Cisco

C'est la décadence.

[Cisco] Installation Cisco Packet Tracer

Introduction

Cisco Packet Tracer est un outil de simulation réseau pour apprendre les réseaux et leurs différentes typologies.

Il est disponible sur **Windows**, **Ubuntu** et **MacOS**.

Prérequis

- Créer un compte Cisco

Installation

Une fois connecté sur votre compte, vous pourrez accéder au téléchargement :

https://www.netacad.com/portal/resources/packet-tracer

[Cisco] IOS

Introduction

IOS pour **Internetwork OS** est le système d'exploitation de Cisco présent sur ses équipements physique et sur Packet Tracer évidemment.

Lignes et interfaces

Sur les équipements, on retrouve deux types de ports :

- Les **interfaces** qui sont les ports réseaux où seront connectés les utilisateurs finaux ou les switch qui sont *in-band*.

- Les **lignes** qui sont soit physiques (port USB ou série) ou virtuelles (*VTY*) qui permettent de se connecter à la **console** (aussi appelé *rollover*).

Les contextes

Il existe dans IOS trois contextes permettant l'exécution de certaines commandes.

Mode racine

C'est le mode permettant l'exécution de toutes les commandes relatives à l'équipement lui-même.

Il est représenté par le prompt suivant :

```
Hostname#
```

Mode de configuration global

C'est le mode permettant la configuration relative à l'équipement lui-même.

Il est représenté par le prompt suivant :

```
Hostname(config)#
```

Pour l'activer il faut exécuter la commande suivante :

```
configure terminal
```

Mode de configuration spécifique

C'est le mode permettant la configuration d'une **ligne** ou d'une **interface** spécifique.

Il est représenté par le prompt suivant où le **X** est le composant spécifique sélectionné :

```
Hostname(config-X)#
```

Pour accéder à une configuration spécifique d'interface on utilise la commande suivante :

```
interface <INTERFACE_NAME> <INTERFACE_NUMBER>
```

Pour accéder à une configuration spécifique dune ligne on utilise la commande suivante :

```
line [console/vty] [NUMBER=0]
```

Niveaux de privilège d'exécution

Il existe dans IOS deux niveaux de privilèges d'exécution : le mode d'exécution **utilisateur** et **privilégié**.

Mode d'exécution utilisateur

Ce mode est celui sélectionné par défaut si un mot de passe a été définit pour le mode d'exécution privilégié.

On dit qu'il a un niveau de privilège à **1**.

Il permet d'accéder en **lecture-seule** à la configuration de l'appareil et permet aussi d'exécuter des commandes de **diagnostic**.

Il est représenté par le caractère **>** dans le prompt :

```
Hostname(config)>
```

Mode d'exécution privilégié

Ce mode permet l'**exécution de l'ensemble des commandes** (dans le contexte approprié).

On dit qu'il a un niveau de privilège à **15**.

Pour y accéder, il faut exécuter la commande suivante :

```
enable
```

Un **mot de passe** peut être définit pour accéder à ce mode grâce à la commande suivante :

```
secret <YOUR_PASSWD>
```

> La commande secret a succédée à la commande password qui n'était pas sécurisée et permettait de voir les mots de passe en clair dans la config.

Le mode d'exécution privilégié est représenté par le caractère **#** dans le prompt :

```
Hostname(config)#
```

VTY

Il existe une autre manière de se connecter à la console plutôt que de passer par le **rollover** qui est *out-of-band* (qui est un port d'administration).

Cette méthode c'est le **VTY**, qui permet de se connecter à la console en passant par une interface.

Le **VTY** permet entre autre, de pouvoir se connecter à distance à la console.

Cela est possible car l'équipement est capable de détecter qu'une trame lui est destinée (en comparant l'adresse MAC de destination à la sienne).

Par défaut il y a 5 **VTY** ouverts sur un équipement nommés **VTY0** puis **VTY1** jusqu'à 4.

Les configurations

Il existe deux types de configuration sur IOS : la **running-config** et la **startup-config**

La running-config

Il s'agit de la configuration actuelle du système qui est mis à jour à chaque exécution de commande sur le système.

Elle est stockée dans la **RAM** et peut, voire doit être sauvegardée car un redémarrage la supprime complètement.

La startup-config

Il s'agit de la configuration de la version du dernier redémarrage qui n'est mis à jour que lors d'un redémarrage contrôlé ou de l'import manuel.

Elle est stockée dans la **NVRAM** (Non-Volatile RAM) et est donc persistante après le redémarrage.

Sauvegarder la running-config dans la startup-config

```
copy running-config startup-config
```

Il existe une commande alternative bien plus courte :

```
wr
```

Aide à la saisie

En cas de difficulté sur les commandes disponibles dans un contexte ou sur un commande spécifique, le caractère **?** peut nous aider en nous donnant des informations.

Il peut être utilisé tout seul :

```
?
```

Ou dans une commande de cette façon :

```
show ip ad?
```

Commandes utiles

Afficher les interfaces

Permet d'afficher des informations sur les interfaces de l'équipement comme l'adresse IP ou son status :

```
show ip interface brief
```

Afficher la table MAC

Cette commande ne fonctionnera que sur un switch :

```
show mac address-table
```

Afficher les ports ouverts et services actifs

Vous pouvez afficher les ports actuellement utilisés par vos services actifs avec cette commande :

```
show ip ports all
```

Do

Permet d'exécuter une commande de contexte racine dans un autre contexte :

```
do <COMMAND>
```

Hostname

Permet de changer le **nom d'hôte** de l'équipement :

```
hostname <HOSTNAME>
```

IP Domain

Permet de changer le **nom d'hôte sur le réseau** de l'équipement :

```
ip domain name <FQDN>
```

> Sur un équipement local, vous pouvez utilisez le hostname.local par exemple.

Motd

Permet de changer la bannière lors de la connexion à l'équipement :

```
banner motd <END_CHARACTER>
```

Exit

Permet de sortir du contexte actuel pour revenir au contexte précédent :

```
exit
```

End

Permet de revenir au contexte racine :

```
end
```

Show version

Permet d'afficher la version de l'image (OS) :

```
show version
```

Le pipe

Il permet de filtrer la sortie d'une commande (comme sous Linux) et admet différentes options.

Il a pour caractère **|** et bénéficie de plusieurs fonctionnalités.

Voici son usage :

```
<COMMAND> | <OPTION> <ARGUMENT>
```

Include

Permet d'agir comme la commande grep sous Linux dans son usage initial, c'est à dire pour capturer uniquement les lignes où le motif spécifié est présent :

```
<COMMAND> | include <MOTIF>
```

Exclude

Permet de faire l'inverse d'include, c'est à dire d'afficher toutes les lignes dont le motif spécifié n'est pas présent :

```
<COMMAND> | exclude <MOTIF>
```

Begin

Agit comme l'include mais affiche toutes lignes à partir de la ligne où le motif a été trouvée :

```
<COMMAND> | begin <MOTIF>
```

Section

Agit comme le begin mais n'affiche que la section du motif, c'est à dire s'il détecte des espaces ou des tabulations :

```
<COMMAND> | section <MOTIF>
```

[Cisco] ICMP

Introduction

Cette page va décrire les différentes possibilités offertes par les équipements Cisco afin de pouvoir débugger avce l'ICMP.

icmp Ping

Voici la syntaxe basique de la commande **ping** par défaut :

```
ping <IP_DST>
```

Remarque : La commande ping peut être interrompue par la combinaison **CTRL + Shift + 9** .

La commande ping peut aussi s'utiliser en mode interactif, ce qui permet de définir les options unes à unes en répondant aux questions :

```
ping
```

On peut aussi spécifier une adresse IP source, pour tester si la machine de destination parvient à accéder à l'interface qu'on souhaite de notre machine :

```
ping <IP_DST> [source IP_SRC]
```

Il est possible de définir le nombre de paquet à envoyer grâce à l'option repeat :

```
ping <IP_DST> [repeat NUMBER]
```

On peut définir le **timeout** (en seconde) :

```
ping <IP_DST> [timeout NUMBER]
```

Ou encore la **taille** du paquet (en octet) :

```
ping <IP_DST> [size NUMBER]
```

On peut activer le mode **Record** (ping étendu) qui permet d'afficher chaque interface de sortie par lesquels le paquet est passé.

