Le but de cette séquence est d'identifier, suivant le protocole utilisé, la route empruntée par un paquet ainsi que de comprendre le principe des protocoles RIP et OSPF.

I) Adressage IP d'une machine :

Chaque « objet IP » est identifié par une adresse IP qui contient :

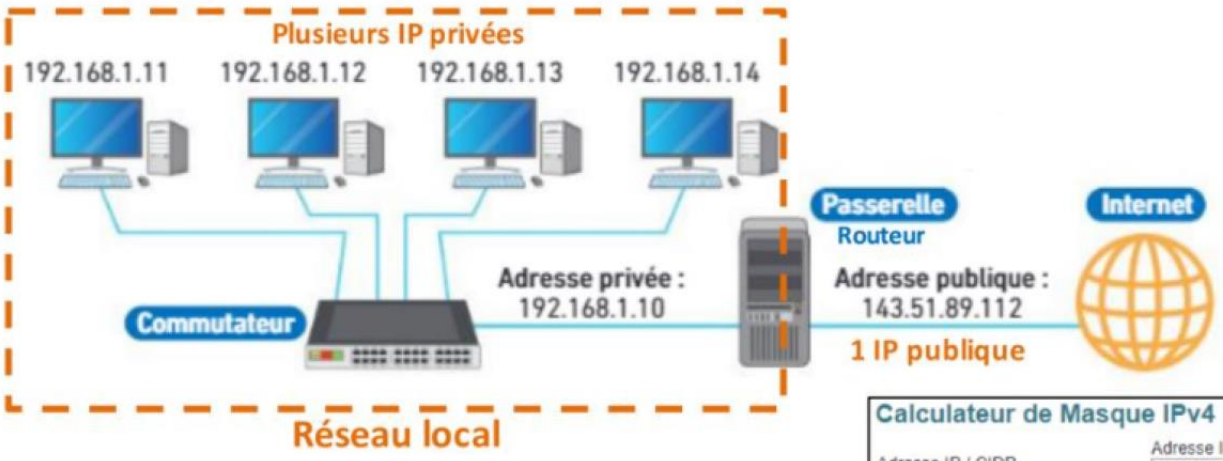
- L'adresse du réseau IP local (extraite grâce au « netmask » ou « masque de sous réseau ») ;
- Le numéro de la machine dans le réseau IP local

Une adresse IPV4 est un identifiant numérique à 32 bits (4 octets)

Chaque « Objet IP » est physiquement connecté à un réseau local par Ethernet, wifi ou Bluetooth.

La communication avec d'autre « Objets IP » appartenant au même réseau se fait directement via le réseau local de niveau 2 par l'intermédiaire d'un « switch » ou « commutateur ».

La communication avec d'autres « Objets IP » d'autres réseau IP distants fait via des passerelles de niveau 3 ou routeur.

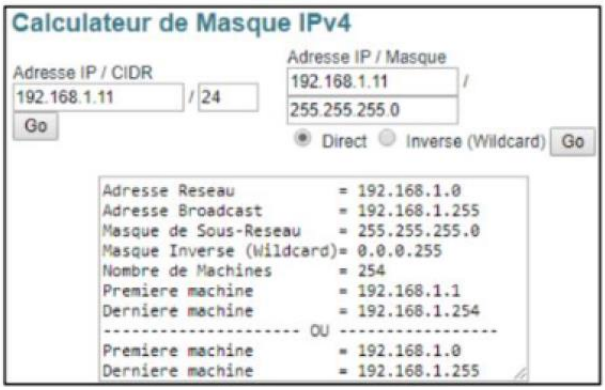


Réseau Machine

IPv4 : 192.168.1.11 (codée sur 4 octets)

Masque de sous réseau : 255.255.255.0 ou 24 (24 bits = 3 octets)

⇨ 256 machines max connectées sur le réseau local



Source : https://cric.grenoble.cnrs.fr/Administrateurs/Outils/CalculMasque/

Remarque : grâce à des adresses de 128 bits au lieu de 32 bits, IPv6 dispose d'un espace d'adressage bien plus important que l'IPv4 (près de 7.9×10^{28} de fois plus) et résout donc le problème de pénurie d'adresses IPv4 publiques liée à la multiplication des objets connectés de la vie quotidienne.

Internet résulte de l'interconnexion de réseaux par des routeurs. Un routeur est composé d'un nombre plus ou moins important d'interfaces réseau (cartes réseau).

Le schéma figure 1 montre 15 machines réparties en 6 réseaux locaux et reliés par 8 routeurs.

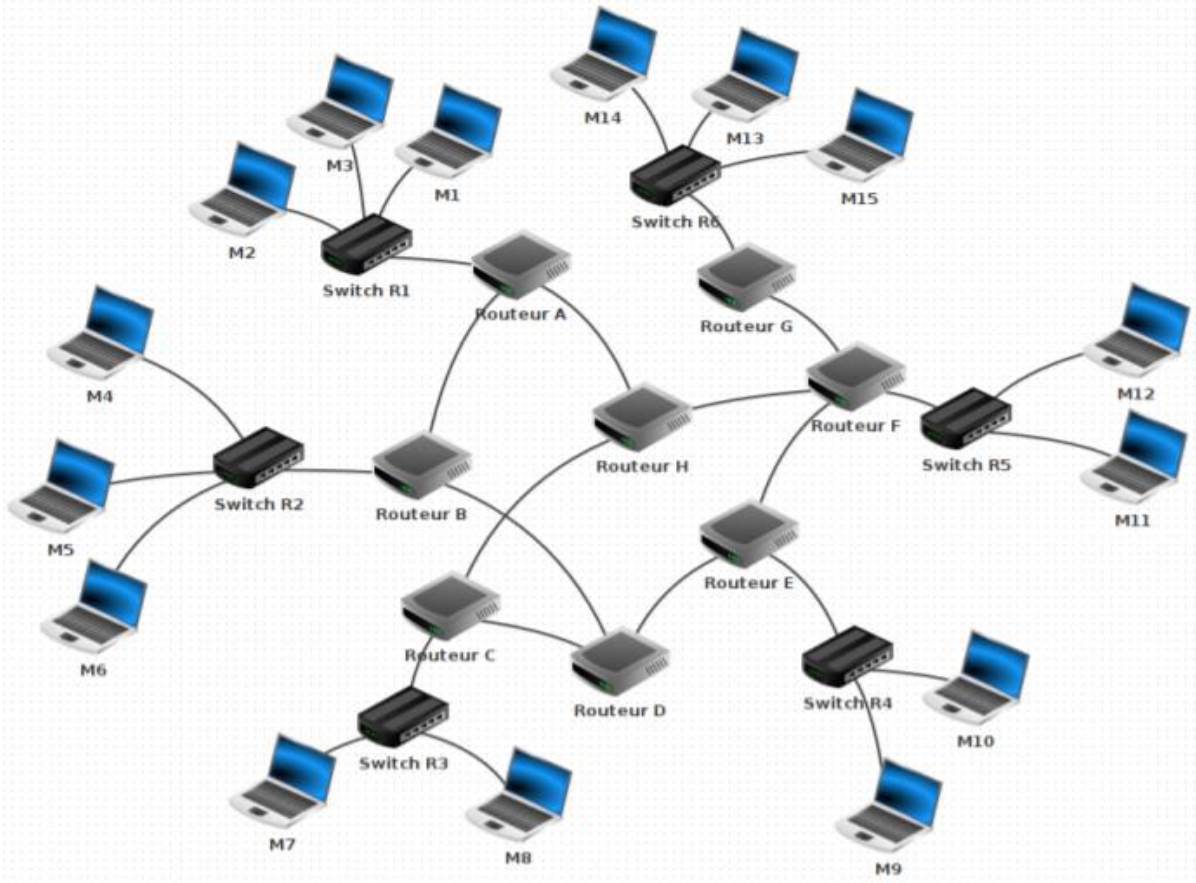


Figure 1

- Cas n°1 : M1 veut communiquer avec M3 :
  Le paquet est envoyé de M1 vers le switch R1, R1 "constate" que M3 se trouve bien dans le réseau local 1, le paquet est donc envoyé directement vers M3.
  On peut résumer le trajet du paquet par : M1 → R1 → M3

- Cas n°2 : M1 veut communiquer avec M6 :
  Le paquet est envoyé de M1 vers le switch R1, R1 « constate » que M6 n'est pas sur le réseau local 1, R1 envoie donc le paquet vers le routeur A. Le routeur A n'est pas connecté directement au réseau localR2 (réseau local de la machine M6), mais il "sait" que le routeur B est connecté au réseau local 2. Le routeur A envoie le paquet vers le routeur B. Le routeur B est connecté au réseau local 2, il envoie le paquet au Switch R2. Le Switch R2 envoie le paquet à la machine M6. ***Soit : M1 → R1→ Routeur A → Routeur B → R2 → M6***

- Cas n°3 : M1 veut communiquer avec M9 :
  M1 → R1 → Routeur A → Routeur B → Routeur D → Routeur E → R4 → M9 ou bien :
  M1 → R1 → Routeur A → Routeur H → Routeur F → Routeur E → R4 → M9
  Il est très important de bien comprendre qu'il existe souvent plusieurs chemins possibles pour relier 2 ordinateurs

- cas n°4 : M13 veut communiquer avec M9 :
  cas a : M13 → R6 → Routeur G → Routeur F → Routeur E → R4 → M9
  ou encore :
  cas b : M13 → R6 → Routeur G → Routeur F → Routeur H → Routeur C →Routeur D → Routeur E → R4 → M9
  On pourrait penser que le **chemin cas a** est plus rapide et donc préférable au **chemin cas b** cela est sans doute vrai, mais imaginez qu'il y ait un problème technique entre le Routeur F et le Routeur E, l'existence du chemin "Routeur F → Routeur H" permettra tout de même d'établir une communication entre M13 et M9.

