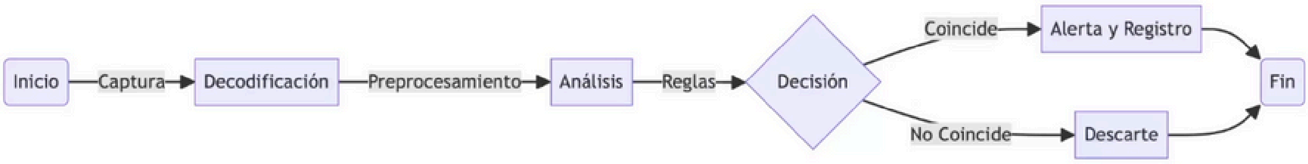
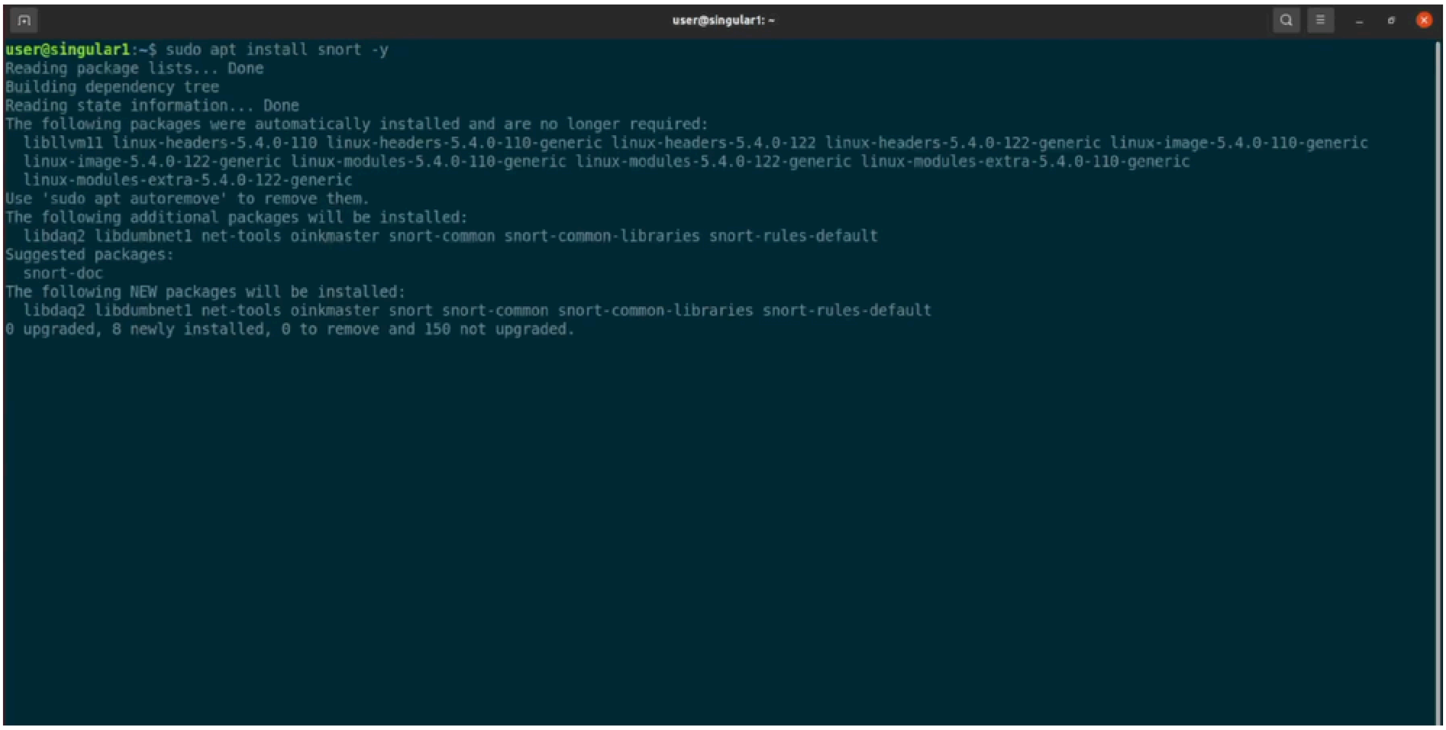
Snort

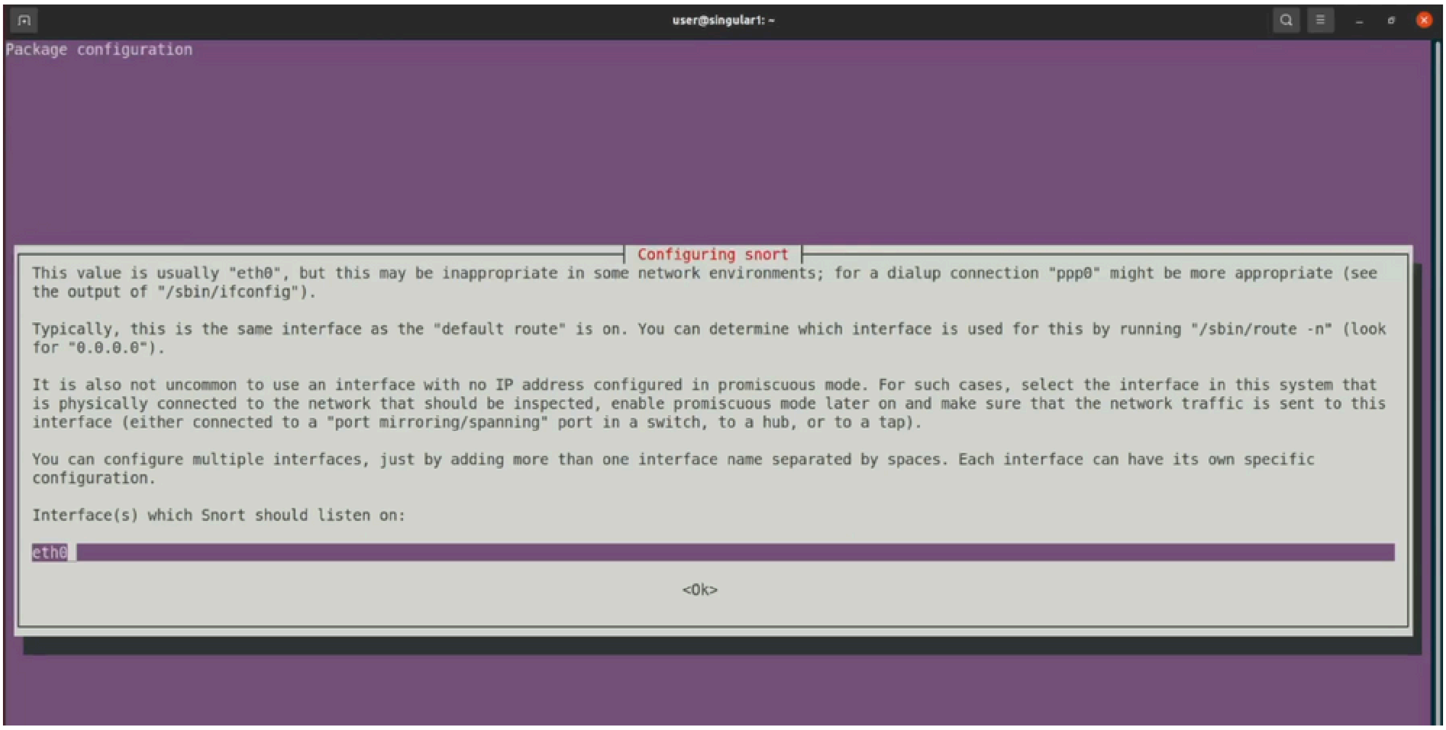
Snort is an open-source intrusion detection and prevention system that provides real-time traffic analysis and packet logging to detect and respond to network threats efficiently.