Pour cela, on entre en mode interactif, on met la valeur de "*Extended commands*" sur **y** et on met la valeur de "*Loose, Strict, Record, Timestamp, Verbose[none]*" sur **R** :

```
R1#ping
Protocol [ip]:
Target IP address: 192.168.0.1
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface:
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0x0000ABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]: R
Number of hops [ 9 ]:
Loose, Strict, Record, Timestamp, Verbose[RV]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.0.1, timeout is 2 seconds:
```

Activer le mode debug icmp

Permet d'afficher les logs dans la console à chaque fois qu'une requête icmp est envoyée ou reçue.

```
debug ip icmp
```

Traceroute

La commande **traceroute** permet d'afficher l'adresse IP des interfaces d'entrées des routeurs que le paquet à traverser avant d'arriver à destination :

```
traceroute <IP_DST>
```

En cas de réussite, la commande affichera **"Port unreachable"**. Dans le cas contraire, elle affichera **"Time exceeded"**.

On peut spécifier un nombre de **tentative** :

```
traceroute <IP_DST> [probe NUMBER]
```

[Cisco] Configuration des interfaces

Pour configurer une interface, il faut passer dans le contexte de configuration globale puis spécifique :

```
configure terminal
```

Puis :

```
interface <INTERFACE_NAME> [SLOT/]<PORT>
```

On peut désormais démarrer l'interface (par défaut éteinte sur les routeurs) :

```
no shutdown
```

On peut aussi définir une adresse IP :

```
ip address <IP> <MASK>
```

Remarque : La notation CIDR n'est pas supportée !

De la même manière, on peut supprimer l'adresse IP définie d'une interface grâce à la commande suivante :

```
no ip address
```

Il est aussi possible de saisir une passerelle par défaut grâce à ces deux commandes :

```
no ip routing
```

```
ip default-gateway <GATEWAY_IP>
```

[Cisco] Configuration des lignes

Pour configurer une ligne, il faut passer dans le contexte de configuration globale puis spécifique :

```
configure terminal
```

Puis :

```
line <LINE_NAME> <LINE_NUMBER>
```

La ligne peut être physique (**console**) ou virtuelle (**VTY**).

Pour configurer la ligne console :

```
line console 0
```

Pour configurer les VTY :

```
line vty 0 4
```

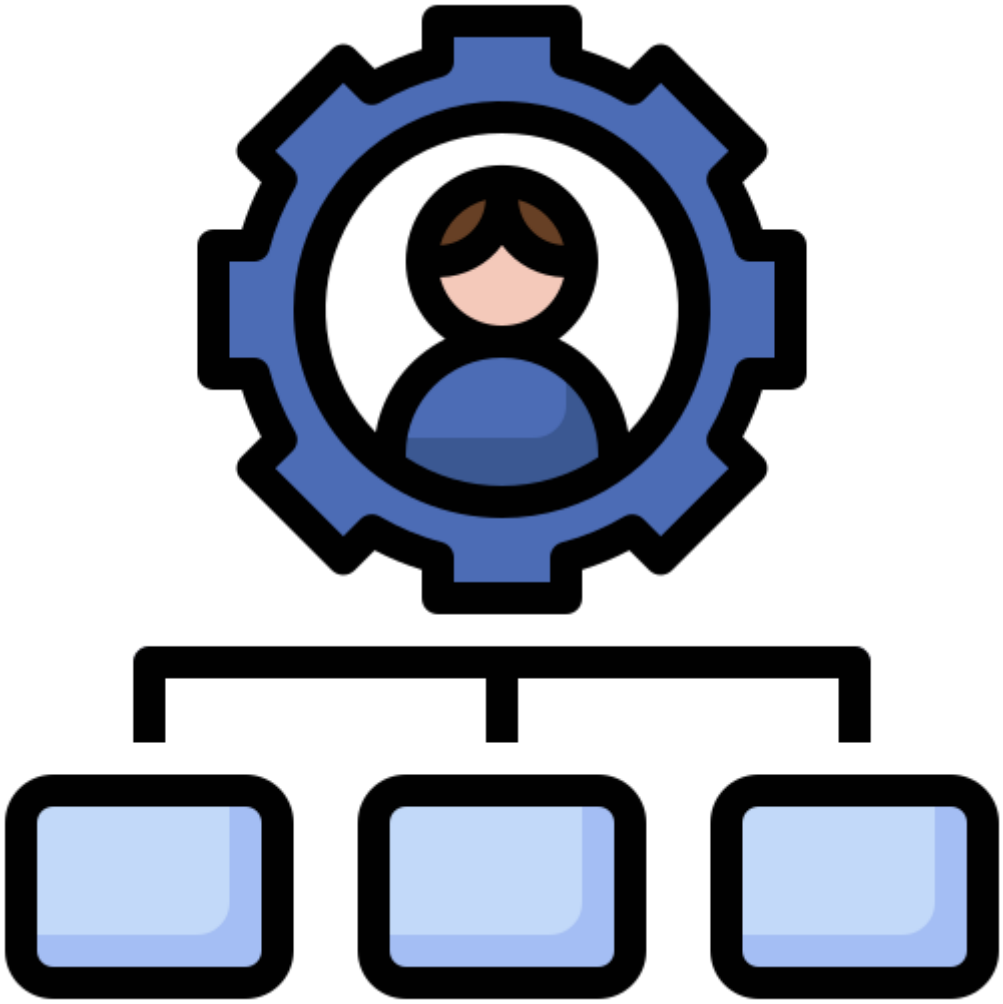
Ou un VTY spécifique :

```
line console 0
```

[Cisco] Gestion des droits

Introduction

Cisco permet la création d'utilisateurs sur ses équipements.



Les modes d'authentification

- Pas de mot de passe requis :

```
no login
```

- Mot de passe identique à tous les administrateurs :

```
login
```

- Ou propre à chacun :

```
login local
```

Créer un utilisateur

Passer en mode configuration globale :

```
conf t
```

Et créer votre utilisateur :

```
username <USER> secret <PASSWORD>
```

Ou en mode non sécurisé :

```
username <USER> password <PASSWORD>
```

Sélectionner la ou les lignes :

```
line vty 0 4
```

Et enfin, précisez le **mode d'authentification.**

Définir le mot de passe du mode privilégié

Tout d'abord, entrer dans le mode de configuration globale :

```
conf t
```

Et définissez votre mot de passe :

```
enable secret <PASSWORD>
```

Définir le mot de passe d'une ligne spécifique

```
line <LINE_TYPE> <LINE_NUMBER>
```

```
password <PASSWD>
```

Et enfin, précisez le **mode d'authentification.**

Chiffrer les mots de passe des utilisateurs

Cette commande va démarrer le service de chiffrement des mots de passe et va ainsi chiffrer tous les mots de passe présents sur le système :

```
service password-encryption
```

Afficher le niveau de privilège actuel

En mode de configuration racine :

```
show privilege
```

Changer le niveau de privilège

Passer de mode d'exécution d'utilisateur à privilégié (1 à 15) :

```
enable
```

Passer de mode d'exécution privilégié à utilisateur (15 à 1) :

```
disable
```

Afficher le mot de passe du mode privilégié

Si les mots de passe ne sont pas chiffrés :

```
show run | include enable
```

Afficher le mot de passe des utilisateurs

Si les mots de passe ne sont pas chiffrés :

```
show run | include username
```

Afficher les utilisateurs actifs

```
show users
```

[Cisco] SSH et Telnet

Introduction

Les services **SSH** et **Telnet** permettent de se connecter à distance sur les équipements et ainsi accéder aux **VTY**.

À noter que l'utilisation de SSH est recommandé car la connexion est chiffrée.

Prérequis

- S'assurer que vos utilisateurs soient déjà configurés
- Avoir un hostname configuré (autre que celui par défaut).
- Avoir un IP domain configuré

Activer les services

Sélectionner les lignes **VTY** :

```
configure terminal
```

```
line vty 0 4
```

SSH

Créer une **clé RSA** de taille 2048 :

```
crypto key generate rsa general modulus 2048
```

Et activer le service SSH :

```
transport input ssh
```

Telnet

```
transport input telnet
```

Les deux

```
transport input ssh telnet
```

[Cisco] VLAN

Introduction

Les **VLANs** pour (*Virtual LAN*) sont des réseaux virtuels mis en place généralement sur les switchs pour segmenter les réseaux.

Généralités

Port Trunk

Connexion qui transporte le trafic de plusieurs VLANs entre les switchs.

Une étiquette VLAN est ajoutée aux trames réseau pour identifier à quel VLAN elles appartiennent dans l'en-tête **802.1Q (IEEE)**.

Une exception est faite pour le VLAN natif qui ne comporte pas de tag mais qui peut circuler sur une interface trunk.

Port Access

Associé à un seul VLAN, l'hôte n'a pas conscience d'être sur un VLAN (dédié aux terminaux).

Affichage des VLANs

Afficher les VLANs configurés

```
show vlan brief
```

Afficher les trunks configurés

```
show interface trunk
```

Configuration

Ils sont considérés dans les équipements Cisco comme des **interfaces** et se configurent donc de la même manière :

```
interface vlan <VLAN_NUMBER>
```

> Le **VLAN1** est présent par défaut et concerne tous les ports du switch s'il n'a pas été modifié.

