

MANUAL PRÁCTICO PARA LA CONFIGURACIÓN DE UN ENTORNO DE PRUEBA QUE SIMULE  
REDES SDN Y CONTROLADORES QUE PERMITA LA OPERACIÓN NORMAL Y EL LANZAMIENTO  
DE ATAQUE

Javier Andrés Bernal Castañeda

Director:

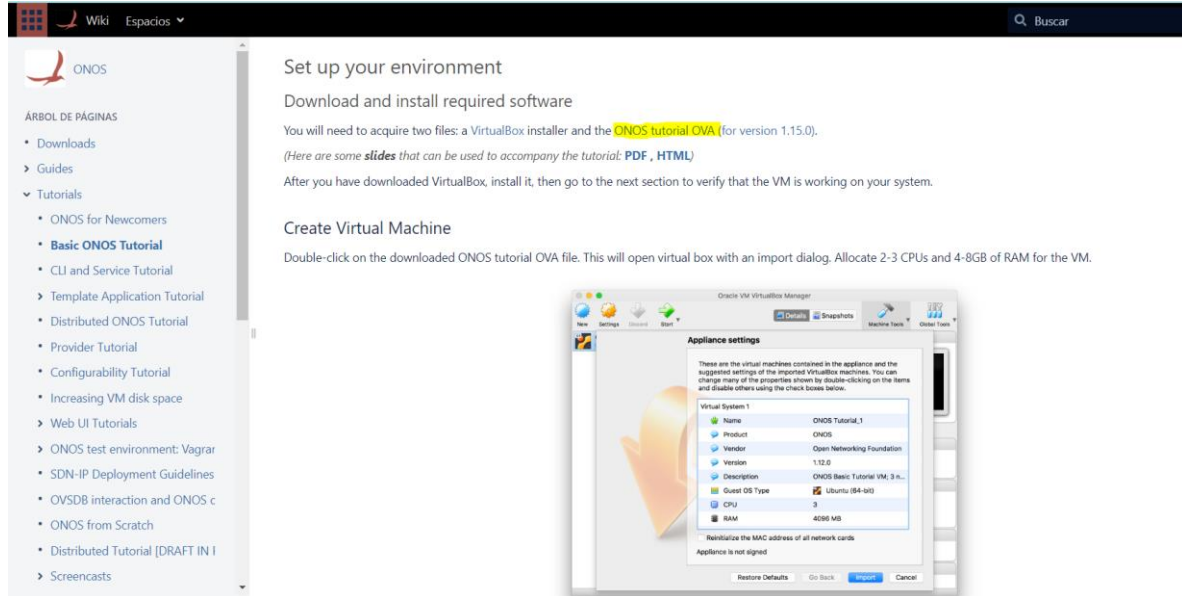
M.Sc. Juliana Alejandra Arévalo Herrera

UNIVERSIDAD SANTO TOMAS  
FACULTAD DE INGENIERÍA DE TELECOMUNICACIONES  
ESPECIALIZACIÓN EN GESTIÓN DE SERVICIOS DE TECNOLOGÍAS DE LA INFORMACIÓN  
BOGOTÁ, 2024

# MANUAL PRÁCTICO ONOS

Diríjase a la siguiente url: <https://wiki.onosproject.org/display/ONOS/Basic+ONOS+Tutorial>

Descargue la ISO ONOS tutorial OVA



The screenshot shows the ONOS Wiki page titled "Set up your environment". The page content includes:

- Set up your environment**
- Download and install required software**

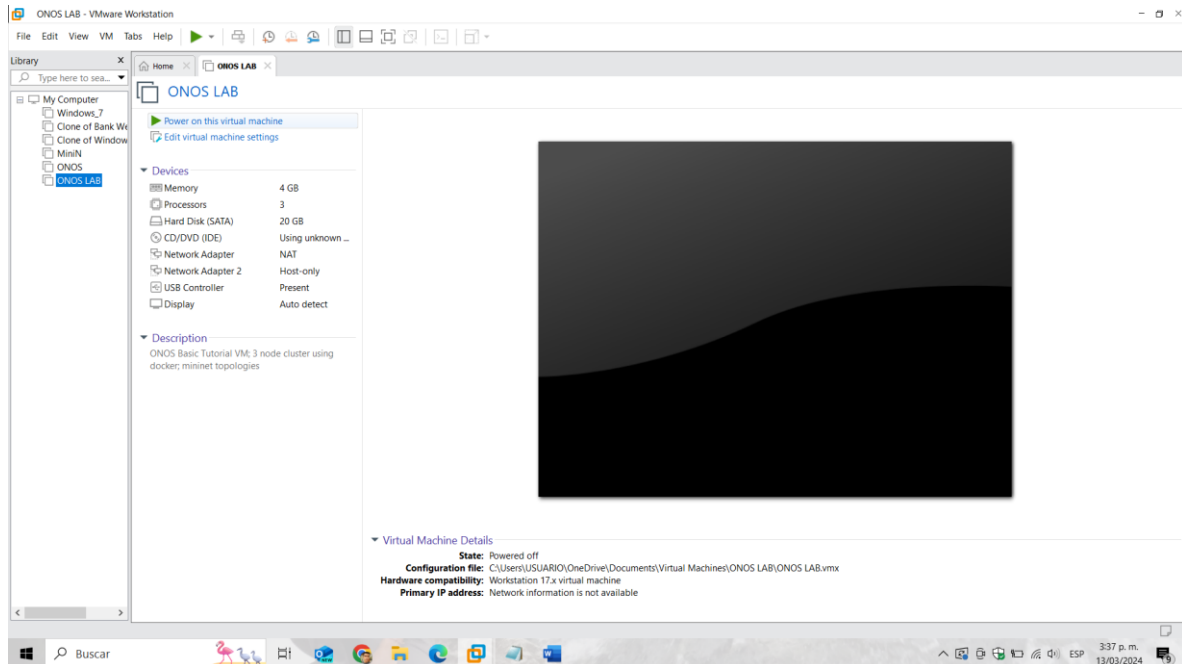
You will need to acquire two files: a VirtualBox installer and the **ONOS tutorial OVA** (for version 1.15.0).  
(Here are some **slides** that can be used to accompany the tutorial: **PDF** , **HTML**)

After you have downloaded VirtualBox, install it, then go to the next section to verify that the VM is working on your system.
- Create Virtual Machine**

Double-click on the downloaded ONOS tutorial OVA file. This will open virtual box with an import dialog. Allocate 2-3 CPUs and 4-8GB of RAM for the VM.

The left sidebar shows a navigation menu with categories like "Downloads", "Guides", "Tutorials", and "Screenscasts". The "Tutorials" section is expanded, showing various guides including "Basic ONOS Tutorial".