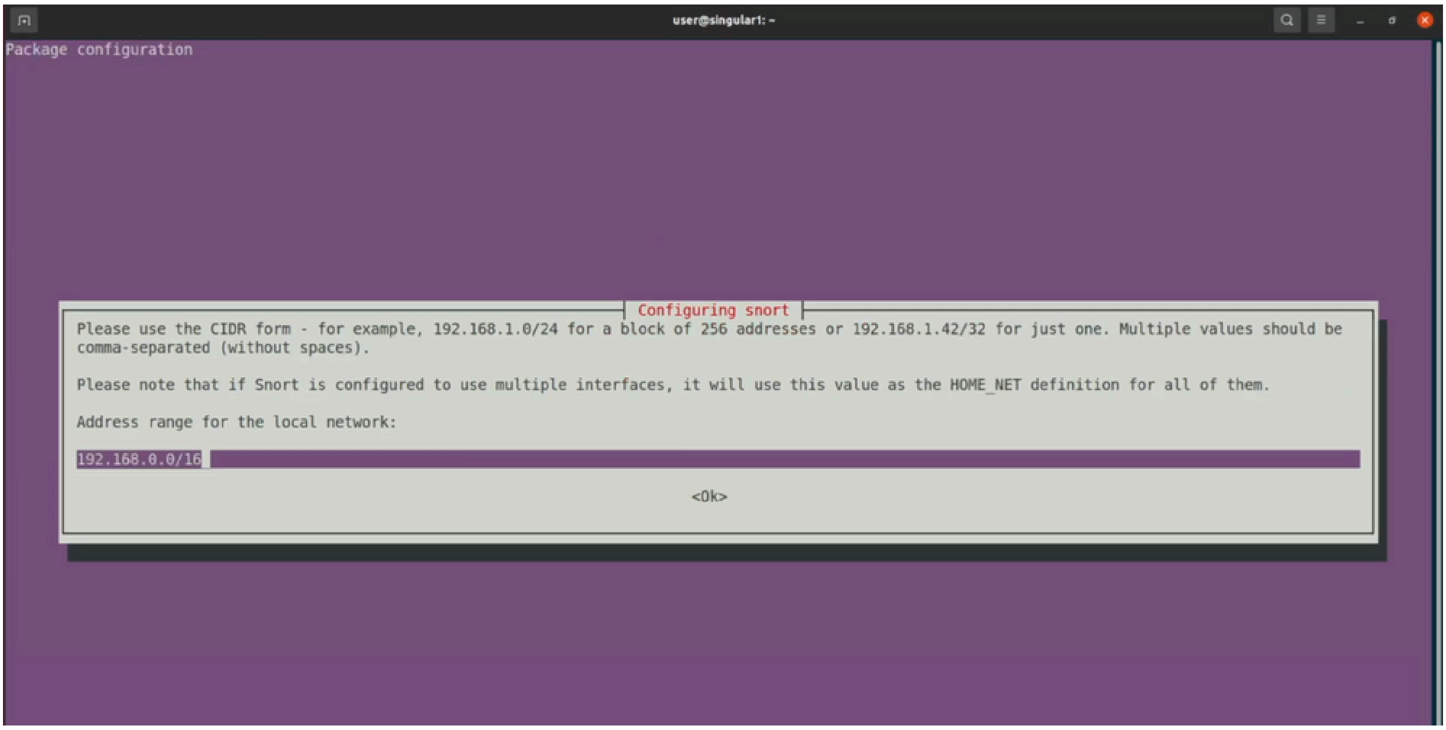
Picure source: own creation



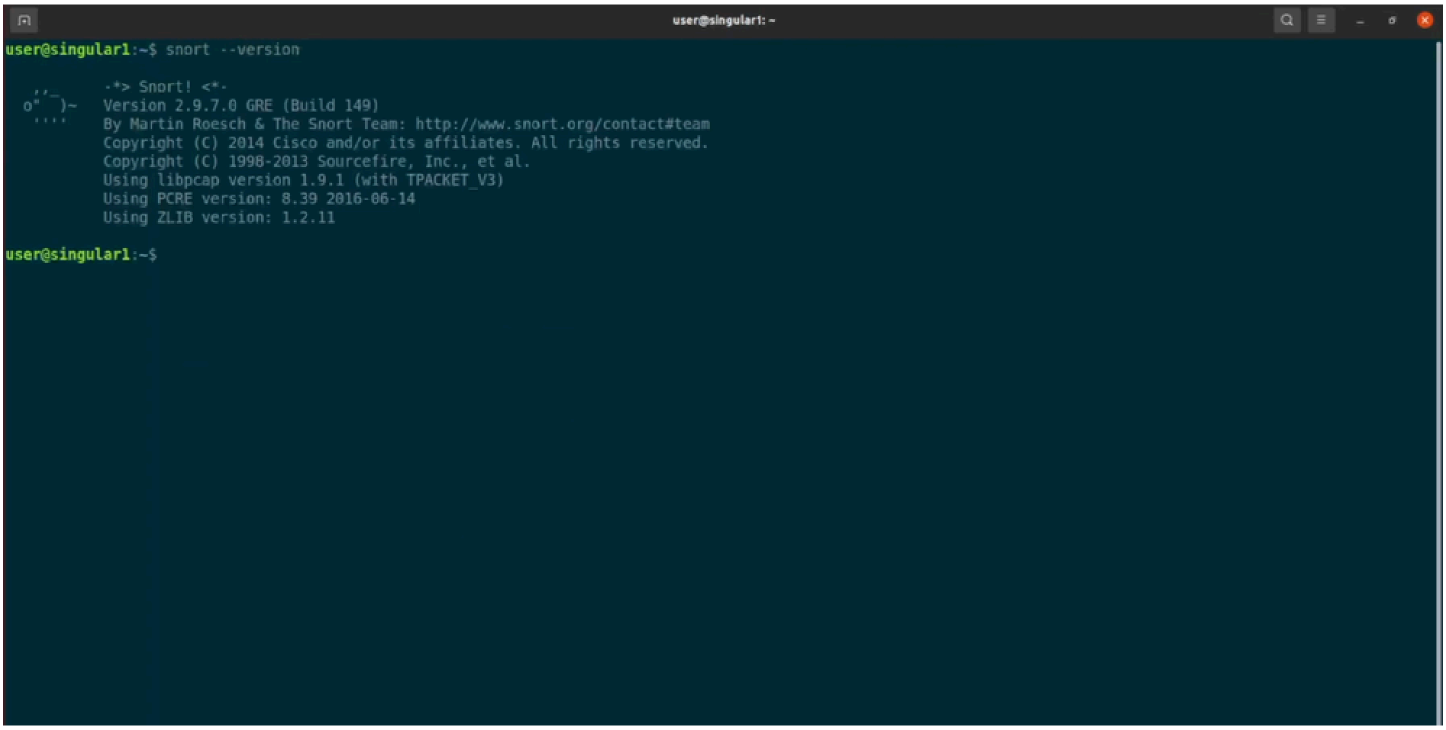
```
user@singular1:~$ sudo apt install snort -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libllvm11 linux-headers-5.4.0-110 linux-headers-5.4.0-110-generic linux-headers-5.4.0-122 linux-headers-5.4.0-122-generic linux-image-5.4.0-110-generic
  linux-image-5.4.0-122-generic linux-modules-5.4.0-110-generic linux-modules-5.4.0-122-generic linux-modules-extra-5.4.0-110-generic
  linux-modules-extra-5.4.0-122-generic
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  libdaq2 libdumbnet1 net-tools oinkmaster snort-common snort-common-libraries snort-rules-default
Suggested packages:
  snort-doc
The following NEW packages will be installed:
  libdaq2 libdumbnet1 net-tools oinkmaster snort snort-common snort-common-libraries snort-rules-default
0 upgraded, 8 newly installed, 0 to remove and 150 not upgraded.
```



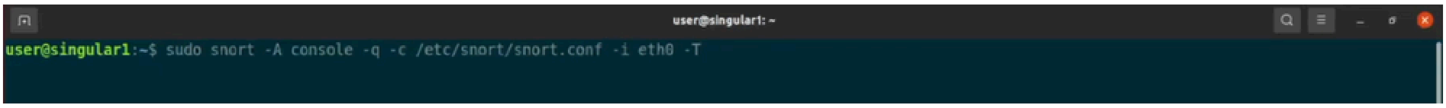
```
Package configuration

Configuring snort

This value is usually "eth0", but this may be inappropriate in some network environments; for a dialup connection "ppp0" might be more appropriate (see
the output of "/sbin/ifconfig").

Typically, this is the same interface as the "default route" is on. You can determine which interface is used for this by running "/sbin/route -n" (look
for "0.0.0.0").

It is also not uncommon to use an interface with no IP address configured in promiscuous mode. For such cases, select the interface in this system that
is physically connected to the network that should be inspected, enable promiscuous mode later on and make sure that the network traffic is sent to this
interface (either connected to a "port mirroring/spanning" port in a switch, to a hub, or to a tap).

You can configure multiple interfaces, just by adding more than one interface name separated by spaces. Each interface can have its own specific
configuration.

Interface(s) which Snort should listen on:

eth0

<Ok>
```



```
Package configuration

Configuring snort

Please use the CIDR form - for example, 192.168.1.0/24 for a block of 256 addresses or 192.168.1.42/32 for just one. Multiple values should be
comma-separated (without spaces).

Please note that if Snort is configured to use multiple interfaces, it will use this value as the HOME_NET definition for all of them.

Address range for the local network:

192.168.0.0/16

<Ok>
```

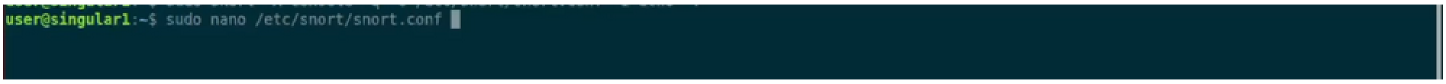


```
user@singular1:~$ snort --version

   ,,_     -*> Snort! <*-
  o"  )~   Version 2.9.7.0 GRE (Build 149)
   ''''    By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
           Copyright (C) 2014 Cisco and/or its affiliates. All rights reserved.
           Copyright (C) 1998-2013 Sourcefire, Inc., et al.
           Using libpcap version 1.9.1 (with TPACKET_V3)
           Using PCRE version: 8.39 2016-06-14
           Using ZLIB version: 1.2.11

user@singular1:~$
```

