

CÓDIGO – MATERIA	: xxxxx – Redes Definidas por Software.
REQUISITO	: Redes de Computadores II.
PROGRAMA - SEMESTRE	: Ingeniería Telemática - 8
PERIODO ACADÉMICO	: 2018-1
INTENSIDAD SEMANAL	: 3 HORAS
CRÉDITOS	: 3

Contenido

Laboratorio – Flujos de Tráfico.....	2
Objetivo Práctica Laboratorio	2
Requisitos Laboratorio	2
Preparación ambiente ejecución VND	2
Instalar servidor web (Apache):	3
Instalar MySQL	4
Instalar PHP	4
Instalación VND	5
Creación y configuración flujos de tráfico.....	7
Creación Red.	7
Configuración flujos de tráfico	10
Configuración enlace troncal.	12
Pruebas en Mininet.	15
Consideraciones adicionales.	15
Informe Laboratorio	16
Referencias.....	16

Laboratorio – Flujos de Tráfico

Objetivo Práctica Laboratorio

CONFIGURAR y ANALIZAR los diferentes flujos de tráfico presentes en una red definida por software (SDN), utilizando un Virtual Network Descriptor (VND).

Requisitos Laboratorio

Para el correcto desarrollo de este laboratorio es necesario contar con lo siguiente:

- Máquina virtual del Laboratorio - Preparación Ambiente.
- Mininet.
- Acceso a Internet

Actividades Laboratorio

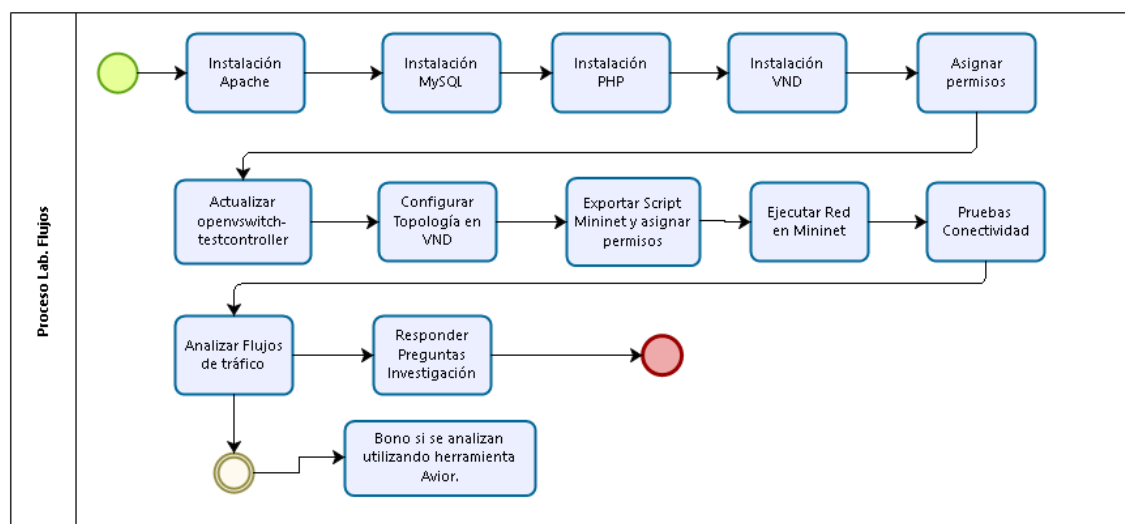


Diagrama 1. Proceso Laboratorio

Preparación ambiente ejecución VND

Un Visual Network Descriptor (VND) es una aplicación web, que cuenta con una interfaz gráfica, que permite la creación de escenarios genéricos para validar redes definidas por software (SDN). La versión de VND que se utilizará en este laboratorio, es posible crear redes SDN y desplegarlas rápidamente en la aplicación Mininet a través de scripts (fontes, 2014).

Previo a la instalación del VND, va a ser necesario instalar una serie de componentes primero.

