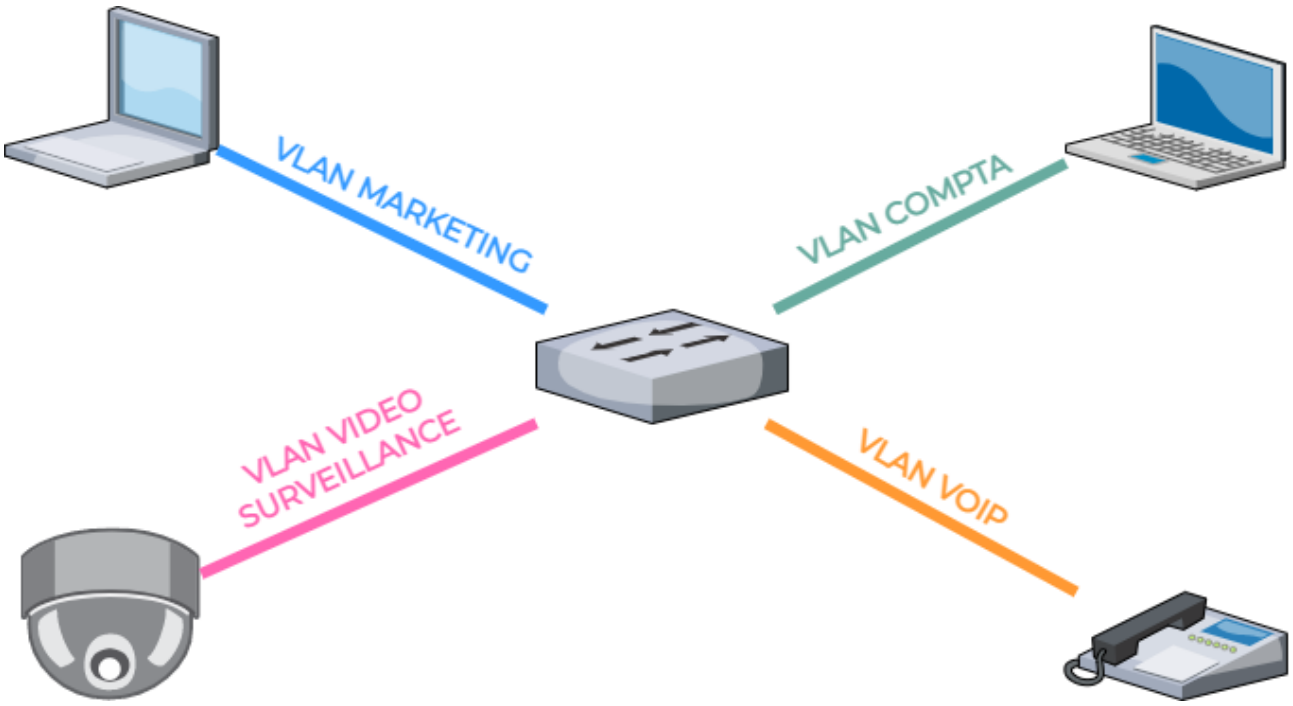
Configurez des VLAN sur un switch CISCO

Dans ce premier chapitre, je vous propose d'apprendre à segmenter votre réseau à l'aide de VLAN. Cela vous permettra de séparer différentes utilisations d'un même réseau, pour configurer de la voix sur IP (VoIP) ou créer une DMZ (zone démilitarisée) entre autres. Ceci vous permettra d'ajouter de la sécurité à votre réseau, mais aussi de l'optimiser.

Allez, on commence tout de suite.

Segmentez votre réseau

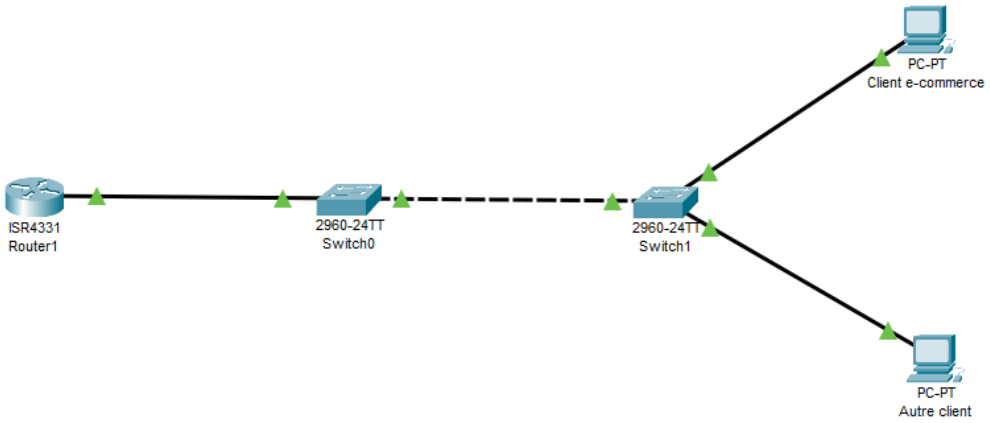
Le VLAN pour Virtual Local Area Network (réseau local virtuel en français), est comme dit dans l'introduction, une façon de segmenter le réseau non pas physiquement mais logiquement.



Cette segmentation est la solution à plusieurs problèmes :

- Problème de sécurité : elle permet d'isoler certaines parties du réseau, comme les serveurs, sans recours au routeur ;
- Problème d'optimisation : la segmentation étant logique, on peut créer plusieurs réseaux avec le même nombre de switch et de câble ;
- Problème de qualité de service : il est possible de réserver de la bande passante pour la VoIP par exemple (la téléphonie par voie IP).

Prenons par exemple le cas d'un datacenter regroupant un grand nombre de serveurs de différents clients. Les clients se connectant au datacenter doivent avoir accès uniquement à leurs serveurs et non pas ceux des autres clients. Un des moyens de faire serait de séparer physiquement les réseaux à partir du routeur donc d'avoir deux interfaces réseau du côté du LAN et de continuer ainsi avec, pour chaque réseau, ses câbles et ses switchs.



Cette solution est en fait viable et fonctionne très bien, mais il est peu probable qu'un datacenter ait pour objectif de n'avoir que deux clients, mais plutôt plusieurs centaines ou milliers. Il faudrait alors que le datacenter prévoit des milliers d'interfaces réseaux, suivi de centaines de milliers de câbles et de switch, car croyez-moi, il est inconcevable d'ajouter tous ces composants à chaque nouveau client, étant donné le coût que cela représente.

Une des façons de faire est donc de n'avoir qu'un seul réseau physique où tous les switchs et câbles seraient déjà installés est de les segmenter à l'aide des VLAN. De cette façon, lorsqu'un nouveau client demande un serveur, par exemple, au datacenter, il vous suffira de lui ajouter un VLAN et de brancher son serveur à ce VLAN. De cette façon, il aura accès uniquement à son serveur et à rien d'autre, bien qu'étant sur le même réseau physique que d'autres personnes (peut-être sur le même switch qu'un autre serveur), de la même manière personne d'autre n'aura accès à son serveur. C'est une façon simple et fiable d'ajouter de la sécurité dans un réseau LAN.