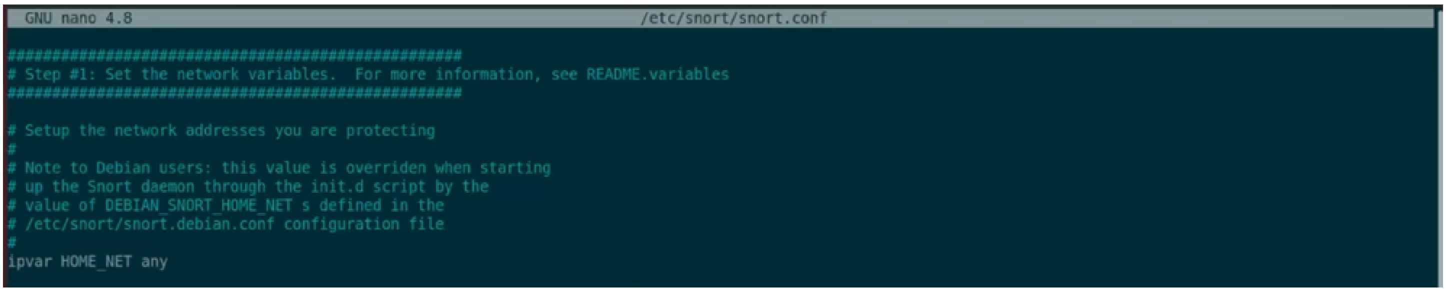


```
user@singular1:~$ sudo snort -A console -q -c /etc/snort/snort.conf -i eth0 -T
```

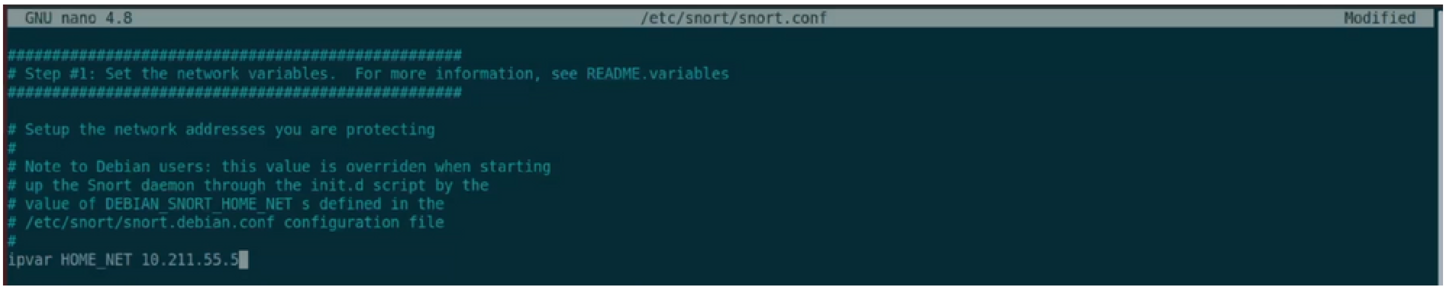
Entramos en el fichero de configuración para especificar que host, ya quer de momento queremos usar snort como IDS, y tenemos que delimitarle un host en concreto, no dejarlo en "any", en éste caso y como ejemplo le ponemos la ip de la maquina que queremos que sea el host:



```
user@singular1:~$ sudo nano /etc/snort/snort.conf
```

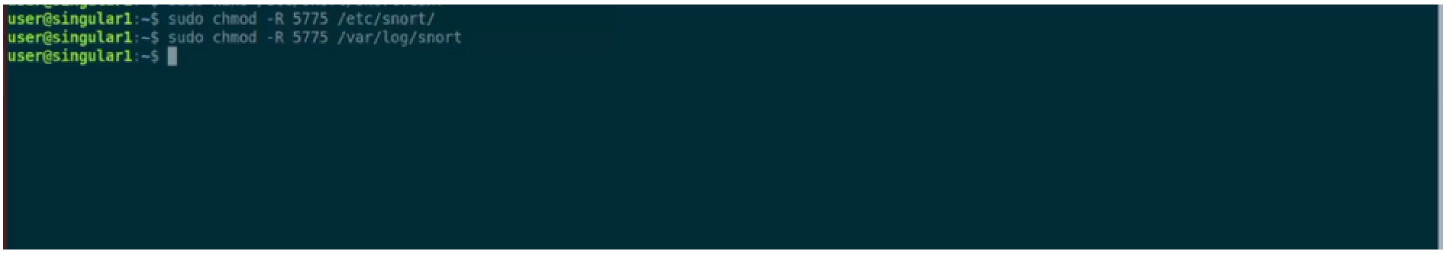


```
GNU nano 4.8                         /etc/snort/snort.conf

###################################################
# Step #1: Set the network variables.  For more information, see README.variables
###################################################

# Setup the network addresses you are protecting
#
# Note to Debian users: this value is overriden when starting
# up the Snort daemon through the init.d script by the
# value of DEBIAN_SNORT_HOME_NET s defined in the
# /etc/snort/snort.debian.conf configuration file
#
ipvar HOME_NET any
```



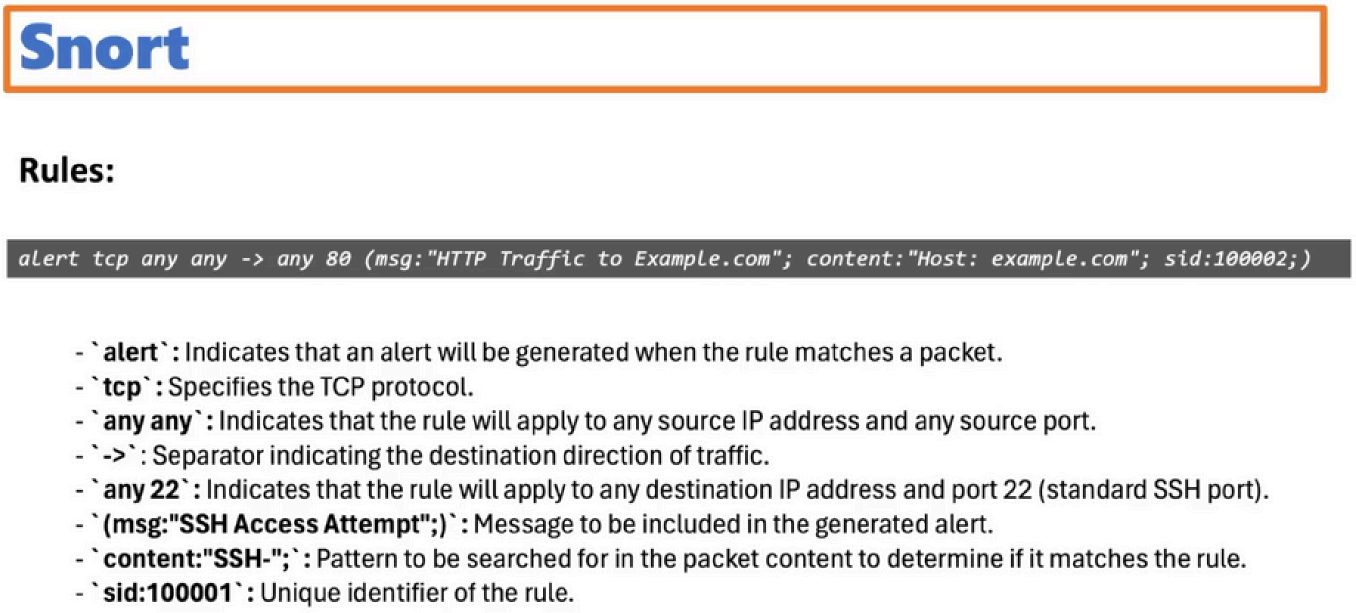
```
GNU nano 4.8                         /etc/snort/snort.conf                                 Modified

###################################################
# Step #1: Set the network variables.  For more information, see README.variables
###################################################

# Setup the network addresses you are protecting
#
# Note to Debian users: this value is overriden when starting
# up the Snort daemon through the init.d script by the
# value of DEBIAN_SNORT_HOME_NET s defined in the
# /etc/snort/snort.debian.conf configuration file
#
ipvar HOME_NET 10.211.55.5
```

Modificamos los permisos de dos carpetas. Razón: asegurarnos que snort tiene los permisos adecuados para acceder, leer y escribir en éstas dos carpetas (directorio de configuración y el directorio de los logs), usamos el comando chmod para cambiar los permisos:



```
user@singular1:~$ sudo chmod -R 5775 /etc/snort/
user@singular1:~$ sudo chmod -R 5775 /var/log/snort
user@singular1:~$ 
```

Ahora vamos a definir las reglas en Snort, aqui tenemos un preview:



Snort

Rules:

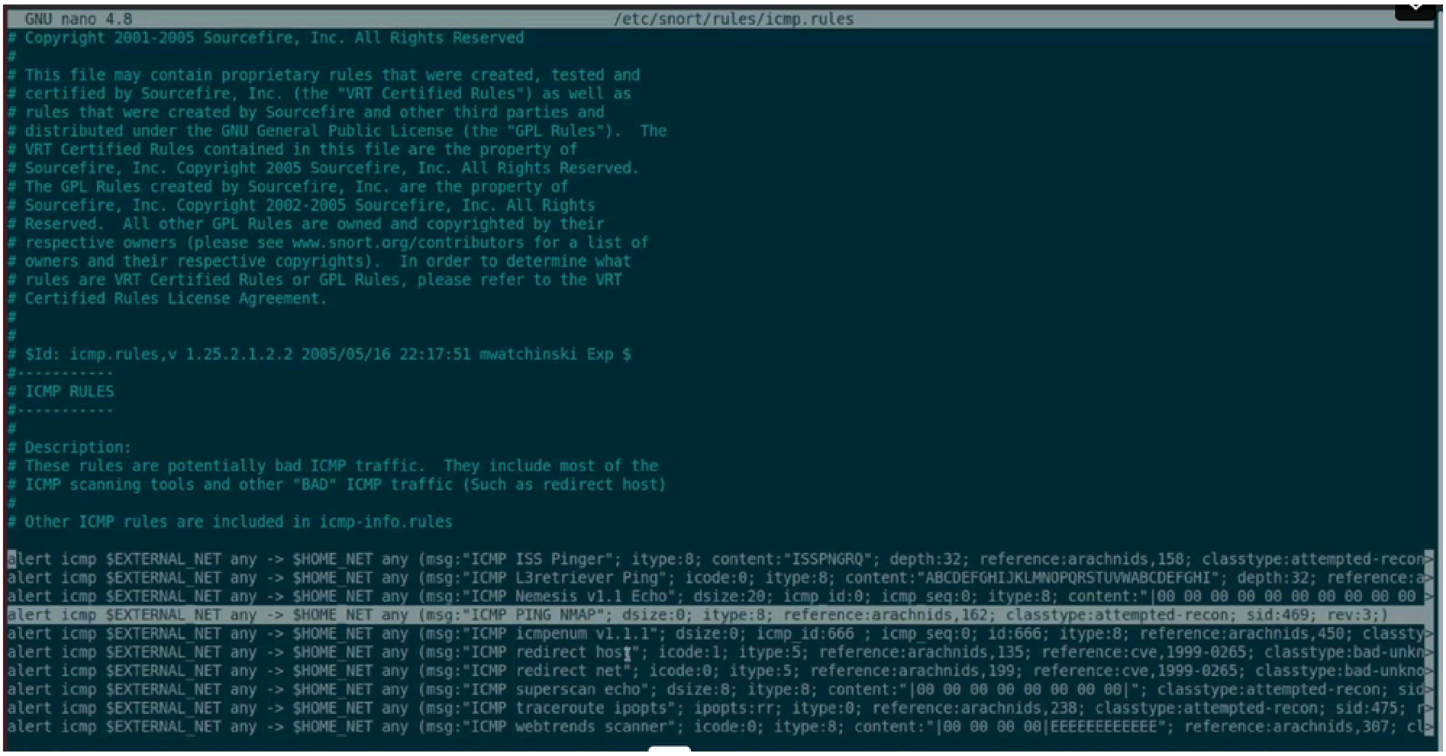
```
alert tcp any any -> any 80 (msg:"HTTP Traffic to Example.com"; content:"Host: example.com"; sid:100002;)
```

- `**alert**`: Indicates that an alert will be generated when the rule matches a packet.
- `**tcp**`: Specifies the TCP protocol.
- `**any any**`: Indicates that the rule will apply to any source IP address and any source port.
- `**->**`: Separator indicating the destination direction of traffic.
- `**any 22**`: Indicates that the rule will apply to any destination IP address and port 22 (standard SSH port).
- `**(msg:"SSH Access Attempt";)**`: Message to be included in the generated alert.
- `**content:"SSH-";**`: Pattern to be searched for in the packet content to determine if it matches the rule.
- `**sid:100001**`: Unique identifier of the rule.

Reglas de protocolo ICMP:

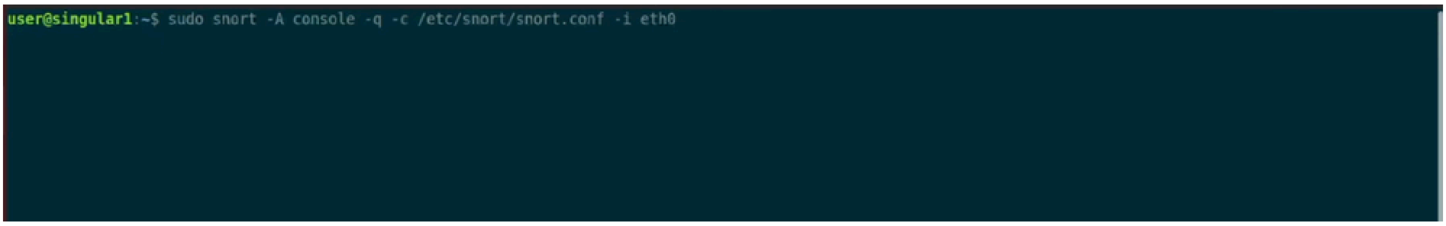
```
user@singular1:~$ sudo nano /etc/snort/rules/icmp.rules 
```

Cogemos una regla de ejemplo y vamos a analizarla (en el transcript está todo), dame información adicional).

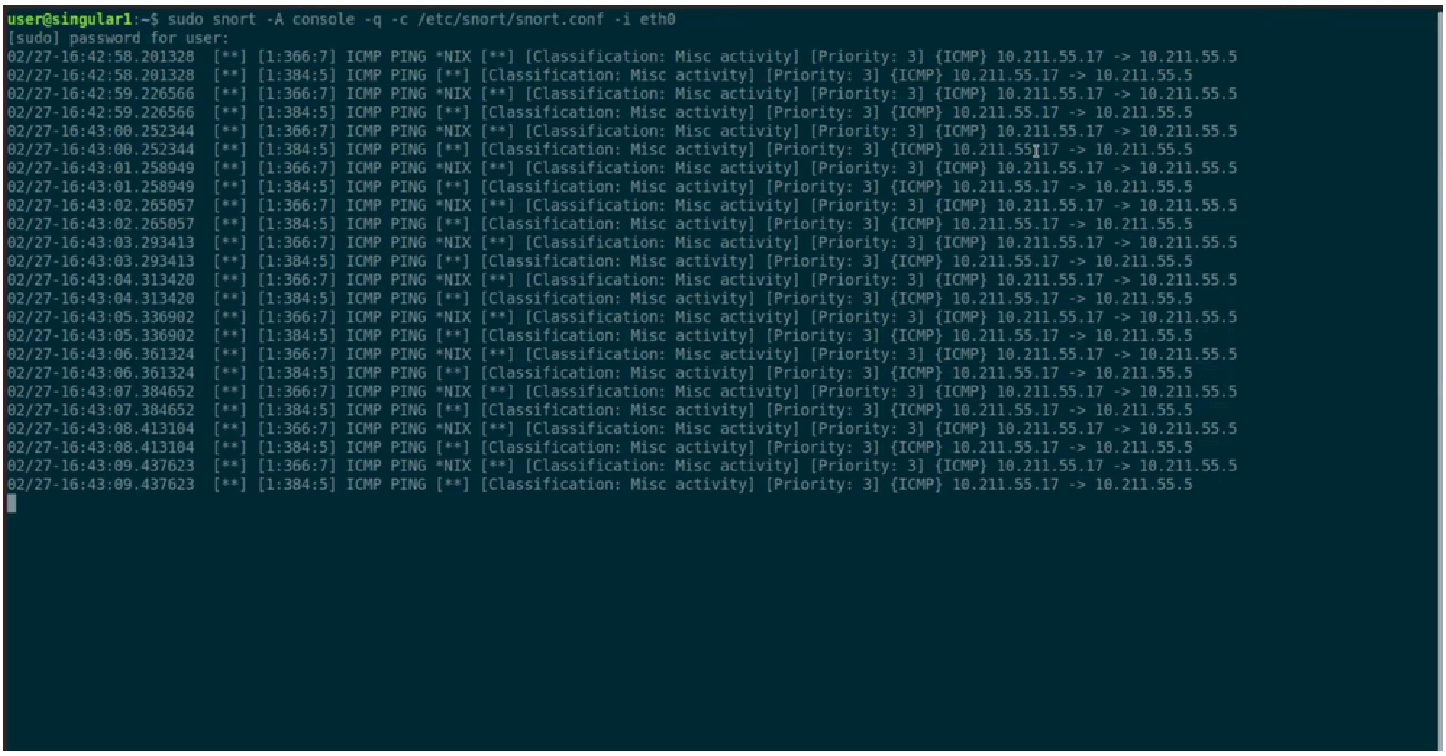


```
  GNU nano 4.8                                                          /etc/snort/rules/icmp.rules
# Copyright 2001-2005 Sourcefire, Inc. All Rights Reserved
#
# This file may contain proprietary rules that were created, tested and
# certified by Sourcefire, Inc. (the "VRT Certified Rules") as well as
# rules that were created by Sourcefire and other third parties and
# distributed under the GNU General Public License (the "GPL Rules").  The
# VRT Certified Rules contained in this file are the property of
# Sourcefire, Inc. Copyright 2005 Sourcefire, Inc. All Rights Reserved.
# The GPL Rules created by Sourcefire, Inc. are the property of
# Sourcefire, Inc. Copyright 2002-2005 Sourcefire, Inc. All Rights
# Reserved.  All other GPL Rules are owned and copyrighted by their
# respective owners (please see www.snort.org/contributors for a list of
# owners and their respective copyrights).  In order to determine what
# rules are VRT Certified Rules or GPL Rules, please refer to the VRT
# Certified Rules License Agreement.
#
#
# $Id: icmp.rules,v 1.25.2.1.2.2 2005/05/16 22:17:51 mwatchinski Exp $
#-----------
# ICMP RULES
#-----------
#
# Description:
# These rules are potentially bad ICMP traffic.  They include most of the
# ICMP scanning tools and other "BAD" ICMP traffic (Such as redirect host)
#
# Other ICMP rules are included in icmp-info.rules

alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"ICMP ISS Pinger"; itype:8; content:"ISSPNGRQ"; depth:32; reference:arachnids,158; classtype:attempted-recon>
alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"ICMP L3retriever Ping"; icode:0; itype:8; content:"ABCDEFGHIJKLMNOPQRSTUVWABCDEFGHI"; depth:32; reference:a>
alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"ICMP Nemesis v1.1 Echo"; dsize:20; icmp_id:0; icmp_seq:0; itype:8; content:"|00 00 00 00 00 00 00 00 00 00 >
alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"ICMP PING NMAP"; dsize:0; itype:8; reference:arachnids,162; classtype:attempted-recon; sid:469; rev:3;)
alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"ICMP icmpenum v1.1.1"; dsize:0; icmp_id:666 ; icmp_seq:0; id:666; itype:8; reference:arachnids,450; classty>
alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"ICMP redirect host"; icode:1; itype:5; reference:arachnids,135; reference:cve,1999-0265; classtype:bad-unkn>
alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"ICMP redirect net"; icode:0; itype:5; reference:arachnids,199; reference:cve,1999-0265; classtype:bad-unkno>
alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"ICMP superscan echo"; dsize:8; itype:8; content:"|00 00 00 00 00 00 00 00|"; classtype:attempted-recon; sid>
alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"ICMP traceroute ipopts"; ipopts:rr; itype:0; reference:arachnids,238; classtype:attempted-recon; sid:475; r>
alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"ICMP webtrends scanner"; icode:0; itype:8; content:"|00 00 00 00|EEEEEEEEEEEE"; reference:arachnids,307; cl>
```

