables -L
```

```
jc@jc:~$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
f2b-sshd  tcp  --  anywhere             anywhere            multiport dport
      s ssh

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain f2b-sshd (1 references)
target     prot opt source               destination
REJECT    all  --  192.168.18.55        anywhere            reject-with icmp
      p-port-unreachable
      RETURN   all  --  anywhere           anywhere
```

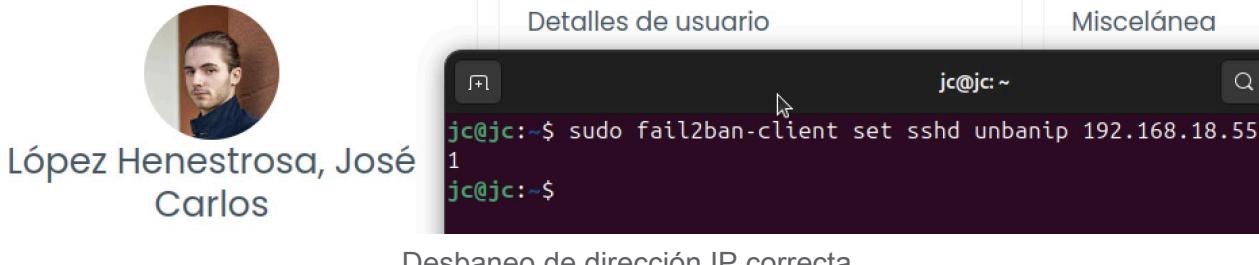
Comprobación de `iptables`

Como podemos ver, la IP de la máquina desde la que hemos ejecutado el ataque se ha añadido a la cadena `f2b-sshd` (`fail2ban`), ya que se está bloqueando todo el tráfico proveniente de la IP `192.168.18.55` y se está respondiendo con un mensaje ICMP Port Unreachable.

# Desbaneo de direcciones IP

Puede que nuestro `fail2ban` haya baneado alguna IP que podemos considerar “correcta”. Ante tal situación, podemos desbanear la IP con este comando:

```
sudo fail2ban-client set sshd unbanip 192.168.18.55
```



Como podemos ver, el comando devuelve 1, lo que indica que la IP se ha desbaneado correctamente. En caso de no desbanear a la IP, devolvería el valor 0.

Por otro lado, y de cara a que no ocurra nuevamente, podemos incluir nuestra IP o conjunto de direcciones IP en una *whitelist* (lista blanca) para que no sean baneadas en el futuro. Esto lo indicaremos una vez más añadiendo la directiva `ignoreip = 192.168.18.55` a la sección [DEFAULT] del fichero `/etc/fail2ban/jail.local`. Podemos indicar tanto una IP concreta como un rango en formato CIDR.



IP de la máquina atacante añadida a la *whitelist* de `fail2ban`

Por último, reiniciamos el servicio `fail2ban` y probamos que la IP no es baneada tras lanzar el ataque con `hydra`.

○ 命令 https://educacionadista

**FORMACIÓN PROFESIONAL**

Área personal Mis cursos Calendario



López Henestrosa, José  
Carlos

[Editar perfil](#)

Información Personal

Dirección de correo:  
[henestrosaonh@gmail.com](mailto:henestrosaonh@gmail.com)

País:  
España

```
jc@jc:~$ tail -f /var/log/fail2ban.log
2025-02-09 01:26:40,084 fail2ban.filter
[10705]: INFO      maxRetry: 3
2025-02-09 01:26:40,084 fail2ban.filter
[10705]: INFO      findtime: 300
2025-02-09 01:26:40,084 fail2ban.actions
[10705]: INFO      banTime: 240
2025-02-09 01:26:40,084 fail2ban.filter
[10705]: INFO      encoding: UTF-8
2025-02-09 01:26:40,084 fail2ban.filter
[10705]: INFO      Added logfile: '/var/log/auth.log' (pos = 19415, hash = f8d5bb1dfc1436b3c27abb40ed104a30d72ad1eb)
2025-02-09 01:26:40,087 fail2ban.filter
[10705]: INFO      [sshd] Ignore 192.168.18.55 by ip
2025-02-09 01:26:40,087 fail2ban.jail
[10705]: INFO      Jail 'sshd' started
2025-02-09 01:27:03,872 fail2ban.filter
[10705]: INFO      [sshd] Ignore 192.168.18.55 by ip
2025-02-09 01:27:03,928 fail2ban.filter
[10705]: INFO      [sshd] Ignore 192.168.18.55 by ip
2025-02-09 01:27:06,650 fail2ban.filter
[10705]: INFO      [sshd] Ignore 192.168.18.55 by ip
```

Ataque con IP ignorada

# Envío de email

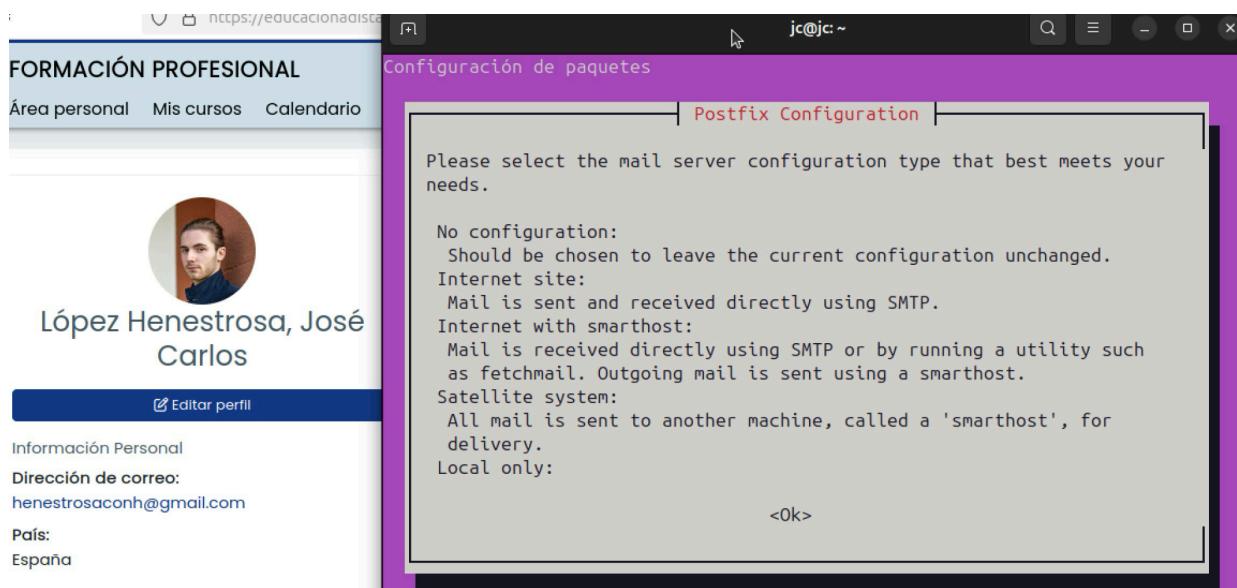
Ahora debemos buscar la manera de enviar un correo electrónico al administrador desde el propio servidor para notificarle que se ha producido una alerta. Para ello, seguiremos los siguientes pasos:

## 1. Configurar un método para enviar correos

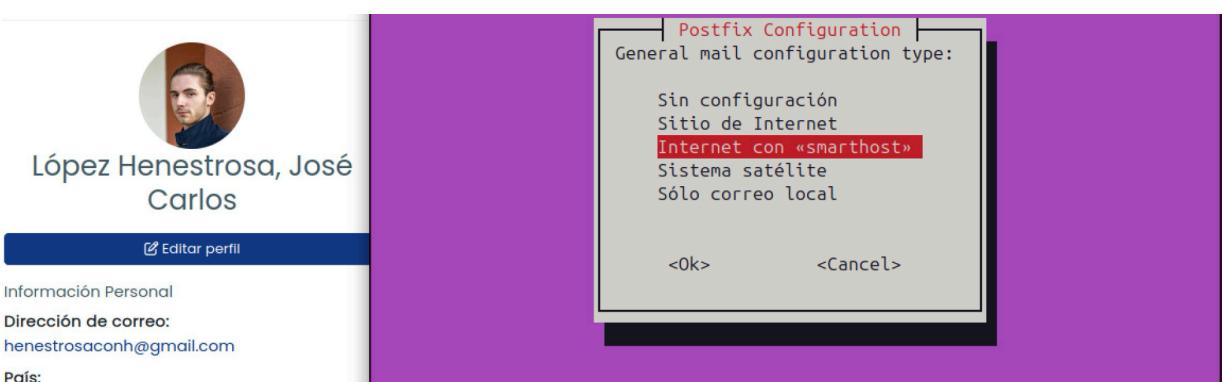
fail2ban puede usar sendmail o mailx para enviar notificaciones. Nosotros usaremos mailx, por lo que lo instalamos el paquete mailutils en Ubuntu:

