

# **RAPPORT DE LABORATOIRE**

## **1 SWITCH & VLAN**

GAUCHER ADRIEN

## TABLE DES MATIÈRES

1   Présentation du laboratoire .....	3
2   Réalisation pas à pas .....	4
2.1   Définition des adresses IP .....	4
2.2   Configuration des switchs .....	4
2.3   Configuration du trunk .....	6
2.4   Création des VLAN .....	6
3   Conclusion .....	8

## 1 | Présentation du laboratoire

Le but de cet exercice est de configurer 2 VLAN sur 2 switches.  
Les switches seront connectés entre eux via un trunk.  
Chaque VLAN sera composé d'un serveur et d'un ordinateur portable.

Un VLAN (**V**irtual **L**ocal **A**rea **N**etwork) permet de partitionner un switch en plusieurs parties qui seront isolées les unes des autres.

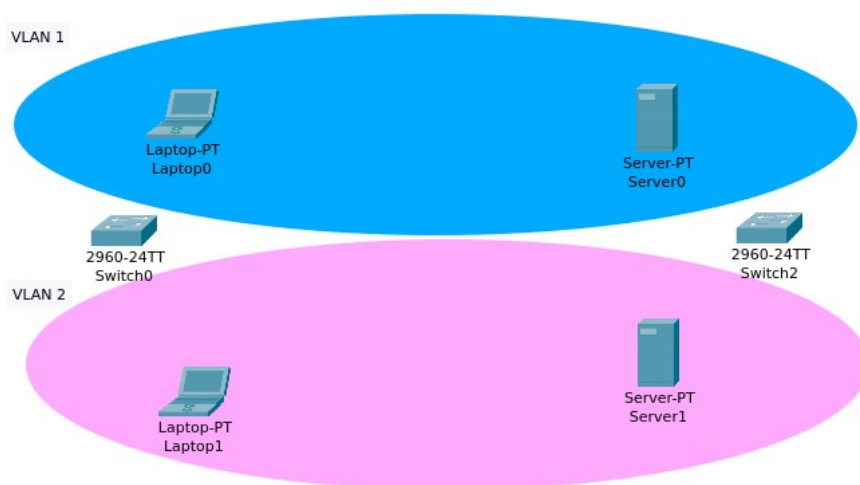
Cette segmentation se situe au niveau logicielle et non physique.

Cette technologie implique différents concepts et protocoles qui seront mis en pratique dans ce laboratoire.

Ce travail sera réalisé sur le logiciel de simulation réseau « *Cisco Packet Tracer* ».

Toutes les images de ce rapport proviennent du logiciel de simulation.

Le visuel suivant présente le réseau et ses différents composants.

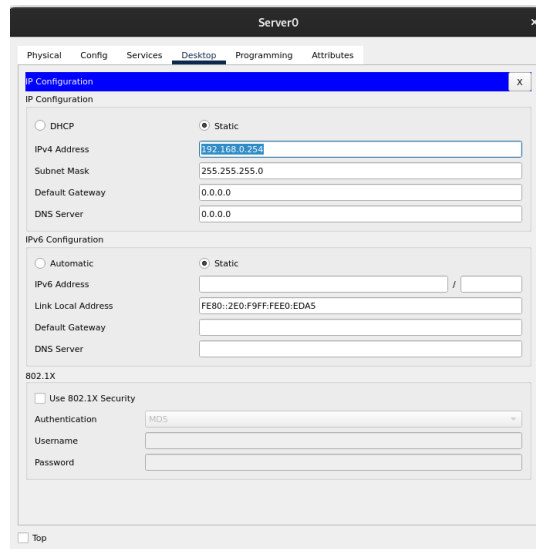


## 2 | Réalisation pas à pas

### 2.1 | Définition des adresses IP

Pour chaque machines présentes sur le réseau (ordinateurs), il faut configurer une adresse IP. Sans cela, impossible pour elles de communiquer.

L'image ci dessous illustre la procédure a réaliser pour la machine *Server0*.



On peut remarquer que le masque de sous réseau est renseigné automatiquement en fonction de l'adresse IP.