Volvemos al servidor y activamos snort con una simple regla, tenemos la otra máquina haciendo ping a ésta así que snort me lo detecta (eth0 es la interfaz del ejemplo), actuando asi cómo una IDE:



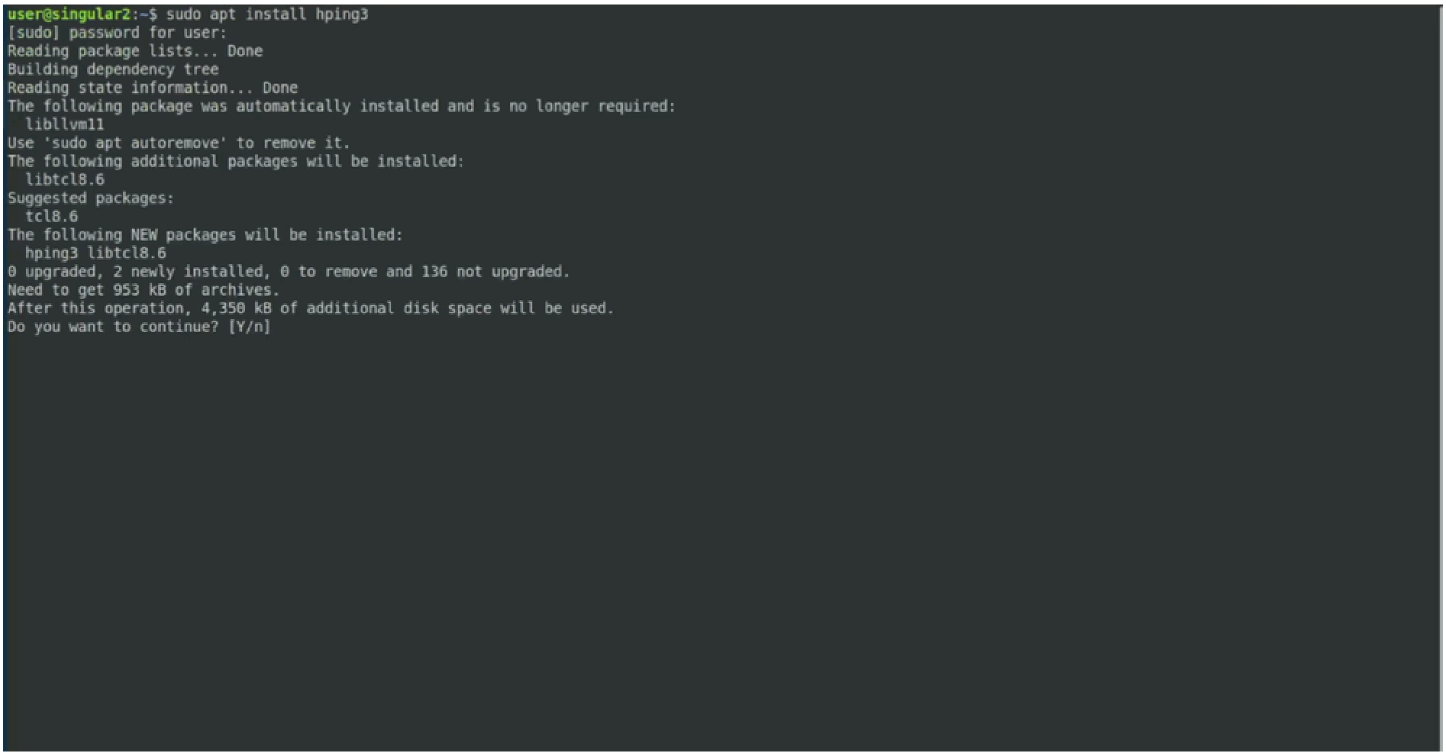
```
user@singular1:~$ sudo snort -A console -q -c /etc/snort/snort.conf -i eth0
```



```
user@singular1:~$ sudo snort -A console -q -c /etc/snort/snort.conf -i eth0
[sudo] password for user:
02/27-16:42:58.201328  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:42:58.201328  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:42:59.226566  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:42:59.226566  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:43:00.252344  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:43:00.252344  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:43:01.258949  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:43:01.258949  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:43:02.265057  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:43:02.265057  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:43:03.293413  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:43:03.293413  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:43:04.313420  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:43:04.313420  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:43:05.336902  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:43:05.336902  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:43:06.361324  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:43:06.361324  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:43:07.384652  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:43:07.384652  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:43:08.413104  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:43:08.413104  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:43:09.437623  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:43:09.437623  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
```

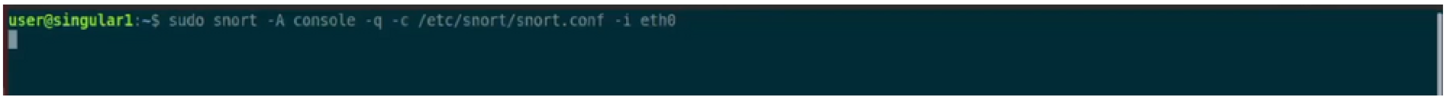
Ahora vamos a simular un ataque de denegación de servicio, realizar muchas peticiones al servidor para ver si lo aguanta, usaremos hping3 para inundar al servidor de peticiones.

En la máquina atacante instalamos hping3:



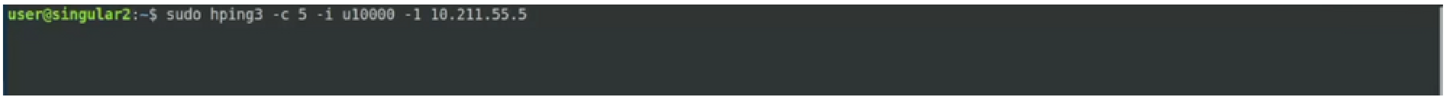
```
user@singular2:~$ sudo apt install hping3
[sudo] password for user:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following package was automatically installed and is no longer required:
  libllvm11
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
  libtcl8.6
Suggested packages:
  tcl8.6
The following NEW packages will be installed:
  hping3 libtcl8.6
0 upgraded, 2 newly installed, 0 to remove and 136 not upgraded.
Need to get 953 kB of archives.
After this operation, 4,350 kB of additional disk space will be used.
Do you want to continue? [Y/n]
```

Levantamos snort en la maquina de defensa para ver cómo reacciona ante varias herramientas de ataque, sin interrumpirlo:



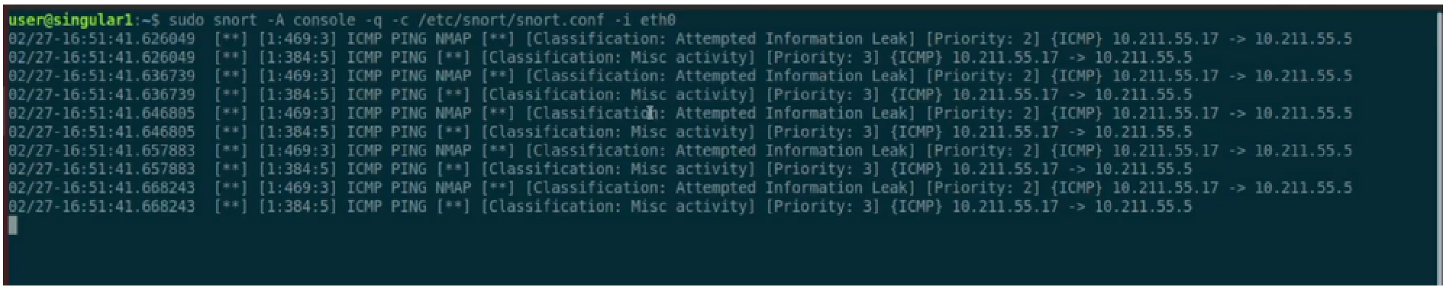
```
user@singular1:~$ sudo snort -A console -q -c /etc/snort/snort.conf -i eth0
```

Volvemos a la maquina de ataque y lanzamos con hping3 el ataque DOS:



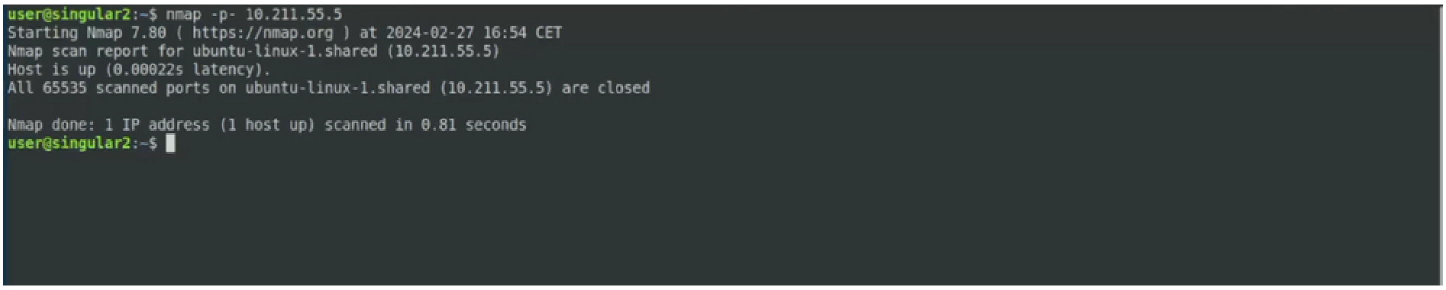
```
user@singular2:~$ sudo hping3 -c 5 -i u10000 -1 10.211.55.5
```

Vemos que en la maquina de defensa snort lo ha detectado perfectamente, nos da las alertas, tal y cómo está configurado, cómo IDE:



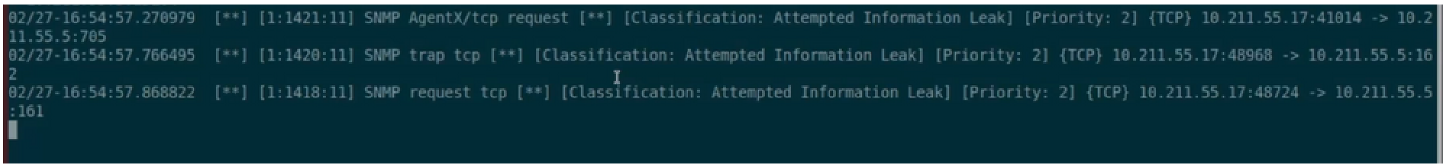
```
user@singular1:~$ sudo snort -A console -q -c /etc/snort/snort.conf -i eth0
02/27-16:51:41.626049  [**] [1:469:3] ICMP PING NMAP [**] [Classification: Attempted Information Leak] [Priority: 2] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:51:41.626049  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:51:41.636739  [**] [1:469:3] ICMP PING NMAP [**] [Classification: Attempted Information Leak] [Priority: 2] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:51:41.636739  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:51:41.646805  [**] [1:469:3] ICMP PING NMAP [**] [Classification: Attempted Information Leak] [Priority: 2] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:51:41.646805  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:51:41.657883  [**] [1:469:3] ICMP PING NMAP [**] [Classification: Attempted Information Leak] [Priority: 2] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:51:41.657883  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:51:41.668243  [**] [1:469:3] ICMP PING NMAP [**] [Classification: Attempted Information Leak] [Priority: 2] {ICMP} 10.211.55.17 -> 10.211.55.5
02/27-16:51:41.668243  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 10.211.55.17 -> 10.211.55.5
```

Volvemos a la maquina de ataque, y hacemos con nmap otro tipo de ataque para la detección de puertos y ver si snort lo ha detectado:



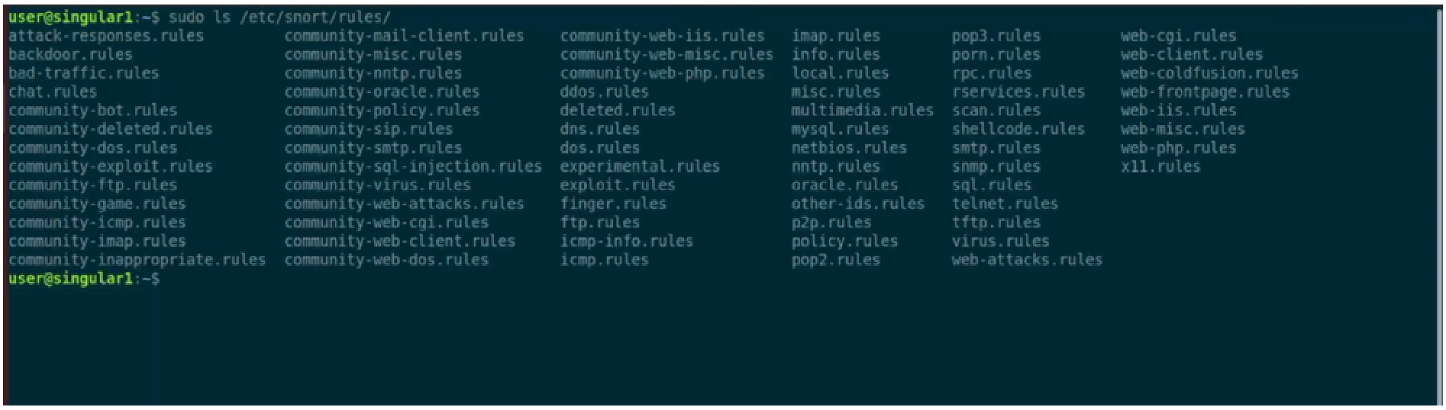
```
user@singular2:~$ nmap -p- 10.211.55.5
Starting Nmap 7.80 ( https://nmap.org ) at 2024-02-27 16:54 CET
Nmap scan report for ubuntu-linux-1.shared (10.211.55.5)
Host is up (0.00022s latency).
All 65535 scanned ports on ubuntu-linux-1.shared (10.211.55.5) are closed

Nmap done: 1 IP address (1 host up) scanned in 0.81 seconds
user@singular2:~$
```

En la maquina de defensa vemos los logs indicando que lo ha detectado (usando las mismas reglas que antes no hemos añadido nada, hemos lanzado snort y ahora estamos probando varias herramientas de ataque).



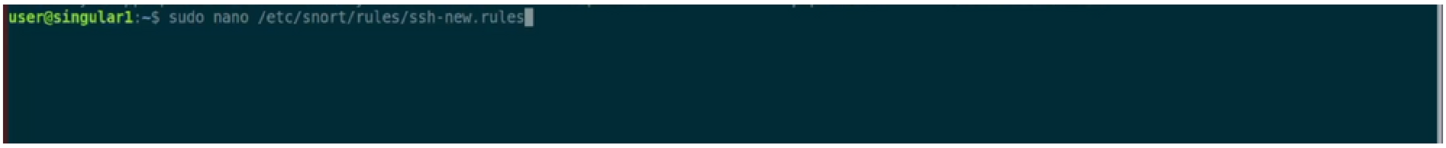
```
02/27-16:54:57.270979  [**] [1:1421:11] SNMP AgentX/tcp request [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 10.211.55.17:41014 -> 10.2
11.55.5:705
02/27-16:54:57.766495  [**] [1:1420:11] SNMP trap tcp [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 10.211.55.17:48968 -> 10.211.55.5:16
2
02/27-16:54:57.868822  [**] [1:1418:11] SNMP request tcp [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 10.211.55.17:48724 -> 10.211.55.5
:161
```

Interrumpimos el servidor de snort y vamos a crear nuestras propias reglas. Aqui tenemos el fichero con todas las reglas que snort tiene por defecto con todas las reglas que vienen al instalarlo. En éste directorio tenemos que crear nuestro documento con nuestras reglas, cómo ejemplo haremos una regla para crear conexiones ssh.



