Sécurisation des communications

Cours de spécialité NSI de Terminale

D Pihoué

Lycée Camille Jullian Bordeaux

5 octobre 2023

Capacités attendues

1. Décrire les principes de chiffrement symétrique (clef partagée) et asymétrique (avec clef privée / clef publique).
2. Décrire l'échange d'une clef symétrique en utilisant un protocole asymétrique pour sécuriser une communication HTTPS.

Capacités attendues

1. Décrire les principes de chiffrement symétrique (clef partagée) et asymétrique (avec clef privée / clef publique).
2. Décrire l'échange d'une clef symétrique en utilisant un protocole asymétrique pour sécuriser une communication HTTPS.

Des exemples de **stéganographie**, qui est une forme de **dissimulation**, ont laissé des traces dans l'histoire, comme la grille de Cardan, cette technique restant utilisée, voir cette page https://www.kaspersky.fr.
Il en va de même de méthodes de **cryptographie**, dont le but est de transmettre publiquement des messages **chiffrés**, voir cette page https://www.kaspersky.fr.

Les méthodes ancestrales de cryptographie

Elles peuvent être séparées en deux catégories.

Les méthodes ancestrales de cryptographie

Elles peuvent être séparées en deux catégories.

1. Celles par **transposition** avec lesquelles les lettres du message en clair sont mélangées selon des procédés le plus souvent mécaniques comme avec la scytale spartiate.

Les méthodes ancestrales de cryptographie

Elles peuvent être séparées en deux catégories.

1. Celles par **transposition** avec lesquelles les lettres du message en clair sont mélangées selon des procédés le plus souvent mécaniques comme avec la scytale spartiate.
2. Celles par **substitution** qui consistent à remplacer les caractères du message en clair par des symboles définis à l'avance. Le chiffre de César, qui consistait à décaler l'alphabet de trois lettres, en est un exemple assez célèbre. D'autres exemples sont proposés ici.

C'est le néerlandais August Kerckhoffs qui a énoncé les six principes de la cryptographie dans son ouvrage *La cryptographie militaire* édité en 1883.

C'est le néerlandais August Kerckhoffs qui a énoncé les six principes de la cryptographie dans son ouvrage *La cryptographie militaire* édité en 1883.

Trois principes révolutionnaires

1. « Le système doit être matériellement, sinon mathématiquement, indéchiffrable. »

C'est le néerlandais August Kerckhoffs qui a énoncé les six principes de la cryptographie dans son ouvrage *La cryptographie militaire* édité en 1883.

Trois principes révolutionnaires

1. « Le système doit être matériellement, sinon mathématiquement, indéchiffrable. »
2. « Il faut qu'il n'exige pas le secret et qu'il puisse sans inconvénient tomber entre les mains de l'ennemi. »

C'est le néerlandais August Kerckhoffs qui a énoncé les six principes de la cryptographie dans son ouvrage *La cryptographie militaire* édité en 1883.

Trois principes révolutionnaires

1. « Le système doit être matériellement, sinon mathématiquement, indéchiffrable. »
2. « Il faut qu'il n'exige pas le secret et qu'il puisse sans inconvénient tomber entre les mains de l'ennemi. »
3. « La clef doit pouvoir être communiquée et retenue sans le secours de notes écrites, et être changée ou modifiée au gré des correspondants. »

Définition

Les méthodes de **cryptanalyse** sont les attaques pour **déchiffrer** sans connaître cette clef secrète.

Définition

Les méthodes de **cryptanalyse** sont les attaques pour **déchiffrer** sans connaître cette clef secrète.

Les méthodes de substitution sont attaquables par analyse des fréquences avec, le cas échéant, une estimation de la longueur de la clef par recherche de l'indice de coïncidence.

Définition

Les méthodes de **cryptanalyse** sont les attaques pour **déchiffrer** sans connaître cette clef secrète.

Les méthodes de substitution sont attaquables par analyse des fréquences avec, le cas échéant, une estimation de la longueur de la clef par recherche de l'indice de coïncidence.

Un court descriptif est consultable sur cette page.

Les deux guerres mondiales ainsi que l'évolution des moyens de communication ont provoqué un essor des machines à **chiffrer** comme Enigma dont le cassage du code a rendu célèbre Alan Turing et accéléré le développement de l'informatique avec les bombes.

