



Instituto Tecnológico de Cancún

Materia:

Fundamentos de telecomunicaciones

Proyecto de Sistemas de Comunicaciones

Profesor:

Ismael Jiménez Sánchez

Alumno:

Ángel Eduardo Hernández Pimentel

26 – octubre - 2020

Introducción

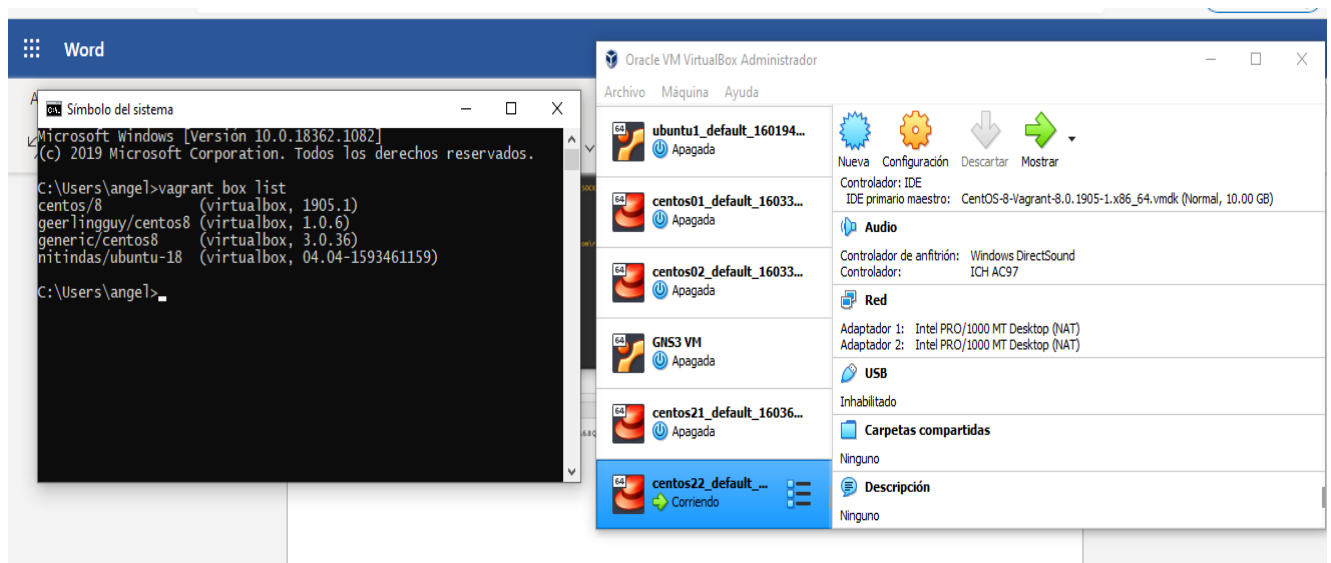
En el siguiente proyecto se mostrará un ejemplo de telecomunicaciones, esta actividad a mostrar tiene como finalidad fomentar la participación en las telecomunicaciones, ya que en la actualidad tienen un papel muy importante y prácticamente está en casi todos los ámbitos algunos son la educación, negocios, en fin, muchas otras áreas de trabajo.

El proyecto fue elaborado gracias a las siguientes herramientas:

