CSE3502 Information Security Management

Review-3

Network Intrusion Detection and Prevention using Snort tool

TEAM MEMBERS

19BCE0607	Shivangi Chaurasia
19BCE2506	Utkarsh Sharma
19BCT0207	Sohan Kantimahanthi

Problem Statement

Every action in today's information era results in the creation of data in some or the other form. According to estimates, over 300,000 tweets are sent every minute, and over 4 million Facebook postings are made every minute. Knowing that additional users and data requires increased security. Security and dependability have become key concerns for individuals and organisations. So observing the current scenario we have decided to address various terminologies, strategies, and practical procedures associated with the SNORT Intrusion Detection and Prevention System(IDPS) in our project . Basic architecture on how attackers find vulnerabilities. Using SNORT tool to detect these vulnerabilities.

Introduction

Security and reliability are the major concern of our daily life usage of any network. But with the swift advancements in network technology, attacks are becoming more sophisticated than defenses. Although firewalls and router-based packet filtering are essential elements of an overall network security topology, they are not enough on their own. So, to brace the network from unauthorized access the idea of SNORT Intrusion Detection System (IDS) and SNORT Intrusion Prevention System (IPS) is attracting security experts. Being an open source IDS, Snort can be easily configured and deployed in any environment. To overcome various challenges like low detection rate, incapable of handling huge traffic, unsupported automated tuning, etc. that are identified during literature review, our project proposes a level based architecture. All the levels are designed as incremental i.e. capable of providing the desired functionality and also its lower levels. To prove the efficiency of the proposed architecture, we integrated all of it into Snort Tool using Code Refactoring. Also proposed an environment setup to evaluate the modified Snort Tool performance in future

SNORT

- Created in 1998
- Founder: Martin Roesch
- Free open source tool
- Developer: CISCO systems
- Version: Snort 2.9.19
- Modes:-

1. Sniffer Mode
2. Packet Logger Mode
3. Intrusion Detection Mode

Components of Snort tool

Components of Snort Snort is logically divided into multiple components. These components work together to detect particular attacks and to generate output in a required format from the detection system.

A Snort-based IDS consists of the following major components:

- Packet Decoder
- Preprocessors
- Detection Engine
- Logging and Alerting System
- Output Modules

1) Packet Decoder:

The packet decoder takes packets from different types of network interfaces and prepares the packets to be preprocessed or to be sent to the detection engine. The interfaces may be Ethernet, SLIP, PPP and so on.

2) Preprocessors:

Preprocessors are components or plug-ins that can be used with Snort to arrange or modify data packets before the detection engine does some operation to find out if the packet is being used by an intruder. Some preprocessors also perform detection by finding anomalies in packet headers and generating alerts. Preprocessors are very important for any IDS to prepare data packets to be analyzed against rules in the detection engine. Hackers use different techniques to fool an IDS in different ways. For example, you may have created a rule to find a signature “scripts/iisadmin” in HTTP packets. If you are matching this string exactly, you can easily be fooled by a hacker who makes slight modifications to this string.

3) The Detection Engine:

The detection engine is the most important part of Snort. Its responsibility is to detect if any intrusion activity exists in a packet. The detection engine employs Snort rules for this purpose. The rules are read into internal data structures or chains where they are matched against all packets. If a packet matches any rule, appropriate action is taken; otherwise the packet is dropped. Appropriate actions may be logging the packet or generating alerts. The detection engine is the time-critical part of Snort. Depending upon how powerful your machine is and how many rules you have defined, it may take different amounts of time to respond to different packets. If traffic on your network is too high when Snort is working in NIDS mode, you may drop some packets and may not get a true real-time response. The load on the detection engine depends upon the following factors: • Number of rules • Power of the machine on which Snort is running • Speed of internal bus used in the Snort machine • Load on the network

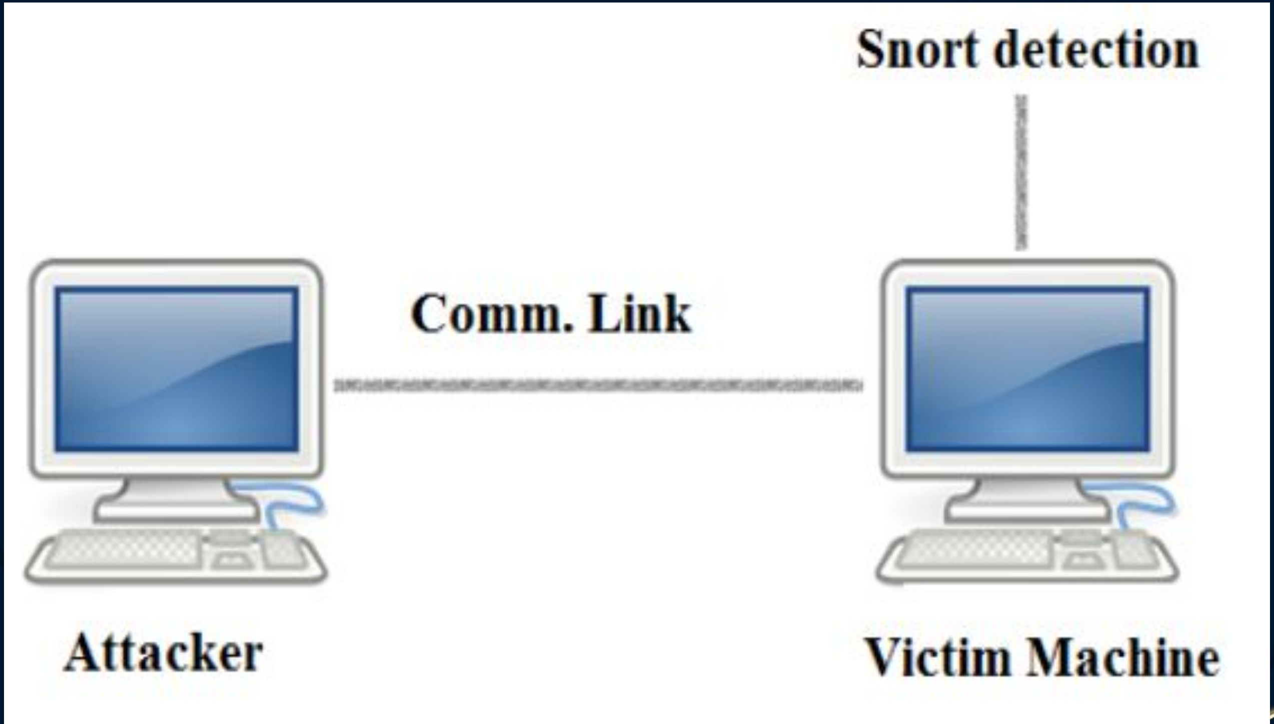
4) Logging and Alerting System:

Depending upon what the detection engine finds inside a packet, the packet may be used to log the activity or generate an alert. Logs are kept in simple text files, tcpdump-style files or some other form. All of the log files are stored under /var/log/snort folder by default. You can use –l command line options to modify the location of generating logs and alerts. Many command line options discussed in the next chapter can modify the type and detail of information that is logged by the logging and alerting system.

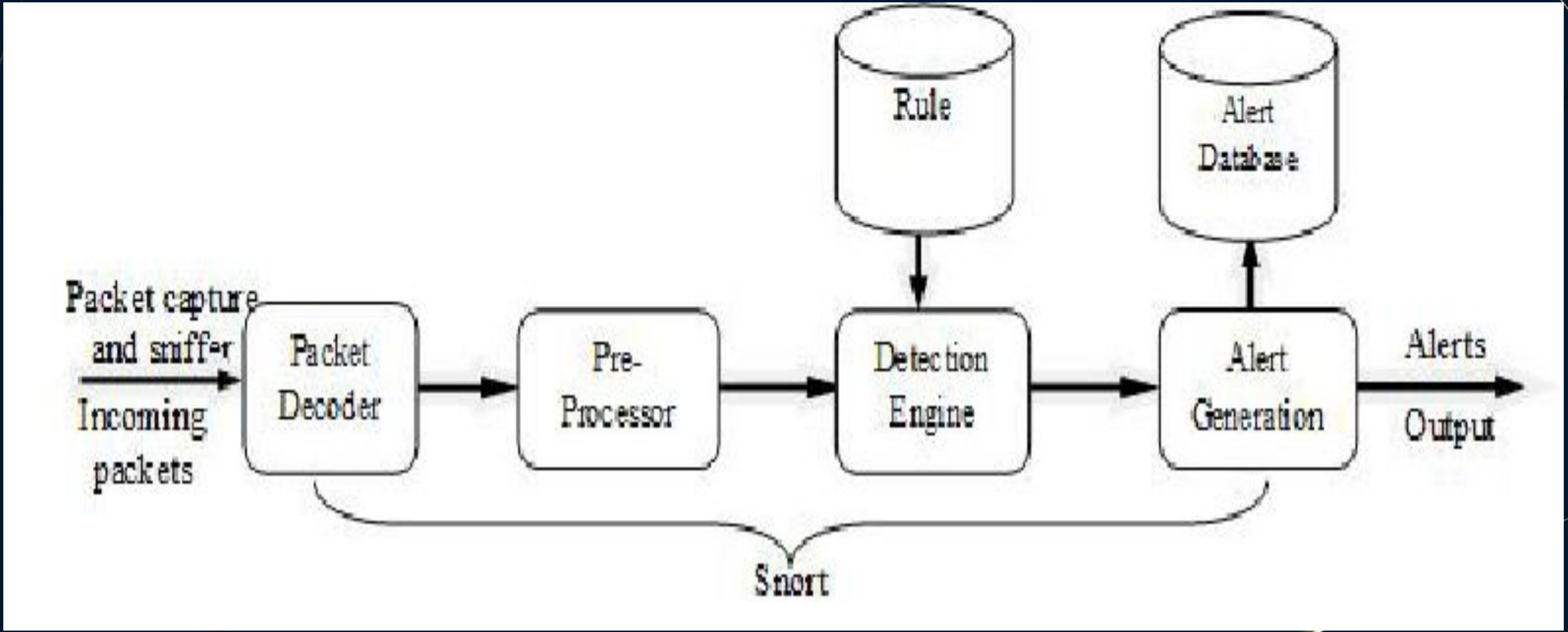
5) Output Module:

Plug-ins can do different operations depending on how you want to save output generated by the logging and alerting system of Snort. Basically these modules control the type of output generated by the logging and alerting system.

