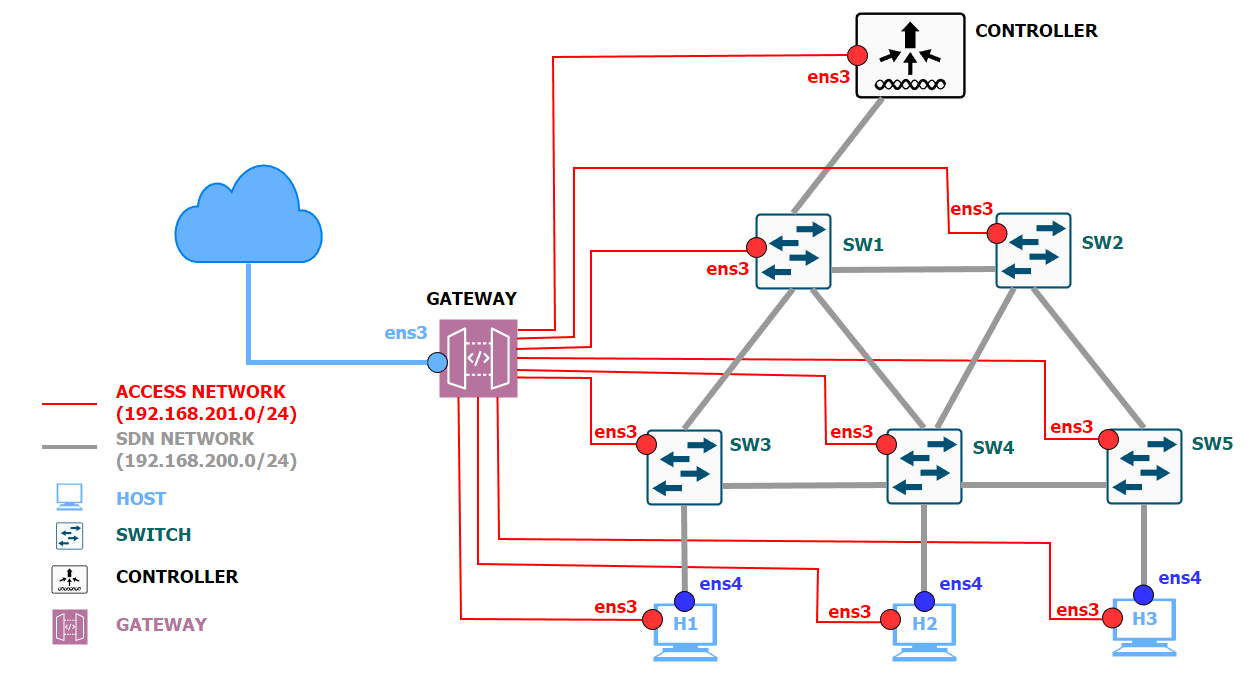
|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | | | |
| **TEL354 - REDES DEFINIDAS POR SOFTWARE (SDN)** | | | | | | | | | | |
|  | | | | | | | | | | |
|  | | | | | | | | | | |
| **Laboratorio N°:** | | | 4 |  |  | | **Semestre:** | 2025-2 |  |  |
|  | | | | | | | | | | |
| **Tema:** | | | Northbound API: Aplicación REST en Python | | | | | |  | |
|  | | | | | | | | | | |
| **Profesor:** Christian Quispe / César Santiváñez | | | | | | | | | | |
|  | | | | | | | | | | |
|  | | | | | | | | | | |
| **INDICACIONES GENERALES** | | | | | | | | | | |
|  | | | | | | | | | | |
| * Durante el laboratorio se puede usar el material provisto por los docentes y blogs de internet. | | | | | | | | | | |
| * Seguir atentamente las indicaciones de los jefes de práctica. | | | | | | | | | | |
|  | | | | | | | | | | |
|  | | | | | | | | | | |
| **OBJETIVOS** | | | | | | | | | | |
| * Adquirir herramientas necesarias para interactuar con el Northbound API (REST API) de un controlador SDN * Ser capaz de automatizar la interacción con un controlador SDN * Desarrollar una aplicación para el controlador Floodlight | | | | | | | | | | |
|  | | | | | | | | | | |
|  | | | | | | | | | | |
| **ACTIVIDADES A REALIZAR** | | | | | | | | | | |
|  | | | | | | | | | | |
|  | N° | ACTIVIDAD | | | | TIEMPO ESTIMADO | | PUNTAJE | |  |
|  | 1 | Experiencia de Laboratorio | | | | 1 hora y 30 minutos | | 5 ptos | |  |
|  | 2 | Informe Previo | | | | 7 días | | 5 ptos | |  |
|  | 3 | Cuestionarios | | | | 2 semanas | | 5 ptos | |  |
|  | 4 | Evaluación continua de PROYECTO / LABORATORIO | | | | 2 semanas | | 5 ptos | |  |

