Introduction SSL avec la transaction d'échantillon et l'échange de paquet

Contenu

[Introduction](#)
[Aperçu d'enregistrement SSL](#)
[Format d'enregistrement](#)
[Type d'enregistrement](#)
[Enregistrez la version](#)
[Longueur de l'enregistrement](#)
[Types d'enregistrements](#)
[Enregistrements de prise de contact](#)
[Enregistrements de spécification de chiffrement de modification](#)
[Enregistrements d'alerte](#)
[Enregistrement de données des applications](#)
[Transaction d'échantillon](#)
[Bonjour l'échange](#)
[Échange de client](#)
[Modification de chiffrement](#)
[Informations connexes](#)

Introduction

Ce document décrit les concepts de base du protocole de Secure Sockets Layer (SSL) et fournit une transaction d'échantillon et une capture de paquet.

Aperçu d'enregistrement SSL

L'unité de base des données dans le SSL est un enregistrement. Chaque enregistrement se compose d'une en-tête record de cinq-octet, suivie des données.

Format d'enregistrement

- **Type** : uint8 - valeurs répertoriées ci-dessous
- **Version** : uint16
- **Longueur** : uint16

Type	Version		Longueur	
T	VH	VL	Main gauche	LL

Type d'enregistrement

Il y a quatre types d'enregistrement dans le SSL :

- **Prise de contact** (22, 0x16)
- **Spécification de chiffrement de modification** (20, 0x14)
- **Alerte** (21, 0x15)
- **Données des applications** (23, 0x17)

Enregistrez la version

La version record est une valeur 16-byte et est formatée dans la commande de réseau.

> Remarque: Pour la version 3 (SSLv3) SSL, la version est 0x0300. Pour la version 1 (TLSv1) de Transport Layer Security, la version est 0x0301. L'appliance de sécurité adaptable Cisco (ASA) ne prend en charge pas la version de version SSL 2 (SSLv2), qui utilise la version 0x0002, ou tout du TLS plus grand que TLSv1.

Longueur de l'enregistrement

La longueur de l'enregistrement est une valeur 16-byte et est formatée dans la commande de réseau.

Dans la théorie, ceci signifie qu'un enregistrement simple peut être jusqu'à 65,535 (2^16 -1) octets dans la longueur. Le TLSv1 RFC2246 déclare que la longueur maximale est de 16,383 (2^14 -1) octets. Des Produits de Microsoft (Microsoft Internet Explorer et Internet Information Services) sont connus pour dépasser ces limites.

Types d'enregistrements

Cette section décrit les quatre types d'enregistrements SSL.

Enregistrements de prise de contact

Les enregistrements de prise de contact contiennent un ensemble de messages qui sont prise de contact utilisée. Ce sont les messages et leurs valeurs :

- **Bonjour demande** (0, 0x00)
- **Client bonjour** (1, 0x01)
- **Serveur bonjour** (2, 0x02)
- **Certificat** (11, 0x0B)
- **Échange de clé de serveur** (12, 0x0C)
- **Demande de certificat** (13, 0x0D)
- **Serveur bonjour fait** (14, 0x0E)
- **Le certificat vérifient** (15, 0x0F)

- **Key Exchange de client** (16, 0x10)
- **De finition** (20, 0x14)

Dans le cas simple, des enregistrements de prise de contact ne sont pas chiffrés. Cependant, un enregistrement de prise de contact qui contient un message de finition est toujours chiffré, car il se produit toujours après qu'un enregistrement de la spécification de chiffrement de modification (CCS).

Enregistrements de spécification de chiffrement de modification

Des enregistrements CCS sont utilisés afin d'indiquer un changement des chiffrements crpytographic. Juste après que l'enregistrement CCS, toutes les données est chiffré avec le nouveau chiffrement. Les enregistrements CCS pourraient ou ne pourraient pas être chiffrés ; dans une connexion simple avec une prise de contact simple, l'enregistrement CCS n'est pas chiffré.