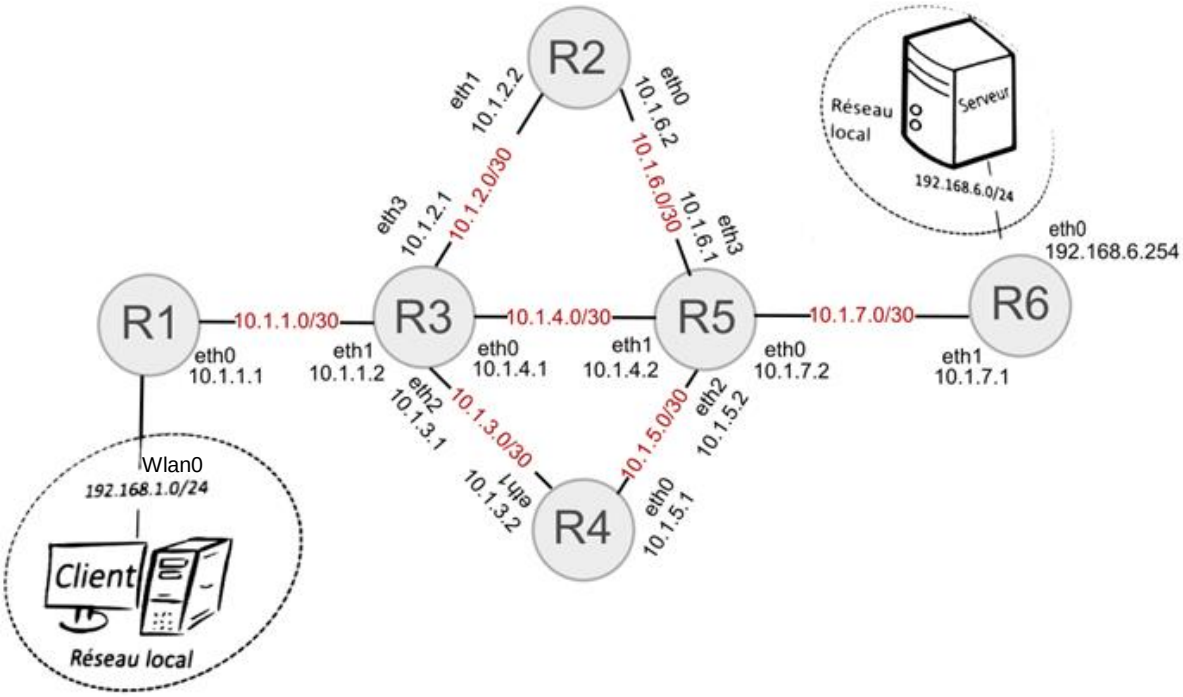
1) Déterminer un chemin possible permettant d'établir une connexion entre la machine M4 et M14 ?

2) Après avoir téléchargé le fichier filius (reseau.fls) correspondant au schéma de la figure 1, passez en mode simulation (flèche verte). Utilisez la ligne de commande pour tester la communication entre M4 et M14 avec un « ping » faire apparaitre le résultat et en déduire la possibilité de communication ou pas.

3) Utiliser la commande "traceroute" pour voir le chemin emprunté par les paquets ?

Comment ça marche ?

Deux machines dans le même réseau local doivent posséder la même adresse réseau. On présente ici la topologie d'un réseau avec 6 routeurs.



Les routeurs **R1** et **R6** sont des **routeurs d'accès** alors que les routeurs **R2..R5** sont des **routeurs internes**.

Les adresses IP utilisées par les machines sont indiquées par une paire **sous-réseau/masque**

Exemple R1 et R3 sont reliés à un sous réseau dont l'adresse est 10.1.1.0 et le masque 30 indique que les (32-30 bits) 2 derniers bits peuvent être utilisé pour associer des adresses IP aux machines ainsi R1 peut être associé à l'adresse 10.1.1.1(eth0) par exemple et R3 à l'adresse 10.1.1.2 (eth1).

L'adresse 10.1.1.0 ne peut être attribuée étant donné que c'est celle du réseau ainsi que la dernière (10.1.1.3) qui sera utilisée pour le broadcast (on communique avec tous les postes connectés)

Lorsqu'il reçoit un paquet, un routeur l'analyse pour récupérer l'adresse de sa destination. En fonction de cette adresse, il doit choisir vers quel routeur voisin retransmettre ce paquet pour le faire progresser vers sa destination.

Il choisit ce voisin à l'aide de sa table de routage qui associe les adresses de destination à des adresses de routeurs. De cette manière, un paquet transite de routeur en routeur jusqu'au client ou au serveur à qui il est destiné.

Par exemple, si le client veut envoyer un message au serveur, il le transmet à son routeur d'accès R1 qui n'a d'autre choix que de le renvoyer à R3 .

R3 peut soit le transmettre à R2 ou à R4 ou à R5. C'est la table de routage de routage de R3 qui indique quel voisin choisir. Ce processus est répété de la même manière pour chaque routeur et, en progressant de cette façon, le paquet va finir par arriver au routeur R6 qui pourra ainsi le délivrer au serveur.

Comment choisir la meilleure route ?

Le plus court chemin est-il aussi le plus rapide. Les table de routages sont-elles figées ou au contraire peuvent-elle évoluer de façon dynamique au gré du changement de la typologie du réseau (routeur en panne, liaisons rompus, ajout de routeur). C'est ce que nous allons voir maintenant.

II) Le protocole RIP

Des algorithmes sont implémentés dans les routeurs (Protocoles de routages).
Ils vont permettre au routeur de s'échanger des informations pour découvrir la topologie du réseau de façon dynamique. Les tables de routage sont recalculées à chaque changement du réseau.

Lors de **son initialisation,** la table de routage d'un routeur appliquant le protocole RIP contient uniquement les réseaux qui sont connectés à lui.
La mise en place du protocole RIP est simple à mettre en œuvre. Il est adapté pour les réseaux de petites tailles (15 routeurs maximum). La table est mise à jour toutes les 30 secondes.

Table de routage de R1			
Destination	Passerelle	Interface	Distance
10.1.1.0/30		eth0	1
192.168.1.0/24		Wlan0	1

On voit ainsi que la table du routeur R1 a des informations sur 2 destinations :

- Le sous réseau 10.1.0.0 qui le relie à R3 ainsi que
- Le sous-réseau 192.168.1.0/24 grâce auquel il est connecté à la machine client.

La colonne passerelle est vide puisque R1 peut atteindre ces 2 destinations directement.

L'interface eth0 signifie que R1 est directement relié au sous-réseau 10.1.1.0 /30 et que l'interface qu'il utilise pour transmettre des paquets est une carte Ethernet (eth).
Le numéro associé au nom de l'interface permet de savoir quelle carte utiliser parmi toutes celles disponibles sur un routeur.

L'interface Wlan0 indique que R1 est connecté au sous-réseau 192.168.1.0/24 via une interface sans fil (Wan)

La colonne distance indiquant le nombre de routeur à traverser au départ vaut 1.

De même, la table de routage de R3 est :

Table de routage de R3			
Destination	Passerelle	Interface	Distance
10.1.1.0/30		eth1	1
10.1.2.0/30		eth3	1
10.1.3.0/30		eth2	1
10.1.4.0/30		eth0	1

4) Il en va de même pour tous les autres routeurs, compléter alors la table de routage pour tous les autres routeurs

Table de routage de R2			
Destination	Passerelle	Interface	Distance
Table de routage de R4			
Destination	Passerelle	Interface	Distance

Table de routage de R5			
Destination	Passerelle	Interface	Distance

Table de routage de R6			
Destination	Passerelle	Interface	Distance

APRES cette phase d'initialisation, un routeur poursuit le protocole en échangeant des demandes avec ses voisins.
Ainsi lorsqu'un de ses voisins reçoit une telle demande, il doit accuser réception en lui renvoyant sa table en réponse.

4 cas possibles alors :

Cas1 : Il découvre **une nouvelle route** vers un sous-réseau qui lui était jusque-là inconnu. Il l'**inscrit** dans sa table.

Cas2 : Il découvre **une route plus courte** vers un sous-réseau connu mais passant par un autre routeur. Il **efface l'ancienne route** de sa table et **inscrit la nouvelle**.

Cas3 : Il reçoit une **nouvelle route plus longue**. Il **l'ignore.**

Cas 4 : Il reçoit **une route existante**, mais **plus longue**, vers un routeur passant **par le même voisin**. Cela veut dire **qu'un problème est apparu sur son ancienne route**. Il **met** donc **à jour sa table avec cette nouvelle route**.