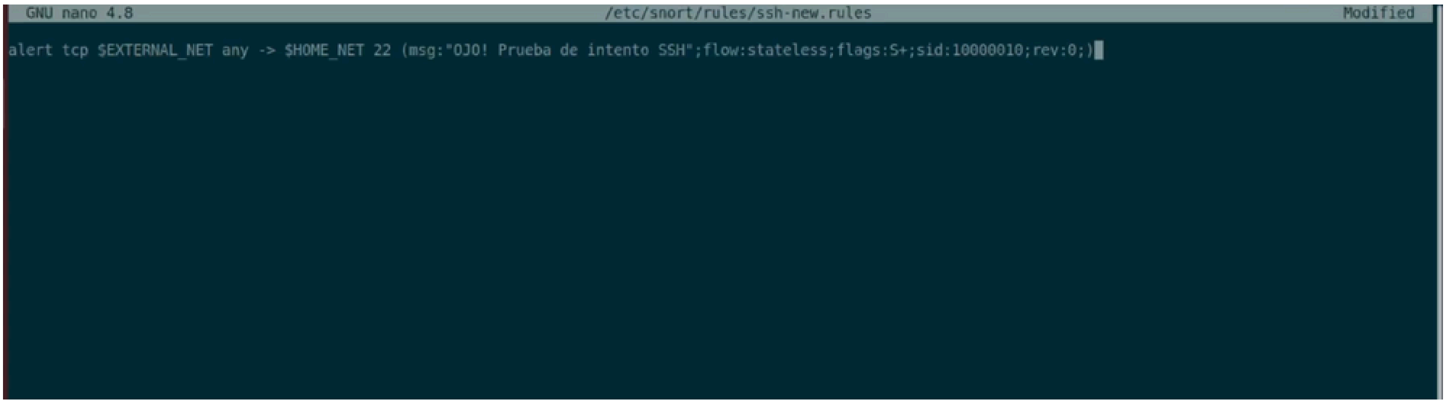
```
user@singular1:~$ sudo ls /etc/snort/rules/
attack-responses.rules         community-mail-client.rules    community-web-iis.rules   imap.rules        pop3.rules         web-cgi.rules
backdoor.rules                 community-misc.rules           community-web-misc.rules  info.rules        porn.rules         web-client.rules
bad-traffic.rules              community-nntp.rules           community-web-php.rules   local.rules       rpc.rules          web-coldfusion.rules
chat.rules                     community-oracle.rules         ddos.rules                misc.rules        rservices.rules    web-frontpage.rules
community-bot.rules            community-policy.rules         deleted.rules             multimedia.rules  scan.rules         web-iis.rules
community-deleted.rules        community-sip.rules            dns.rules                 mysql.rules       shellcode.rules    web-misc.rules
community-dos.rules            community-smtp.rules           dos.rules                 netbios.rules     smtp.rules         web-php.rules
community-exploit.rules        community-sql-injection.rules  experimental.rules        nntp.rules        snmp.rules         x11.rules
community-ftp.rules            community-virus.rules          exploit.rules             oracle.rules      sql.rules
community-game.rules           community-web-attacks.rules    finger.rules              other-ids.rules   telnet.rules
community-icmp.rules           community-web-cgi.rules        ftp.rules                 p2p.rules         tftp.rules
community-imap.rules           community-web-client.rules     icmp-info.rules           policy.rules      virus.rules
community-inappropriate.rules  community-web-dos.rules        icmp.rules                pop2.rules        web-attacks.rules
user@singular1:~$
```

Usamos nano para crear el fichero, en éste caso se llama ssh-new.rules:



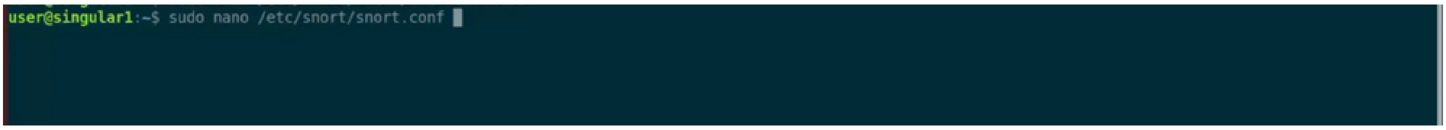
```
user@singular1:~$ sudo nano /etc/snort/rules/ssh-new.rules
```

Escribimos la regla para crear conexiones ssh:



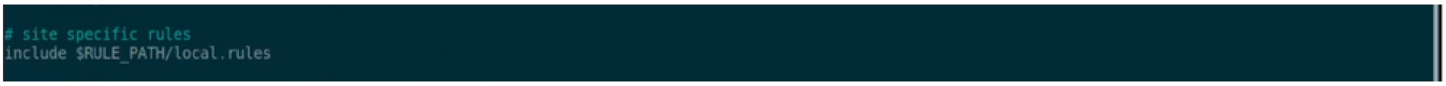
```
alert tcp $EXTERNAL_NET any -> $HOME_NET 22 (msg:"OJO! Prueba de intento SSH";flow:stateless;flags:S+;sid:10000010;rev:0;)
```

Ahora hay que añadir ésta regla nueva en el fichero de configuración, par aque snort la tenga en cuenta, accedemos al fichero de configuración:



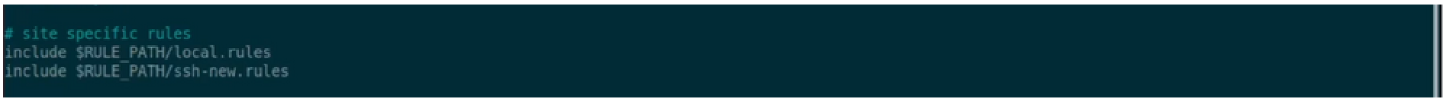
```
user@singular1:~$ sudo nano /etc/snort/snort.conf
```

Ahora buscamos la etiqueta que especifica que reglas vamos a utilizar:



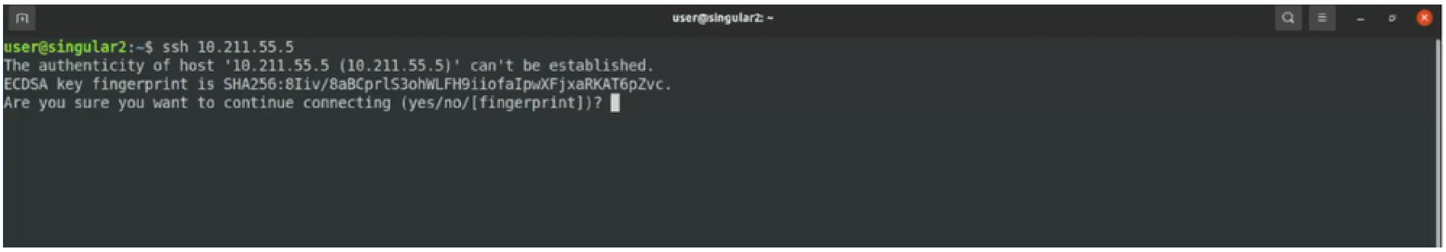
```
# site specific rules
include $RULE_PATH/local.rules
```

Y le añadimos el nombre de la nueva regla o le añadimos la ruta:



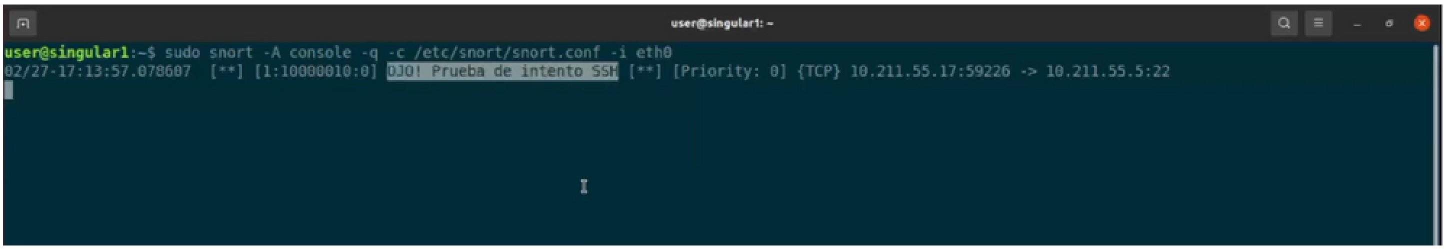
```
# site specific rules
include $RULE_PATH/local.rules
include $RULE_PATH/ssh-new.rules
```

Ahora vamos a levantar snort para que que esté activado y volvemos a la máquina de ataque para lanzar peticiones ssh y que snort las detecte:



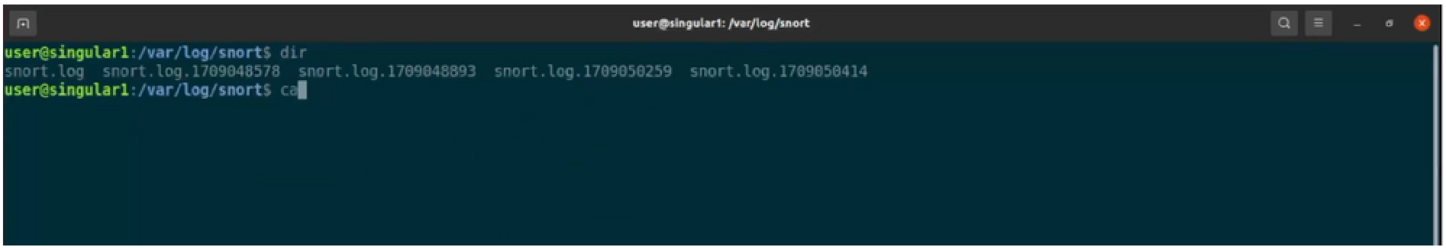
```
user@singular2:~$ ssh 10.211.55.5
The authenticity of host '10.211.55.5 (10.211.55.5)' can't be established.
ECDSA key fingerprint is SHA256:8Iiv/8aBCprlS3ohWLFH9iiofaIpwXFjxaRKAT6pZvc.
Are you sure you want to continue connecting (yes/no/[fingerprint])?
```

En la maquina de defensa vemos que snort lo detecta perfectamente y nos dice que regla se ha activado.