Instalar servidor web (Apache):

Podemos instalar Apache fácilmente desde el gestor de paquetes de Ubuntu, apt.

- `sudo apt-get update`
- `sudo apt-get install apache2`

A continuación, agregamos una sola línea al archivo `/etc/apache2/apache2.conf` para suprimir un mensaje de advertencia.

- `sudo nano /etc/apache2/apache2.conf`
- `ServerName dominio_del_servidor_o_IP`

Como se aprecia en la siguiente captura de pantalla:

```
# Include of directories ignores editors' and dpkg's backup files,
# see README.Debian for details.

# Include generic snippets of statements
IncludeOptional conf-enabled/*.conf

# Include the virtual host configurations|:
IncludeOptional sites-enabled/*.conf

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet

ServerName 127.0.0.1
```

Imagen 1. apache2.conf

Ahora reiniciamos el servicio de apache y validamos que los cambios se han configurado adecuadamente con los siguientes comandos:

- `sudo systemctl restart apache2`
- `sudo apache2ctl configtest`

Por último, deberemos habilitar las conexiones a nuestro servidor web de apache con el siguiente comando:

- `sudo ufw allow in "Apache Full"`

Instalar MySQL

Ahora bien, debemos instalar una base de datos mysql, para lograr que el VND funcione de manera adecuada. Podemos utilizar el gestor de paquetes de Ubuntu para instalar MySQL de la siguiente forma:

- `sudo apt-get update`
- `sudo apt-get install mysql-server`

Luego, procedemos a configurar la base de datos, utilizando el siguiente comando (podemos dejar las opciones por defecto):

- `mysql_secure_installation`

Valida el estado del servicio de MySQL con el siguiente comando:

- `systemctl status mysql.service`

Instalar PHP

PHP es el componente de nuestra configuración que procesará el código del VND para mostrar el contenido de manera dinámica. Podemos utilizar el gestor de paquetes de Ubuntu para instalar MySQL de la siguiente forma:

- `sudo apt-get install php libapache2-mod-php php-mcrypt php-mysql`

Ahora bien, para probar que el servicio PHP funcione de manera adecuada vamos a crear un archivo info.php de la siguiente manera:

- `sudo nano /var/www/html/info.php`

Modificamos el nuevo archivo así:

```
<?php
phpinfo();
?>
```

Al acceder a la siguiente dirección: <http://127.0.0.1/info.php> veremos el siguiente sitio web:

<div> <div>127.0.0.1/info.php</div> <div>Search</div> </div> <div> <div>PHP Version 7.0.22-0ubuntu0.16.04.1</div> <div>php</div> </div>	
System	Linux user-VirtualBox 4.10.0-38-generic #42~16.04.1-Ubuntu SMP Tue Oct 10 16:32:20 UTC 2017 x86_64
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.0/apache2
Loaded Configuration File	/etc/php/7.0/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/7.0/apache2/conf.d
Additional .ini files parsed	/etc/php/7.0/apache2/conf.d/10-mysqld.ini, /etc/php/7.0/apache2/conf.d/10-opcache.ini, /etc/php/7.0/apache2/conf.d/10-pdo.ini, /etc/php/7.0/apache2/conf.d/20-calendar.ini, /etc/php/7.0/apache2/conf.d/20-ctype.ini, /etc/php/7.0/apache2/conf.d/20-exif.ini, /etc/php/7.0/apache2/conf.d/20-fileinfo.ini, /etc/php/7.0/apache2/conf.d/20-ftp.ini, /etc/php/7.0/apache2/conf.d/20-gettext.ini, /etc/php/7.0/apache2/conf.d/20-iconv.ini, /etc/php/7.0/apache2/conf.d/20-json.ini, /etc/php/7.0/apache2/conf.d/20-mcrypt.ini, /etc/php/7.0/apache2/conf.d/20-mysql.ini, /etc/php/7.0/apache2/conf.d/20-pdo_mysql.ini, /etc/php/7.0/apache2/conf.d/20-phar.ini, /etc/php/7.0/apache2/conf.d/20-posix.ini, /etc/php/7.0/apache2/conf.d/20-readline.ini, /etc/php/7.0/apache2/conf.d/20-shmop.ini, /etc/php/7.0/apache2/conf.d/20-sockets.ini, /etc/php/7.0/apache2/conf.d/20-sysmsg.ini, /etc/php/7.0/apache2/conf.d/20-sysvsem.ini, /etc/php/7.0/apache2/conf.d/20-sysvshm.ini, /etc/php/7.0/apache2/conf.d/20-tokenizer.ini
PHP API	20151012
PHP Extension	20151012
Zend Extension	320151012
Zend Extension Build	API320151012.NTS
PHP Extension Build	API20151012.NTS
Debug Build	no

