Arthur Kalinowska, 2TL1

ADMIN I :
Synthèse sur TLS/SSL

Date : 11/01/2023 - Année académique 2022-2023

CHAPITRE 1 : INTRODUCTION

Lorsqu'Internet est devenu disponible au public et que l'électronique a connu un essor dans le monde, de nombreuses questions se sont posées vis-à-vis de la sécurité sur le web ainsi que la fiabilité des transactions bancaires par Internet et par carte bancaire.

Nombreux étaient les sceptiques qui exprimaient leur inquiétude par rapport aux cyberattaques et au vol de données en ligne. Fort heureusement, aujourd'hui le protocole SSL/TLS garantit une sécurité sur les sites web.

Mais quels sont ces protocoles ? Comment fonctionnent-ils ? Sont-ils différents et quels sont leurs liens avec les protocoles HTTP et HTTPS ? C'est ce dont il va être question dans cette synthèse sur les protocoles SSL et TLS.

CHAPITRE 2 : LE PROTOCOLE SSL

- **Principe de SSL [1][2][3][4][5][8][9][14]**

  Le protocole SSL (Secure Sockets Layer ou Couche de Sockets Sécurisés) est un protocole créé par Netscape dans les années 90 pour garantir la sécurité sur le web.

  Plus en détails, il a été conçu pour crypter et authentifier les données envoyées entre une application et un serveur web de sorte à empêcher tout vol de données de la part d'une entité malveillante.

  Ce standard est aujourd'hui appliqué par tous les sites actuels via des certificats SSL qui garantissent :

  1) Un haut degré de confidentialité grâce à un chiffrement des données transmises qui permet d'empêcher tout vol de données sensibles comme :
     - Des numéros de carte bancaire.
     - Des mots de passe de comptes.
     - Des identifiants de carte d'identité.
     - Des informations personnelles.
  2) L'authentification des appareils qui envoient et reçoivent les données transmises qui permet d'assurer que ceux ci ne sont pas malveillants.
  3) L'intégrité des données qui permet de dire que les données ont été envoyées et reçues sans aucune modification ou altération malveillante de la part d'un tiers.

  Ces certificats sont apparus en 1995 et font toujours partie de tous les sites webs actuels utilisant l'HTTPS. Ils sont très importants pour garantir la sécurité sur Internet.

  Remarque : Aujourd'hui lorsqu'on parle de certificats SSL ou du protocole SSL, on désigne en fait les certificats TLS et le protocole TLS (⇒ Voir chapitre 3 sur le protocole TLS pour mieux comprendre et où les certificats sont expliqués plus en détails).

- **Historique [1][9]**

  Au cours de son développement, plusieurs versions de SSL ont vu le jour :

  - 1994 : SSL 1.0

    ⇒ Pure spécification théorique développée par Netscape mais qui n'a jamais été réellement mise en œuvre.

  - Février 1995 : SSL 2.0

    ⇒ Première version réellement utilisée de SSL.

    ⇒ Sécurisation des données via un échange de clés de chiffrement (⇒Voir chapitre 3 où les techniques sont expliquées plus en détails).

  - Novembre 1996 : SSL 3.0

    ⇒ Apport de l'authentification serveur-client.

    ⇒ Dernière version du protocole SSL qui inspirera son successeur TLS (voir chapitre suivant).

- **Les avantages et les limites du protocole SSL [1][2][3][4][5][8]**

  Le protocole SSL possède de nombreux avantages mais aussi de nombreuses limites ;

  - <u>Avantages :</u>

    ⇒ Transparence pour l'utilisateur : Un utilisateur d'un navigateur web envoie des données chiffrées sans qu'il n'ait à les manipuler pour se connecter à un site de e-commerce sécurisé par SSL.

    ⇒ Début des principes de sécurité du web : Bien qu'il soit devenu obsolète aujourd'hui, le protocole SSL a posé les bases de la sécurité sur Internet et beaucoup contribué à son essor.

    ⇒ Les certificats SSL : Voir chapitre 3.

- Limites :

  ⇒ Failles de sécurité : Bien qu'il permette une sécurité sur le web, le protocole SSL possède de nombreuses failles.

  ⇒ Pour un exemple détaillé. Voir Chapitre 4.

- **Les améliorations [4][9][15][16]**

  Face aux limites du protocole SSL lors de son développement, plusieurs améliorations ont été mises en places :

  - Nouvelles versions de SSL

    ⇒ SSL 3.0 : Version la plus améliorée du protocole SSL, elle permet de faire face à de nombreuses failles de sécurité.

