Sécurité des réseaux

Introduction :

Notre usage d'Internet repose sur le bon fonctionnement de ce réseau décentralisé, et nécessite de sécuriser certaines informations qui y circulent. Nous allons nous intéresser dans une première partie à la manière dont nos données sont acheminées sur les réseaux. Nous aborderons dans une deuxième partie les grands principes de chiffrement et, dans une troisième, leur application à la sécurisation d'une communication web.

1 | Routage

Définition

Routeur :

Les routeurs sont chargés d'acheminer les paquets (paquet = entité de transmission) entre les réseaux. Ils s'appuient sur des tables de routage qui répertorient les différentes routes possibles dans leur voisinage réseau.

- Ces tables de routage peuvent être composées **manuellement** : c'est le **routage statique**. Il implique d'effectuer des modifications à chaque évolution ou incident sur le réseau. Cela nécessite une intervention humaine sur chaque routeur concerné pour mettre à jour la table de routage correspondante.
- Des protocoles ont été définis pour permettre de réaliser ces mises à jour de manière **automatique** : c'est le **routage dynamique**. Il permet aux routeurs de communiquer entre eux afin d'actualiser leurs tables de routage chaque fois que nécessaire, en cas d'incident, d'indisponibilité ou de modification du réseau,

pour continuer à sélectionner les meilleures routes possibles parmi celles restant accessibles.

Ce calcul des routes consiste à déterminer **le plus court chemin** que peut emprunter un paquet, il s'agit donc d'une mesure de distance.
La démarche algorithmique est celle de la recherche du plus court chemin sur un graphe, le critère de **pondération** pouvant varier d'un algorithme à l'autre. Cette pondération consiste à donner plus ou moins de poids à une arête du graphe selon le critère choisi, par exemple le débit.

Dans le cadre de ce cours nous nous intéressons aux seuls protocoles de **routage interne** (IGP pour *Interior Gateway Protocol*), applicables au sein de systèmes autonomes, correspondant en général à une même organisation. Nous présenterons donc, et comparerons, les principales caractéristiques des protocoles de routage RIP et OSPF.
Le routage externe, que nous n'abordons pas ici, est quant à lui utilisé entre des systèmes autonomes.

a. Protocole RIP

RIP signifie *Routing Information Protocol*, soit « protocole d'information de routage ». Il fait partie des plus anciens protocoles de routage.

À retenir

> C'est un protocole dit à **vecteur de distance**. Il retient les routes minimisant le nombre de routeurs qu'un paquet IP doit traverser pour atteindre sa destination.

Le protocole RIP met en œuvre l'algorithme de Bellman-Ford. Sa métrique est le **nombre de sauts à effectuer entre la source et la destination**, avec un maximum fixé à quinze sauts.

Astuce

> Au-delà de quinze sauts, la destination est considérée comme inaccessible.

Tous les routeurs sont impliqués, sans spécialisation ni hiérarchisation. La propagation des meilleures routes s'effectue par le biais de **communications régulières entre les routeurs**, cette communication périodique restant indépendante d'éventuelles modifications sur le réseau.

b. Protocole OSPF

OSPF signifie *Open Shortest Path First Protocol*, soit « protocole ouvert du plus court chemin ».

Initialement développé dans les années 1980 sous forme propriétaire par Digital Equipment Corporation, il est ensuite devenu un standard ouvert à partir de 1989.

À retenir

> OSPF est un protocole à **état de liens**.

Il met en œuvre l'algorithme du plus court chemin de Dijkstra. Sa métrique est le **coût des liens**. Ce coût des liens est basé sur le débit disponible entre les routeurs pour favoriser les routes disposant d'une bande passante plus importante.

C'est un routage avec une composante hiérarchique permettant le découpage par zones, dans lequel certains routeurs sont spécialisés. Les routeurs échangent avec leurs voisins des informations sur la topologie du réseau.

Astuce

> La communication n'est pas systématique, elle a lieu uniquement lorsqu'un changement survient, et porte uniquement sur la modification survenue.