Lorsqu'un routeur reçoit une route, il doit bien prendre soin de considérer que la distance associée à cette route doit être augmenté de 1.

Ainsi, après avoir échangé une demande RIP avec R3, R1 contient :

Table de routage de R1			
Destination	Passerelle	Interface	Distance
10.1.1.0/30		eth0(10.1.1.1)	1
192.168.1.0/24		Eth1	1
10.1.2.0/30	10.1.1.2	eth0(10.1.1.1)	2
10.1.3.0/30	10.1.1.2	eth0(10.1.1.1)	2
10.1.4.0/30	10.1.1.2	eth0(10.1.1.1)	2

5) Compléter alors la table de routage de R3 après cet échange.

Table de routage de R3			
Destination	Passerelle	Interface	Distance
10.1.1.0/30		eth1	1
10.1.2.0/30		eth3	1
10.1.3.0/30		eth2	1
10.1.4.0/30		eth0	1

En répétant ces demandes RIP et en mettant à jour leurs tables de routage selon l'algorithme précédent les routeurs vont finir au bout d'un certain temps par avoir la même vision du réseau et des meilleures routes à suivre pour acheminer un paquet. Voici la table finale du routeur R1.

Table de routage de R1			
Destination	Passerelle	Interface	Distance
10.1.1.0/30		eth0(10.1.1.1)	1
192.168.1.0/24		eth1	1
10.1.2.0/30	10.1.1.2	eth0(10.1.1.1)	2
10.1.3.0/30	10.1.1.2	eth0(10.1.1.1)	2
10.1.4.0/30	10.1.1.2	eth0(10.1.1.1)	2
10.1.7.0/30	10.1.1.2	eth0(10.1.1.1)	3
192.168.6.0/24	10.1.1.2	eth0(10.1.1.1)	4

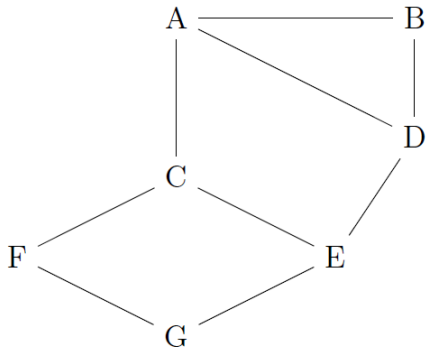
Détection de pannes :

Le protocole RIP doit permettre également de déterminer si une liaison est en panne.
Pour cela un routeur considère qu'un voisin est en panne s'il ne reçoit pas de réponse à une demande RIP après un certain laps de temps.

- Par défaut, ce délai est de 3 minutes. Quand un routeur détecte qu'un sous-réseau devient inaccessible, il envoi à ses voisins cette information sous la forme d'une route avec une distance infinie, qui correspond à une valeur de 16.

- La mise en place du protocole RIP est simple à mettre en œuvre. Comme nous venons de le voir ce protocole est adapté pour les réseaux de petites tailles (15 routeurs maximum), de plus RIP ne prend pas en compte la vitesse des liens, au contraire du protocole OSPF que nous verrons plus tard.

Exercice 1 :
Soit un graphe représentant les liaisons entre différents routeurs :



Le protocole RIP permet de construire les tables de routage des différents routeurs, en indiquant pour chaque routeur la distance, en nombre de sauts, qui le sépare d'un autre routeur.

Pour le réseau ci-dessus, on dispose des tables de routage suivantes :

Table de routage du routeur A		
Destination	Routeur suivant	Distance
B	B	1
C	C	1
D	D	1
E	C	2
F	C	2
G	C	3

Table de routage du routeur B		
Destination	Routeur suivant	Distance
A	A	1
C	A	2
D	D	1
E	D	2
F	A	3
G	D	3

Table de routage du routeur C		
Destination	Routeur suivant	Distance
A	A	1
B	A	2
D	E	2
E	E	1
F	F	1
G	F	2

Table de routage du routeur D		
Destination	Routeur suivant	Distance
A	A	1
B	B	1
C	E	2
E	E	1
F	A	3
G	E	2

Table de routage du routeur E		
Destination	Routeur suivant	Distance
A	C	2
B	D	2
C	C	1
D	D	1
F	G	2
G	G	1

Table de routage du routeur F		
Destination	Routeur suivant	Distance
A	C	2
B	C	3
C	C	1
D	C	3
E	G	2
G	G	1

6) Compléter alors la table de routage du routeur G : Les chemins devront être le plus court possible !

Table de routage Du routeur G		
Destination	Routeur suivant	Distance

7) Le routeur C tombe en panne. Reconstruire la table de routage du routeur A en suivant le protocole RIP.

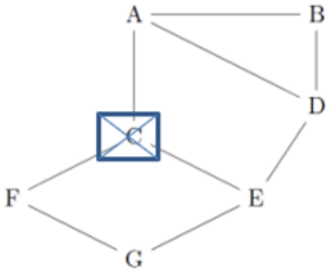


Table de routage Du routeur A		
Destination	Routeur suivant	Distance

Exercice 2 :

8) Dans le logiciel filius, ouvrez le document Routage.fls

9) Passez en mode simulation en cliquant sur 

10) Allez sur l'ordinateur dont l'adresse IP est 192.168.1.1 (en double cliquant sur cet ordinateur), puis installez le logiciel « Ligne de commande » .

11) Lancez le logiciel « Ligne de commande » , puis tapez la commande ping 192.168.2.2. Observez le chemin suivi par l'information.

Vous remarquerez que tout se passe bien, aucun paquet n'est perdu, les deux ordinateurs peuvent communiquer sans problème.

Configuration des tables de routage en routage statique :

12) Retournez en mode conception cliquant sur  .

13) Double cliquez sur chacun des 4 routeurs et décochez la case « Routage automatique » .

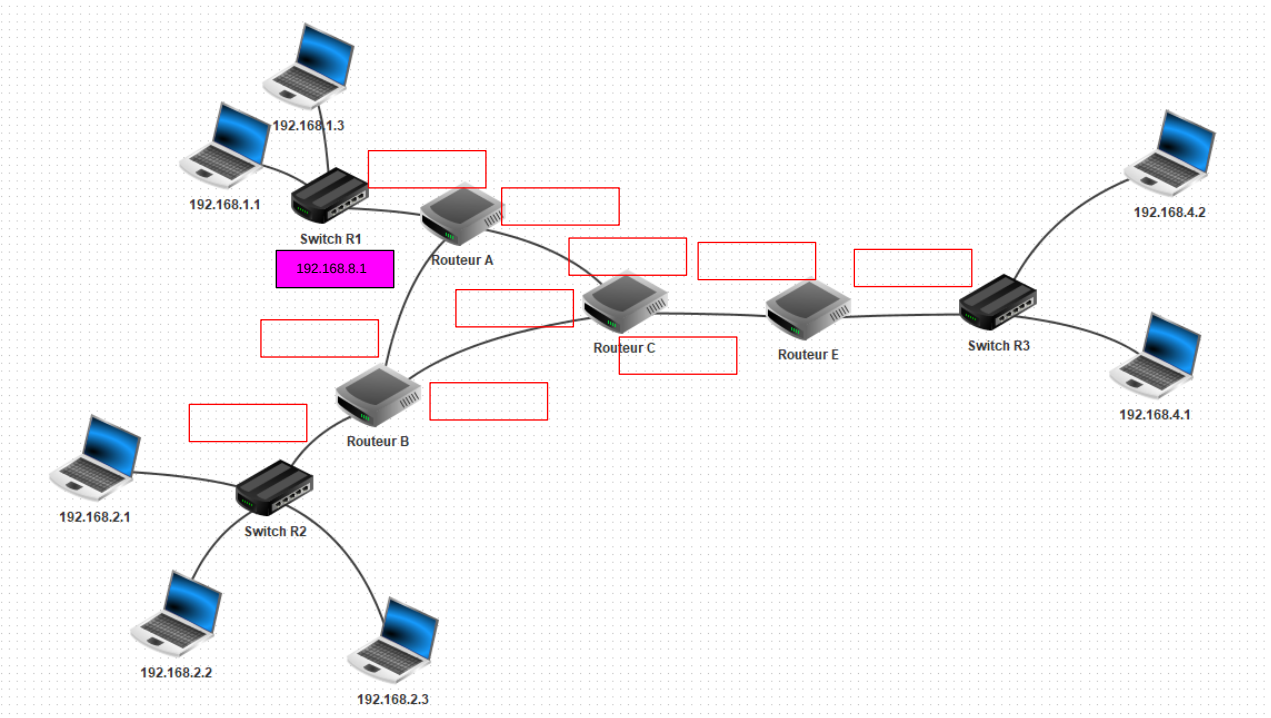
14) Vérifier que cette fois la commande ping ne fonctionne plus. Observez où s'arrêtent les paquets.

Il va falloir configurer manuellement les tables de routage des routeurs de sorte que les ordinateurs puissent à nouveau communiquer

15) Dans un premier temps, configurez la table de routage du routeur A et celle du routeur B pour que le ping entre 192.168.1.1 et 192.168.2.2 puisse fonctionner. Pour cela, il va falloir aller dans l'onglet « Table de routage » du routeur A et ajouter une ligne pour indiquer que tous les paquets à destination du réseau 192.168.2.0 (avec le masque 255.255.255.0) doivent être envoyés au routeur B (adresse 192.168.8.2) en sortant du routeur A par l'interface dont l'adresse IP est 192.168.8.1.