Imagen 2. PHP funcionando adecuadamente.

Instalación VND.

Una vez tengamos instalados todos los servicios necesarios para el VND, podemos proceder con la instalación del VND desarrollado por el ingeniero Ramon Fontes. La herramienta fue desarrollada para que estudiantes y profesionales, puedan practicar las funcionalidades de las redes SDN, además, cuenta con las siguientes herramientas:

- Creación de escenarios de red SDN a través de GUI.
- Creación automática de Mininet Scripts.
- Creación automática de scripts de Openflow Controllers.
- Creación automática de archivos NSDL.

Primero accedemos al repositorio en github (<https://github.com/ramonfontes/vnd-sdn-version>) y clonamos el proyecto a nuestro ambiente virtual:

- `git clone https://github.com/ramonfontes/vnd-sdn-version.git`

Una vez terminada la clonación del proyecto, debemos copiar el proyecto al siguiente directorio: `/var/www/html`, como se aprecia a continuación:

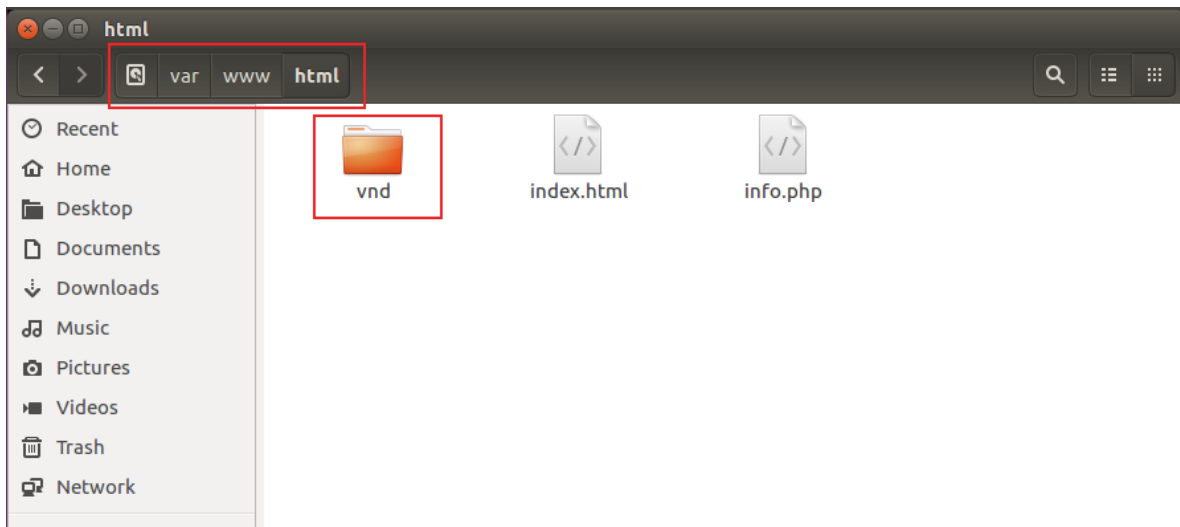


Imagen 3. Proyecto VND en directorio web.