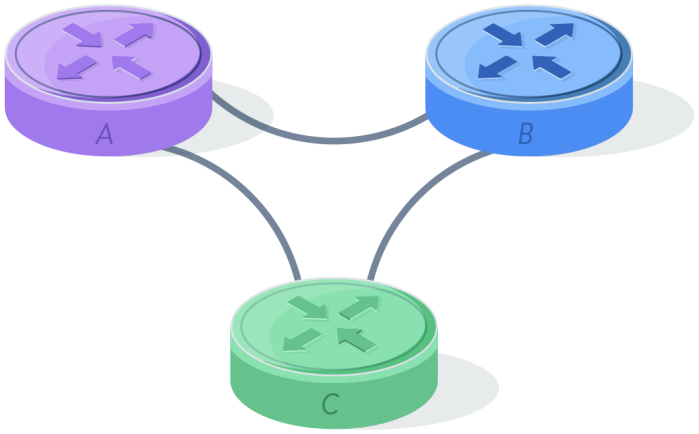
c. Comparaison des protocoles

À retenir

> La principale différence entre les deux protocoles porte sur la **métrique**. Le coût d'une route est évalué très différemment d'un protocole à l'autre :
>
> - RIP cherche à **minimiser le nombre de sauts**.
> - OSPF prend en compte la **bande passante entre les routeurs**.

Illustrons cette différence avec une portion de réseau comportant trois routeurs A, B, et C interconnectés.

Interconnexion de trois routeurs



Dans cette configuration, il existe deux routes possibles pour aller de A à B :

- la route $A \to B$ (route directe) ;
- la route $A \to C \to B$ (route indirecte).

1. Calculons le coût des routes selon RIP, dont la métrique est le nombre de sauts.

Route	Coût

route $A \to B$	1 saut
route $A \to C \to B$	1 saut + 1 saut (2 sauts)

→ Le protocole RIP minimise le nombre de sauts et considère de ce fait la route directe de A à B comme étant la meilleure.

Toutefois le protocole RIP ne prend pas en compte d'éventuelles différences de bande passante au niveau des liaisons entre les trois routeurs ; alors que le protocole OSPF détermine justement le coût des liens sur la base des débits.

2. Considérons que les routeurs A et B soient reliés entre eux en Ethernet ($10\,\text{Mbps}$), et que les autres liaisons soient en FastEthernet ($100\,\text{Mbps}$). OSPF calcule le coût des liens en divisant un débit maximum théorique de $100\,\text{Mbps}$ (par défaut, ajustable si nécessaire pour des débits supérieurs) par la bande passante du lien pour chaque segment.

Avec cette base de calcul :

- les segments en Ethernet ont un coût de $10\left(\frac{100\,\text{Mbps}}{10\,\text{Mbps}}\right)$.
- les segments en FastEthernet ont un coût de $1\left(\frac{100\,\text{Mbps}}{100\,\text{Mbps}}\right)$.

Muni de ces valeurs, nous calculons le coût des routes selon OSPF.

Route	Coût
route $A \to B$	10
route $A \to C \to B$	$1+1\ (2)$

→ C'est la route indirecte $A \rightarrow C \rightarrow B$, dotée d'une bien meilleure bande passante, qui est privilégiée par OSPF, malgré un nombre supérieur de routeurs à traverser qui avait conduit RIP à ne pas l'utiliser.

Parmi les autres différences notables entre les deux protocoles, on retiendra que :

- RIP nécessite davantage de trafic du fait de la transmission périodique des tables de routages complètes, tandis qu'OSPF ne transmet que les incréments et uniquement en cas de survenue d'une modification ;
- la vitesse de convergence (c'est-à-dire le temps nécessaire pour l'actualisation de l'ensemble des routeurs concernés) est plus rapide avec OSPF qu'avec RIP ;
- la meilleure efficacité de l'algorithme d'OSPF est au prix d'une puissance de calcul supérieure à celle requise pour RIP ;
- RIP est plus simple à configurer mais peu adapté à des topologies complexes, contrairement à OSPF.

Au-delà des différences techniques d'un protocole à l'autre, le routage vise à transmettre de manière efficiente les flux réseaux en choisissant à tout moment les meilleures routes possibles. La capacité de résilience offerte par le routage dynamique permet de composer sans intervention humaine, avec des incidents ou des modifications.