16) Là vous pouvez constater en réessayant le ping que les paquets arrivent bien à l'ordinateur 192.168.2.2, mais que la réponse n'arrive pas à destination. Il faudra alors configurer la table de routage du routeur B pour que le ping puisse réussir.

17) Sur le schéma ci-dessus, donner une adresse possible pour chaque interface de routeur



18) Configurez ensuite toutes les tables de routage des routeurs du réseau de sorte que tous ordinateurs puissent communiquer entre eux 2 à 2 pour cela, complétez les tables de routage et tester.

Table de routage Routeur A		
Destination	Passerelle	Interface
192.168.1.0	192.168.1.254	192.168.1.254
192.168.8.0	192.168.8.1	192.168.8.1
192.168.2.0		
192.168.9.0		
default		

Table de routage Routeur B		
Destination	Passerelle	Interface
192.168.8.0		
192.168.2.0		
192.168.1.0		
default		

Table de routage Routeur C		
Destination	Passerelle	Interface
192.168.9.0		
192.168.3.0		
192.168.7.0		
192.168.8.0		
192.168.2.0		
192.168.4.0		
192.168.1.0		

Table de routage Routeur E		
Destination	Passerelle	Interface
192.168.4.0		
192.168.3.0		
default		

III) Protocole OSPF

Nous venons de voir que le **protocole RIP** permet de configurer les tables de routage avec l**es routes les plus courtes** en **Nombre de routeurs traversés**.
Malheureusement, cette notion de distance ne garantit pas que les routes soient meilleurs en termes de débit puisque la nature des liaisons n'est pas prise en compte(*fibre optique, satellite, sans fil, etc.…*).

Nous avons vu aussi que le protocole RIP n'était pas adapté aux grands réseaux car il ignore les routes de plus de 15 sauts.

C'est pour pallier à ces défauts que le protocole OSPF (Open Shorted Path First) a été développé dans les années 90.

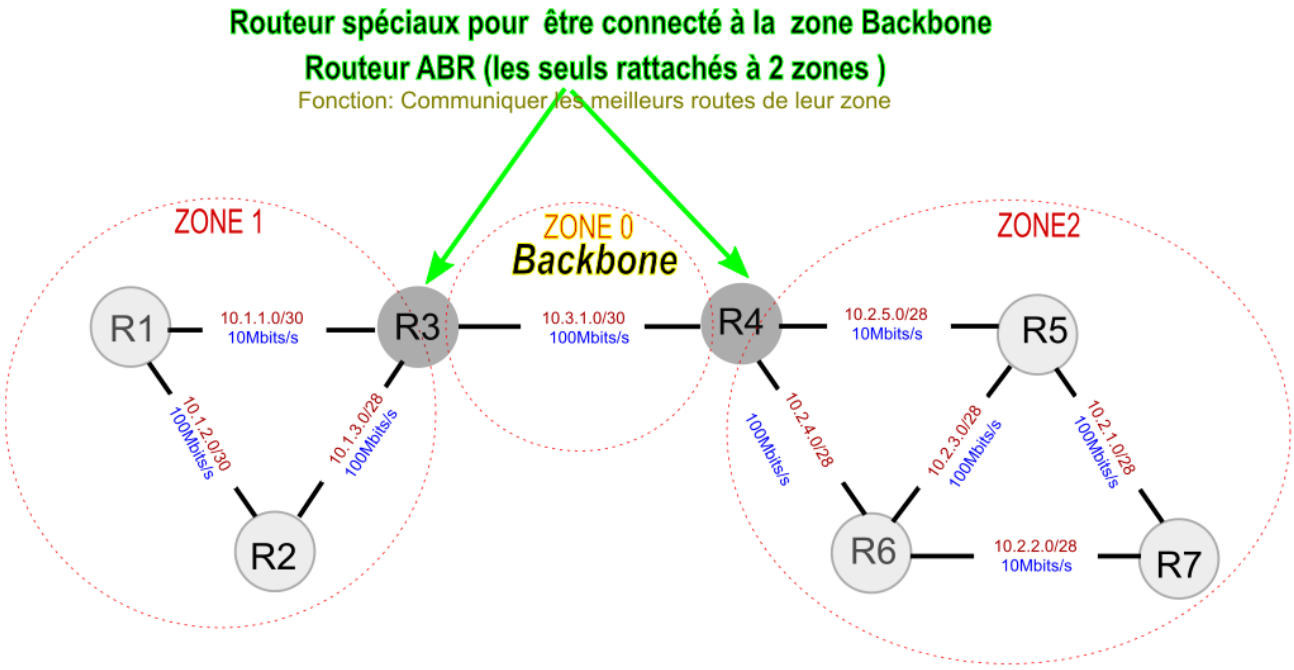
Ce protocole prend en compte la bande passante(bits/s) des liaisons de communication pour calculer les meilleures routes.

Contrairement au protocole RIP, le nombre de routeurs traversés par un paquet n'a plus d'importance dans le choix de la route. La notion de distance utilisés dans OSPF est uniquement liée au coût des liaisons qu'il faut emprunter pour relier 2 routeurs.

Pour pénaliser les liaisons lentes, le coût est fixé à $\frac{10^8}{d}$ où d est la bande passante en bit/s de la liaison.
La valeur 10^8 a été choisie pour associer un coût de 1 à une liaison FastEthernet de 100Mbits/s.
Ainsi une liaison plus lente par satellite de 20Mbits/s aura un coût de 5 et un câble Ethernet 10Mbits aura un cout de 10.

Dans un premier temps, chaque routeur, après avoir été initialisé, tente de découvrir ses voisins afin d'établir une relation de voisinage.

Dans le protocole OSPF les machines sont classées en différentes zones (ensemble de machines) et les routeurs limitent leur recherche de voisins dans la zone qui leur est affectée.



Topologie d'un réseau OSPF réparti dans 3 zones

La zone 0 obligatoire est appelée Backbone

Etapes de fonctionnement :
Phase 1 :

1. R choisi un identificateur unique, par exemple **sa plus grande adresse IP parmi celle de ses sous réseaux**.

2. Le routeur envoi des messages du type HELLO à travers tous ses interfaces réseau (paquets qui contiennent son identificateur, le numéro de la zone et la liste des voisins avec qui il a déjà établi une relation de voisinage)

3. Quand un routeur reçoit un paquet Hello de R , il vérifie si son identificateur apparait déjà dans la liste de ses voisins. Si c'est le cas, il envoi simplement un accusé de réception à R pour lui signaler qu'il est toujours actif. Sinon il répond en envoyant les informations dont il dispose sur la topologie du réseau. Idem pou R los de la réception des données.

A la suite de ses échanges, les routeurs d'une même zone ont tous la même vision topologique du réseau.

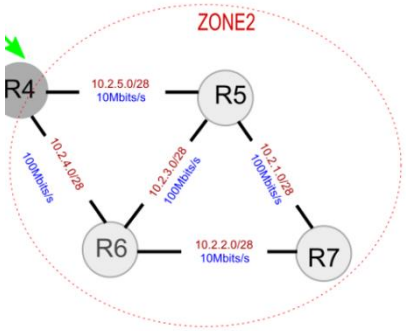
Phase 2 :
Au sein de chaque routeur on exécute un **algorithme que vous connaissez déjà permettant de déterminer la meilleur route (moindre cout, chemin le plus court) entre lui et les autres routeurs de la zone.**

19) Donner le nom de cet algorithme

En appliquant les différentes étapes décrites précédemment à l'exemple donné ci-dessus.
Lors de phase d'initialisation les routeurs reçoivent leur identificateur (la valeur choisie est la plus grande adresse IP de leurs interfaces réseaux)

20) Compléter alors le tableau suivant :

Routeur	Identificateur
R1	10.1.2.1
R2	10.1.3.2
R3	10.3.1.1
R4	
R5	
R6	
R7	



Voyons comment R5 interagit avec ses voisins et la manière dont il construit sa vision de la topologie du réseau de la zone 2.

Topologie pour R5			
LIEN	Sous-réseau	Coût	Zone
R5-R4	10.2.5.0/28	10	2
R5-R6	10.2.3.0/28	1	2
R5-R7	10.2.1.0/28	1	2

Topologie pour R4			
LIEN	Sous-réseau	Coût	Zone
R4-R5	10.2.5.0/28	10	2
R4-R6	10.2.4.0/28	1	2

Topologie pour R6			
LIEN	Sous-réseau	Coût	Zone
R6-R4	10.2.4.0/28	1	2
R6-R5	10.2.3.0/28	1	2
R6-R7	10.2.2.0/28	10	2

Topologie pour R7			
LIEN	Sous-réseau	Coût	Zone
R7-R5	10.2.1.0/28	1	2
R7-R6	10.2.2.0/28	10	2

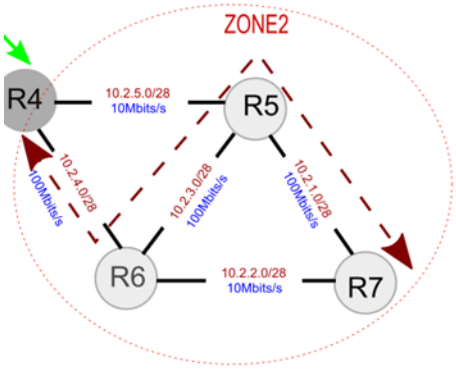
A la fin des échanges R5 contient l'ensemble de la topologie du réseau en zone 2

Topologie pour R5			
LIEN	Sous-réseau	Coût	Zone
R5-R4	10.2.5.0/28	10	2
R5-R6	10.2.3.0/28	1	2
R5-R7	10.2.1.0/28	1	2
R4-R6	10.2.4.0/28	1	2
R6-R7	10.2.2.0/28	10	2

Puisque R5 à maintenant la topologie complète de sa zone, il peut passer à la troisième étape du protocole.