Enregistrements d'alerte

Des enregistrements d'alerte sont utilisés afin d'indiquer au pair qu'une condition s'est produite. Quelques alertes sont des avertissements, alors que d'autres sont mortelles et font échouer la connexion. Les alertes pourraient ou ne pourraient pas être chiffrées, et pourraient se produire pendant une prise de contact ou pendant le transfert des données. Il y a deux types d'alertes :

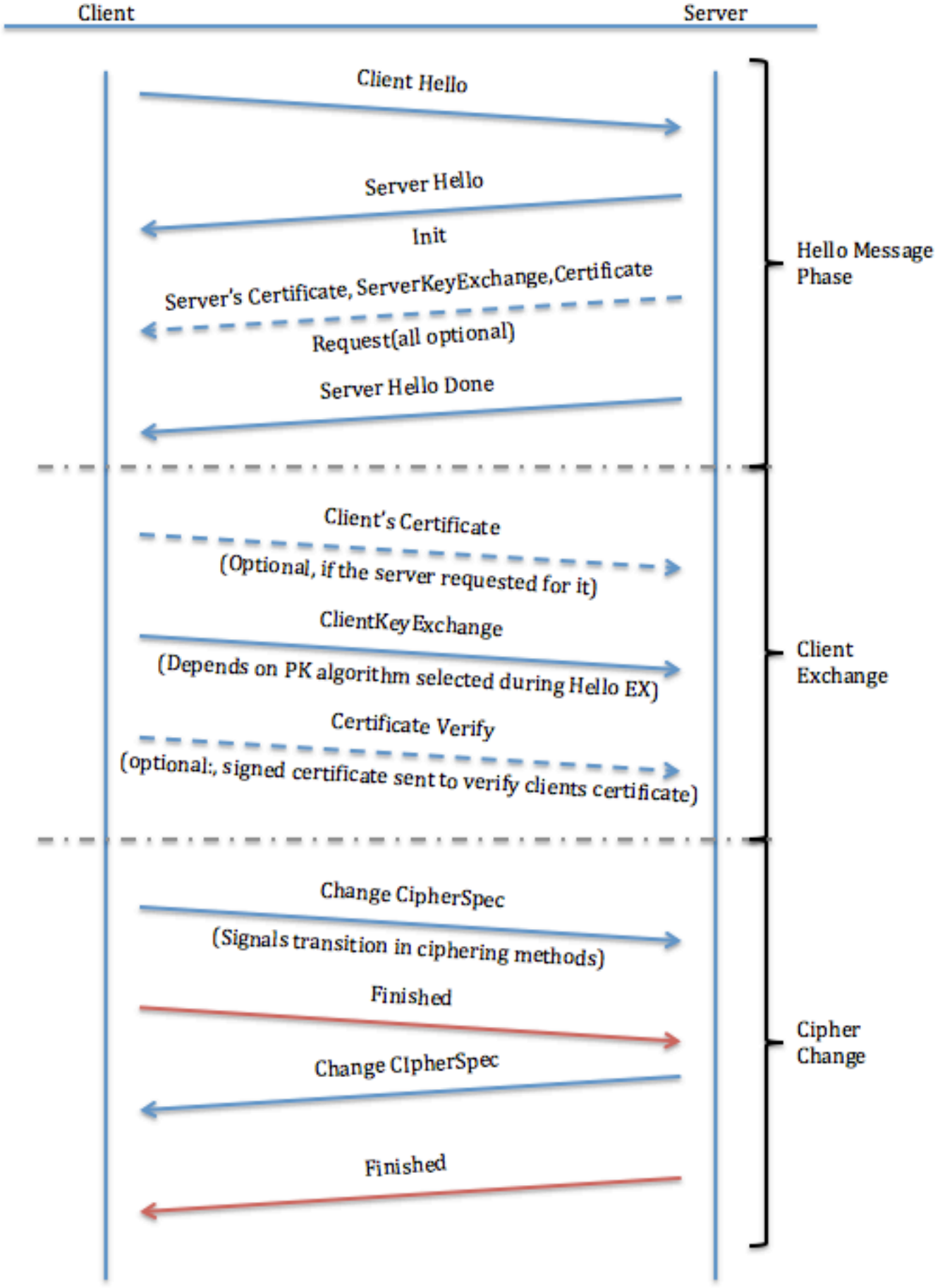
- **Alertes de fermeture :** La connexion entre le client et le serveur doit être correctement fermée afin d'éviter n'importe quel genre d'attaques de troncation. On envoie un message de **close_notify** qui indique au destinataire que l'expéditeur n'enverra plus des messages sur cette connexion.
- **Alertes d'erreur :** Quand une erreur est détectée, l'interlocuteur le détectant envoie un message à l'autre interlocuteur. Sur la transmission ou la réception d'un message d'alerte mortel, les deux interlocuteurs ferment immédiatement la connexion. Quelques exemples des alertes d'erreur sont :
  - **unexpected_message** (mortel)
  - **decompression_failure**
  - **handshake_failure**