Mais la transmission rapide des informations n'est pas le seul aspect à prendre en compte, en particulier sur des réseaux ouverts (réseaux Wifi où les échanges ne sont pax cryptés) : la **sécurité des informations** doit également pouvoir être garantie dans certains cas.
Nous allons maintenant nous intéresser aux méthodes de chiffrement qui peuvent être mises en œuvre pour sécuriser les flux réseaux.

2 | Principes de chiffrement

La **cryptologie** est une discipline aussi ancienne que le besoin de

secret dans les correspondances. Elle est pratiquée depuis l'Antiquité et s'est perfectionnée au cours de l'histoire.

Astuce

> La cryptologie moderne s'appuie depuis plusieurs décennies sur la puissance de calcul toujours plus importante des outils informatiques.

Cette discipline se décompose en deux branches complémentaires :

1. la **cryptographie** qui vise à chiffrer les messages pour les rendre incompréhensibles par des personnes non autorisées ;
2. la **cryptanalyse** qui cherche à déchiffrer les messages rendus incompréhensibles par la cryptographie.

Exemple

Pendant la Seconde Guerre mondiale, la machine conçue par Alan Turing était destinée à la cryptanalyse. Le mathématicien britannique et son équipe de Bletchley Park travaillaient sur des messages allemands, chiffrés avec la machine à coder Enigma, interceptés par les services secrets britanniques.

a. Bases de la cryptographie

À retenir

> La cryptographie consiste à appliquer un **algorithme de chiffrement**, à un message en **texte clair**, afin de le transformer en un **message chiffré**.

- Le message en clair est le message de départ, lisible et compréhensible par un humain, tel que son auteur l'a rédigé.
- Le message chiffré est le résultat incompréhensible de la transformation du message en clair par l'algorithme de chiffrement.

→ Le déchiffrement est le résultat de la transformation d'un message chiffré en message clair.

Pour chiffrer ou pour déchiffrer un message, un algorithme utilise un paramètre bien spécifique ; une **clé**.

Attention

> Il ne faut pas confondre déchiffrement et décryptage.
> Les abus de langage sont fréquents, notamment pour le chiffrement qui est régulièrement désigné par le terme « cryptage » (un faux anglicisme).

À retenir

> - En cryptographie, le déchiffrement consiste à décoder à l'aide de sa clé un message préalablement chiffré.
> - En cryptanalyse, le décryptage consiste à « casser le code », c'est-à-dire à découvrir un chiffrement par cryptanalyse, sans en connaître la clé de chiffrement.

Il existe de nombreux algorithmes de chiffrement, que l'on peut regrouper en deux grandes familles de systèmes cryptographiques :

- la **cryptographie symétrique**, également appelée cryptographie à clé secrète ou à clé partagée ;
- la **cryptographie asymétrique**, également appelée cryptographie à clé publique.

b. Cryptographie à clé secrète

Les systèmes cryptographiques à clé secrète sont symétriques, c'est-à-dire que les fonctions mathématiques qui composent les algorithmes de chiffrement sont réversibles. Il suffit de les appliquer en sens inverse avec la clé pour pouvoir déchiffrer un message préalablement chiffré.

À retenir

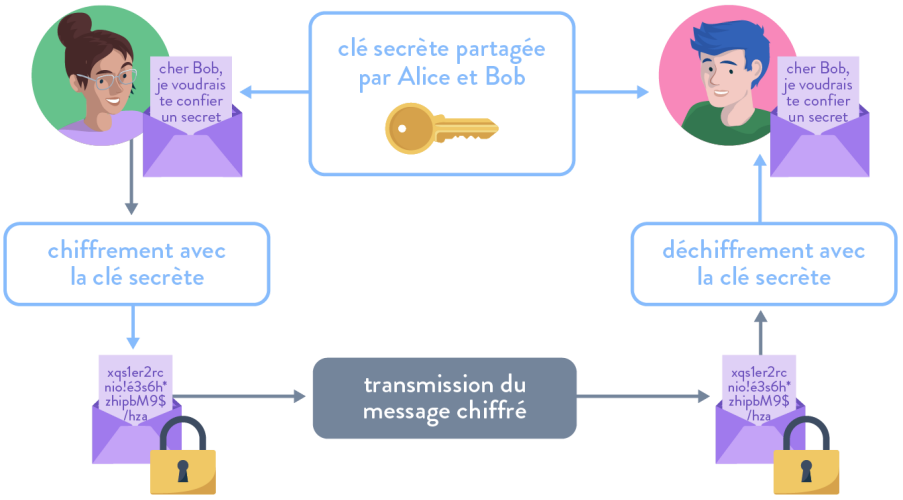
> La cryptographie à clé secrète repose sur l'utilisation d'une **même clé** pour chiffrer et pour déchiffrer un message. Cette clé est donc nécessairement partagée entre l'émetteur et le destinataire.