On peut aussi définir une adresse IP pour le switch sur ce VLAN et une gateway si on le souhaite comme on le ferait sur une interface classique.

Créer un VLAN

En mode **configuration globale**, tapez les deux commandes suivantes :

```
vlan <VLAN_ID>
```

```
name <VLAN_NAME>
```

> La commande **name** est facultative (vous n'êtes pas obligé de définir un nom pour votre VLAN).

Port en mode access

Après avoir sélectionner l'interface, exécutez la commande suivante :

```
switchport mode access
```

```
switchport access vlan <VLAN_ID>
```

Port en mode trunk

Après avoir sélectionner l'interface, exécutez les deux commandes suivantes :

```
switchport trunk encapsulation dot1q
```

```
switchport mode trunk
```

Et pour sélectionner les VLANs qui pourront être utilisés sur l'interface, exécutez cette commande :

```
switchport trunk allowed vlan <VLAN_ID>,<VLAN_ID_2>
```

Quelques exemples pratiques si vous manipulez beaucoup de VLANs sur un port trunk :

```
switchport trunk allowed vlan 1,100-200
```

```
switchport trunk allowed vlan add 300
```

```
switchport trunk allowed vlan remove 100
```

```
switchport trunk allowed vlan except 400
```

```
switchport trunk allowed vlan all
```

> Par défaut, tous les VLANs sont autorisés sur les ports trunks !

On peut définir le **VLAN natif** sur un port trunk :

```
switchport trunk native vlan <VLAN_ID>
```

> Par défaut, il s'agit du **VLAN 1**.

Port en mode dynamique

Le port va négocier avec le voisin s'il doit passer en mode trunk ou non (technologie propriétaire à Cisco) :

```
switchport mode dynamic desirable
```

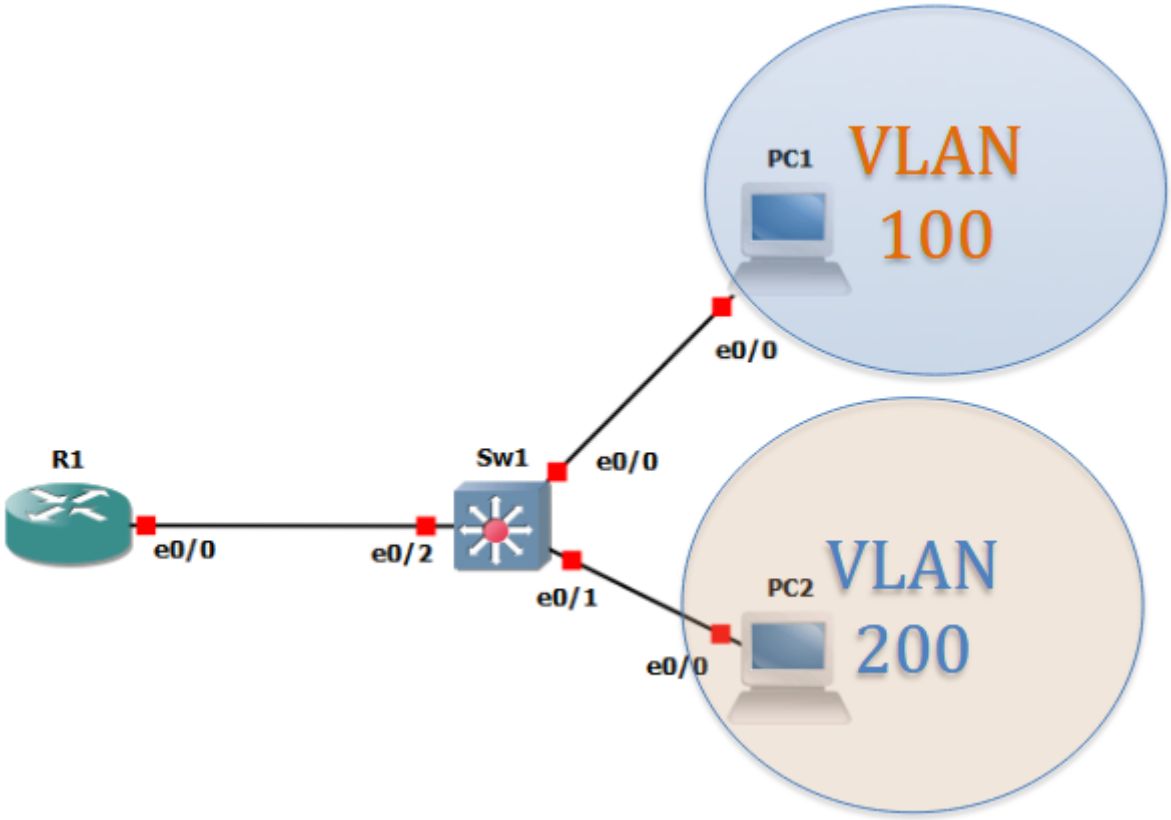
Et voici la commande pour configurer le port en mode trunk si mon voisin me le demande :

```
switchport mode dynamic auto
```

Configurer un routeur on-the-stick

L'objectif va être de configurer un routeur de manière à permettre la communication entre les différents VLANs.

On admet la topologie suivante :



Tout d'abord, l'interface qui va relier le routeur au switch devra être configuré en mode Trunk.

Ensuite, il va falloir créer et configurer les VLANs 100 et 200 sur le switch.

Puis la dernière étape consiste à configurer le routeur pour prendre en charge les VLANs.

Pour cela, on commence par segmenter l'**interface physique E0/0** en 2 **sous-interfaces** (virtuelles) avec les commandes :

```
interface e0/0.100
```

```
encapsulation dot1q 100
```

> Si le VLAN est un VLAN natif, ajoutez le mot-clé **native** à la fin de la commande précédente.

Puis on saisis l'adresse IP de l'interface virtuelle (qui correspond à la passerelle du VLAN 100) :

```
ip address <IP> <MASK>
```

On réitère pour la deuxième interface virtuelle :

```
interface e0/0.200
```

```
encapsulation dot1q 200
```

Puis on saisis l'adresse IP de l'interface virtuelle (qui correspond à la passerelle du VLAN 200) :

```
ip address <IP> <MASK>
```

Enfin, on démarre l'interface physique (qui va démarrer l'ensemble des sous-interfaces) :