Una vez descargado el archivo .OVA ejecútelo con ayuda de un software de virtualización como VMware y lance una máquina virtual



The screenshot shows the VMware Workstation interface with the "ONOS LAB" virtual machine selected. The "Virtual Machine Details" pane is open, showing the following configuration:

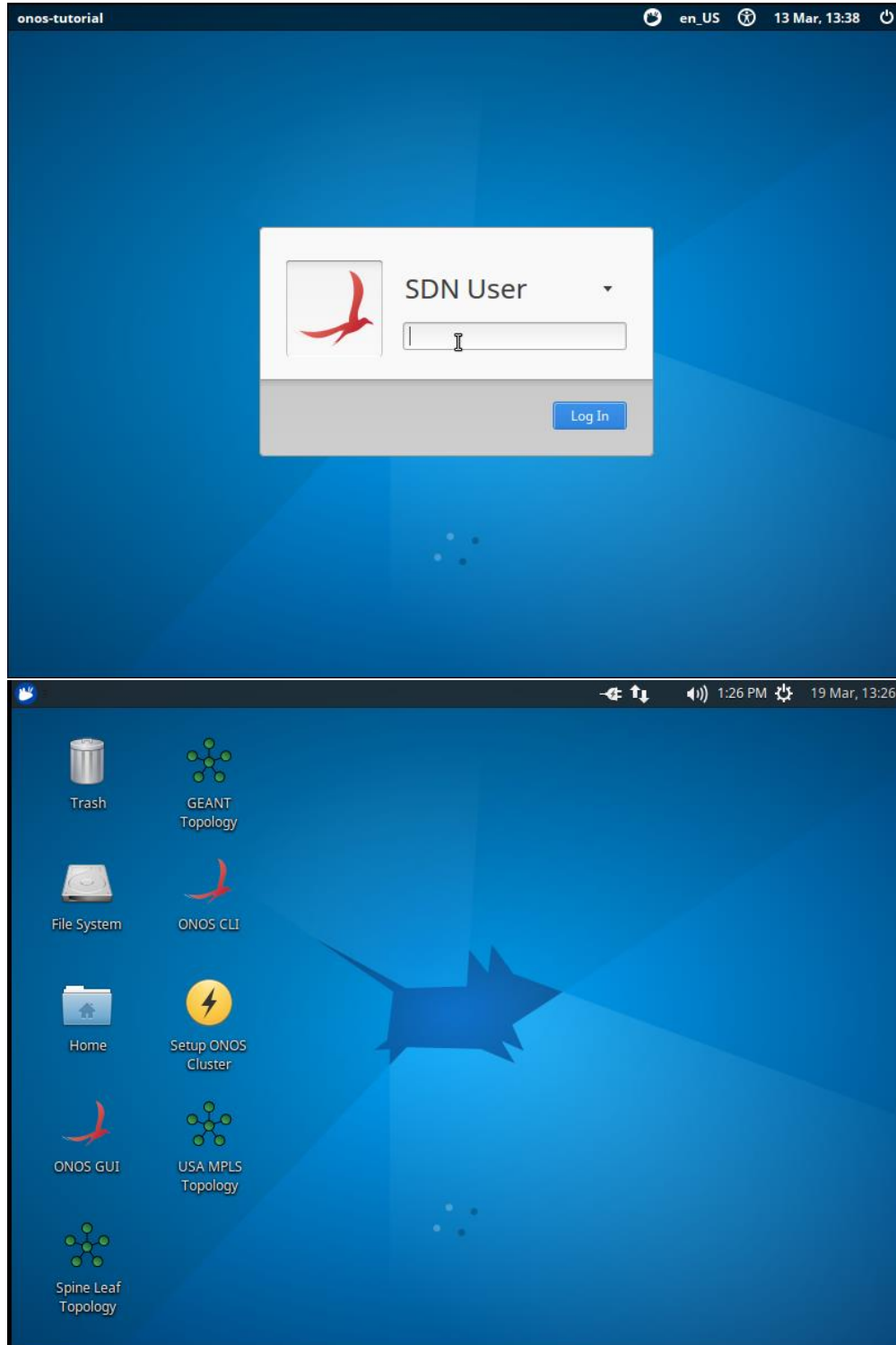
- State:** Powered off
- Configuration file:** C:\Users\USUARIO\OneDrive\Documents\Virtual Machines\ONOS LAB\ONOS LAB.vmx
- Hardware compatibility:** Workstation 17x virtual machine
- Primary IP address:** Network information is not available

The left sidebar shows the "Devices" section with the following settings:

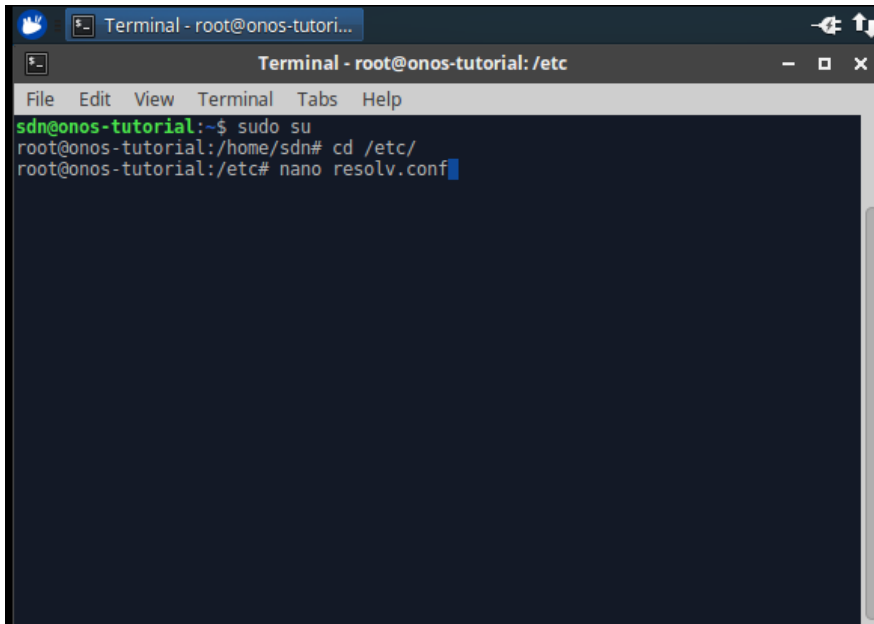
- Memory:** 4 GB
- Processors:** 3
- Hard Disk (SATA):** 20 GB
- CD/DVD (IDE):** Using unknown...
- Network Adapter:** NAT
- Network Adapter 2:** Host-only
- USB Controller:** Present
- Display:** Auto detect

The "Description" section shows: "ONOS Basic Tutorial VM; 3 node cluster using docker; mininet topologies".

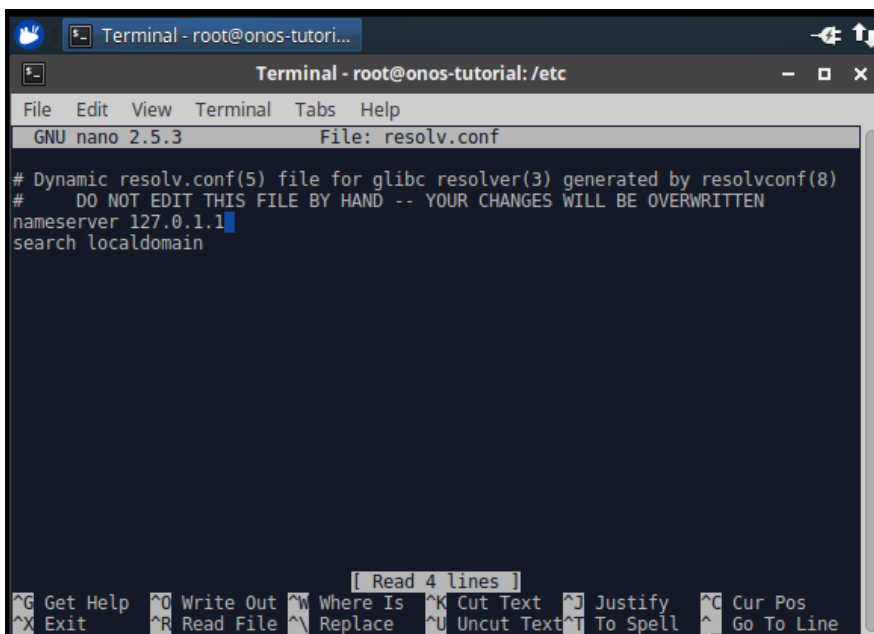
Ejecute la máquina virtual  
password:rocks