Por último, debemos dar permisos de ejecución al directorio `/bin-debug/scripts` dentro del proyecto vnd. Recordar utilizar el comando `chmod`.

Una vez hayamos terminado estos procesos, podemos acceder a la aplicación a través del enlace: <http://127.0.0.1/vnd/bin-debug/main.html>, como se aprecia a continuación:

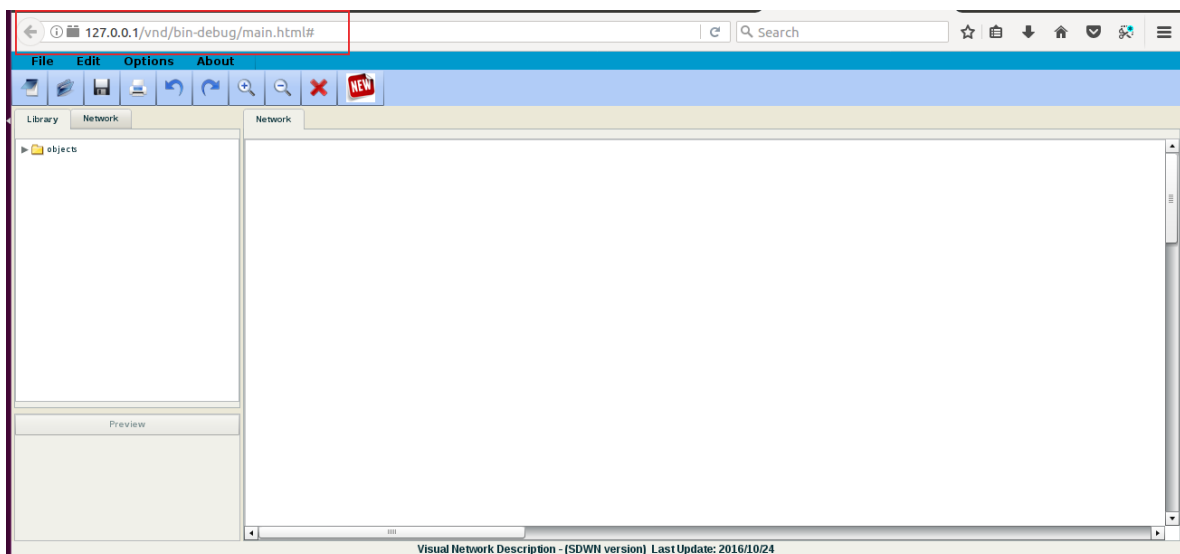


Imagen 4. Aplicación VND operando.

Por último, para lograr que nuestra red pueda ser ejecutada correctamente en la herramienta Mininet, deberemos instalar un controlador específico para los Open vSwitch.

```
sudo apt-get install openvswitch-testcontroller
```

Una vez terminada la instalación del Open vSwitch test-controller, pasamos a crear un enlace simbólico a al controlador:

```
sudo ln /usr/bin/ovs-testcontroller /usr/bin/controller
```

Creación y configuración flujos de tráfico.

Creación Red.

Ingresa a la herramienta VND y construya el siguiente diagrama de red, utilizando los nodos y enlaces ubicados en la librería de objetos:

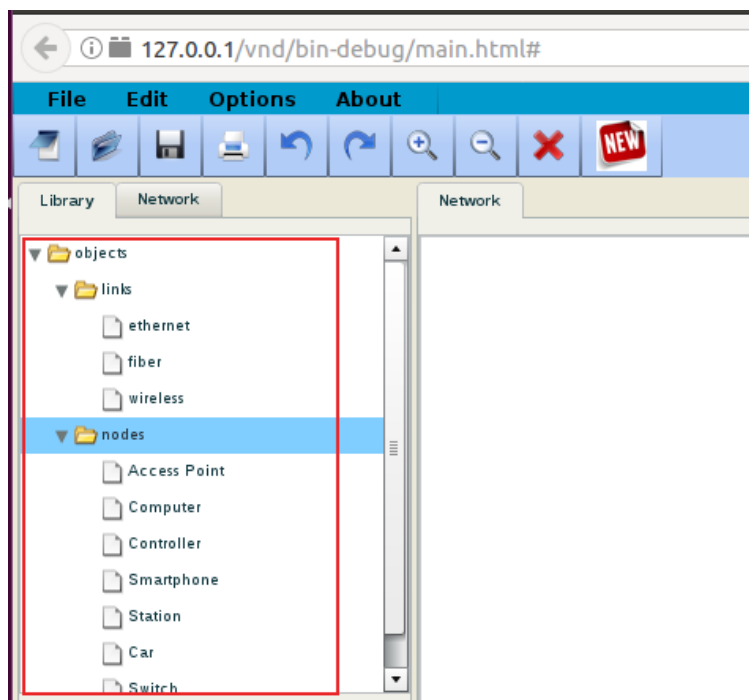


Imagen 5. Librería Objetos.

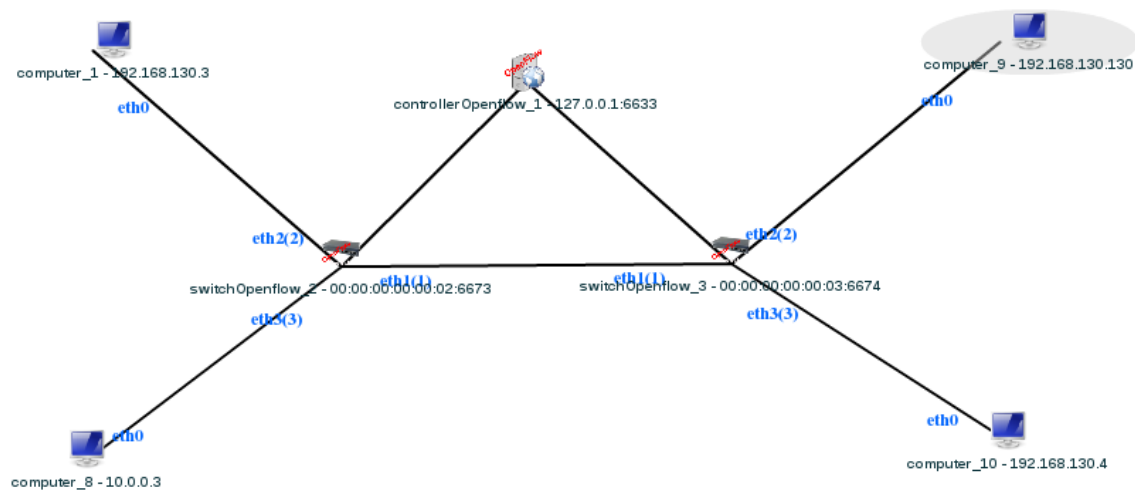


Diagrama 2. Topología Red

En este escenario, se establecen tres subredes:

- Red de Administración (VLAN 10): 192.168.130.0/25
- Red de Tecnología (VLAN 12): 192.168.130.128/26
- Red aislada (VLAN 2): 10.0.0.0/8

Ingresamos a cada nodo y configuramos la dirección IP y las máscaras de subred correspondiente:

The image shows a window titled "Configuration" with three tabs: "Configuration", "Quality of Service", and "Queue Configuration". The "Configuration" tab is active. Inside the window, there is a sub-header "(Configuration)" and a title "equipo 1 - 192.168.130.130". Below this, there are several fields for configuring a host:

- id**: computer_1
- name**: equipo 1
- active**: true (with a dropdown arrow)
- Computer IP Address**: 192.168.130.130
- Computer Mask**: 255.255.255.128 (with a dropdown arrow)
- Computer MAC Address**: 00:00:00:00:00:01
- notes**: Vlan Administración

Below these fields is an "Apply Settings" button. At the bottom, there is a "Position" section with "x:" and "y:" coordinates, both set to 101 and 54 respectively.