Les deux guerres mondiales ainsi que l'évolution des moyens de communication ont provoqué un essor des machines à **chiffrer** comme Enigma dont le cassage du code a rendu célèbre Alan Turing et accéléré le développement de l'informatique avec les bombes.

À l'issue de la seconde guerre mondiale Claude Shannon développe la théorie de l'information dont la cryptologie forme l'une des branches.

Les deux guerres mondiales ainsi que l'évolution des moyens de communication ont provoqué un essor des machines à **chiffrer** comme Enigma dont le cassage du code a rendu célèbre Alan Turing et accéléré le développement de l'informatique avec les bombes.

À l'issue de la seconde guerre mondiale Claude Shannon développe la théorie de l'information dont la cryptologie forme l'une des branches.

Cette théorie est aussi un fondement théorique de l'informatique.

Définition

Un chiffrement est symétrique lorsque la même clef est utilisée pour chiffrer et pour déchiffrer.

Définition

Un chiffrement est symétrique lorsque la même clef est utilisée pour chiffrer et pour déchiffrer.

Ce type de méthode est aussi appelé **chiffrement à clef secrète**.

Description formelle

Un système de chiffrement symétrique est décrit par 3 algorithmes :

Description formelle

Un système de chiffrement symétrique est décrit par 3 algorithmes :

- pour générer une **clef secrète**, notée k, partagée par les deux interlocuteurs ;

Description formelle

Un système de chiffrement symétrique est décrit par 3 algorithmes :

- pour générer une **clef secrète**, notée k, partagée par les deux interlocuteurs ;
- pour **chiffrer**, prenant en entrée un message clair m ainsi que la clef secrète k et renvoyant un message chiffré c ;

Description formelle

Un système de chiffrement symétrique est décrit par 3 algorithmes :

- pour générer une **clef secrète**, notée k, partagée par les deux interlocuteurs ;
- pour **chiffrer**, prenant en entrée un message clair m ainsi que la clef secrète k et renvoyant un message chiffré c ;
- pour **déchiffrer**, prenant en entrée un message chiffré c ainsi que la clef secrète k et renvoyant le message clair m.

Le système est **correct** si la propriété

```
dechiffrement(chiffrement(m, k), k) = m
```

est vérifiée.

Le système est **correct** si la propriété

```
dechiffrement(chiffrement(m, k), k) = m
```

est vérifiée.

Il est **sûr** s'il est impossible de trouver de l'information sur m à partir de la seule connaissance de c.

Les données, message et clef, sont généralement codées en binaire et les clefs sont des suites de bits aléatoires.

Les données, message et clef, sont généralement codées en binaire et les clefs sont des suites de bits aléatoires.

Première catégorie

Les systèmes par **flots** par lesquels les bits du message m sont chiffrés un à un en utilisant à chaque fois un bit de masque généré aléatoirement à partir de la clef.

Les données, message et clef, sont généralement codées en binaire et les clefs sont des suites de bits aléatoires.

Première catégorie

Les systèmes par **flots** par lesquels les bits du message m sont chiffrés un à un en utilisant à chaque fois un bit de masque généré aléatoirement à partir de la clef.

C'est le modèle du chiffrement de Vernam.

Les données, message et clef, sont généralement codées en binaire et les clefs sont des suites de bits aléatoires.

Première catégorie

Les systèmes par **flots** par lesquels les bits du message m sont chiffrés un à un en utilisant à chaque fois un bit de masque généré aléatoirement à partir de la clef.

C'est le modèle du chiffrement de Vernam.

Le chiffrement par flots le plus utilisé actuellement est sans doute ChaCha20.

Seconde catégorie

Les systèmes par **blocs** par lesquels le message m est découpé en blocs de taille fixe, chaque bloc est **chiffré** à l'aide d'une clef dérivée de la clef initiale. Ils procèdent en général de façon itérative en un certain nombre de **tours**.

Seconde catégorie

Les systèmes par **blocs** par lesquels le message m est découpé en blocs de taille fixe, chaque bloc est **chiffré** à l'aide d'une clef dérivée de la clef initiale. Ils procèdent en général de façon itérative en un certain nombre de **tours**.

Le DES, pour *Data Encryption Standard*, fut le premier schéma standardisé en 1977. Il utilisait des blocs de 64 bits et des clefs de 56 bits.