- Vagrant
- VirtualBox
- Gns3
- Putty
- Python
- Wireshark

Fase 1

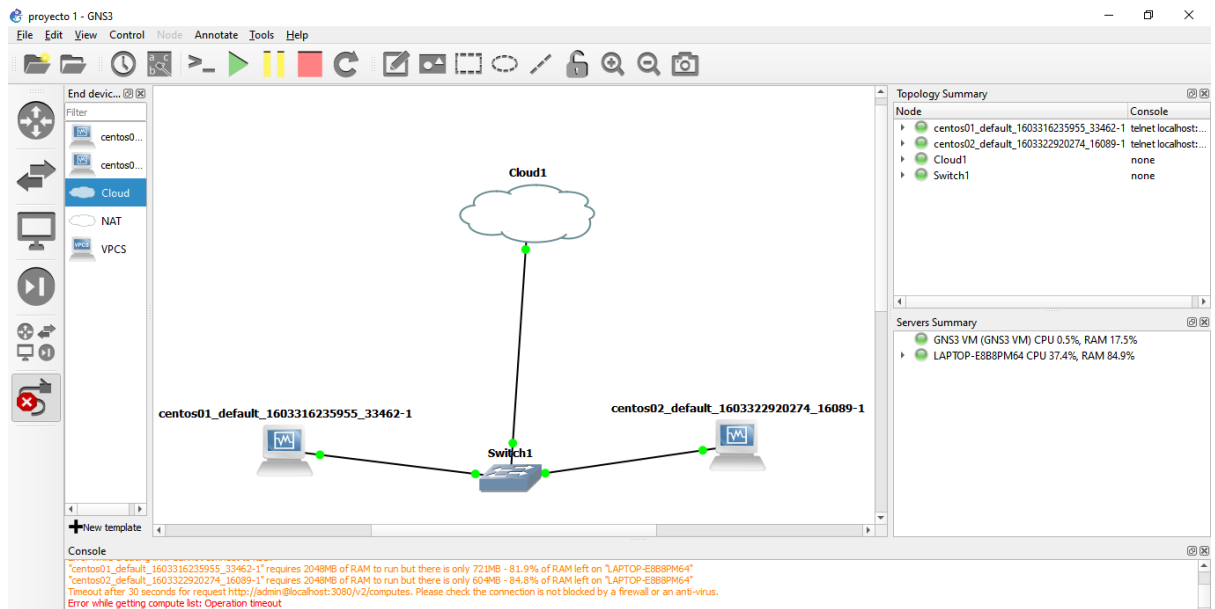
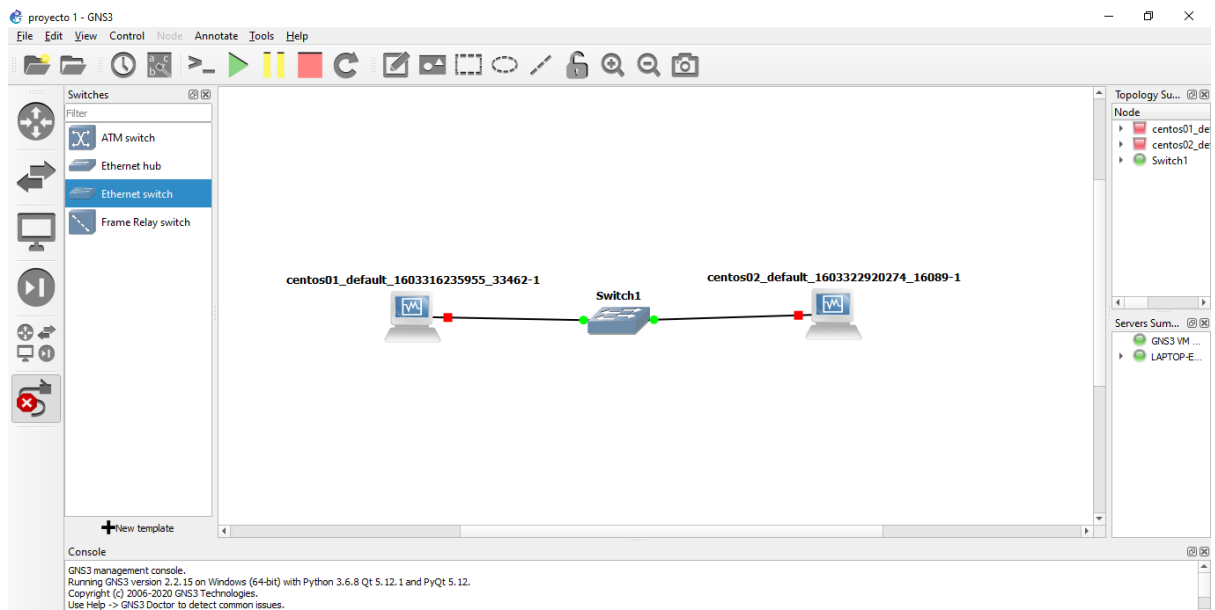
En esta parte instalamos 2 centos en virtualbox usando vagrant, para esto creamos dos carpetas para cada uno de los centos, esto con el fin de que aparezcan directamente en el virtualbox.