  - TLS

    ⇒ Successeur du protocole SSL devenu obsolète à cause de nombreuses failles de sécurité. Son nom est surtout dû au rachat du protocole par l'IETF (⇒Voir chapitre 3 - Le protocole TLS).

- **Le déclin [3][4]**

  Petit à petit, le protocole SSL connut un déclin, que ce soit par…

  - Son remplacement par le protocole TLS.
  - Les nombreuses failles de sécurité de la dernière version de SSL 3.0 comme POODLE (⇒ voir plus haut).
  - Les actions de Google pénalisant les sites web non sécurisés via des versions récentes de TLS.

  …jusqu'à ce qu'il finisse par être déclaré obsolète.

CHAPITRE 3 : LE PROTOCOLE TLS

- **Principe de TLS [1][2][3][4][5][8][14]**

  Le protocole TLS (Transport Layer Security ou Sécurité de la Couche Transport) est une amélioration du protocole SSL.

  Au niveau technique, il fonctionne de la même manière que SSL et lui ressemble même fortement.

  Son changement de nom vient surtout du fait que vers 1999, l'organisme de l'IETF (Internet Engineering Task Force) a poursuivi le développement de SSL tandis que Netscape n'y était tout simplement plus impliqué. Il s'agit juste d'un changement de priorité.

  En raison de leur ressemblance très forte, il est commun d'utiliser les deux termes de manière confuse. En effet, bien que SSL ne soit plus à jour depuis 1996 (Dernière version : SSL 3.0), beaucoup de personnes utilisent le terme "SSL/TLS" ou désignent le protocole TLS par "SSL" car le nom est resté dans la mémoire de tous.

- **Historique [1][9]**

  Voici l'évolution des versions du protocole TLS au fil du temps :

  - Janvier 1999 : TLS v1.0

    ⇒ Première version du protocole TLS développée par l'IETF.

  - Avril 2006 : TLS v1.1

    ⇒ Version améliorée de la v1.0 .

  - 2008 : TLS v1.2

    ⇒ Version améliorée de la v1.1. Plus robuste et plus performante, c'est encore la norme la plus utilisée à ce jour.

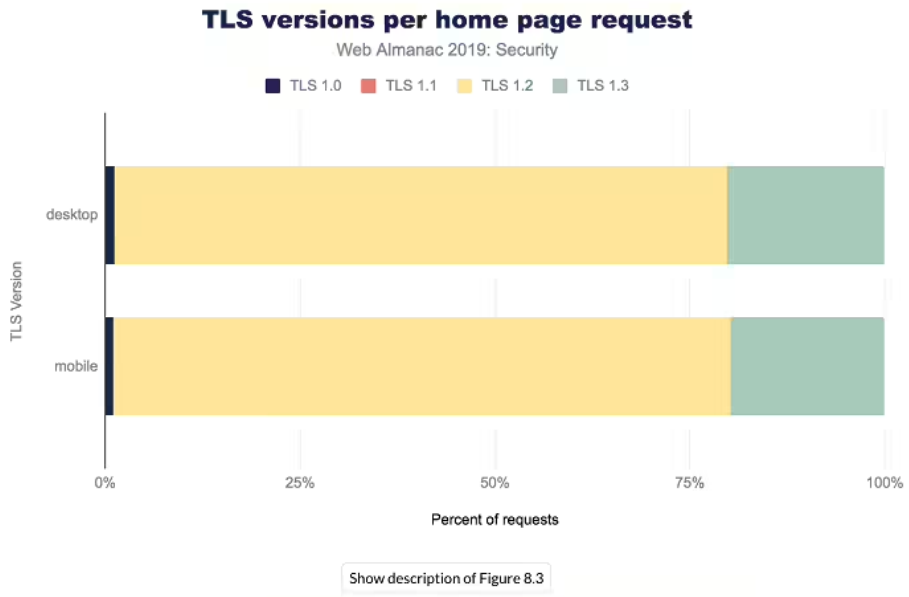


Figure 8.3. Usage of TLS protocol versions for home page requests only.

⇒ Sur ce schéma, on peut voir qu'en 2019, environ 80% des pages web sur ordinateur et mobile utilisent la norme TLS 1.2 .

- Août 2018 : TLS v1.3

  ⇒ Version la plus récente et la plus performante de TLS, elle permet de faire face aux récentes failles découvertes depuis 2008.

- **Les techniques du protocole TLS/SSL [6][9][11][12][13][15][16]**

  Ces techniques s'appliquent aussi au protocole SSL.

