UNSW AUSTRALIA

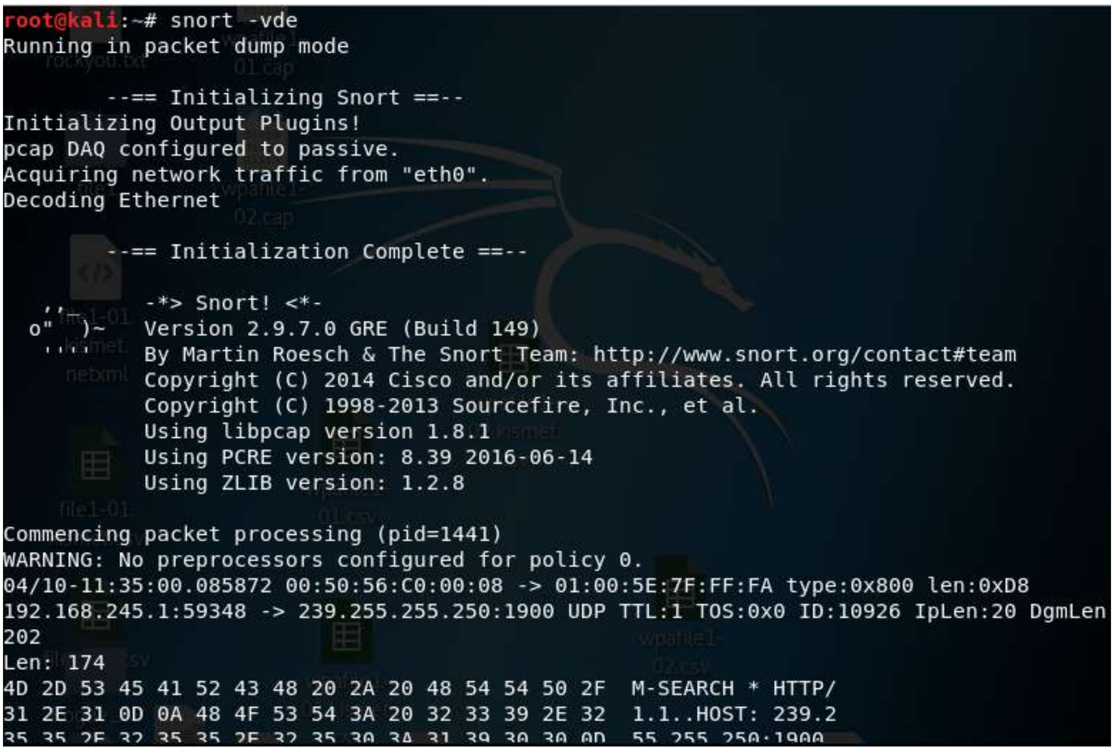
COMP9337: Securing Wireless Networks

GROUP SWN19-B

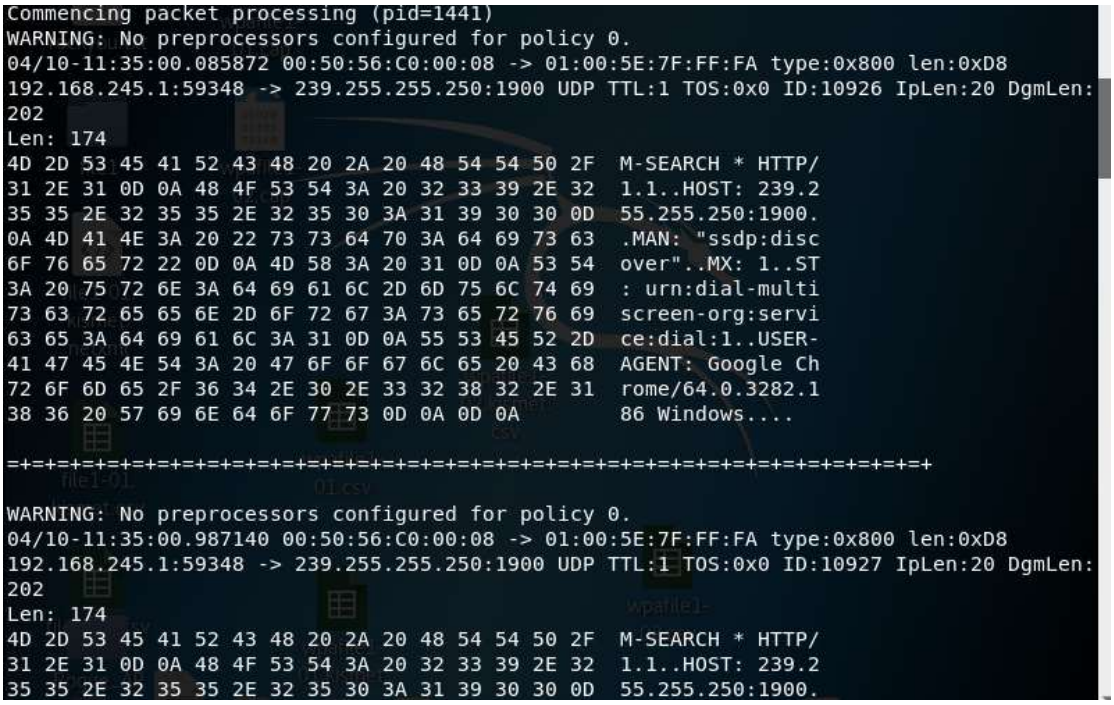
Name: Wenxun Peng ZID: z5195349

Name: Tianyi Jiang ZID: z5159471

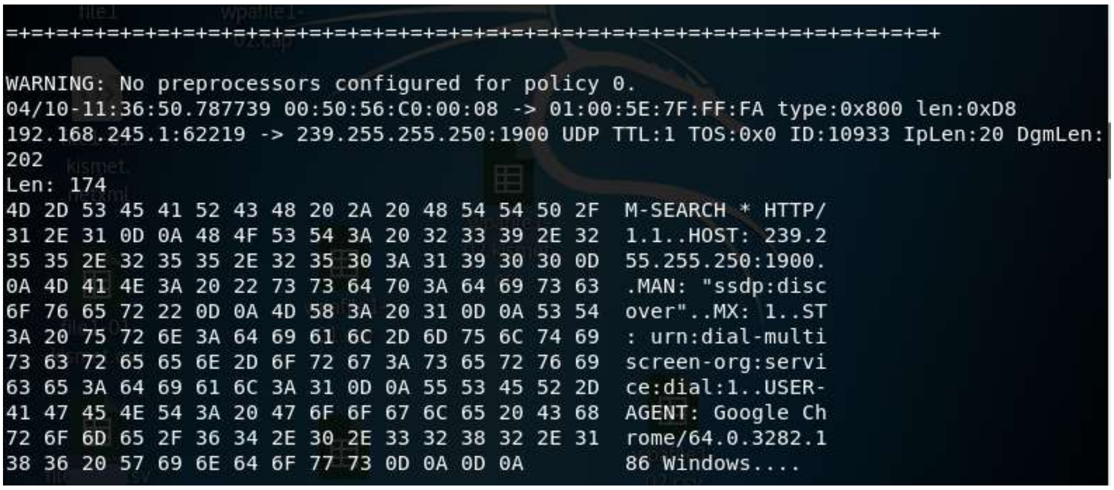
Task 1: Find the flags that will display data-link headers and the application layer data?



```
root@kali:~# snort -vde
Running in packet dump mode

        --== Initializing Snort ==--
Initializing Output Plugins!
pcap DAQ configured to passive.
Acquiring network traffic from "eth0".
Decoding Ethernet

        --== Initialization Complete ==--

   ,,_     -*> 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 1.8.1
           Using PCRE version: 8.39 2016-06-14
           Using ZLIB version: 1.2.8

Commencing packet processing (pid=1441)
WARNING: No preprocessors configured for policy 0.
04/10-11:35:00.085872 00:50:56:C0:00:08 -> 01:00:5E:7F:FF:FA type:0x800 len:0xD8
192.168.245.1:59348 -> 239.255.255.250:1900 UDP TTL:1 TOS:0x0 ID:10926 IpLen:20 DgmLen
202
Len: 174
4D 2D 53 45 41 52 43 48 20 2A 20 48 54 54 50 2F  M-SEARCH * HTTP/
31 2E 31 0D 0A 48 4F 53 54 3A 20 32 33 39 2E 32  1.1..HOST: 239.2
35 35 2E 32 35 35 2E 32 35 30 3A 31 39 30 30 0D  55.255.250:1900.
```



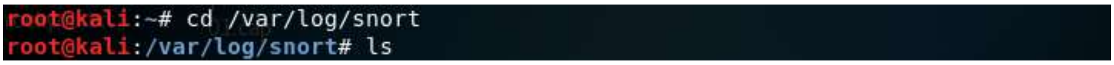
```
Commencing packet processing (pid=1441)
WARNING: No preprocessors configured for policy 0.
04/10-11:35:00.085872 00:50:56:C0:00:08 -> 01:00:5E:7F:FF:FA type:0x800 len:0xD8
192.168.245.1:59348 -> 239.255.255.250:1900 UDP TTL:1 TOS:0x0 ID:10926 IpLen:20 DgmLen:
202
Len: 174
4D 2D 53 45 41 52 43 48 20 2A 20 48 54 54 50 2F  M-SEARCH * HTTP/
31 2E 31 0D 0A 48 4F 53 54 3A 20 32 33 39 2E 32  1.1..HOST: 239.2
35 35 2E 32 35 35 2E 32 35 30 3A 31 39 30 30 0D  55.255.250:1900.
0A 4D 41 4E 3A 20 22 73 73 64 70 3A 64 69 73 63  .MAN: "ssdp:disc