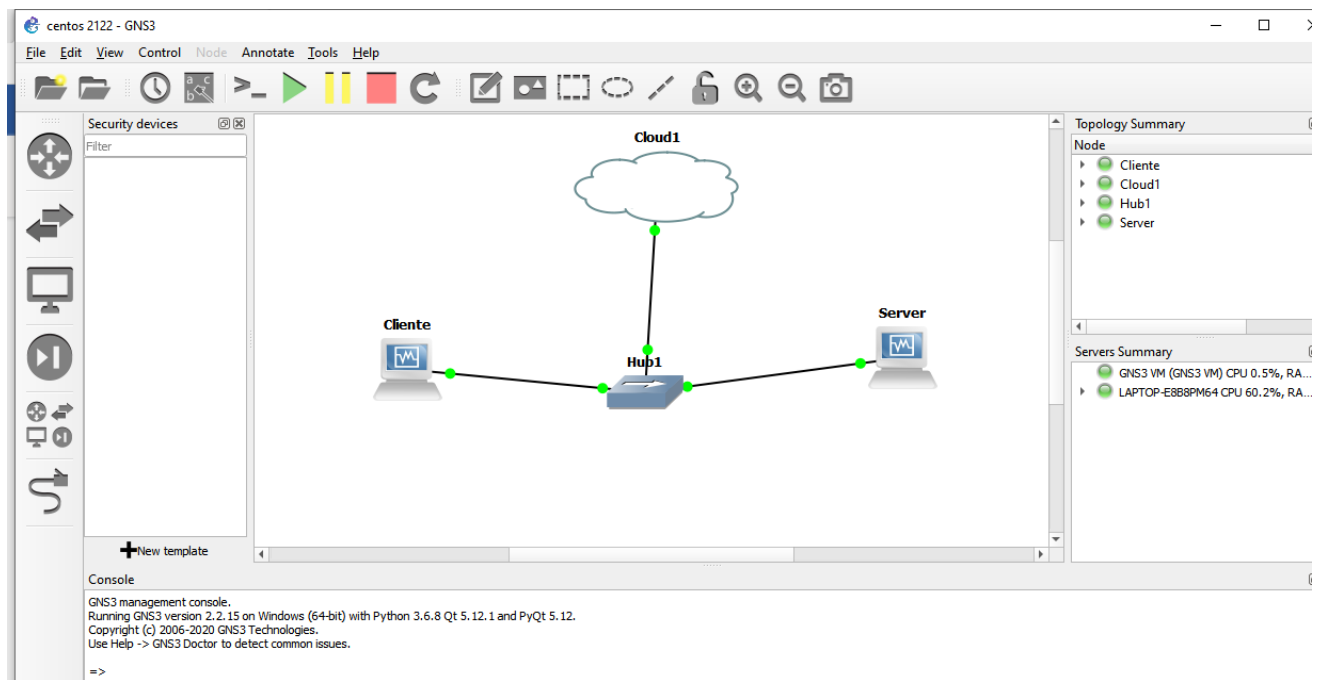


Fase 2

Esta fase consiste en conectar en GNS3, las dos VMs de Centos con un switch ethernet.