Phase 3 :

On exécute donc l'algorithme pour déterminer le plus court chemin entre lui et les autres routeurs de la zone 2. On voit bien dans l'exemple que pour R5 il faut passer par R6 pour atteindre R4 plutôt que d'utiliser le sous-réseau 10.2.5.0 /28.



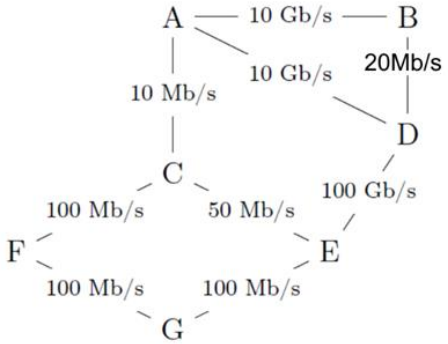
Le coût est de 2 pour ce chemin tandis qu'il est de 10 pour la liaison directe R5-R4

Ces plus courts chemins ne sont que ceux de la zone 2. Pour construire une table de routage pour l'ensemble du réseau, le routeur R5 doit savoir quel routeur de sa zone se charge de communiquer avec la Backbone.

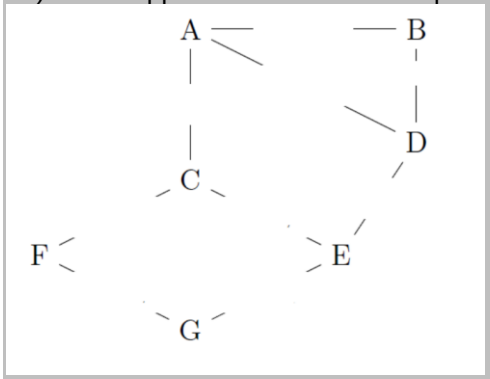
C'est dans la phase d'initialisation que le routeur R4 informe tous les routeurs de la zone 2 qu'il va jouer le rôle du routeur ABR.
Ainsi, R4 communique à toutes les autres zones (via la zone 0) les plus courts chemins entre lui et les autres routeurs de la zone2 et inversement il reçoit les plus courts chemins de la zone 1 qu'il communique à R5.
De cette manière, R5 apprend par exemple qu'il existe une route avec un coût=3 pour atteindre R1en passant par R4.
En intégrant à sa table de routage les informations sur les meilleures routes de la zone1, il obtient la table suivante.

Table de routage de R5			
Destination	Passerelle	Liaison	Coût
10.2.1.0/28		fasteth1	1
10.2.3.0/28		fasteth0	1
10.2.4.0/28	10.2.3.1(R6)	fasteth0	2
10.3.1.0/28	10.2.3.1	fasteth0	3
10.1.2.0/28	10.2.3.1	fasteth0	5
10.1.3.0/28	10.2.3.1	fasteth0	4

Soit un graphe représentant les liaisons entre différents routeurs selon le protocole OSPF.



21) Faire apparaitre le cout de chaque liaison.



22) Le routeur A doit transmettre un message au routeur G, en empruntant le chemin dont la somme des coûts est le plus petit possible. Par la méthode Dijkstra compléter le tableau suivant et en déduire le chemin le plus court.

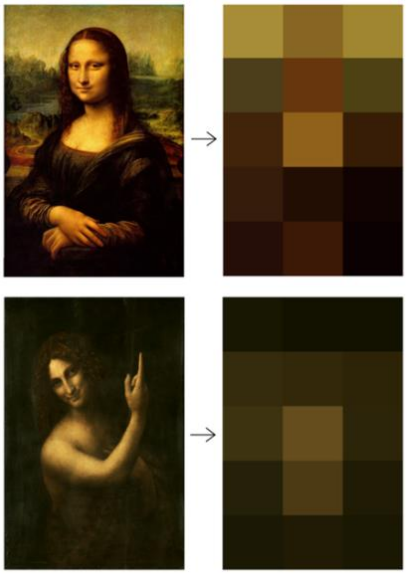
A	B	C	D	E	F	G	Choix

IV) LE WEB SÉCURISÉ

La protection des informations et des communications est un besoin très ancien.

C'est la fonction principale de la cryptographie

- le plus ancien document chiffré est une recette secrète de poterie qui date du XVIe siècle av. J.-C., qui a été découverte dans l'actuelle Irak.
- Chiffre de César lors de l'antiquité



a. Les fonctions de hachage

- Une fonction de hachage calcule une empreinte à partir d'un fichier ou plus généralement d'une donnée.
- Une empreinte a une taille fixe de quelques octets.

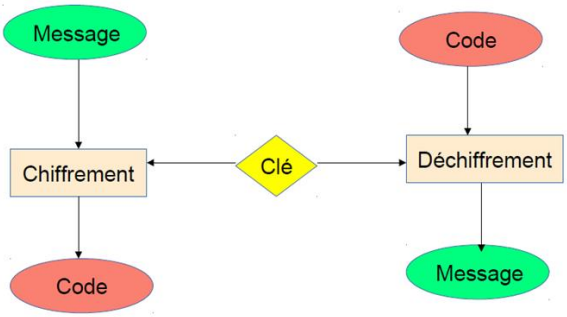
Exemple pédagogique du principe des fonctions de hachage appliqué à des images (voir ci-contre): on considère ici une fonction de hachage consistant à convertir une image haute résolution en une empreinte très basse résolution.

L'empreinte est beaucoup plus légère en mémoire. Elle perd une grande partie de l'information mais elle reste suffisante pour distinguer rapidement les deux images.

b. Chiffrement symétrique :

Définition :

Un algorithme de chiffrement symétrique utilise la même clé pour chiffrer et déchiffrer.



Méthode de César:

On choisit un décalage de 4 . Ce qui veut dire que A est remplacé par E, B par F, … , Z est remplacé par D. Pour les chiffres 0 est remplacé par 4, 1 par 5, … , 9 est remplacé par 3. Les espaces sont supprimés.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D

23) Crypter le message suivant : "ton code est 1234"
24) Décrypter le message suivant : "wepyxfsf"

Chiffrement par flots

Le chiffrement par flots est un chiffrement bits à bits.

Exemple : le chiffrement de Vernam :

On utilise une clé K de la taille du message M

- Pour chiffrer, on calcule C = K xor M
- Pour déchiffrer, on calcule M = K xor C

Le chiffrement est très sûr, mais nécessite une clé très grande (de la taille du message).

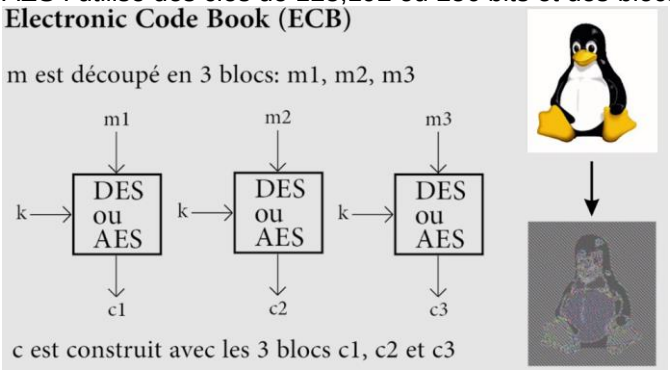
Chiffrement par blocs

Le chiffrement s'applique à des blocs de même taille. Nécessiter de découper le message en blocs de taille identique

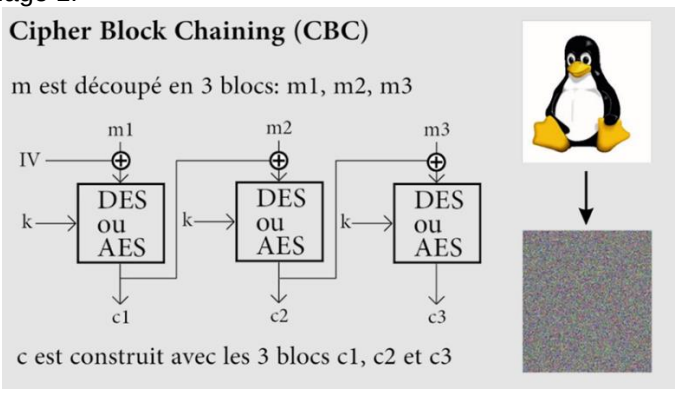
Exemples de chiffrement par bloc :

- DES : utilise une clé de 64 bits donc 2^{64} possibilités et des blocs de 64 bits du message à crypter.