  - Le cryptage ou chiffrement

    Sur base d'un algorithme de chiffrement/cryptage (comme MD5H), on va modifier un message en clair pour le rendre chiffré/crypté.

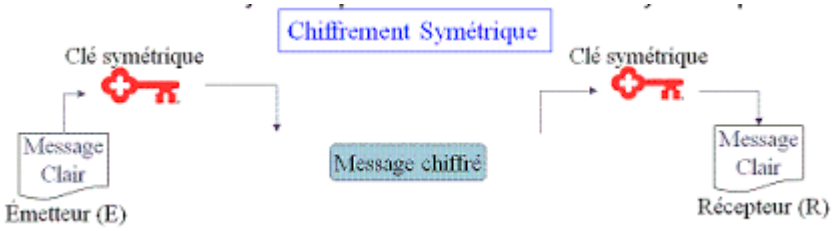
    Il existe deux types de techniques de chiffrement/cryptage :

    - Symétrique :

      ⇒ C'est une technique très utile quand elle est combinée avec d'autres algorithmes mais désastreuse si c'est la seule technique utilisée.

⇒ Se fait à l'aide d'une clé unique pour chiffrer et déchiffrer le message.

⇒ Exemple :



Dans cet exemple, on voit qu'un émetteur chiffre son message clair en message chiffré via une clé symétrique.

Ce message crypté est ensuite envoyé à un récepteur qui le déchiffre via la même clé symétrique.

⇒ Avantage : rapide à mettre en place.

⇒ Inconvénients :

1) La clé de chiffrement doit être connue du client et du serveur. Elle doit donc être transmise.
2) Cette transmission de clé ne peut pas être sécurisée.
3) Au plus il y a de destinataires, au plus il y a de clés symétriques ⇒ Redondance.

- Asymétrique :

Cette fois, on a deux clés :

Privée : C'est la clé principale et elle n'est jamais partagée. Elle est utilisée pour le déchiffrement de messages.

Publique : C'est la jumelle de la privée et est partageable. Elle est utilisée pour le chiffrement de messages.

⇒ Remarque : Pour les signatures des certificats SSL/TLS, c'est le procédé inverse : les données sont chiffrées par la clé privée et déchiffrées par la publique.

⇒ Exemple :

Dans cet exemple, un émetteur chiffre un message clair message chiffré via une clé publique R.

Ce message est envoyé au récepteur qui le déchiffre avec sa propre clé privée.

- Les certificats SSL/TLS - Version simple

Les certificats SSL/TLS sont des fichiers contenant les données concernant la sécurisation d'un site web.

Un certificat contient :

- Une clé publique.
- La date de validité du certificat.
- La date d'expiration du certificat.
- L'algorithme de chiffrement.

Ainsi que deux notions importantes :

- L'empreinte du certificat :

⇒ De manière courte, via un algorithme de hachage tel que MD5H, on va crypter le certificat et mettre le résultat de l'algorithme dans le certificat.

⇒ Si le certificat est modifié par un tiers, alors on peut vite s'en rendre compte grâce à la comparaison des empreintes.

- La signature :

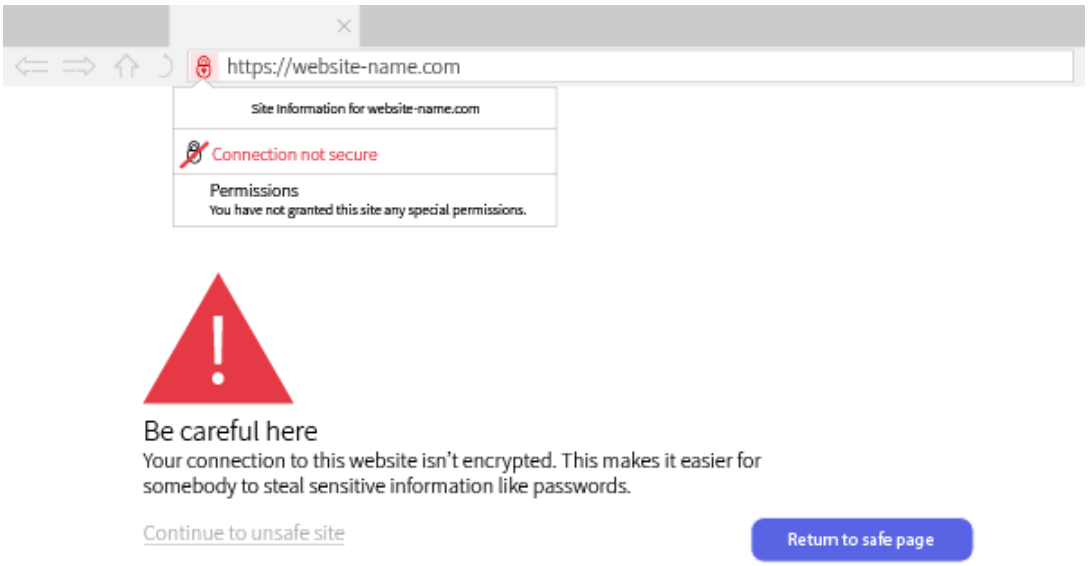
  ⇒ Elle permet de verrouiller l'empreinte d'un certificat et ainsi empêcher toute falsification malveillante.