### 2.2 | Configuration des switches

Premièrement les switches sont reliés entre eux via des cables sur les ports 23 et 24.

Nous pouvons constater la présence du protocole *spanning tree*.

C'est lui qui va empêcher une boucle de communication entre les switches en fermant un port sur le *switch2*.

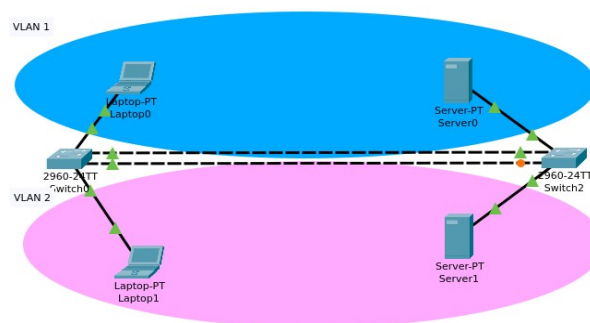


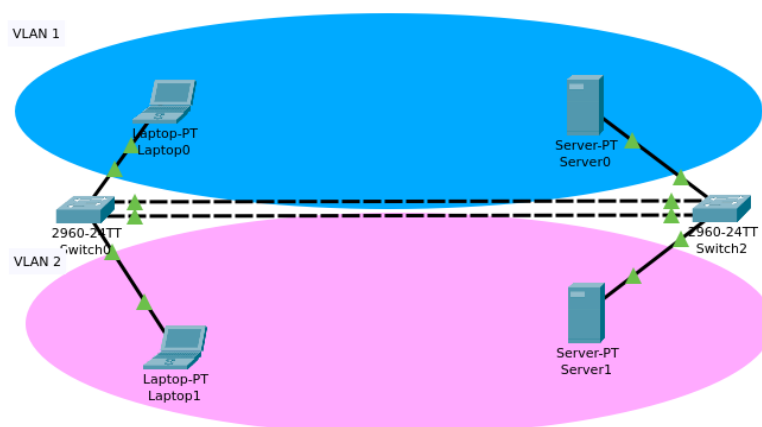
Illustration du spanning tree

Maintenant nous pouvons passer à la configuration des interfaces sur les switches.  
L'agrégation des ports 23 et 24 sera réalisée grâce au protocole *LACP*.  
Cela sert regrouper plusieurs ports d'un même switch en un seul, pour augmenter la bande passante et assurer une redondance en cas de problème sur un port.  
Pour réaliser cette opération il nous faudra rentrer dans les switches et écrire des commandes.  
La photo suivante illustre cette manipulation.

The screenshot displays the CLI interface of a switch named "Switch0". The tabs at the top are Physical, Config, CLI (selected), and Attributes. Below the tabs, it says "IOS Command Line Interface".

```
Switch>  
Switch>  
Switch>  
Switch>  
Switch>  
Switch>  
Switch>  
Switch>  
Switch>  
Switch>  
Switch>  
Switch>en  
Switch#conf t  
Enter configuration commands, one per line. End with CNTL/Z.  
Switch(config)#host Switch0  
Switch(config)#int range fast/23-24  
Switch0(config-if-range)#channel-group 1 mode active  
Switch0(config-if-range)#  
Creating a port-channel interface Port-channel 1  
  
LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/23, changed state to down  
  
LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/23, changed state to up  
  
LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/24, changed state to down  
  
LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/24, changed state to up  
  
Switch0(config-if-range)#channel-protocol lacp  
Switch0(config-if-range)#sh run  
  
% Invalid input detected at '^' marker.  
  
Switch0(config-if-range)#sh config  
A
```

Nous pouvons constater que l'opération est un succès, le protocole *spanning tree* n'est plus actif. L'agrégat est effectif, les ports 23 et 24 sont considérés comme un seul.