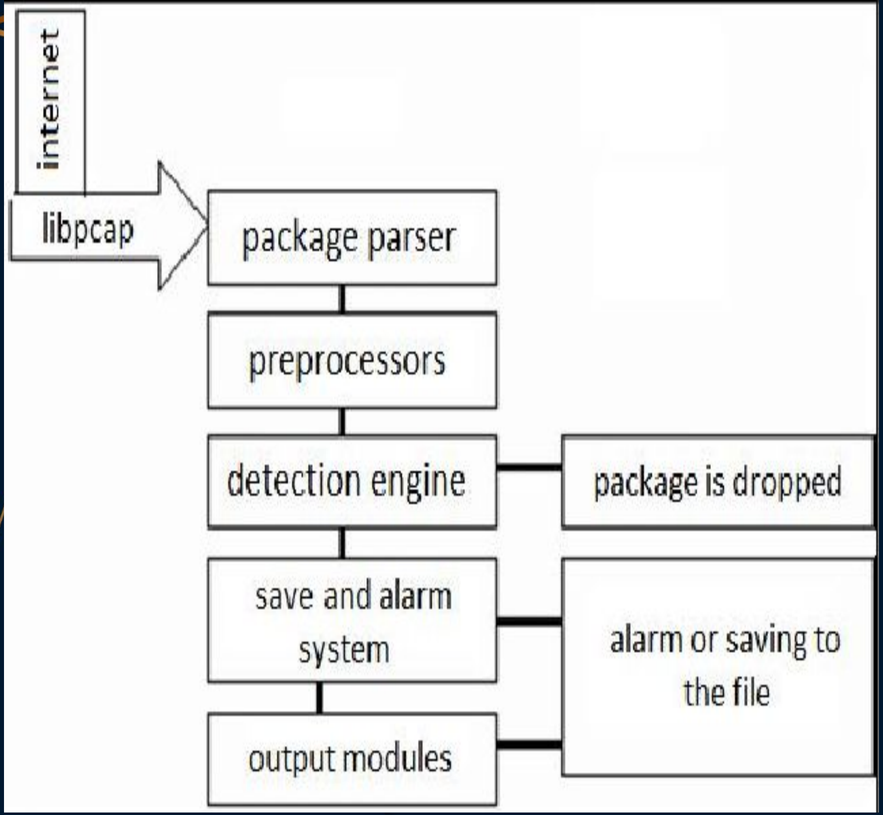
Architecture Diagram

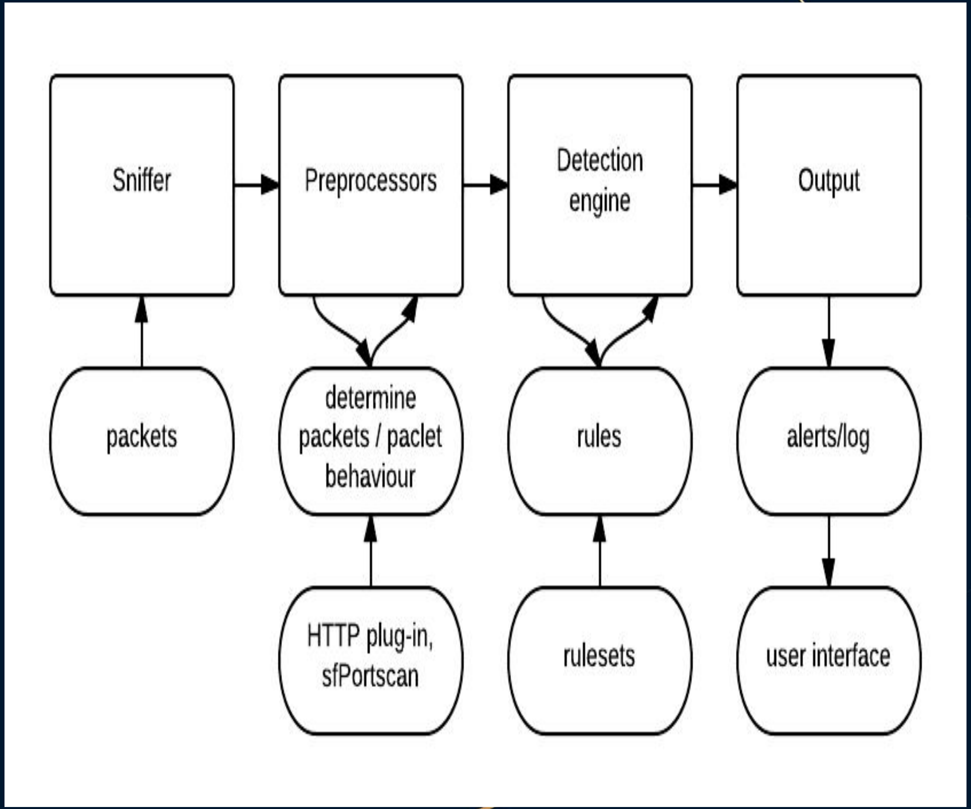


Architecture of Snort IDS



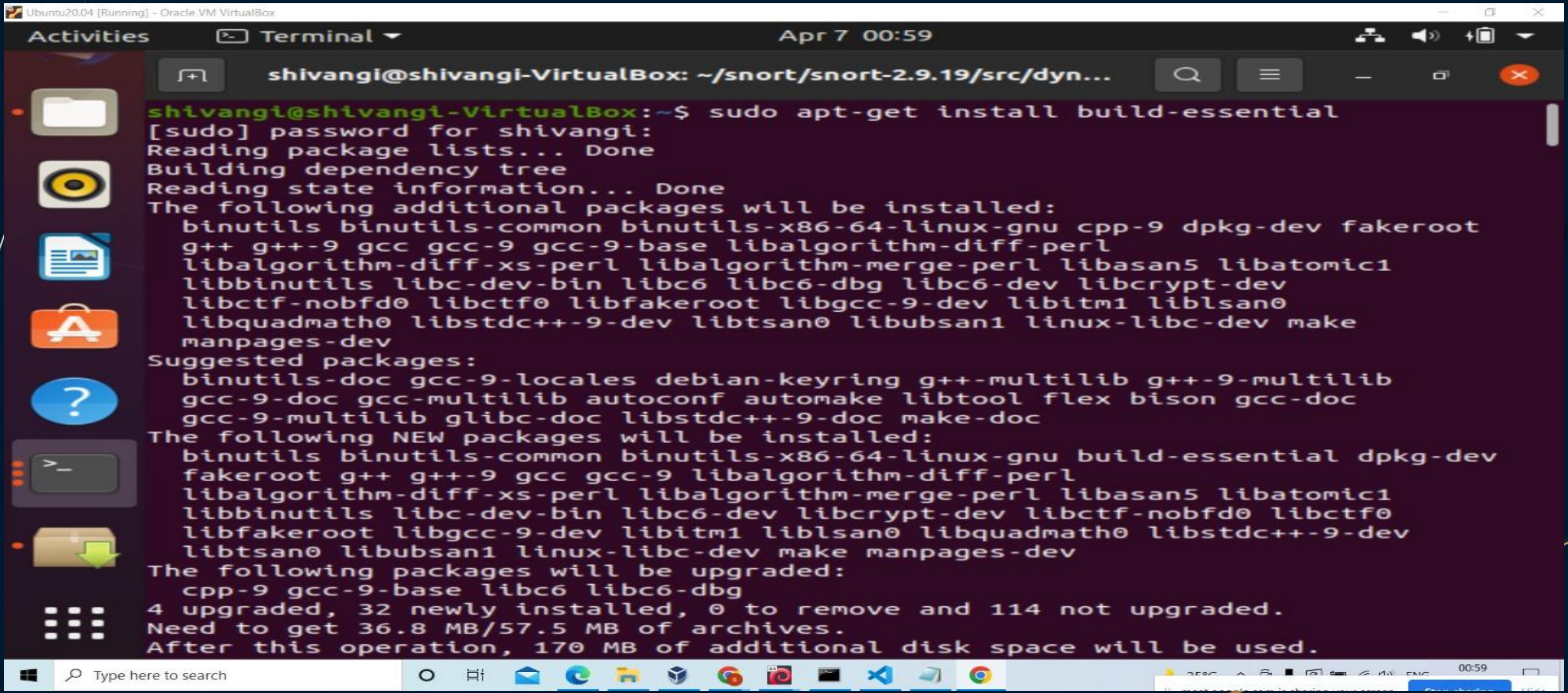
High level Architecture





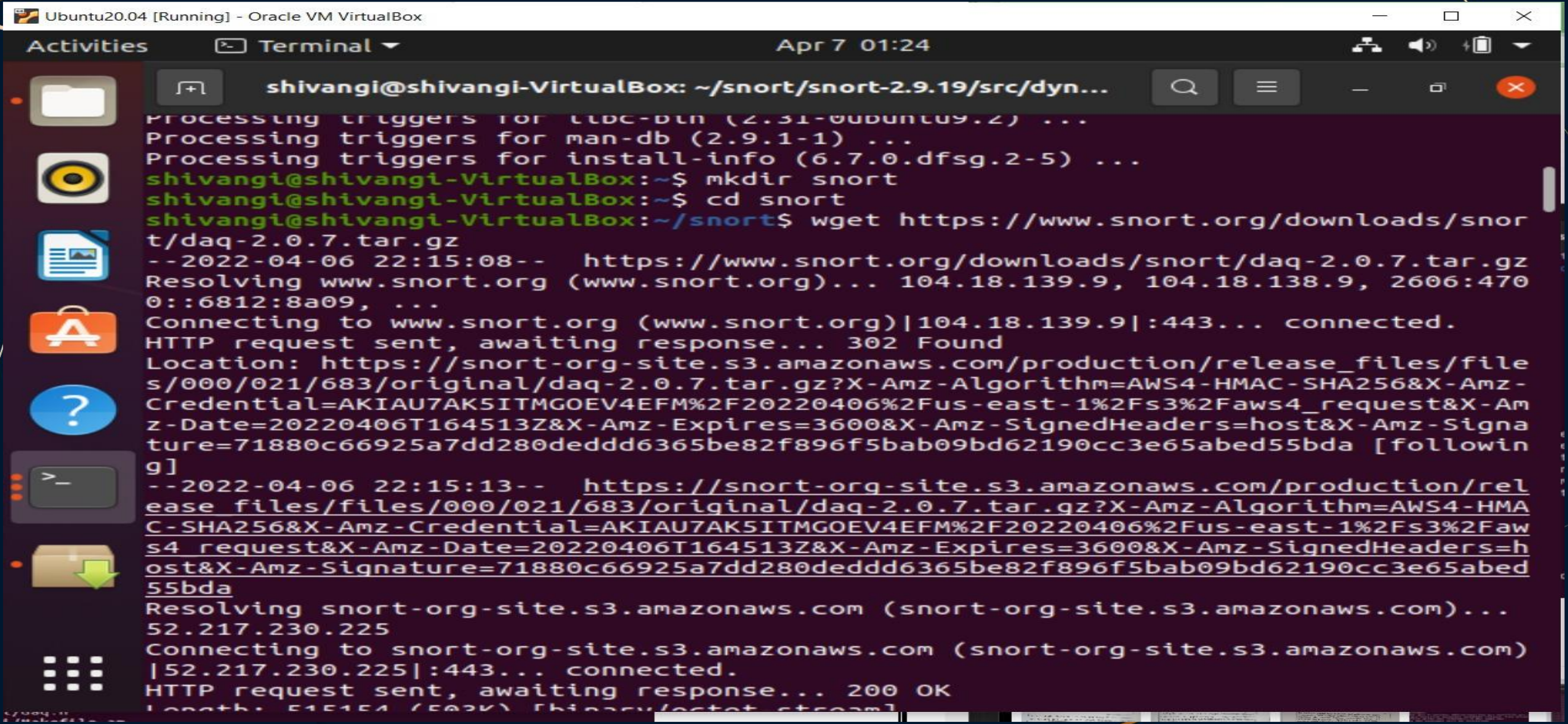
Implementation:

Installation & setting up of environment before installation of Snort


Ubuntu20.04 [Running] - Oracle VM VirtualBox
Activities
Terminal
Apr 7 00:59
shivangi@shivangi-VirtualBox: ~/snort/snort-2.9.19/src/dyn...
Type here to search
00:59