Seconde catégorie

Les systèmes par **blocs** par lesquels le message m est découpé en blocs de taille fixe, chaque bloc est **chiffré** à l'aide d'une clef dérivée de la clef initiale. Ils procèdent en général de façon itérative en un certain nombre de **tours**.

Le DES, pour *Data Encryption Standard*, fut le premier schéma standardisé en 1977. Il utilisait des blocs de 64 bits et des clefs de 56 bits.

Déclaré obsolète vers 1997, il a été remplacé en 2001 par l'AES, pour *Advanced Encryption Standard*, basé sur l'algorithme élaboré par les deux chercheurs belges Joan Daemen et Vincent Rijmen nommé Rijndael.

En plus d'être sûrs, ces chiffrements sont très efficaces et permettent de chiffrer de grandes quantités de données en temps réel.

En plus d'être sûrs, ces chiffrements sont très efficaces et permettent de chiffrer de grandes quantités de données en temps réel.

Leur principal inconvénient est la nécessité d'utiliser une clef secrète et donc de disposer d'un moyen d'échanger cette clef sans qu'elle soit interceptée.

Problématique

Une solution consiste à créer cette clef à la volée au début de la communication, mais comment faire en sorte que deux interlocuteurs qui ne se connaissent pas parviennent de manière sûre à construire un secret commun ?

Problématique

Une solution consiste à créer cette clef à la volée au début de la communication, mais comment faire en sorte que deux interlocuteurs qui ne se connaissent pas parviennent de manière sûre à construire un secret commun ?

Une solution à ce problème a été proposée en 1976 par Whitfield Diffie et Martin Hellman, deux étasuniens avec le chiffrement asymétrique.

On nomme Alice et Bob les deux interlocuteurs.

On nomme Alice et Bob les deux interlocuteurs.

Description par analogie

Alice fabrique un cadenas et une clef, elle transmet le cadenas ouvert à Bob qui l'utilise pour fermer une boîte dans laquelle il a placé son message. Ainsi, seule Alice peut ouvrir la boîte et lire le message.

On nomme Alice et Bob les deux interlocuteurs.

Description par analogie

Alice fabrique un cadenas et une clef, elle transmet le cadenas ouvert à Bob qui l'utilise pour fermer une boîte dans laquelle il a placé son message. Ainsi, seule Alice peut ouvrir la boîte et lire le message.

Vocabulaire

- Le cadenas ouvert constitue la **clef publique** d'Alice et la clef associée au cadenas et sa **clef privée**.
- Le système est **asymétrique** car la clef de chiffrement n'est pas la même que celle de déchiffrement.
- Ce système est aussi appelé **chiffrement à clef publique**.

Parallèlement à ce principe, DIFFIE et HELLMAN ont élaboré un protocole d'échange de clef qui porte leurs noms.

Parallèlement à ce principe, DIFFIE et HELLMAN ont élaboré un protocole d'échange de clef qui porte leurs noms.

Définition

On considère trois nombres entiers p, a et x avec p premier et $1 \leqslant a \leqslant p-1$.

Parallèlement à ce principe, DIFFIE et HELLMAN ont élaboré un protocole d'échange de clef qui porte leurs noms.

Définition

On considère trois nombres entiers p, a et x avec p premier et $1 \leqslant a \leqslant p-1$.

- Le calcul du nombre entier y définit par $y = a^x \mod p$ est facile.

Parallèlement à ce principe, DIFFIE et HELLMAN ont élaboré un protocole d'échange de clef qui porte leurs noms.

Définition

On considère trois nombres entiers p, a et x avec p premier et $1 \leqslant a \leqslant p-1$.

- Le calcul du nombre entier y définit par $y = a^x \mod p$ est facile.
- Le calcul du nombre entier x connaissant y, a et p est très difficile si p est très grand.

Parallèlement à ce principe, DIFFIE et HELLMAN ont élaboré un protocole d'échange de clef qui porte leurs noms.

Définition

On considère trois nombres entiers p, a et x avec p premier et $1 \leqslant a \leqslant p-1$.

- Le calcul du nombre entier y définit par $y = a^x \mod p$ est facile.
- Le calcul du nombre entier x connaissant y, a et p est très difficile si p est très grand.

On dit que x est le logarithme discret de y de base a et modulo p.

En pratique, le protocole se déroule en 5 étapes.

En pratique, le protocole se déroule en 5 étapes.