Parcourez la trame d'un VLAN

Pour ceux qui ne se rappellent pas, c'est l'occasion de relire comment se décompose une trame Ethernet car une trame de VLAN est sensiblement la même avec un petit ajout.

Cet ajout est normalisé par le protocole 802.1q (vous entendrez souvent « dot one Q » en anglais). Voilà comment elle se décompose :

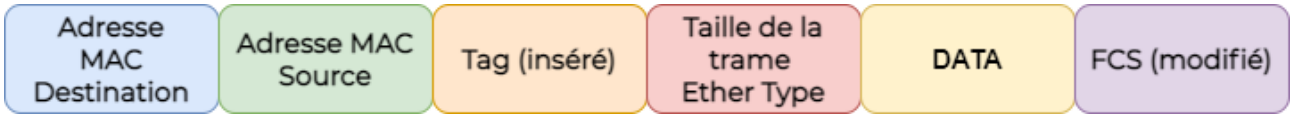


tableau 802.1Q
Elle commence de la même façon avec :

- Le préambule ;
- L'adresse de destination ;
- L'adresse source ;

Puis, le protocole dot1q ajoute deux fois 4 octets (deux pour le VLAN et deux pour l'Ether Type) à cet endroit de la trame :

- Tout d'abord, le champ Ether Type où est renseigné le protocole utilisé ici 0X8100 (pour 802.1q). Dans wireshark, vous verrez directement 802.1Q car il vous facilite la lecture ;
- Puis, les deux octets 802.1q décomposés comme ceci :
    - 3 bits pour la priorité (donc 8 priorités), 111 étant la priorité la plus haute ;
    - 1 bit appelé CFI qui doit être à 0 (pour des problèmes de compatibilité) ;
    - puis 12 bits pour le VLAN ID ce qui fait 4096 VLAN possibles dans un réseau ;
    - suit un autre Ether Type ;
    - puis les données.

Maintenant que nous avons vu ce qu'était un VLAN, voyons comment le créer sur votre switch CISCO, afin de placer le nouveau client d'e-commerce sur un réseau sécurisé, auquel personne d'autre n'aura accès.

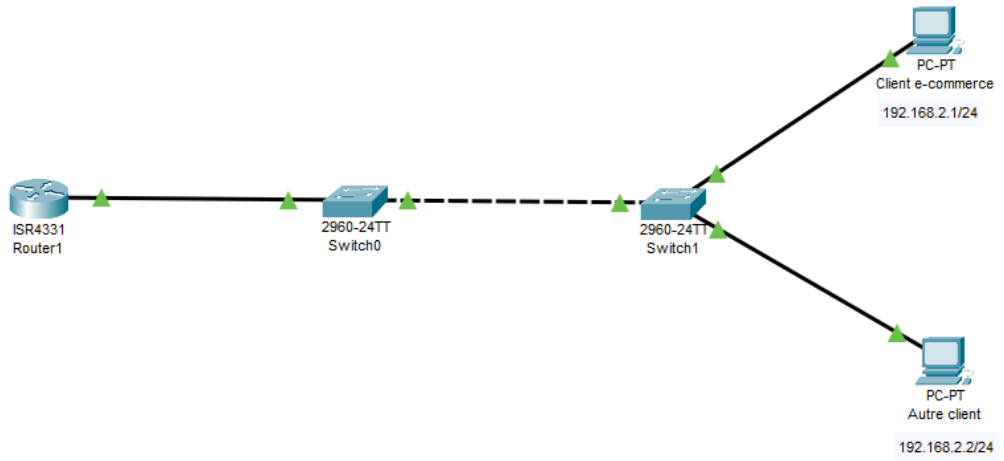
Création de votre architecture:

Mettez en place cette petite architecture en configurant routeur, switch et terminaux pour qu'ils soient en mesure de communiquer et tout en ligne de commande!

Inscrire vos lignes de commandes à la suite:

Créez un VLAN sur votre switch CISCO

Ajoutez ces adresses aux PC avant de commencer la configuration des VLAN.



Maquette du data center
En tant qu'administrateur·rice du datacenter, il vous faut ajouter un premier client au réseau et donc un premier VLAN.

Pour créer un VLAN sur votre switch, il vous faut tout d'abord entrer en mode configuration globale : switch1# configure terminal

Ajoutez un VLAN

Puis entrez cette commande :

```
switch1(config)# vlan { vlan-id | vlan-range }
```

- Le pipe `|` entre “vlan-id” et “vlan-range” et un "ou", ce qui signifie que vous pouvez utiliser la commande vlan soit en ajoutant un seul vlan-id ou en ajoutant un vlan-range.
- Les crochets `{}` signifient qu’un de ces arguments est obligatoire après la commande vlan.

Pour notre datacenter qui prévoit d’avoir 1000 clients cette année, la commande serait :

```
switch1(config)# vlan 1-1001
```

Le vlan 0 représentant ce que l’on appelle l’access c’est-à-dire l’absence de VLAN.

Le vlan 1 est le vlan natif, c’est-à-dire le vlan par défaut, si vous ne configurez par vos interfaces, elles seront toutes sur ce VLAN. Nous reviendrons sur ce concept juste après.

La commande `no vlan` permet de supprimer un VLAN dans le cas où vous vous seriez trompés. Elle fonctionne de la même manière que `vlan` .

```
switch1(config)# no vlan 1
```

Configurez votre VLAN

Votre VLAN étant créé, il ne vous manque plus qu’à le configurer. La commande pour entrer en mode configuration vlan est… exactement la même. En fait, cette commande créée un VLAN s’il n'existe pas déjà. Dans le cas contraire, il passe en mode configuration sur celle-ci.

Si vous voulez configurer le VLAN 2 :

```
switch1(config)# vlan 2 (pour entrer en mode configuration vlan)

switch1(config-vlan)# name e-commerce (pour lui donner un nom)

switch1(config-vlan)# state active

switch1(config-vlan)# no shutdown (pour l’activer)
```

Ces commandes fonctionnent aussi pour des ranges comme à la création d’ID.

Attribuez un port au VLAN

C’est la dernière étape avant que votre VLAN ne soit configuré. Cette étape consiste à attribuer votre VLAN à un port de votre switch. De cette façon, vous allez pouvoir construire son réseau tout en l’optimisant.

En effet, votre VLAN n’est actuellement disponible sur aucun port de vos switchs et donc de votre réseau. Vous pouvez voir l’association d’un port à votre VLAN comme le fait de brancher un câble sur votre réseau physique. Pour brancher le serveur du nouveau client, il va falloir associer tous les ports qui partent du routeur et qui mènent à lui. Dans notre exemple c’est le port Gi0/0/0 qui est relié au serveur d’e-commerce. Nous allons donc associer ce port au VLAN 2 que nous venons de créer.