Imagen 6. Configuración Hosts

Una vez configurados los 4 equipos mostrados en la topología de red, ingresamos al controlador y modificamos el campo Openflow Controller por la opción ovs-controller y cambiamos el puerto al 6653.

Configuration

(Configuration)

Controller_1 - 127.0.0.1:6653

id: Controller_1

name: Controller_1

active: true

Controller IP Address: 127.0.0.1

Controller Port: 6653

Remote Controller: ☐ yes ☒ no

Openflow Controller: ovs-controller

Apply Settings Create Flow Table

Position x: 220.65 y: 13.3

Imagen 7. Configuración Controlador.

Una vez configurados los parámetros básicos del controlador y los hosts de la red, se requiere configurar los flujos de información con el objetivo de comunicar los dispositivos que están en diferentes segmentos de la red (físicos o lógicos); lo que se quiere es poder comunicar los diferentes equipos de una VLAN creando un enlace troncal entre ambos switches.

Configuración flujos de tráfico

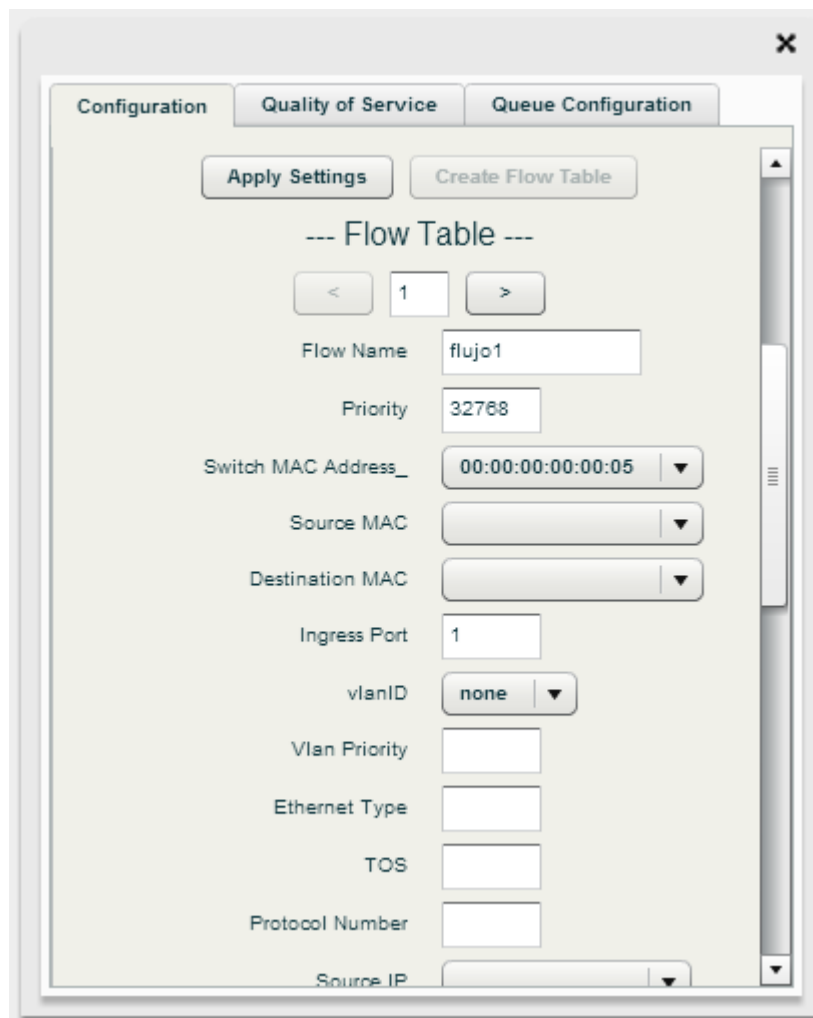
Ingresa al componente que representa al controlador floodlight y de click en la opción **create flow table**; esta opción le permitirá al usuario crear los flujos que serán procesados en los switches de la red.