Étapes du protocole

1. Alice et Bob choisissent un grand nombre premier p et un nombre entier a compris entre 1 et $p-1$.

En pratique, le protocole se déroule en 5 étapes.

Étapes du protocole

1. Alice et Bob choisissent un grand nombre premier p et un nombre entier a compris entre 1 et $p-1$.
2. Alice choisit un nombre entier x_1 qu'elle garde secret, Bob fait de même avec x_2.

En pratique, le protocole se déroule en 5 étapes.

Étapes du protocole

1. Alice et Bob choisissent un grand nombre premier p et un nombre entier a compris entre 1 et $p-1$.
2. Alice choisit un nombre entier x_1 qu'elle garde secret, Bob fait de même avec x_2.
3. Alice calcule $y_1 = a^{x_1} \mod p$ et Bob calcule $y_2 = a^{x_2} \mod p$.

En pratique, le protocole se déroule en 5 étapes.

Étapes du protocole

1. Alice et Bob choisissent un grand nombre premier p et un nombre entier a compris entre 1 et $p-1$.
2. Alice choisit un nombre entier x_1 qu'elle garde secret, Bob fait de même avec x_2.
3. Alice calcule $y_1 = a^{x_1} \mod p$ et Bob calcule $y_2 = a^{x_2} \mod p$.
4. Alice et Bob s'échangent les nombres y_1 et y_2 par un canal non sécurisé.

En pratique, le protocole se déroule en 5 étapes.

Étapes du protocole

1. Alice et Bob choisissent un grand nombre premier p et un nombre entier a compris entre 1 et $p-1$.
2. Alice choisit un nombre entier x_1 qu'elle garde secret, Bob fait de même avec x_2.
3. Alice calcule $y_1 = a^{x_1} \mod p$ et Bob calcule $y_2 = a^{x_2} \mod p$.
4. Alice et Bob s'échangent les nombres y_1 et y_2 par un canal non sécurisé.
5. Alice calcule $y_2^{x_1} \mod p$ et Bob $y_1^{x_2} \mod p$ qui sont égaux à $a^{x_1 x_2} \mod p$ et ce nombre va constituer la clef secrète k qu'ils sont seuls à partager.

Sécurité

Si une tierce personne intercepte les échanges non sécurisés entre Alice et Bob alors

Sécurité

Si une tierce personne intercepte les échanges non sécurisés entre Alice et Bob alors

- elle ne peut pas trouver la clef secrète k car il lui manque systématiquement une information ;

Sécurité

Si une tierce personne intercepte les échanges non sécurisés entre Alice et Bob alors

- elle ne peut pas trouver la clef secrète k car il lui manque systématiquement une information ;
- elle ne peut pas trouver les nombres entiers x_1 et x_2 car la résolution du logarithme discret est un problème trop difficile.

Définition

On appelle **fonction à sens unique à trappe** une fonction $y = f(x)$ qui vérifie les deux propriétés suivantes.

1. Le calcul de y connaissant x est facile mais la recherche de x connaissant y est très difficile. La fonction f est ainsi **à sens unique**.
2. Il existe cependant une **trappe** qui va permettre de trouver x à partir de y si on dispose de cette information.

Un autre usage du chiffrement asymétrique repose sur l'algorithme RSA, publié en 1977, acronyme des noms de ses trois découvreurs Ron Rivest, Adi Shamir et Len Adleman respectivement étasunien, israélien et étasunien.

Un autre usage du chiffrement asymétrique repose sur l'algorithme RSA, publié en 1977, acronyme des noms de ses trois découvreurs Ron Rivest, Adi Shamir et Len Adleman respectivement étasunien, israélien et étasunien.

Un exemple de fonctionnement de cet algorithme est présenté ici.

Un autre usage du chiffrement asymétrique repose sur l'algorithme RSA, publié en 1977, acronyme des noms de ses trois découvreurs Ron Rivest, Adi Shamir et Len Adleman respectivement étasunien, israélien et étasunien.

Un exemple de fonctionnement de cet algorithme est présenté ici.

De plus, les deux opérations de chiffrement et de déchiffrement peuvent être réalisées dans n'importe quel ordre, leur composition conduit toujours au message initial.

Avec ce système de chiffrement asymétrique, chaque utilisatrice ou utilisateur peut afficher sa clef publique.

Avec ce système de chiffrement asymétrique, chaque utilisatrice ou utilisateur peut afficher sa clef publique.