Voici les commandes :

```
switch1(config)# interface gigabitethernet 0/0/0 (on passe
sur l’interface voulue)

switch1(config-if)# switchport access vlan 2 (on ajoute le
VLAN)
```

Vérifiez votre configuration

Ne jamais oublier de vérifier votre configuration et pour cela il existe des commandes très simples :

```
switch1# show vlan id 2 (pour voir un vlan en particulier)

switch1# show vlan summary (pour avoir un résumé des VLANs
existants)

switch1# show running-config vlan 2 (on spécifie le VLAN ou
le range que l’on veut vérifier)

switch1# show vlan (pour voir tous les VLAN existants)

VLAN Name                              Status    Ports

---- -------------------------------- ---------
------------------------------

1    default    active
Gi0/0, Gi0/2, Gi0/3, Gi1/0

Gi1/1, Gi1/2, Gi1/3, Gi2/0

Gi2/1, Gi2/2, Gi2/3, Gi3/0

Gi3/1, Gi3/2, Gi3/3

2   e-commerce  active
```

```
Gi0/1
(On voit ici que le VLAN 2 est créé et qu'il est associé au

port Gi0/1)

1002 fddi-default                       act/unsup

1003 token-ring-default                 act/unsup

1004 fddinet-default                    act/unsup

1005 trnet-default                      act/unsup
```

On s'aperçoit de plus qu'il existe un VLAN 1.

Ça me rappelle quelque chose ?

Changez le VLAN ID du VLAN natif

Ce VLAN 1 s'appelle le VLAN natif. En fait, sur un switch CISCO (même chez d'autres constructeurs) par défaut, tous les ports sont configurés sur le VLAN natif c'est-à-dire le VLAN 1. Du coup, lorsque l'on branche un ordinateur sur un port dont le VLAN n'est pas configuré, cet ordinateur est en fait sur le VLAN 1.

Il est possible et même recommandé de changer l'ID du VLAN natif. Pas d'inquiétude, chaque chose en son temps on en reparle juste après un nouveau concept.

Vérifiez que vous n'avez pas accès au serveur d'e-commerce

Pour vérifier votre configuration, je vous propose de tester la connexion vers le serveur du client d'e-commerce depuis un autre poste ne se trouvant pas sur le même VLAN.

Pour cela, configurez le deuxième PC (appelé autre_client sur la maquette) avec l'adresse 192.168.2.2/24. Sans VLAN, ces deux serveurs devraient pouvoir communiquer, mais ils ne se trouvent plus sur le même VLAN. C'est comme s'ils n'étaient plus sur le même réseau...

Tentez de faire un PING entre les deux… ça ne fonctionne pas et c'est tant mieux. Bien qu'étant sur le même réseau (192.168.2.X/24), ils ne peuvent pas communiquer, car ils sont sur un VLAN différents.

Configurez maintenant cette interface au VLAN 2 et refaites le test.

Magie, ça fonctionne !!!

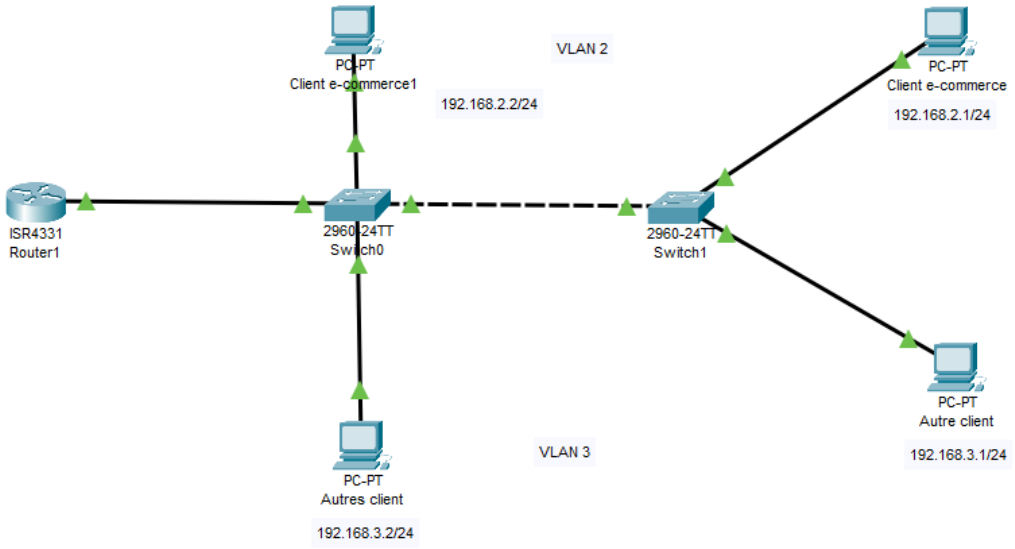
Vous avez noté que le réseau est en 192.168.2 ?

C'est une bonne pratique pour s'y retrouver dans les VLAN que de faire correspondre le 3e octet (ici le 2) avec l'ID du VLAN (2 aussi). Bien évidemment, tous les VLAN pourraient être configurés avec la même plage réseau, cela ne changerait rien.

Faites passer plusieurs VLAN sur le même câble

Le datacenter grandit et il se retrouve maintenant avec plusieurs clients, plusieurs serveurs et donc plusieurs switchs. Et à ce moment, le client de la société d'e-commerce pour laquelle vous avez attribué le VLAN 2 vous demande un accès à un second serveur, cependant sur le switch auquel son premier serveur est branché, il n'y a plus de place.

Vous le branchez donc sur un autre switch comme ceci :



Evolution du data center
Changez l'adresse de l'autre client en 192.168.3.1/24 et placez-le sur le VLAN 3 et ajoutez un deuxième identique sur le switch 0.

Et n'oubliez pas d'ajouter le nouveau serveur sur le VLAN2.

Faites passer plusieurs VLAN dans le trunk

Une fois la nouvelle interface du nouveau switch configurée, tentez de faire un ping entre les deux serveurs… ça ne fonctionne pas. En effet, pour que cela fonctionne, il faudrait que le câble qui relie les deux switchs soit aussi sur le VLAN 2 mais dans ce cas, les switchs du VLAN 3 ne pourraient pas communiquer entre eux !

Et pour que deux VLAN ou plus partagent le même câble, on configure les interfaces de ce câble en "**trunk**".

C'est quoi un trunk ?

Nous l'avons déjà vu, mais le trunk est un moyen du protocole dot1q. Il permet de faire communiquer deux appareils (des switchs en général) et leurs VLAN respectifs. De cette façon, le VLAN 2 du switch 1 et du switch 2 peuvent communiquer ensemble.

Attention, cela ne veut pas dire que des VLAN de différents ID peuvent communiquer ! Le VLAN 2 ne peut toujours pas communiquer avec le VLAN 3 !

La configuration se passe sur l'interface qui relie les deux switchs, comme ceci :

```
switch1# configure terminal
switch1(config)# interface GigabitEthernet 0/1
switch1(config-if)#switchport trunk encapsulation dot1q
switch1(config-if)# switchport mode trunk
switch1(config-if)# switchport trunk allow vlan 2-3
```

- La commande `switchport trunk encapsulation dot1q` force l'interface à passer en dot1q (sinon elle est en auto).
- La commande `switchport mode trunk` passe le lien en trunk.
- La commande `switchport trunk allow vlan 2-3` ajoute les VLAN 2 et 3 au trunk (en plus du VLAN natif)

On fait la même opération sur l'interface du switch2. Et on retente le ping entre les deux serveurs du VLAN2… heureusement cela fonctionne comme on le veut.

Vous avez maintenant un moyen très utile pour segmenter et sécuriser vos réseaux privés. Cela vous sera très utile pour l'administration de votre LAN. Étrangement, vous découvrirez dans le prochain chapitre comment relier ces VLAN entre eux, et saurez bien évidemment pourquoi vous devez le faire.