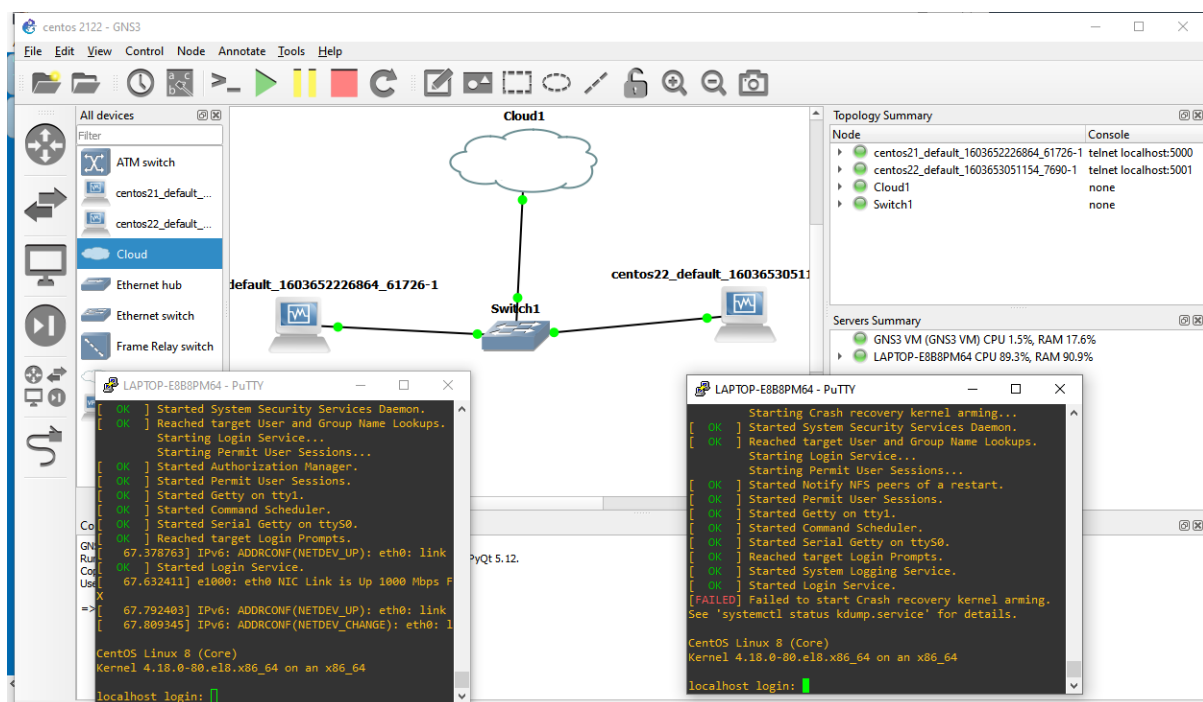
Al tener la conexión hay que hacer algunas configuraciones en preferencias y agregar las maquinas, después configurar cada una de ellas, activar la parte de network y cambiar la parte de telnet.