Il suffit de chiffrer un message avec cette clef, seule la personne destinataire est en capacité de déchiffrer avec sa clef privée.

Avec ce système de chiffrement asymétrique, chaque utilisatrice ou utilisateur peut afficher sa clef publique.

Il suffit de chiffrer un message avec cette clef, seule la personne destinataire est en capacité de déchiffrer avec sa clef privée.

On dit qu'il est à **clef publique**.

Avec ce système de chiffrement asymétrique, chaque utilisatrice ou utilisateur peut afficher sa clef publique.

Il suffit de chiffrer un message avec cette clef, seule la personne destinataire est en capacité de déchiffrer avec sa clef privée.

On dit qu'il est à **clef publique**.

Cet algorithme est sûr au sens où personne ne dispose de la puissance de calcul nécessaire pour trouver la clef privée dès lors que les deux clefs sont assez longues.

Avec ce système de chiffrement asymétrique, chaque utilisatrice ou utilisateur peut afficher sa clef publique.

Il suffit de chiffrer un message avec cette clef, seule la personne destinataire est en capacité de déchiffrer avec sa clef privée.

On dit qu'il est à **clef publique**.

Cet algorithme est sûr au sens où personne ne dispose de la puissance de calcul nécessaire pour trouver la clef privée dès lors que les deux clefs sont assez longues.

Cependant les temps de calculs sont très longs, ce qui ne le rend pas praticable pour chiffrer de grandes quantités de données.

Des failles

Rien empêche un tiers malveillant de s'infiltrer entre les deux interlocuteurs soit

- s'il y a distribution des clefs, en se faisant passer pour l'interlocuteur, par exemple par copie (c'est le hameçonnage) ;
- s'il y a échange d'une clef secrète, en transmettant la même valeur $y_3 = a^{x_3} \mod p$ à Alice et Bob pour constituer une clef secrète avec chacun des deux.

Des failles

Rien empêche un tiers malveillant de s'infiltrer entre les deux interlocuteurs soit

- s'il y a distribution des clefs, en se faisant passer pour l'interlocuteur, par exemple par copie (c'est le hameçonnage) ;
- s'il y a échange d'une clef secrète, en transmettant la même valeur $y_3 = a^{x_3} \mod p$ à Alice et Bob pour constituer une clef secrète avec chacun des deux.

Il s'agit de l'attaque dite **de l'homme au milieu**, MITM pour *Man In The Middle attack*.

Une solution repose sur l'**authentification** des deux interlocuteurs.

Une solution repose sur l'**authentification** des deux interlocuteurs.

Les 3 algorithmes du protocole de **signature électronique**

- Un de génération de clef qui renvoie une **clef publique** pk et une **clef privée** sk, cette dernière étant uniquement communiquée à Alice.

Une solution repose sur l'**authentification** des deux interlocuteurs.

Les 3 algorithmes du protocole de **signature électronique**

- Un de génération de clef qui renvoie une **clef publique** pk et une **clef privée** sk, cette dernière étant uniquement communiquée à Alice.
- Un de **signature**, qui prend en arguments un message m, public ou privé, ainsi qu'une **clef privée** sk et renvoie une **signature** publique s.

Une solution repose sur l'**authentification** des deux interlocuteurs.

Les 3 algorithmes du protocole de **signature électronique**

- Un de génération de clef qui renvoie une **clef publique** pk et une **clef privée** sk, cette dernière étant uniquement communiquée à Alice.
- Un de **signature**, qui prend en arguments un message m, public ou privé, ainsi qu'une **clef privée** sk et renvoie une **signature** publique s.
- Un de **vérification**, qui prend en arguments un message m avec sa signature s et qui renvoie `Vrai` si la signature est valide et `Faux` sinon.

Une solution repose sur l'**authentification** des deux interlocuteurs.

Les 3 algorithmes du protocole de **signature électronique**

- Un de génération de clef qui renvoie une **clef publique** pk et une **clef privée** sk, cette dernière étant uniquement communiquée à Alice.
- Un de **signature**, qui prend en arguments un message m, public ou privé, ainsi qu'une **clef privée** sk et renvoie une **signature** publique s.
- Un de **vérification**, qui prend en arguments un message m avec sa signature s et qui renvoie `Vrai` si la signature est valide et `Faux` sinon.

Ainsi, Alice envoie à Bob le couple (m, s) et Bob authentifie la signature d'Alice ce qui lui permet d'être sûr qu'Alice est bien l'expéditrice.