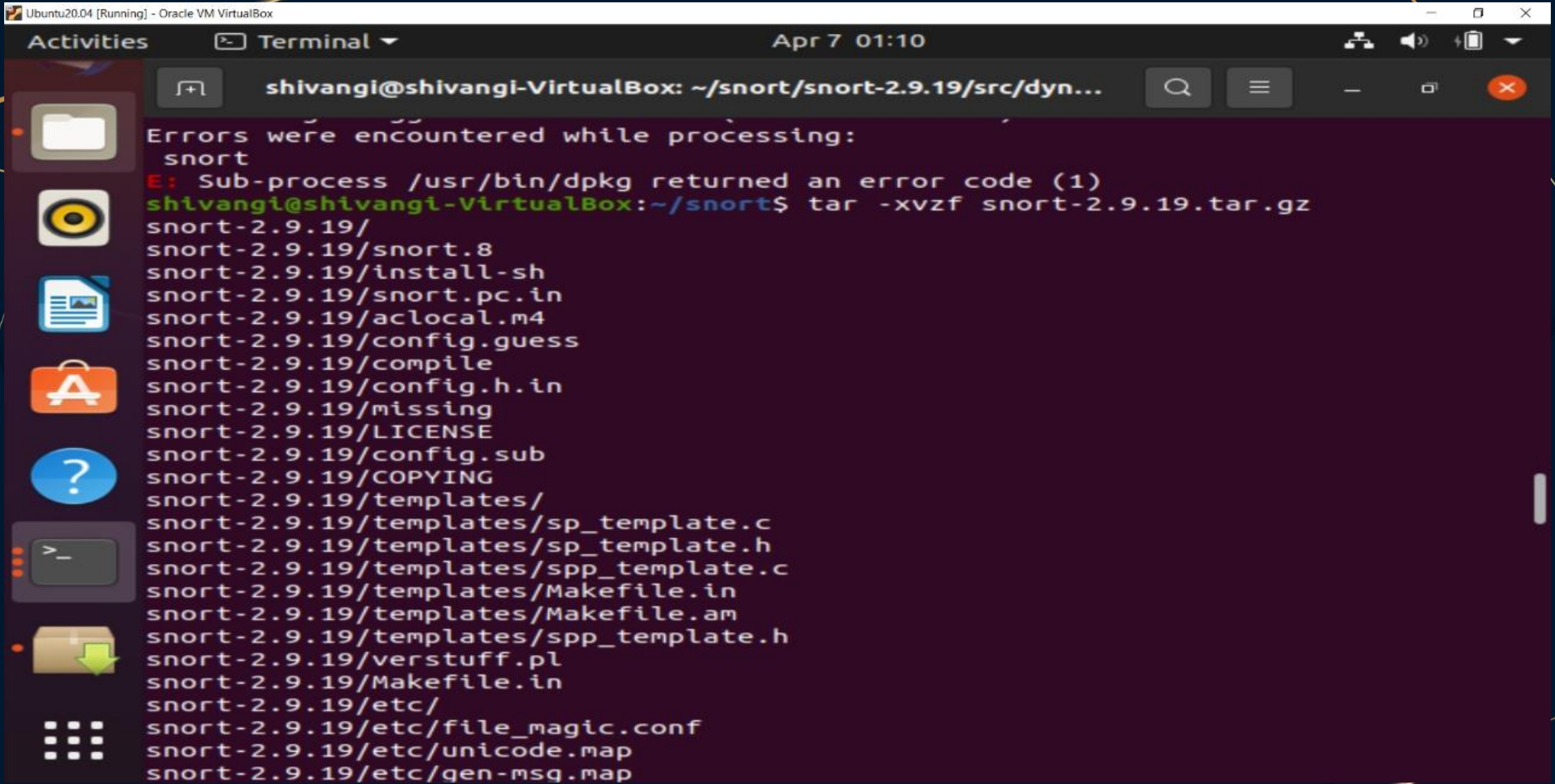
```
shivangi@shivangi-VirtualBox:~$ sudo apt-get install build-essential
[sudo] password for shivangi:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu cpp-9 dpkg-dev fakeroot
  g++ g++-9 gcc gcc-9 gcc-9-base libalgorithm-diff-perl
  libalgorithm-diff-xs-perl libalgorithm-merge-perl libasan5 libatomic1
  libbinutils libc-dev-bin libc6 libc6-dbg libc6-dev libcrypt-dev
  libctf-nobfd0 libctf0 libfakeroot libgcc-9-dev libitm1 liblsan0
  libquadmath0 libstdc++-9-dev libtsan0 libubsan1 linux-libc-dev make
  manpages-dev
Suggested packages:
  binutils-doc gcc-9-locales debian-keyring g++-multilib g++-9-multilib
  gcc-9-doc gcc-multilib autoconf automake libtool flex bison gcc-doc
  gcc-9-multilib glibc-doc libstdc++-9-doc make-doc
The following NEW packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu build-essential dpkg-dev
  fakeroot g++ g++-9 gcc gcc-9 libalgorithm-diff-perl
  libalgorithm-diff-xs-perl libalgorithm-merge-perl libasan5 libatomic1
  libbinutils libc-dev-bin libc6-dev libcrypt-dev libctf-nobfd0 libctf0
  libfakeroot libgcc-9-dev libitm1 liblsan0 libquadmath0 libstdc++-9-dev
  libtsan0 libubsan1 linux-libc-dev make manpages-dev
The following packages will be upgraded:
  cpp-9 gcc-9-base libc6 libc6-dbg
4 upgraded, 32 newly installed, 0 to remove and 114 not upgraded.
Need to get 36.8 MB/57.5 MB of archives.
After this operation, 170 MB of additional disk space will be used.
```

Installation of Snort 2.9.19



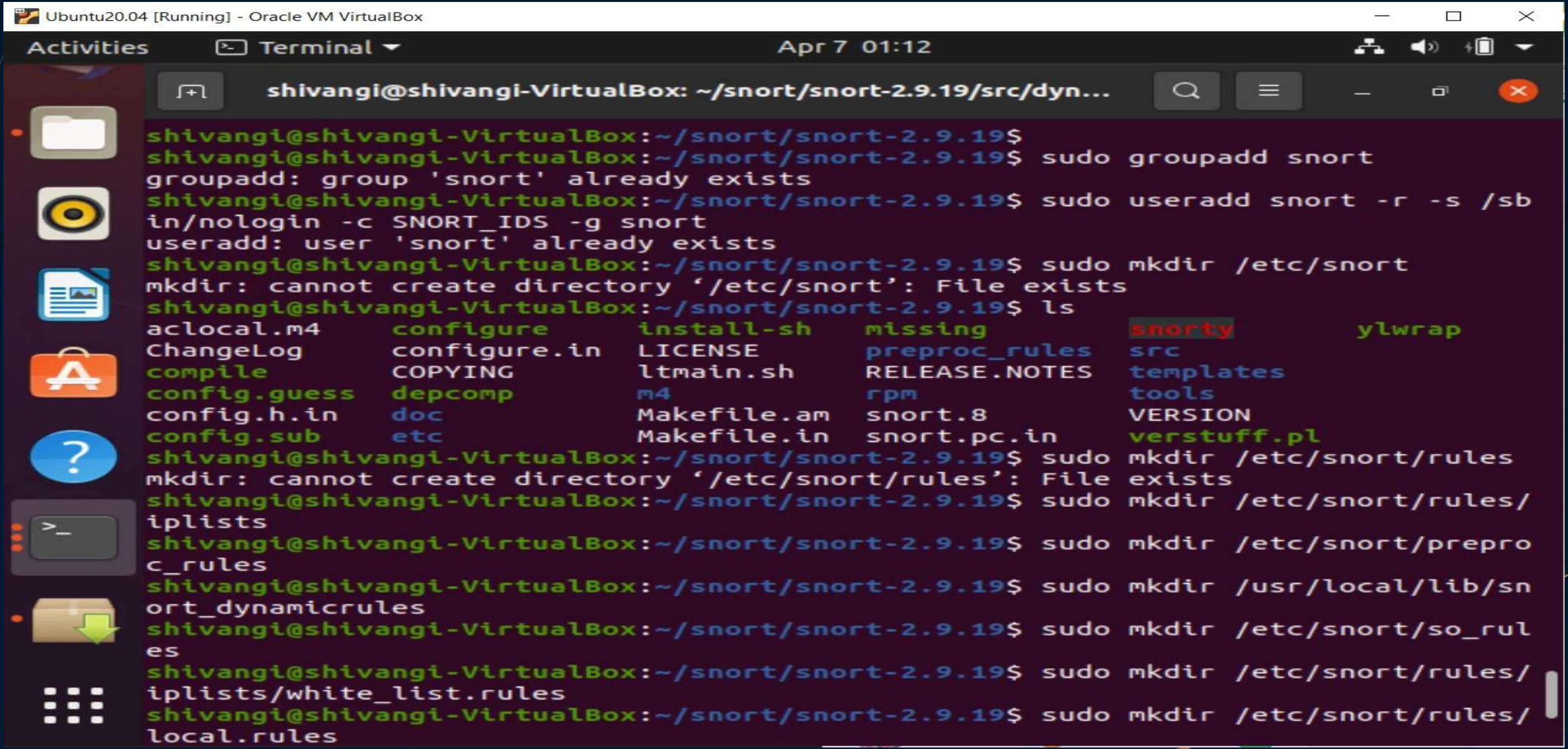
Ubuntu20.04 [Running] - Oracle VM VirtualBox
Activities
Terminal
Apr 7 01:24
shivangi@shivangi-VirtualBox: ~/snort/snort-2.9.19/src/dyn...

```
Processing triggers for libc-bin (2.31-0ubuntu9.2) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for install-info (6.7.0.dfsg.2-5) ...
shivangi@shivangi-VirtualBox:~$ mkdir snort
shivangi@shivangi-VirtualBox:~$ cd snort
shivangi@shivangi-VirtualBox:~/snort$ wget https://www.snort.org/downloads/snor
t/daq-2.0.7.tar.gz
--2022-04-06 22:15:08--  https://www.snort.org/downloads/snort/daq-2.0.7.tar.gz
Resolving www.snort.org (www.snort.org)... 104.18.139.9, 104.18.138.9, 2606:470
0::6812:8a09, ...
Connecting to www.snort.org (www.snort.org)|104.18.139.9|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://snort-org-site.s3.amazonaws.com/production/release_files/file
s/000/021/683/original/daq-2.0.7.tar.gz?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-
Credential=AKIAU7AK5ITMGOEV4EFM%2F20220406%2Fus-east-1%2Fs3%2Faws4_request&X-Am
z-Date=20220406T164513Z&X-Amz-Expires=3600&X-Amz-SignedHeaders=host&X-Amz-Signa
ture=71880c66925a7dd280deddd6365be82f896f5bab09bd62190cc3e65abed55bda [followin
g]
--2022-04-06 22:15:13--  https://snort-org-site.s3.amazonaws.com/production/rel
ease_files/files/000/021/683/original/daq-2.0.7.tar.gz?X-Amz-Algorithm=AWS4-HMA
C-SHA256&X-Amz-Credential=AKIAU7AK5ITMGOEV4EFM%2F20220406%2Fus-east-1%2Fs3%2Faw
s4_request&X-Amz-Date=20220406T164513Z&X-Amz-Expires=3600&X-Amz-SignedHeaders=h
ost&X-Amz-Signature=71880c66925a7dd280deddd6365be82f896f5bab09bd62190cc3e65abed
55bda
Resolving snort-org-site.s3.amazonaws.com (snort-org-site.s3.amazonaws.com)...
52.217.230.225
Connecting to snort-org-site.s3.amazonaws.com (snort-org-site.s3.amazonaws.com)
|52.217.230.225|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 515154 (503K) [binary/octet-stream]
```


Ubuntu20.04 [Running] - Oracle VM VirtualBox
Activities
Terminal
Apr 7 01:10
shivangi@shivangi-VirtualBox: ~/snort/snort-2.9.19/src/dyn...