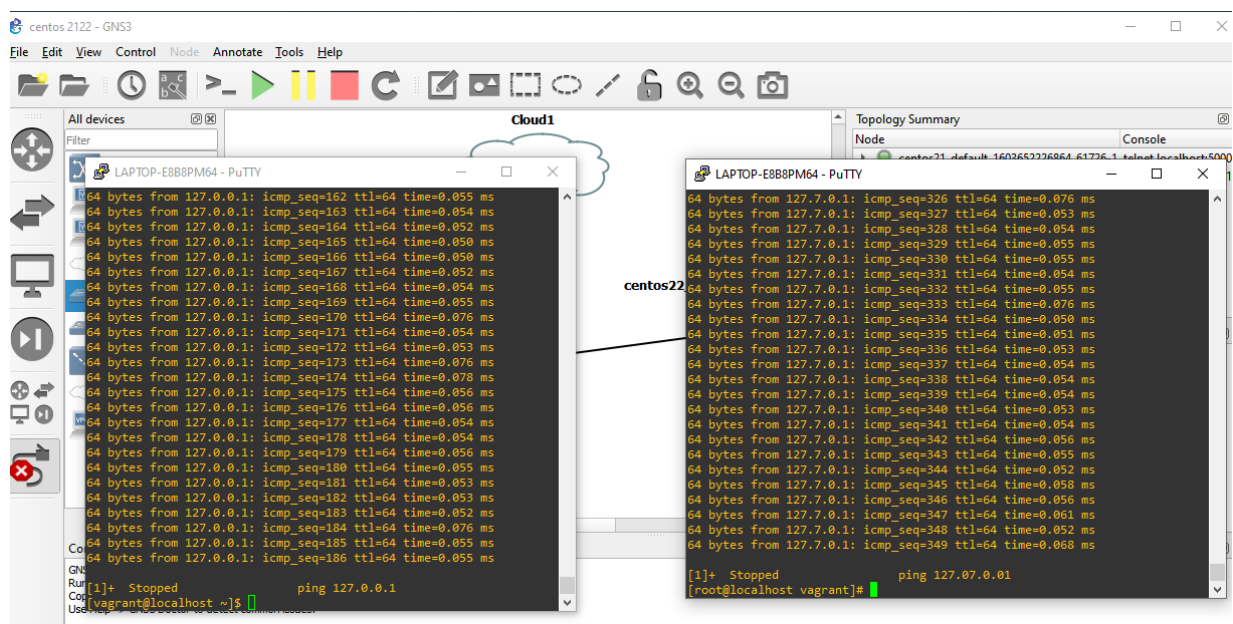




Fase 3

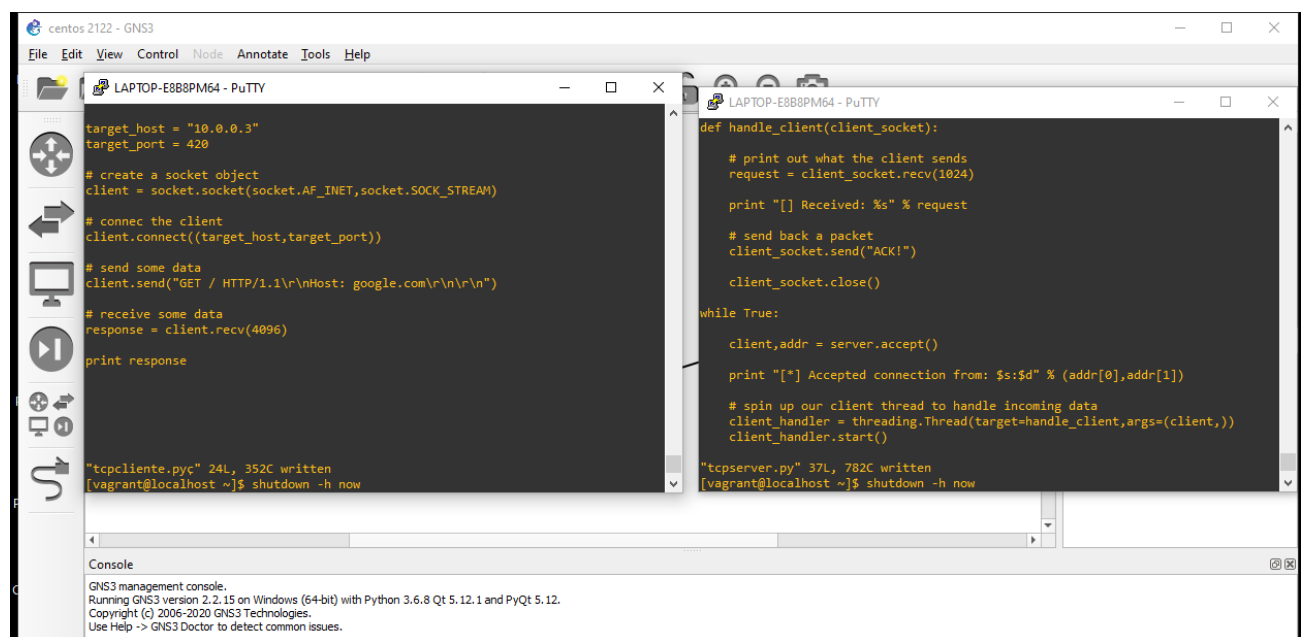
Esta es una de las partes más interesantes ya que es donde la magia ocurre, ya que es esta parte usamos la herramienta putty para configurar los puertos desde luego que por medio de comandos, al usar putty debimos agregar localhost, telnet y el número de puerto.