6F 76 65 72 22 0D 0A 4D 58 3A 20 31 0D 0A 53 54  over"..MX: 1..ST
3A 20 75 72 6E 3A 64 69 61 6C 2D 6D 75 6C 74 69  : urn:dial-multi
73 63 72 65 65 6E 2D 6F 72 67 3A 73 65 72 76 69  screen-org:servi
63 65 3A 64 69 61 6C 3A 31 0D 0A 55 53 45 52 2D  ce:dial:1..USER-
41 47 45 4E 54 3A 20 47 6F 6F 67 6C 65 20 43 68  AGENT: Google Ch
72 6F 6D 65 2F 36 34 2E 30 2E 33 32 38 32 2E 31  rome/64.0.3282.1
38 36 20 57 69 6E 64 6F 77 73 0D 0A 0D 0A        86 Windows....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
04/10-11:35:00.987140 00:50:56:C0:00:08 -> 01:00:5E:7F:FF:FA type:0x800 len:0xD8
192.168.245.1:59348 -> 239.255.255.250:1900 UDP TTL:1 TOS:0x0 ID:10927 IpLen:20 DgmLen:
202
Len: 174
4D 2D 53 45 41 52 43 48 20 2A 20 48 54 54 50 2F  M-SEARCH * HTTP/
31 2E 31 0D 0A 48 4F 53 54 3A 20 32 33 39 2E 32  1.1..HOST: 239.2
35 35 2E 32 35 35 2E 32 35 30 3A 31 39 30 30 0D  55.255.250:1900.
```



```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
04/10-11:36:50.787739 00:50:56:C0:00:08 -> 01:00:5E:7F:FF:FA type:0x800 len:0xD8
192.168.245.1:62219 -> 239.255.255.250:1900 UDP TTL:1 TOS:0x0 ID:10933 IpLen:20 DgmLen:
202
Len: 174
4D 2D 53 45 41 52 43 48 20 2A 20 48 54 54 50 2F  M-SEARCH * HTTP/
31 2E 31 0D 0A 48 4F 53 54 3A 20 32 33 39 2E 32  1.1..HOST: 239.2
35 35 2E 32 35 35 2E 32 35 30 3A 31 39 30 30 0D  55.255.250:1900.
0A 4D 41 4E 3A 20 22 73 73 64 70 3A 64 69 73 63  .MAN: "ssdp:disc
6F 76 65 72 22 0D 0A 4D 58 3A 20 31 0D 0A 53 54  over"..MX: 1..ST
3A 20 75 72 6E 3A 64 69 61 6C 2D 6D 75 6C 74 69  : urn:dial-multi
73 63 72 65 65 6E 2D 6F 72 67 3A 73 65 72 76 69  screen-org:servi
63 65 3A 64 69 61 6C 3A 31 0D 0A 55 53 45 52 2D  ce:dial:1..USER-
41 47 45 4E 54 3A 20 47 6F 6F 67 6C 65 20 43 68  AGENT: Google Ch
72 6F 6D 65 2F 36 34 2E 30 2E 33 32 38 32 2E 31  rome/64.0.3282.1
38 36 20 57 69 6E 64 6F 77 73 0D 0A 0D 0A        86 Windows....
```