Enregistrement de données des applications

Ces enregistrements contiennent les données des applications réelles. Ces messages sont diffusés par la couche record et sont fragmentés, compressés, et chiffrés, basé sur l'état en cours de connexion.

Transaction d'échantillon

Cette section décrit une transaction d'échantillon entre le client et serveur.



Bonjour l'échange

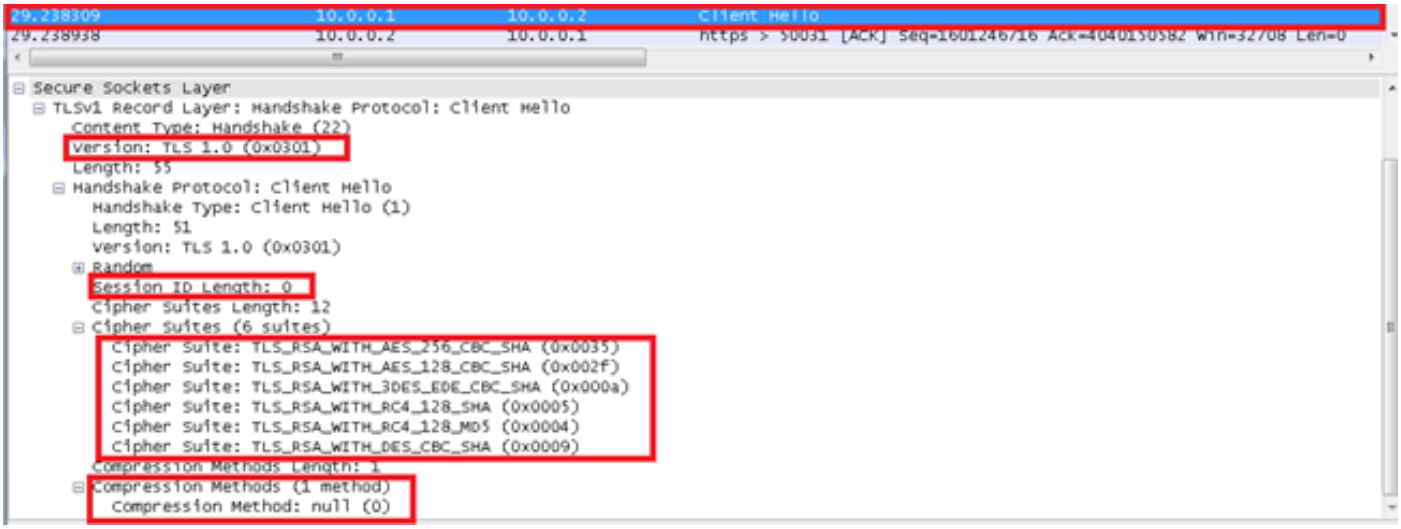
Quand un client et serveur SSL commencent au communitcate, ils conviennent sur une version de protocole, les algorithmes de chiffrement choisis, s'authentifient sur option, et emploient des techniques publiques de chiffrement à clé afin de générer des secrets partagés. Ces processus sont exécutés dans le protocole handshake. En résumé, le client envoie un message Hello de client au serveur, qui doit répondre avec un message Hello de serveur ou une erreur fatale se produit et la connexion échoue. Le client bonjour et le serveur bonjour sont utilisés pour établir des capacités d'amélioration de la sécurité entre le client et serveur.

