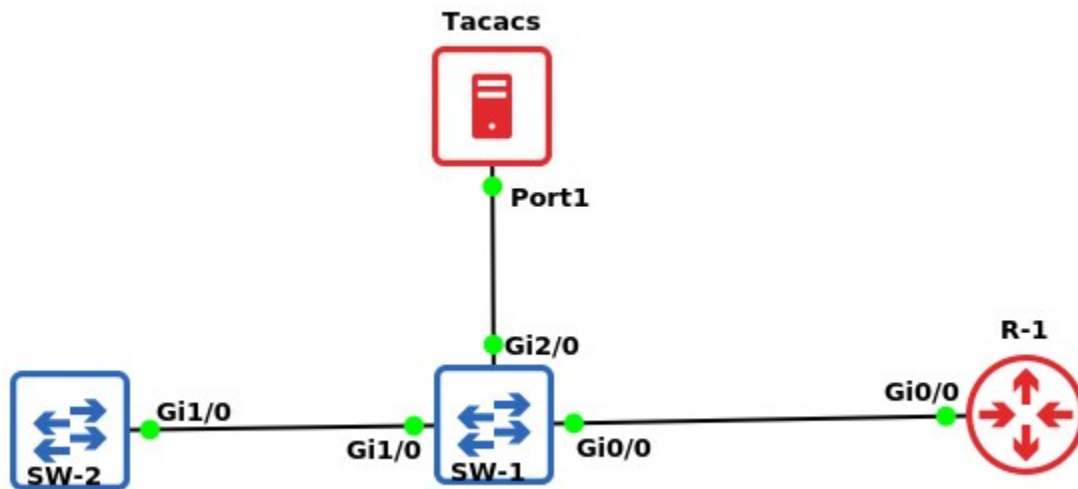


# Práctica 3



## Objetivos de la Práctica.

Crear un Script que permita la configuración los dispositivos de capa 2 y capa 3 de la topología para que tengan acceso a través del servicio de Tacacs+. Deberá de manipulas archivos de texto como también crear algunas funciones.

## Actividades

En el área de trabajo conecte un dispositivo “**Cloud**” hacia “**SW-2**” que tenga por interface el “**bridge**” (**switch virtual**) correspondiente que usted tiene configurado en su equipo físico para que se permita la conexión a los Switches y Routers, ejemplo “**virtbr1**”. Utilice “**nmtui**” para activar y/o desactivar el bridge para su correcto funcionamiento.

Realice previamente las pruebas de conectividad “**PING**” y que también son alcanzarlos por el protocolo “**SSH**”. **Nota:** Resuelva primeramente estos contratiempos antes de avanzar.

Ejemplo:

```
ping 192.168.100.254
ssh -l python 192.168.100.254
-----
username: python
password: pythonautomationcourse
enable: pythonautomationcourse
-----
```

Una vez en el área de trabajo el “**Cloud**” con conectividad a los dispositivos, generar un Script realice las siguientes acciones:

0.- Realice las actividades del archivo **“TacacsConfiguration.pdf”**

1.- Cree un archivo con nombre **“ip.txt”** en la raíz de su proyecto con las direcciones de los dispositivos:

**192.168.100.1**

**192.168.100.2**

**192.168.100.254**

2.- Cree un archivo con nombre **“commandos.txt”** que contendra la configuración para que los dispositivos de capa 2 y 3.

**(Vea el archivo Config-Tacacs-on-Routers-and-Switches.txt que se encuentra en la Practica3).**

3.- Cree la funcion **main** y mandela a llamar.

4.- Cree una funcion que permita limpiar la pantalla de la terminal.

Para esta actividad requiere importar de la funcion **“os”** los modulos **“name”** y **“system”**. Siga las recomendaciones de **PEP8** para la creacion de funciones.

Esta funcion debera ser llamada **“clear\_screen”**.

Debe de funcionar tanto para **Windows** como para **GNU/Linux**.

5.- El archivo **“ip.txt”** deberá de ser leído y se guardara cada linea en una variable de tipo Lista con nombre **“ip\_list”**.

6.- Cree una funcion que permita realizar **pings** a los dispositivos e integrela al **“main.py”**

7.- Determine que dispositivos estan activos y cuales no. Guarde esta informacion en archivos con nombre **“{}\_file.txt”.format(ip)**. Cada archivo deberá de tener la frase:

**El Dispositivo {} esta vivo.**

**El Dispositivo {} no esta vivo.**

8.- El archivo **“commandos.txt”** deberá de ser leído y se guardara cada linea en una variable de tipo Lista con nombre **“commandos\_list”**.

9.- Cree el respectivo diccionario para utilizar la libreria **Netmiko**.

10.- Ingrese en modo privilegiado al Router y/o Switch.

11.- Envie la lista de los comandos **“comandos\_list”** al o a los dispositivos que estan activos.

12. Salve los cambios generados en los dispositivos.

13.- Ahora realice pruebas de conectividad con los nombres de usuarios y password que creo en el servidor **“Tacacs+”**.

**(Vea el Archivo “TacacsConfiguration.pdf”).**

14.- Comparta su repositorio de **GitHub**.