Asegúrese de que la maquina tenga salida a internet, de no ser así ejecute las siguientes instrucciones dentro del shell:

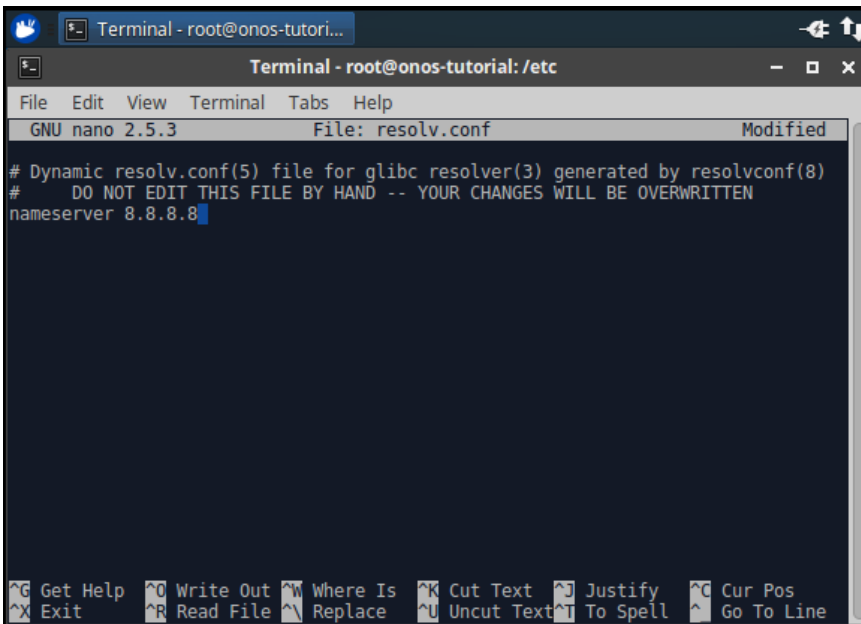


```
Terminal - root@onos-tutori...
Terminal - root@onos-tutorial: /etc
File Edit View Terminal Tabs Help
sdn@onos-tutorial:~$ sudo su
root@onos-tutorial:/home/sdn# cd /etc/
root@onos-tutorial:/etc# nano resolv.conf
```



```
Terminal - root@onos-tutori...
Terminal - root@onos-tutorial: /etc
File Edit View Terminal Tabs Help
GNU nano 2.5.3 File: resolv.conf
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)
# DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN
nameserver 127.0.1.1
search localdomain
[ Read 4 lines ]
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^_ Replace ^U Uncut Text ^T To Spell ^_ Go To Line
```

Reemplace el valor de la dirección IP del nameserver por la IP 8.8.8.8, el cual corresponde al DNS de Google



```
Terminal - root@onos-tutori...
Terminal - root@onos-tutorial: /etc
File Edit View Terminal Tabs Help
GNU nano 2.5.3 File: resolv.conf Modified
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)
# DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN
nameserver 8.8.8.8
Get Help Write Out Where Is Cut Text Justify Cur Pos
Exit Read File Replace Uncut Text To Spell Go To Line
```

Guarde los cambios presionando ctrl+X y compruebe la conexión a internet.

Antes de continuar asegúrese de actualizar los paquetes instalados ejecutando los siguientes comandos dentro del shell:

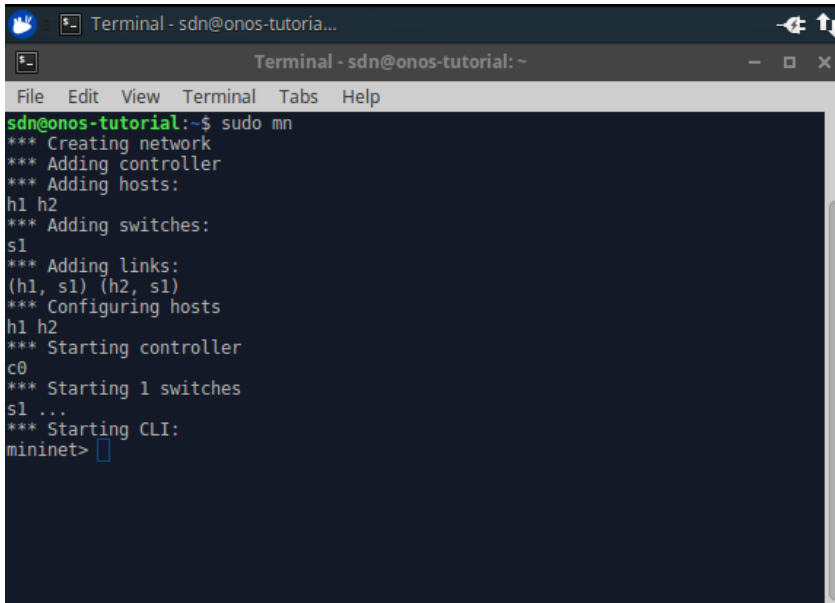
```
sudo apt update
sudo apt upgrade
```

Instale el administrador de paquetes pip y verifique su instalación. Adicionalmente instale a través del gestor de paquetes pip la biblioteca *requests* la cual será utilizada más adelante. Para ello ejecute en el Shell los siguientes comandos:

```
wget https://bootstrap.pypa.io/pip/2.7/get-pip.py
python2.7 get-pip.py
pip --version
pip install requests
```

## Desplegando la primer topología de red en mininet

Para empezar a utilizar mininet se ejecuta desde el shell el siguiente comando para lanzar una topología por defecto compuesta por 2 host, 1 switch y 1 controlador



```
Terminal - sdn@onos-tutoria...
Terminal - sdn@onos-tutorial: ~
File Edit View Terminal Tabs Help
sdn@onos-tutorial:~$ sudo mn
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2
*** Adding switches:
s1
*** Adding links:
(h1, s1) (h2, s1)
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Starting CLI:
mininet>
```

A continuación se enlistan una serie de comandos útiles para interactuar con la red:

**net:** Proporciona información detallada sobre la topología de la red, incluyendo la configuración de los enlaces y las direcciones IP asignadas a cada nodo.

**nodes:** Muestra una lista de todos los nodos en la topología de Mininet, incluidos los hosts, switches y controladores.

**pingall:** Este comando envía paquetes de ping desde cada host a todos los demás hosts en la red, permitiéndote probar la conectividad entre ellos.

**iperf:** Te permite medir el rendimiento de la red al enviar tráfico de prueba entre dos hosts y calcular la tasa de transferencia de datos.

**dump:** Este comando muestra una descripción detallada de todos los nodos en la topología, incluyendo sus atributos y configuraciones.

**link:** (o links) Muestra información sobre los enlaces entre los nodos en la topología, incluyendo el ancho de banda, la latencia y otros parámetros configurados.