Exemple

1. Alice choisit une fonction de **hachage** h qu'elle applique à son message m pour en produire une empreinte h(m).

Exemple

1. Alice choisit une fonction de **hachage** h qu'elle applique à son message m pour en produire une empreinte h(m).
2. Alice chiffre cette empreinte h(m) avec sa **clef privée** pour constituer sa **signature** s.

Exemple

1. Alice choisit une fonction de **hachage** h qu'elle applique à son message m pour en produire une empreinte h(m).
2. Alice chiffre cette empreinte h(m) avec sa **clef privée** pour constituer sa **signature** s.
3. Alice envoie à Bob le couple (m, s) ainsi que le choix de la fonction de hachage h.

Exemple

1. Alice choisit une fonction de **hachage** h qu'elle applique à son message m pour en produire une empreinte h(m).
2. Alice chiffre cette empreinte h(m) avec sa **clef privée** pour constituer sa **signature** s.
3. Alice envoie à Bob le couple (m, s) ainsi que le choix de la fonction de hachage h.

   À cette étape, Alice peut aussi chiffrer le tout avec la **clef publique** de Bob. Elle sera alors certaine que seul Bob peut déchiffrer avec sa **clef privée**.

Exemple

1. Alice choisit une fonction de **hachage** h qu'elle applique à son message m pour en produire une empreinte h(m).
2. Alice chiffre cette empreinte h(m) avec sa **clef privée** pour constituer sa **signature** s.
3. Alice envoie à Bob le couple (m, s) ainsi que le choix de la fonction de hachage h.

   À cette étape, Alice peut aussi chiffrer le tout avec la **clef publique** de Bob. Elle sera alors certaine que seul Bob peut déchiffrer avec sa **clef privée**.
4. À réception, après avoir éventuellement déchiffré, Bob calcule l'empreinte h(m) du message et la compare avec le déchiffrement de la signature s à partir de la **clef publique** d'Alice.

La conception du protocole HTTPS, pour *HyperText Transfer Protocol Secure* repose sur les algorithmes de chiffrements présentés mais aussi sur les principes qui suivent :

La conception du protocole HTTPS, pour *HyperText Transfer Protocol Secure* repose sur les algorithmes de chiffrements présentés mais aussi sur les principes qui suivent :

Principes d'élaboration

- il doit être compatible avec HTTP,

La conception du protocole HTTPS, pour *HyperText Transfer Protocol Secure* repose sur les algorithmes de chiffrements présentés mais aussi sur les principes qui suivent :

Principes d'élaboration

- il doit être compatible avec HTTP,
- il doit être modulaire et maintenir la compatibilité dans le futur, il faut pouvoir changer la difficulté des problèmes à résoudre sans remettre en cause tout le protocole,

La conception du protocole HTTPS, pour *HyperText Transfer Protocol Secure* repose sur les algorithmes de chiffrements présentés mais aussi sur les principes qui suivent :

Principes d'élaboration

- il doit être compatible avec HTTP,
- il doit être modulaire et maintenir la compatibilité dans le futur, il faut pouvoir changer la difficulté des problèmes à résoudre sans remettre en cause tout le protocole,
- il doit être performant donc limiter l'usage des méthodes asymétriques au minimum nécessaire pour la sécurité.

La conception du protocole HTTPS, pour *HyperText Transfer Protocol Secure* repose sur les algorithmes de chiffrements présentés mais aussi sur les principes qui suivent :

Principes d'élaboration

- il doit être compatible avec HTTP,
- il doit être modulaire et maintenir la compatibilité dans le futur, il faut pouvoir changer la difficulté des problèmes à résoudre sans remettre en cause tout le protocole,
- il doit être performant donc limiter l'usage des méthodes asymétriques au minimum nécessaire pour la sécurité.

Ce protocole est la réunion du protocole HTTP, pour les interactions avec le serveur WEB, et du protocole TLS, pour *Transport Layer Security*, au niveau de la couche transport.

Propriétés

Ce protocole ajoute une phase permettant l'**authentification** du serveur et la mise en place d'une **clef secrète** de **chiffrement symétrique** appelée **clef de session**.