Jefe de práctica responsable: Juan Manuel Becerra Avila (a20090417@pucp.pe)

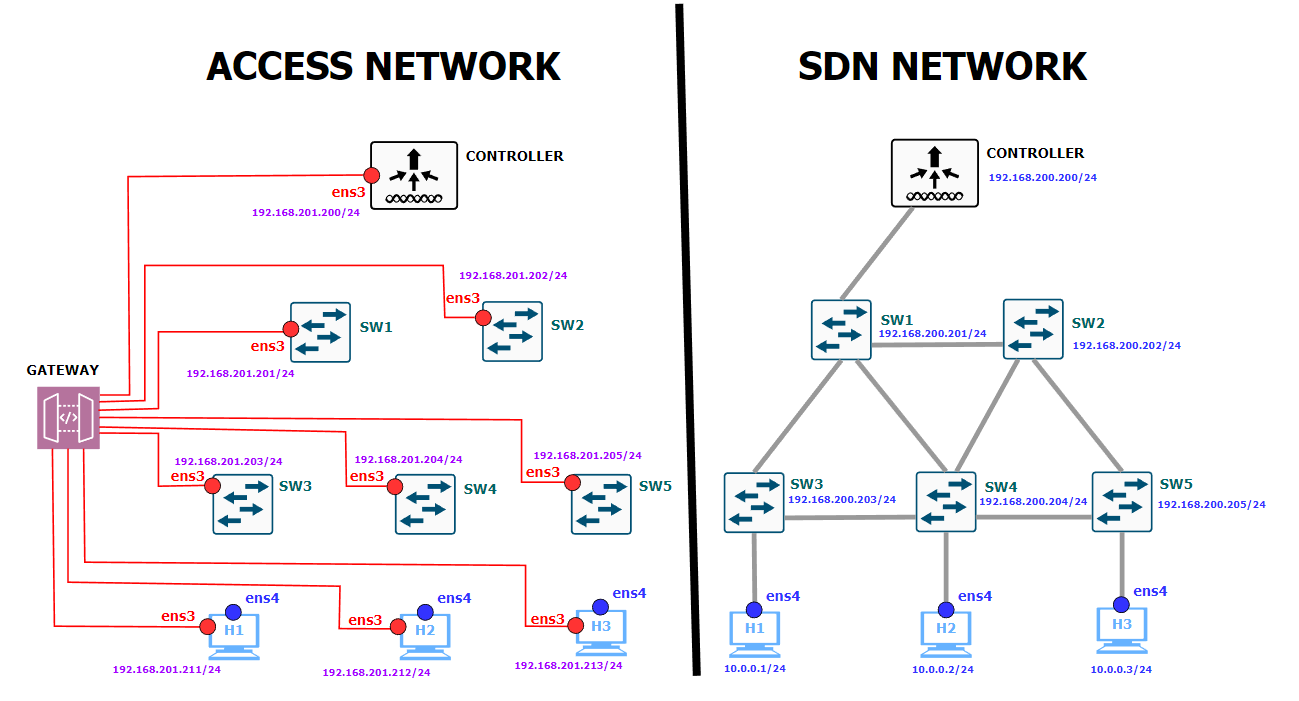
**ESCENARIO**

El Grupo de Investigación de Redes Avanzadas (GIRA-PUCP), perteneciente a la sección de Telecomunicaciones, cuenta con un emulador de redes avanzadas denominado VNRT, en el cual se ha desplegado la topología con la que se trabajará en el presente ciclo. A continuación se describen los componentes de la topología en cuestión:



Tal y como se puede observar, se tienen 10 máquinas virtuales distribuidas de la siguiente manera: 3 hosts (clientes de la red), 5 switches ([Open vSwitch](https://docs.openvswitch.org/en/latest/intro/what-is-ovs/)), un controlador([Floodlight Controller](https://floodlight.atlassian.net/wiki/spaces/floodlightcontroller/overview)) y un gateway (puerta de ingreso con accesos limitados).

Si desglosamos la topología mostrada líneas arriba, podemos identificar dos tipos de redes:



Para acceder a cada uno de los elementos dentro de la topología, es necesario establecer conexiones ssh a diferentes puertos previamente configurados.

|  |  |
| --- | --- |
| Virtual Machine | PORT |
| Gateway | 22 |
| CONTROLLER | 5800 |
| SW1 | 5801 |
| SW2 | 5802 |
| SW3 | 5803 |
| SW4 | 5804 |
| SW5 | 5805 |
| H1 | 5811 |
| H2 | 5812 |
| H3 | 5813 |

El comando a usar para tener un acceso vía ssh exitoso es:

|  |
| --- |
| $ ssh ubuntu@$ip\_gateway -p $port |

|  |
| --- |
| ✔ **Nota**  Las credenciales para acceder a la máquinas virtuales son:   * user: ubuntu * password: ubuntu |
| ✔ **Nota**  Debe reemplazar las variables **ip\_gateway** y **port** de acuerdo a su conveniencia. |

|  |
| --- |
| ✔ **Importante**  El usuario ubuntu tiene permisos de sudoers en todas las VMs a excepción del Gateway. Cualquier tipo de modificación dentro del Gateway está prohibida, recuerden que este dispositivo sólo será usado para acceder a sus demás VMs. |

**GUÍA DE LABORATORIO**

Indicaciones generales:

* Puntaje total: **5 pts**
* El desarrollo de la guía de laboratorio es personal. Cada alumno deberá presentar un archivo en formato PDF con el nombre TEL354\_RF\_LAB4\_[YYY].pdf, dónde YYY es su código PUCP. Este archivo contendrá sus respuestas así como cualquier código desarrollado que se le pida.
* Fecha máxima de entrega del IF: **Martes 07 de Octubre a las 20:00 hrs (Horario 0891) y Jueves 09 de Octubre a las 21:00 hrs (Horario 0892)**. No se podrán subir los informes después de dicha hora ni se recibirán por ningún otro medio.
* Subir el informe previo a la actividad de Clasroom correspondiente (LABORATORIO 4 - EXPERIENCIA DE LABORATORIO) con el siguiente formato: TEL354\_RF\_LAB4\_[YYY].pdf, donde [YYY] es su código PUCP.
* Es responsabilidad del alumno revisar y practicar los conceptos previos necesarios para el correcto desarrollo del laboratorio. Durante la sesión, no se dará asesoría respecto a temas asignados en el informe previo. Cualquier duda o consulta podrá ser resuelta por los jefes de práctica y/o profesor (previa coordinación) antes del laboratorio.

|  |
| --- |
|  |

**ACTIVIDAD 1: Desarrollo de aplicación REST en Python (5 pts)**

La universidad ha modernizado su red a una arquitectura SDN donde el acceso a los recursos está basado en roles de usuarios (estudiante, profesor, administrativo, investigador, etc.) asociados a sus objetivos de negocio. En particular, el acceso por parte de un alumno a un servidor está determinado por los cursos al que el alumno está matriculado en el semestre. Se le pide desarrollar una aplicación en python3 para el control de la red (creación de rutas) bajo el paradigma SDN. Use las siguientes suposiciones:

* La red está compuesta por switches openflow y un controlador Floodlight.
* El centro de estudios tiene cursos, alumnos matriculados, y servidores para cada curso, en los cuales se alojan servicios para el desarrollo de los cursos (p. ej. web, ftp, ssh)
* Por defecto no hay conectividad entre ningún host de la red. La aplicación debe permitir al administrador controlar el acceso de los alumnos a los servicios, de acuerdo a una serie de políticas declaradas previamente.
* Existe un servicio NAC que - al autorizar el acceso del usuario a la red - mapea la identidad del alumno a la dirección MAC del equipo que el alumno usa para conectarse. Ud. puede asumir que esta asociación es conocida (p.ej., es un atributo de la clase “alumno”).
* Modo reactivo: durante operación regular (no será implementada en este laboratorio), cuando un usuario intente comunicarse con un servidor, el primer paquete será interceptado y enviado al controlador, quien informará a la aplicación que el “usuario X (identificado por su MAC)” quiere comunicarse con el “servicio Y del servidor Z (identificado por su IP, protocolo, puerto)”. La aplicación instalará una ruta entre el usuario y el servidor si el alumno está matriculado a algún curso con acceso al servicio, caso contrario instalará una regla para dropear el paquete.
* Modo proactivo: la aplicación también permite instalar rutas “manualmente” por ejemplo para troubleshooting o para sesiones que sean muy sensibles a la latencia. Este laboratorio se enfocará en la implementación de esta función.

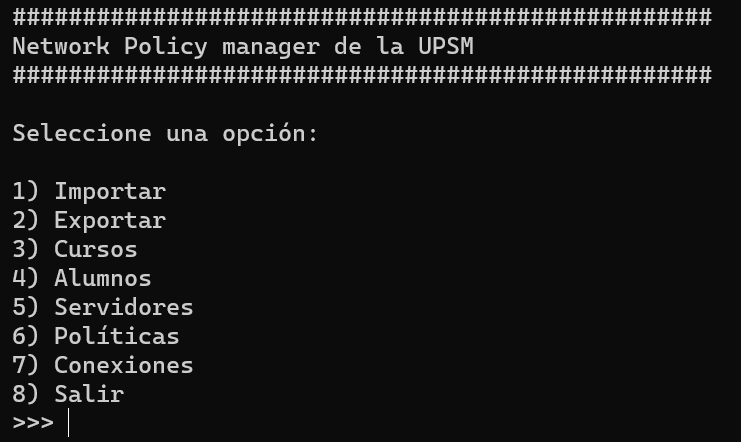
Lineamientos de la aplicación:

La interfaz de usuario de la aplicación tendrá un menú principal será un menú por línea de comandos que tendrá la siguiente estructura:

1. Importar
2. Exportar
3. Cursos
4. Alumnos
5. Servidores
6. Políticas
7. Conexiones

Por ejemplo:

Cada submenú tendrá opciones CRUD: Create, Read (lista todo, o ver atributo de solo uno, Update, Delete), como se muestra en la tabla siguiente. Se indica con un asterisco (\*) aquellas funciones que Ud. deberá implementar.



|  |  |  |
| --- | --- | --- |
| Menú | Opción | Descripción |
| Importar | Nombre de archivo (\*) | Importa los valores del archivo (formato YAML). |
| Exportar | Nombre de archivo |  |
| Cursos | Crear |  |
| Listar (\*) | Lista los cursos existentes y su estado |
| Mostrar detalle(\*) | Lista propiedades del curso, incluyendo a los alumnos |
| Actualizar (\*) | Agregar o eliminar a un alumno |
| Borrar |  |
| Alumnos | Crear |  |
| Listar (\*) | Muestra a los alumnos existentes (considere filtros) |
| Mostrar detalle (\*) | Muestra las propiedades del alumno (código, nombre, y MAC) |
| Actualizar |  |
| Borrar |  |
| Servidores y servicios | Crear |  |
| Listar (\*) | Mostrar los servidores existentes (nombre e IP) |
| Mostrar detalle(\*) | Muestra los servicios que brinda el servidor |
| Actualizar |  |
| Borrar |  |
| Conexiones | Crear (\*) | Crear las rutas entre los alumnos y los servicios permitidos en los cursos matriculados. Asigna un “handler” a la conexión. |
| Listar (\*) | Muestra la conexiones creadas manualmente |
| Mostrar detalle | Mostrar la ruta |
| Recalcular | Recalcular la ruta |
| Actualizar | Reemplaza la ruta actual con la nueva ruta calculada |
| Borrar (\*) | Eliminar la conexión identificada por su “handler”. |