Client bonjour

Le client bonjour envoie ces attributs au serveur :

- **Version de Protocol :** La version du protocole SSL par lequel le client souhaite communiquer pendant cette session.
- **ID de session :** L'ID d'une session que le client souhaite l'utiliser pour cette connexion. Dans le premier client bonjour de l'échange, l'ID de session est vide (référez-vous à la copie d'écran de capture de paquet après la note ci-dessous).
- **Suite de chiffrement :** Ceci est passé du client au serveur dans le message Hello de client. Il contient les combinaisons des algorithmes de chiffrement pris en charge par le client par ordre préférence du client (premier choix d'abord). Chaque suite de chiffrement définit un algorithme principal d'échange et une spécification de chiffrement. Le serveur sélectionne une suite de chiffrement ou, si aucun choix acceptable n'est présenté, renvoie une alerte de panne de prise de contact et ferme la connexion.
- **Méthode de compression :** Inclut une liste d'algorithmes de compression pris en charge par le client. Si le serveur ne prend en charge aucune méthode envoyée par le client, la connexion échoue. La méthode de compression peut également être nulle.

  Remarque: L'adresse IP du serveur dans les captures est 10.0.0.2 et l'adresse IP de client est 10.0.0.1.



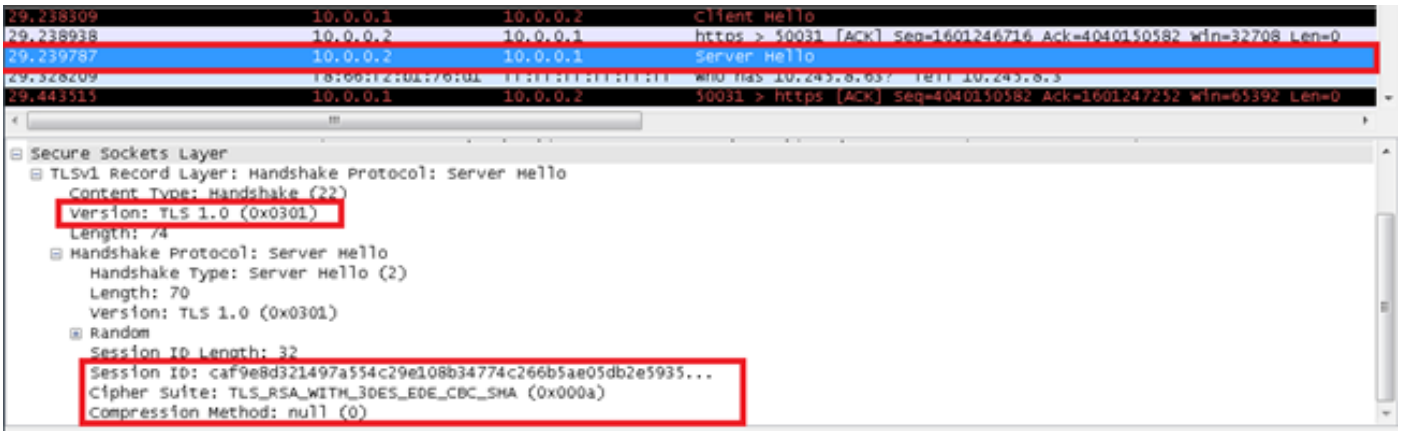
Serveur bonjour

Le serveur renvoie ces attributs au client :

- **Version de Protocol :** La version choisie du protocole SSL que le client prend en charge.
- **ID de session :** C'est l'identité de la session qui correspond à cette connexion. Si l'ID de

session envoyé par le client dans le client bonjour n'est pas vide, le serveur regarde dans le cache de session pour une correspondance. Si une correspondance est trouvée et le serveur est disposé à établir la nouvelle connexion utilisant l'état spécifié de session, le serveur répond avec la même valeur qui a été fournie par le client. Ceci indique une session reprise et dicte que les interlocuteurs doivent poursuivre directement aux messages de finition. Autrement, ce champ contient une valeur différente qui identifie la nouvelle session. Le serveur pourrait renvoyer un **session_id** vide afin d'indiquer que la session ne sera pas cachée, et ne peut pas donc être repris.