```
sudo apt install mailutils
```

Al ejecutar el comando, tendremos que seleccionar la configuración del servidor de correo. En nuestro caso, como solo tenemos que enviar alertas con un correo de Gmail, la opción más simple es “Internet site”.

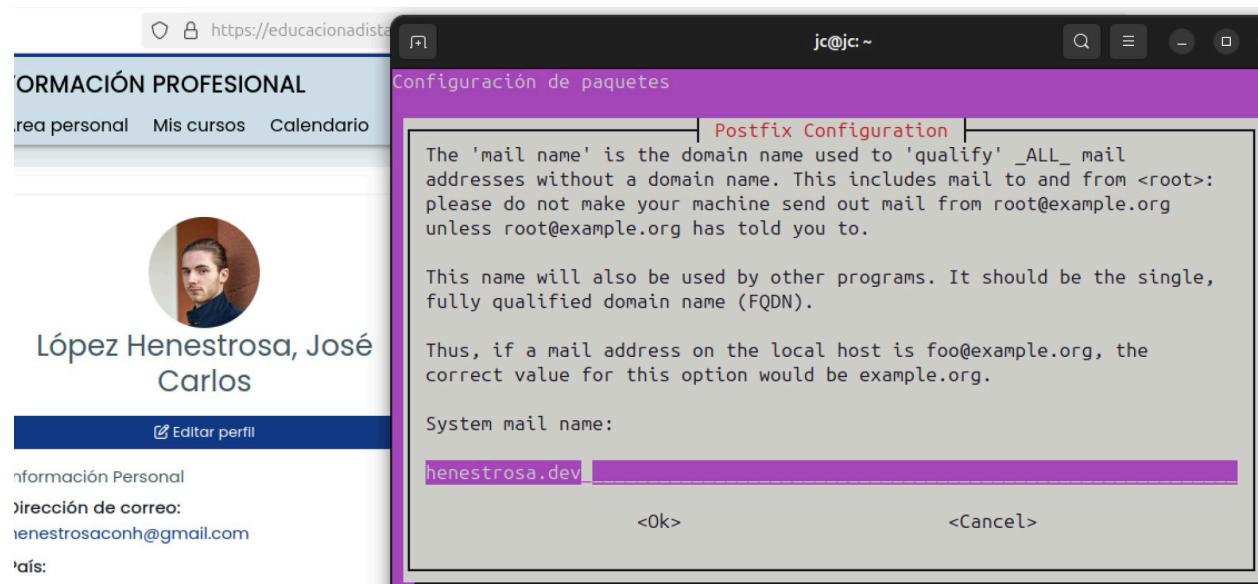


Pantalla de configuración al instalar el paquete mailutils



Pantalla de configuración al instalar el paquete mailutils

Luego, encontraremos otro ajuste con respecto a **System mail name** (el nombre de correo del sistema), en el que tendremos que introducir nuestro nombre de dominio. Se puede dejar en blanco si no tenemos nombre de dominio, ya que vamos a utilizar Gmail en lugar del servidor local como servidor de correo.



Pantalla de configuración al instalar el paquete `mailutils`

Con esto, ya tendremos, por el momento, la configuración inicial de `postfix`.

## 2. Instalar `libsas12-modules`

Este paquete contiene los módulos necesarios para la implementación de la autenticación **SASL (Simple Authentication and Security Layer)**, el cual es un marco de autenticación que se utiliza para añadir soporte de autenticación a servicios como el correo electrónico. En el caso de `postfix`, SASL es utilizado para autenticar el servidor de correo con un servidor remoto, como Gmail.

Procedemos a instalarlo con el siguiente comando:

```
sudo apt install libsas12-modules
```

## 3. Configurar `postfix` para usar Gmail

Abrimos la configuración de `postfix`:

```
sudo vim /etc/postfix/main.cf
```

Y añadimos o modificamos las siguientes líneas al final del archivo para que apunten a Gmail:

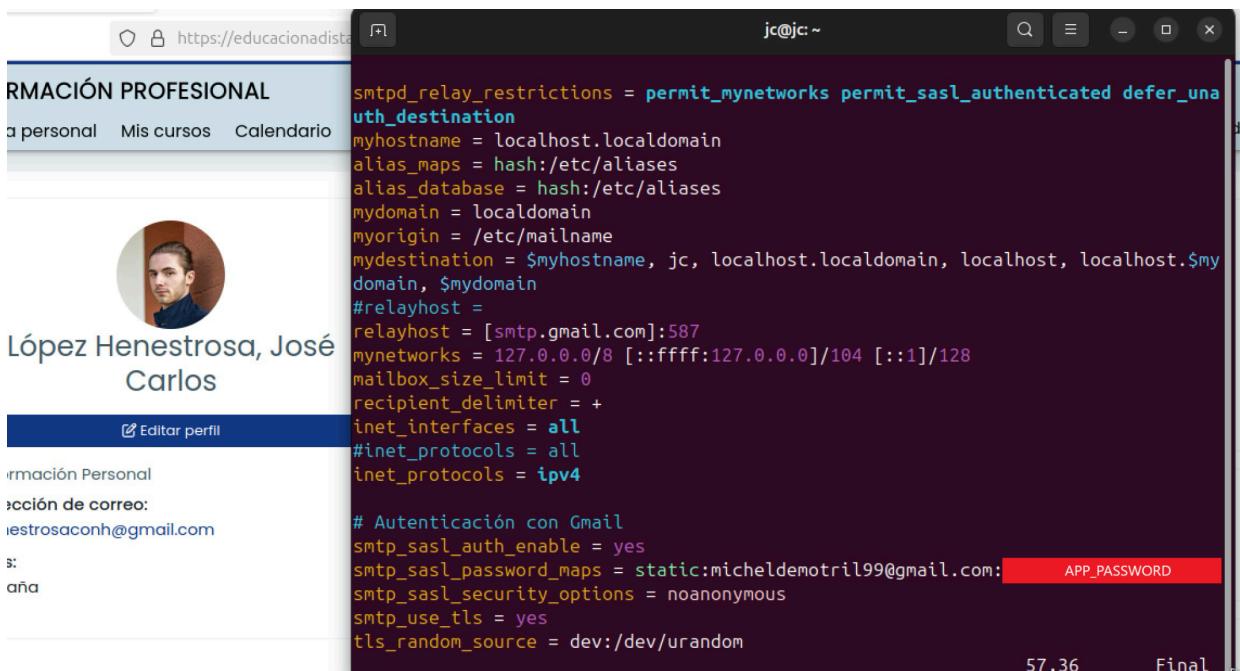
```
# Configuración para usar Gmail SMTP
relayhost = [smtp.gmail.com]:587
```

```

# Autenticación con Gmail
smtp_sasl_auth_enable = yes
smtp_sasl_password_maps = static:correo@gmail.com:app_password
smtp_sasl_security_options = noanonymous
smtp_use_tls = yes
tls_random_source = dev:/dev/urandom

# Otras configuraciones necesarias
myhostname = localhost.localdomain
mydomain = localdomain
myorigin = /etc/mailname
inet_interfaces = all
inet_protocols = ipv4

```



Configuración aplicada al archivo /etc/postfix/main.cf

Para generar la APP\_PASSWORD con una cuenta de Google, es conveniente visitar [este enlace](#).

#### 4. Configurar el nombre de la máquina de correo (FDQN)

Editamos el archivo /etc/mailname y sustituimos el contenido por lo siguiente:

```
localhost.localdomain
```

#### 5. Reiniciar postfix

Tras esto, reiniciamos postfix para que los cambios surtan efecto:

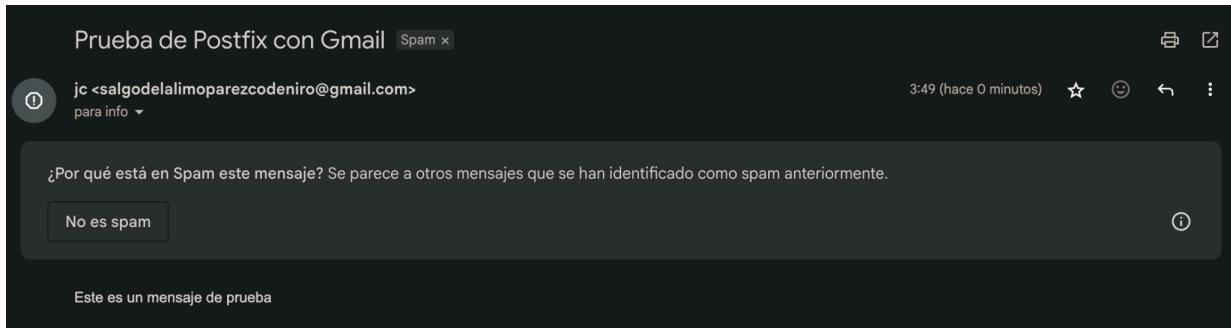
```
sudo systemctl restart postfix
```

## 6. Probar la configuración

Para asegurarnos de que `postfix` está configurado correctamente y que los correos se envían a través de Gmail, podemos enviar un correo de prueba:

```
echo "Este es un mensaje de prueba" | mail -s "Prueba de Postfix con Gmail" info@henestrosa.dev
```

Acto seguido, revisamos la bandeja de spam del correo `info@henestrosa.dev` para comprobar que, efectivamente, hemos recibido el correo de prueba correctamente.



Correo enviado con el comando de arriba

## 7. Configurar fail2ban para enviar correos

Tenemos que volver a modificar el archivo `/etc/fail2ban/jail.local` y modificar las siguientes líneas:

```
[DEFAULT]
# Dirección del administrador que recibirá los correos
destemail = info@henestrosa.dev