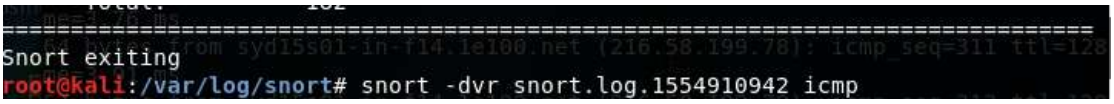
Task 2: Run a command in snort to capture only ICMP packets. (For testing, you may have to use ping to generate some ICMP packets if your network is not busy)

First, we can use the command below to know the location of the log files:

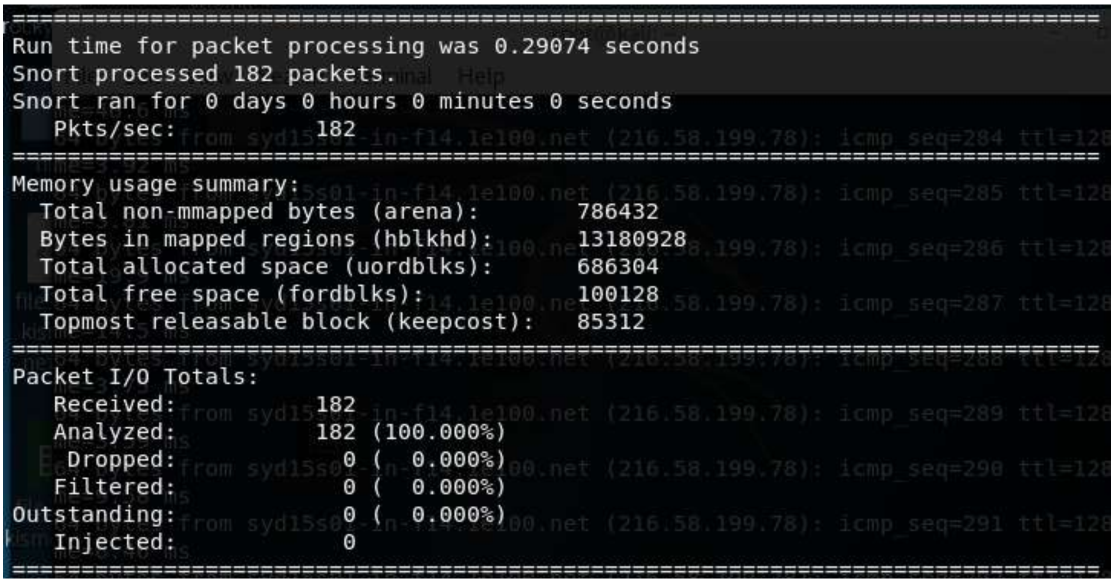


```
root@kali:~# cd /var/log/snort
root@kali:/var/log/snort# ls
```

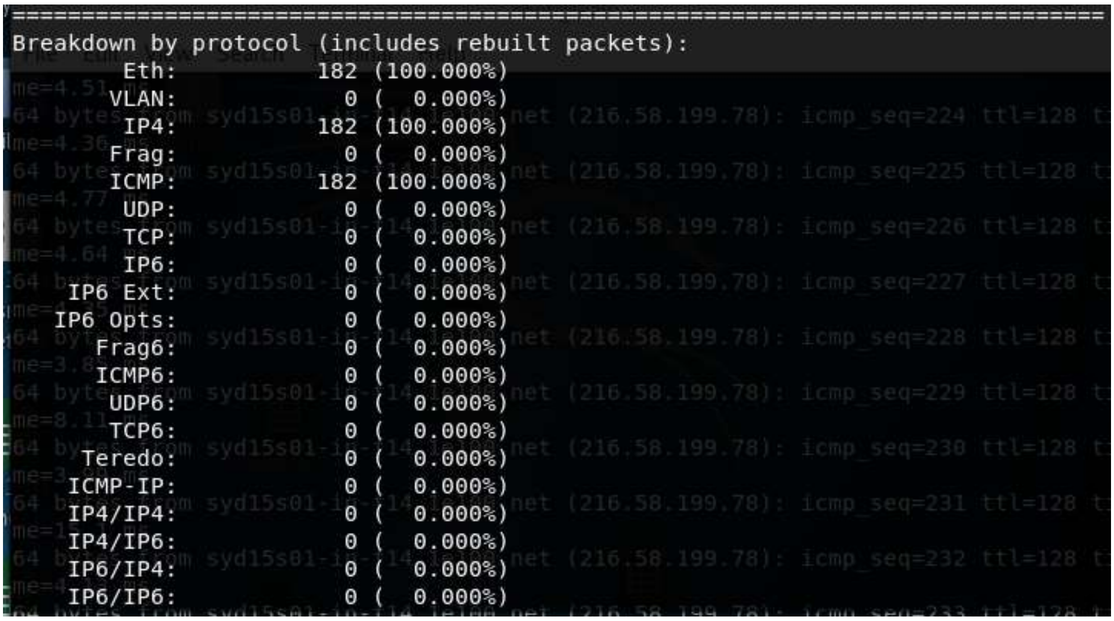
And then, we can get the ICMP packets:



```
===============================================================================
Snort exiting
root@kali:/var/log/snort# snort -dvr snort.log.1554910942 icmp
```

