	**Architecture matérielle**	**NSI T^{ale}**
	HTTPS	Cours/TD

*

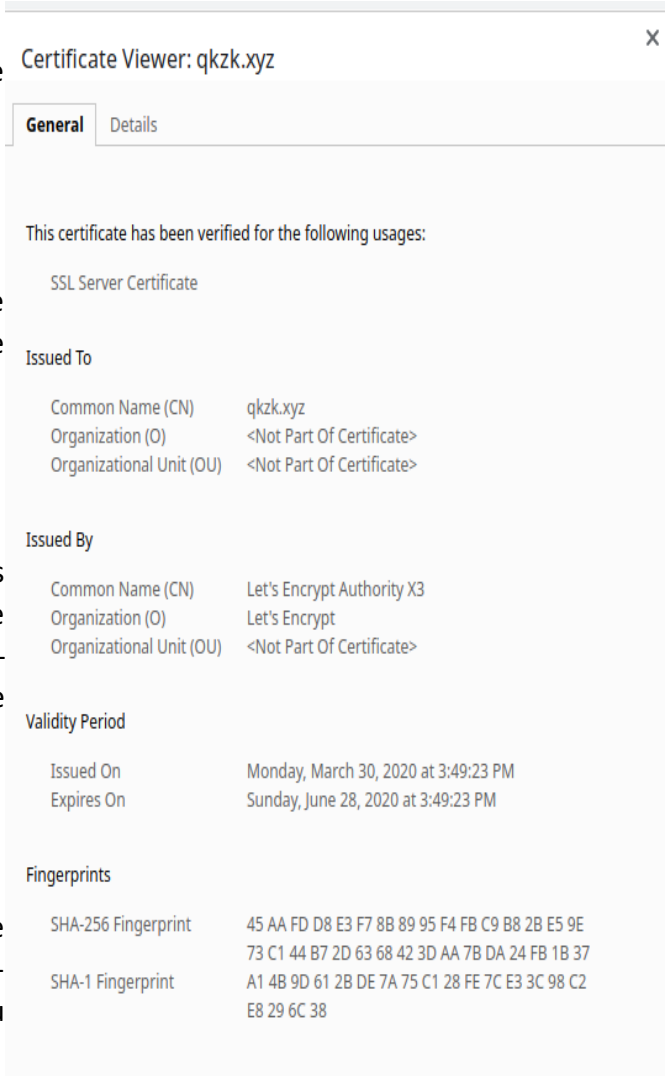
1. Principe

HTTPS (littéralement « protocole de transfert hypertextuel sécurisé ») est la combinaison du protocole **HTTP** et d'une couche de chiffrement, généralement **TLS** (sécurité de la couche transport).

2. Objectifs

Trois objectifs sont visés par ce protocole :

- **Authenticité** : un certificat assure que vous visitez bien le site voulu.
- **Confidentialité** : les échanges sont chiffrés et ne peuvent être lus par un tiers.
- **Intégrité** : HTTPS rend la technique "man in the middle" quasi impossible.



2.1. Authenticité

HTTPS permet aux visiteurs de vérifier l'identité du site web auquel il se connecte. Cette identité est assurée par un certificat délivré par une autorité.

2.2. Confidentialité

Les échanges étant chiffrés, seuls les détenteurs des clés peuvent les lire. Cela assure que votre numéro de carte bancaire n'est pas lisible par un tiers qui parviendrait à pirater un routeur ou lorsque vous êtes sur une connexion partagée.

2.3. Intégrité

Lors de ce chiffrage on ajoute une étape qui permet de s'assurer que le message transmis est intègre (c'est-à-dire qu'il n'a pas été altéré par une tierce personne ou des erreurs de transmission).

3. *Etapes*

HTTPS procède en deux temps :

1) Négociation (poignée de main) : échange de clés, validées par un certificat (cryptographie **asymétrique**)
2) Communication : échange de données chiffrées (cryptographie **symétrique**)

3.1.Étapes d'une connexion http… qui se passe mal

1) Vous ouvrez votre navigateur et vous rendez sur un site HTTP. Votre navigateur envoie une requête `GET` à l'adresse ip du site. Cet envoi se fait en clair. Toutes les machines entre votre ordinateur et le serveur peuvent la lire.
2) Le serveur reçoit la requête `GET` et y répond avec le contenu de la page souhaitée.
3) Vous remplissez votre formulaire : numéro de carte bleue, date d'expiration, etc.
   `Submit` !
   Votre navigateur envoie une requête `POST` avec les informations en clair. N'importe qui peut les lire.
   Toutes les machines intermédiaires peuvent lire ce message.
   (Il y a **trente** sauts entre mon pc et mon site web, par exemple)
4) Un des routeurs entre votre PC et le serveur web a été piraté ! **Tous les messages sont interceptés et envoyés vers un site extérieur.**
   Les pirates disposent maintenant de vos données bancaires.
5) Le routeur transmet néanmoins l'information à votre site… qui n'est au courant de rien. Il valide la transaction et vous en informe.
6) Vous recevez cette confirmation et ne vous doutez de rien.

Etc.

3.2.*Comment empêcher cette intrusion ?*

En deux temps :

1) **Négociation : la poignée de main**

La phase de négociation assure l'**authenticité** de l'interlocuteur.

0. Lorsque vous vous connectez, vous recevez un certificat transmis par le site. Ce certificat a été délivré par une autorité, une entreprise qui généralement ne fait que ça et en laquelle tout le monde a confiance.
   Ainsi, vous êtes rassuré : le site visité n'est pas celui d'un faussaire.

Chiffrement **asymétrique** : pour initialiser la connexion.