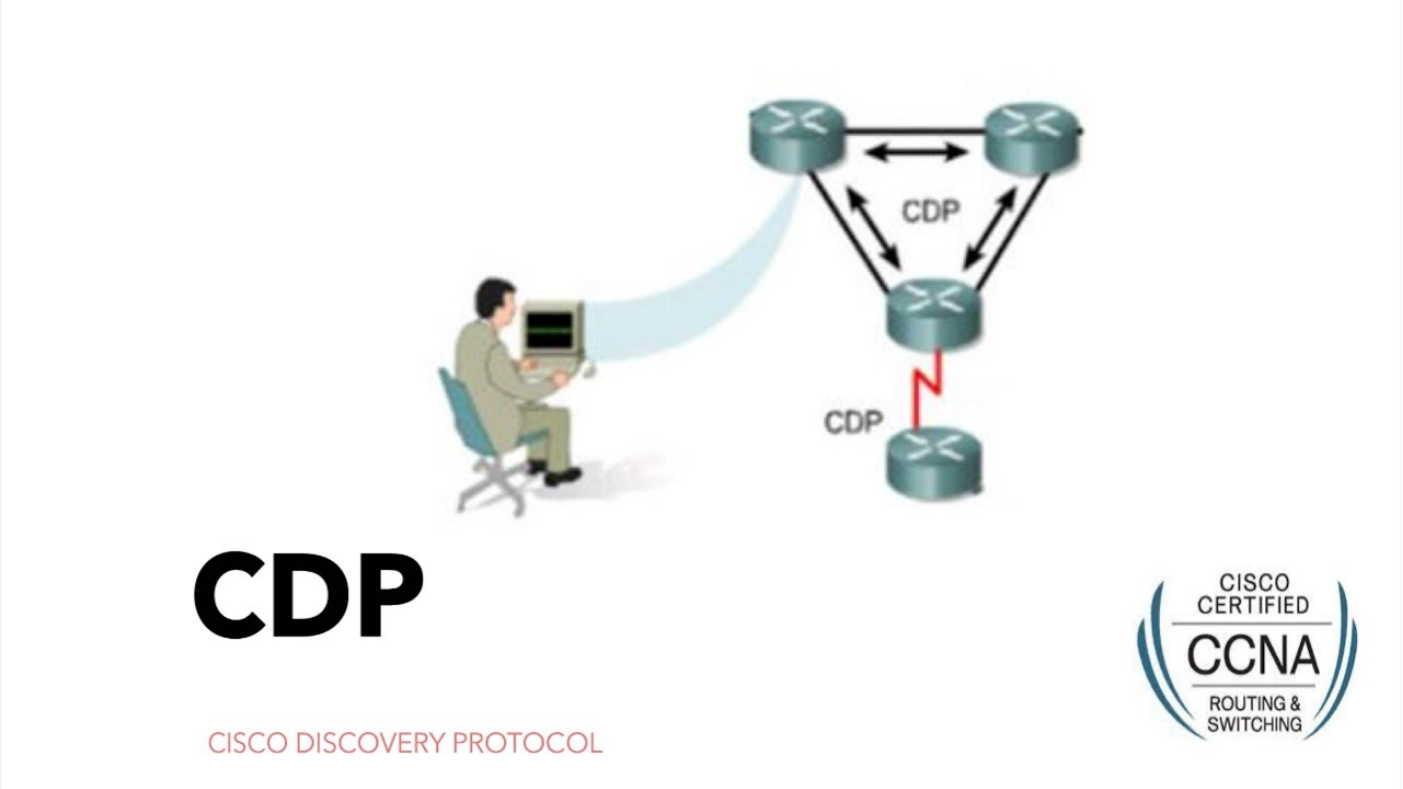
```
no shut
```

> Et n'oubliez pas de configurer la passerelle sur les machines hôtes !

[Cisco] CDP

Introduction

Le **CDP** pour *Cisco Discovery Protocol*, est un protocole propre aux équipements Cisco qui permet d'obtenir des informations sur les équipements voisins du réseau.



Fonctionnement

Lorsque le CDP est activé sur un équipement Cisco, il va envoyer une trame en **multicast** sur le réseau à l'adresse suivante :

```
0100.0ccc.cccc
```

Tous les équipements Cisco ont cette adresse multicast et reçoivent donc la trame s'ils sont sur le même réseau que l'émetteur.

Ce paquet est envoyé par défaut toutes les **60 secondes** qui peut être modifié par l'administrateur.

Ce timer est appelé **Hello timer**. Tous les équipements qui le reçoivent répondent en indiquant leur adresse MAC, adresse IP, hostname et le type d'équipement (routeur, switch etc).

Le protocole comprend un deuxième timer qu'on appelle le **Hold timer** qui correspond à la durée à partir de laquelle on considère qu'un équipement est mort (ou déconnecté du réseau). Sa valeur par défaut est de **180 secondes**.

Un équipement qui n'aura pas répondu après **3** trames CDP envoyés sera donc considéré comme mort.

Commandes

Activer CDP

```
cdp run
```

Désactiver CDP

```
no cdp run
```

Afficher la table CDP

```
show cdp neighbors
```

Pour l'afficher avec davantage de détails :

```
show cdp neighbors detail
```

[Cisco] IPv6

Introduction

De la même manière qu'il est possible de définir une adresse IPv4 sur une interface, il est possible d'attribuer une adresse IPv6.

Il est même possible d'en attribuer plusieurs pour une même interface.

Configuration

Tout d'abord, il faut passer dans le contexte de configuration globale :

```
configure terminal
```

Puis :

```
interface <INTERFACE_NAME> [SLOT/]<PORT>
```

On peut définir l'adresse IPv6 :

```
ipv6 address <IP> <CIDR_MASK>
```

Et démarrer l'interface :

```
no shutdown
```

Activer le routage IPv6

```
ipv6 unicast routing
```

[Cisco] DHCP

Introduction

Sur les routeurs Cisco, il est possible de mettre en place un **serveur DHCP** qui va se charger d'attribuer automatiquement des adresses IP aux hôte d'un réseau.

Il est aussi possible de créer un **relais DHCP**.

Client DHCP

Voici comment utiliser une adresse fournie par le serveur DHCP pour un client qui souhaite profiter de ce service depuis la configuration spécifique de l'interface :

```
ip address dhcp
```

```
no shut
```

Serveur DHCP

Accéder au mode de configuration globale :

```
conf t
```

Créer un **pool** DHCP (groupe bénéficiaire du service) :

```
ip dhcp pool <POOL_NAME>
```

Une fois dans le shell de configuration du pool, définissez le réseau d'action :

```
network <NET_ID> <CIDR_MASK>
```

Et définissez la **passerelle** fournie dans l'offre DHCP (généralement il s'agit de l'interface du routeur connecté au réseau cible du DHCP) :

```
default-router <GATEWAY_IP>
```

Exclure des IP de l'offre DHCP

Pour exclure une **seule** adresse IP :

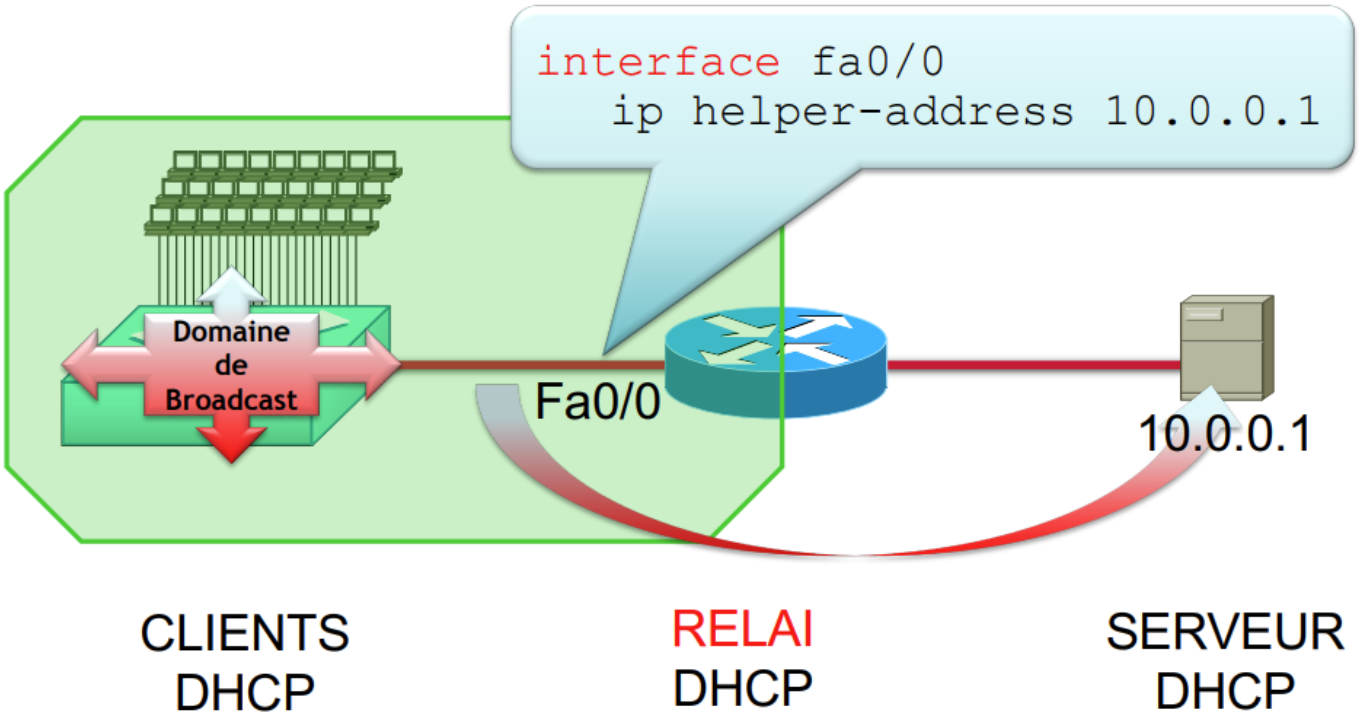
```
ip dhcp excluded-address <IP>
```

Pour exclure une **plage** d'adresses IP :

```
ip dhcp excluded-address <FROM_IP> <TO_IP>
```

Relais

L'objectif du **relai DHCP** est de configurer le routeur de sorte à transmettre les requêtes DHCP du réseau A qui contient de client DHCP vers le réseau B où il y a le serveur DHCP.



Sur l'interface du routeur appartenant au **réseau B**, exécutez la commande suivante :

```
ip helper-address 192.168.11.6
```

[Cisco] Sécurisation

Introduction

Vous retrouverez sur cette page des procédures et des commandes qui vous aideront à sécuriser des équipements Cisco.