En cryptologie les correspondants A et B sont usuellement appelés Alice et Bob (Bob étant parfois remplacé par son cousin français Bernard). Un tiers E, usuellement appelé Ève, voudrait connaître la teneur des échanges entre Alice et Bob.
Voyons quelles sont les actions menées par Alice, Bob et Ève dans le cadre de la cryptographie à clé secrète.

Exemple

> Alice et Bob s'accordent sur une clé secrète qu'ils utiliseront pour protéger leurs échanges.
>
> Alice utilise la clé secrète pour chiffrer le message.
> Alice transmet le message chiffré à Bob.
>
> Bob utilise la clé secrète pour déchiffrer le message.
> Bob peut lire le message déchiffré.

> Si Ève voit passer le message chiffré au cours de sa transmission entre Alice et Bob, elle sera incapable d'en comprendre la teneur car elle ne connaît pas la clé secrète nécessaire au déchiffrement.



Astuce

> Il n'est pas nécessaire que l'algorithme utilisé soit secret, mais il faut que la clé le soit.

Avec un algorithme robuste et une clé suffisamment longue, les attaques par force brute seront déraisonnablement longues en raison du nombre très important de possibilités à tester.

Il existe de nombreux algorithmes de chiffrement symétrique. Le plus connu est l'algorithme AES.

Algorithme AES

L'algorithme Rijndael a été conçu par deux cryptologues belges, Johan Daemen et Vincent Rijmen, à la fin des années 1990. Il a servi de base en 2000 à une nouvelle norme de chiffrement adoptée par le gouvernement américain, connue sous l'appellation **AES** (pour *Advanced Encryption Standard*, soit « norme de chiffrement avancé »).

Cet algorithme opère différentes transpositions et substitutions en traitant les données par blocs de 128 bits. Il fonctionne avec des clés de 128, 192 ou 256 bits.
Il remplace, au début des années 2000, l'algorithme DES (*Data Encryption Standard*, soit « norme de chiffrement de données »). Le DES utilisait une clé de 56 bits, suffisante lors de son invention en 1975, mais insuffisante en 2000 pour faire face aux progrès des attaques de cryptanalyse.

À retenir

- Le principal avantage du chiffrement symétrique réside dans sa rapidité, les calculs informatiques requis étant relativement légers.
- Le principal inconvénient de la cryptographie à clé secrète est qu'il nécessite le partage préalable du secret, c'est-à-dire celui de la clé, entre l'émetteur et le destinataire du message.

En effet, si le canal de transmission est considéré comme suffisamment peu fiable au point de devoir chiffrer les messages, il serait dangereux d'y faire transiter la clé alors qu'elle doit rester secrète.
Les systèmes cryptographiques symétriques obligent, en outre, à avoir autant de clés que de correspondants pour maintenir une confidentialité individuelle.

D'autres solutions ont été imaginées pour pallier le problème parfois insoluble de la transmission sécurisée de la clé secrète : des systèmes à clé publique.

c. Cryptographie à clé publique

Les systèmes cryptographiques à clé publique sont asymétriques. Ils s'appuient sur des fonctions non réversibles.

À retenir

Cette approche cryptographique nécessite l'utilisation d'une **paire de clés**. Ce sont deux clés distinctes mais liées, et elles sont de nature différente : l'une d'elles est une clé publique et l'autre une clé privée.