# Dirección desde la que se envían los correos
sender = fail2ban@henestrosa.dev

# Método de envío de correos (mail, sendmail o exim)
mta = mail

# Acción que fail2ban tomará. En este caso, envía el correo electrónico con
# el log y el archivo de acción.
action = %(action_mwl)s
```

```

# ACTIONS
#
# Destination email address used solely for the interpolations in
# jail.{conf,local,d/*} configuration files.
destemail = info@henestrosa.dev

# Sender email address used solely for some actions
#sender = root@<fq-hostname>
sender = salgodelalimoparezcodeniro@gmail.com

# E-mail action. Since 0.8.1 Fail2Ban uses sendmail MTA for the
# mailing. Change mta configuration parameter to mail if you want to
# revert to conventional 'mail'.
mta = sendmail

# Choose default action. To change, just override value of 'action' with the
# interpolation to the chosen action shortcut (e.g. action_mw, action_mwl, etc) in jail
.local
# globally (section [DEFAULT]) or per specific section
#action = %(action_)s
action = %(action_mwl)s

```

Cambios realizados en el archivo /etc/fail2ban/jail.local

Tras realizar los cambios, reiniciamos fail2ban para aplicar la nueva configuración:

## 8. Verificar que los correos se envían correctamente

Para asegurarnos de que fail2ban está enviando los correos correctamente, podemos forzar una prueba de intento de registro fallido con este comando:

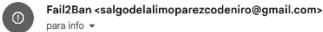
```
sudo fail2ban-client set sshd banip 1.2.3.4
```

```
jc@jc:~$ sudo fail2ban-client set sshd banip 1.2.3.4
1
jc@jc:~$
```

Baneo manual de una IP para recibir el correo

A continuación, miramos la bandeja de *spam* de la dirección de correo que configuramos como destemail (info@henestrosa.dev) y comprobamos que, efectivamente, hemos recibido el correo con éxito:

[Fail2Ban] sshd: banned 1.2.3.4 from jc Spam x

 4:31 (hace 0 minutos) ⚡ ⓘ

¿Por qué está en Spam este mensaje? Se parece a otros mensajes que se han identificado como spam anteriormente.

No es spam

Traducir al español X

Hi,

The IP 1.2.3.4 has just been banned by Fail2Ban after 0 attempts against sshd.

Here is more information about 1.2.3.4 :  
% [\[whois.apnic.net\]](http://whois.apnic.net)  
% Whois data copyright terms <http://www.apnic.net/db/dbccopyright.html>

% Information related to '1.2.3.0 - 1.2.3.255'

% Abuse contact for '1.2.3.0 - 1.2.3.255' is [helpdesk@apnic.net](mailto:helpdesk@apnic.net)

```

inetnum: 1.2.3.0 - 1.2.3.255
netname: Debugon-prefix
descr: APNIC Debugon Project
descr: APNIC Pty Ltd
country: AU
org: ORG-RQA1-AP
admin-c: AR302-AP
tech-c: AR302-AP
abuse-c: AA1412-AP
status: ASSIGNED PORTABLE
mnt-by: APNIC-HM
mnt-routes: MAINT-AU-APNIC-GM85-AP
mnt-irt: IRT-APNICRANDNET-AU
last-modified: 2020-11-25T06:34:44Z
source: APNIC

irt: IRT-APNICRANDNET-AU
address: PO Box 3646
address: South Brisbane, QLD 4101
address: Australia
e-mail: helpdesk@apnic.net
abuse-mailbox: helpdesk@apnic.net
admin-c: AR302-AP
tech-c: AR302-AP
auth: # Filtered
remarks: helpdesk@apnic.net was validated on 2021-02-09
mnt-by: MAINT-AU-APNIC-GM85-AP
last-modified: 2021-03-09T01:10:21Z
source: APNIC

organisation: ORG-RQA1-AP
org-name: Resource Quality Assurance
org-type: LIR
country: AU
address: 6 Cordelia Street, South Brisbane
e-mail: research@apnic.net
mnt-ref: APNIC-HM
mnt-by: APNIC-HM
last-modified: 2023-09-05T02:15:46Z
source: APNIC

role: ABUSE APNICRANDNETAU
address: PO Box 3646
address: South Brisbane, QLD 4101
address: Australia
country: ZZ
phone: +000000000
e-mail: helpdesk@apnic.net
admin-c: AR302-AP
tech-c: AR302-AP
nic-hdl: AA1412-AP
remarks: Generated from irt object IRT-APNICRANDNET-AU
abuse-mailbox: helpdesk@apnic.net
mnt-by: APNIC-ABUSE
last-modified: 2021-03-09T01:10:22Z
source: APNIC