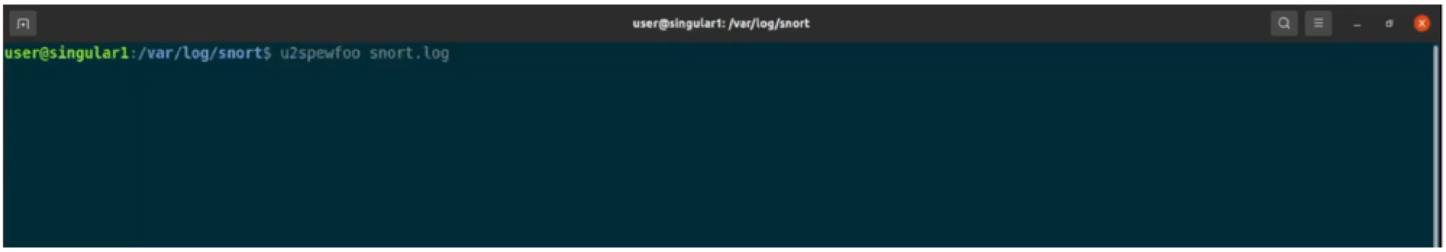
```
user@singular1:~$ sudo snort -A console -q -c /etc/snort/snort.conf -i eth0
02/27-17:13:57.078607  [**] [1:10000010:0] OJO! Prueba de intento SSH [**] [Priority: 0] {TCP} 10.211.55.17:59226 -> 10.211.55.5:22
```

Ahora vamos a ver los logs en snort:

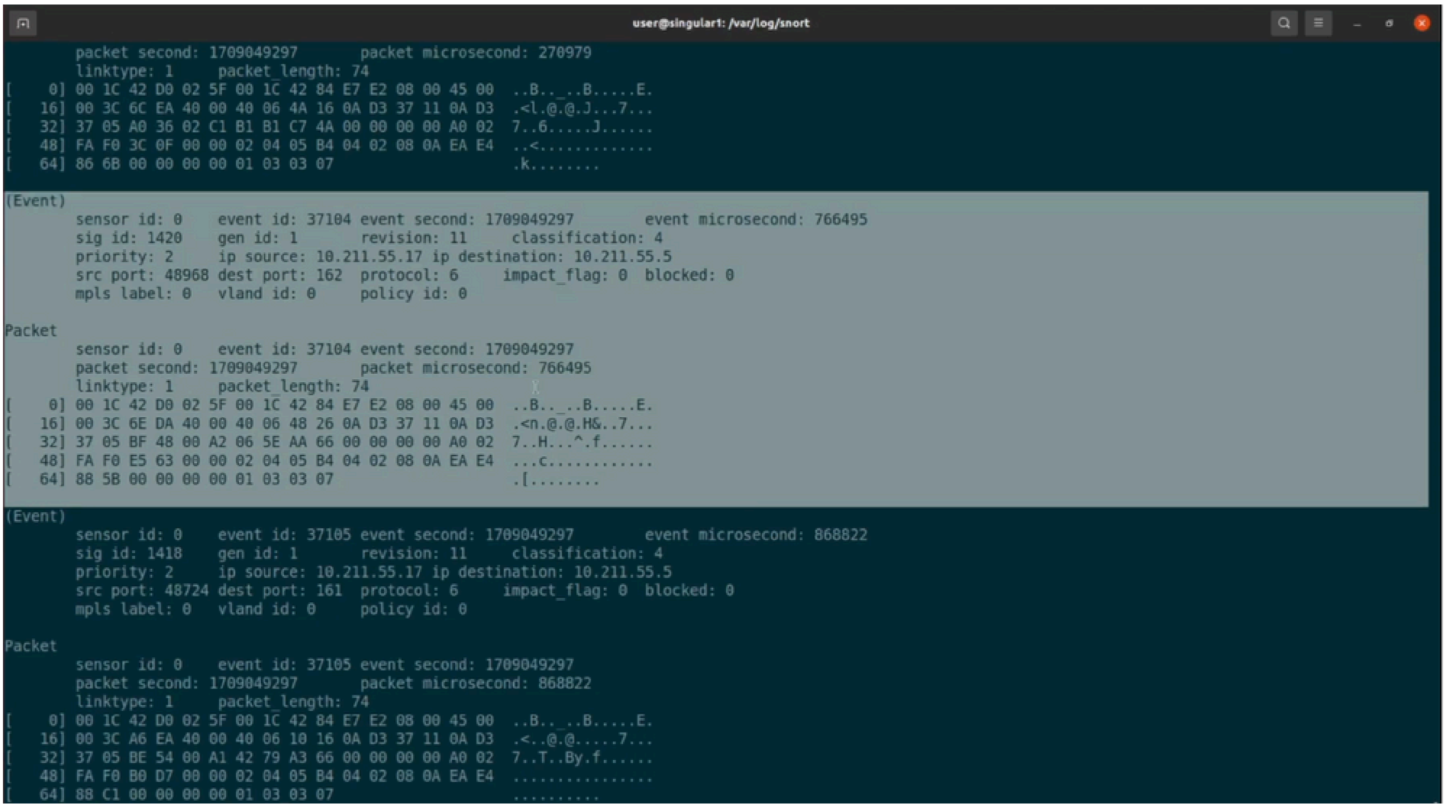


```
user@singular1:/var/log/snort$ dir
snort.log  snort.log.1709048578  snort.log.1709048893  snort.log.1709050259  snort.log.1709050414
user@singular1:/var/log/snort$ ca
```

Usando u2spewfoo podemos leer el archivo snort.log (sino está en binario). Y podríamos luego analizar, y hay un nivel de detallo impresionante.



```
user@singular1:/var/log/snort$ u2spewfoo snort.log
```



```
        packet second: 1709049297       packet microsecond: 270979
        linktype: 1     packet_length: 74
[    0] 00 1C 42 D0 02 5F 00 1C 42 84 E7 E2 08 00 45 00  ..B.._..B.....E.
[   16] 00 3C 6C EA 40 00 40 06 4A 16 0A D3 37 11 0A D3  .<l.@.@.J...7...
[   32] 37 05 A0 36 02 C1 B1 B1 C7 4A 00 00 00 00 A0 02  7..6.....J......
[   48] FA F0 3C 0F 00 00 02 04 05 B4 04 02 08 0A EA E4  ..<.............
[   64] 86 6B 00 00 00 00 01 03 03 07                    .k........

(Event)
        sensor id: 0    event id: 37104 event second: 1709049297        event microsecond: 766495
        sig id: 1420    gen id: 1       revision: 11     classification: 4
        priority: 2     ip source: 10.211.55.17 ip destination: 10.211.55.5
        src port: 48968 dest port: 162  protocol: 6     impact_flag: 0  blocked: 0
        mpls label: 0   vland id: 0     policy id: 0

Packet
        sensor id: 0    event id: 37104 event second: 1709049297
        packet second: 1709049297       packet microsecond: 766495
        linktype: 1     packet_length: 74
[    0] 00 1C 42 D0 02 5F 00 1C 42 84 E7 E2 08 00 45 00  ..B.._..B.....E.
[   16] 00 3C 6E DA 40 00 40 06 48 26 0A D3 37 11 0A D3  .<n.@.@.H&..7...
[   32] 37 05 BF 48 00 A2 06 5E AA 66 00 00 00 00 A0 02  7..H...^.f......
[   48] FA F0 E5 63 00 00 02 04 05 B4 04 02 08 0A EA E4  ...c............
[   64] 88 5B 00 00 00 00 01 03 03 07                    .[........

(Event)
        sensor id: 0    event id: 37105 event second: 1709049297        event microsecond: 868822
        sig id: 1418    gen id: 1       revision: 11     classification: 4
        priority: 2     ip source: 10.211.55.17 ip destination: 10.211.55.5
        src port: 48724 dest port: 161  protocol: 6     impact_flag: 0  blocked: 0
        mpls label: 0   vland id: 0     policy id: 0

Packet
        sensor id: 0    event id: 37105 event second: 1709049297
        packet second: 1709049297       packet microsecond: 868822
        linktype: 1     packet_length: 74
[    0] 00 1C 42 D0 02 5F 00 1C 42 84 E7 E2 08 00 45 00  ..B.._..B.....E.
[   16] 00 3C A6 EA 40 00 40 06 10 16 0A D3 37 11 0A D3  .<..@.@.....7...
[   32] 37 05 BE 54 00 A1 42 79 A3 66 00 00 00 00 A0 02  7..T..By.f......
[   48] FA F0 B0 D7 00 00 02 04 05 B4 04 02 08 0A EA E4  ................
[   64] 88 C1 00 00 00 00 01 03 03 07                    ..........
```