- La **clé publique**, fournie par le destinataire, est utilisée par l'émetteur pour chiffrer le message clair.
- La **clé privée** est utilisée par le destinataire pour déchiffrer le message qui a été chiffré par l'émetteur avec la clé publique.

Astuce

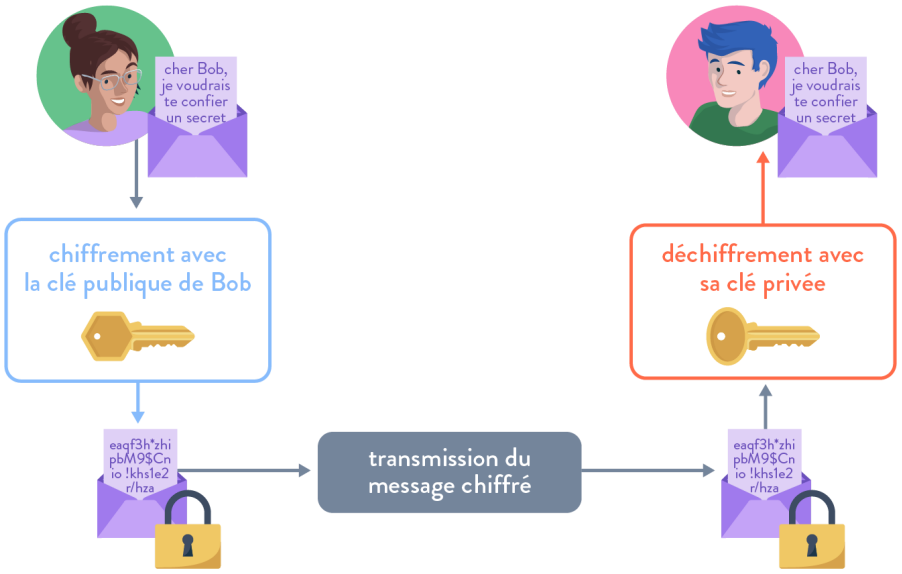
> Comme son nom l'indique, la clé publique peut être communiquée librement. En effet, si elle permet de chiffrer le message, elle ne permet pas de le déchiffrer. Sa transmission sur un canal non sécurisé ne pose donc aucun problème particulier.
> La clé privée n'a pas à être transmise, car elle est utilisée uniquement par le destinataire, pour déchiffrer le message qui lui a été transmis après chiffrement par la clé publique.

Retrouvons Alice, Bob et Ève, cette fois dans le contexte d'un système cryptographique à clé publique.

Exemple

> Alice souhaite pouvoir écrire des messages à Bob sans qu'Ève ne puisse en connaître la teneur.
>
> Pour pouvoir recevoir des messages d'Alice en utilisant un système cryptographique asymétrique, Bob génère une paire de clés :
>
> - une clé publique qu'il communique à Alice (et qu'il pourra communiquer à quiconque voudrait lui écrire) ;
> - une clé privée qu'il conserve précieusement et ne communique à personne.
>
> Alice utilise la clé publique mise à disposition par Bob pour chiffrer son message à l'attention de Bob.
> Alice transmet le message chiffré à Bob.
>
> Bob reçoit le message chiffré transmis par Alice.
> Bob déchiffre le message chiffré avec sa clé privée.

Ève peut éventuellement connaître la clé publique de Bob utilisée par Alice, mais cela n'a pas d'importance car cette clé publique ne lui sera d'aucune utilité : elle permet uniquement le chiffrement des messages. La clé privée est nécessaire pour pouvoir déchiffrer le message. Comme Bob ne la transmet à aucun moment, Ève ne peut pas en avoir connaissance.



Attention

La paire de clés détenue par Bob ne lui permet cependant pas de répondre à Alice.

S'il veut pouvoir écrire à Alice de manière confidentielle, il faut qu'Alice génère à son tour une paire de clés, et qu'elle communique sa clé publique à Bob. Bob pourra alors chiffrer son message avec la clé publique d'Alice, qui utilisera sa clé privée pour le déchiffrer.

Il existe de nombreux algorithmes à clé publique. Le plus connu est l'algorithme RSA.