Sécurisation

Auto-secure

La commande suivante va exécuter tout un tas de configurations pour sécuriser le système :

```
auto secure
```

Politique de mot de passe

Vous pouvez définir une politique de mot de passe en interdisant les mots de passe inférieurs à 8 caractères par exemple :

```
security passwords min-length <MIN_LEN>
```

Politique d'authentification

Il est possible de définir une **politique de sécurité d'authentification** où l'administrateur peut choisir 3 paramètres :

1. Le nombre de tentatives autorisé avant blocage.
2. L'interval de temps du compteur de nombre de connexion.
3. Le temps de blocage.

Voici la commande :

```
login block-for <3> attemps <2> within <1>
```

Politique d'inactivité

Par défaut, une période d'inactivité de 10 minute provoquera une déconnexion automatique sur une session EXEC (privilégiée).

Vous pouvez modifier cette valeur grâce à la commande suivante :

```
exec-timeout <MINUTE> <SECOND>
```

[Cisco] Historique

Introduction

L'historique de commandes peut être géré sur les systèmes IOS un peu de la même manière que sur Unix.

Manuel

Afficher l'historique

```
show history
```

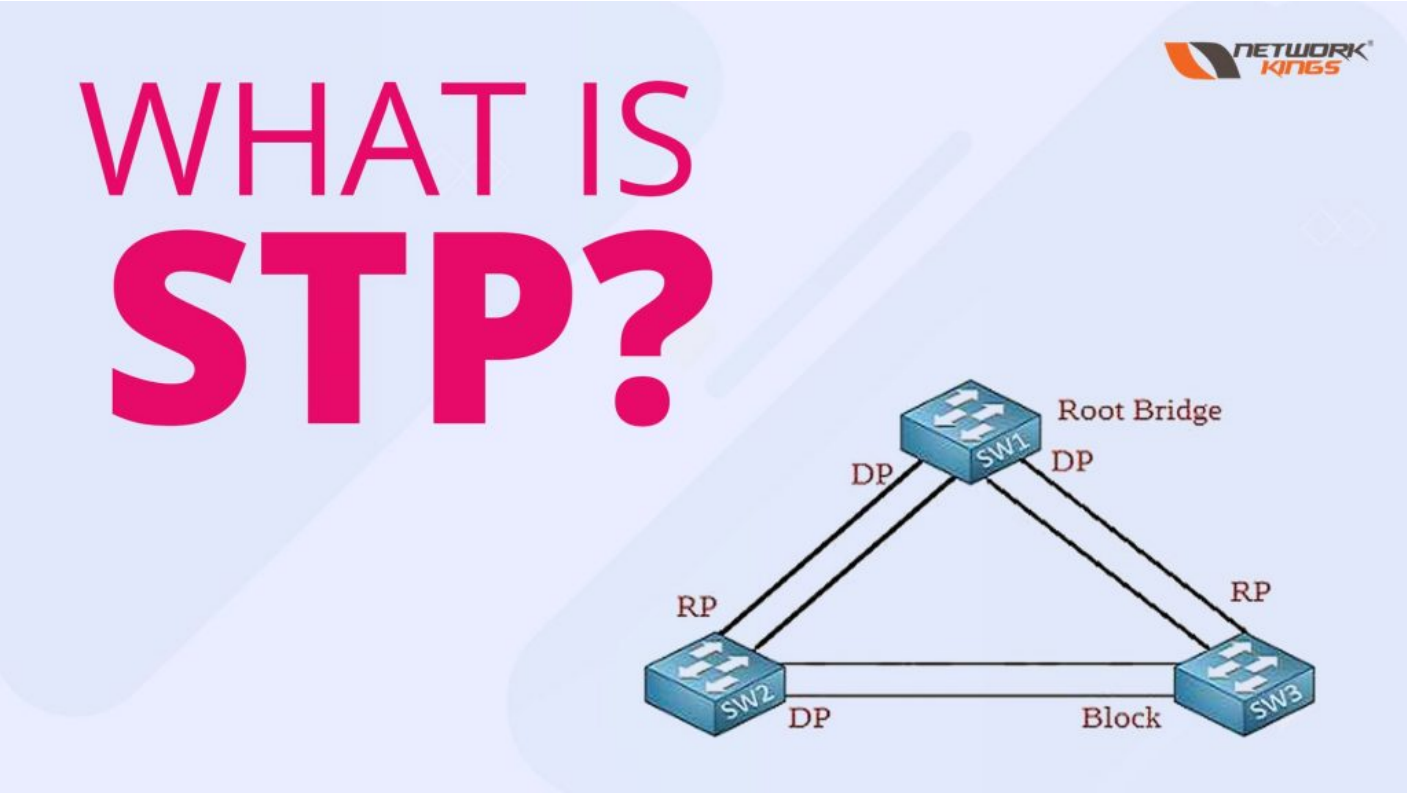
Définir la taille de l'historique

```
terminal history size <COMMAND_SIZE>
```

[Cisco] STP

Introduction

STP pour *Spanning-Tree Protocol*, est un protocole qui permet la tolérance à la panne ainsi que la répartition de charge entre les switchs sur un réseau.



Manuel

Afficher la configuration STP d'un VLAN

```
show spanning-tree vlan <VLAN_ID>
```

> Si aucun VLAN particulier n'est configuré, mettre le **VLAN ID à 1** (par défaut).

Définir la priorité PVSTP

```
spanning-tree vlan <VLAN_ID> priority <0-65535>
```

Définir le port priority

Après avoir sélectionné l'interface en question :

```
spanning-tree vlan <VLAN_ID> port-priority <0-255>
```

Définir le coût

```
spanning-tree vlan <VLAN_ID> cost <COST>
```

> La priorité la plus faible aura le meilleur BPDU !

Définir un port-fast

Une fois l'interface sélectionnée, entrez les commandes suivantes :

```
spanning-tree portfast
```

```
spanning-tree bpduguard enable
```

Définir un port point-to-point

```
spanning-tree link-type point-to-point
```

Définir un port-fast edge

```
spanning-tree portfast edge
```

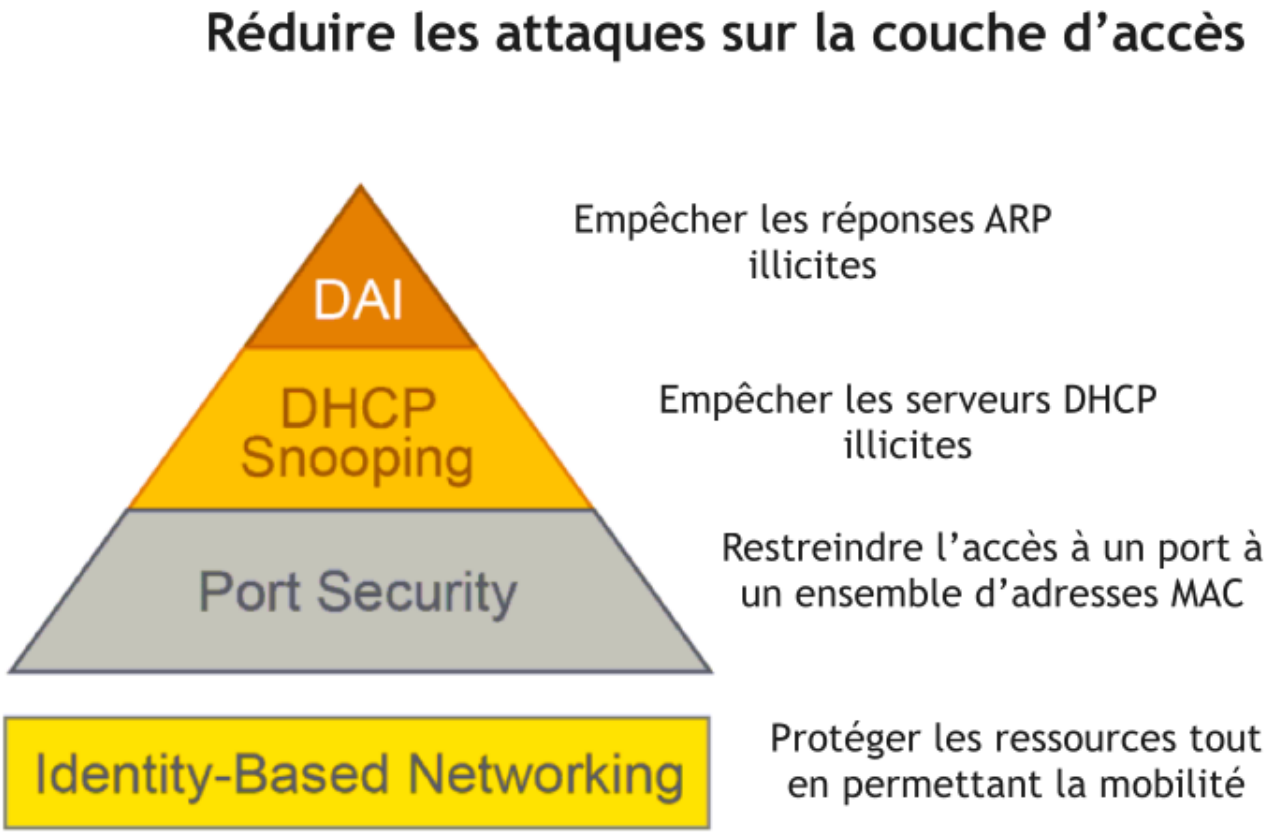
Activer le Rapid STP

```
spanning-tree mode rapid pvst
```