Fase 4

En esta fase conectamos wireshark para visualizar el tráfico de las maquinas que agregamos en este caso los dos CentOS los cuales nombramos cliente y server, también agregamos los scripts de python.



```
centos22_default_1603653051154_7690 [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
python3-dnf-plugin-spacewalk.noarch : 3 exceptions
python3-dnf-plugin-spacewalk.noarch : DNF plugin for Spacewalk
python3-libstoragegmt-clibs.x86_64 : Python 3 C extension module for
python3-libstoragegmt-clibs.x86_64 : libstoragegmt
python3-iscsi-initiator-utils.x86_64 : Python 3.6 bindings to
python3-iscsi-initiator-utils.x86_64 : iscsi-initiator-utils
python3-dbus-python-client-gen.noarch : Python Library for Generating
python3-dbus-python-client-gen.noarch : dbus-python Client Code
python3-spacewalk-backend-libs.noarch : Spacewalk client tools libraries for
python3-spacewalk-backend-libs.noarch : Fedora 23
python3-dnf-plugin-versionlock.noarch : Version Lock Plugin for DNF
python3-rhn-virtualization-host.noarch : RHN/Spacewalk Virtualization support
python3-rhn-virtualization-host.noarch : specific to the Host system
python3-dbus-signature-pyparsing.noarch : Parser for a D-Bus Signature
python3-rhn-virtualization-common.noarch : Files needed by
python3-rhn-virtualization-common.noarch : rhn-virtualization-host
python3-subscription-manager-rhsm.x86_64 : A Python library to communicate with
python3-subscription-manager-rhsm.x86_64 : ...: a Red Hat Unified Entitlement Platform
python3-keycloak-httpd-client-install.noarch : Tools to configure Apache HTTPD
python3-keycloak-httpd-client-install.noarch : as Keycloak client
python3-dnf-plugin-post-transaction-actions.noarch : Post transaction actions
python3-dnf-plugin-post-transaction-actions.noarch : Plugin for DNF
===== Summary Matched: python3 =====
platform-python-pip.noarch : A tool for installing and managing Python3 packages
lvagrant@localhost ~1$
```

```
centos22_default_1603653051154_7690 [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
Installing      : python36-3.6.8-2.module_el8.1.0+245+c39af44f.x86_64 4/7
Running scriptlet: python36-3.6.8-2.module_el8.1.0+245+c39af44f.x86_64 4/7
Installing      : python3-pip-9.0.3-16.el8.noarch 5/7
Cleanup         : platform-python-pip-9.0.3-13.el8.noarch 6/7
Cleanup         : platform-python-setuptools-39.2.0-4.el8.noarch 7/7
Running scriptlet: platform-python-setuptools-39.2.0-4.el8.noarch 7/7
Verifying       : python3-pip-9.0.3-16.el8.noarch 1/7
Verifying       : python36-3.6.8-2.module_el8.1.0+245+c39af44f.x86_64 2/7
Verifying       : python3-setuptools-39.2.0-5.el8.noarch 3/7
Verifying       : platform-python-pip-9.0.3-16.el8.noarch 4/7
Verifying       : platform-python-pip-9.0.3-13.el8.noarch 5/7
Verifying       : platform-python-setuptools-39.2.0-5.el8.noarch 6/7
Verifying       : platform-python-setuptools-39.2.0-4.el8.noarch 7/7

Upgraded:
platform-python-pip-9.0.3-16.el8.noarch
platform-python-setuptools-39.2.0-5.el8.noarch

Installed:
python36-3.6.8-2.module_el8.1.0+245+c39af44f.x86_64
python3-pip-9.0.3-16.el8.noarch
python3-setuptools-39.2.0-5.el8.noarch

Complete!
lvagrant@localhost ~1$
```