role: APNIC RESEARCH
address: PO Box 3646
address: South Brisbane, QLD 4101
address: Australia
country: AU
phone: +61-7-3858-3188
fax-no: +61-7-3858-3199
e-mail: research@apnic.net
nic-hdl: AR302-AP
tech-c: AH256-AP
admin-c: AH256-AP
mnt-by: MAINT-APNIC-AP
last-modified: 2018-04-04T04:26:04Z
source: APNIC

```

% This query was served by the APNIC Whois Service version 1.88.25 (WHOIS-UK4)

Lines containing failures of 1.2.3.4 (max 1000)  
2025-02-11T03:24:59.594799+00:00 jc sudo: jc : TTY=pts/0 ; PWD=/home/jc ; USER=root ; COMMAND=/usr/bin/fail2ban-client set sshd banip 1.2.3.4  
2025-02-11T03:31:40.502459+00:00 jc sudo: jc : TTY=pts/0 ; PWD=/home/jc ; USER=root ; COMMAND=/usr/bin/fail2ban-client set sshd banip 1.2.3.4

Regards,  
Fail2Ban

## Prueba de recepción de correo con detalle de IP baneada

# Protección de otros servicios con fail2ban

fail2ban permite ampliar el ámbito de protección más allá de SSH, por lo que vamos a probar a bastionar otro servicio contra ataques de fuerza bruta con dicho paquete, como el servidor web Apache.

Para ello, seguimos los siguientes pasos:

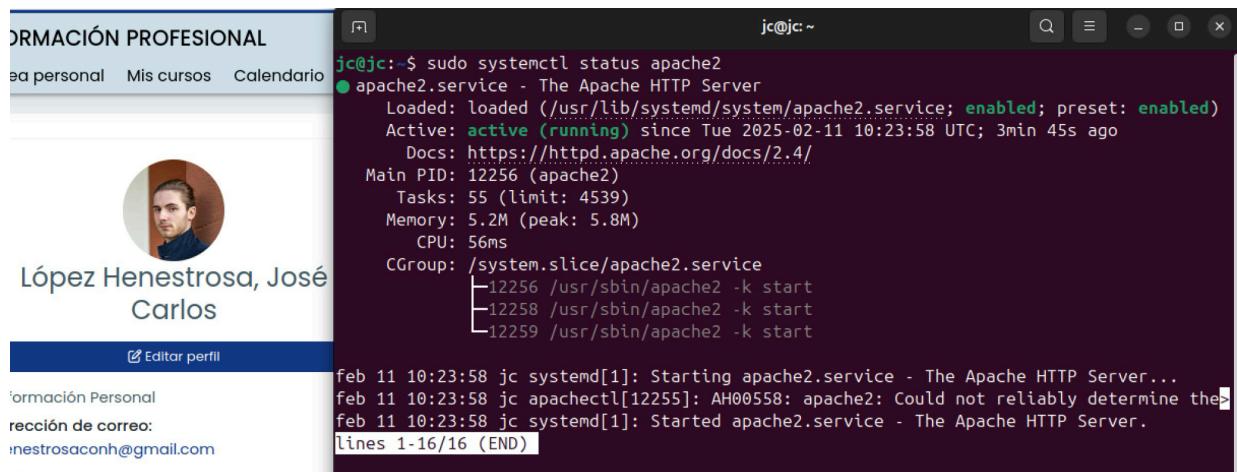
## 1. Instalar Apache

Ejecutamos el siguiente comando:

```
sudo apt install apache2
```

Y verificamos que está en ejecución:

```
sudo systemctl status apache2
```



Apache funcionando correctamente

## 2. Configurar jail para Apache

Añadimos la siguiente configuración al archivo `/etc/fail2ban/jail.local` a la sección `[apache-auth]` para proteger Apache:

```
[apache-auth]
enabled = true
port = http,https
filter = apache-auth
logpath = /var/log/apache2/error.log
```

```
maxretry = 5
bantime = 3600
findtime = 600
```



López Henestrosa, José  
Carlos

```
[apache-auth]
enabled = true
port = http,https
filter = apache-auth
logpath = %(apache_error_log)s
maxretry = 5
bantime = 3600
findtime = 600
```

Configuración de la sección [apache-auth] del archivo /etc/fail2ban/jail.local

### 3. Configurar el filtro de fail2ban para Apache

Editamos o creamos el archivo /etc/fail2ban/filter.d/apache-auth.conf, el cual es necesario para definir el filtro que fail2ban usará para detectar intentos de acceso fallidos en los logs de Apache. Si una dirección IP aparece en los logs con estos patrones más de los permitido (maxretry en el archivo jail.local), fail2ban la bloquea.

Dentro de dicho archivo, tenemos que verificar que las directivas failregex e ignoreregex están tal que así, las cuales son las que vienen por defecto:

```
failregex = ^client (?:(?:denied by server configuration|used wrong
authentication scheme)|b
    ^user (?!)<F-USER>(?:\S*|.*?)</F-USER>
(?:(?:auth|(?:(?:oriz|entic)ation failure|not found|denied by provider))|b
    ^Authorization of user <F-USER>(?:\S*|.*?)</F-USER> to access
.*? failed|b
    ^%(auth_type)suser <F-USER>(?:\S*|.*?)</F-USER>: password
mismatch|b
    ^%(auth_type)suser `<F-USER>(?:[^']*|.*?)</F-USER>' in realm
`.+` (auth|(?:(?:oriz|entic)ation failure|not found|denied by provider))|b
    ^%(auth_type)sinvalid nonce .* received - length is not|b
    ^%(auth_type)srealm mismatch - got `(?:(?:[^']*|.*?))' but
expected|b
    ^%(auth_type)sunknown algorithm `(?:(?:[^']*|.*?))' received|b
    ^invalid qop `(?:(?:[^']*|.*?))' received|b
    ^%(auth_type)sinvalid nonce .*? received - user attempted time
travel|b
    ^(?:(?:No h|H)ostname \S+ provided via SNI(?:, but no hostname
provided| and hostname \S+ provided| for a name based virtual host))|b
ignoreregex =
```



```

failregex = ^client (?:(denied by server configuration|used wrong authentication scheme)|  
b  
        ^user (?!)<F-USER>(?:\S*|.*?)</F-USER> (?:(auth(?:oriz|entic)ation failure|n  
ot found|denied by provider)\b  
        ^Authorization of user <F-USER>(?:\S*|.*?)</F-USER> to access .?* failed\b  
        ^%(auth_type)suser <F-USER>(?:\S*|.*?)</F-USER>; password mismatch\b  
        ^%(auth_type)suser `<F-USER>(?:[^']*|.*?)</F-USER>' in realm `.+' (auth(?:or  
iz|entic)ation failure|not found|denied by provider)\b  
        ^%(auth_type)sinvalid nonce .* received - length is not\b  
        ^%(auth_type)srealm mismatch - got `(?:[^']*|.*?)' but expected\b  
        ^%(auth_type)sunknown algorithm `(?:[^']*|.*?)' received\b  
        ^invalid qop `(?:[^']*|.*?)' received\b  
        ^%(auth_type)sinvalid nonce .* received - user attempted time travel\b  
        ^(?:No h|H)ostrname \S+ provided via SNI(?:, but no hostname provided| and ho  
stname \S+ provided| for a name based virtual host)\b  
  
ignoreregex =

```

Configuración del archivo /etc/fail2ban/jail.local

#### 4. Crear un directorio protegido

Para que fail2ban registre intentos de acceso fallidos, primero tenemos que asegurarnos de que Apache tenga un directorio protegido con .htaccess, por lo que procedemos a crear uno con el siguiente comando:

```
sudo mkdir /var/www/html/protegido
```

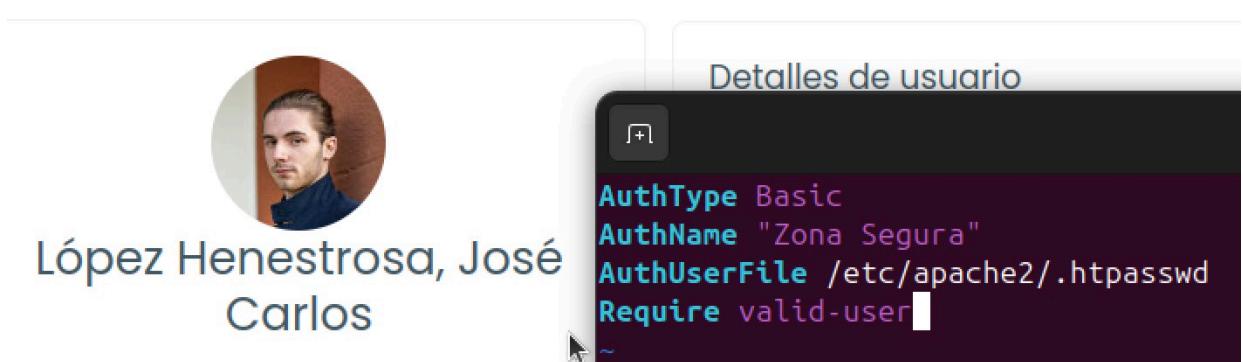
#### 5. Configurar la autenticación del directorio protegido

Creamos el archivo .htaccess dentro de /var/www/html/protegido y añadimos lo siguiente:

```

AuthType Basic
AuthName "Zona Segura"
AuthUserFile /etc/apache2/.htpasswd
Require valid-user

```



```

AuthType Basic
AuthName "Zona Segura"
AuthUserFile /etc/apache2/.htpasswd
Require valid-user

```

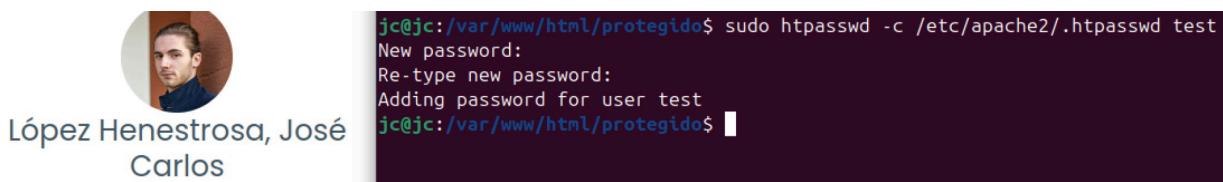
Configuración del archivo /var/www/html/protegido/.htaccess

## 6. Crear usuarios para autenticación básica

Generamos un usuario y contraseña para este directorio:

```
sudo htpasswd -c /etc/apache2/.htpasswd test
```

Al ejecutarlo, nos pedirá que introduzcamos una contraseña.



Añadiendo usuario y contraseña para el directorio protegido `/var/www/html/protegido`

## 7. Habilitar el uso de .htaccess en Apache

Editamos la configuración de Apache:

```
sudo vim /etc/apache2/sites-available/000-default.conf
```

Y buscamos la sección `<Directory /var/www/html>`, la cual tiene que estar tal que así:

```
<Directory /var/www/html>
    AllowOverride All
</Directory>
```

The screenshot shows a web browser displaying a user profile for "López Henestrosa, José Carlos". Below the profile picture, there is a link to "Editar perfil". To the right of the browser is a terminal window titled "jc@jc:/var/www/html/protegido". The terminal displays the Apache configuration file (`000-default.conf`). The relevant section is shown in yellow, which includes the `<Directory /var/www/html>` block with `AllowOverride All` set.

Configuración del archivo `/etc/apache2/sites-available/000-default.conf`

## 8. Reiniciar fail2ban y Apache

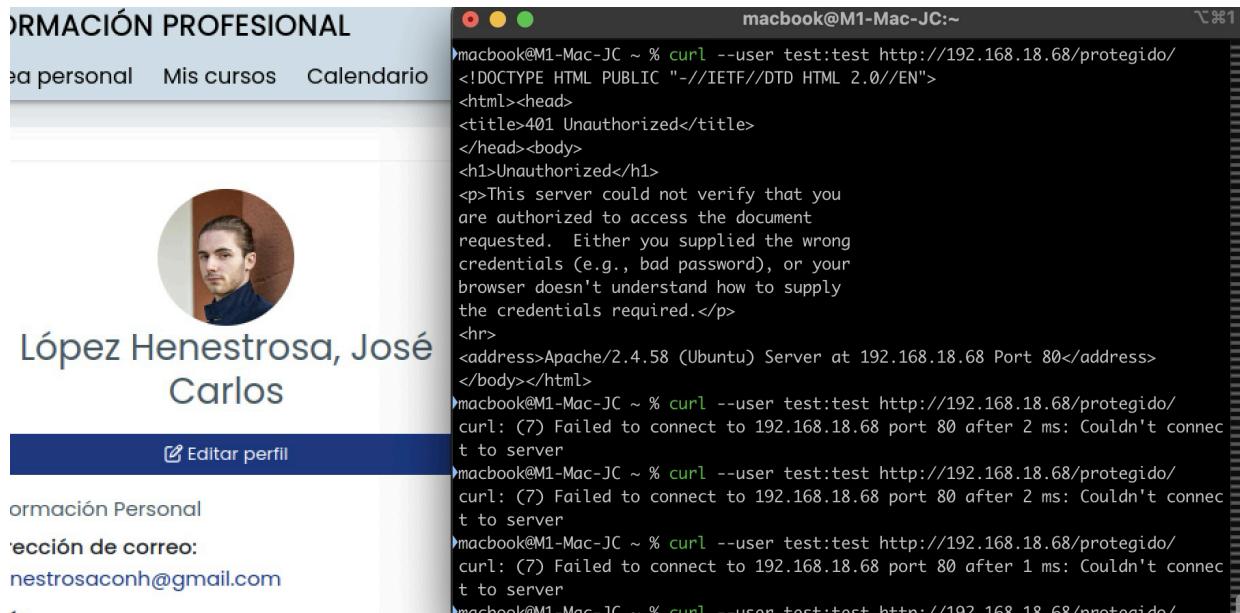
Necesario para que los cambios surtan efecto.

