Trabajo Práctico Final

Fail2ban

Laboratorio de Redes y Sistemas Operativos

Integrantes:

Gabriel Maugeri

Martin Fernandez Lovotti

Gonzalo Zamorano

Profesor:

José Luis Di Biase

Universidad Nacional de Quilmes

Índice

1. Introducción

Fail2ban es una herramienta de seguridad para sistemas Linux que protege contra ataques de fuerza bruta y otros intentos de acceso no deseados, monitoreando los archivos de registro de servicios como SSH, Apache, etc., y bloqueando automáticamente las direcciones IP que muestran patrones de actividad maliciosa.
En detalle, Fail2ban realiza las siguientes acciones:

1. Monitoreo de registros:
Analiza los archivos de registro de los servicios que se le configuren, buscando patrones de intentos fallidos de acceso, como múltiples inicios de sesión fallidos o intentos de acceder a archivos o directorios inexistentes.

2. Identificación de ataques:
Cuando detecta un patrón sospechoso, Fail2ban identifica la dirección IP desde la que se está produciendo el intento de ataque.

3. Bloqueo de IP:
Una vez identificada la IP maliciosa, Fail2ban utiliza el firewall del sistema (como iptables) para bloquear esa dirección IP, impidiendo que pueda realizar más intentos de acceso.

4. Desbloqueo automático:
Después de un período de tiempo configurado (por ejemplo, 10 minutos, 1 hora, etc.), Fail2ban puede desbloquear automáticamente la IP bloqueada, permitiendo nuevamente el acceso desde esa dirección, a menos que siga mostrando un comportamiento sospechoso.

2. Instalación

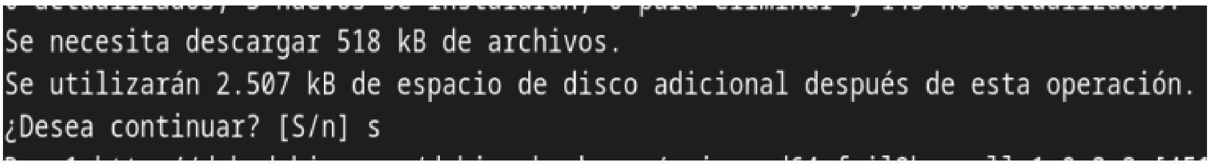
Se realizarán los siguientes pasos para instalar la versión de fail2ban v1.0.2.

Equipo utilizado para instalación, configuración y pruebas:
Procesador: 11th Gen Intel Core i5-1135G7 @ 2.40GHz 2.42 GHz
RAM: 16GB
Virtual Box (SO Linux Ubuntu 24.04.2)

1. Usar el gestor de paquetes para instalar el servicio.

   ```
   sudo apt install fail2ban
   ```

2. Permitir que instale dependencias adicionales con la opcion "s".

   Adicionalmente también puede instalar las siguientes dependencias:
   - **python3-pyasyncore**: módulo de Python para sockets asíncronos. Permite la gestión de conexiones de red sin bloquear el programa, ideal para aplicaciones que requieren alta concurrencia.
   - **python3-pyinotify**: biblioteca de python que permite monitorear cambios en el sistema de archivos en sistemas Linux.
   - **python3-setuptools**: facilita la creación y distribución de paquetes de Python.

3. Configuración

Una vez instalado, el servicio fail2ban debería estar corriendo. Para verificar el estado del mismo, ejecutar el comando:

```
:~$ service fail2ban status
```

Si el servicio está activo, se debería visualizar de la siguiente forma:

```
● fail2ban.service - Fail2Ban Service
     Loaded: loaded (/usr/lib/systemd/system/fail2ban.service; enabled; preset:>
     Active: active (running) since Thu 2025-07-03 19:34:03 -03; 5min ago
       Docs: man:fail2ban(1)
   Main PID: 4279 (fail2ban-server)
      Tasks: 5 (limit: 4609)
     Memory: 31.2M (peak: 31.7M)
        CPU: 1.231s
     CGroup: /system.slice/fail2ban.service
             └─4279 /usr/bin/python3 /usr/bin/fail2ban-server -xf start
```

En caso que el servicio no esté activo ("*Active: inactive*"), iniciar el mismo:

```
~$ service fail2ban start
```

Fail2Ban posee un archivo de configuración principal (*jail.conf*) alojado en /etc/fail2ban/