Algorithme RSA

L'algorithme de chiffrement asymétrique RSA a été conçu au MIT. Le sigle RSA fait référence aux noms des trois cryptographes Ron Rivest, Adi Shamir et Leonard Adleman qui ont conçu cet algorithme en 1977.

Leur algorithme s'appuie sur l'utilisation d'un produit de deux grands nombres premiers et de fonctions à sens unique, dont le principe est d'être faciles à calculer mais beaucoup plus difficiles à inverser.
L'algorithme RSA utilise des clés de $1,024$ à $4,096$ bits.

À retenir

- Le principal avantage du chiffrement asymétrique est qu'il n'y a aucune clé secrète à transmettre, il suffit de communiquer la clé publique. En outre, chaque utilisateur n'a besoin que d'une clé privée unique pour pouvoir déchiffrer les messages en provenance de plusieurs correspondants, lesquels utiliseront la clé publique.
- Le principal inconvénient du chiffrement asymétrique est sa lenteur, du fait de la complexité des calculs informatiques requis.

L'autre inconvénient est qu'en cas de pluralité de correspondants, ce système cryptographique ne permet pas directement au destinataire de **savoir avec certitude qui est l'émetteur du message**.

Exemple

En utilisant la clé publique de Bob, Ève pourrait lui envoyer un message chiffré en se faisant passer pour Alice.

→ La clé privée de Bob lui permet de déchiffrer le message, mais pas d'en authentifier l'expéditeur.

Le chiffrement asymétrique permet en outre de **signer numériquement** un document ou un logiciel. Cela permet au destinataire de s'assurer de l'identité de l'auteur et de l'absence de modification du fichier transmis.

Exemple

Alice peut utiliser sa clé privée pour générer la signature numérique d'un document. Bob utilisera alors la clé publique d'Alice pour vérifier la signature du document transmis, qui lui en garantit l'authenticité et l'intégrité.

d. Comparaison des deux familles de chiffrement

Le tableau ci-après résume les principales caractéristiques distinctives des chiffrements symétrique et asymétrique.

À retenir

Famille cryptographique	symétrique	asymétrique
Clés	unique (commune), secrète	paire de clés
Échange de clés	problématique	non problématique (clé publique)
Temps de calcul	faible	important
Usages	confidentialité uniquement	confidentialité, authentification et intégrité

Nous allons maintenant étudier un cas d'usage mettant en œuvre ces différents systèmes cryptographiques pour sécuriser nos échanges sur Internet.

3 | Sécurisation des communications web

Dans la partie précédente, nous avons principalement étudié la cryptographie sous l'angle de la protection du contenu d'un message, mais nous avons indiqué que les outils cryptographiques asymétriques peuvent également servir à d'autres usages tels que l'**authentification** et la **signature de ces messages**.

Nous allons en découvrir un usage concret et coordonné avec la sécurisation des communications mises en œuvre par le protocole HTTPS. Le chiffrement a, en effet, pris une importance déterminante dans le fonctionnement des échanges numériques d'informations.

a. Limites de http

Le web s'est d'abord développé sur la base très ouverte du **protocole HTTP** : celui-ci était parfaitement adapté pour partager librement des connaissances. Mais l'essor du réseau a fait apparaître d'autres usages nécessitant de pouvoir authentifier des sites et sécuriser certains échanges, notamment ceux comportant des informations personnelles ou sensibles.

Exemple

> Les sites de commerce électronique traitent des ensembles de données sensibles :
>
> - données à caractère personnel de leurs clients (nom, adresses électronique et physique, téléphone) ;
> - données bancaires pour les paiements (référence de compte ou de carte bancaire).

L'absence de chiffrement dans le protocole HTTP exposait les données transmises à des risques d'écoutes et de manipulations au cours de leur acheminement par le réseau.

b. Sécurisation des échanges

HTTPS est une extension sécurisée du protocole HTTP, dans laquelle le protocole de communication est chiffré avec des protocoles cryptographiques.

Rappel

> Les URL en HTTP commencent par `http://` et utilisent par défaut le port 80.
> Les URL en HTTPS commencent par `https://` et utilisent par défaut le port 443.