La aplicación será capaz de exportar/importar su estado en un archivo formato YAML con la sintaxis mostrada en la figura siguiente.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  31323334353637383940414243444546 | # Bloque de alumnos  alumnos:  - nombre: Juan Perez  codigo: 20012482  mac: 00:44:11:22:44:A7:2A  - nombre: John Smith  codigo: 20041321  mac: 00:44:11:2F:33:D8:3C  - nombre: Luisa Marvel  codigo: 20080621  mac: 00:44:11:3F:22:C3:77  # Bloque de cursos:  cursos:  - codigo: TEL354  estado: DICTANDO  nombre: Redes Definidas por Software  alumnos:  - 20012482  - 20041321  servidores:  - nombre: Servidor 1  servicios\_permitidos:  - ssh  - codigo: TEL123  estado: INACTIVO  nombre: Telepathy and Telekinetics  - 20080621  - 20041321  servidores:  - nombre: Servidor 1  servicios\_permitidos:  - web  # Bloque de servidores / servicios  servidores:  - nombre: "Servidor 1"  ip: 10.0.0.3  servicios:  - nombre: ssh  protocolo: TCP  puerto: 23  - nombre: web  protocolo: TCP  puerto: 80 |

Recomendaciones:

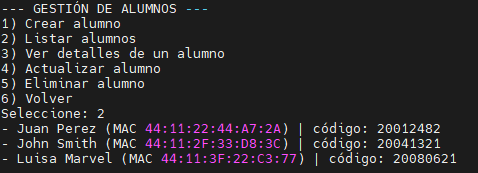
* Verifique que el módulo de reactive routing del controlador Floodlight esté desactivado.
* Utilice las clases y funciones implementadas en las actividades 1, 2 y 3 del Informe Previo para leer y almacenar los datos de ambos archivos en los objetos de tipo alumno, curso y servidor
* Para la creación y eliminación de conexiones, se debe validar que el alumno esté autorizado. En caso que no lo esté, devolver un mensaje de error.
* Un alumno se considera autorizado para acceder a un servicio X en el servidor Y si y sólo si está matriculado en algún curso con las siguientes características:
  + estado = DICTANDO
  + tiene al servidor Y en su lista de servidores
  + tiene al servicio X como parte de los “servicios permitidos” para el servidor Y.
* Para la creación de las rutas, considere lo siguiente:
* Utilice las funciones implementadas en las actividades 1, 2 y 3 para hallar la ruta entre 2 host, compuesto por una secuencia de DPID y puertos
* Defina una función que reciba la ruta, y los datos de la conexión e inserte los flows necesarios para habilitar la conectividad entre el alumno y el servicio solicitado.
* Utilice match en L2, L3, y L4, en ambos sentidos (host->servidor, y servidor->host), permitiendo sólo los protocolos y puertos necesarios
* No olvide los flows para ARP.
* Utilizar la siguiente estructura en su programa (puede añadir más funciones):

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  2526 | # imports  class alumno:  # implementado en informe previo  class curso:  # implementado en informe previo    class servidor:  # implementado en informe previo  def get\_attachment\_point():  # implementado en informe previo  def get\_route():  # implementado en informe previo    def build\_route():  # reciba la ruta los datos de conexión e inserte los flow entry necesarios  def menu():  # implemente el menu    def main():  # cargue los archivos, instancie las clases y llame al menú |

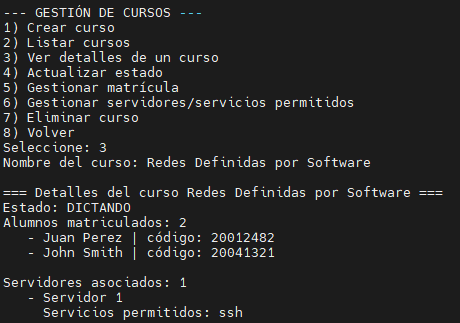
Adjunte el link a un repositorio Github (nombre del repositorio con formato *TEL354\_LAB4\_<CODIGO\_ALUMNO>*) el cual contenga su código desarrollado. Luego ejecute las siguientes acciones y adjunte snapshots de la ejecución de las mismas.