En résumé

- Vous pouvez segmenter un réseau physique grâce au VLAN sans modifier le câblage.
- Les VLANs sont standardisés par le protocole **802.1q** appelé aussi **Dot1q**.
- Les commandes à retenir sont :
  - **`vlan { vlan-id | vlan-range }`**, pour la création de VLAN ;
  - **`name vlan-name`**, pour lui donner un nom ;
  - **`switchport access vlan vlan-id`**, pour assigner un VLAN à une interface (une fois que vous êtes sur la configuration de l'interface).
- Il est possible de vérifier vos VLAN avec les commandes :
  - **`show running-config vlan [ vlan_id | vlan_range ]`**;
  - **`show vlan`**[brief | id [vlan_id | vlan_range] | name name | summary].
- Pour faire passer plusieurs VLANs sur une interface il faut la passer en mode **trunk** (une fois que vous êtes sur la configuration de l'interface) :
  - **`switchport mode trunk`** ;

- **switchport trunk allowed vlan**{ vlan-list all | none [add |except | none | remove { vlan-list }]}

Créez des liens entre vos VLAN pour l'administration

Dans ce chapitre, vous allez continuer l'administration de votre réseau local en connectant les VLANs à Internet et en créant des routes entre ces VLANs. Ces routes vous permettront, en tant qu'administrateur, d'avoir accès aux autres VLANs, mais aussi de créer un lien entre deux groupes qui en auraient le besoin.

Managez vos switchs en vous attribuant un VLAN

Le réseau du datacenter est maintenant configuré pour les clients, mais pas pour vous ! En effet, les VLANs permettent de segmenter le réseau et donc de le sécuriser.

La conséquence ?

Vous vous retrouvez dans l'incapacité de vous connecter à vos propres switchs depuis le réseau !

Lorsque l'on se connecte sur un appareil dans Cisco Packet Tracer, il simule une connexion physique directe, ce qui ne sera pas le cas dans la réalité. Vous n'irez pas vous connecter à chaque appareil en vous baladant dans l'entreprise, mais prendrez contrôle à distance par des moyens que nous verrons lors du chapitre sur la sécurité.

Ajoutez un VLAN pour l'administration

Ajoutez donc le VLAN 99 et donnez-lui pour nom : « administration ».

C'est depuis ce VLAN que vous administrerez les switchs et le routeur du datacenter ainsi que les autres serveurs du datacenter.

Pour vous connecter à un switch, il vous faut :

- 1 PC avec le protocole telnet (terminal network), c'est un protocole vous permettant de vous connecter à un autre appareil (PC, switch, routeur) à distance.
- Une adresse sur laquelle vous connecter.

C'est ce deuxième point qu'il vous manque. Tous les PC ont telnet par défaut, par contre votre switch n'a pas d'adresse IP !

Ajoutez une interface virtuelle pour le management

Pour vous connecter au switch, vous allez lui ajouter une interface virtuelle (ou VLAN interface). Ceci vous permettra d'ajouter une adresse à votre switch et donc de vous y connecter.

```
switch2(config)#interface vlan 99
switch2(config-if)# ip address 192.168.99.2 255.255.255.0
switch2(config-if)#no shutdown
switch2(config-if)#end
```

Vérifiez avec la commande :**`show running-config`**. Vous devriez avoir une interface de plus avec ce nom et cette adresse. Maintenant, le switch est disponible depuis le VLAN 99. Tentez

un ping depuis le poste de l’administrateur que vous avez ajouté. Si c’est ok, c’est que vous pourrez vous y connecter.

Faites la même chose depuis le switch 1 :

- Ajoutez une interface VLAN 99 avec l’adresse 192.168.99.1/24 ;
- N’oubliez pas d’ajouter le VLAN 99 au trunk de chaque switch ;
- Ajoutez le VLAN 99 au switch 1 (sans lui attribuer de port) ;
- Faites un ping pour vérifier.

Vous pouvez ainsi configurer tous vos matériels depuis votre poste d’administrateur. En revanche, si un client vous demande de redémarrer son serveur, vous ne pouvez le faire que physiquement à ce stade. Cela risque de vous ralentir considérablement dans votre administration d’un datacenter. Pour éviter cela, il va falloir créer des routes entre le VLAN 99 et les autres VLANs.

Managez les serveurs depuis le poste de l’administrateur

Vous avez dû remarquer qu'aucun serveur n'avait accès au routeur, et donc à Internet ! C’est la première étape : nous allons d’abord les relier au serveur et depuis celui-ci créer la route entre le VLAN 99 et les autres.

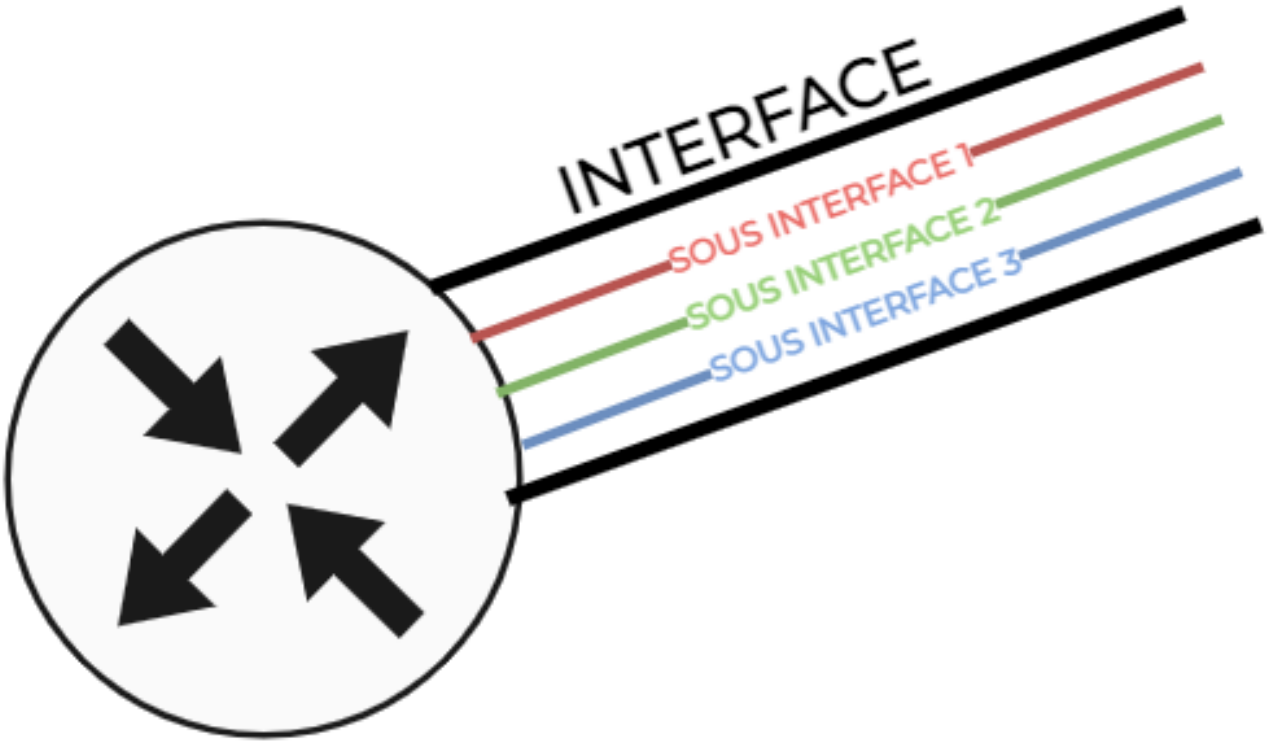
Reliez vos VLANs au routeur

Effectivement, ils ne le sont pas. Entre le routeur et le switch, il n'y a qu'un câble. Cela vous oblige à configurer un trunk. Cependant, vous n'avez qu'une interface du côté du routeur et plusieurs VLANs à configurer.

