

Después de revisar un poco lo que hace [DenyHosts](#) , vamos a ver la forma de configurar e instalar [InjectionDenied](#) , que es un sencillo programa de Jordi Blasco Pallarés, que hace algo muy similar a lo que hace DenyHosts con el demonio SSH, pero en este caso bloquea las IPs desde las que se intentan hacer inyecciones PHP o SQL a un servidor web. Este programa está basado en bash/gawk, por lo que es muy sencillo de entender y modificar aunque, como es lógico, tenemos que tener instalado gawk en nuestro sistema si queremos que InjectionDenied funcione adecuadamente...

INSTALACIÓN Y CONFIGURACIÓN DEL PROGRAMA

Para instalar el programa, basta con bajarse la última versión de Sourceforge, la 0.0.1:

```
wget http://downloads.sourceforge.net/injectiondenied/injectiondenied-0.01.tar
```

Una vez que tengamos el programa, será necesario extraer los archivos:

```
tar -xf injectiondenied-0.01.tar
```

El programa consta de tres archivos, que deben estar en las trayectorias que se indican:

- /etc/init.d/injectdenied (que debe ser ejecutable)
- /etc/injectiondenied/rules (que contiene las reglas)
- /usr/local/bin/hackreport.sh (que tiene que ser ejecutable).

InjectionDenied: detección y bloqueo de ataques PHP y SQL

Escrito por Fernando Acero - Última actualización 20 de Septiembre de 2008

Yo soy usuario de Mandriva y este programa está pensado para Debian, por lo que usa ciertos recursos específicos para Debian, como start-stop-daemon. La solución es instalar el paquete dpkg que contiene este recurso; para ello, usaremos el comando siguiente:

```
urpmi dpkg
```

El programa tal como está no funciona; es necesario configurar algunas cosas antes de arrancarlo por primera vez. Primero editaremos /usr/local/bin/hackreport.sh y cambiaremos las líneas siguientes, para que apunten a nuestro archivo de logs de acceso al Apache, en mi caso /var/log/httpd/access_log:

```
gawk -v RULE="$i" '{if(match ($7,RULE)!=0){print $1}}' /var/log/httpd/access_log >>
/tmp/banned_ip.dat
gawk -v RULE="$i" '{if(match ($7,RULE)!=0){print $1 "t " $6 " " $7}}' /var/log/httpd/access_log
>> /tmp/hack.log
```

También modificaremos las líneas siguientes, para que los correos con las alertas lleguen a la cuenta de correo que deseemos:

```
mail -s "[INJECT] Posibles intents de Hack" root@server.com < /tmp/hack.log
mail -s "[INJECT] Entradas al hosts.deny" root@server.com < /etc/hosts.deny
```

Finalmente hay que modificar /etc/init.d/injectdenied para que permita la opción restart, elimine el proceso y para que muestre información sobre el PID que se ha arrancado. El código de pruebas que estoy usando es éste; es muy parecido al original, pero no me convence demasiado y es posible que lo cambie para que use el mandato "daemon" en lugar de start-stop-daemon, por ser más cercano al Linux Standard Base:

```
start)
# Modificado por Fernando Acero 16SEP08
echo -n "Iniciando $DESC: $NAME"
start-stop-daemon -b --start --quiet --pidfile /var/run/injectiondenied.pid --exec $DAEMON
```

InjectionDenied: detección y bloqueo de ataques PHP y SQL

Escrito por Fernando Acero - Última actualización 20 de Septiembre de 2008

```
ps -fea | grep hackreport|head -1| gawk '{print $2}' > /var/run/injectiondenied.pid
# Modificado por Fernando Acero 16SEP08 (cambiado texto)
echo "ARRANCADO"
;;

stop)
# Modificado por Fernando Acero 16SEP08 (cambiado texto)
echo -n "Deteniendo $DESC: injectiondenied"
start-stop-daemon --stop --quiet --pidfile /var/run/injectiondenied.pid
# Modificado por Fernando Acero 16SEP08 (activado kill)
kill $PID
# Modificado por Fernando Acero 16SEP08 (cambiado texto)
echo "DETENIDO"
;;
# Bloque restart modificado por Fernando Acero 16SEP08 (añadido bloque completo)
restart)

echo -n "Deteniendo $DESC: injectiondenied"
start-stop-daemon --stop --quiet --pidfile /var/run/injectiondenied.pid
kill $PID
echo "DETENIDO"

echo -n "Iniciando $DESC: $NAME"
start-stop-daemon -b --start --quiet --pidfile /var/run/injectiondenied.pid --exec $DAEMON
ps -fea | grep hackreport|head -1| gawk '{print $2}' > /var/run/injectiondenied.pid
echo "ARRANCADO $PID"
;;

*)
```

Como se puede ver, el mandato restart es simplemente la copia de los mandatos stop y start sin la finalización de stop ";" para que siga con la ejecución del arranque tras parar de forma efectiva el proceso.

Como no estoy seguro de su estabilidad, lo he metido en cron.hourly para que se reinicie cada hora. Por el momento tengo este programa en pruebas, por lo que poco puedo decir de su

InjectionDenied: detección y bloqueo de ataques PHP y SQL

Escrito por Fernando Acero - Última actualización 20 de Septiembre de 2008

funcionamiento en este momento, aunque también es cierto que en algunas cosas me parece algo incompleto, por ejemplo, no tenía el código para restart. De todos modos, también estoy usando [Snort](#) con [psad](#) y los módulos [apache ModSecurity](#) y [suhosin](#), entre otros en mi servidor, todos ellos más que recomendables, por lo que no creo que su funcionamiento sea demasiado crítico para mejorar la seguridad de mi servidor, pero por probar no se pierde nada.

"Copyleft 2008 Fernando Acero Martín. Verbatim copying, translation and distribution of this entire article is permitted in any digital medium, provided this notice is preserved."