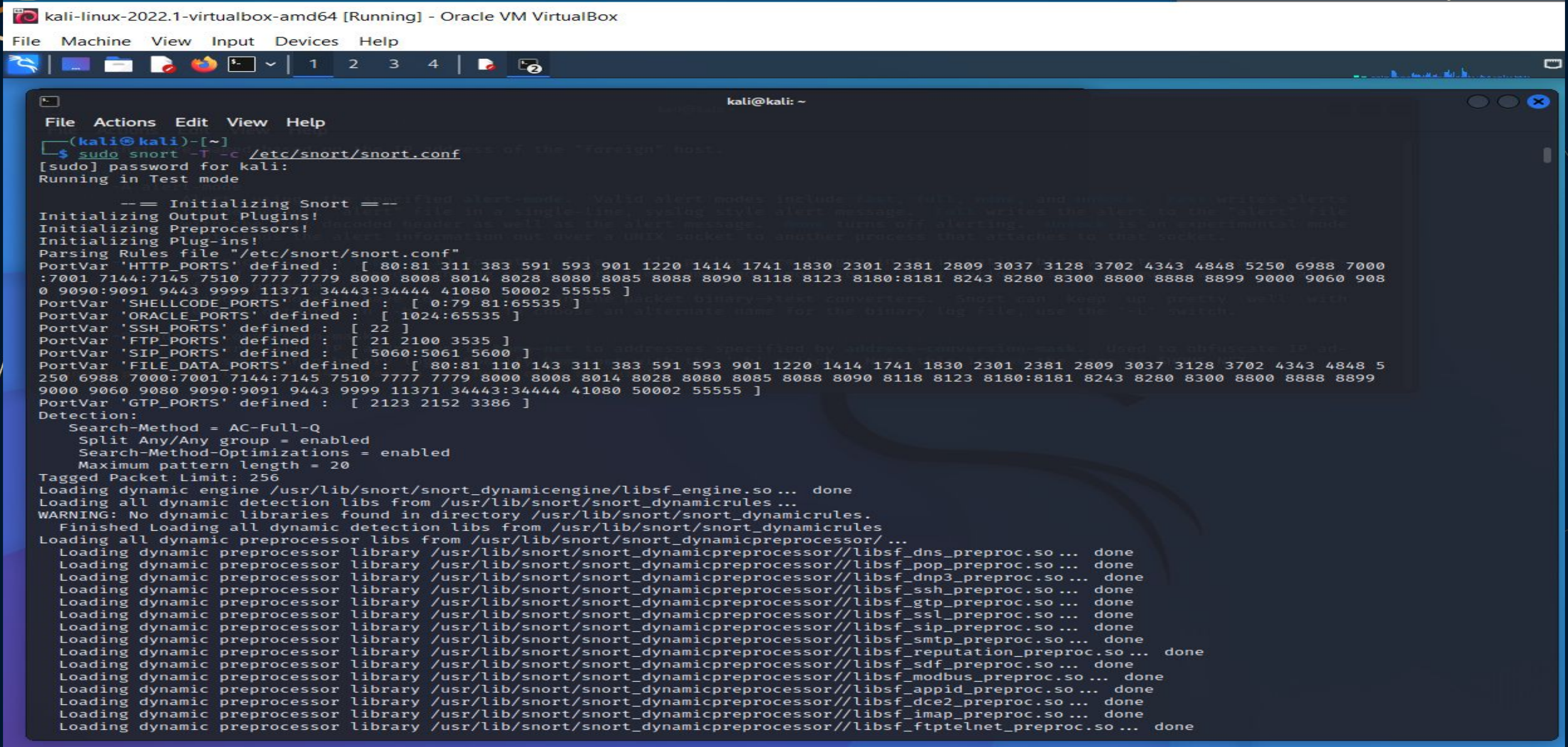
```
Errors were encountered while processing:
 snort
E: Sub-process /usr/bin/dpkg returned an error code (1)
shivangi@shivangi-VirtualBox:~/snort$ tar -xvzf snort-2.9.19.tar.gz
snort-2.9.19/
snort-2.9.19/snort.8
snort-2.9.19/install-sh
snort-2.9.19/snort.pc.in
snort-2.9.19/aclocal.m4
snort-2.9.19/config.guess
snort-2.9.19/compile
snort-2.9.19/config.h.in
snort-2.9.19/missing
snort-2.9.19/LICENSE
snort-2.9.19/config.sub
snort-2.9.19/COPYING
snort-2.9.19/templates/
snort-2.9.19/templates/sp_template.c
snort-2.9.19/templates/sp_template.h
snort-2.9.19/templates/spp_template.c
snort-2.9.19/templates/Makefile.in
snort-2.9.19/templates/Makefile.am
snort-2.9.19/templates/spp_template.h
snort-2.9.19/verstuff.pl
snort-2.9.19/Makefile.in
snort-2.9.19/etc/
snort-2.9.19/etc/file_magic.conf
snort-2.9.19/etc/unicode.map
snort-2.9.19/etc/gen-msg.map
```

Setting up of necessary directories & files


Ubuntu20.04 [Running] - Oracle VM VirtualBox
Activities
Terminal
Apr 7 01:12
shivangi@shivangi-VirtualBox: ~/snort/snort-2.9.19/src/dyn...

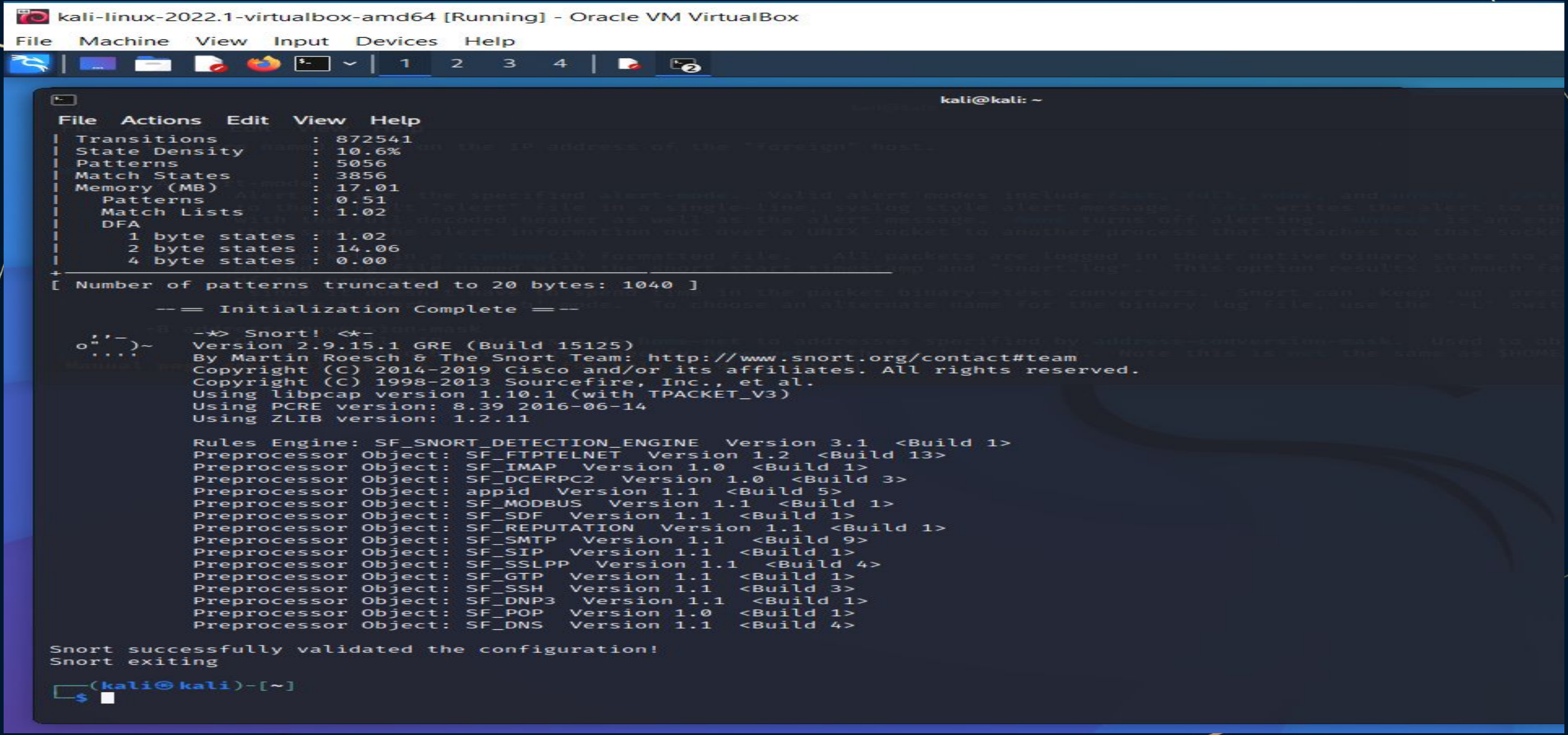
```
shivangi@shivangi-VirtualBox:~/snort/snort-2.9.19$
shivangi@shivangi-VirtualBox:~/snort/snort-2.9.19$ sudo groupadd snort
groupadd: group 'snort' already exists
shivangi@shivangi-VirtualBox:~/snort/snort-2.9.19$ sudo useradd snort -r -s /sb
in/nologin -c SNORT_IDS -g snort
useradd: user 'snort' already exists
shivangi@shivangi-VirtualBox:~/snort/snort-2.9.19$ sudo mkdir /etc/snort
mkdir: cannot create directory ‘/etc/snort’: File exists
shivangi@shivangi-VirtualBox:~/snort/snort-2.9.19$ ls
aclocal.m4    configure     install-sh   missing        snorty       ylwrap
ChangeLog     configure.in  LICENSE      preproc_rules  src
compile       COPYING       ltmain.sh    RELEASE.NOTES  templates
config.guess  depcomp       m4           rpm            tools
config.h.in   doc           Makefile.am  snort.8        VERSION
config.sub    etc           Makefile.in  snort.pc.in    verstuff.pl
shivangi@shivangi-VirtualBox:~/snort/snort-2.9.19$ sudo mkdir /etc/snort/rules
mkdir: cannot create directory ‘/etc/snort/rules’: File exists
shivangi@shivangi-VirtualBox:~/snort/snort-2.9.19$ sudo mkdir /etc/snort/rules/
iplists
shivangi@shivangi-VirtualBox:~/snort/snort-2.9.19$ sudo mkdir /etc/snort/prepro
c_rules
shivangi@shivangi-VirtualBox:~/snort/snort-2.9.19$ sudo mkdir /usr/local/lib/sn
ort_dynamicrules
shivangi@shivangi-VirtualBox:~/snort/snort-2.9.19$ sudo mkdir /etc/snort/so_rul
es
shivangi@shivangi-VirtualBox:~/snort/snort-2.9.19$ sudo mkdir /etc/snort/rules/
iplists/white_list.rules
shivangi@shivangi-VirtualBox:~/snort/snort-2.9.19$ sudo mkdir /etc/snort/rules/
local.rules
```

Validation configuration of Snort


kali-linux-2022.1-virtualbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
kali@kali: ~
File Actions Edit View Help

```
┌──(kali㉿kali)-[~]
└─$ sudo snort -T -c /etc/snort/snort.conf
[sudo] password for kali:
Running in Test 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 908
0 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 1414 1741 1830 2301 2381 2809 3037 3128 3702 4343 4848 5
250 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 'GTP_PORTS' defined :  [ 2123 2152 3386 ]
Detection:
   Search-Method = AC-Full-Q
    Split Any/Any group = enabled
    Search-Method-Optimizations = enabled
    Maximum pattern length = 20
Tagged Packet Limit: 256
Loading dynamic engine /usr/lib/snort/snort_dynamicengine/libsf_engine.so...  done
Loading all dynamic detection libs from /usr/lib/snort/snort_dynamicrules...
WARNING: No dynamic libraries found in directory /usr/lib/snort/snort_dynamicrules.
  Finished Loading all dynamic detection libs from /usr/lib/snort/snort_dynamicrules
Loading all dynamic preprocessor libs from /usr/lib/snort/snort_dynamicpreprocessor/...
  Loading dynamic preprocessor library /usr/lib/snort/snort_dynamicpreprocessor//libsf_dns_preproc.so...  done
  Loading dynamic preprocessor library /usr/lib/snort/snort_dynamicpreprocessor//libsf_pop_preproc.so...  done
  Loading dynamic preprocessor library /usr/lib/snort/snort_dynamicpreprocessor//libsf_dnp3_preproc.so...  done
  Loading dynamic preprocessor library /usr/lib/snort/snort_dynamicpreprocessor//libsf_ssh_preproc.so...  done
  Loading dynamic preprocessor library /usr/lib/snort/snort_dynamicpreprocessor//libsf_gtp_preproc.so...  done
  Loading dynamic preprocessor library /usr/lib/snort/snort_dynamicpreprocessor//libsf_ssl_preproc.so...  done
  Loading dynamic preprocessor library /usr/lib/snort/snort_dynamicpreprocessor//libsf_sip_preproc.so...  done
  Loading dynamic preprocessor library /usr/lib/snort/snort_dynamicpreprocessor//libsf_smtp_preproc.so...  done
  Loading dynamic preprocessor library /usr/lib/snort/snort_dynamicpreprocessor//libsf_reputation_preproc.so...  done
  Loading dynamic preprocessor library /usr/lib/snort/snort_dynamicpreprocessor//libsf_sdf_preproc.so...  done
  Loading dynamic preprocessor library /usr/lib/snort/snort_dynamicpreprocessor//libsf_modbus_preproc.so...  done
  Loading dynamic preprocessor library /usr/lib/snort/snort_dynamicpreprocessor//libsf_appid_preproc.so...  done
  Loading dynamic preprocessor library /usr/lib/snort/snort_dynamicpreprocessor//libsf_dce2_preproc.so...  done
  Loading dynamic preprocessor library /usr/lib/snort/snort_dynamicpreprocessor//libsf_imap_preproc.so...  done
  Loading dynamic preprocessor library /usr/lib/snort/snort_dynamicpreprocessor//libsf_ftptelnet_preproc.so...  done
```