Aucun problème pour l'administrateur que vous êtes, car il existe une solution. Vous allez créer une sub-interface pour chaque VLAN sur le routeur. Ces sub-interfaces seront reliées à l'interface physique du routeur, celle où le câble est branché, et fonctionneront de la même manière que n'importe quelle interface physique.

Créez des sub-interfaces et reliez vos VLANs à Internet

La sub-interface que vous allez créer sera dépendante de l’interface reliant le routeur au switch 1, c'est-à-dire l’interface XXX. Elle en portera d’ailleurs le nom, suivi d’un point et d’un chiffre.



Les sous-interfaces
Par exemple si je veux créer des sub-interfaces dépendantes de l'interface GigabitEthernet 0/0/1, je vais créer les interfaces :

- `GigabitEthernet 0/0/1.1`
- `GigabitEthernet 0/0/1.2`
- `GigabitEthernet 0/0/1.3 ...`

Pour créer vos sub-interfaces, rien de plus simple, il vous suffit de taper ces quelques commandes :

```
Router1#configure terminal

Router1(config)#int gigabitEthernet 0/0/0
Router1(config-if)#no shut
Router1(config-if)#exit

!-- On créé la sub-interface et on la déclare comme étant le
VLAN 1 et le VLAN natif

Router1(config)#int gigabitEthernet 0/0/0.1

Router1(config-subif)#encapsulation dot1Q 1 native

Router1(config-subif)#ip address 192.168.1.254 255.255.255.0
Router1(config-subif)#exit
```

```
!-- On créé le sub-interface et on la déclare comme étant le
VLAN 2

Router1(config)#int fastEthernet 0/0/0.2

Router1(config-subif)#encapsulation dot1Q 2

Router1(config-subif)#ip address 192.168.2.254 255.255.255.0
Router1(config-subif)#exit

!-- On créé le sub-interface et on la déclare comme étant le
VLAN 99

Router1(config)#int fastEthernet 0/0/0.99

Router1(config-subif)#encapsulation dot1Q 99

Router1(config-subif)#ip address 192.168.99.254 255.255.255.0
Router1(config-subif)#exit
```

Une fois que vous avez configuré l'interface du switch 1 en trunk pour tous les VLANs, vous pouvez pinger un serveur du client d'e-commerce depuis le routeur.

Allez, on vérifie.

```
Routeur1#ping 192.168.2.1
```

Si votre routeur peut pinger votre serveur, c'est que ce dernier sera en mesure d'avoir Internet une fois que le WAN sera configuré !

Vos VLANs sont maintenant connectés au routeur, ce qu'il leur donne accès à Internet une fois que le routeur y est connecté.

Il ne vous manque plus qu'à créer la route entre le VLAN 99 et les autres.

Créez une route entre le VLAN 99 et les autres

Vous pouvez, pour commencer, faire un **`show ip interface brief`** . Cela vous permet de visualiser les interfaces et les adresses IP.

```
routeur1#sh ip int brief

Interface                  IP-Address      OK? Method Status
               Protocol

GigabitEthernet0/0         unassigned      YES NVRAM  up
          up

GigabitEthernet0/0.1       192.168.1.254   YES manual up
                up
```

```
GigabitEthernet0/0.2       192.168.2.254   YES manual up
                  up

GigabitEthernet0/0.3       192.168.3.254   YES manual up
                  up

GigabitEthernet0/0.99      192.168.99.254  YES manual up
                  up

GigabitEthernet0/1         unassigned      YES NVRAM
 administratively down down

GigabitEthernet0/2         unassigned      YES NVRAM
 administratively down down

GigabitEthernet0/3         unassigned      YES NVRAM
 administratively down down
```

Il vous faut ensuite ajouter le **`gateway`** aux PC comme ceci :

- Administrateur = IP 192.168.99.99 MASQUE 255.255.255.0 PASSERELLE 192.168.99.254
- Client d'e-commerce= IP 192.168.2.1 MASQUE 255.255.255.0 PASSERELLE 192.168.2.254
- Autres clients= IP 192.168.3.1 MASQUE 255.255.255.0 PASSERELLE 192.168.3.254

Vérifiez en faisant un ping du poste de l'administrateur vers un des serveurs.

Ça fonctionne et c'est tant mieux. Vérifiez si les serveurs ont accès au poste de l'administrateur… Ça fonctionne aussi, et ça, ce n'est pas bon mais chaque chose en son temps. Nous verrons cela dans la partie 4 de ce cours, consacrée à la sécurité justement.

Maintenant que votre LAN est configuré et que vos VLANs ont accès aux routeurs, il ne vous manque plus qu'à ajouter certains services vous permettant de vous faciliter l'administration de votre datacenter. Il vous faut aussi rendre les serveurs de vos clients accessibles depuis le NET. Vous découvrirez tout cela dans le prochain chapitre.

En résumé

- En ajoutant des VLANs interfaces au switch et aux routeurs, vous pouvez les administrer depuis n'importe quel poste :
    - switch(config)#interface vlan 99
- switch(config-if)# ip address adresse-ip masque
  switch(config-if)#no shutdown
  switch(config-if)#end

- Pour relier vos VLANs aux routeurs, vous devez créer des **sub-interfaces** et leur attribuer un VLAN et une adresse IP. Cette interface deviendra le **gateway** du VLAN :
    - Router(config)#int fastEthernet 0/0.X
    - Router(config-subif)#encapsulation dot1Q X
    - Router(config-subif)#ip address adresse-ip masque
    - Router(config-subif)#exit
- Une fois vos VLANs reliés au routeur, le routeur crée les routes permettant la communication entre eux.

Configurez les protocoles NTP, NAT et DHCP

Maintenant que vous savez connecter un réseau privé, il vous faut lui ajouter des services comme la gestion du temps (pour que les serveurs soient tous à la même heure) ou l'octroi automatique d'adresse IP (plus simple que de le faire manuellement). Il vous faut également rendre les serveurs de vos clients accessibles depuis le net. C'est ce que je vous propose dans ce chapitre. Allez, on commence !

Allouez des adresses IP automatiquement

Si vous avez déjà loué un serveur chez un hébergeur (donc un datacenter), vous savez que c'est assez rapide. Cette rapidité n'est pas le fait d'une personne réalisant toutes les tâches à effectuer pour rendre le serveur disponible. Toutes ses tâches sont bien évidemment automatisées. Et l'une d'elles n'est autre que le service DHCP.