En esta sección del controlador, se configuran los siguientes 3 parámetros:

- Nombre del flujo.
- La prioridad del flujo: que en este caso se deja el valor por defecto.

- Se selecciona el switch al que se le creará el flujo.

Se especifica el puerto por donde se espera el flujo.



The screenshot shows a configuration window titled 'Flow Table' with three tabs: 'Configuration', 'Quality of Service', and 'Queue Configuration'. The 'Configuration' tab is active. At the top, there are two buttons: 'Apply Settings' and 'Create Flow Table'. Below these, the 'Flow Table' section is displayed. It includes a navigation bar with '<', '1', and '>' buttons. The configuration fields are as follows:

- Flow Name: flujo1
- Priority: 32768
- Switch MAC Address_: 00:00:00:00:00:05 (dropdown)
- Source MAC: (dropdown)
- Destination MAC: (dropdown)
- Ingress Port: 1
- vlanID: none (dropdown)
- Vlan Priority: (empty field)
- Ethernet Type: (empty field)
- TOS: (empty field)
- Protocol Number: (empty field)
- Source IP: (dropdown)

Imagen 8. Configuración flujos.

Ahora, configuramos las acciones. En esta sección del controlador, se configuran 2 parámetros:

- El número de salidas por donde pasará el flujo.
- Se establece el ID de la VLAN. Para el escenario propuesto, la VLAN de Administración se le ha asignado como ID el número 10.

Configuration Quality of Service Queue Configuration

----- Actions -----

Set Output 3

Set Source MAC

Set Destination MAC

Set Enqueue

Set Vlan ID 10

Set Vlan Priority

Set Strip Vlan

Set TOS

Set Source IP

Set Destination IP

Set Source TCP/UDP Port

Set Destination TCP/UDP Port

notes

Apply Settings

Imagen 9. Configurando acciones.

Al finalizar damos clic en la opción “Apply Settings”. Una vez hecho esto, el controlador nos pedirá crear el siguiente flujo. Créelo en caso de ser necesario y si necesita comunicar otras VLANs.

- ❓ Reflexione: ¿Por qué es necesario configurar los flujos y las acciones que tendrá cada uno?
¿Esta configuración debe realizarse siempre que se habla de una red SDN?

Configuración enlace troncal.

Para configurar un enlace troncal entre ambos switches, debemos configurar un nuevo flujo. En esta sección el proceso es similar al especificado en la configuración de los flujos de tráfico y seleccionamos los siguientes parámetros.

- Nombre del flujo.
- La prioridad del flujo: que en este caso se deja el valor por defecto.
- Se selecciona el switch al que se le creará el flujo.
- Se especifica el puerto por donde se espera el flujo.
- Se especifica el ID de la VLAN, que se quiere comunicar.

The image shows a network configuration window titled "Flow Table". It has three tabs: "Configuration", "Quality of Service", and "Queue Configuration". The "Configuration" tab is active. The window contains the following fields and values:

Field	Value
Flow Name	flujo3
Priority	32768
Switch MAC Address_	00:00:00:00:00:05
Source MAC	
Destination MAC	
Ingress Port	3
vlanID	10
Vlan Priority	
Ethernet Type	
TOS	
Protocol Number	
Source IP	
Destination IP	

Imagen 10. Flujo enlace troncal.

Ahora pasaremos a configurar las acciones del nuevo flujo, nuestro objetivo es establecer una sola salida, que es la que se conecta al otro switch de nuestra topología de red. No se establece VLAN ID y se especifica el Strip VLAN, el cual permite etiquetar las subredes cuando se establece una comunicación por el canal. Seleccione **yes** para el etiquetado de las VLAN.