Initialement réservé aux seuls échanges sensibles, l'usage du protocole HTTPS s'est généralisé ces dernières années. Les navigateurs web alertent désormais les internautes lorsque la connexion à un site n'est pas correctement sécurisée.

Le protocole HTTPS s'appuie sur des protocoles cryptographiques **sécurisant la couche de transport**.

c. Sécurité de la couche de transport

La sécurisation des communications est assurée au niveau de la couche de transport par le protocole cryptographique TLS.
Le sigle TLS signifie *Transport Layer Security*, soit « sécurité de la couche de transport ». Ce protocole est une évolution de SSL (*Secure Sockets Layer*) utilisée avant lui pour la sécurisation des échanges sur le web.

À retenir

> Le protocole TLS répond à trois objectifs de sécurité :
>
> - authentification ;
> - confidentialité ;
> - intégrité.

- L'**authentification** consiste à s'assurer de l'identité des tiers avec lesquels on échange. Dans le contexte d'une consultation web, l'authentification porte au

moins sur le serveur, et peut si nécessaire concerner aussi le client.

- La **confidentialité** consiste à chiffrer les échanges pour les rendre incompréhensibles pour des tiers observant le trafic entre le client et le serveur.
- **L'intégrité** consiste à s'assurer que les données échangées n'ont pas été altérées lors de leur acheminement.

d. Déroulement d'une session TLS

Une connexion TLS s'appuie sur un certain nombre d'échanges préliminaires entre le client et le serveur, dans le but de sécuriser un échange souhaité de données applicatives.

L'établissement d'une session se déroule en plusieurs phases consécutives.

1. **Négociation des paramètres de sécurité**
Le client et le serveur doivent se mettre d'accord sur les paramètres de sécurité qui seront utilisés, en particulier les méthodes de chiffrement : client et serveur doivent être capables de mettre en œuvre les mêmes algorithmes s'ils veulent se comprendre.

2. **Authentification du serveur et échange de clés**
Le serveur transmet au client un certificat numérique contenant sa clé publique. Le client peut vérifier la validité du certificat numérique auprès d'un tiers de confiance pour authentifier le serveur.
Le client génère une clé secrète, qu'il chiffre avec la clé publique du serveur avant de la lui transmettre. Dans le cas optionnel où le serveur exige une authentification du client, celui-ci envoie aussi un certificat.

3. **Adoption de la clé secrète par le serveur**
Le serveur déchiffre avec sa clé privé le message transmis par le client et prend connaissance de la clé secrète proposée par celui-ci. Il applique la méthode de chiffrement convenu précédemment.

À l'issue de ces phases :

- le serveur a été authentifié (et le client aussi si nécessaire) ;

- les flux réseaux entre le client et le serveur sont protégés : toutes les données applicatives sont encapsulées, garantissant leur confidentialité et leur intégrité.

e. Chiffrement hybride

À retenir

> Le protocole TLS a recours à un chiffrement hybride, qui combine l'usage des deux familles de systèmes cryptographiques que sont les chiffrements symétrique et asymétrique.

Le chiffrement asymétrique est utilisé dans un premier temps afin de pouvoir sécuriser un canal de communication entre le client et le serveur, afin d'authentifier le serveur et de pouvoir lui transmettre confidentiellement une clé secrète de chiffrement symétrique.

Le chiffrement asymétrique utilisé à ce stade, gourmand en calcul, ne concerne que les premiers échanges et porte sur de faibles volumes de données. Une fois la clé secrète transmise et déchiffrée par le serveur, les échanges basculent sur un chiffrement symétrique, beaucoup plus rapide à réaliser et donc moins pénalisant pour la transmission sécurisée de l'ensemble des données applicatives.

Le chiffrement hybride exploité par le protocole TLS exploite donc judicieusement les avantages particuliers de chacun des systèmes cryptographiques.

> Conclusion :
>
> Nous avons étudié et comparé la manière dont les protocoles de routage RIP et OSPF déterminaient les meilleures routes pour transmettre des flux de données. Nous avons ensuite présenté le chiffrement de données, effectué soit de manière symétrique avec une clé secrète, soit de manière asymétrique avec une clé publique. Nous avons enfin montré une application combinée de ces deux méthodes dans le cadre de la sécurisation des communications web.