Lancez le service DHCP

Sur les routeurs CISCO, le protocole DHCP est un service, mais il n'est pas lancé automatiquement. Il vous faut le lancer. Alors connectez-vous et tapez la commande : **`service dhcp`**

```
Router1(config)# service dhcp
```

Votre service DHCP est maintenant lancé, il vous reste maintenant à le configurer. Vous devrez le faire pour chaque VLAN.

Configurez le service DHCP pour chaque VLAN

```
routeur1(config)#ip dhcp pool e-commerce
routeur1(dhcp-config)#network 192.168.2.0 255.255.255.0
routeur1(dhcp-config)#default-router 192.168.2.254
```

Expliquons cela :

- `routeur1(config)#ip dhcp pool e-commerce` : vous créez ce que l'on appelle un pool DHCP, c'est-à-dire un serveur DHCP, il faut en créer un par VLAN. Là, vous lui donnez le nom d'e-commerce.
- `routeur1(dhcp-config)#network 192.168.2.0 255.255.255.0` : cela vous permet de spécifier le réseau du service DHCP, ici 192.168.2.0/24. Il s'agit du réseau du VLAN 2 (celui du client d'e-commerce).
- `routeur1(dhcp-config)#default-router 192.168.2.254` : vous indiquez à qui les clients DHCP (les serveurs de votre client d'e-commerce) doivent s'adresser. Ici il s'agit de l'interface du VLAN 2 créée sur le routeur.

Vérifiez la configuration

Pour vérifier la configuration, il vous suffit de vous connecter au PC et de lancer la commande :**`ipconfig /renew`**.

Si vous obtenez l'adresse 192.168.2.1/24 c'est que vous avez réussi. Faites de même avec le second PC et vérifiez que vous obtenez l'adresse 192.168.2.2/24.

Maintenant que vos machines sont sur le même réseau (grâce au VLAN et aux adresses IP), il serait bon qu'elles soient à la même heure.

Synchronisez les horloges de vos machines

Votre client d'e-commerce a maintenant deux serveurs, car son activité ne peut être supportée par un seul. Il serait judicieux de synchroniser l'heure de ces deux serveurs.

Vous vous demandez pourquoi ? Imaginez que le data center mette en place une sauvegarde de ces serveurs planifiée à des heures bien précises pour chacun et que l'un des serveurs soit en retard d'une heure. La sauvegarde du serveur en retard ne se lancera pas à la bonne heure et démarrera peut-être en même temps qu'un autre serveur ce qui risque de poser problème...

C'est pour cela qu'on utilise un serveur de temps, appelé NTP (pour Network Time Protocol).

Vous allez configurer ce protocole sur votre routeur Cisco (qui fera office de serveur). Les clients (les serveurs d'e-commerce) recevront l'heure du routeur et seront donc tous à la même heure.

Configurez votre routeur

C'est lui qui donnera l'heure.

Nous procédons de cette façon pour le bien du cours, dans le monde du travail le server NTP sera une source d'Internet (http://www.pool.ntp.org/zone/fr).

Tapez la commande suivante :

```
routeur1(config)#ntp master 1

routeur1(config)#end
```

Il ne vous manque plus qu'à configurer les clients (les switchs ici).

Configurez vos switchs

Connectez-vous à vos switchs et tapez les commandes suivantes :

```
switch1#conf t
switch1(config)#ntp server 192.168.99.254
switch1(config)#exit
```

La commande `ntp server 192.168.99.254` vous permet d'ajouter un serveur NTP, ici on a choisi l'interface de management du routeur (le VLAN 99).

```
switch1#sh ntp associations
address         ref clock       st   when  poll reach  delay
offset   disp
*~192.168.99.254  .LOCL.     1     41     64     7 14.583
36391.0  0.966
* sys.peer, # selected, + candidate, - outlyer, x
falseticker, ~ configure
```

La commande `show ntp associations` vous permet de voir avec qui votre matériel est associé. Les signes * et ~ signifient que le serveur est configuré et connecté.

```
switch1#sh ntp associations detail

192.168.99.254 configured, ipv4, our_master, sane, valid,
stratum 1

ref ID .LOCL., time DE5EFFF2.CFB68DE6 (04:24:18.811 UTC Fri
Mar 23 2018)

our mode client, peer mode server, our poll intvl 64, peer
poll intvl 64

root delay 0.00 msec, root disp 0.64, reach 3, sync dist
3490.64

delay 17.57 msec, offset 32551.8028 msec, dispersion 63.19,
jitter 3417.38 msec

precision 2**13, version 4

assoc id 37545, assoc name 192.168.99.254

assoc in packets 7, assoc out packets 7, assoc error packets
0

org time 00000000.00000000 (00:00:00.000 UTC Mon Jan 1 1900)

rec time DE5EFFFF.AC369ACB (04:24:31.672 UTC Fri Mar 23 2018)

xmt time DE5EFFFF.AC369ACB (04:24:31.672 UTC Fri Mar 23 2018)

filtdelay =    17.57   11.72 11.32   17.02    6.97    8.30
  20.01    0.00

filtoffset = 32551.8 29492.0 29384.8 29234.2 29122.5 28884.5
28748.6    0.00

filterror =     0.24    1.12    1.15    1.18    1.21
1.24    1.27 16000.0

minpoll = 6, maxpoll = 10
```

La commande **`show ntp associations detail`** vous permet d'en voir un peu plus sur le serveur NTP. Notamment si la connexion est *sane* (saine) et valide.

```
switch1#sh clock detail
*04:37:10.048 UTC Fri Mar 23 2018
Time source is NT
```
La commande **`show clock detail`** vous montre que la source de l'horloge est un serveur NTP.

Pour aller plus loin, vous pouvez si vous le souhaitez ajouter à votre maquette un serveur et configurer dessus le service NTP.

Sachez que sur Linux (il suffit d'installer le paquet ntp) et sur Windows, c'est en interface graphique dans les réglages de l'heure.

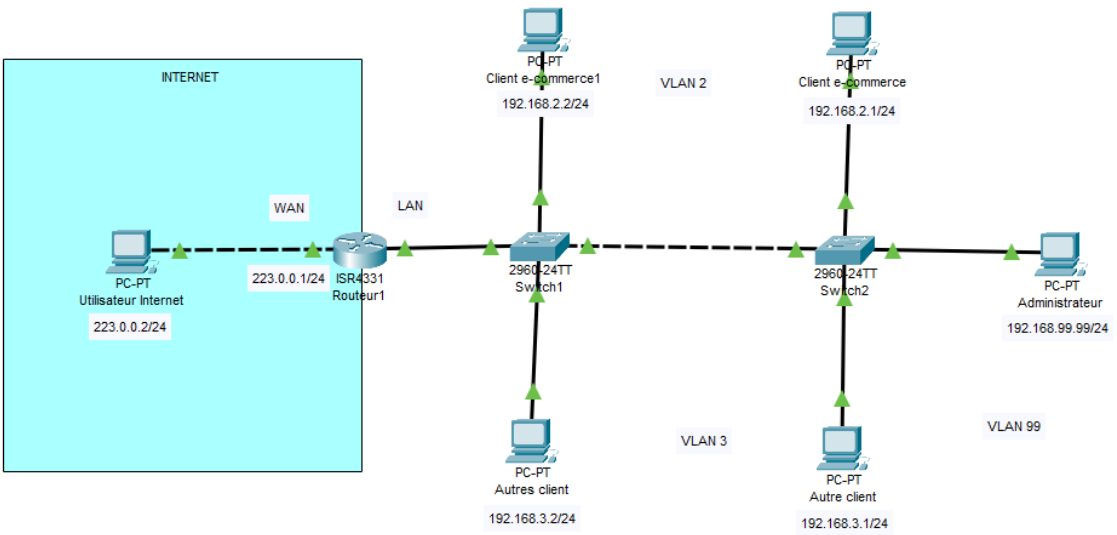
Vos serveurs sont à l'heure et ont une adresse privée, ce qui leur permet d'avoir accès à Internet une fois que le NAT sera configuré. Cependant, sans adresse publique, ils ne seront pas accessibles depuis l'extérieur, c'est-à-dire depuis Internet. Vous devez donc maintenant connecter votre data center à Internet.