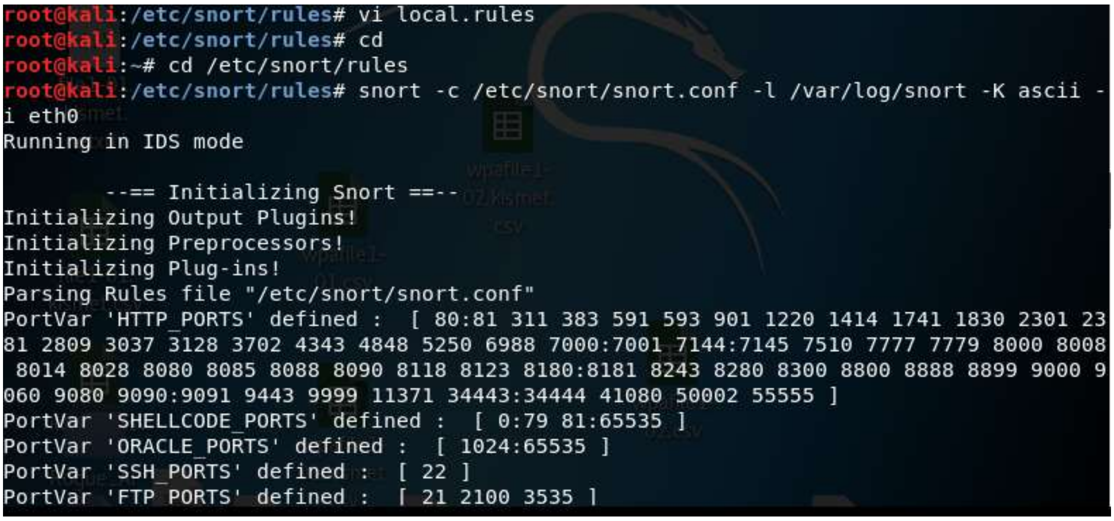


```
===============================================================================
Run time for packet processing was 0.29074 seconds
Snort processed 182 packets.
Snort ran for 0 days 0 hours 0 minutes 0 seconds
   Pkts/sec:          182
===============================================================================
Memory usage summary:
  Total non-mmapped bytes (arena):       786432
  Bytes in mapped regions (hblkhd):      13180928
  Total allocated space (uordblks):      686304
  Total free space (fordblks):           100128
  Topmost releasable block (keepcost):   85312
===============================================================================
Packet I/O Totals:
   Received:          182
   Analyzed:          182 (100.000%)
    Dropped:            0 (  0.000%)
   Filtered:            0 (  0.000%)
Outstanding:            0 (  0.000%)
   Injected:            0
===============================================================================
```

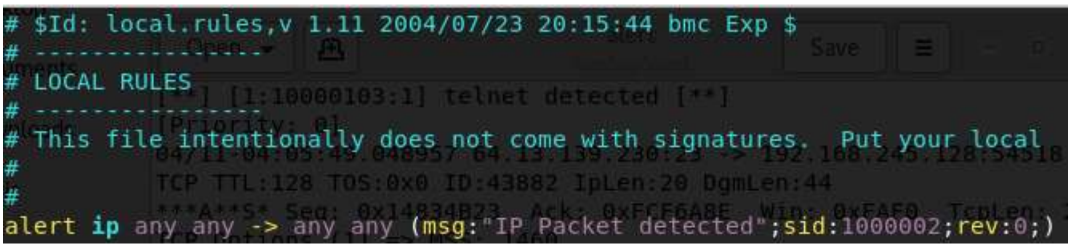


```
===============================================================================
Breakdown by protocol (includes rebuilt packets):
        Eth:          182 (100.000%)
       VLAN:            0 (  0.000%)
        IP4:          182 (100.000%)
       Frag:            0 (  0.000%)
       ICMP:          182 (100.000%)
        UDP:            0 (  0.000%)
        TCP:            0 (  0.000%)
        IP6:            0 (  0.000%)
    IP6 Ext:            0 (  0.000%)
   IP6 Opts:            0 (  0.000%)
      Frag6:            0 (  0.000%)
      ICMP6:            0 (  0.000%)
       UDP6:            0 (  0.000%)
       TCP6:            0 (  0.000%)
     Teredo:            0 (  0.000%)
    ICMP-IP:            0 (  0.000%)
    IP4/IP4:            0 (  0.000%)
    IP4/IP6:            0 (  0.000%)
    IP6/IP4:            0 (  0.000%)
    IP6/IP6:            0 (  0.000%)
```

Task 3: Now execute Snort with the new rule in effect using the following command.

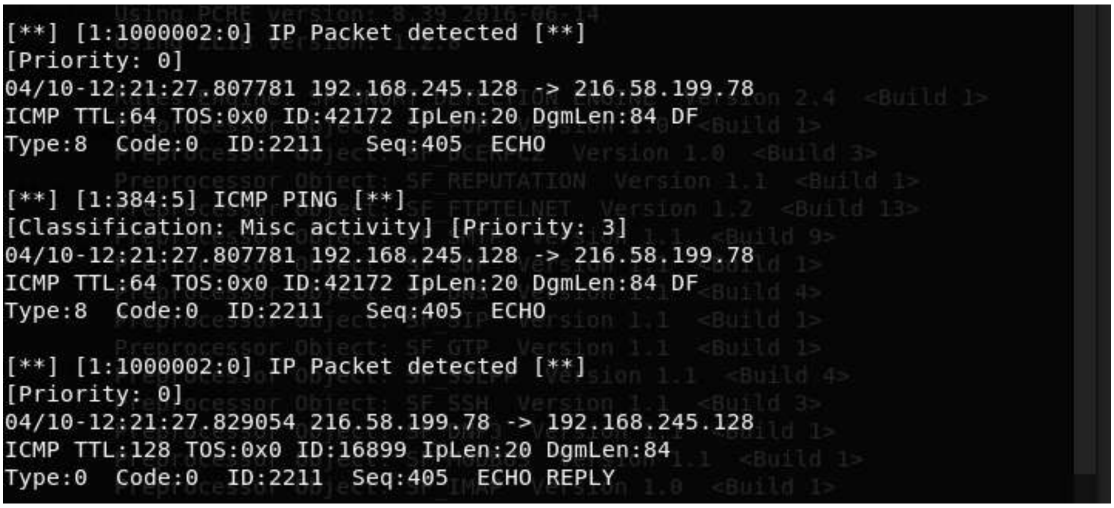


```
root@kali:/etc/snort/rules# vi local.rules
root@kali:/etc/snort/rules# cd
root@kali:~# cd /etc/snort/rules
root@kali:/etc/snort/rules# snort -c /etc/snort/snort.conf -l /var/log/snort -K ascii -
i eth0
Running in IDS mode

        --== Initializing Snort ==--
Initializing Output Plugins!
Initializing Preprocessors!
Initializing Plug-ins!
Parsing Rules file "/etc/snort/snort.conf"
PortVar 'HTTP_PORTS' defined :  [ 80:81 311 383 591 593 901 1220 1414 1741 1830 2301 23
81 2809 3037 3128 3702 4343 4848 5250 6988 7000:7001 7144:7145 7510 7777 7779 8000 8008
 8014 8028 8080 8085 8088 8090 8118 8123 8180:8181 8243 8280 8300 8800 8888 8899 9000 9
060 9080 9090:9091 9443 9999 11371 34443:34444 41080 50002 55555 ]
PortVar 'SHELLCODE_PORTS' defined :  [ 0:79 81:65535 ]
PortVar 'ORACLE_PORTS' defined :  [ 1024:65535 ]
PortVar 'SSH_PORTS' defined :  [ 22 ]
PortVar 'FTP_PORTS' defined :  [ 21 2100 3535 ]
```