1. Ce certificat étant transmis, il contient donc une **clé publique** qui permet de chiffrer un message.
2. De votre côté le navigateur chiffre sa clé publique avec la clé publique du certificat et la retourne au serveur.
   Tout le monde peut intercepter ce message, mais **seul le serveur** peut le déchiffrer.
3. Il déchiffre avec sa clé privée,
   Il calcule une **clé secrète** et la chiffre avec votre clé publique.
   Il vous la renvoie.
4. Vous recevez le message, le déchiffrez avec votre **clé privée** (seul vous pouvez le faire).

BOOM. **Client et serveur sont seuls détenteurs d'une clé secrète commune.**

2) Chiffrement symétrique : durant la communication

Le chiffrement asymétrique est très pratique mais aussi très lent et coûteux en ressources.
Une fois l'authenticité avérée et qu'une clé secrète a pu être échangée, la communication réelle commence.
Toutes les données sont maintenant chiffrées de manière symétrique avec la clé secrète.
Commence alors l'échange HTTP habituel… mais tous les messages - y compris les adresses (pages internes visitées, mots clés) sont chiffrées.
Vos messages sont maintenant confidentiels (seul le serveur et vous détenez la clé secrète) et authentiques (la clé secrète permet aussi de vérifier que le message n'a pas été altéré durant le transport).

3.3. *Résumé des étapes d'une communication TLS*

A chaque envoi de données le serveur :
- découpe les données en paquets,
- compresse les données,
- chiffre les paquets avec votre clé secrète commune,
- signe les données avec sa clé privée,
- les envoie

A chaque réception de données le client :
- déchiffre avec la clé secrète,
- vérifie la signature avec la clé publique du serveur,
- décompresse les données,
- les assemble

4. Précisions

4.1. *Choix d'un algorithme de chiffrement.*

Afin d'assurer la rétrocompatibilité, plusieurs algorithmes sont proposés lors de la poignée de main.
Certains serveurs n'ont pas été mis à jour depuis longtemps, certains clients aussi parfois. Aussi tous n'ont peut-être pas les algorithmes les plus récents à leur disposition.

Durant la poignée de main, le client et le serveur commencent par choisir un algorithme de chiffrement commun.

- Les algorithmes **asymétriques** courants sont **RSA** (avec différentes longueurs de clé) qui repose sur l'arithmétique modulaire et un algorithme similaire à Diffie Hellman mais reposant les courbes elliptiques.
- L'algorithme **symétrique** courant est **AES**. Successeur de DES.
- Ces algorithmes sont tous rendus publics, cela peut sembler contre-intuitif mais c'est le meilleur moyen de repérer les failles et de les corriger.

4.2. *Délivrance des certificats.*

L'autorité des entreprises qui délivrent les certificats est assurée parce qu'elles rendent publiques leurs opérations.
Malheureusement, les révélations d'Edward Snowden en 2013 laissent penser qu'il est possible de les attaquer afin d'établir de faux certificats.

4.3. *Contenu des certificats.*

Quelles informations trouve-t-on dans un certificat ?
Le responsable du site a demandé ce certificat et, suite à des échanges d'informations (et parfois contre de l'argent), un certificat lui a été remis. Qui est le propriétaire du site, où habite-t-il etc.
Tous les certificats ne se valent pas. Mais tous sont sûrs.
Celui-ci comporte une clé que vous pouvez vérifier auprès de l'autorité.

Simplifions le contenu d'un certificat :

- `adresse : https://qkzk.xyz`, <—– nom du site
- `délivré par : Let's Encrypt` <—— une autorité
- `date de validité : du 30 mars 2020 au 28 juin 2020`
- `empreinte : FF 45 67 FA...` <—— vous permet de vérifier ce certificat auprès de Let's Encrypt

4.4. *HTTPS EveryWhere.*

À l'initiative de Google et des acteurs des réseaux décentralisés TOR, de nombreux acteurs du web militent pour que tous les sites passent en HTTPS. En 2018, plus de 50% des sites majeurs l'avaient fait

4.5. *HTTPS sur l'intégralité du site.*

Il ne suffit pas de sécuriser la phase de login ou le tunnel de paiement. Il faut aussi s'assurer que toutes les ressources sont sécurisées. Cela revient à s'assurer que les liens extérieurs pointent tous vers des pages en HTTPS.

4.6.*Implémentation.*

Tous les navigateurs modernes sur toutes les plateformes supportent HTTPS. Néanmoins, vous devez avoir confiance en son implémentation. Voilà pourquoi les mises à jour du navigateur sont parmi les plus importantes à effectuer.

4.7.*Le cadenas.*

Le cadenas présent à gauche de la barre d'adresse présente aussi des informations sur les cookies (eux-mêmes sécurisés) et parfois les autorisations (en tout cas pour chrome) données au site. Il est toujours possible d'effacer directement un cookie depuis la barre d'adresse. De même vous pouvez révoquer une autorisation.

4.8.*SSL et TLS.*

Les certificats sont communément appelés "certificats SSL" mais ce protocole a été totalement remplacé par TLS depuis des années.

En pratique, les certificats sont indépendants des protocoles.

Un site hors d'age qui renouvelle régulièrement ses certificats peut toujours utiliser SSL.

4.9.*HTTPS = sécurité ?.*

Oui et non. La communication est sécurisée et personne ne peut la lire. Mais si le site marchand ou la personne sur whatsapp / snapchat… fait n'importe quoi avec vos données…

Souvenez-vous de SONY : HTTPS dès 2011… mais les mots de passe du PlayStation Network étaient enregistrés en clair : 77 millions de comptes compromis lors d'un piratage et environ 3 milliards de $ de perte…

Vidéo à voir :

https://www.youtube.com/watch?v=7W7WPMX7arI

Source : https://qkzk.xyz/docs/nsi/cours_terminale/architecture/securisation/https/