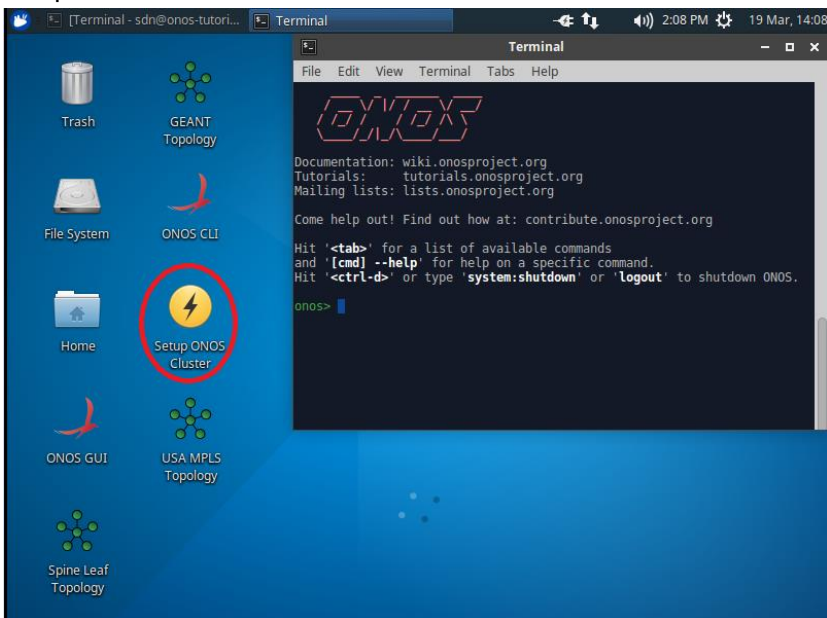
**h1 ifconfig:** Este es un ejemplo de cómo puedes ejecutar comandos específicos del sistema operativo en un nodo específico. En este caso, muestra la configuración de red del host "h1".

**h1 traceroute h2:** Permite realizar un traceroute desde el host "h1" al host "h2", mostrando la ruta que sigue el paquete a través de la red.

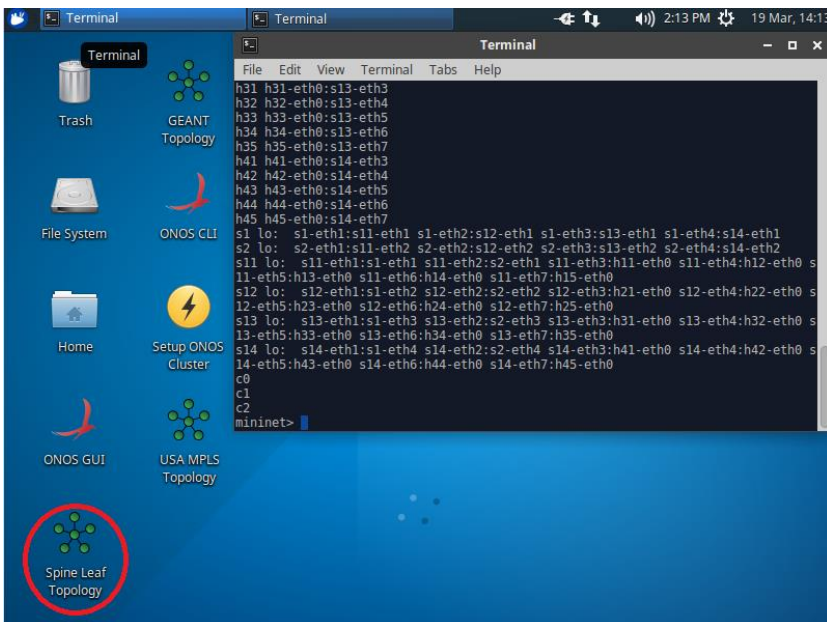
**h1 ping -c 5 h2:** Envía cinco paquetes de ping desde el host "h1" al host "h2". Puedes especificar diferentes opciones de ping según tus necesidades de prueba.

## Desplegando la primer topología de red en mininet

Para iniciar ONOS ejecutamos el programa “Setup ONOS Cluster” desde el escritorio de la máquina virtual



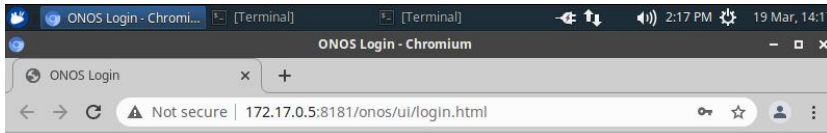
Ejecute el programa “Spine Leaf Topology” también ubicado en el escritorio de la máquina virtual, este programa se encarga de lanzar una topología spine-leaf dentro de mininet la cual está compuesta por 20 host, 6 switch y 3 controladores.



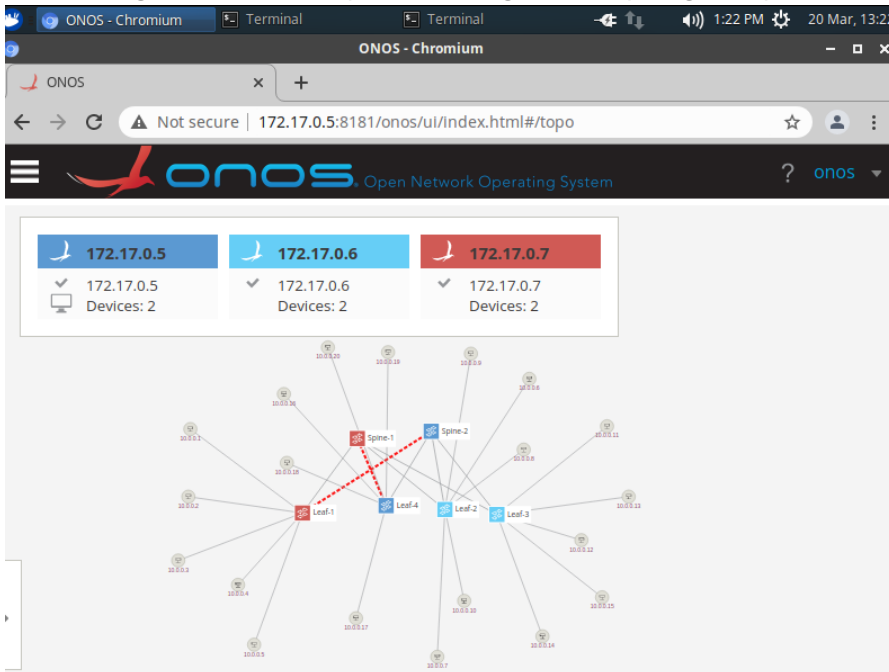
Ejecute la “ONOS GUI” desde el escritorio de la máquina virtual, esta interfaz gráfica de usuario es una herramienta de administración basada en web que proporciona una forma visual de interactuar y administrar el controlador SDN ONOS. Para acceder use las credenciales:

*username:onos*

*password:rocks*



Una vez ingrese le deberá aparecer la siguiente topología en pantalla:



Para obtener información sobre la red puede hacer click sobre los dispositivos o enlaces lo cual mostrará una ventana con información al respecto