```
sudo systemctl restart fail2ban  
sudo systemctl restart apache2
```

## 9. Lanzar ataque contra Apache

Desde otra máquina conectada a la red local, lanzamos 5 intentos de conexión fallida con este comando:

```
curl --user test:test http://192.168.18.68/protegido/
```



The screenshot shows a web browser window with the following details:

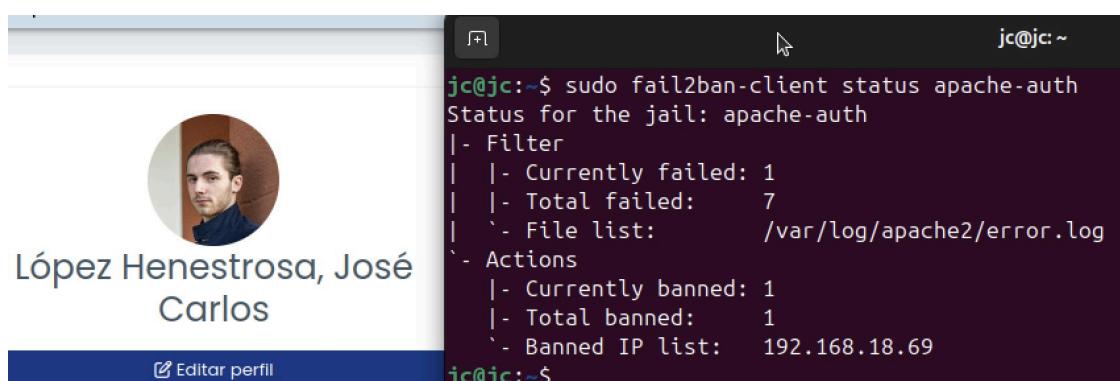
- URL:** http://192.168.18.68/protegido/
- Content:** An Apache error page indicating a 401 Unauthorized status. The message states: "This server could not verify that you are authorized to access the document requested. Either you supplied the wrong credentials (e.g., bad password), or your browser doesn't understand how to supply the credentials required."
- Terminal Log:** On the right, a terminal window titled "macbook@M1-Mac-JC:~" shows the command being run and its output, which includes multiple failed connection attempts.

Ataque desde máquina conectada a la red local

## 10. Comprobar baneo de la dirección IP de la máquina atacante

Después de 5 intentos fallidos (según `maxretry` en la sección `apache-auth` configurada anteriormente), verificamos si la IP ha sido bloqueada mediante la `jail apache-auth`:

```
sudo fail2ban-client status apache-auth
```

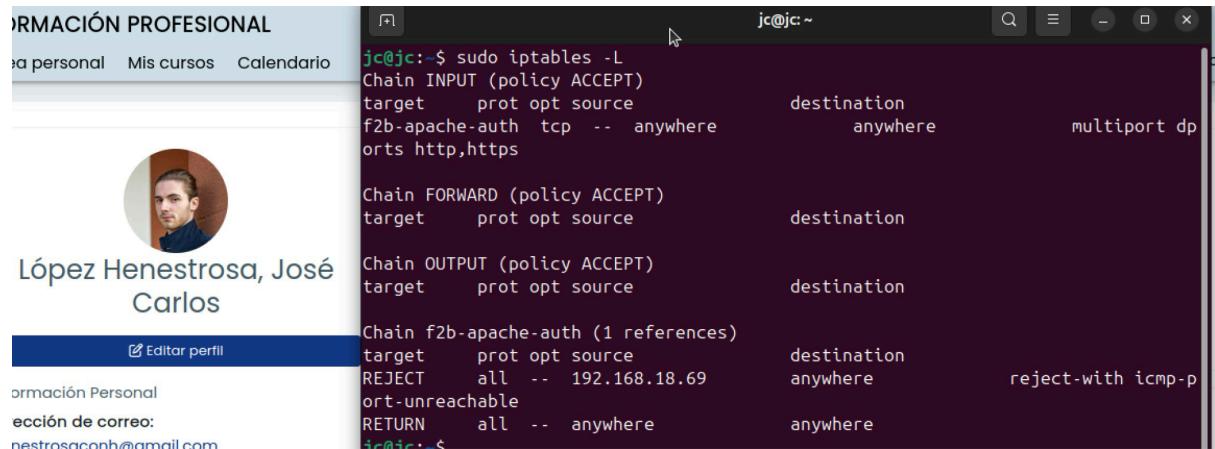


The terminal window displays the following output:

```
jc@jc:~$ sudo fail2ban-client status apache-auth
Status for the jail: apache-auth
|- Filter
| |- Currently failed: 1
| |- Total failed:    7
| `-' File list:        /var/log/apache2/error.log
`- Actions
   |- Currently banned: 1
   |- Total banned:     1
   `-' Banned IP list:   192.168.18.69
```

La dirección IP desde la que hemos realizado los intentos fallidos de autenticación está baneada

También podemos revisar `iptables`, donde se indica que la IP en cuestión es rechazada.



The screenshot shows a dual-pane interface. On the left is a user profile for 'López Henestrosa, José Carlos' with a photo, a blue 'Editar perfil' button, and contact information: 'Formación Personal' and 'dirección de correo: nestrosaconh@gmail.com'. On the right is a terminal window titled 'jc@jc: ~' showing the output of the command `sudo iptables -L`. The output lists several chains: INPUT (policy ACCEPT), FORWARD (policy ACCEPT), OUTPUT (policy ACCEPT), and f2b-apache-auth (1 reference). The f2b-apache-auth chain contains rules for REJECT and RETURN, both targeting source IP 192.168.18.69.