  ⇒ Elle utilise le principe de chiffrage asymétrique sauf que la clé privée d'un serveur chiffre les données d'un message. Ensuite ce message est envoyé à des sites web qui regardent leurs certificats disponibles et vérifient avec leur clé publique s' ils arrivent à déchiffrer le message. Si oui, alors il peut y avoir connexion et non dans le cas contraire.

L'empreinte et la signature, lorsqu'elles sont combinées, sécurisent énormément les échanges de données sur Internet et maximisent l'authentification et l'intégrité des données.

Il y a plusieurs types de certificats qui peuvent s'appliquer à un ou plusieurs sites web classés en 3 types : générique, à domaine unique et à domaines multiples. Ils sont aussi classés selon le niveau de validation : validation de domaine, validation d'organisation et validation étendue.

⇒ Exemple de page web sans certificats SSL/TLS.

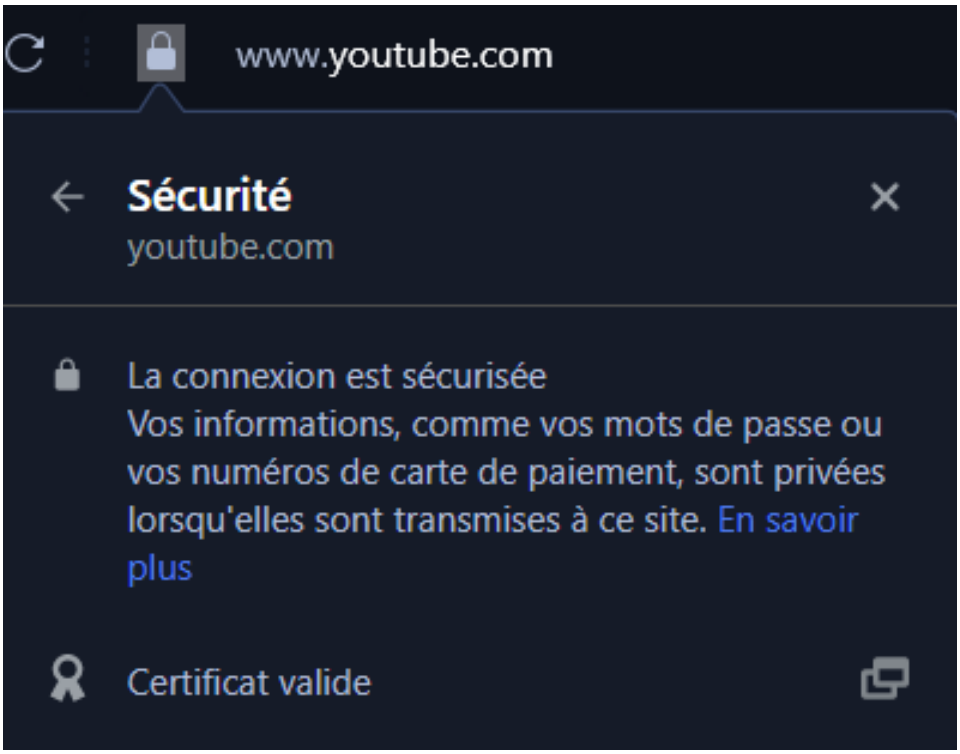


⇒Dans cet exemple, on peut voir que les pages web qui n'utilisent pas de certificats SSL/TLS ont un message d'alerte. Cela va même plus car Google a décidé de sanctionner les pages web qui ont peu voir aucune sécurité et à l'inverse, de récompenser les pages ayant une bonne sécurité ainsi que de bons certificats SSL/TLS.

⇒ Exemple de certificat SSL/TLS.

⇒ Sur cet exemple, on peut voir les différentes informations d'un certificat SSL/TLS telles que la durée de validité, l'empreinte du certificat ou encore l'émetteur et le récepteur.