- AES : utilise des clés de 128,192 ou 256 bits et des blocs de 128 bits.



- Chaque bloc est crypté avec la clé k n'est pas assez sécurisé, on voit bien qu'on retrouve les formes de l'image 1.



- Ici chaque bloc est d'abord un XOR avec le bloc précédent puis crypté avec la clé k. Le premier bloc est un XOR avec un vecteur d'initialisation IV puis crypté avec la clé k.

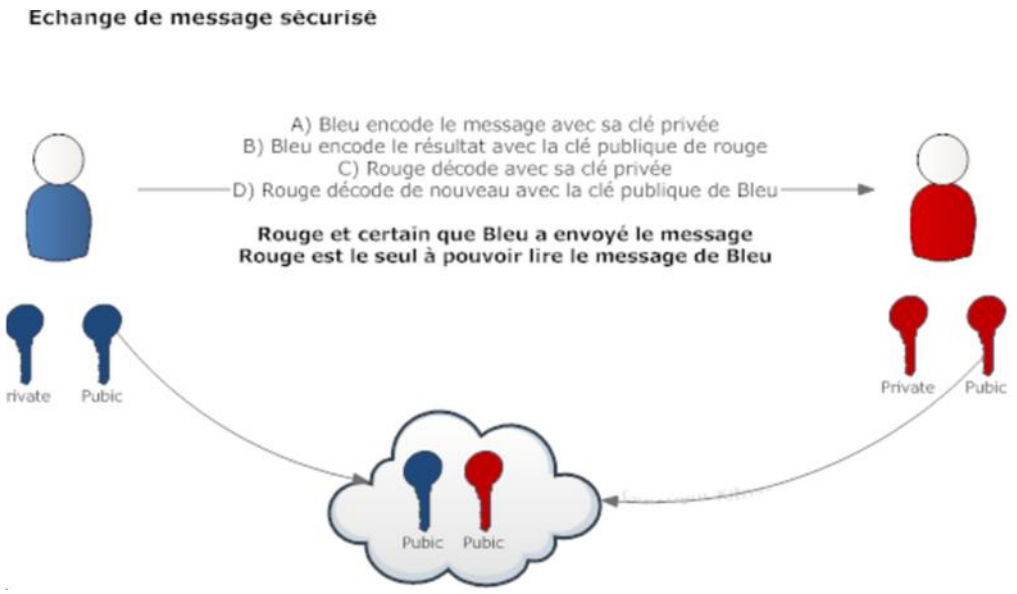
c. Chiffrement asymétrique

Le principe du chiffrement asymétrique est d'avoir 2 clés (que l'on fabrique soi-même) :

- Quand on encode avec la première clé, on peut décoder avec la 2e clé.
- Quand on encode avec la 2e clé, on peut décoder avec la 1ère clé.

Par convention, on appelle une des 2 clés la clé privée et l'autre la clé publique.
La clé privée n'est jamais transmise à personne.
La clé publique est en revanche diffusée publiquement sans problème.
Dans une transmission simple, ce système permet 2 choses essentielles :

- Assurer la confidentialité d'un message transmis : personne ne peut le lire sauf le destinataire
- Signer un message : le destinataire est certain que le message vient bien du bon interlocuteur

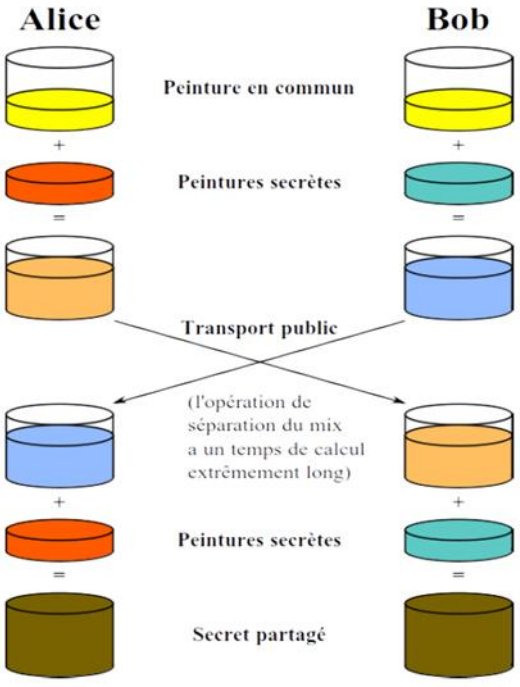


Pour que cela soit possible, il y a forcément un lien entre la clé privée et la publique.

Échange de clés Diffie-Hellman

En **cryptographie**, l'échange de clés Diffie-Hellman, du nom de ses auteurs **Whitfield Diffie** et **Martin Hellman**, est une méthode, publiée en 1976, par laquelle deux agents, nommés par convention **Alice et Bob**, peuvent se mettre d'accord sur un nombre (qu'ils peuvent utiliser comme **clé** pour chiffrer la conversation suivante) sans qu'un troisième agent appelé **Ève** puisse découvrir le nombre, même en ayant écouté tous leurs échanges. Cette idée valut en 2015 aux deux auteurs le prix Turing.

A la fin du protocole, Alice et Bob possèdent la même couleur brune, qui représente la couleur secrète partagée. En supposant qu'il est difficile pour Ève d'extraire les couleurs utilisées pour obtenir les couleurs publiques orange et bleue, Ève ne connaît pas la couleur brune finale.



Concrètement, on définit la couleur jaune par deux valeurs numériques (clé de chiffrement publique) :
p=470 300
g=846 081

Ensuite Alice choisit un nombre qu'elle est seule à connaitre (clé de chiffrement privée) :
a=186 397

Bob choisit un nombre qu'il est le seul à connaitre (clé de chiffrement privée) :
b=753 484

25) Calculer :
$A=g^a$ modulo(p)
$B=g^b$ modulo(p)

Ce sont ces deux secrets qui vont transiter :

26) Calculer :
$K=B^a$ modulo(p)
$K=A^b$ modulo(p)

Conclusion

Ce protocole est vulnérable à « l'attaque de l'homme du milieu », qui implique un attaquant capable de lire et de modifier tous les messages échangés entre Alice et Bob.

La **parade classique** à cette attaque consiste **à signer les échanges** de valeurs à l'aide d'une paire de **clés asymétriques certifiées** par une tierce partie fiable, ou dont les moitiés publiques ont été échangées auparavant par les deux participants.

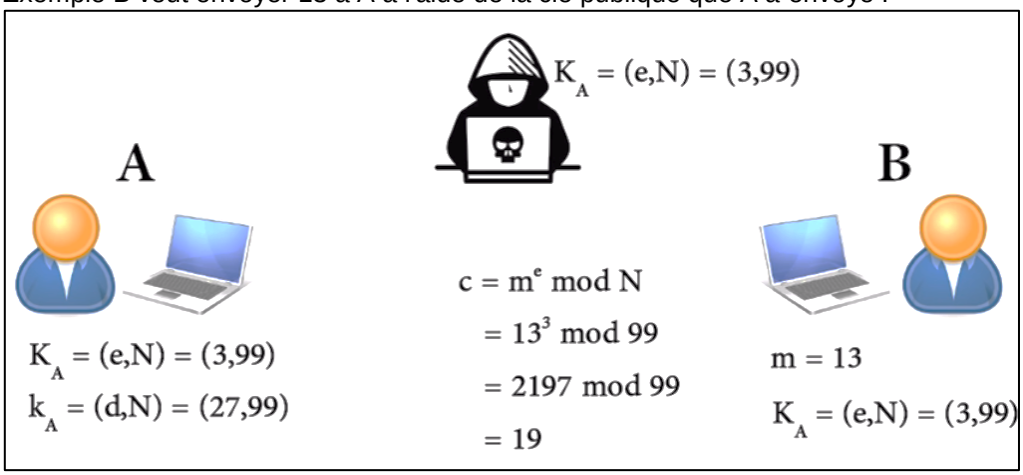
Alice peut ainsi être assurée que la clé qu'elle reçoit provient effectivement de Bob, et réciproquement pour Bob.