```
jc@jc:~$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source          destination
f2b-apache-auth  tcp  --  anywhere       anywhere        multiport dports http,https

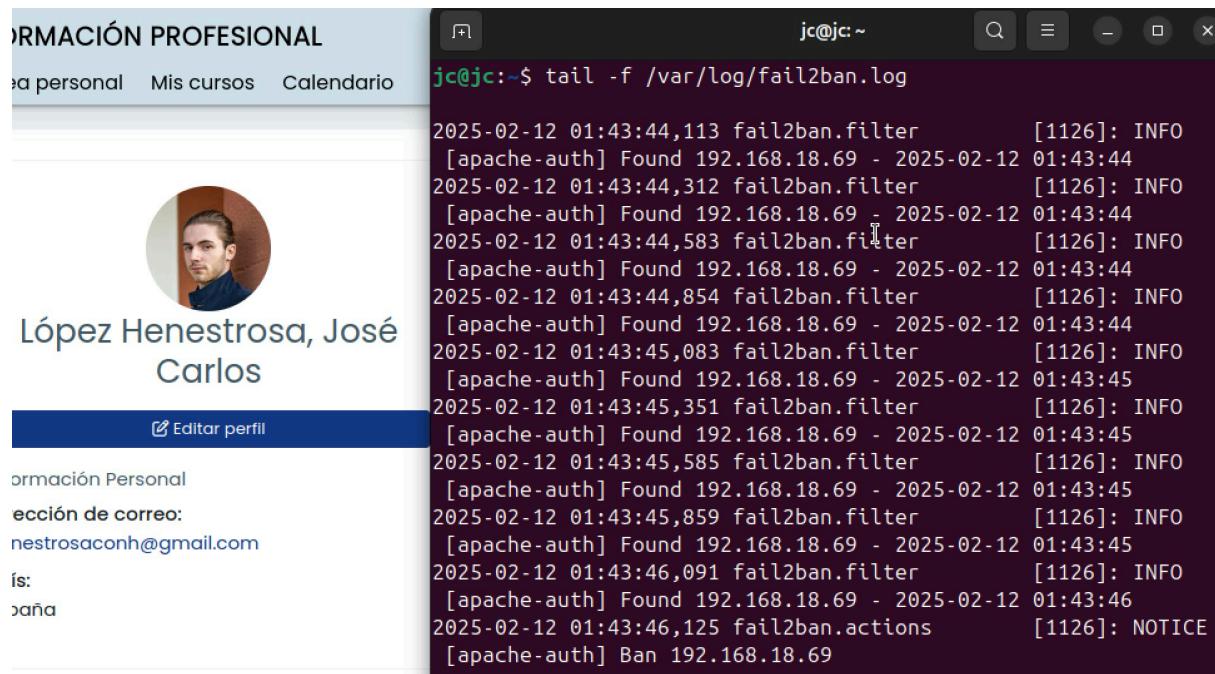
Chain FORWARD (policy ACCEPT)
target     prot opt source          destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source          destination

Chain f2b-apache-auth (1 references)
target     prot opt source          destination
REJECT    all  --  192.168.18.69      anywhere        reject-with icmp-prohibited
RETURN    all  --  anywhere        anywhere
```

Las peticiones procedentes de la máquina atacante son bloqueadas

Por último, también podemos comprobar el log de `fail2ban` donde se registra el ataque y la acción de `fail2ban` ante el ataque al servidor Apache.



The screenshot shows a dual-pane interface. On the left is a user profile for 'López Henestrosa, José Carlos' with a photo, a blue 'Editar perfil' button, and contact information: 'Formación Personal' and 'dirección de correo: nestrosaconh@gmail.com'. On the right is a terminal window titled 'jc@jc: ~' showing the output of the command `tail -f /var/log/fail2ban.log`. The log file shows multiple entries from 'fail2ban.filter' and 'fail2ban.actions' logs, indicating repeated failed login attempts from IP 192.168.18.69 and the subsequent banning of that IP.

```
jc@jc:~$ tail -f /var/log/fail2ban.log
2025-02-12 01:43:44,113 fail2ban.filter      [1126]: INFO
[apache-auth] Found 192.168.18.69 - 2025-02-12 01:43:44
2025-02-12 01:43:44,312 fail2ban.filter      [1126]: INFO
[apache-auth] Found 192.168.18.69 - 2025-02-12 01:43:44
2025-02-12 01:43:44,583 fail2ban.filter      [1126]: INFO
[apache-auth] Found 192.168.18.69 - 2025-02-12 01:43:44
2025-02-12 01:43:44,854 fail2ban.filter      [1126]: INFO
[apache-auth] Found 192.168.18.69 - 2025-02-12 01:43:44
2025-02-12 01:43:45,083 fail2ban.filter      [1126]: INFO
[apache-auth] Found 192.168.18.69 - 2025-02-12 01:43:45
2025-02-12 01:43:45,351 fail2ban.filter      [1126]: INFO
[apache-auth] Found 192.168.18.69 - 2025-02-12 01:43:45
2025-02-12 01:43:45,585 fail2ban.filter      [1126]: INFO
[apache-auth] Found 192.168.18.69 - 2025-02-12 01:43:45
2025-02-12 01:43:45,859 fail2ban.filter      [1126]: INFO
[apache-auth] Found 192.168.18.69 - 2025-02-12 01:43:45
2025-02-12 01:43:46,091 fail2ban.filter      [1126]: INFO
[apache-auth] Found 192.168.18.69 - 2025-02-12 01:43:46
2025-02-12 01:43:46,125 fail2ban.actions   [1126]: NOTICE
[apache-auth] Ban 192.168.18.69
```

fail2ban detecta y bloquea el ataque a Apache

# Bibliografía

---

- Documentación de fail2ban. <https://github.com/fail2ban/fail2ban/wiki>
- Documentación de Hydra. <https://hydra.cc/docs/intro/>
- Documentación de Apache. <https://httpd.apache.org/docs/>
- DigitalOcean. (2022, 4 de julio). *How To Protect SSH with Fail2Ban on Ubuntu 18.04.* <https://www.digitalocean.com/community/tutorials/how-to-protect-ssh-with-fail2ban-on-ubuntu-20-04>
- DigitalOcean. (2020, 6 de julio). *How To Install and Configure Postfix as a Send-Only SMTP Server on Ubuntu 18.04.* <https://www.digitalocean.com/community/tutorials/how-to-install-and-configure-postfix-as-a-send-only-smtp-server-on-ubuntu-18-04>
- UKHost4u. (2020, 6 de julio). *How To Protect an Apache Server with Fail2Ban on Ubuntu 20.04.* <https://www.ukhost4u.com/how-to-protect-an-apache-server-with-fail2ban-on-ubuntu-20-04/>
- nixCraft. (2023, 12 de agosto). *How To Set Apache Password Protected Directories With .htaccess File.* <https://www.cyberciti.biz/faq/howto-setup-apache-password-protect-directory-with-htaccess-file/>