⇒ Exemple de page web avec certificat SSL/TLS.

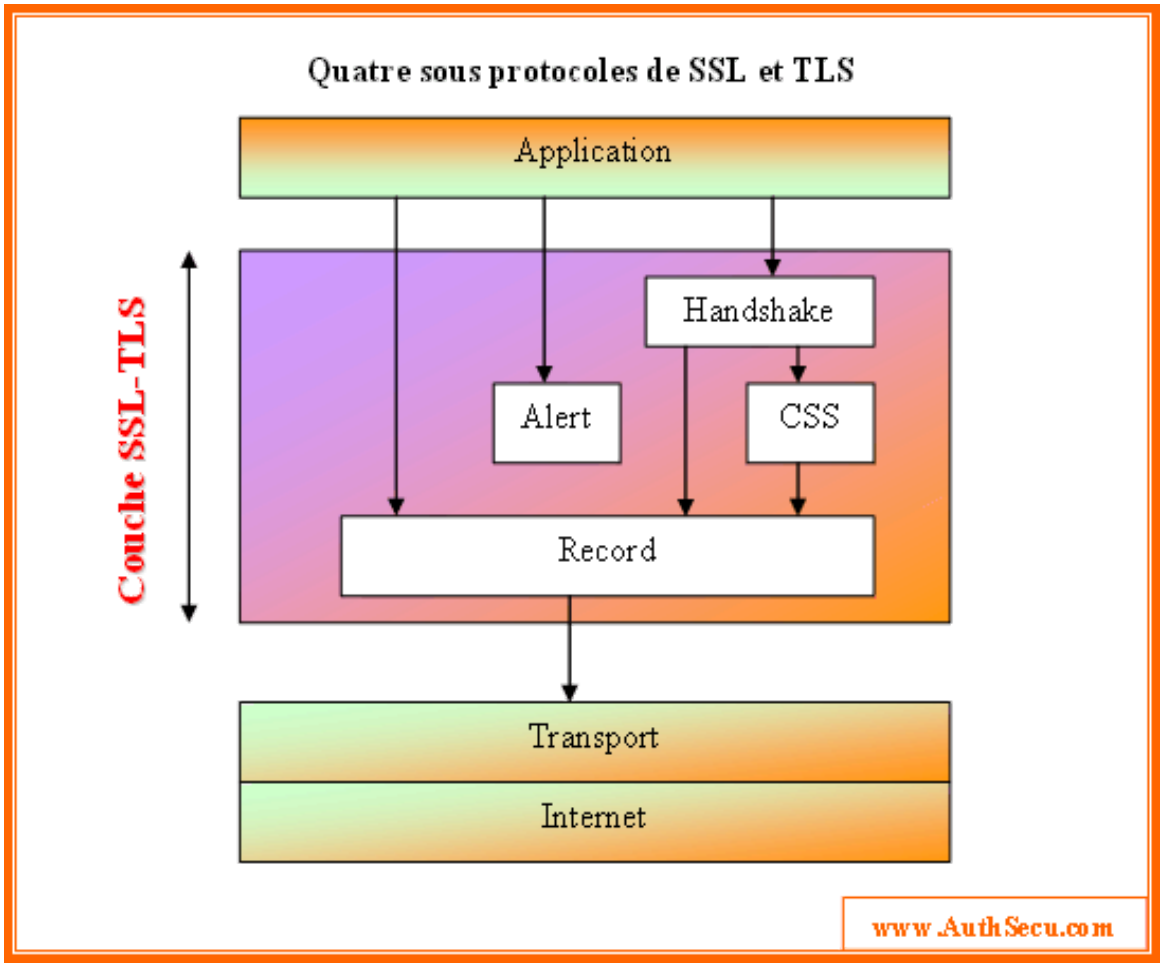


⇒ Dans cet exemple d'une page web sécurisé en HTTPS, on peut voir à côté de la barre d'adresse un cadenas. Ce cadenas symbolise l'utilisation de certificats SSL/TLS valides et donc de l'utilisation du protocole SSL/TLS.