[Cisco] Sécurité des switchs

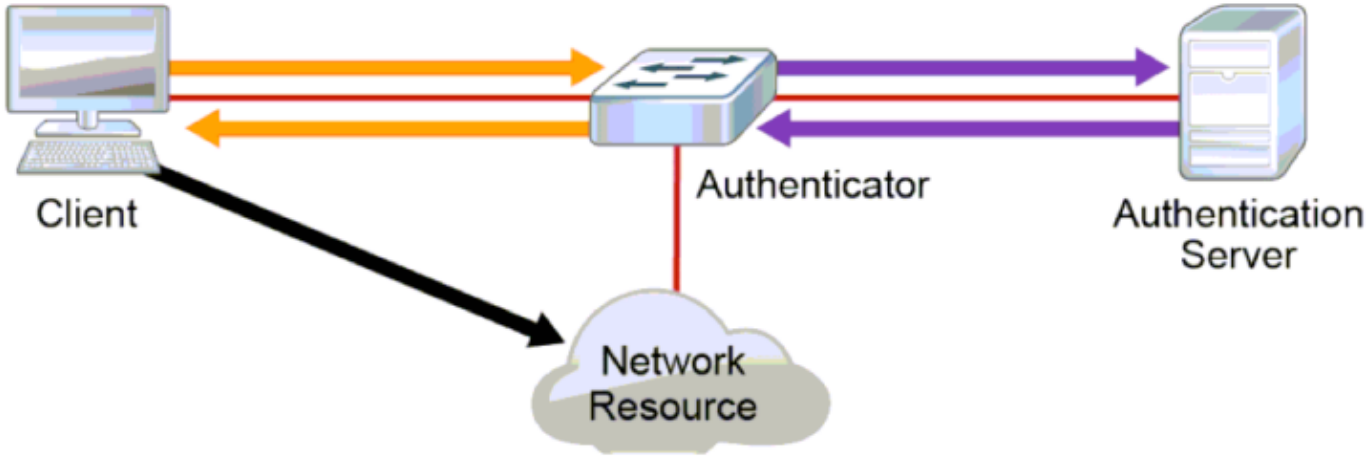
Introduction

Sur les switchs Cisco, il est possible d'activer des options pour renforcer la sécurité des équipements connectés au réseau.



Identify-Based Networking

Basé sur le protocole 802.1x, il permet d'exiger une identification pour accéder au réseau.



Autoriser une adresse MAC manuellement

Après avoir sélectionné l'interface :

```
switchport port-security mac-address <MAC>
```

```
switchport port-security
```

> Il faut autoriser chaque adresse MAC une à une, ce qui peut devenir fastidieux.

Autoriser la première adresse MAC

Après avoir sélectionné l'interface :

```
switchport port-security maximum 1
```

```
switchport port-security
```

> Seulement la première adresse MAC qui se connecte pourra accéder au réseau mais si le courant du switch se coupe et qu'un pirate se connecte, il pourra accéder au réseau.

Autoriser la première adresse MAC + sticky

Après avoir sélectionné l'interface :

```
switchport port-security maximum 1
```

```
switchport port-security mac-address sticky
```

```
switchport port-security
```

> Il s'agit de la méthode la plus recommandée.

Port Security

Shutdown

Avec cette méthode, si l'adresse MAC source d'une trame n'est pas autorisée, **l'interface sera coupée**.

Après avoir sélectionné l'interface :

```
switchport port-security violation shutdown
```

> Il s'agit du mode utilisé par défaut.

Restrict

Avec cette méthode, si l'adresse MAC source de la trame n'est pas autorisée, il **supprime la trame** et en **informe** l'administrateur.

Après avoir sélectionné l'interface :

```
switchport port-security violation restrict
```

Protect

Avec cette méthode, si l'adresse MAC source de la trame n'est pas autorisée, il **supprime la trame sans informer** l'administrateur.

Après avoir sélectionné l'interface :

```
switchport port-security violation protect
```

DHCP snooping

L'objectif est d'empêcher un pirate de mettre son propre serveur DHCP sur le réseau ou de DOS votre serveur DHCP.

Interface trusted

L'objectif va être de définir les ports sur lesquels les serveurs DHCP légitimes sont présents.

Tout d'abord, activez le DHCP snooping :

```
ip dhcp snooping
```

On peut aussi l'activer sur un VLAN spécifique :

```
ip dhcp snooping vlan <VLAN_ID>
```

Ensuite, sélectionnez l'interface sur laquelle le serveur DHCP est présent puis faite :

```
ip dhcp snooping trust
```

Vérifier les interfaces de confiance

```
show ip dhcp snooping
```

DAI

L'objectif va être d'empêcher l'ARP spoofing.

Pour cela, le DHCP snooping doit être configuré pour fonctionner.

Configurer une correspondance MAC/IP

Tout d'abord définissez une liste d'accès :

```
arp access-list <NAME>
```

Puis lancez la commande suivante :

```
permit ip host <IP> mac host <MAC>
```

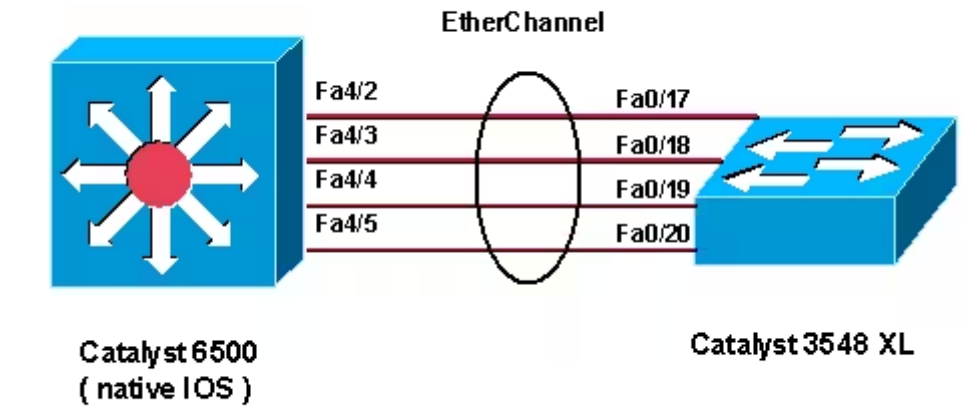
Et enfin :

```
ip arp inspection filter <NAME> vlan <VLAN_ID>
```

[Cisco] Etherchannel

Introduction

L'etherchannel permet de faire de l'aggrégation de lien dans un but de répartition de charge. L'objectif est de "fusionner" (aggréger) plusieurs liens pour augmenter la bande passante disponible.



Manuel

Création d'un PortChannel

L'objectif va être de créer un **PortChannel**, aussi appelé **Etherchannel**, qui est une interface virtuelle qui va aggréger tous les ports que l'on aura sélectionné au préalable.

Pour cela on sélectionne la plage d'interface physique que l'on veut aggréger :

```
int range e0/0-1
```

Ou on les sélectionne manuellement :

```
int e0/0, e0/1
```

Puis on créé le PortChannel :

```
channel-group <CH_ID> mode <desirable|active>
```

> Le choix du mot clé **desirable** sélectionnera le protocole **PaGP** (propriétaire Cisco) pour le PortChannel et le mot-clé **active** sélectionnera le protocole **LACP** (standard libre).