## Extraer información de ONOS a través de su API

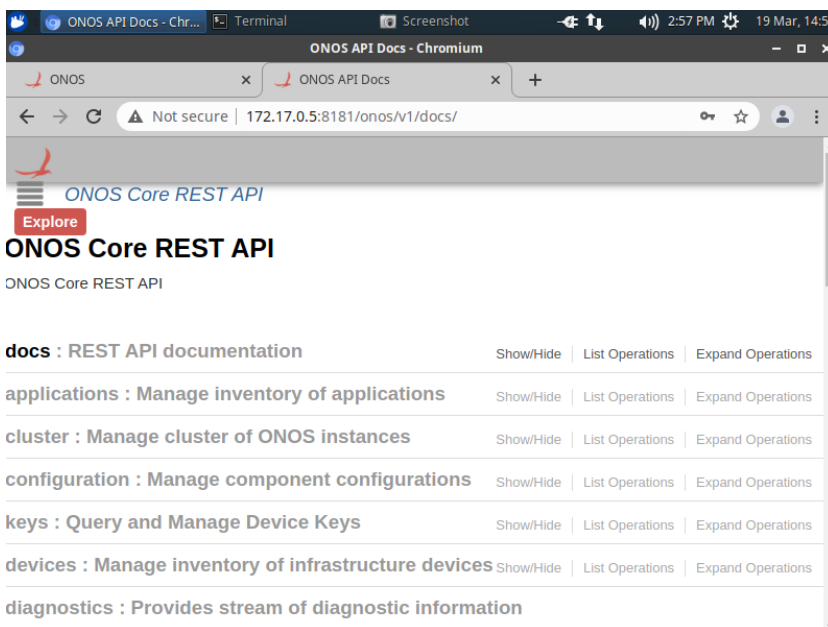
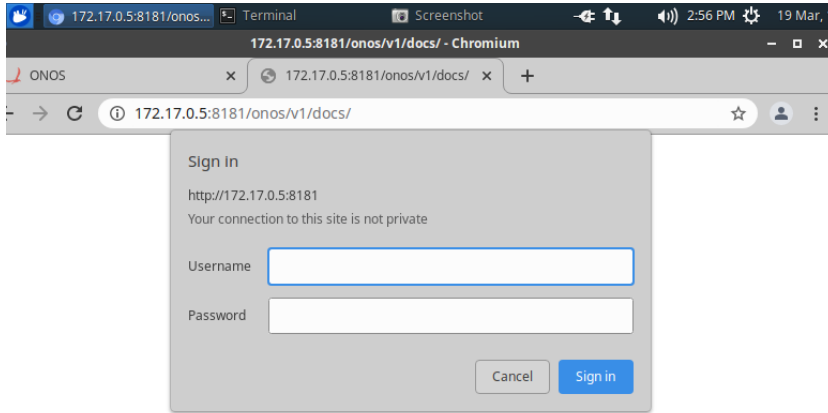
ONOS cuenta con una API REST que permite obtener información de la red:

En este caso puede acceder a la documentación de la API a través de la url

<http://172.17.0.5:8181/onos/v1/docs>

username:onos

password:rocks



A continuación se mostrará un ejemplo en el cual se extrae información de la red a través de un script escrito en Python.

Para esto se requiere instalar la biblioteca *requests* :

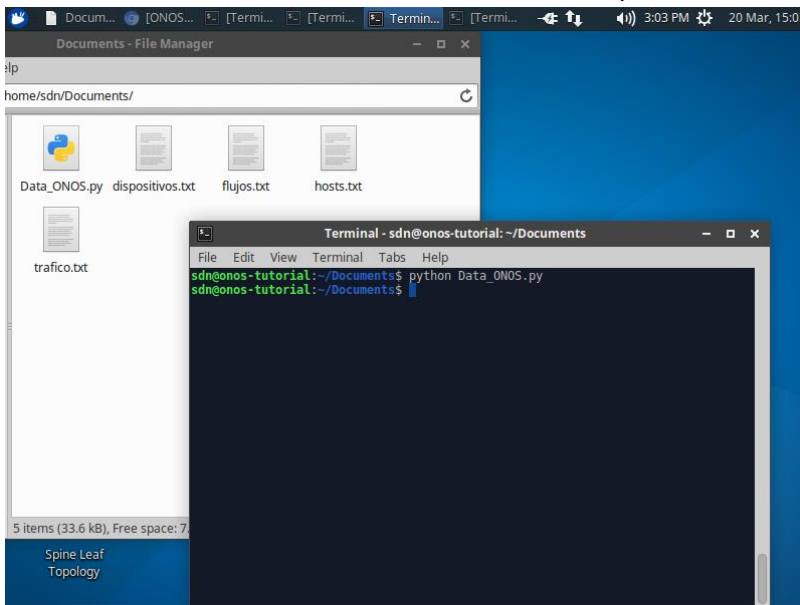
```
pip3 install requests
```

Descargue el script `Data_ONOS.py` del siguiente enlace [] y guárdelo en cualquier directorio

Desde la Shell de mininet debe generar tráfico, para esto puede usar el comando *pingall*

```
Terminal
File Edit View Terminal Tabs Help
3, s12) (h24, s12) (h25, s12) (h31, s13) (h32, s13) (h33, s13) (h34, s13) (h35,
s13) (h41, s14) (h42, s14) (h43, s14) (h44, s14) (h45, s14) (s11, s1) (s11, s2)
(s12, s1) (s12, s2) (s13, s1) (s13, s2) (s14, s1) (s14, s2)
*** Configuring hosts
h11 h12 h13 h14 h15 h21 h22 h23 h24 h25 h31 h32 h33 h34 h35 h41 h42 h43 h44 h45
*** Starting controller
c0 c1 c2
*** Starting 6 switches
s1 s2 s11 s12 s13 s14 ...
*** Waiting for switches to connect
s1 s2 s11 s12 s13 s14
*** Sending a gratuitous ARP from each host
h11 h12 h13 h14 h15 h21 h22 h23 h24 h25 h31 h32 h33 h34 h35 h41 h42 h43 h44 h45
*** Starting CLI:
mininet>
mininet>
mininet>
mininet> ping all
*** Unknown command: ping all
mininet> pingall
*** Ping: testing ping reachability
h11 -> x x x x x x x x x x x x x x x x x
h12 -> x x x x x x x x x x x x x x x x x
h13 -> x x
```

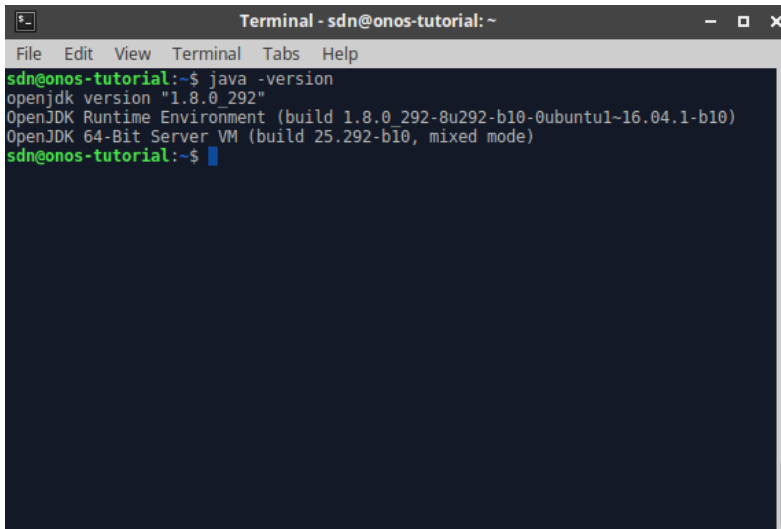
Luego desde otra Shell ejecute el script. Se crearán en el mismo directorio cuatro archivos de texto llamados dispositivos, flujos, hosts y tráfico. En estos estará guardada en formato JSON la información solicitada a la API a través del script.