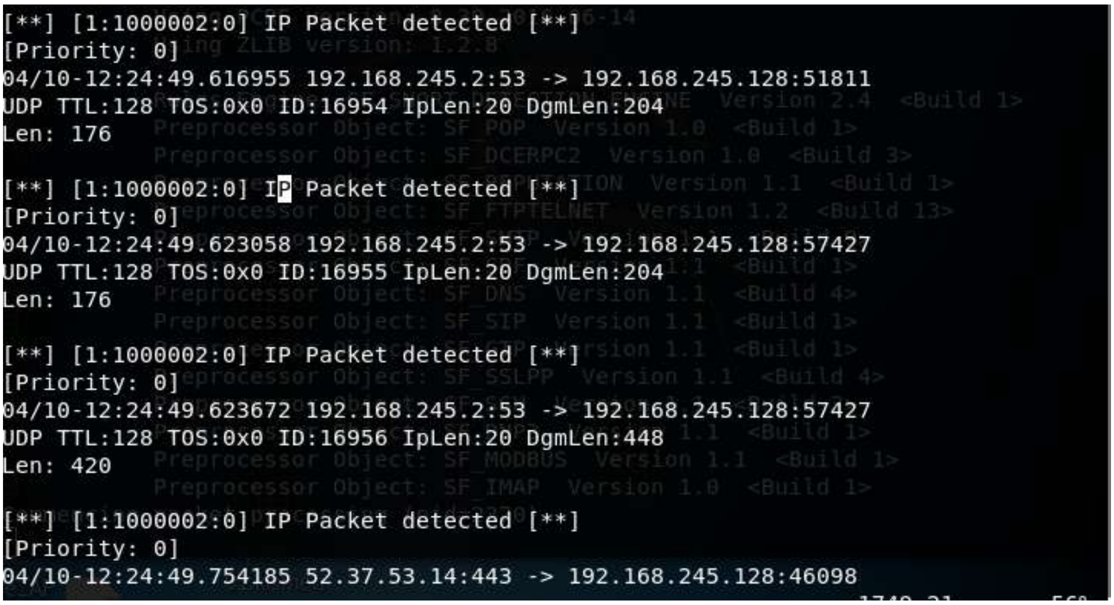


```
# $Id: local.rules,v 1.11 2004/07/23 20:15:44 bmc Exp $
# ----------------
# LOCAL RULES
# ----------------
# This file intentionally does not come with signatures.  Put your local
#
#
alert ip any any -> any any (msg:"IP Packet detected";sid:1000002;rev:0;)
```

Thus, we can get below from alert file:



```
[**] [1:1000002:0] IP Packet detected [**]
[Priority: 0]
04/10-12:21:27.807781 192.168.245.128 -> 216.58.199.78
ICMP TTL:64 TOS:0x0 ID:42172 IpLen:20 DgmLen:84 DF
Type:8  Code:0  ID:2211   Seq:405  ECHO

[**] [1:384:5] ICMP PING [**]
[Classification: Misc activity] [Priority: 3]
04/10-12:21:27.807781 192.168.245.128 -> 216.58.199.78
ICMP TTL:64 TOS:0x0 ID:42172 IpLen:20 DgmLen:84 DF
Type:8  Code:0  ID:2211   Seq:405  ECHO

[**] [1:1000002:0] IP Packet detected [**]
[Priority: 0]
04/10-12:21:27.829054 216.58.199.78 -> 192.168.245.128
ICMP TTL:128 TOS:0x0 ID:16899 IpLen:20 DgmLen:84
Type:0  Code:0  ID:2211  Seq:405  ECHO REPLY
```

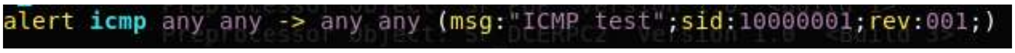


```
[**] [1:1000002:0] IP Packet detected [**]
[Priority: 0]
04/10-12:24:49.616955 192.168.245.2:53 -> 192.168.245.128:51811
UDP TTL:128 TOS:0x0 ID:16954 IpLen:20 DgmLen:204
Len: 176

[**] [1:1000002:0] IP Packet detected [**]
[Priority: 0]
04/10-12:24:49.623058 192.168.245.2:53 -> 192.168.245.128:57427
UDP TTL:128 TOS:0x0 ID:16955 IpLen:20 DgmLen:204
Len: 176

[**] [1:1000002:0] IP Packet detected [**]
[Priority: 0]
04/10-12:24:49.623672 192.168.245.2:53 -> 192.168.245.128:57427
UDP TTL:128 TOS:0x0 ID:16956 IpLen:20 DgmLen:448
Len: 420

[**] [1:1000002:0] IP Packet detected [**]
[Priority: 0]
04/10-12:24:49.754185 52.37.53.14:443 -> 192.168.245.128:46098
```

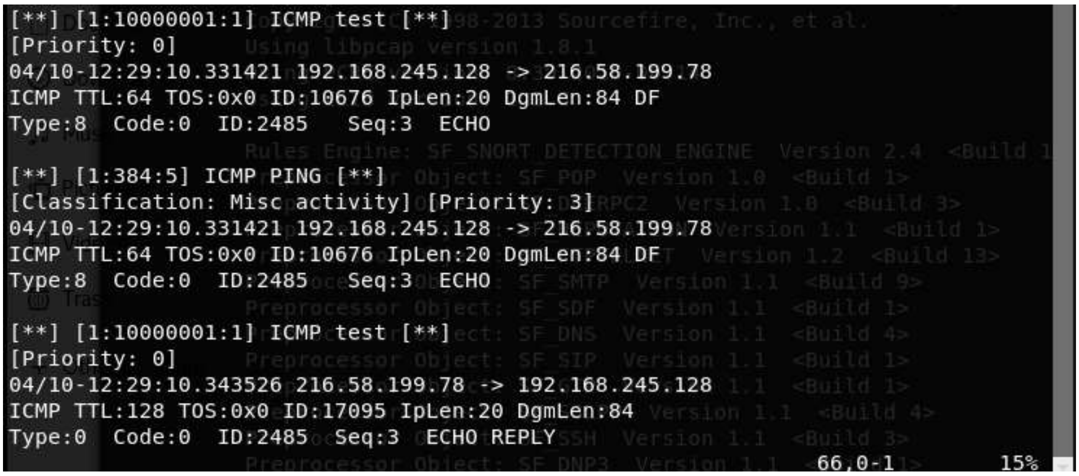
Since the rule will generate an alert message for every captured IP packet, it will soon fill up your disk space if you leave it there and it is useless for finding something wrong in the network.