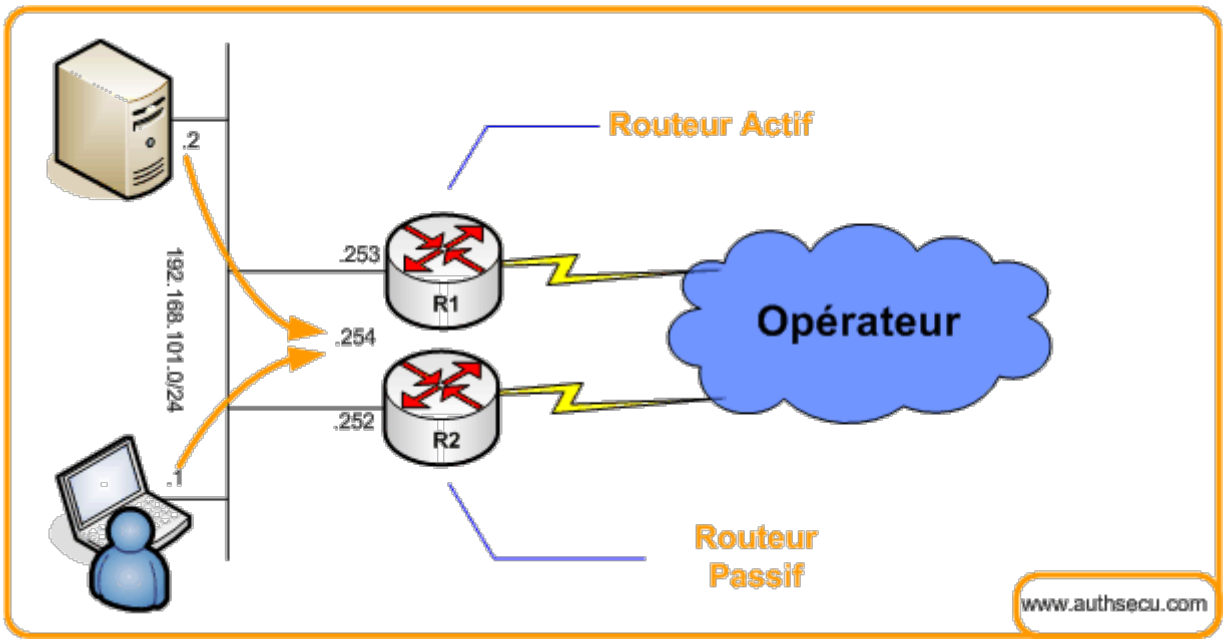
Affichage des Etherchannels

```
show etherchannel summary
```

[Cisco] HSRP

Introduction

Le **HSRP** est un protocole qui permet d'établir de la haute disponibilité pour les passerelles par défaut sur un réseau. Il fonctionne en UDP sur le port 1985.



Manuel

Créer le routeur virtuel

Tout d'abord, créez un routeur virtuel sur chaque routeur physique en sélectionnant l'interface qui est sur le réseau qui bénéficiera de la HA :

```
standby <POOL_ID> ip <VIRTUAL_IP_GATEWAY>
```

> Le pool est le groupe de routeur qui devra être redondé (il est bien de mettre le VLAN ID comme pool ID).

Définir une priorité

Si vous voulez que R1 soit le routeur actif, il faut lui mettre une priorité supérieur à R2 (100 par défaut) :

```
standby <POOL_ID> priority <PRIORITY>
```

Activer la préemption

La préemption permet de maintenir le routeur actif qui a la plus grande priorité même après une panne :

```
standby 1 preempt
```

> Doit impérativement être configuré sur les deux routeurs !

Modifier les timers

Par défaut, le timer HELLO est sur 3 secondes et le timer HOLD est sur 10 secondes :

```
standby <POOL_ID> <HELLO_TIME> <HOLD_TIME>
```

Si vous voulez l'indiquer en milliseconde :

```
standby <POOL_ID> msec <HELLO_TIME> msec <HOLD_TIME>
```

Activer le tracking

Le tracking permet de superviser une interface

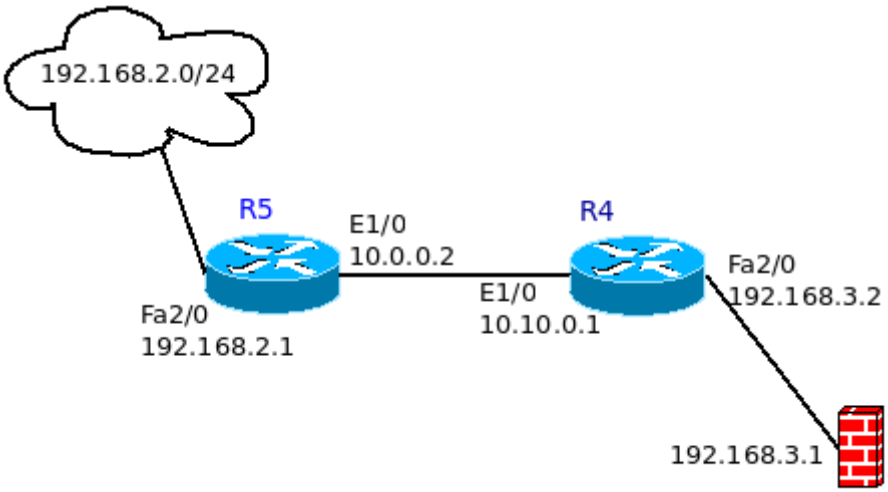
```
track <HELLO_TIME> interface <IFACE> line-protocol
```

```
standby <POOL_ID> track <HELLO_TIME> <HOLD_TIME>
```

[Cisco] Routage statique

Introduction

Le routage va permettre de faire communiquer les réseaux entre-eux en passant par des routeurs (hops).



Manuel

Désactiver le routage

Si vous souhaitez désactiver le routage (pour faire fonctionner l'équipement comme un PC) :

```
no ip routing
```

Créer une route statique

```
ip route <DST_NET_ID> <DST_NET_MASK> <NEXT_HOP>
```

> Le **Next Hop** correspond à l'interface du routeur qui est sur votre réseau par laquelle vous voulez passer pour accéder au réseau de destination.

Pour ajouter une distance administrative personnalisée (définie à 1 par défaut) :

```
ip route <DST_NET_ID> <DST_NET_MASK> <NEXT_HOP> <DIST>
```

> La route de backup n'apparaîtra pas dans la table de routage (il n'affiche que la route préférée).

Créer une route par défaut

```
ip route 0.0.0.0 0.0.0.0 <NEXT_HOP>
```

Afficher les routes

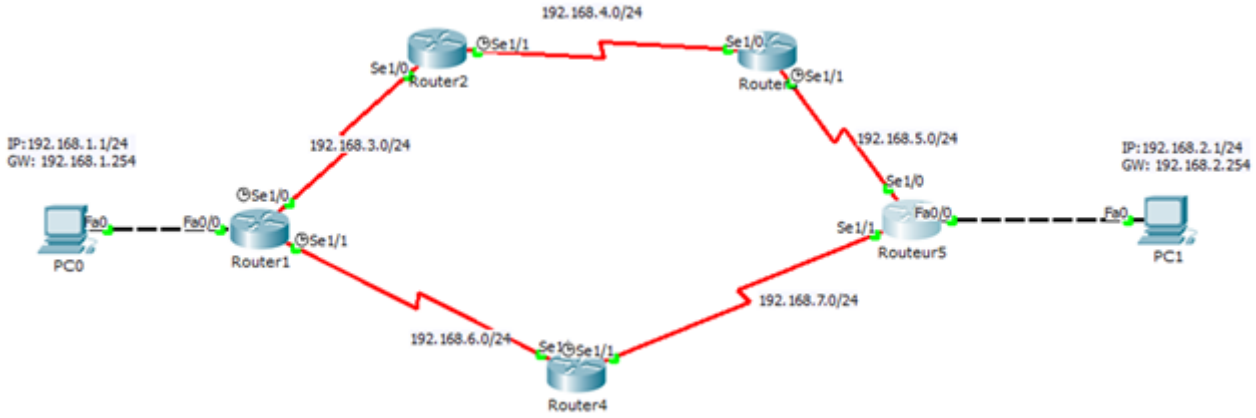
```
show ip route
```

[Cisco] Routage RIP

Introduction

Le routage **RIP** pour *Routage Information Protocol*, est un protocole de routage qui fonctionne par rumeur, c'est à dire que chaque routeur qui utilise ce protocole va envoyer toutes les 30 secondes, sa table de routage à ses voisins.

À savoir que RIP dans sa version 2 fonctionne en multicast via l'IP 224.0.0.9 sur le port 520 en UDP.



Manuel

Afficher les interface RIP

```
show ip route rip
```

Debug RIP

```
debug ip rip
```

Activer RIP

```
router rip
```

```
version 2
```

> L'utilisation de la version permet d'utiliser le multicast au lieu d'un broadcast.

Puis on active RIP sur chaque interface en indiquant l'identifiant du réseau de l'interface que l'on souhaite :

```
network <NET_ID>
```

Désactiver la globalisation

```
router rip
```

```
no auto-summary
```

[Cisco] Routage OSPF

Introduction

OSPF pour *Open Shortest Path First* est un protocole de routage dynamique standard. Il a une distance administrative de 110 et sa métrique est basée sur la somme des coûts (10^8/Bande Passante).

Configuration

Activer OSPF

Sur une seule interface :

```
int fa0/0
ip ospf <PROC> area <AREA>
```

> Le numéro de processus est superflu (mettre 1), et le numéro d'aire dépend de votre infrastructure (0 par défaut).