Exemple : système RSA : créé par Ronald Rivest, Adi Shamir et Leonard Adleman

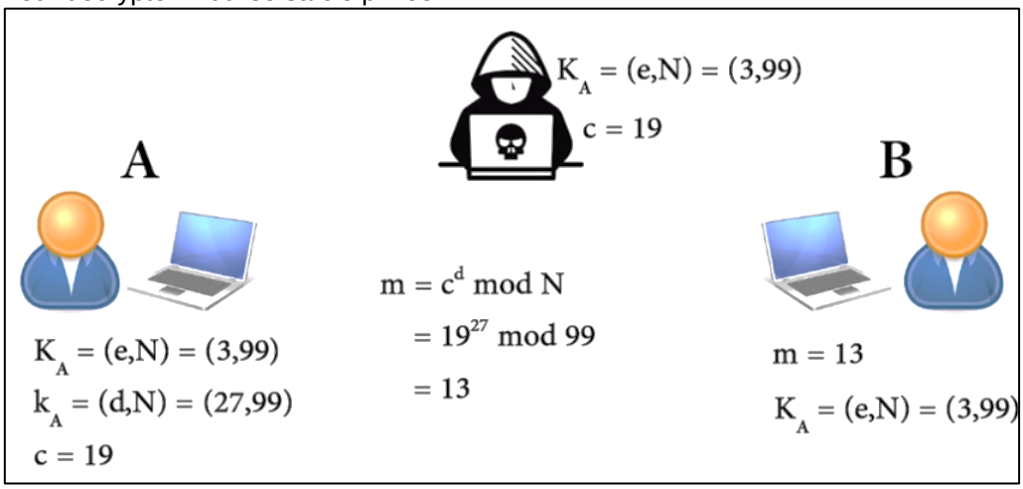
Ce système est basé sur le principe de la factorisation. Ex: 35=x*y il n'existe pas d'algorithme rapide qui permet de trouver les deux nombres premiers x=5 et y=7. Plus le résultat est grand, plus x et y sont long à trouver.(sauf avec un ordinateur quantique mais très peu répandu pour le moment)

- On commence par choisir deux nombres premiers : p=11 et q=9 par exemple.
- On calcule N=p*q=11*9 dans notre exemple
- On définit une fonction phi(N)=(p-1)(q-1)=10*8=80 pour l'exemple.
- On choisit e tel que pgcd(e,phi(N))=1 (pgdc:plus grand diviseur commun) ici dans notre exemple on peut prendre e=3 (en général on prend un nombre premier).
- On calcule $d=e^{-1}$ modulo(phi(N)) équivaut à dire d*e modulo(phi(N))=1 c'est-à-dire d*3 modulo(80)=1, le plus petit d possible est 27 car 3*27=81 et 81 modulo 80=1.
- Donc la clé public de A sera K_A=(e,N) et la clé privée k_A=(d,N)

Exemple B veut envoyer 13 à A à l'aide de la clé publique que A a envoyé :



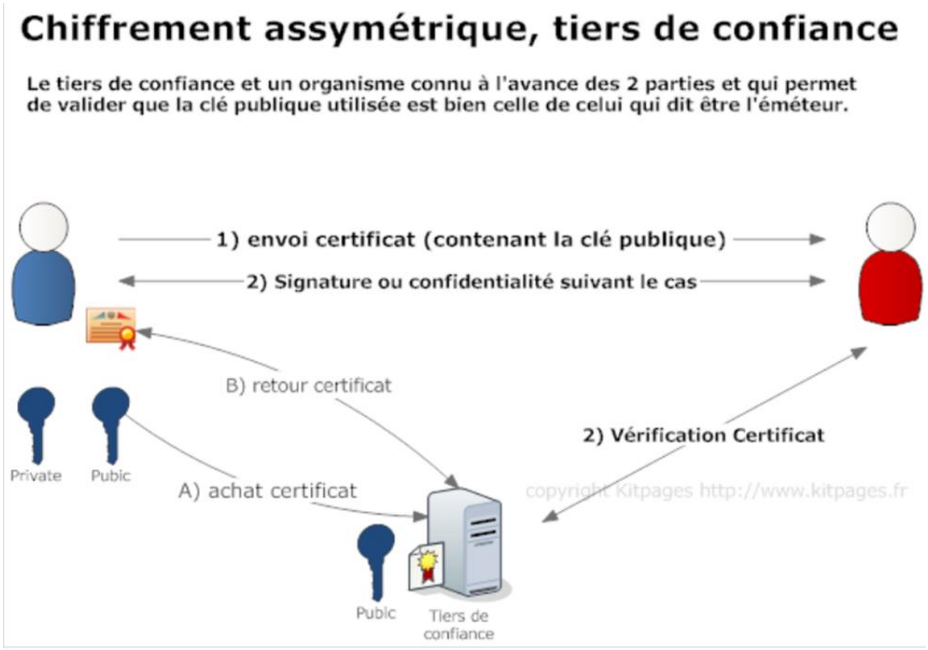
Pour décrypter A utilise sa clé privée



Il existe d'autre système comme ElGamal qui est basé sur le même principe mais au lieu de la factorisation, il est basé sur le logarithme discret.

d. Tiers de confiance

Le problème du chiffrement par clé publique est qu'il faut parfois vérifier que la clé publique est bien celle de celui qu'on croit être l'émetteur.
Pour cela on utilise un "tiers de confiance"

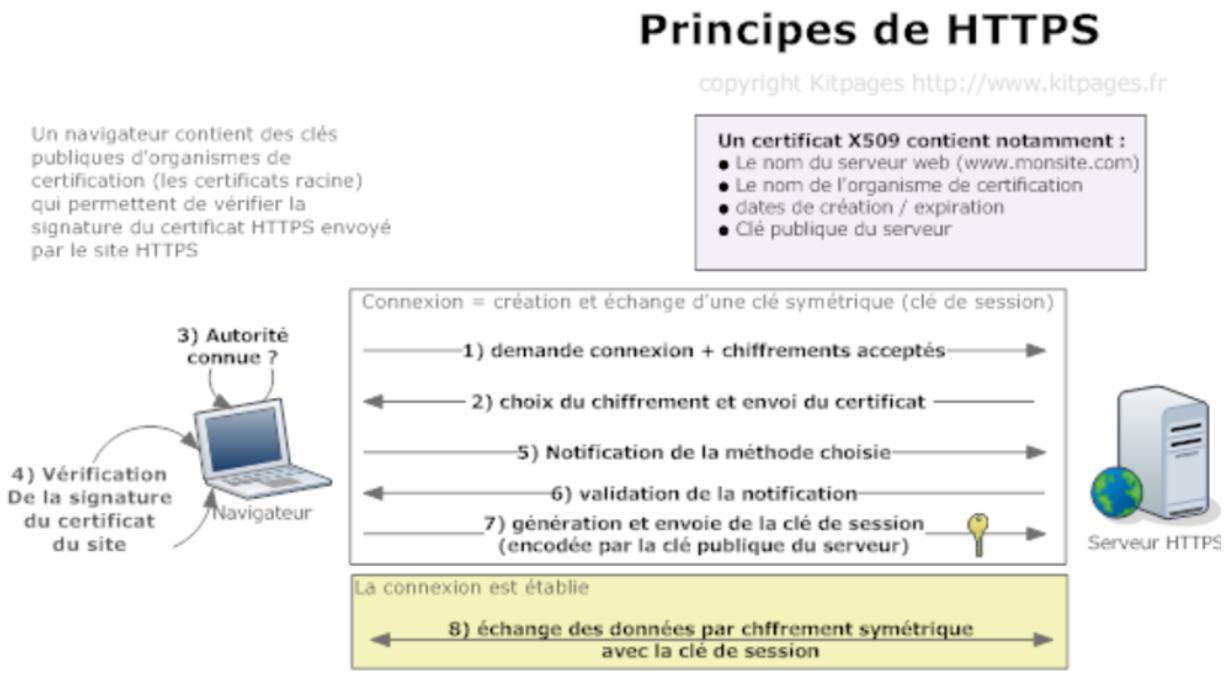


e. Conclusion et HTTPS

Il faut bien comprendre qu'il y a 2 étapes de chiffrement :

- HTTPS utilise un chiffrement asymétrique pour établir une connexion et échanger une clé symétrique.
- Ensuite les échanges utilisent un chiffrement symétrique

L'intérêt de ce système est que le chiffrement symétrique est beaucoup moins gourmand en ressources. Une fois la connexion établie, l'HTTPS consomme des ressources CPU assez raisonnables. (notons que l'établissement de la connexion est très consommatrice en ressources).



f. Le certificat

Un certificat est délivré par une entreprise comme Thawte ou Verisign. Son rôle est de garantir que la clé publique envoyée par le serveur HTTPS correspond bien au site demandé.

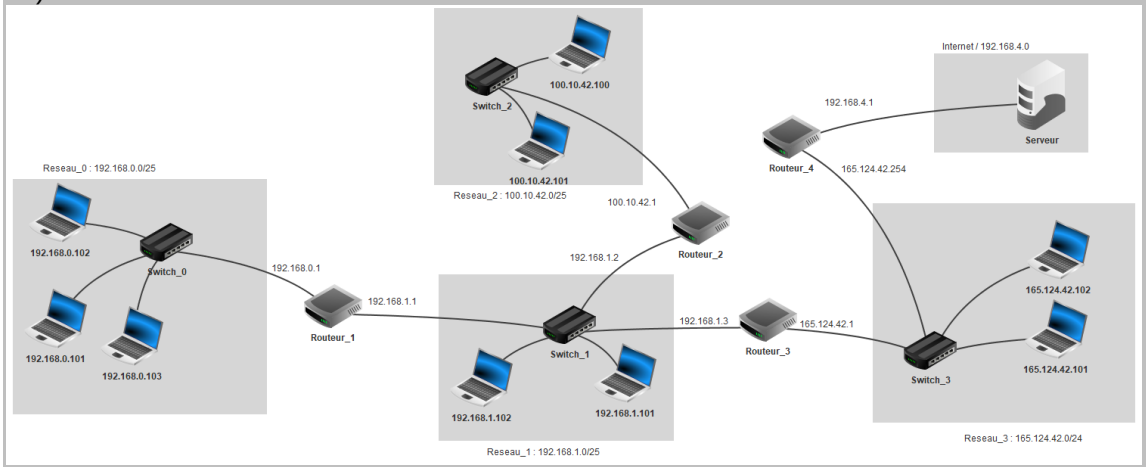
Le navigateur dispose d'une ribambelle de certificats racine correspondant aux organismes de certifications. Ces certificats racine permettent de valider le certificat du serveur HTTPS. Une fois le certificat validé, les échanges peuvent commencer.