- **Suite de chiffrement :** Comme sélectionné par le serveur de la liste qui a été envoyée du client.
- **Méthode de compression :** Comme sélectionné par le serveur de la liste qui a été envoyée du client.
- **Demande de certificat :** Le serveur envoie au client une liste de tous les Certificats qui sont configurés là-dessus, et permet au client pour sélectionner qui le délivre un certificat veut l'utiliser pour l'authentification.

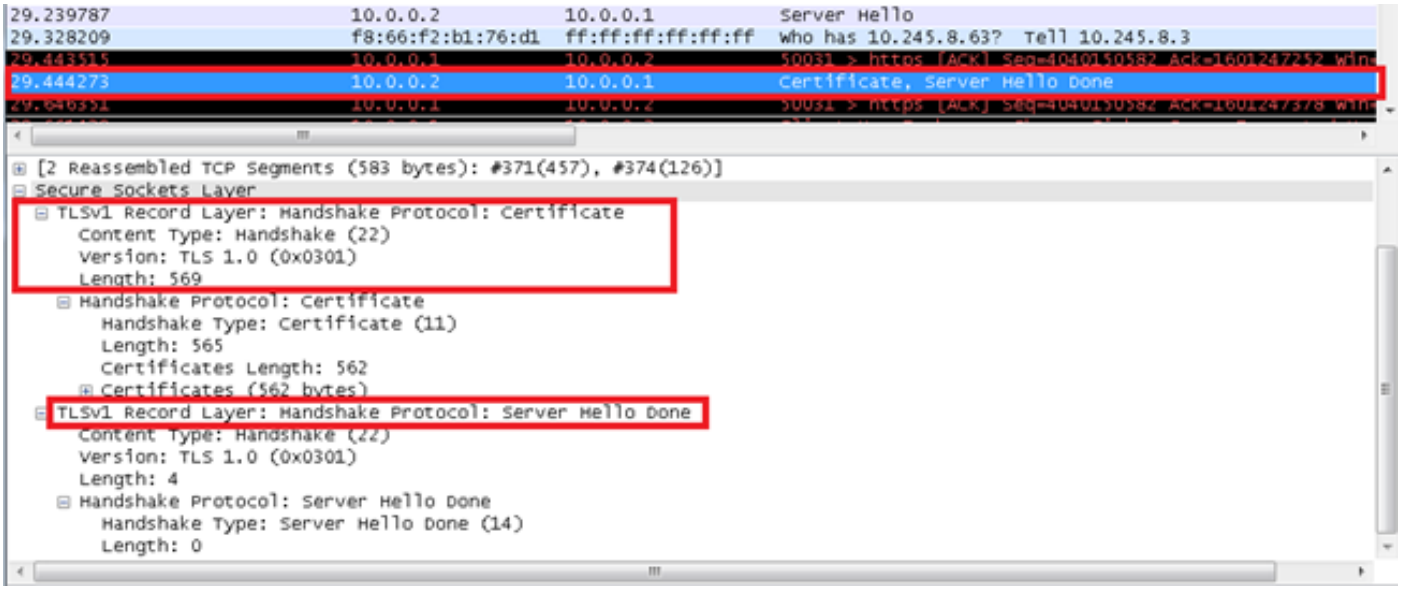


Pour des demandes de reprise de session SSL :

- Le serveur peut envoyer bonjour une demande au client aussi bien. C'est de rappeler seulement le client qu'il devrait commencer la renégociation avec une demande de client bonjour si commode. Le client ignore bonjour la demande du serveur si le processus de prise de contact est déjà en cours.
- Les messages de prise de contact ont plus de priorité au-dessus de la transmission des données des applications. La renégociation doit commencer en pas plus d'un ou deux fois le délai de transmission d'un message de données des applications de longueur maximale.