- **Les sous protocoles de TLS/SSL [1][8][9]**

  Le protocole TLS/SSL fonctionne grâce au moyen de plusieurs sous protocoles ;

  ⇒ Organisation de ces sous protocoles au niveau des couches de TCP/IP :

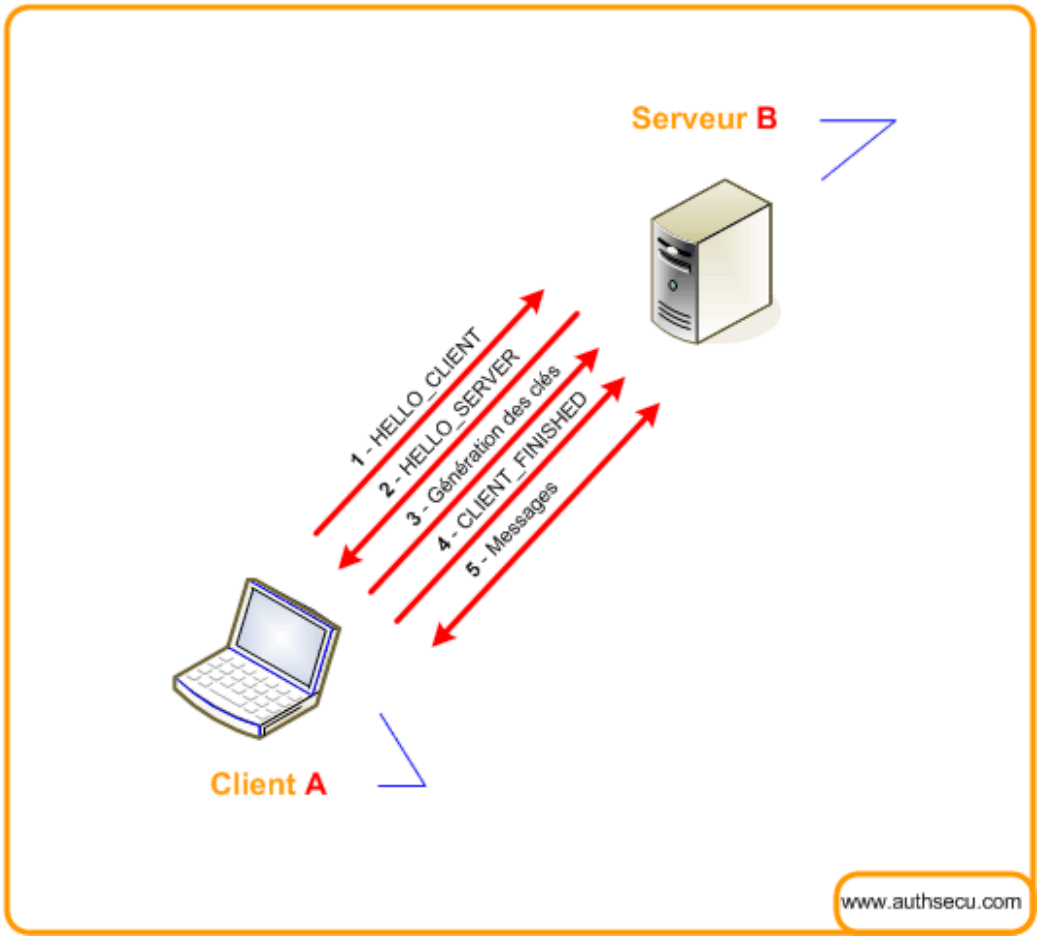


  ⇒ On peut voir ici que le protocole TLS travaille entre la couche Transport et Application (En lien avec HTTP et HTTPS ⇒ Voir chapitre 4).

  - Handshake Protocol

    ⇒ Il s'agit d'une phase de négociation client-serveur pour choisir les algorithmes de chiffrements et les clés de chiffrements symétriques.

⇒ Exemple type d'échange :



1) Message du client : "HELLO_CLIENT"

   ⇒ Contient plusieurs informations importantes telles que la liste des algorithmes de compression, d'échange de clé et de chiffrement que le client supporte ainsi que la version la plus haute de SSL qu'il puisse utiliser.

2) Réponse du serveur : "HELLO_SERVER"

   ⇒ Contient les méthodes qui seront utilisées ainsi que la version la plus haute de SSL que le client puisse utiliser.

   ⇒ Authentification auprès du serveur.

3) Génération des clés de chiffrement symétrique

   ⇒ Principes de fonctionnement des chiffrements symétriques et asymétriques.

4) Message du client : “CLIENT_FINISHED”

   ⇒Message chiffré et signé.

5) Message du serveur : “SERVER_FINISHED” + échange de données

   ⇒ Le message du serveur n’est pas visible car dénié par le sous protocole CCS.

   ⇒ Les échanges de données se font maintenant de manière chiffrée et elles sont sécurisées.

- Change Cipher Spec Protocol (CCS)

  ⇒ Protocole comportement qu’un seul et unique message du même nom que celui-ci.