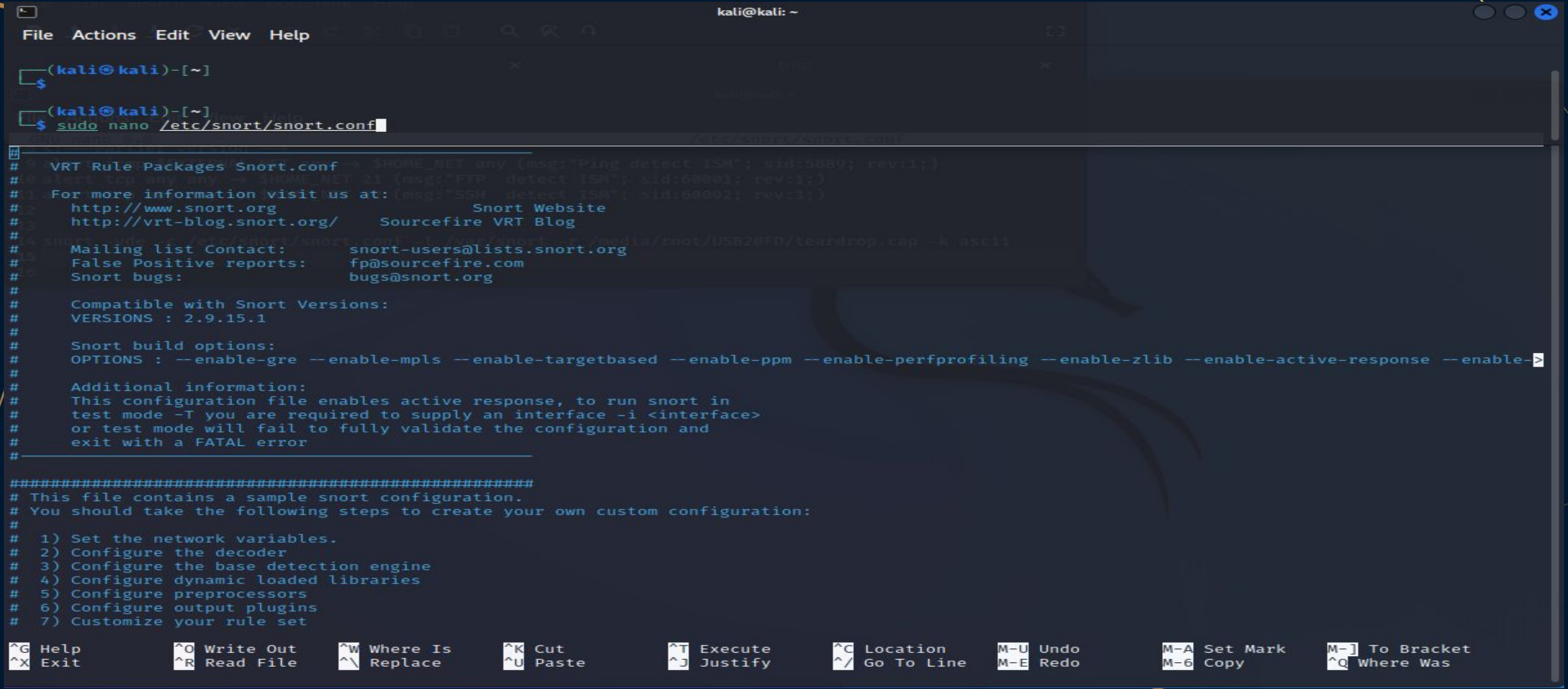
Snort successfully validated the configuration



kali-linux-2022.1-virtualbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
kali@kali: ~
File Actions Edit View Help

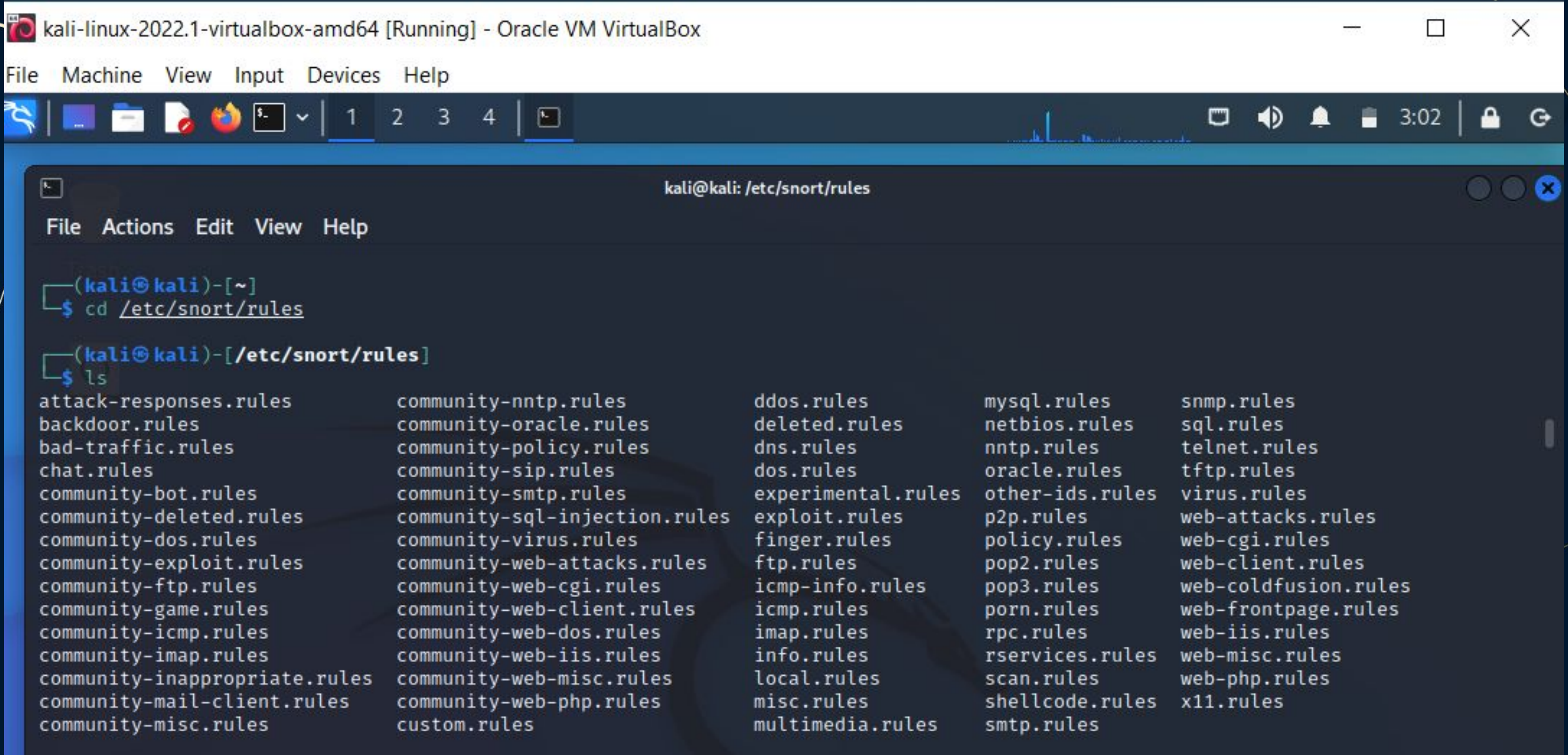
```
| Transitions          : 872541
| State Density        : 10.6%
| Patterns             : 5056
| Match States         : 3856
| Memory (MB)          : 17.01
|   Patterns           : 0.51
|   Match Lists        : 1.02
|   DFA
|     1 byte states : 1.02
|     2 byte states : 14.06
|     4 byte states : 0.00
+----------------------------------------------------
[ Number of patterns truncated to 20 bytes: 1040 ]

        --== Initialization Complete ==--

   ,,_     -*> Snort! <*-
  o"  )~   Version 2.9.15.1 GRE (Build 15125)
   ''''    By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
           Copyright (C) 2014-2019 Cisco and/or its affiliates. All rights reserved.
           Copyright (C) 1998-2013 Sourcefire, Inc., et al.
           Using libpcap version 1.10.1 (with TPACKET_V3)
           Using PCRE version: 8.39 2016-06-14
           Using ZLIB version: 1.2.11

           Rules Engine: SF_SNORT_DETECTION_ENGINE  Version 3.1  <Build 1>
           Preprocessor Object: SF_FTPTELNET  Version 1.2  <Build 13>
           Preprocessor Object: SF_IMAP  Version 1.0  <Build 1>
           Preprocessor Object: SF_DCERPC2  Version 1.0  <Build 3>
           Preprocessor Object: appid  Version 1.1  <Build 5>
           Preprocessor Object: SF_MODBUS  Version 1.1  <Build 1>
           Preprocessor Object: SF_SDF  Version 1.1  <Build 1>
           Preprocessor Object: SF_REPUTATION  Version 1.1  <Build 1>
           Preprocessor Object: SF_SMTP  Version 1.1  <Build 9>
           Preprocessor Object: SF_SIP  Version 1.1  <Build 1>
           Preprocessor Object: SF_SSLPP  Version 1.1  <Build 4>
           Preprocessor Object: SF_GTP  Version 1.1  <Build 1>
           Preprocessor Object: SF_SSH  Version 1.1  <Build 3>
           Preprocessor Object: SF_DNP3  Version 1.1  <Build 1>
           Preprocessor Object: SF_POP  Version 1.0  <Build 1>
           Preprocessor Object: SF_DNS  Version 1.1  <Build 4>

Snort successfully validated the configuration!
Snort exiting

┌──(kali㉿kali)-[~]
└─$ 
```

Configuration of Settings file(snort.conf)



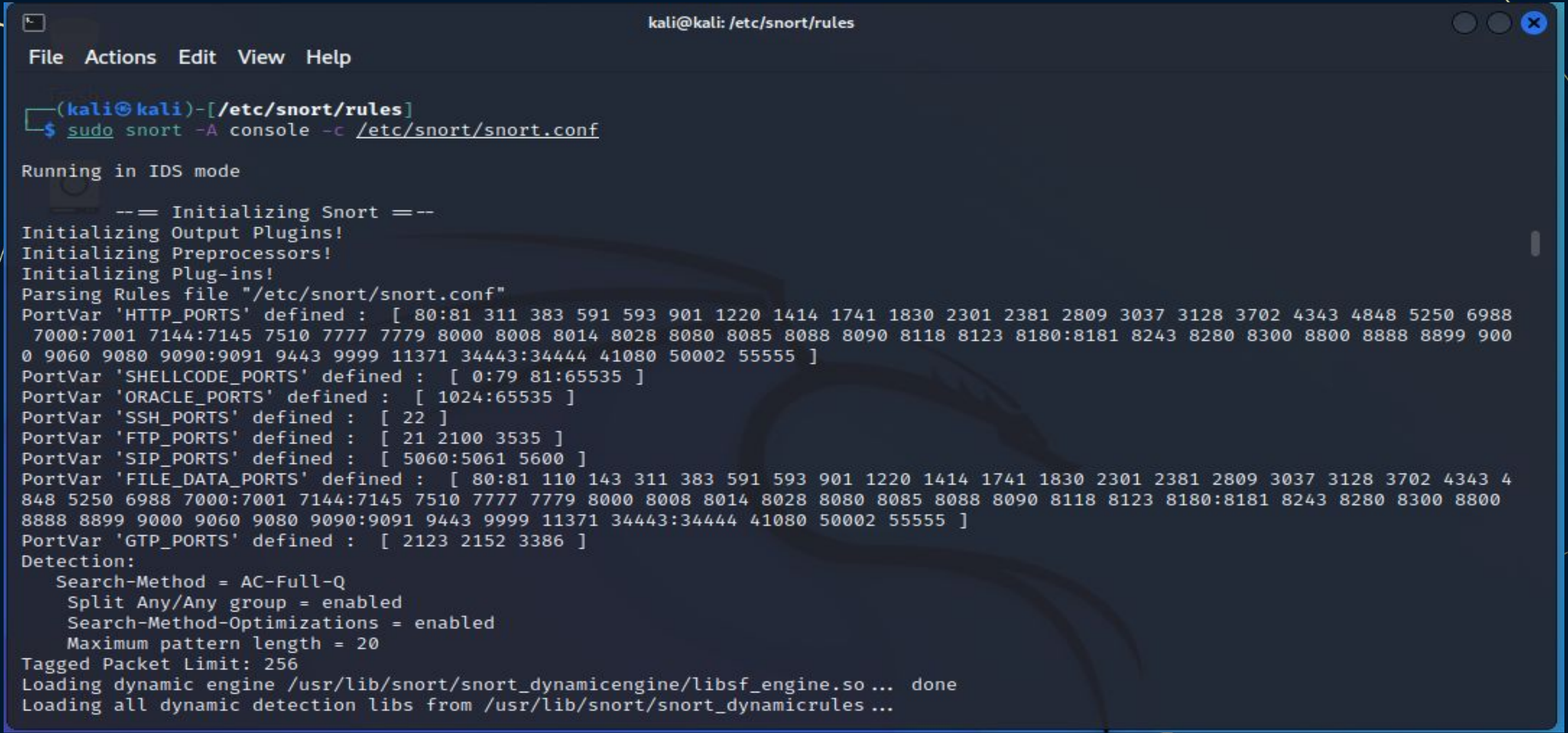
```
┌──(kali㉿kali)-[~]
└─$

┌──(kali㉿kali)-[~]
└─$ sudo nano /etc/snort/snort.conf
#--------------------------------------------------
#   VRT Rule Packages Snort.conf
#
#   For more information visit us at:
#     http://www.snort.org                   Snort Website
#     http://vrt-blog.snort.org/    Sourcefire VRT Blog
#
#     Mailing list Contact:      snort-users@lists.snort.org
#     False Positive reports:    fp@sourcefire.com
#     Snort bugs:                bugs@snort.org
#
#     Compatible with Snort Versions:
#     VERSIONS : 2.9.15.1
#
#     Snort build options:
#     OPTIONS : --enable-gre --enable-mpls --enable-targetbased --enable-ppm --enable-perfprofiling --enable-zlib --enable-active-response --enable->
#
#     Additional information:
#     This configuration file enables active response, to run snort in
#     test mode -T you are required to supply an interface -i <interface>
#     or test mode will fail to fully validate the configuration and
#     exit with a FATAL error
#--------------------------------------------------

###################################################
# This file contains a sample snort configuration.
# You should take the following steps to create your own custom configuration:
#
#  1) Set the network variables.
#  2) Configure the decoder
#  3) Configure the base detection engine
#  4) Configure dynamic loaded libraries
#  5) Configure preprocessors
#  6) Configure output plugins
#  7) Customize your rule set
^G Help          ^O Write Out     ^W Where Is      ^K Cut           ^T Execute       ^C Location      M-U Undo         M-A Set Mark     M-] To Bracket
^X Exit          ^R Read File     ^\ Replace       ^U Paste         ^J Justify       ^/ Go To Line    M-E Redo         M-6 Copy         ^Q Where Was
```