Configuration Quality of Service Queue Configuration

----- Actions -----

Set Output 1

Set Source MAC

Set Destination MAC

Set Enqueue

Set Vlan ID none

Set Vlan Priority

Set Strip Vlan yes

Set TOS

Set Source IP

Set Destination IP

Set Source TCP/UDP Port

Set Destination TCP/UDP Port

notes

Apply Settings

Imagen 11. Acciones Flujo troncal

El proceso que acaba de hacer, aplíquelo para el otro switch. Una vez realizada toda la configuración en el controlador, exporte el escenario como archivo .sh, para poder ejecutarlo en mininet.

Pruebas en Mininet.

Una vez exportado el script de mininet, le damos permisos de ejecución y lo ejecutamos. Al finalizar nos aparece el siguiente output en la consola:

```
user@user-VirtualBox:~/mininet-projects$ sudo ./lab-flujos-sdn-1.sh
*** Creating nodes
*** Creating links
*** Starting network
*** Configuring hosts
h7 h8 h9 h10
*** Running CLI
*** Starting CLI:
mininet>
```

Imagen 12. Ejecución ambiente Mininet.

- ❓ Realice las pruebas de conectividad en todos los hosts. ¿Es posible que todos los hosts se vean entre ellos? Cualquiera sea su respuesta a la pregunta anterior, ¿Por qué ocurre esta situación? ¿Cuáles son las razones solo existe visibilidad entre algunos hosts y no con todos?
- ❓ Analice los flujos de cada uno de los switches en la herramienta Mininet. Para esta actividad recomendamos instalar la aplicación Avior, que se integra con el controlador SDN utilizado, para tener una gestión gráfica de nuestra red SDN. Nota: Realizar esta actividad utilizando Avior, tendrá un bono sobre la nota final del laboratorio.
- ❓ ¿Qué ventajas ofrece implementar un controlador SDN en la creación de VLAN?
- ❓ ¿Qué diferencias puede mencionar con respecto a los switches capa 2 con los que ha trabajado anteriormente?

Consideraciones adicionales.

Puedes validar la configuración de los elementos de la red utilizando el comando dump:

```
mininet> dump
```

Es posible revisar los flujos utilizando los siguientes comandos:

```
$ sudo ovs-ofctl dump-flows s*
and
mininet> dpctl dump-flows
...
```

```
mininet> sh ovs-ofctl dump-flows s*
```

El proceso de instalación de Avior, lo encuentran en la siguiente Guía:
<https://github.com/1PhoenixM/avior-service/blob/master/README.md>

Informe Laboratorio

Elabore un informe de laboratorio donde se evidencie lo siguiente:

- Trabajo realizado durante el laboratorio (instalaciones, configuraciones, diseños, comandos, entre otros). Elabore un diagrama de flujo, a través del cual, logré reflexionar respecto a los procedimientos realizados, tenga en cuenta mencionar las dificultades encontradas y como trabajo para resolverlas.
- Responda cada una de las preguntas aquí planteadas, teniendo en cuenta los procesos que siguió para dar respuesta a cada pregunta.
- Conclusiones del trabajo realizado. Tenga en cuenta que las conclusiones deben ser una síntesis del informe de laboratorio presentado, por lo que, en esta sección se deben incluir los resultados, un análisis de lo aprendido y el resumen del informe incluyendo los procesos ejecutados.

Referencias

Linux Foundation. (2014). Open vSwitch, 31(5), 1–58.

fontes, r. (24 de sep de 2014). *github.com/ramonfontes*. Obtenido de
<https://github.com/ramonfontes/vnd-sdn-version>

Jarrin, A. (2014). *Desarrollo Guías de Laboratorio para la Implementación de redes SDN en el Laboratorio de Redes*. Cali.