Task 4: Write the new rule that you used.

We can write below rules and we can use this rule like above.



```
alert icmp any any -> any any (msg:"ICMP test";sid:10000001;rev:001;)
```



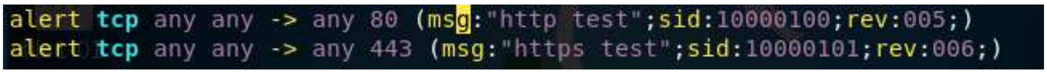
```
[**] [1:10000001:1] ICMP test [**]
[Priority: 0]
04/10-12:29:10.331421 192.168.245.128 -> 216.58.199.78
ICMP TTL:64 TOS:0x0 ID:10676 IpLen:20 DgmLen:84 DF
Type:8  Code:0  ID:2485   Seq:3  ECHO

[**] [1:384:5] ICMP PING [**]
[Classification: Misc activity] [Priority: 3]
04/10-12:29:10.331421 192.168.245.128 -> 216.58.199.78
ICMP TTL:64 TOS:0x0 ID:10676 IpLen:20 DgmLen:84 DF
Type:8  Code:0  ID:2485   Seq:3  ECHO

[**] [1:10000001:1] ICMP test [**]
[Priority: 0]
04/10-12:29:10.343526 216.58.199.78 -> 192.168.245.128
ICMP TTL:128 TOS:0x0 ID:17095 IpLen:20 DgmLen:84
Type:0  Code:0  ID:2485  Seq:3  ECHO REPLY
                                                  66,0-1        15%
```

Task 5: Explain what the following rule is doing?

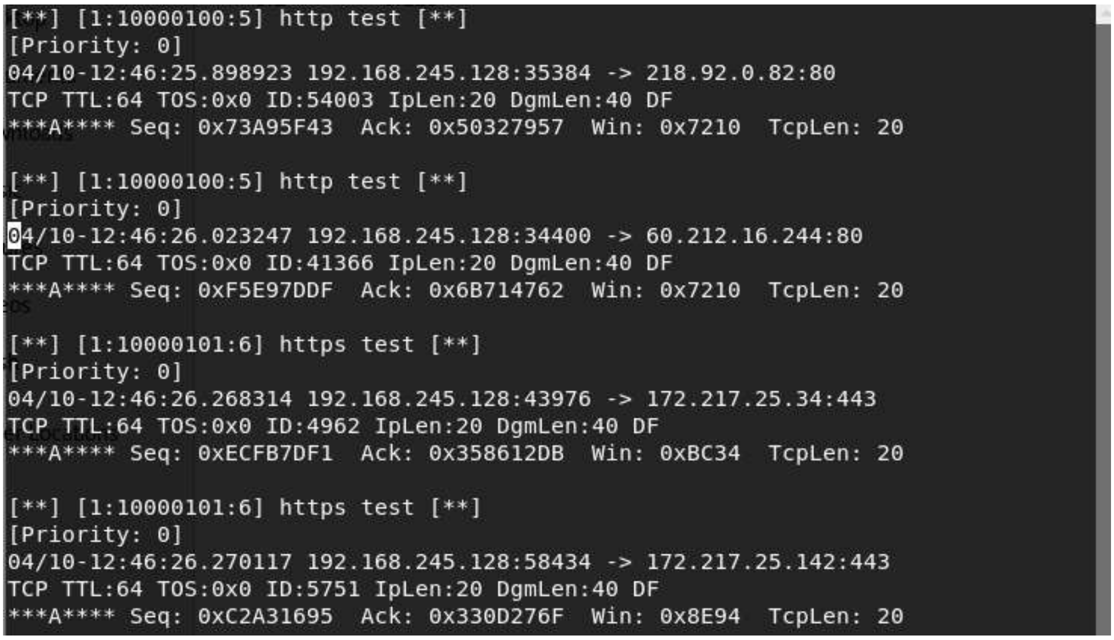
alert tcp !192.168.1.0/24 any -> 192.168.1.0/24 !:1024

Alert any source IP address except 192.168.1.0/24, whose destination is 192.168.1.0/24 except the port is less than or equal 1024.

TASK 6: Write a rule to alert when a HTTP GET is detected.



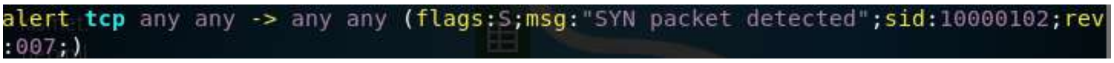
```
alert tcp any any -> any 80 (msg:"http test";sid:10000100;rev:005;)
alert tcp any any -> any 443 (msg:"https test";sid:10000101;rev:006;)
```



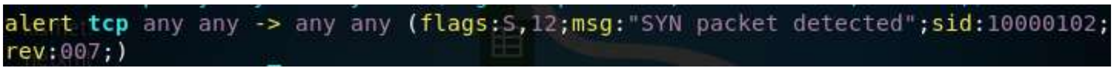
```
[**] [1:10000100:5] http test [**]
[Priority: 0]
04/10-12:46:25.898923 192.168.245.128:35384 -> 218.92.0.82:80
TCP TTL:64 TOS:0x0 ID:54003 IpLen:20 DgmLen:40 DF
***A**** Seq: 0x73A95F43  Ack: 0x50327957  Win: 0x7210  TcpLen: 20

[**] [1:10000100:5] http test [**]
[Priority: 0]
04/10-12:46:26.023247 192.168.245.128:34400 -> 60.212.16.244:80
TCP TTL:64 TOS:0x0 ID:41366 IpLen:20 DgmLen:40 DF
***A**** Seq: 0xF5E97DDF  Ack: 0x6B714762  Win: 0x7210  TcpLen: 20

[**] [1:10000101:6] https test [**]
[Priority: 0]
04/10-12:46:26.268314 192.168.245.128:43976 -> 172.217.25.34:443
TCP TTL:64 TOS:0x0 ID:4962 IpLen:20 DgmLen:40 DF
***A**** Seq: 0xECFB7DF1  Ack: 0x358612DB  Win: 0xBC34  TcpLen: 20

[**] [1:10000101:6] https test [**]
[Priority: 0]
04/10-12:46:26.270117 192.168.245.128:58434 -> 172.217.25.142:443
TCP TTL:64 TOS:0x0 ID:5751 IpLen:20 DgmLen:40 DF
***A**** Seq: 0xC2A31695  Ack: 0x330D276F  Win: 0x8E94  TcpLen: 20
```

From the rules, we can know that we alert the packets from any source and the destination is http (port is 80) or https (port is 443). Thus, the first some alerts record the HTTP GET.