## MININET + SFLOW

Para utilizar s-flow en conjunto con mininet inicie instalando java en la maquina virtual, escriba el siguiente comando en el Shell:

```
sudo apt install default-jdk
```



```
Terminal - sdn@onos-tutorial: ~
File Edit View Terminal Tabs Help
sdn@onos-tutorial:~$ java -version
openjdk version "1.8.0_292"
OpenJDK Runtime Environment (build 1.8.0_292-8u292-b10-0ubuntu1~16.04.1-b10)
OpenJDK 64-Bit Server VM (build 25.292-b10, mixed mode)
sdn@onos-tutorial:~$
```

A continuación instale el colector sFlow-RT

```
wget https://inmon.com/products/sFlow-RT/sflow-rt.tar.gz
```

```
tar -xvzf sflow-rt.tar.gz
```

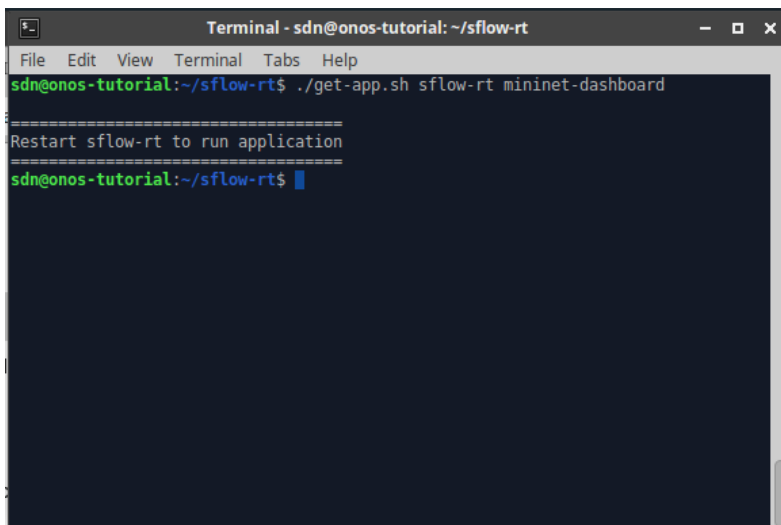
```
cd sflow-rt
```

Instale 2 apps para el colector que le permitirán una mejor visualización de las métricas

Inicie instalando un dashboard:

```
cd sflow-rt
```

```
./get-app.sh sflow-rt mininet-dashboard
```



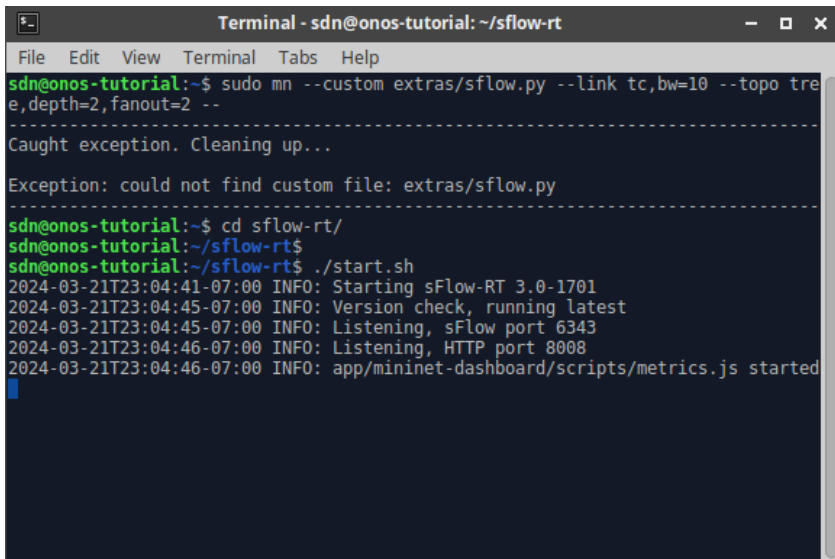
```
Terminal - sdn@onos-tutorial: ~/sflow-rt
File Edit View Terminal Tabs Help
sdn@onos-tutorial:~/sflow-rt$ ./get-app.sh sflow-rt mininet-dashboard
=====
Restart sflow-rt to run application
=====
sdn@onos-tutorial:~/sflow-rt$
```

Ahora instale el navegador de flujos y de métricas

```
sudo ./get-app.sh sflow-rt browse-metrics
```

```
sudo ./get-app.sh sflow-rt browse-flows
```

Inicie sflow-rt ejecutando el archivo `./start.sh` desde el directorio `/sflow-rt`

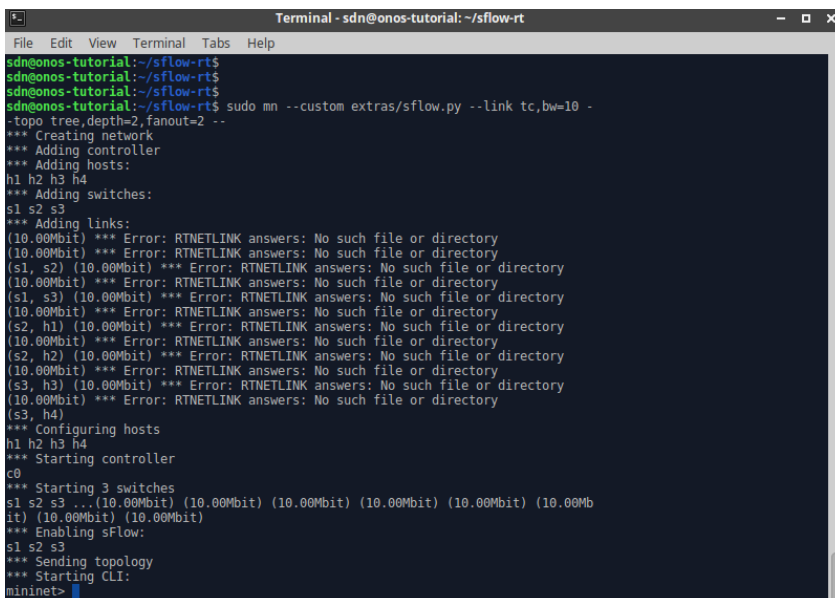


```
Terminal - sdn@onos-tutorial: ~/sflow-rt
File Edit View Terminal Tabs Help
sdn@onos-tutorial:~$ sudo mn --custom extras/sflow.py --link tc,bw=10 --topo tree,depth=2,fanout=2 --
-----
Caught exception. Cleaning up...

Exception: could not find custom file: extras/sflow.py
-----
sdn@onos-tutorial:~$ cd sflow-rt/
sdn@onos-tutorial:~/sflow-rt$
sdn@onos-tutorial:~/sflow-rt$ ./start.sh
2024-03-21T23:04:41-07:00 INFO: Starting sFlow-RT 3.0-1701
2024-03-21T23:04:45-07:00 INFO: Version check, running latest
2024-03-21T23:04:45-07:00 INFO: Listening, sFlow port 6343
2024-03-21T23:04:46-07:00 INFO: Listening, HTTP port 8008
2024-03-21T23:04:46-07:00 INFO: app/mininet-dashboard/scripts/metrics.js started
```

Lance una topología sencilla de mininet configurando sflow en los dispositivos

```
sudo mn --custom extras/sflow.py --link tc,bw=10 --topo tree,depth=2,fanout=2 --
```