Analysing the rules file


kali-linux-2022.1-virtualbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
3:02
kali@kali: /etc/snort/rules
File Actions Edit View Help

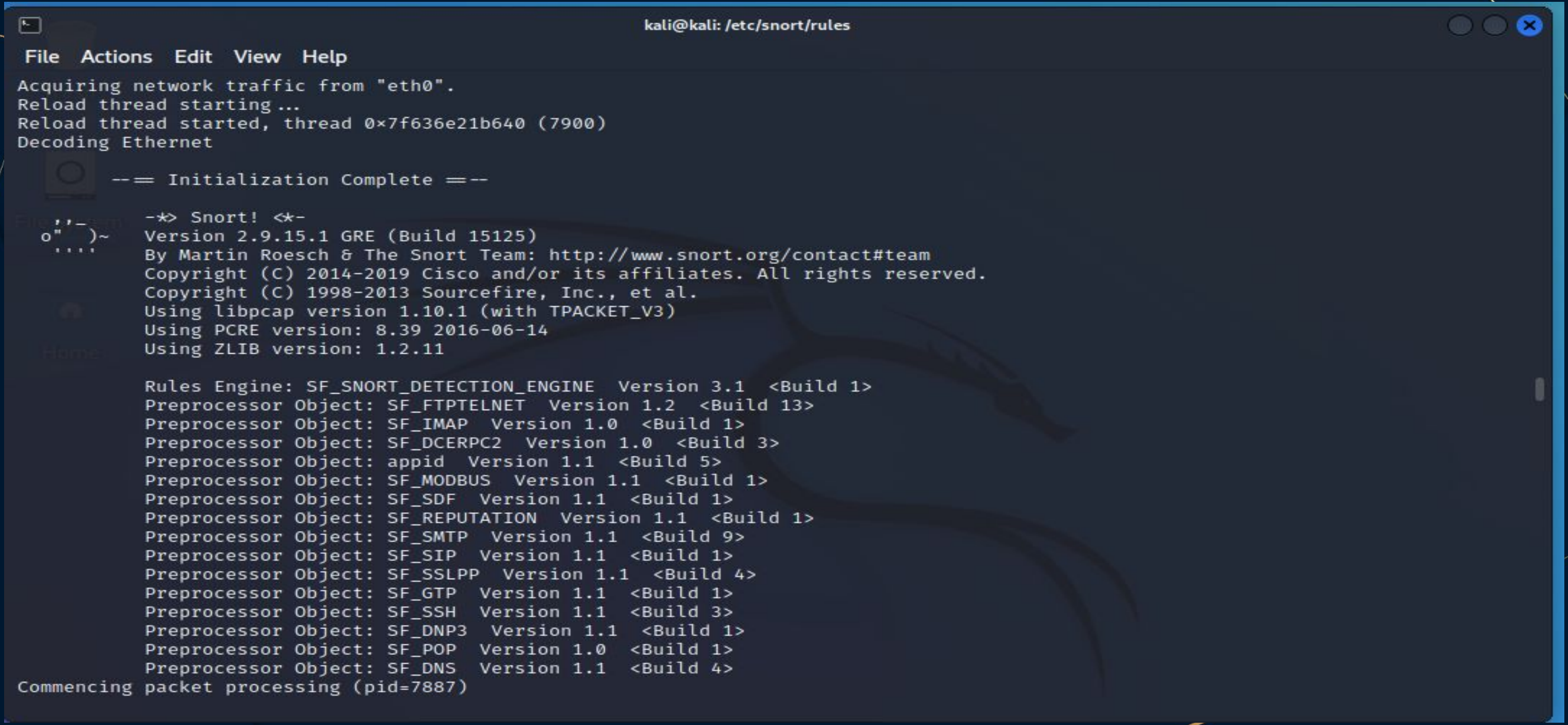
```
┌──(kali㉿kali)-[~]
└─$ cd /etc/snort/rules

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

Running Snort in IDS mode



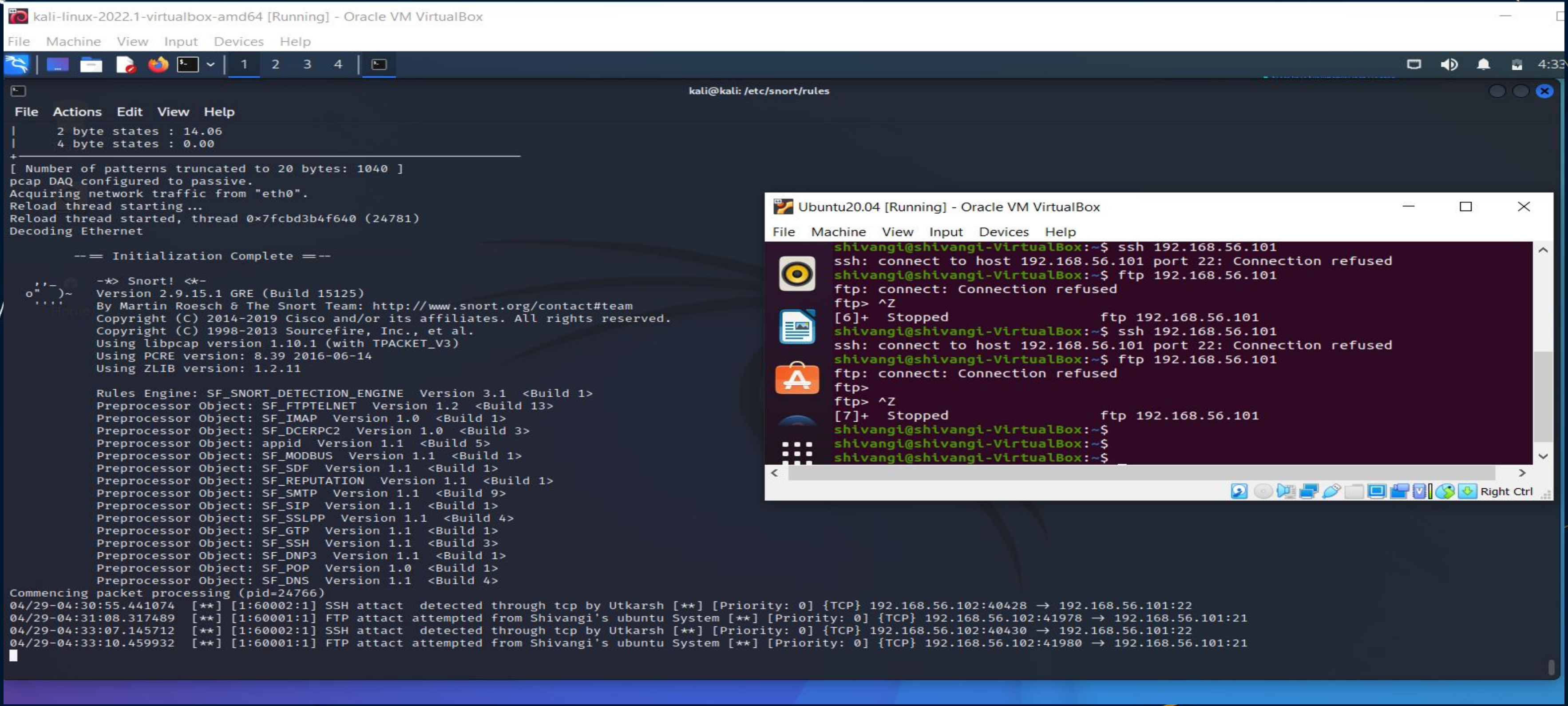
```
┌──(kali㉿kali)-[/etc/snort/rules]
└─$ sudo snort -A console -c /etc/snort/snort.conf

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 900
0 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 1414 1741 1830 2301 2381 2809 3037 3128 3702 4343 4
848 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 'GTP_PORTS' defined :  [ 2123 2152 3386 ]
Detection:
   Search-Method = AC-Full-Q
    Split Any/Any group = enabled
    Search-Method-Optimizations = enabled
    Maximum pattern length = 20
Tagged Packet Limit: 256
Loading dynamic engine /usr/lib/snort/snort_dynamicengine/libsf_engine.so ...  done
Loading all dynamic detection libs from /usr/lib/snort/snort_dynamicrules ...
```

Snort tool ready to detect attacks



```
Acquiring network traffic from "eth0".
Reload thread starting ...
Reload thread started, thread 0x7f636e21b640 (7900)
Decoding Ethernet

        --== Initialization Complete ==--

   ,,_     -*> Snort! <*-
  o"  )~   Version 2.9.15.1 GRE (Build 15125)
   ''''    By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
           Copyright (C) 2014-2019 Cisco and/or its affiliates. All rights reserved.
           Copyright (C) 1998-2013 Sourcefire, Inc., et al.
           Using libpcap version 1.10.1 (with TPACKET_V3)
           Using PCRE version: 8.39 2016-06-14
           Using ZLIB version: 1.2.11