Sur toutes les interfaces du routeur :

```
router ospf <PROC>
```

```
network <NET_ID> <REV_MASK> area <AREA>
```

> Le **<REV_MASK>** correspond au masque inversé (0 devient 255 et 255 devient 0).

Changer la valeur de référence

```
router ospf 1
auto-cost reference-bandwidth <BW>
```

> Par défaut cette valeur est définie à **100** (en Mb/s).

Configurer le coût d'une interface manuellement

```
int e0/0
ip ospf cost <COST>
```

Configurer la bande passante d'une interface manuellement

```
int e0/0
bandwidth <BW>
```

Changer les valeurs des timers

Pour le timer **Hello** :

```
int e0/0
ip ospf hello-interval <TIME>
```

Pour le timer **Dead** :

```
int e0/0
ip ospf dead-interval <TIME>
```

Définir la priorité

Pour définir manuellement quel routeur sera désigné (**DR**) et quel routeur sera désigné en backup (**BDR**), on peut modifier la priorité :

```
int e0/0
ip ospf priority <PRIORITY>
```

> La priorité **la plus grande** du réseau sera élue comme DR et une priorité de 0 indiquera que le routeur ne peut être DR ou BDR mais seulement DROTHER.

Créer des annonces globalisantes

Si nous avons un cas avec plusieurs sous-réseaux de destination qui ont des adresses contiguës, nous pouvons créer une annonce globalisante pour créer qu'une seule entrée dans la table de routage OSPF :

```
router ospf <PROC>
```

Puis :

```
area <AREA> range <IP> <MASK>
```

> Les deux commandes ci-dessus sont à réaliser sur le routeur situé à la frontière entre les deux aires.

> Le champ <AREA> doit être remplacé par le numéro de l'aire du réseau de destination.

Si vous utiliser un routeur qui utilise 2 protocoles (dont un qui n'est pas OSPF) sur deux pattes différentes (routeur dit ASBR), on doit configurer de la sorte :

```
router ospf 1
summary-address range <NET_ID> <MASK>
```

Annoncer une route par défaut

Sur le routeur par défaut :

```
default-information originate
```

Ou alors si le routeur n'a pas de route par défaut :

```
default-information originate always
```

On peut aussi définir des métriques :

```
default-information originate metric <METRIC>
```

Afficher la configuration OSPF

```
show ip ospf interface brief
```

Et pour afficher les voisins qui utilisent la même configuration OSPF que notre routeur et qui sont dans la même aire :

```
show ip ospf neighbor
```

Pour afficher la cartographie des routes connues :

```
show ip ospf database
```

Afficher la table de routage OSPF :

```
show ip route ospf
```

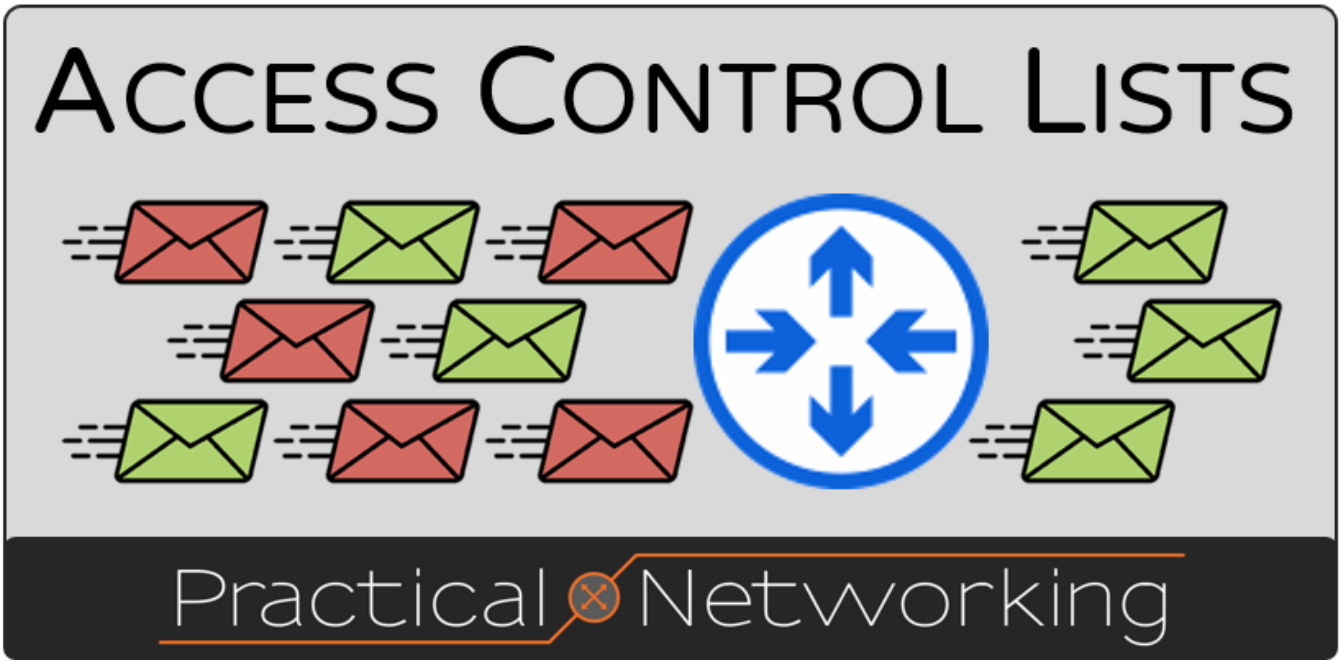
Afficher le Router-ID :

```
show ip ospf | inc ID
```

[Cisco] ACL

Introduction

Les **ACLs**, pour *Access Control Lists*, sont des listes de sécurité qui permettent de filtrer les adresses IPs sources pour accéder à certaines ressources.



Configuration

Créer une Access List

Pour une ACL standard :

```
access-list <ACL_ID> <permit|deny> <IP> <REV_MASK>
```

> Le champ **<REV_MASK>** correspond au masque inversé de l'IP à laquelle la règle doit être appliquée.

Et pour une ACL étendue :

```
ip access-list extended <NAME|ID>
```

Puis définissez vos règles :

```
<LINE> <permit|deny> <PROTOCOL> <SRC_IP> <SRC_REV_MASK> <DST_IP> <DST_REV_MASK> eq <PORT>
```

> Notez que les règles fonctionnent de haut en bas et que par défaut, tout le trafic est bloqué.

Appliquer une ACL

Pour appliquer une ACL à une interface :

```
int e0/0
ip access-group <ACL_ID|NAME> <in|out>
```

Pour l'appliquer aux lignes VTY :

```
line vty 0 4
access-class <ACL_ID|NAME> <in|out>
```

Afficher les ACLs

Pour afficher vos ACLs vous pouvez utiliser la commande suivante :

```
show access-lists
```

Vous pouvez aussi afficher une ACL spécifique :

```
show access-lists <ACL_ID|ACL_NAME>
```

[Cisco] NAT et PAT

Introduction

Le NAT pour Network Address Translation et le PAT pour Port Address Translation permmettent d'utiliser une adresse IP publique pour communiquer sur Internet, ce qui serait impossible avec des adresses IP privées.

Configuration

NAT et PAT statique

Tout d'abord, il faut définir les interfaces qui représentent le réseau interne :

```
int e0/0
ip nat inside
```

Et le réseau externe :

```
int e0/1
ip nat outside
```

Ensuite il faut définir l'adresse IP privée qui pourra utiliser un adresse IP publique via le NAT :

```
ip nat inside source static <PRIVATE_IP> <PUBLIC_IP>
```

Pour définir le **PAT statique** :

```
ip nat inside source static tcp <PRIVATE_IP> <PRIVATE_PORT> <PUBLIC_IP> <PUBLIC_PORT>
```

NAT et PAT dynamique

Tout d'abord, il faut définir les interfaces qui représentent le réseau interne :

```
int e0/0
ip nat inside
```

Et le réseau externe :

```
int e0/1
ip nat outside
```

On définit le pool dynamique :

```
ip nat pool <POOL> <BEGIN_RANGE_IP> <END_RANGE_IP> netmask <MASK>
```

Puis on créé une **access list** pour choisir quelles sont les IPs qui bénéficierons du NAT dynamique :

```
access-list <ACL_ID> permit <NET_ID> <REV_MASK>
```

Et enfin, on active le NAT :

```
ip nat inside source list <ACL_ID> pool <POOL>
```

Ou alors pour faire du PAT, on ne définit pas de pool et on active le PAT :

```
ip nat inside source list <ACL> interface <PUBLIC_INT> overload
```

[Cisco] QoS

Introduction

La **QoS** ou qualité de service est la mise en place de priorité selon le type de trafic.

Configuration