  ⇒Sert à indiquer au Record Protocol que la négociation des algorithmes de chiffrement est finie et que ceux-ci ont été choisis et mis en place.

- Alarm Protocol

  ⇒ Sert à générer des messages d’alertes concernant les erreurs envoyées entre un client et un serveur.

  ⇒ 2 types : Fatal ou Warning (Si Fatal, alors la connexion est abandonnée).

⇒ Exemples d'erreurs de type Fatal :

bad_record_mac: réception d'un MAC erroné

decompression_failure: les données appliquées à la fonction de compression sont invalides

handshake_failure: impossibilité de négocier les bons paramètres

illegal_parameter: un paramètre échangé au cours du protocole Handshake ne correspond pas avec les autres paramètres

unexpected_message: message non reconnu.

⇒ Exemples d'erreurs de type Warning :

bad_certificate: le certificat n'est pas bon

certificate_expired: certificat périmé

certificat_revoked: certificat révoqué

certificat_unknown: certificat invalide pour des raisons précisés au dessus

close_notify: la fin d'une connexion

no_certificate: réponse négative à une demande de certificat

unsupported_certificate: le certificat reçu n'est pas reconnu

- Record Protocol

⇒ Intervient après le Change Cipher Spec Protocol.

⇒ Garantit l'intégrité des données ainsi que leur confidentialité.

- **Les limites et améliorations de TLS [8][10][15][16]**

  Le protocole TLS a lui aussi connu des limites ainsi que des améliorations ;

  - Limites

    Prenons un exemple de la vie quotidienne, un personnage du nom de User1 va dans un bar pour regarder un match de foot et veut se connecter au wifi public du bar du nom de BAR_PUBLIC.

    Ce qu'il ne sait pas, c'est que dans le bar, il y a un pirate informatique du nom de Evil1 qui a réussi à remplacer le vrai réseau wifi par un faux du même nom.

    Plusieurs cas sont alors possibles :

    1) Le réseau web fonctionne sous le protocole HTTP et n'a donc aucune protection :

       ⇒ Tout échange de données apparaît en clair.

       ⇒ Evil1 à donc accès à tout type d'infos et peut faire ce qu'il veut.

    2) Le réseau fonctionne sous le protocole HTTPS et utilise un chiffrement symétrique :

       ⇒ Rappel : pour un chiffrement symétrique, on utilise une clé de chiffrement.

       ⇒ Problème : cette clé doit être envoyée au destinataire pour qu'il puisse déchiffrer les données reçues.

       ⇒ Si Evil1 est déjà sur le réseau, il va aussi obtenir cette clé de chiffrement symétrique et donc il aura aussi accès aux données malgré le chiffrement.

3) Le réseau fonctionne sous le protocole HTTPS et utilise un chiffrement asymétrique :

⇒ Rappel : Dans le chiffrement asymétrique, on utilise 2 clés ; une clé privée pour déchiffrer des données et une clé publique pour crypter des données.

⇒ Le serveur reçoit une demande de connexion du client et lui envoie une clé publique. User1, avec cette clé, va crypter ses données et les envoyer au serveur.

⇒ Bien que Evil1 ait reçu eux aussi la clé publique, il ne peut rien faire pour décrypter les données.

⇒ Problème : Malgré cela, ce n'est pas suffisant pour garantir une connexion sécurisée.

⇒ Evil1 peut contourner le chiffrement en se faisant passer pour un relais entre serveur et client (comme un proxy) et ainsi duper et le serveur et le User1 en générant sa clé privée et sa clé publique qu'il envoie à User1 et ainsi de suite.

⇒ Il y a aussi eu de nombreuses failles de sécurité telles que POODLE qui permettait de déchiffrer les données chiffrées facilement.

- <u>Améliorations</u>

⇒ Les certificats SSL/TLS : Voir la partie explicative sur les certificats SSL/TLS.

⇒ TLS v3.0 : Sortie en 2018, elle a permis de corriger de nombreuses failles de sécurité ainsi que de renforcer l'authentification et l'intégrité des données partagées lors d'une connexion en ligne.

- **Le futur de TLS [7][10]**

Avec TLS 1.3, la sécurité sur le web a été largement renforcée par rapport aux versions précédentes. Néanmoins, il y a aura toujours des problèmes qui apparaîtront tôt ou tard et nul doute que des mises à jour pourront être apportées pour les corriger.