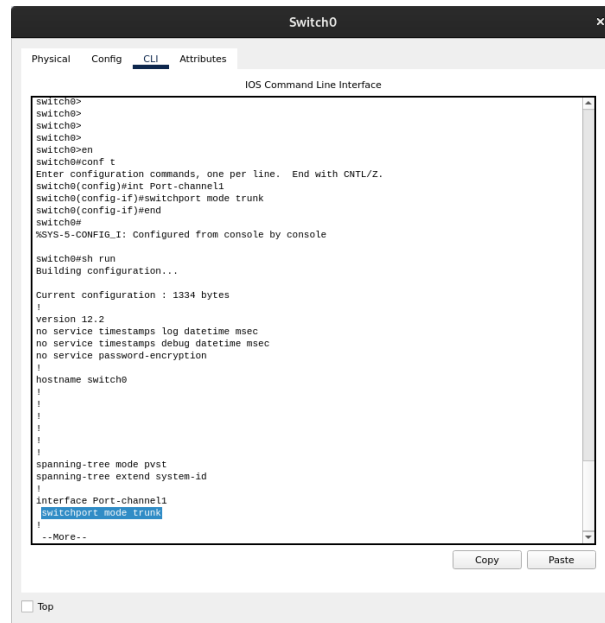


### État du réseau après configuration de l'agrégat

## 2.3 | Configuration du *trunk*

Pour que les informations relatives aux VLAN puissent être transmises, nous devons installer un *trunk* sur l'agrégat défini précédemment.

La photo suivante montre cette opération effectuée via la ligne de commande sur le *switch0*.



```
Switch0
Switch0#
Switch0#
Switch0#
Switch0#en
Switch0#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Switch0(config)#int Port-channel1
Switch0(config-if)#switchport mode trunk
Switch0(config-if)#end
Switch0#
NSYS-5-CONFIG-I: Configured from console by console

Switch0#sh run
Building configuration...

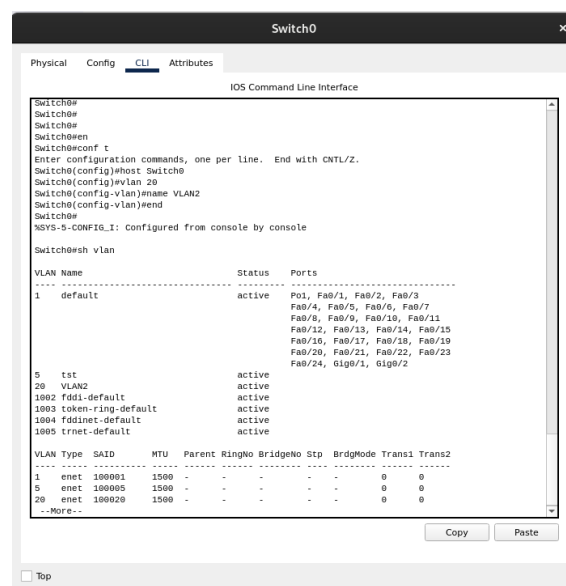
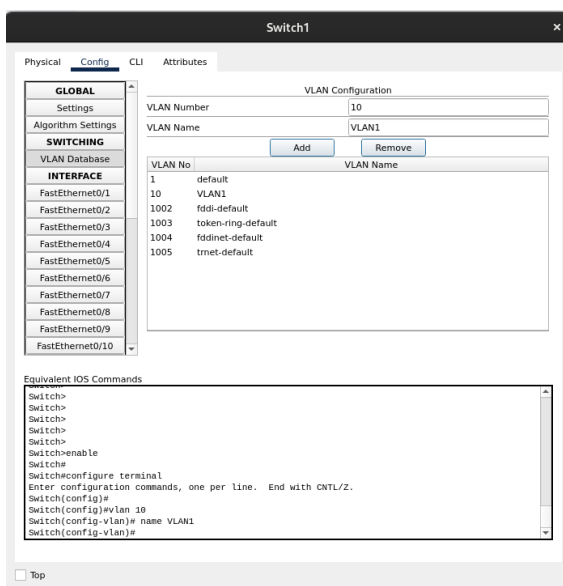
Current configuration : 1334 bytes
!
version 12.2
no service timestamps log datetime msec
no service timestamps debug datetime msec
no service password-encryption
!
hostname Switch0
!
!
!
!
!
spanning-tree mode pvst
spanning-tree extend system-id
!
interface Port-channel1
switchport mode trunk
!
--More--
```