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.50.194	23.37.83.178	TCP	54	51255 → 80 [FIN, ACK] Seq=1 Ack=1 Win=1026 Len=0
2	0.026978	23.37.83.178	192.168.50.194	TCP	60	80 → 51255 [FIN, ACK] Seq=1 Ack=2 Win=237 Len=0
3	0.027045	192.168.50.194	23.37.83.178	TCP	54	51255 → 80 [ACK] Seq=2 Ack=2 Win=1026 Len=0
4	5.035591	ASUSTekC_86:b4:18	ASUSTekC_86:57:7e	ARP	60	Who has 192.168.50.194? Tell 192.168.50.1
5	5.035618	ASUSTekC_86:57:7e	ASUSTekC_b8:b4:18	ARP	42	192.168.50.194 is at 70:4d:7b:06:57:7e
6	14.751482	192.168.50.194	35.186.224.47	TLSv1.2	89	Application Data
7	14.782380	35.186.224.47	192.168.50.194	TCP	60	443 → 50756 [ACK] Seq=1 Ack=36 Win=266 Len=0
8	14.813528	35.186.224.47	192.168.50.194	TLSv1.2	85	Application Data
9	14.854875	192.168.50.194	35.186.224.47	TCP	54	50756 → 443 [ACK] Seq=36 Ack=32 Win=1026 Len=0

```

> Frame 1: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on Interface \Device\NPF_{69C96369-ED97-46A6-9643-670FEB74639F}, Id 0
> Ethernet II, Src: ASUSTekC_86:57:7e (70:4d:7b:06:57:7e), Dst: ASUSTekC_b8:b4:18 (04:d4:c4:b8:b4:18)
> Internet Protocol Version 4, Src: 192.168.50.194, Dst: 23.37.83.178
> Transmission Control Protocol, Src Port: 51255, Dst Port: 80, Seq: 1, Ack: 1, Len: 0

```

```

0000  04 d4 c4 b8 b4 18 70 4d 7b 06 57 7e 00 00 45 00  .....ph {..h...E-
0010  00 28 ce 5d 40 00 00 00 00 00 c0 a8 32 c2 17 25  -(..)0... ..2...X
0020  53 b2 c8 37 00 50 21 62 d7 74 d9 f3 57 c0 50 11  S..7Plb..t..lP-
0030  04 02 5e 5c 00 00  ..A\..

```

```

<Socket>={-tl--tcp} {-ul--udp} {-Ul--udplite} {-Sl--sctp} {-wl--raw}
           {-xl--unix} --ax25 --ipx --netrom
<AF>=Use '-6l-4' or '-A <af>' or '--<af>'; default: inet
List of possible address families (which support routing):
inet (DARPA Internet) inet6 (IPv6) ax25 (AMPR AX.25)
netrom (AMPR NET/ROM) ipx (Novell IPX) ddp (Appletalk DDP)
x25 (CCITT X.25)
[root@localhost vagrant]# netstat -tapn | grep LISTEN
tcp        0      0 0.0.0.0:22          0.0.0.0:*           LISTEN      785/sshd
tcp        0      0 0.0.0.0:111         0.0.0.0:*           LISTEN      1/systemd
tcp6       0      0 :::22              :::*                LISTEN      785/sshd
tcp6       0      0 :::111             :::*                LISTEN      1/systemd
[root@localhost vagrant]#