           Rules Engine: SF_SNORT_DETECTION_ENGINE  Version 3.1  <Build 1>
           Preprocessor Object: SF_FTPTELNET  Version 1.2  <Build 13>
           Preprocessor Object: SF_IMAP  Version 1.0  <Build 1>
           Preprocessor Object: SF_DCERPC2  Version 1.0  <Build 3>
           Preprocessor Object: appid  Version 1.1  <Build 5>
           Preprocessor Object: SF_MODBUS  Version 1.1  <Build 1>
           Preprocessor Object: SF_SDF  Version 1.1  <Build 1>
           Preprocessor Object: SF_REPUTATION  Version 1.1  <Build 1>
           Preprocessor Object: SF_SMTP  Version 1.1  <Build 9>
           Preprocessor Object: SF_SIP  Version 1.1  <Build 1>
           Preprocessor Object: SF_SSLPP  Version 1.1  <Build 4>
           Preprocessor Object: SF_GTP  Version 1.1  <Build 1>
           Preprocessor Object: SF_SSH  Version 1.1  <Build 3>
           Preprocessor Object: SF_DNP3  Version 1.1  <Build 1>
           Preprocessor Object: SF_POP  Version 1.0  <Build 1>
           Preprocessor Object: SF_DNS  Version 1.1  <Build 4>
Commencing packet processing (pid=7887)
```

Performing SSH & FTP attack from ubuntu to kali linux



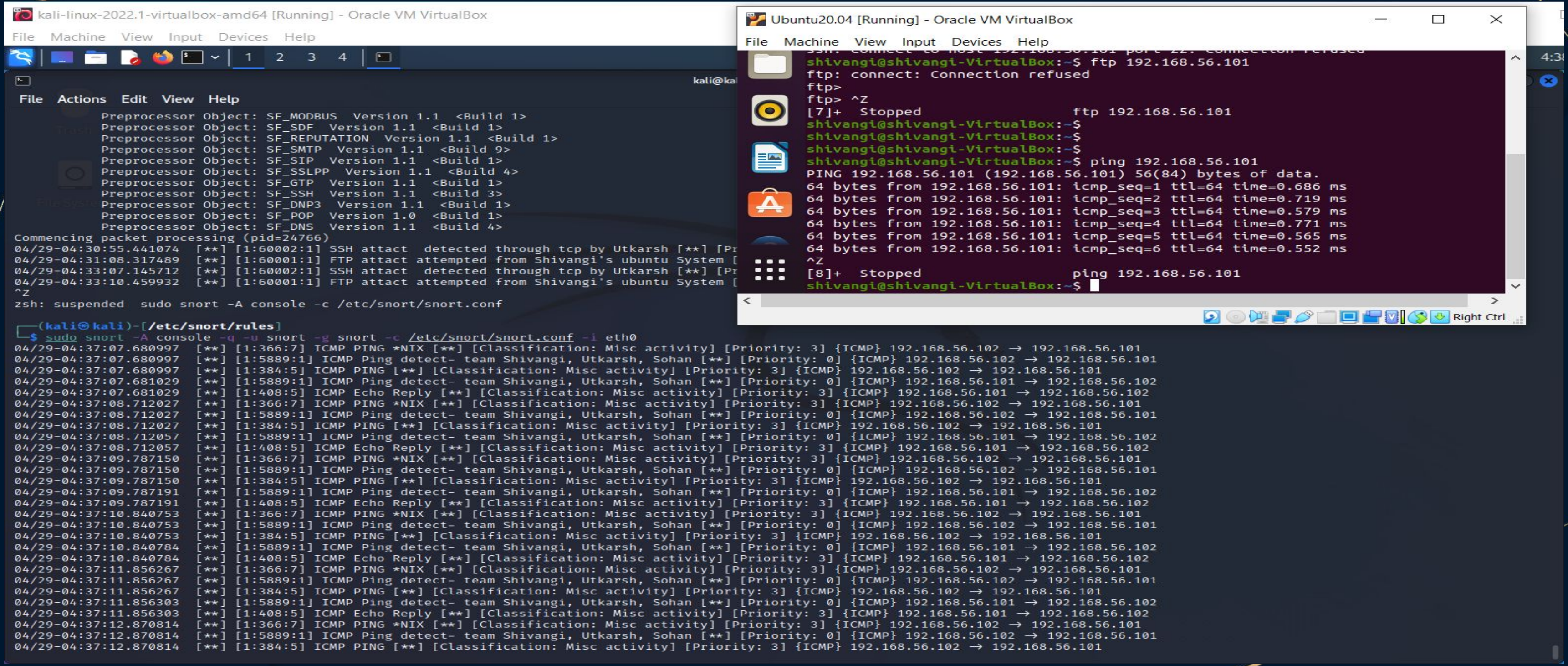
kali-linux-2022.1-virtualbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
4:33
kali@kali: /etc/snort/rules
File Actions Edit View Help

```
|      2 byte states : 14.06
|      4 byte states : 0.00
+------------------------------------------------------------------
[ Number of patterns truncated to 20 bytes: 1040 ]
pcap DAQ configured to passive.
Acquiring network traffic from "eth0".
Reload thread starting ...
Reload thread started, thread 0×7fcbd3b4f640 (24781)
Decoding Ethernet

        --== Initialization Complete ==--

   ,,_     -*> Snort! <*-
  o"  )~   Version 2.9.15.1 GRE (Build 15125)
   ''''    By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
           Copyright (C) 2014-2019 Cisco and/or its affiliates. All rights reserved.
           Copyright (C) 1998-2013 Sourcefire, Inc., et al.
           Using libpcap version 1.10.1 (with TPACKET_V3)
           Using PCRE version: 8.39 2016-06-14
           Using ZLIB version: 1.2.11

           Rules Engine: SF_SNORT_DETECTION_ENGINE  Version 3.1  <Build 1>
           Preprocessor Object: SF_FTPTELNET  Version 1.2  <Build 13>
           Preprocessor Object: SF_IMAP  Version 1.0  <Build 1>
           Preprocessor Object: SF_DCERPC2  Version 1.0  <Build 3>
           Preprocessor Object: appid  Version 1.1  <Build 5>
           Preprocessor Object: SF_MODBUS  Version 1.1  <Build 1>
           Preprocessor Object: SF_SDF  Version 1.1  <Build 1>
           Preprocessor Object: SF_REPUTATION  Version 1.1  <Build 1>
           Preprocessor Object: SF_SMTP  Version 1.1  <Build 9>
           Preprocessor Object: SF_SIP  Version 1.1  <Build 1>
           Preprocessor Object: SF_SSLPP  Version 1.1  <Build 4>
           Preprocessor Object: SF_GTP  Version 1.1  <Build 1>
           Preprocessor Object: SF_SSH  Version 1.1  <Build 3>
           Preprocessor Object: SF_DNP3  Version 1.1  <Build 1>
           Preprocessor Object: SF_POP  Version 1.0  <Build 1>
           Preprocessor Object: SF_DNS  Version 1.1  <Build 4>
Commencing packet processing (pid=24766)
04/29-04:30:55.441074  [**] [1:60002:1] SSH attact  detected through tcp by Utkarsh [**] [Priority: 0] {TCP} 192.168.56.102:40428 → 192.168.56.101:22
04/29-04:31:08.317489  [**] [1:60001:1] FTP attact attempted from Shivangi's ubuntu System [**] [Priority: 0] {TCP} 192.168.56.102:41978 → 192.168.56.101:21
04/29-04:33:07.145712  [**] [1:60002:1] SSH attact  detected through tcp by Utkarsh [**] [Priority: 0] {TCP} 192.168.56.102:40430 → 192.168.56.101:22
04/29-04:33:10.459932  [**] [1:60001:1] FTP attact attempted from Shivangi's ubuntu System [**] [Priority: 0] {TCP} 192.168.56.102:41980 → 192.168.56.101:21
```

Ubuntu20.04 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
shivangi@shivangi-VirtualBox:~$ ssh 192.168.56.101
ssh: connect to host 192.168.56.101 port 22: Connection refused
shivangi@shivangi-VirtualBox:~$ ftp 192.168.56.101
ftp: connect: Connection refused
ftp> ^Z
[6]+  Stopped                 ftp 192.168.56.101
shivangi@shivangi-VirtualBox:~$ ssh 192.168.56.101
ssh: connect to host 192.168.56.101 port 22: Connection refused
shivangi@shivangi-VirtualBox:~$ ftp 192.168.56.101
ftp: connect: Connection refused
ftp>
ftp> ^Z
[7]+  Stopped                 ftp 192.168.56.101
shivangi@shivangi-VirtualBox:~$
shivangi@shivangi-VirtualBox:~$
shivangi@shivangi-VirtualBox:~$
```

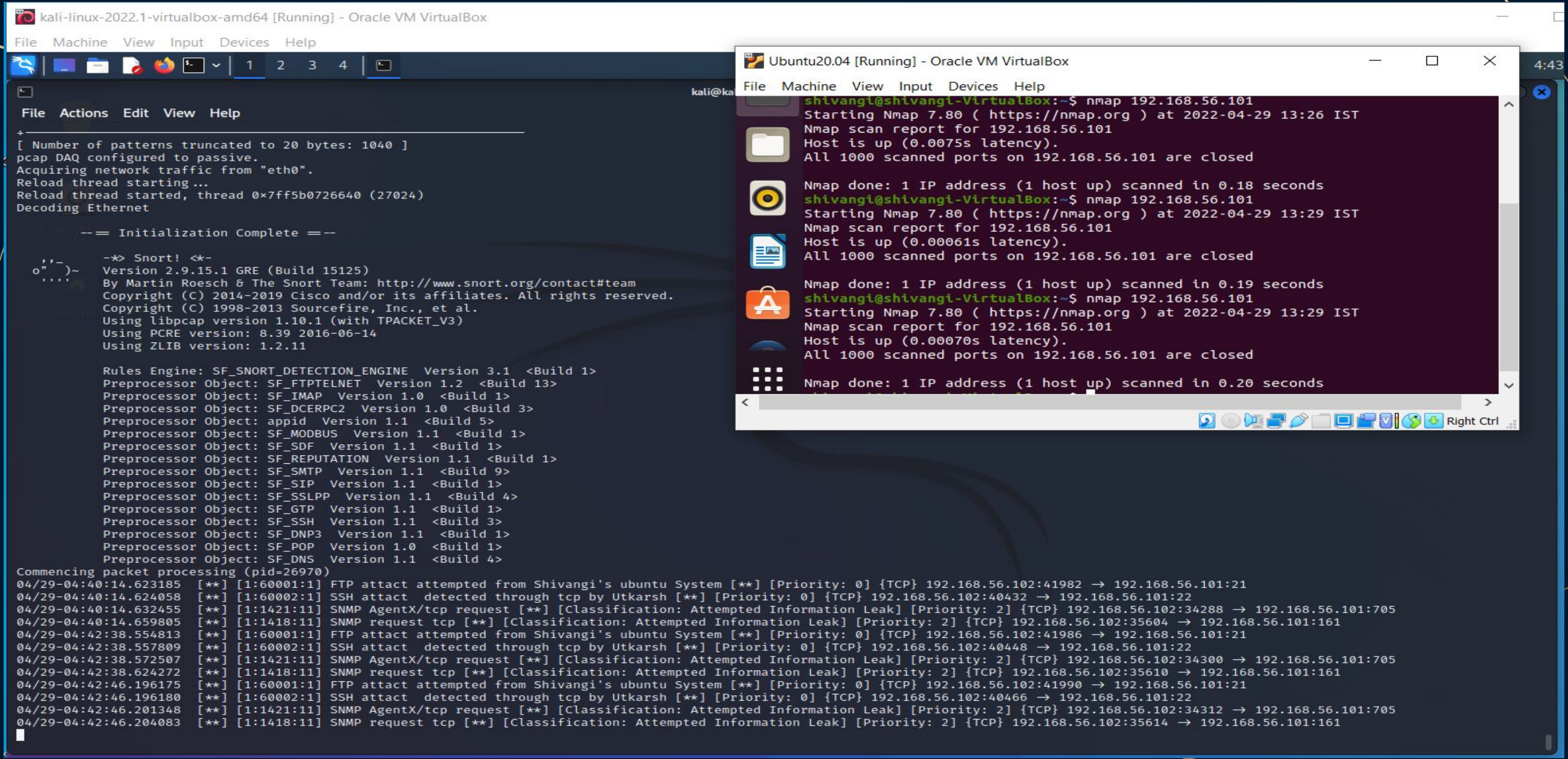
Performing ping from Ubuntu


kali-linux-2022.1-virtualbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
kali@kal
File Actions Edit View Help
Ubuntu20.04 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
             Preprocessor Object: SF_MODBUS  Version 1.1  <Build 1>
             Preprocessor Object: SF_SDF  Version 1.1  <Build 1>
             Preprocessor Object: SF_REPUTATION  Version 1.1  <Build 1>
             Preprocessor Object: SF_SMTP  Version 1.1  <Build 9>
             Preprocessor Object: SF_SIP  Version 1.1  <Build 1>
             Preprocessor Object: SF_SSLPP  Version 1.1  <Build 4>
             Preprocessor Object: SF_GTP  Version 1.1  <Build 1>
             Preprocessor Object: SF_SSH  Version 1.1  <Build 3>
             Preprocessor Object: SF_DNP3  Version 1.1  <Build 1>
             Preprocessor Object: SF_POP  Version 1.0  <Build 1>
             Preprocessor Object: SF_DNS  Version 1.1  <Build 4>
Commencing packet processing (pid=24766)
04/29-04:30:55.441074  [**] [1:60002:1] SSH attact  detected through tcp by Utkarsh [**] [Pr
04/29-04:31:08.317489  [**] [1:60001:1] FTP attact attempted from Shivangi's ubuntu System [
04/29-04:33:07.145712  [**] [1:60002:1] SSH attact  detected through tcp by Utkarsh [**] [Pr
04/29-04:33:10.459932  [**] [1:60001:1] FTP attact attempted from Shivangi's ubuntu System [
^Z
zsh: suspended  sudo snort -A console -c /etc/snort/snort.conf