Propriétés

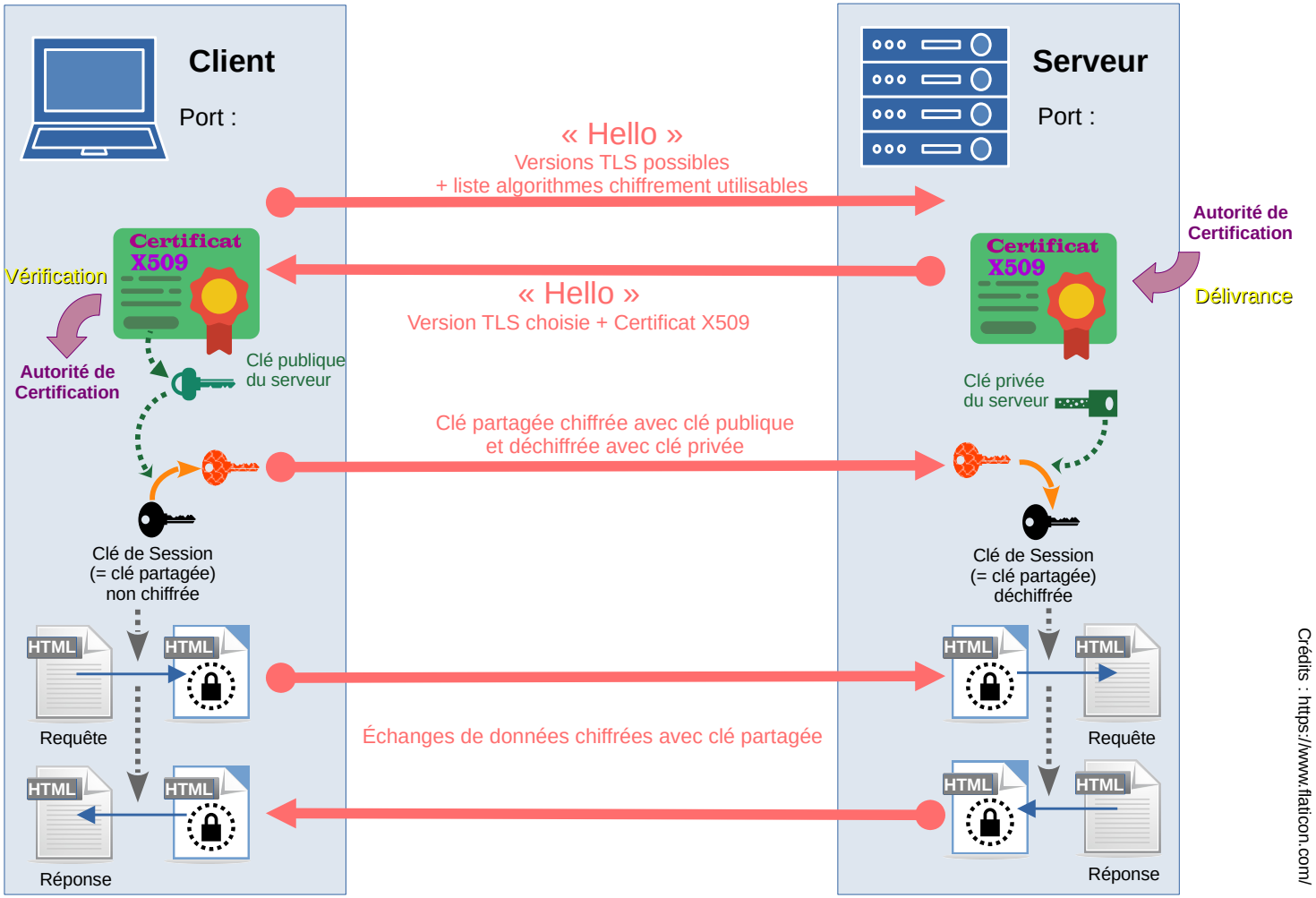
Ce protocole ajoute une phase permettant l'**authentification** du serveur et la mise en place d'une **clef secrète** de **chiffrement symétrique** appelée **clef de session**.

- L'**authentification** empêche l'attaque de l'homme au milieu.

Propriétés

Ce protocole ajoute une phase permettant l'**authentification** du serveur et la mise en place d'une **clef secrète** de **chiffrement symétrique** appelée **clef de session**.

- L'**authentification** empêche l'attaque de l'homme au milieu.
- Le **chiffrement** empêche les routeurs et autres machines intermédiaires de connaître les données échangées.