Configurez l'accès à Internet à votre réseau

Il est important de bien comprendre cette notion car vous allez l'appliquer tout de suite.

Configurez votre WAN

Commencez par ajouter un PC au routeur comme sur le schéma. Pour notre exemple, nous le branchons directement sur le routeur, ce qui n'est pas le cas sur le terrain. En effet, pour nous approcher de la réalité, il aurait fallu brancher un autre routeur, puis le PC. C'est ce que nous ferons dans la prochaine partie, lorsque vous apprendrez à connecter des routeurs entre eux. Mais pour le moment, cette astuce vous suffira à comprendre le NAT, sur un routeur CISCO.



Le WAN dans le data center
Ensuite, ajouter une adresse à l'interface WAN de votre routeur. Pour cela connectez-vous sur le routeur et configurez l'interface WAN du routeur comme ceci :

```
routeur1(config)#interface gigabitEthernet 0/0/1
```

```
routeur1(config-if)#ip address 223.0.0.1 255.255.255.0
routeur1(config-if)#no shutdown
```
Puis configurez le PC "Utilisateur Internet" avec l'adresse IP 223.0.0.2/24.

Ceci mérite une explication. Grâce à cette configuration, l'Utilisateur_Internet et l'interface WAN du routeur sont sur le même réseau. Vous pouvez le vérifier en faisant un PING de l'un vers l'autre. Nous simulons en fait, pour notre exemple, que l'Utilisateur_Internet et le routeur sont bien connectés à Internet. Encore une fois, dans la prochaine partie nous verrons comment cela est réellement fait (la partie publique du réseau), mais pour le moment nous étudions la partie privée du réseau.

Maintenant, pour donner accès à Internet à votre LAN, vous allez devoir configurer sur votre routeur, ce que l'on appelle une route par défaut. Vous vous souvenez que, pour les VLAN, votre routeur a créé des routes entre les VLAN (des réseaux donc) ? Eh bien, une route par défaut, c'est une route vers tous les réseaux que votre routeur ne connaît pas.

Votre routeur connaît maintenant vos VLAN et l'adresse WAN. Il sait également comment acheminer des messages entre ces réseaux. Dans le cas où un message serait envoyé vers un réseau qu'il ne connaît pas (Internet), il l'enverra vers la route par défaut. Il s'agit en fait de la passerelle du routeur1. Cette passerelle dans notre cas, c'est le PC Utilisateur_Internet. Dans la vraie vie, ce sera un autre routeur.

Votre route par défaut va donc dire au routeur : « Dans le cas où tu ne connais pas le réseau du destinataire, dirige le paquet (le message) vers ce réseau ». Ce sera ensuite au prochain routeur de savoir où envoyer le paquet. Vous comprendrez mieux tout cela lors de la prochaine partie consacrée au WAN.

Connectez-vous donc à votre routeur et tapez ces commandes :

```
routeur1(config)#ip route 0.0.0.0 0.0.0.0 223.0.0.2
```

```
routeur1(config)#end
```
La route 0.0.0.0 avec le masque 0.0.0.0 est la route par défaut. Le dernier argument est la passerelle 223.0.0.2.

Voilà votre route par défaut est créé. Vérifions cela en faisant un PING depuis un poste du LAN vers le PC Utilisateur_Internet.

Ça ne fonctionne pas ? C'est normal, le NAT n'est pas encore configuré. C'est la prochaine étape de ce cours !

Configurez le NAT

Qu'est ce que le NAT ?

Le NAT, en anglais Network Address Translation, est une fonctionnalité qui permet de partager Internet dans un réseau local, en associant une adresse IP publique à un ou plusieurs postes ou serveurs qui possèdent une adresse IP privée.

Pour configurer le NAT, vous allez devoir créer un groupe, appelé NAT_INTERNET_VLAN2, composé du réseau 192.168.2.0/24 et que vous autoriserez grâce à la commande `permit` , à faire quelque chose. Ce quelque chose sera défini dans une autre commande. Vous l'appliquerez juste

après à l'interface WAN du routeur (il faut répéter cette opération pour chaque VLAN que vous voulez autoriser à accéder à Internet).

```
routeur1(config)#ip access-list standard NAT_INTERNET_VLAN2
routeur1(config-std-nacl)#permit 192.168.2.0 0.0.0.255
routeur1(config-std-nacl)#exit
```

- La première ligne crée la liste et lui donne un nom.
- La deuxième ligne autorise le réseau 192.168.2.0/24. Le dernier argument est le " wildcard mask " il s'agit d'un masque inversé. En fait 0.0.0.255 = 255.255.255.0.

```
routeur1(config)#int gi0/1
routeur1(config-if)#ip nat outside
routeur1(config-if)#exit
```

Vous indiquez ici, que l'interface WAN est l'interface de sortie du NAT.

```
routeur1(config)#int gigabitEthernet 0/0.2
routeur1(config-subif)#ip nat inside
routeur1(config-if)#exit
```

Ensuite, vous indiquez que l'interface de votre VLAN 2 est en entrée du NAT (à répéter pour chaque VLAN).

```
routeur1(config)#ip nat inside source list NAT_INTERNET_VLAN2
interface GigabitEthernet0/1 overload
```

Cette commande applique à l'interface WAN, le groupe que vous avez créé au début. Vous autorisez donc le VLAN 2 à utiliser le NAT en sortie sur l'interface WAN.

Retentez le ping depuis un serveur du VLAN 2 vers l'Utilisateur_Internet pour vérifier que cela fonctionne bien.