## 2.4 | Creation des *VLAN*

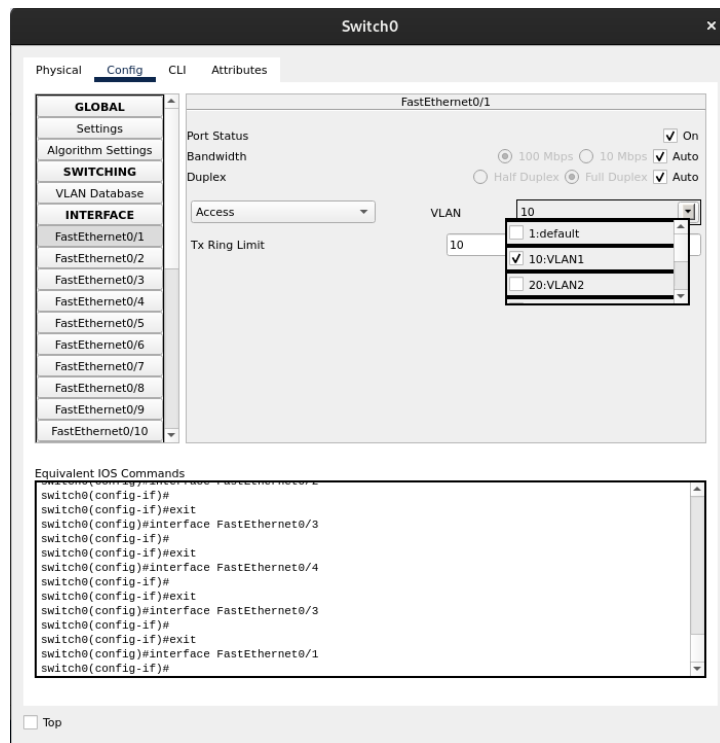
La création et configuration des *VLAN* se fera également au niveau du switch.

Les illustrations ci dessous montrent 2 méthodes pour créer un *VLAN*.

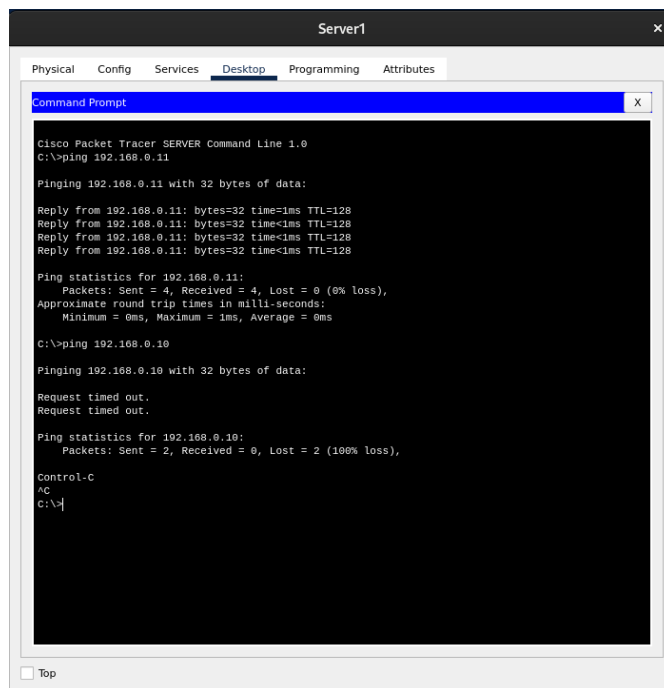
Via l'interface graphique (a gauche) ou le terminal (a droite).



Pour finir nous devons assigner les ports des switches connectant nos différentes machines aux VLAN créés précédemment.  
La manipulation ci dessous est a effectuer sur chaque ports.



Il est maintenant possible de tester la communication entre les différentes machines.  
Dans la photo suivante on peut voir le *Server1* envoyer des pings sur les IP des différentes machines.



### 3 | Conclusion

Après différents tests on peut constater que les machines de ce réseau ne peuvent communiquer que dans leurs VLAN respectifs.