CHAPITRE 4 : LIENS AVEC HTTP ET HTTPS

- **HTTP [2][4]**

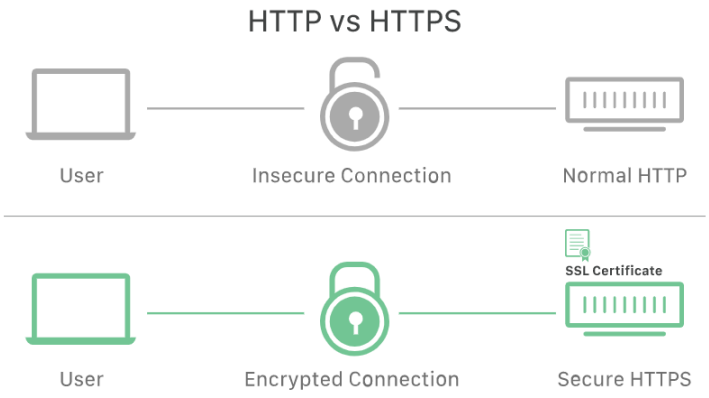
  Connu comme le fondement du World Wide Web. HTTP (Hypertext Transfer Protocol) est un protocole de la couche application utilisé pour charger des pages sur Internet à l'aide de liens hypertextes et transférer des informations lors de connexions réseau entre des appareils.

  Son principal souci est qu'il ne dispose d'aucune sécurité. Les piratages et vol de données y sont alors très faciles.

- **HTTPS [4][15][16]**

  Comme HTTP avec en plus une sécurisation des échanges de données (S) via un chiffrement symétrique ou asymétrique.

  Il est rapidement devenu le protocole par excellence sur le web. En utilisant les techniques précédemment vu, le protocole est largement au dessus de HTTP en termes de sécurité, d'intégrité de données et d'authentification.



CHAPITRE 5 : CONCLUSION

En conclusion, le protocole SSL et son amélioration TLS ont permis d'améliorer la sécurité, l'intégrité des données et l'authentification sur les réseaux webs au travers de diverses techniques utilisées par HTTPS comme les chiffrements symétriques et asymétriques ainsi que l'utilisation de certificats SSL/TLS.

Ces deux protocoles ont permis au protocole HTTP de devenir HTTPS, causant en conséquence le déclin et bientôt la fin des sites utilisant HTTP. Un mal pour un bien.

SOURCES

1) Wikipédia, “Transport Layer Security”, 1/09/2022, Lien URL, 12-01-2023
2) Digicert, “Que sont SSL, TLS et HTTPS”, N.A. , Lien URL, 12-01-2023
3) Kinsta, “Comment fonctionne le SSL et pourquoi c’est important”, 28/06/2021, Lien URL, 12-01-2023
4) CloudFlare, “Qu’est ce que le SSL ? | Définition du SSL”, N.A. , Lien URL, 12-01-2023
5) GlobalSign, “Qu’est ce que le SSL”, N.A. , Lien URL, 12-01-2023
6) CertEurope, “Tout savoir sur les certificats SSL ou TLS”, N.A. , Lien URL, 12-01-2023
7) Olivier Levillain, “TLS : passé, présent et futur ?”, 1/06/2017, Lien URL, 12-01-2023
8) S.LAZAAR ENSA, “Le protocole SSL Secure Socket Layer”, 2006, Lien URL , 12-01-2023
9) FRAMEIP.COM - Vincent LIMORTE, François VERRY et _SebF, “Protocole SSL et TLS”, 2017, Lien URL, 12-01-2023
10) Fasterize, “TLS 1.3, la nouvelle version la plus performante du web sécurisé”, 8/11/2018, Lien URL, 12-01-2023
11) Certificat.fr, “Qu’est ce qu’un certificat SSL - L’histoire du SSL en bref”, N.A. , Lien URL, 12-01-2023
12) Cookie Connecté, “Comprendre le chiffrement SSL/TLS avec des emojis (et le https)”, 20/02/2018, Lien URL, 12-01-2023
13) e-monsite.com, “Qu’est-ce-qu’un certificat SSL et à quoi sert-il ?”, 18/01/2021, Lien URL, 12-01-2023
14) the roadmap, “SSH vs TLS vs SSL”, 25/11/2021, Lien URL, 12-01-2023
15) CodeRocks & apprendre, “Comprendre HTTPS et le chiffrement SSL TLS en animation 3D”, 26/05/2022, Lien URL, 12-01-2023
16) CodeRocks & apprendre, “Comprendre le certificat SSL TLS dans une connexion HTTPS (Partie2)”, 2/07/2022, Lien URL, 12-01-2023