```


No.	Time	Source	Destination	Protocol	Length	Info
11	21.183196	192.168.50.194	239.255.255.250	SSDP	211	M-SEARCH * HTTP/1.1
12	22.184573	192.168.50.194	239.255.255.250	SSDP	211	M-SEARCH * HTTP/1.1
13	22.678197	192.168.50.194	192.168.50.255	UDP	86	57621 → 57621 Len=44
14	23.898253	192.168.50.194	192.168.50.255	NBNS	92	Name query NB WORKGROUP<1c>
15	23.184952	192.168.50.194	239.255.255.250	SSDP	211	M-SEARCH * HTTP/1.1
16	23.597190	35.190.245.14	192.168.50.194	TCP	65	4870 → 50751 [PSH, ACK] Seq=1 Ack=1 Win=39 Len=11
17	23.638915	192.168.50.194	35.190.245.14	TCP	54	50751 → 4870 [ACK] Seq=1 Ack=12 Win=1022 Len=0
18	23.848208	192.168.50.194	192.168.50.255	NBNS	92	Name query NB WORKGROUP<1c>
19	24.598433	192.168.50.194	192.168.50.255	NBNS	92	Name query NB WORKGROUP<1c>
20	25.437540	192.168.50.194	192.168.1.255	TCP	66	51257 → 1080 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
21	26.438176	192.168.50.194	192.168.1.255	TCP	66	[TCP Retransmission] 51257 → 1080 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
22	26.479255	192.168.50.194	192.168.1.255	TCP	66	[TCP Retransmission] 51257 → 1080 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
23	28.544176	ASUSTekC_86:57:7e	ASUSTekC_b8:b4:18	ARP	42	who has 192.168.50.1? Tell 192.168.50.194
24	28.544429	ASUSTekC_b8:b4:18	ASUSTekC_86:57:7e	ARP	80	192.168.50.1 is at 04:d4:c4:b8:b4:18
25	32.440135	192.168.50.194	192.168.1.255	TCP	66	[TCP Retransmission] 51257 → 1080 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
26	34.926935	192.168.50.194	35.186.224.25	TCP	55	51234 → 443 [ACK] Seq=1 Ack=1 Win=1027 Len=1 [TCP segment of a reassembled PDU]
27	34.954289	35.186.224.25	192.168.50.194	TCP	66	443 → 51234 [ACK] Seq=1 Ack=2 Win=608 Len=0 SLE=1 SRE=2

> Frame 1: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface \Device\NPF_{69C90560-ED07-46A8-9645-670FEB74639F}, Id 0
 > Ethernet II, Src: ASUSTekC_86:57:7e (78:4d:7b:86:57:7e), Dst: ASUSTekC_b8:b4:18 (04:d4:c4:b8:b4:18)
 > Internet Protocol Version 4, Src: 192.168.50.194, Dst: 23.37.83.178
 > Transmission Control Protocol, Src Port: 51255, Dst Port: 80, Seq: 1, Ack: 1, Len: 0

Conclusión

En conclusión, este proyecto muestra de forma precisa y hace identificar el tráfico de una red al usar dos máquinas virtuales usando CentOS y como algo adicional quiero dejar en claro que fue una experiencia genial ya que no había utilizado la mayoría de las herramientas que fueron necesarias para llevar a cabo este proyecto, la parte más genial es donde ocurre toda la magia, aunque honestamente me llegue a confundir más de una vez, pero sin duda alguna son herramientas interesantes que valen la pena conocer y practicar con ellas.

Una parte más a recalcar es que en la actualidad son muy necesarias las telecomunicaciones ya que lo utilizamos en un diversos ámbitos, somos afortunados de tener las telecomunicaciones en nuestra vida diaria.