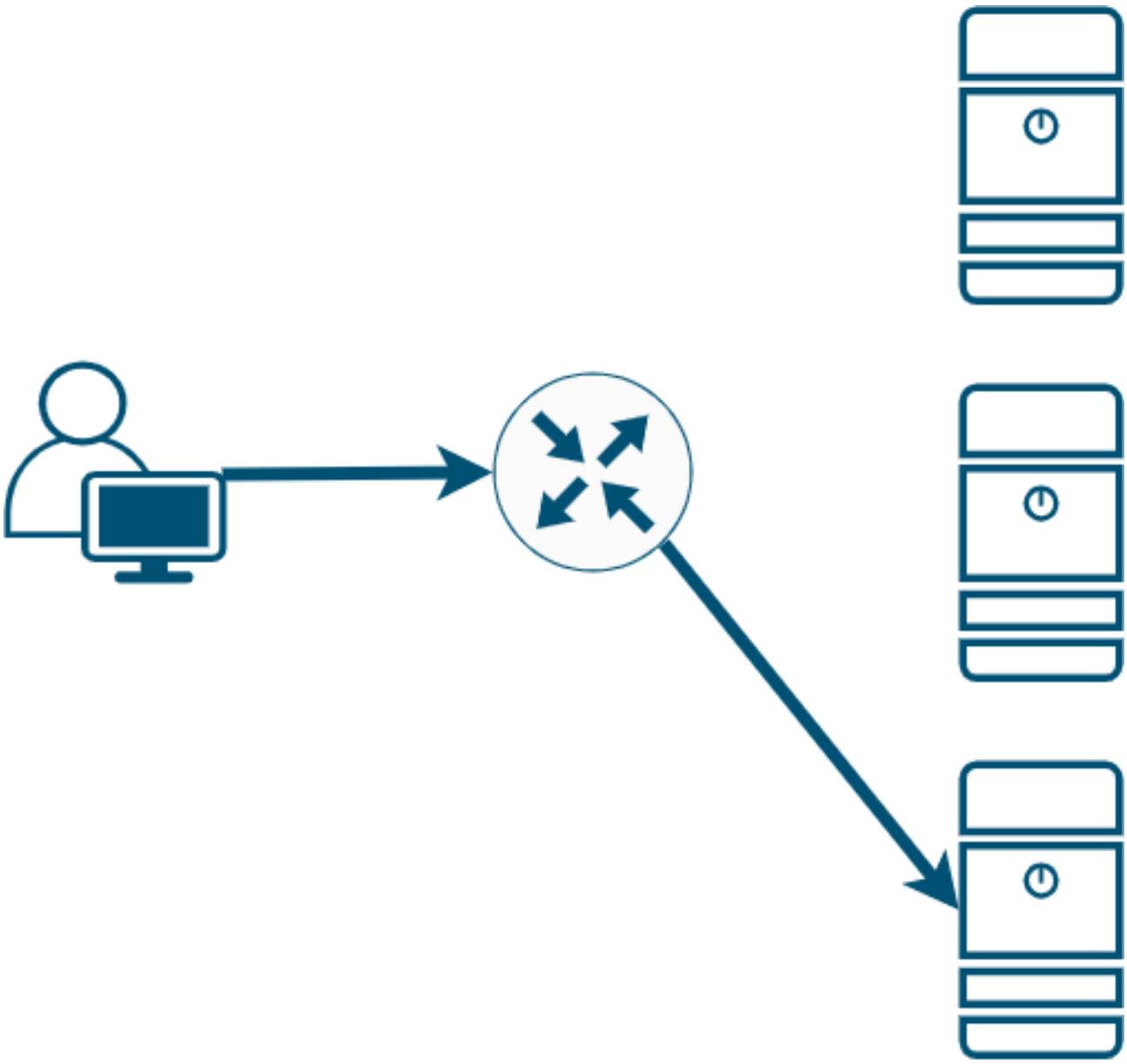
Voilà, vos serveurs ont accès à Internet. Vous allez maintenant faire l'opération inverse et rendre vos serveurs accessibles depuis Internet. Ainsi, les clients pourront faire leurs achats sur le site d'e-commerce.

Rendez vos serveurs accessibles depuis Internet

En effet, ce n'est pas parce que votre serveur a accès à Internet qu'un utilisateur d'Internet a accès, lui, à votre serveur. Pour cela, il faudrait que votre serveur ait une adresse IP publique et non privée comme c'est le cas actuellement.

Par contre, si vous mettez une adresse publique directement sur votre serveur, il ne sera plus sur le bon réseau et n'aura plus accès au routeur !

Ce type de NAT s'appelle du **port forwarding**. Cette technique vous permet de dire au routeur : « Si quelqu'un demande cette adresse publique, renvoie-le vers cette adresse privée ».



NAT : Port forwarding
Dans notre exemple, nous allons rediriger tous les messages (donc le PING) à destination de l'adresse WAN du routeur vers le serveur d'e-commerce (192.168.2.1). On appelle cela « NAT one-and-one », c'est-à-dire que tous les protocoles et ports seront redirigés vers l'adresse interne.

```
routeur1(config)#ip nat inside source static 192.168.2.1
223.0.0.1
routeur1(config)#end
```

Maintenant, pour vérifier, faites une requête internet depuis l'Utilisateur_Internet vers l'adresse 223.0.0.1. Cela fonctionne et si vous éteignez le serveur d'e-commerce...la requête est timeout elle ne fonctionne plus. C'est que la redirection fonctionne bien.

Dans le cas où vous auriez à configurer un serveur qui héberge un service WEB, vous auriez redirigé uniquement le port HTTP ou HTTPS comme ceci :

```
routeur1(config)#ip nat inside source static tcp 192.168.2.1
80 223.0.0.1 80
routeur1(config)#end
```

Dans cette configuration, on spécifie le protocole et le port ; ce qui permet d'utiliser une seule adresse IP publique pour de nombreux serveurs.

Votre LAN est enfin configuré (adresses IP, horloges, accès Internet et accès depuis Internet) et prêt à être exploité par vos clients.

Dans le prochain chapitre, vous verrez comment optimiser ce LAN et pallier les pannes de switchs et de câbles.

En résumé

- Vous pouvez gérer vos adresses automatiquement grâce au service DHCP.
- Pour le créer, il vous faut créer un pool et l'associer à un réseau du routeur :
    - routeur1(config)#**`ip dhcp pool nomDuPool`**
    - routeur1(dhcp-config)#**`network adresse masque`**
    - routeur1(dhcp-config)#**`default-router adresseDeLInterface`**
- Un serveur NTP est un serveur qui donne l'heure à ses clients. De cette façon, les clients ont tous la même heure.
- Pour le créer, vous devez entrer l'adresse du serveur NTP, il en existe sur le web :
    - switch1(config)# `ntp server adresseNTPServeur`
- Pour configurer l'accès à Internet sur votre routeur, il vous faut tout d'abord configurer une route par défaut, avec le prochain saut comme gateway.
    - routeur1(config)# `ip route 0.0.0.0 0.0.0.0 adresseGateway`
- Ensuite, vous devez configurer le NAT en commençant par un groupe :
    - routeur1(config)# `ip access-list standard nomDuGroupe`
    - routeur1(config-std-nacl)# `permit adresseRéseau masqueInversé`
- Puis indiquez les interfaces de sortie et d'entrée du NAT :
    - routeur1(config)# `int interfaceDeSortie`
    - routeur1(config-if)# `ip nat outside`
    - routeur1(config)# `int interfaceDEntrée`
    - routeur1(config-if)# `ip nat inside`
- Pour finir, vous devez appliquer la règle de groupe à l'interface de sortie du NAT :
  `ip nat inside source list nomDuGroupe interface GigabitEthernet0/1 overload`

- Pour configurer une redirection static depuis Internet, vers un serveur privé, il faut entrer la commande :
    - `ip nat inside source static tcp adressePrivée portPrivé adressePublique portPublique`