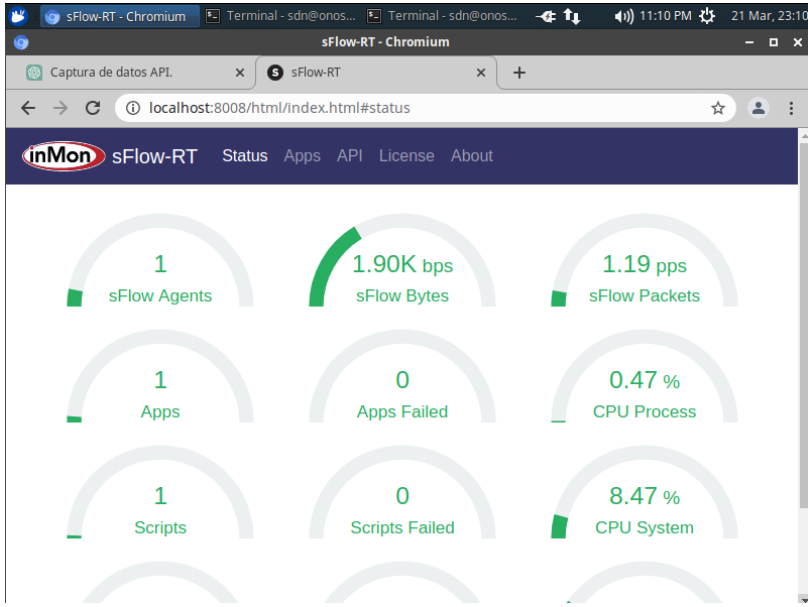


```
Terminal - sdn@onos-tutorial: ~/sflow-rt
File Edit View Terminal Tabs Help
sdn@onos-tutorial:~/sflow-rt$
sdn@onos-tutorial:~/sflow-rt$
sdn@onos-tutorial:~/sflow-rt$
sdn@onos-tutorial:~/sflow-rt$ sudo mn --custom extras/sflow.py --link tc,bw=10 --
-topo tree,depth=2,fanout=2 --
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2 h3 h4
*** Adding switches:
s1 s2 s3
*** Adding links:
(10.00Mbit) *** Error: RTNETLINK answers: No such file or directory
(10.00Mbit) *** Error: RTNETLINK answers: No such file or directory
(s1, s2) (10.00Mbit) *** Error: RTNETLINK answers: No such file or directory
(10.00Mbit) *** Error: RTNETLINK answers: No such file or directory
(s1, s3) (10.00Mbit) *** Error: RTNETLINK answers: No such file or directory
(10.00Mbit) *** Error: RTNETLINK answers: No such file or directory
(s2, h1) (10.00Mbit) *** Error: RTNETLINK answers: No such file or directory
(10.00Mbit) *** Error: RTNETLINK answers: No such file or directory
(s2, h2) (10.00Mbit) *** Error: RTNETLINK answers: No such file or directory
(10.00Mbit) *** Error: RTNETLINK answers: No such file or directory
(s3, h3) (10.00Mbit) *** Error: RTNETLINK answers: No such file or directory
(10.00Mbit) *** Error: RTNETLINK answers: No such file or directory
(s3, h4)
*** Configuring hosts
h1 h2 h3 h4
*** Starting controller
c0
*** Starting 3 switches
s1 s2 s3 ... (10.00Mbit) (10.00Mbit) (10.00Mbit) (10.00Mbit) (10.00Mbit) (10.00Mbit)
(10.00Mbit) (10.00Mbit)
*** Enabling sFlow:
s1 s2 s3
*** Sending topology
*** Starting CLI:
mininet>
```

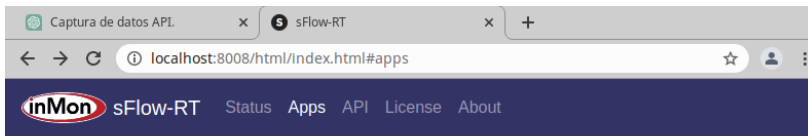
Vaya en el navegador a la siguiente dirección

<http://localhost:8008/html/index.html#status>

Le aparecerá la siguiente ventana



En la pestaña APPS aparecerán las funcionalidades instaladas



Click on an application in the list to access the application's home page:

- [browse-flows](#)
- [browse-metrics](#)
- [mininet-dashboard](#)

Visit [Applications](#) for information on downloading applications.

## WIRESHARK

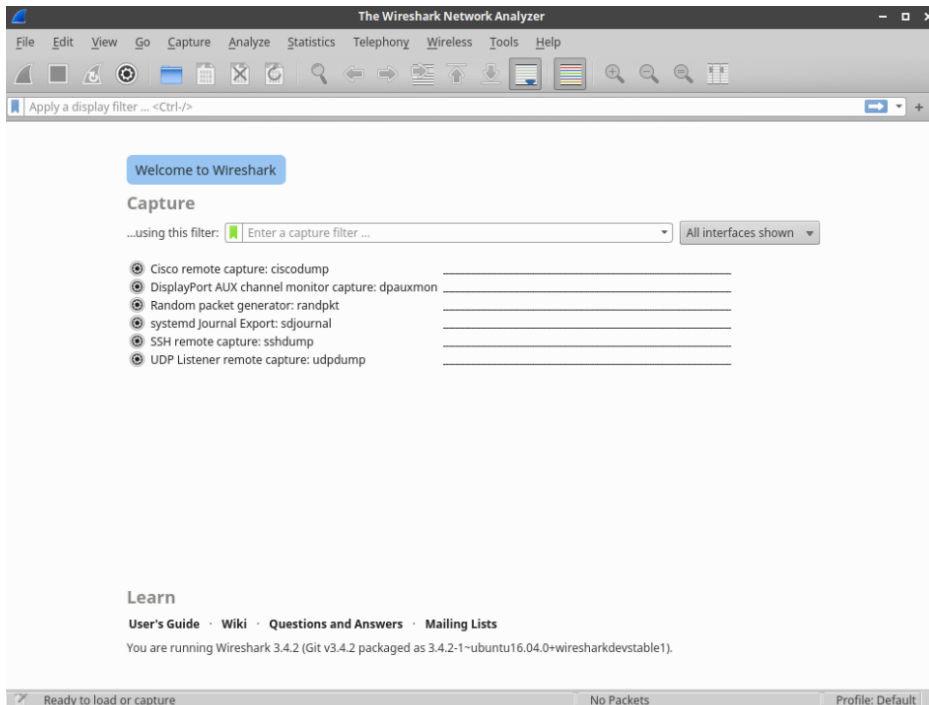
Instale y lance wireshark desde la Shell con los siguientes comandos:

```
sudo add-apt-repository ppa:wireshark-dev/stable
```

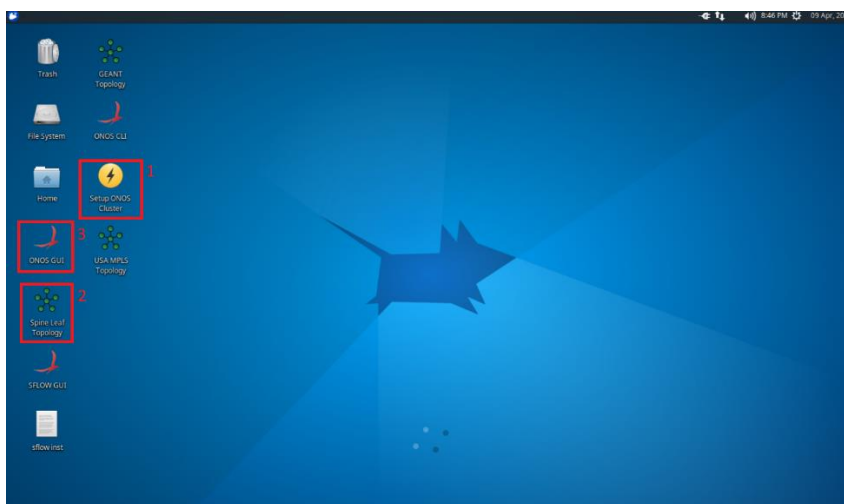
```
sudo apt update
```

```
sudo apt install wireshark -y
```

```
sudo wireshark
```