Task 7: Write a rule that generates an alert if it detects TCP connection attempt using SYN packets.



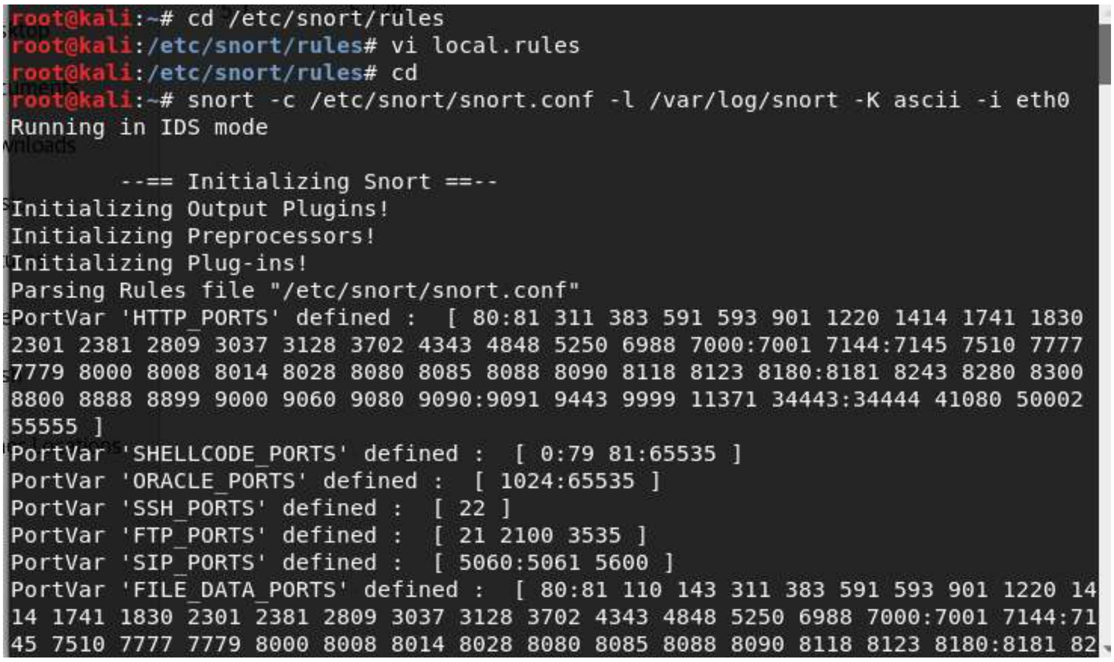
```
alert tcp any any -> any any (flags:S;msg:"SYN packet detected";sid:10000102;rev
:007;)
```



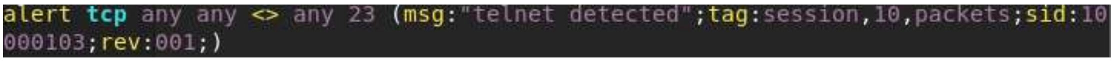
```
alert tcp any any -> any any (flags:S,12;msg:"SYN packet detected";sid:10000102;
rev:007;)
```

We use these two rules and also ask tutor, but we still can't get the result. Tutor says maybe it's the different version of the kali.

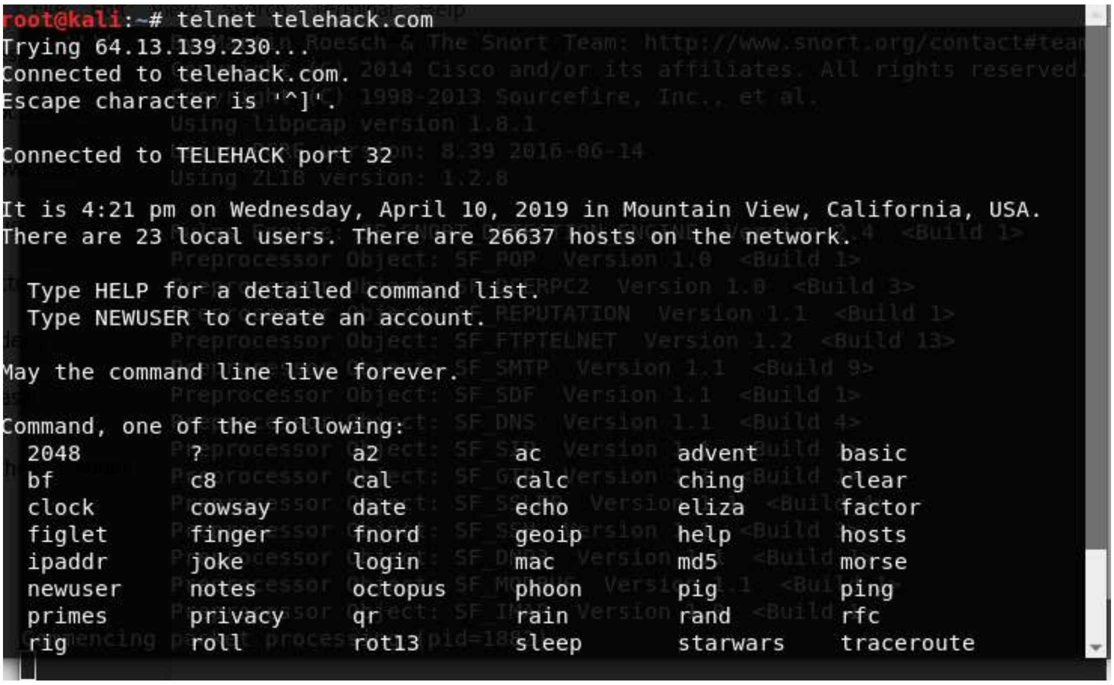
Task 8: Write a rule that detects a telnet session initiation. Once this session is detected, the rule should log the next 10 packets of this session.