Para visualizarlo ejecutamos:

```
sudo nano /etc/fail2ban/jail.conf
```

Dentro del archivo, los parámetros de configuración principales son **ignoreip**, **bantime**, **findtime**, **maxretry**.

```
  GNU nano 7.2                          /etc/fail2ban/jail.conf

ignoreip = 127.0.0.1/8 ::1

bantime  = 10m

findtime  = 10m

maxretry = 5
```

- **ignoreip**: Ignorar una dirección ip (por defecto la dirección ip nuestra 127.0.0.1).
- **bantime**: Duración del bloqueo en segundos (por defecto 10 minutos = 600 segundos).
- **findtime**: Ventana de tiempo en segundos para buscar intentos fallidos (por defecto 10 minutos = 600 segundos).
- **maxretry**: Número de intentos fallidos antes de bloquear una IP (por defecto 5 intentos fallidos).

Para hacer modificaciones NO se altera el archivo de configuración principal (*jail.conf*), sino que se crea una copia del mismo con el nombre *jail.local*, lo que permite sobreescribir la configuración predeterminada.

Con el siguiente comando se crea y copia el archivo jail.conf:

```
sudo cp /etc/fail2ban/jail.conf /etc/fail2ban/jail.local
```

Abrir el archivo creado para modificarlo:

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

En la sección [DEFAULT] se podrán configurar los parámetros principales mencionados anteriormente.

```
[DEFAULT]

#
# MISCELLANEOUS OPTIONS
#
```

En la sección # JAILS se encontrarán los servicios sobre los que se aplicarán acciones.

```
#
# JAILS
#
```

Activar protección contra ataques del servicio **ssh**

Agregar debajo de JAILS:

```
[sshd]

enabled = true
port    = ssh
logpath = %(sshd_log)s
backend = %(sshd_backend)s
```

[sshd]
Declarar el servicio en cuestión

- enabled = true

  Activa el jail.

- port = ssh

Define el puerto que será protegido. Al colocar ssh, toma el valor definido en */etc/services* (por lo general el 22).

- logpath = %(sshd_log)s

  Define la ruta al archivo de logs que fail2ban debe analizar.
  La variable *%(sshd_log)s* es una variable que está definida en el archivo *paths-common.conf* y apunta hacia la ruta *var/log/auth.log*.

```
# Default values for several jails:

sshd_log = %(syslog_authpriv)s
```

```
syslog_authpriv = /var/log/auth.log
```

- backend = %(sshd_backend)s
  Indica el método que fail2ban usará para leer los logs.
  La variable *%(sshd_backend)s* se define en el archivo *paths-arch.conf*.

```
  GNU nano 7.2                     pa


# These services will log to the jour
# default.
syslog_backend = systemd
sshd_backend = systemd
```

El valor *systemd* indica que el sistema usará *journalctl* (el visor de logs de systemd) para leer los registros.

Habiendo agregado la configuración para el servicio ssh en *jail.local* ya se encuentra activa la protección.

Para verificarlo se puede consultar con:

```
sudo fail2ban-client status
```

```
gonzalo@gonzalo-VirtualBox:/etc/fail2ban$ sudo fail2ban-client status
Status
|- Number of jail:      1
`- Jail list:   sshd
```

4. Uso

4.1. Ejemplo de uso para SSH

Para corroborar el correcto funcionamiento y protección sobre el servicio ssh se puede ejecutar la siguiente prueba:

4.1.1. Configurar en el archivo *etc/fail2ban/jail.local* una ventana de tiempo de búsqueda de 3 minutos, como máximo 3 intentos fallidos y un tiempo de suspensión de 1 minuto.

```
  GNU nano 7.2                          jail.local

# "bantime" is the number of seconds that a host is banned.
bantime  = 1m

# A host is banned if it has generated "maxretry" during the last "findtime"
# seconds.
findtime  = 3m

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

4.1.2. Iniciar servicio ssh:

```
sudo service ssh start
```

4.1.3. Revisar que se encuentre activo:

```
sudo service ssh status
```

```
gonzalo@gonzalo-VirtualBox:~$ sudo service ssh status
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/usr/lib/systemd/system/ssh.service; disabled; preset: enabled)
     Active: active (running) since Wed 2025-07-09 22:18:40 -03; 1s ago
TriggeredBy: ● ssh.socket
       Docs: man:sshd(8)
             man:sshd_config(5)
    Process: 4542 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
   Main PID: 4543 (sshd)
      Tasks: 1 (limit: 4605)
     Memory: 2.1M (peak: 2.3M)
        CPU: 20ms
     CGroup: /system.slice/ssh.service
             └─4543 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"
```

4.1.4. Desde otro equipo realizar 3 intentos de conexión ssh en menos de 3 minutos, colocando la contraseña incorrecta.