Crédits : https://www.flaticon.com/

Déroulement de TLS *Handshake*

Déroulement de TLS *Handshake*

1. Le client envoie au serveur un message initial indiquant des options.

Déroulement de TLS *Handshake*

1. Le client envoie au serveur un message initial indiquant des options.
2. Le serveur répond au client en envoyant entre autres son certificat X509 délivré par son **autorité de certification** et sa **clef publique** pour le protocole **asymétrique** choisi.

Déroulement de TLS *Handshake*

1. Le client envoie au serveur un message initial indiquant des options.
2. Le serveur répond au client en envoyant entre autres son certificat X509 délivré par son **autorité de certification** et sa **clef publique** pour le protocole **asymétrique** choisi.
3. Le client vérifie le certificat au moyen de la **clef publique** de l'**autorité de certification** ainsi que la validité du certificat.

Déroulement de TLS *Handshake*

1. Le client envoie au serveur un message initial indiquant des options.
2. Le serveur répond au client en envoyant entre autres son certificat X509 délivré par son **autorité de certification** et sa **clef publique** pour le protocole **asymétrique** choisi.
3. Le client vérifie le certificat au moyen de la **clef publique** de l'**autorité de certification** ainsi que la validité du certificat.
4. Client et serveur conviennent d'une **clef secrète** de **chiffrement symétrique** k via le protocole **asymétrique**.

Déroulement de TLS *Handshake*

1. Le client envoie au serveur un message initial indiquant des options.
2. Le serveur répond au client en envoyant entre autres son certificat X509 délivré par son **autorité de certification** et sa **clef publique** pour le protocole **asymétrique** choisi.
3. Le client vérifie le certificat au moyen de la **clef publique** de l'**autorité de certification** ainsi que la validité du certificat.
4. Client et serveur conviennent d'une **clef secrète** de **chiffrement symétrique** k via le protocole **asymétrique**.

   k est désignée comme **clef de session**.

Déroulement de TLS *Handshake*

1. Le client envoie au serveur un message initial indiquant des options.
2. Le serveur répond au client en envoyant entre autres son certificat X509 délivré par son **autorité de certification** et sa **clef publique** pour le protocole **asymétrique** choisi.
3. Le client vérifie le certificat au moyen de la **clef publique** de l'**autorité de certification** ainsi que la validité du certificat.
4. Client et serveur conviennent d'une **clef secrète** de **chiffrement symétrique** k via le protocole **asymétrique**.

   k est désignée comme **clef de session**.
5. Les données qui seront échangées seront toutes chiffrées avec la clef k via le **chiffrement symétrique** retenu.

La **confidentialité** des données est ainsi garantie.

La **confidentialité** des données est ainsi garantie.

Un autre protocole MAC, pour *Message Authentification Code*, assure l'**intégrité** des données, toujours avec la clef k.

L'**authentification** du serveur est garantie par un **tiers de confiance** nommé **autorité de certification**.

L'**authentification** du serveur est garantie par un **tiers de confiance** nommé **autorité de certification**.

Les 3 niveaux de certification

L'**authentification** du serveur est garantie par un **tiers de confiance** nommé **autorité de certification**.

Les 3 niveaux de certification

DV pour *Domain Validated*, l'autorité a juste vérifié que la personne qui a demandé le certificat est bien celle qui contrôle le nom de domaine certifié.

L'**authentification** du serveur est garantie par un **tiers de confiance** nommé **autorité de certification**.

Les 3 niveaux de certification

DV pour *Domain Validated*, l'autorité a juste vérifié que la personne qui a demandé le certificat est bien celle qui contrôle le nom de domaine certifié.

OV pour *Organisation Validated*, l'autorité effectue des vérifications relatives à l'existence légale de la personne physique ou morale associée à l'organisation certifiée.

L'**authentification** du serveur est garantie par un **tiers de confiance** nommé **autorité de certification**.

Les 3 niveaux de certification

DV pour *Domain Validated*, l'autorité a juste vérifié que la personne qui a demandé le certificat est bien celle qui contrôle le nom de domaine certifié.

OV pour *Organisation Validated*, l'autorité effectue des vérifications relatives à l'existence légale de la personne physique ou morale associée à l'organisation certifiée.

EV pour *Extended Validated*, l'autorité vérifie les documents légaux fournis par l'entreprise à certifier et recoupe ces vérifications avec les registres publics.