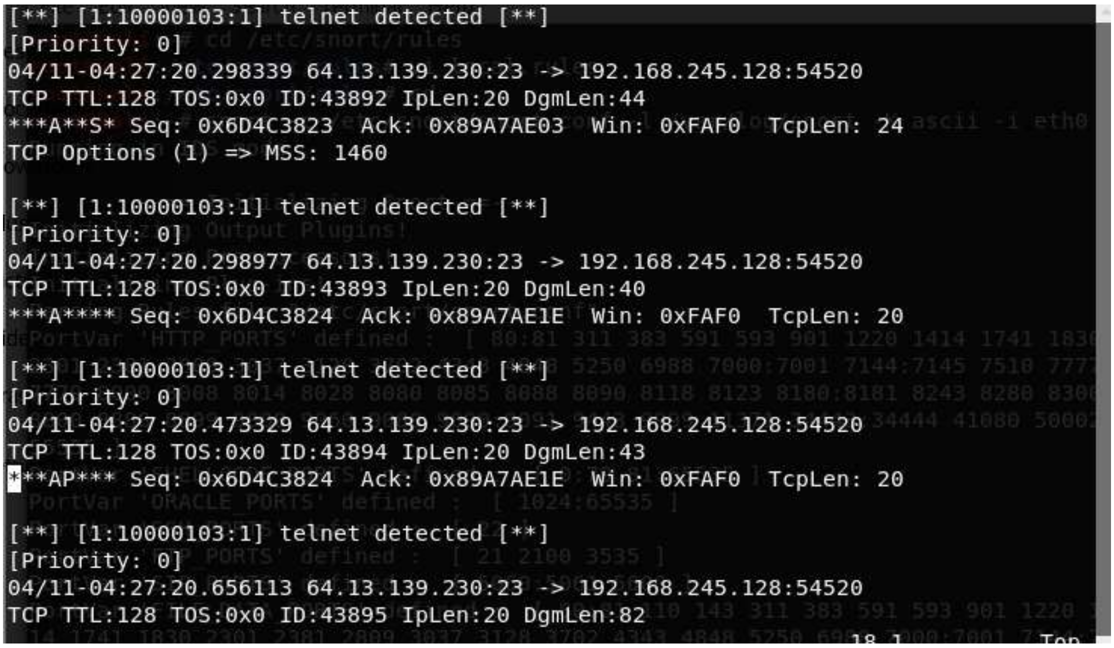
```
root@kali:~# cd /etc/snort/rules
root@kali:/etc/snort/rules# vi local.rules
root@kali:/etc/snort/rules# cd
root@kali:~# snort -c /etc/snort/snort.conf -l /var/log/snort -K ascii -i eth0
Running in IDS mode

        --== Initializing Snort ==--
Initializing Output Plugins!
Initializing Preprocessors!
Initializing Plug-ins!
Parsing Rules file "/etc/snort/snort.conf"
PortVar 'HTTP_PORTS' defined :  [ 80:81 311 383 591 593 901 1220 1414 1741 1830
2301 2381 2809 3037 3128 3702 4343 4848 5250 6988 7000:7001 7144:7145 7510 7777
7779 8000 8008 8014 8028 8080 8085 8088 8090 8118 8123 8180:8181 8243 8280 8300
8800 8888 8899 9000 9060 9080 9090:9091 9443 9999 11371 34443:34444 41080 50002
55555 ]
PortVar 'SHELLCODE_PORTS' defined :  [ 0:79 81:65535 ]
PortVar 'ORACLE_PORTS' defined :  [ 1024:65535 ]
PortVar 'SSH_PORTS' defined :  [ 22 ]
PortVar 'FTP_PORTS' defined :  [ 21 2100 3535 ]
PortVar 'SIP_PORTS' defined :  [ 5060:5061 5600 ]
PortVar 'FILE_DATA_PORTS' defined :  [ 80:81 110 143 311 383 591 593 901 1220 14
14 1741 1830 2301 2381 2809 3037 3128 3702 4343 4848 5250 6988 7000:7001 7144:71
45 7510 7777 7779 8000 8008 8014 8028 8080 8085 8088 8090 8118 8123 8180:8181 82
```



```
alert tcp any any <> any 23 (msg:"telnet detected";tag:session,10,packets;sid:10
000103;rev:001;)
```



```
root@kali:~# telnet telehack.com
Trying 64.13.139.230...
Connected to telehack.com.
Escape character is '^]'.

Connected to TELEHACK port 32

It is 4:21 pm on Wednesday, April 10, 2019 in Mountain View, California, USA.
There are 23 local users. There are 26637 hosts on the network.

  Type HELP for a detailed command list.
  Type NEWUSER to create an account.

May the command line live forever.

Command, one of the following:
  2048        ?           a2          ac          advent      basic
  bf          c8          cal         calc        ching       clear
  clock       cowsay      date        echo        eliza       factor
  figlet      finger      fnord       geoip       help        hosts
  ipaddr      joke        login       mac         md5         morse
  newuser     notes       octopus     phoon       pig         ping
  primes      privacy     qr          rain        rand        rfc
  rig         roll        rot13       sleep       starwars    traceroute
```



```
[**] [1:10000103:1] telnet detected [**]
[Priority: 0]
04/11-04:27:20.298339 64.13.139.230:23 -> 192.168.245.128:54520
TCP TTL:128 TOS:0x0 ID:43892 IpLen:20 DgmLen:44
***A**S* Seq: 0x6D4C3823  Ack: 0x89A7AE03  Win: 0xFAF0  TcpLen: 24
TCP Options (1) => MSS: 1460

[**] [1:10000103:1] telnet detected [**]
[Priority: 0]
04/11-04:27:20.298977 64.13.139.230:23 -> 192.168.245.128:54520
TCP TTL:128 TOS:0x0 ID:43893 IpLen:20 DgmLen:40
***A**** Seq: 0x6D4C3824  Ack: 0x89A7AE1E  Win: 0xFAF0  TcpLen: 20

[**] [1:10000103:1] telnet detected [**]
[Priority: 0]
04/11-04:27:20.473329 64.13.139.230:23 -> 192.168.245.128:54520
TCP TTL:128 TOS:0x0 ID:43894 IpLen:20 DgmLen:43
***AP*** Seq: 0x6D4C3824  Ack: 0x89A7AE1E  Win: 0xFAF0  TcpLen: 20

[**] [1:10000103:1] telnet detected [**]
[Priority: 0]
04/11-04:27:20.656113 64.13.139.230:23 -> 192.168.245.128:54520
TCP TTL:128 TOS:0x0 ID:43895 IpLen:20 DgmLen:82
```