┌──(kali㉿kali)-[/etc/snort/rules]
└─$ sudo snort -A console -q -u snort -g snort -c /etc/snort/snort.conf -i eth0
04/29-04:37:07.680997  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.56.102 → 192.168.56.101
04/29-04:37:07.680997  [**] [1:5889:1] ICMP Ping detect- team Shivangi, Utkarsh, Sohan [**] [Priority: 0] {ICMP} 192.168.56.102 → 192.168.56.101
04/29-04:37:07.680997  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.56.102 → 192.168.56.101
04/29-04:37:07.681029  [**] [1:5889:1] ICMP Ping detect- team Shivangi, Utkarsh, Sohan [**] [Priority: 0] {ICMP} 192.168.56.101 → 192.168.56.102
04/29-04:37:07.681029  [**] [1:408:5] ICMP Echo Reply [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.56.101 → 192.168.56.102
04/29-04:37:08.712027  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.56.102 → 192.168.56.101
04/29-04:37:08.712027  [**] [1:5889:1] ICMP Ping detect- team Shivangi, Utkarsh, Sohan [**] [Priority: 0] {ICMP} 192.168.56.102 → 192.168.56.101
04/29-04:37:08.712027  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.56.102 → 192.168.56.101
04/29-04:37:08.712057  [**] [1:5889:1] ICMP Ping detect- team Shivangi, Utkarsh, Sohan [**] [Priority: 0] {ICMP} 192.168.56.101 → 192.168.56.102
04/29-04:37:08.712057  [**] [1:408:5] ICMP Echo Reply [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.56.101 → 192.168.56.102
04/29-04:37:09.787150  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.56.102 → 192.168.56.101
04/29-04:37:09.787150  [**] [1:5889:1] ICMP Ping detect- team Shivangi, Utkarsh, Sohan [**] [Priority: 0] {ICMP} 192.168.56.102 → 192.168.56.101
04/29-04:37:09.787150  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.56.102 → 192.168.56.101
04/29-04:37:09.787191  [**] [1:5889:1] ICMP Ping detect- team Shivangi, Utkarsh, Sohan [**] [Priority: 0] {ICMP} 192.168.56.101 → 192.168.56.102
04/29-04:37:09.787191  [**] [1:408:5] ICMP Echo Reply [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.56.101 → 192.168.56.102
04/29-04:37:10.840753  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.56.102 → 192.168.56.101
04/29-04:37:10.840753  [**] [1:5889:1] ICMP Ping detect- team Shivangi, Utkarsh, Sohan [**] [Priority: 0] {ICMP} 192.168.56.102 → 192.168.56.101
04/29-04:37:10.840753  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.56.102 → 192.168.56.101
04/29-04:37:10.840784  [**] [1:5889:1] ICMP Ping detect- team Shivangi, Utkarsh, Sohan [**] [Priority: 0] {ICMP} 192.168.56.101 → 192.168.56.102
04/29-04:37:10.840784  [**] [1:408:5] ICMP Echo Reply [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.56.101 → 192.168.56.102
04/29-04:37:11.856267  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.56.102 → 192.168.56.101
04/29-04:37:11.856267  [**] [1:5889:1] ICMP Ping detect- team Shivangi, Utkarsh, Sohan [**] [Priority: 0] {ICMP} 192.168.56.102 → 192.168.56.101
04/29-04:37:11.856267  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.56.102 → 192.168.56.101
04/29-04:37:11.856303  [**] [1:5889:1] ICMP Ping detect- team Shivangi, Utkarsh, Sohan [**] [Priority: 0] {ICMP} 192.168.56.101 → 192.168.56.102
04/29-04:37:11.856303  [**] [1:408:5] ICMP Echo Reply [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.56.101 → 192.168.56.102
04/29-04:37:12.870814  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.56.102 → 192.168.56.101
04/29-04:37:12.870814  [**] [1:5889:1] ICMP Ping detect- team Shivangi, Utkarsh, Sohan [**] [Priority: 0] {ICMP} 192.168.56.102 → 192.168.56.101
04/29-04:37:12.870814  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.56.102 → 192.168.56.101
```

```
ssh: connect to host 192.168.56.101 port 22: Connection refused
shivangi@shivangi-VirtualBox:~$ ftp 192.168.56.101
ftp: connect: Connection refused
ftp>
ftp> ^Z
[7]+  Stopped                 ftp 192.168.56.101
shivangi@shivangi-VirtualBox:~$
shivangi@shivangi-VirtualBox:~$
shivangi@shivangi-VirtualBox:~$
shivangi@shivangi-VirtualBox:~$ ping 192.168.56.101
PING 192.168.56.101 (192.168.56.101) 56(84) bytes of data.
64 bytes from 192.168.56.101: icmp_seq=1 ttl=64 time=0.686 ms
64 bytes from 192.168.56.101: icmp_seq=2 ttl=64 time=0.719 ms
64 bytes from 192.168.56.101: icmp_seq=3 ttl=64 time=0.579 ms
64 bytes from 192.168.56.101: icmp_seq=4 ttl=64 time=0.771 ms
64 bytes from 192.168.56.101: icmp_seq=5 ttl=64 time=0.565 ms
64 bytes from 192.168.56.101: icmp_seq=6 ttl=64 time=0.552 ms
^Z
[8]+  Stopped                 ping 192.168.56.101
shivangi@shivangi-VirtualBox:~$
```

NMAP attack



kali-linux-2022.1-virtualbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
File Actions Edit View Help
kali@ka
Ubuntu20.04 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl
4:43

```
shivangi@shivangi-VirtualBox:~$ nmap 192.168.56.101
Starting Nmap 7.80 ( https://nmap.org ) at 2022-04-29 13:26 IST
Nmap scan report for 192.168.56.101
Host is up (0.0075s latency).
All 1000 scanned ports on 192.168.56.101 are closed

Nmap done: 1 IP address (1 host up) scanned in 0.18 seconds
shivangi@shivangi-VirtualBox:~$ nmap 192.168.56.101
Starting Nmap 7.80 ( https://nmap.org ) at 2022-04-29 13:29 IST
Nmap scan report for 192.168.56.101
Host is up (0.00061s latency).
All 1000 scanned ports on 192.168.56.101 are closed

Nmap done: 1 IP address (1 host up) scanned in 0.19 seconds
shivangi@shivangi-VirtualBox:~$ nmap 192.168.56.101
Starting Nmap 7.80 ( https://nmap.org ) at 2022-04-29 13:29 IST
Nmap scan report for 192.168.56.101
Host is up (0.00070s latency).
All 1000 scanned ports on 192.168.56.101 are closed

Nmap done: 1 IP address (1 host up) scanned in 0.20 seconds
```

```
+
[ Number of patterns truncated to 20 bytes: 1040 ]
pcap DAQ configured to passive.
Acquiring network traffic from "eth0".
Reload thread starting ...
Reload thread started, thread 0×7ff5b0726640 (27024)
Decoding Ethernet

        --== Initialization Complete ==--

   ,,_     -*> Snort! <*-
  o"  )~   Version 2.9.15.1 GRE (Build 15125)
   ''''    By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
           Copyright (C) 2014-2019 Cisco and/or its affiliates. All rights reserved.
           Copyright (C) 1998-2013 Sourcefire, Inc., et al.
           Using libpcap version 1.10.1 (with TPACKET_V3)
           Using PCRE version: 8.39 2016-06-14
           Using ZLIB version: 1.2.11

           Rules Engine: SF_SNORT_DETECTION_ENGINE  Version 3.1  <Build 1>
           Preprocessor Object: SF_FTPTELNET  Version 1.2  <Build 13>
           Preprocessor Object: SF_IMAP  Version 1.0  <Build 1>
           Preprocessor Object: SF_DCERPC2  Version 1.0  <Build 3>
           Preprocessor Object: appid  Version 1.1  <Build 5>
           Preprocessor Object: SF_MODBUS  Version 1.1  <Build 1>
           Preprocessor Object: SF_SDF  Version 1.1  <Build 1>
           Preprocessor Object: SF_REPUTATION  Version 1.1  <Build 1>
           Preprocessor Object: SF_SMTP  Version 1.1  <Build 9>
           Preprocessor Object: SF_SIP  Version 1.1  <Build 1>
           Preprocessor Object: SF_SSLPP  Version 1.1  <Build 4>
           Preprocessor Object: SF_GTP  Version 1.1  <Build 1>
           Preprocessor Object: SF_SSH  Version 1.1  <Build 3>
           Preprocessor Object: SF_DNP3  Version 1.1  <Build 1>
           Preprocessor Object: SF_POP  Version 1.0  <Build 1>
           Preprocessor Object: SF_DNS  Version 1.1  <Build 4>
Commencing packet processing (pid=26970)
04/29-04:40:14.623185  [**] [1:60001:1] FTP attact attempted from Shivangi's ubuntu System [**] [Priority: 0] {TCP} 192.168.56.102:41982 → 192.168.56.101:21
04/29-04:40:14.624058  [**] [1:60002:1] SSH attact  detected through tcp by Utkarsh [**] [Priority: 0] {TCP} 192.168.56.102:40432 → 192.168.56.101:22
04/29-04:40:14.632455  [**] [1:1421:11] SNMP AgentX/tcp request [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 192.168.56.102:34288 → 192.168.56.101:705
04/29-04:40:14.659805  [**] [1:1418:11] SNMP request tcp [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 192.168.56.102:35604 → 192.168.56.101:161
04/29-04:42:38.554813  [**] [1:60001:1] FTP attact attempted from Shivangi's ubuntu System [**] [Priority: 0] {TCP} 192.168.56.102:41986 → 192.168.56.101:21
04/29-04:42:38.557809  [**] [1:60002:1] SSH attact  detected through tcp by Utkarsh [**] [Priority: 0] {TCP} 192.168.56.102:40448 → 192.168.56.101:22
04/29-04:42:38.572507  [**] [1:1421:11] SNMP AgentX/tcp request [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 192.168.56.102:34300 → 192.168.56.101:705
04/29-04:42:38.624272  [**] [1:1418:11] SNMP request tcp [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 192.168.56.102:35610 → 192.168.56.101:161
04/29-04:42:46.196175  [**] [1:60001:1] FTP attact attempted from Shivangi's ubuntu System [**] [Priority: 0] {TCP} 192.168.56.102:41990 → 192.168.56.101:21
04/29-04:42:46.196180  [**] [1:60002:1] SSH attact  detected through tcp by Utkarsh [**] [Priority: 0] {TCP} 192.168.56.102:40466 → 192.168.56.101:22
04/29-04:42:46.201348  [**] [1:1421:11] SNMP AgentX/tcp request [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 192.168.56.102:34312 → 192.168.56.101:705
04/29-04:42:46.204083  [**] [1:1418:11] SNMP request tcp [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 192.168.56.102:35614 → 192.168.56.101:161
```

Conclusion

After familiarizing with IDS and its classifications, different Snort based Intrusion Detection techniques are discussed and practically performed in this project to upkeep the security of an organization against attacks. Snort based IDPS using efficient rules, Bayseian Network, Honeypot, Hardware-assisted technique, Neural Networks and Multi-Sensors like techniques can protect from simple intrusions to dangerous DoS and DDoS type attacks in high speed and Cloud environments with considerable drawbacks. Various challenges are identified and discussed which are to be considered while designing efficient IDS for any network. There are still many ways to enhance the efficiency of Snort based Intrusion Detection and Prevention System. In future we will integrate the proposed design into Snort tool and evaluate it to achieve better detection rate with less false alarms.

THANK YOU!