> Link de repositorio GitHUb: ***https://github.com/Xolmo/TEL354\_LAB4\_20220378.git***

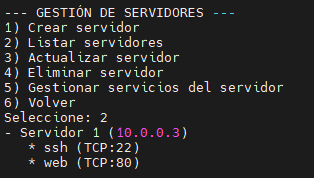
1. Importe el archivo database.yaml que se encuentra en la carpeta ‘Reporte Final’
2. Liste a todos los alumnos.



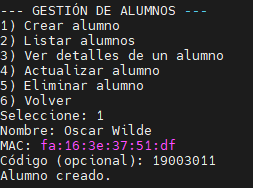
1. Liste los alumnos que están en curso TEL354.



1. Liste los servidores y luego muestre los detalles del “Servidor 1”. Verifique que la IP del “Servidor 1” sea la misma IP que la asignada a la VM H3 en su topología.



1. Agregue al alumno “Oscar Wilde”, con código “19003011” y la MAC igual a la MAC de la interfaz de la VM H1 (con lo que informa al sistema que el alumno está actualmente conectado a través de la máquina H1).



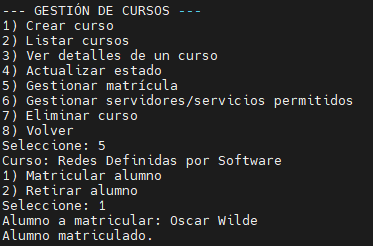
1. Alumno “Oscar Wilde” inicia una sesión SSH con el “Servidor 1” (es decir, intente una conexión SSH desde la VM H1 a la VM H3). Indique el resultado

No se puede establecer una conexión.

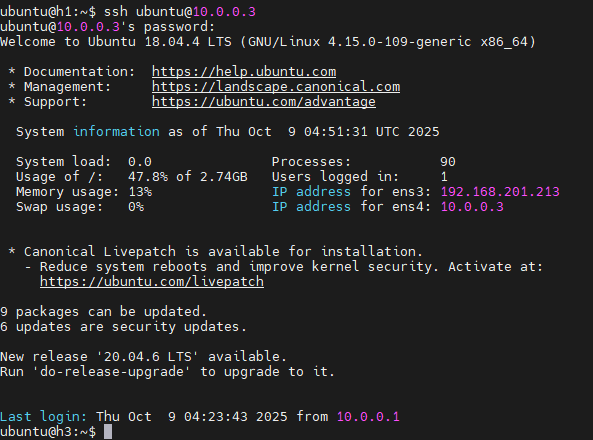
1. Liste los cursos que tienen acceso al servicio SSH en el “Servidor 1”



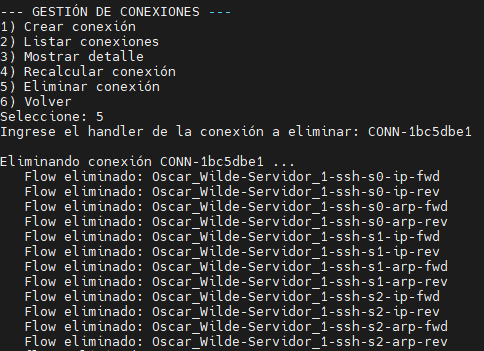
1. Agregue a “Oscar Wilde” al curso TEL354



1. Reintente la conexión SSH de la VM H1 a la VM H3 “Servidor 1”. Indique el resultado e inserte un snapshot.



1. Borre la conexión creada entre “Oscar Wilde” y el “Servidor 1”.



1. Reintente 10) indicando el resultado e insertando otro snapshot.