```
ssh hostName@direcionIP
```

4.1.5. Visualizar las últimas líneas de logs de fail2ban:

```
tail -f /var/log/fail2ban.log
```

Se revisan las últimas lineas siguiendo “en vivo” (-f de follow) la aparición de nueva información en el archivo.

```
2025-07-09 22:11:59,287 fail2ban.jail           [4498]: INFO    Jail 'sshd' started
2025-07-09 22:24:48,999 fail2ban.filter         [4498]: INFO    [sshd] Found 192.168.1.42 - 2025-07-09
22:24:48
2025-07-09 22:25:04,450 fail2ban.filter         [4498]: INFO    [sshd] Found 192.168.1.42 - 2025-07-09
22:25:04
2025-07-09 22:25:18,713 fail2ban.filter         [4498]: INFO    [sshd] Found 192.168.1.42 - 2025-07-09
22:25:18
2025-07-09 22:25:19,015 fail2ban.actions        [4498]: NOTICE  [sshd] Ban 192.168.1.42
```

Se observan intentos de conexión ssh desde la dirección IP 192.168.1.42 con fecha y horario. Al llegar al tercer intento fallido, se ejecuta la acción de protección suspendiendo la IP mencionada por 1 minuto.

4.1.6. Al cumplirse 1 minuto de suspensión, la dirección IP 192.168.1.42 se desbanea automáticamente y puede volver a intentar realizar conexiones ssh.

```
2025-07-09 22:25:19,015 fail2ban.actions        [4498]: NOTICE  [sshd] Ban 192.168.1.42
2025-07-09 22:26:18,090 fail2ban.actions        [4498]: NOTICE  [sshd] Unban 192.168.1.42
```

4.2. Ejemplo de uso para FTP

Para corroborar el correcto funcionamiento y protección sobre el servicio ftp se puede ejecutar la siguiente prueba:

4.2.1. Modificar la configuración agregando al archivo *jail.local* los parámetros necesarios para incluir el servicio ftp:

```
[vsftpd]
enabled = true
port    = ftp,ftp-data,ftps,ftps-data
logpath = %(vsftpd_log)s
```

4.2.2. Reiniciar el servicio fail2ban con el comando:

```
service fail2ban restart
```

4.2.3. Iniciar el servicio ftp si es que no está en ejecución, esto se realiza con el comando:

```
service vsftpd start
```

4.2.4. Ejecutar el comando ftp e intentar conectarse al equipo ingresando mal usuario y/o contraseña.

4.2.5. Luego del tercer intento, al consultar el fail2ban.log se verá lo siguiente:

```
vm-labo@vm-labo-VirtualBox:/etc/fail2ban$ tail -f /var/log/fail2ban.log
2025-07-10 19:30:29,194 fail2ban.filter         [6092]: INFO      findtime: 60
2025-07-10 19:30:29,194 fail2ban.actions        [6092]: INFO      banTime: 180
2025-07-10 19:30:29,194 fail2ban.filter         [6092]: INFO      encoding: UTF-8
2025-07-10 19:30:29,194 fail2ban.filter         [6092]: INFO    Added logfile: '/var/log/vsftpd.log' (pos = 771,
6e724333)
2025-07-10 19:30:29,200 fail2ban.jail           [6092]: INFO    Jail 'sshd' started
2025-07-10 19:30:29,201 fail2ban.jail           [6092]: INFO    Jail 'vsftpd' started
2025-07-10 19:30:59,664 fail2ban.filter         [6092]: INFO    [vsftpd] Found 127.0.0.1 - 2025-07-10 19:30:59
2025-07-10 19:31:22,907 fail2ban.filter         [6092]: INFO    [vsftpd] Found 127.0.0.1 - 2025-07-10 19:31:22
2025-07-10 19:31:38,125 fail2ban.filter         [6092]: INFO    [vsftpd] Found 127.0.0.1 - 2025-07-10 19:31:38
2025-07-10 19:31:38,841 fail2ban.actions        [6092]: NOTICE  [vsftpd] Ban 127.0.0.1
2025-07-10 19:34:38,594 fail2ban.actions        [6092]: NOTICE  [vsftpd] Unban 127.0.0.1
```

Ya no podrá conectarse por ftp desde la ip local, mostrando la siguiente pantalla al volver a intentarlo:

```
vm-labo@vm-labo-VirtualBox:/etc/fail2ban$ ftp localhost
ftp: Can't connect to `127.0.0.1:21': Descriptor de archivo erróneo
ftp: Can't connect to `localhost:ftp'
ftp>
```

4.3. Ejemplo de uso para HTTP

Para corroborar el correcto funcionamiento y protección sobre el servicio http se puede ejecutar la siguiente prueba:

4.3.1. Instalar apache, para iniciar un servidor web:

```
sudo apt install apache2
```

4.3.2. Instalar SSL, para permitir el cifrado de comunicación entre el servidor web y el cliente:

```
sudo apt install openssl
```

4.3.3. Configurar certificado SSL autofirmado para apache, a través de la guía en la siguiente web:

[How To Create a Self-Signed SSL Certificate for Apache in Ubuntu 18.04 | DigitalOcean]

4.3.4. Configurar autenticación de contraseña con Apache, mediante la guía en la siguiente web:

[Cómo configurar la autenticación de contraseña con Apache enUbuntu 18.04 | DigitalOcean]

4.3.5. Iniciar servidor web:

```
sudo service apache2 start
```

4.3.5. Modificar la configuración agregando al archivo *jail.local* los parámetros necesarios para incluir el servicio http a través del servidor apache:

```
#
# HTTP servers
#

[apache-auth]
enabled  = true
port     = http,https
logpath  = %(apache_error_log)s
```

En archivo *paths-common.conf*

```
apache_error_log = /var/log/apache2/*error.log
```

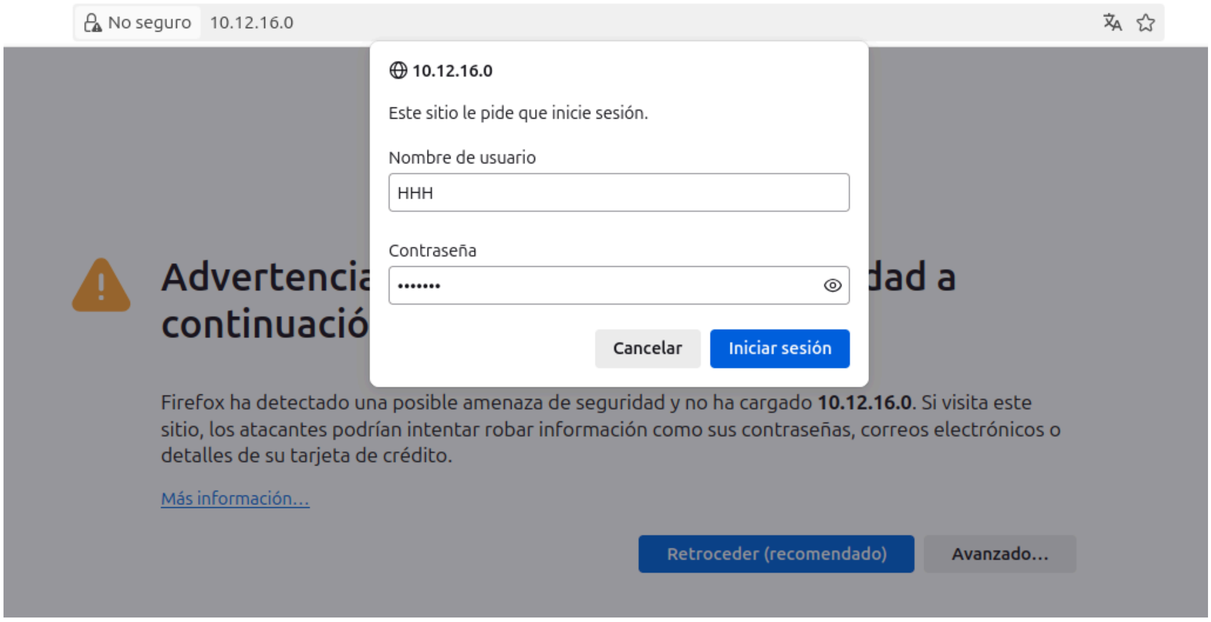
4.3.6. Reiniciar el servicio fail2ban para actualizar la configuración:

```
sudo service fail2ban restart
```

4.3.7. Desde un equipo conectado a la misma red, intentar ingresar a la web mediante un navegador. En este caso, nuestra dirección IP es 10.12.16.0. Colocar en la barra de dirección:

https://10.12.16.0

4.3.8. El sitio solicitará autenticación con nombre de usuario y contraseña. Realizar 3 intentos de autenticación incorrectos dentro de un plazo menor a 3 minutos:



4.3.9. Revisar el archivo de logs de fail2ban:

```
tail -f /var/log/fail2ban/fail2ban.log
```

```
2025-07-10 20:56:32,220 fail2ban.filter         [7987]: INFO    [apache-auth] Found 10.12.3.255 - 2025-07-10 20:56:32
2025-07-10 20:56:38,486 fail2ban.filter         [7987]: INFO    [apache-auth] Found 10.12.3.255 - 2025-07-10 20:56:38
2025-07-10 20:56:41,716 fail2ban.filter         [7987]: INFO    [apache-auth] Found 10.12.3.255 - 2025-07-10 20:56:41
2025-07-10 20:56:41,830 fail2ban.actions        [7987]: NOTICE  [apache-auth] Ban 10.12.3.255
```

Se observan intentos de autenticación desde la dirección IP 10.12.3.255 con fecha y horario. Al llegar al tercer intento fallido, se ejecuta la acción de protección suspendiendo la conexión de la IP mencionada por 1 minuto.

4.3.10. Al intentar acceder a la web desde la dirección IP suspendida, la conexión falla:

4.4. Comandos Adicionales

Reiniciar el servicio Fail2Ban:

```
sudo systemctl restart fail2ban
```

Se utiliza luego de realizar cambios en la configuración. (Luego de modificar el jail.local)

Desbloquear manualmente una IP:

```
sudo fail2ban-client set sshd unbanip
192.168.1.50
```

Quita el bloqueo a una IP previamente baneada.

Banear manualmente una IP:

```
sudo fail2ban-client set sshd banip 192.168.1.50
```

Fuerza el bloqueo de una IP incluso si no superó el umbral de intentos

Ver detalles de una jail específica (por ejemplo, sshd):

```
sudo fail2ban-client status sshd
```

Muestra la cantidad de IPs actualmente bloqueadas, los intentos fallidos y la lista de IPs baneadas.

```
Status for the jail: sshd
|- Filter
|  |- Currently failed: 0
|  |- Total failed:     6
|  `- File list:        /var/log/auth.log
`- Actions
   |- Currently banned: 2
   |- Total banned:     2
   `- Banned IP list:   192.168.1.42 127.0.0.1
```

Ver estado general del servicio Fail2Ban:

```
sudo fail2ban-client status
```

Muestra todas las "jails" activas y su estado actual.

```
Status
|- Number of jail:      3
`- Jail list:   apache-auth, sshd, vsftpd
```

5. Conclusiones

Fail2Ban es una herramienta eficaz y relativamente sencilla de implementar para mejorar la seguridad en sistemas Linux. Su principal fortaleza radica en la automatización del bloqueo de IPs maliciosas en base a patrones de comportamiento detectados en archivos de log.

Durante este trabajo práctico pudimos comprobar cómo Fail2Ban se adapta a distintos servicios como SSH, FTP o HTTP, y cómo responde de forma automática a intentos de fuerza bruta, disminuyendo el riesgo de accesos no autorizados.

El uso de archivos de configuración separados (jail.local) facilita una administración más segura y ordenada de los parámetros. Además, los comandos disponibles permiten tanto una administración automatizada como intervenciones manuales en caso de ser necesario.

En resumen, Fail2Ban representa una primera línea de defensa muy valiosa para cualquier servidor expuesto a la red, siendo un complemento ideal para firewalls y otras herramientas de monitoreo y seguridad.

6. Referencias

6.1. Sitio oficial de Fail2Ban: https://www.fail2ban.org

6.2. Fail2Ban en GitHub https://github.com/fail2ban/fail2ban

Repositorio oficial con código fuente, reportes de errores y aportes de la comunidad.

6.3. Guía práctica de Fail2Ban en Linux Adictos https://www.linuxadictos.com/que-es-fail2ban-y-como-funciona.html

6.4. Ubuntu Manpages – fail2ban-client(1) https://manpages.ubuntu.com/manpages/focal/en/man1/fail2ban-client.1.html Manual completo del cliente de línea de comandos fail2ban-client.