V) Exercices :

27) Trois machines ont respectivement pour adresses IP:
90.8.220.5
90.8.220.20

90.8.220.37.
Est-ce que ces machines appartiennent toutes les 3 au même réseau : 90.8.220.0/27?
Si non combien de routeurs sont nécessaires pour faire communiquer ces machines (proposer des adresses pour les interfaces du ou des routeurs)?

28) Soit le réseau suivant :



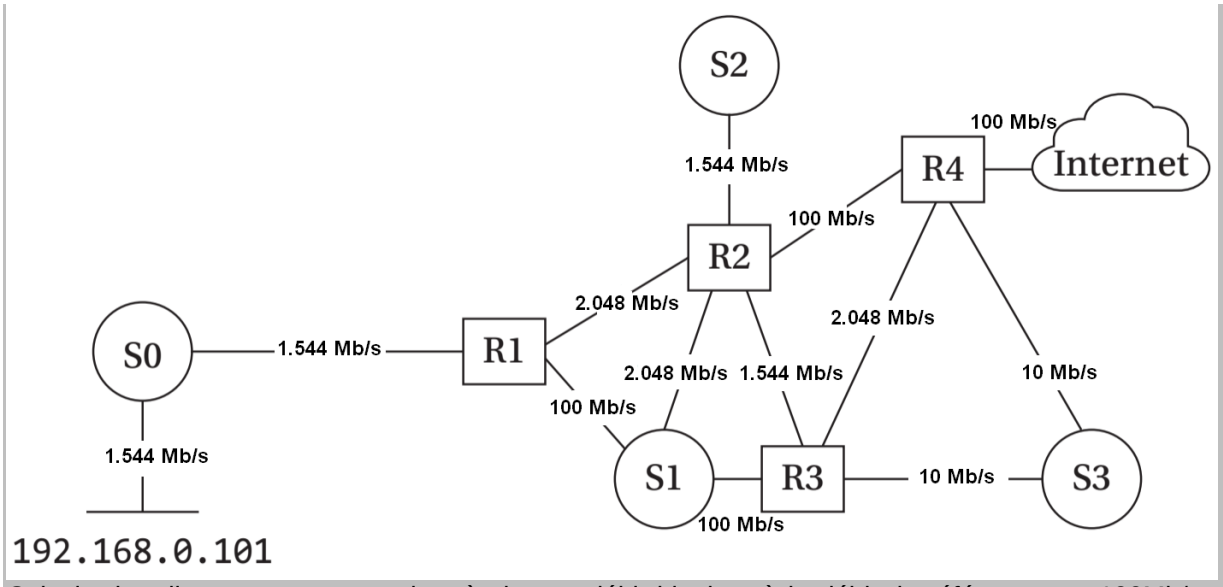
Donner les tables de routage statique du routeur 1, 2 et 3 :

Routeur 1		
Destination	Passerelle	Distance
Reseau_0:192.168.0.0/25		
Reseau_1:192.168.1.0/25		
Reseau_2:100.10.42.0/25		
Reseau_3:165.124.42.0/24		
Reseau_4:192.168.4.0		

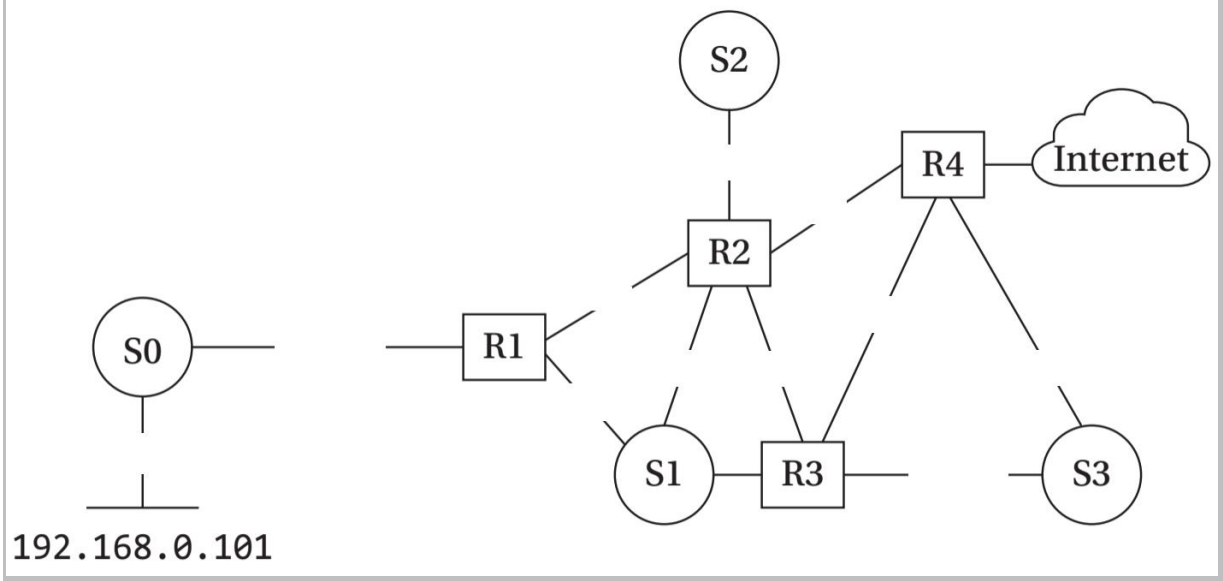
Routeur 2		
Destination	Passerelle	Distance
Reseau_1:192.168.1.0/25		
Reseau_2:100.10.42.0/25		
Reseau_0:192.168.0.0/25		
Reseau_3:165.124.42.0/24		
Reseau_4:192.168.4.0		

Routeur 3		
Destination	Passerelle	Distance
Reseau_1:192.168.1.0/25		
Reseau_3:165.124.42.0/24		
Reseau_4:192.168.4.0		
Reseau_0:192.168.0.0/25		
Reseau_2:100.10.42.0/25		

Soit la représentation du réseau suivante modifiée en rajoutant des connexions entre serveurs :
Routeur 1 et 2
Routeur 2 et 3
Routeur 2 et 4
Routeur 3 et 4



Calculer les distances correspondant à chaque débit binaire où le débit de référence est 100Mb/s et compléter le graphe avec ces valeurs :



Trouver la distance et le chemin le plus court pour aller de l'ordinateur 192.168.0.101 que l'on appellera D vers internet.

D	S0	S1	S2	S3	R1	R2	R3	R4	Internet	Choix

Chiffrement de Vigenère :

Blaise de Vigenère (1523-1596) fut un cryptographe français du XVIe siècle. Il fut à l'origine du premier chiffrement à clé difficile à casser.

Le chiffrement de Vigenère est une variante du chiffrement de César. Au lieu d'utiliser une clé unique pour chiffre chaque lettre d'un message en procédant à un décalage, on utilise une clé variable. On se sert **d'une phrase clé** dont chaque lettre va déterminer une clé : **la première lettre du message** est chiffrée à l'aide de la **première lettre de la phrase**, la deuxième lettre du message est chiffrée à l'aide de la deuxième lettre de la phrase, et ainsi de suite. Si la phrase clé est plus courte que le message, on recommande le parcours au début.

Pour simplifier, le message est écrit en lettres capitales et on chiffre uniquement les lettres. Les lettres sont numérotées suivant leur place dans l'alphabet de 0 à 25. Le numéro d'une lettre de la phrase donne la valeur du décalage pour la lettre correspondante du message.
Par exemple, si la lettre de la phrase clé est un E, alors le décalage est de 4 places vers la droite. Donc, un R est chiffré par un V.

Si la lettre de la phrase clé est un G, alors le décalage est de 6 places vers la droite. Donc un V est chiffré par un B.

- La fonction ord renvoie le code ASCII d'un caractère. Par exemple, ord('E') vaut 69 et 69-65=4, donc le décalage est de 4 places (les codes ASCII vont de 65 à 90 pour les majuscules).
- La fonction chr renvoie le caractère correspondant au code donné en paramètre. Par exemple, chr(69)='E'.

29) Le chiffrement de Vigenère est-il symétrique ou asymétrique ?
30) Ecrire une fonction lettres qui prend en paramètre une phrase (str) et qui renvoie la phrase sans les espaces (on ne garde que les lettres).
31) Ecrire une fonction vigenere qui prend en paramètres deux chaînes de caractères, le message à chiffrer et la phrase clé, et renvoie le message chiffré.
32) Il est possible d'écrire une fonction dechiffre(texte,phrase) permettant de décoder le message chiffré par la fonction vigenere. Quelles seraient les deux différences avec la fonction de la question 20 ?
33) Déchiffrer le message suivant à la main :
    message_crypte = "EIVB TZK DI QVLSCNM F GYCSRVWJ VYKW T SLTHAJXVVK"
    phrase_cle = "CETTE PHRASE SERT A CHIFFRER UN MESSAGE"
34) Vérifier avec votre programme et faire une capture.