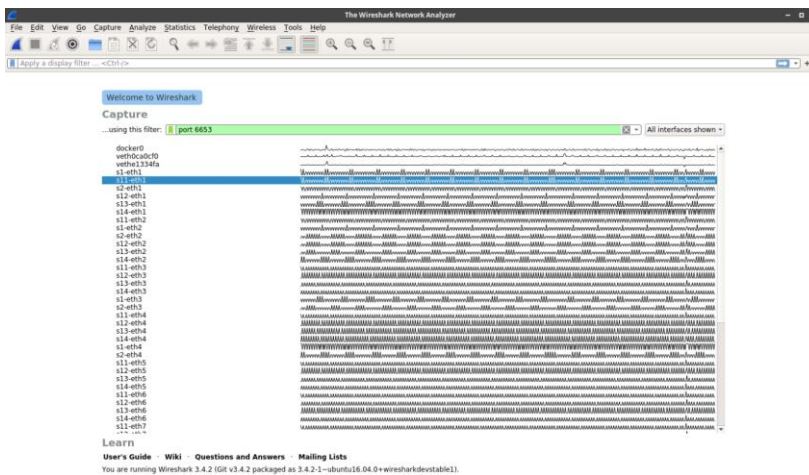
Ejecute el setup ONOS cluster, el spine leaf topology y por último el ONOS GUI



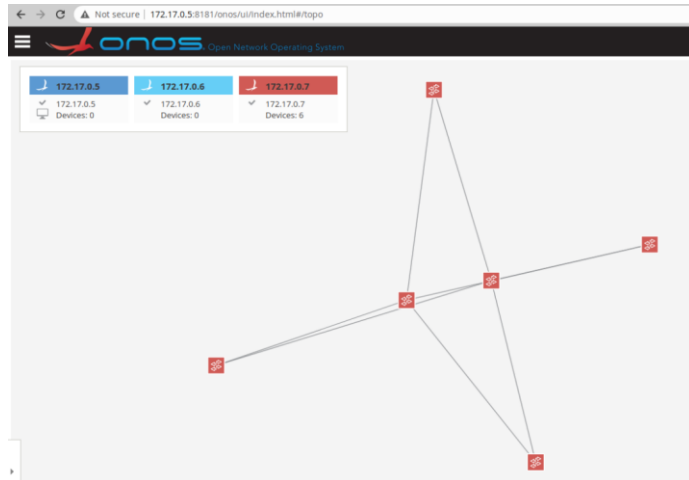
Al desplegar la topología se puede observar la dirección IP y el puerto asociado a la interfaz de cada controlador, esto será útil para interceptar el tráfico con wireshark

```
Terminal
File Edit View Terminal Tabs Help
*** Killing stale mininet node processes
pkill -9 -f mininet:
*** Shutting down stale tunnels
pkill -9 -f Tunnel=Ethernet
pkill -9 -f .ssh/mn
rm -f ~/.ssh/mn/*
*** Cleanup complete
*** Adding controllers
Connecting to remote controller at 172.17.0.5:6653
c0 (172.17.0.5)
Connecting to remote controller at 172.17.0.6:6653
c1 (172.17.0.6)
Connecting to remote controller at 172.17.0.7:6653
c2 (172.17.0.7)
*** Creating network
*** Adding hosts:
h11 h12 h13 h14 h15 h21 h22 h23 h24 h25 h31 h32 h33 h34 h35 h41 h42 h43 h44 h45
*** Adding switches:
s1 s2 s11 s12 s13 s14
*** Adding links:
(h11, s11) (h12, s11) (h13, s11) (h14, s11) (h15, s11) (h21, s12) (h22, s12) (h23, s12) (h24, s12) (h25, s12) (h31, s13) (h32, s13) (h33, s13) (h34, s13) (h35, s13) (h41, s14) (h42, s14) (h43, s14) (h44, s14) (h45, s14) (s11, s1) (s11, s2) (s12, s1) (s12, s2) (s13, s1) (s13, s2) (s14, s1) (s14, s2)
```

Para capturar el tráfico con wireshark definimos el puerto 6653 e iniciamos la captura de tráfico

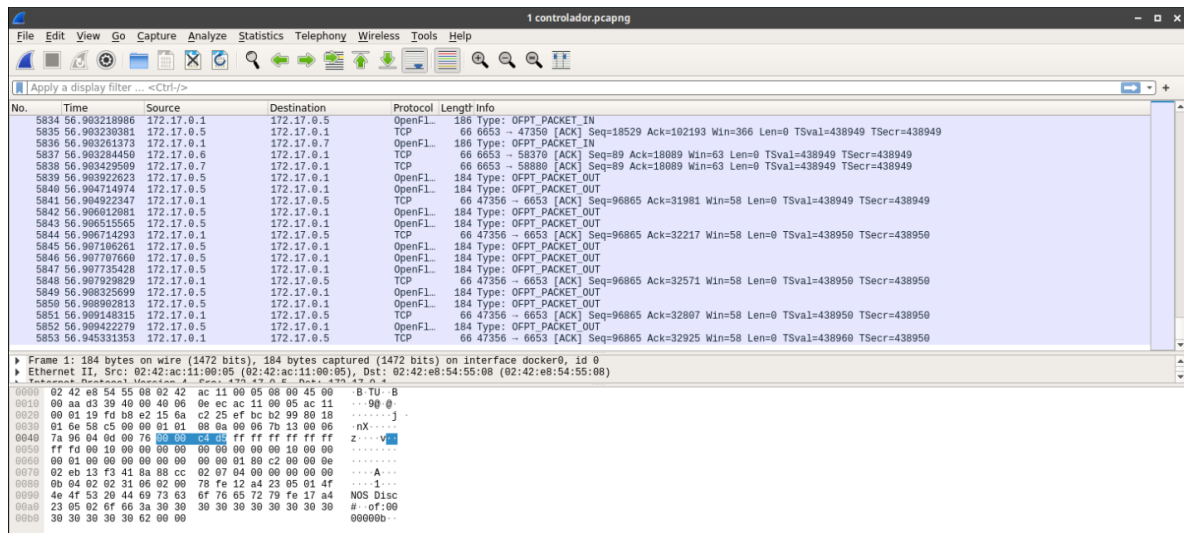


Generalmente la topología iniciará asignando todos los switches a un mismo controlador.

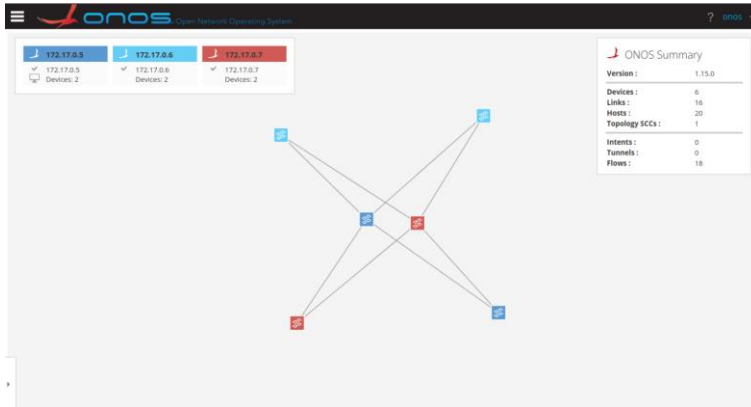


Genere algo de tráfico entre switches para capturarlo con wireshark. Puede escribir los siguientes comandos en la Shell de mininet:

```
s1 ping -c 5 s2
s1 ping -c 5 s11
s1 ping -c 5 s12
s1 ping -c 5 s13
s1 ping -c 5 s14
iperf s1 s2
```



Abra nuevamente la GUI de ONOS y presione la tecla E para distribuir uniformemente los 6 switches en los 3 controladores



Genere nuevamente tráfico entre los switches y capture los paquetes