Serveur bonjour fait

Le message fait de serveur bonjour est envoyé par le serveur afin d'indiquer l'extrémité du serveur bonjour et des messages associés. Après qu'il envoie ce message, le serveur attend une réponse de client. Dès réception du message fait de serveur bonjour, le client vérifie que le serveur a fourni un certificat valide, s'il y a lieu, et des contrôles que les paramètres Hello de serveur sont acceptables.



Certificat de serveur, échange de clé de serveur, et demande de certificat (facultative)

- **Certificat de serveur :** Si le serveur doit être authentifié (qui est généralement le cas), le serveur envoie son certificat juste après le message Hello de serveur. Le type de certificat doit être approprié pour l'algorithme sélectionné d'échange de clé de suite de chiffrement, et est généralement un certificat X.509.v3.
- **Échange de clé de serveur :** Le message d'échange de clé de serveur est envoyé par le serveur s'il n'a aucun certificat. Si le Diffie ? Hellman que (CAD) des paramètres sont inclus avec le certificat de serveur, ce message n'est pas utilisé.
- **Demande de certificat :** Un serveur peut sur option demander un certificat du client, si approprié pour la suite sélectionnée de chiffrement.

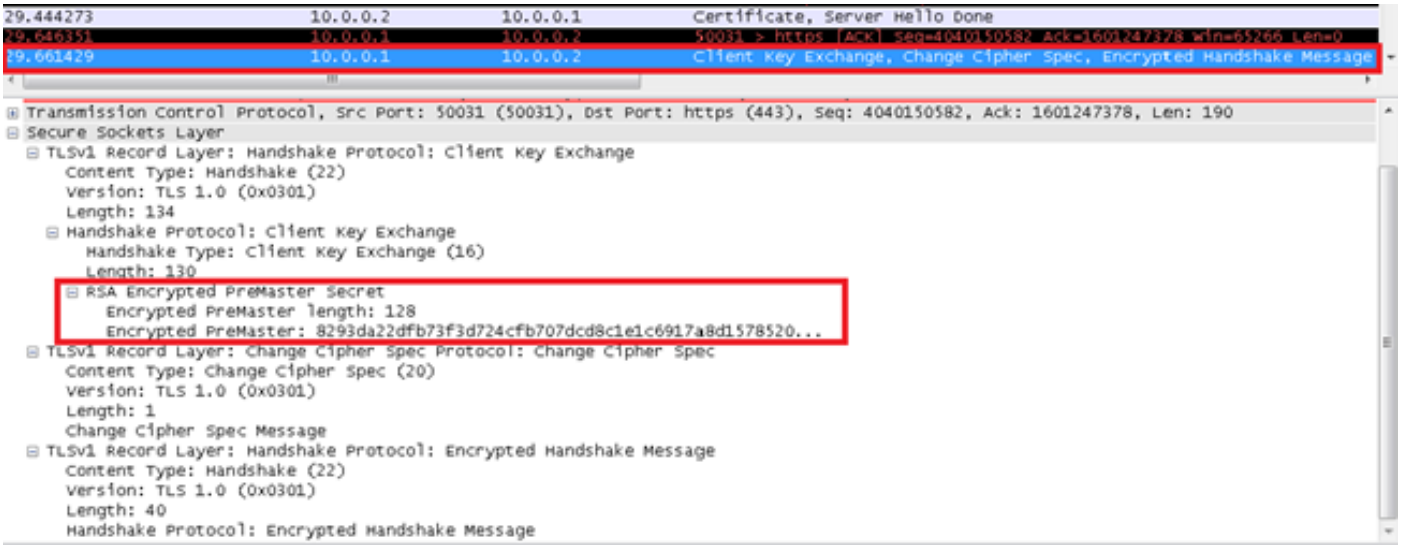
Échange de client

Certificat client (facultatif)

C'est le premier message que le client envoie après qu'il reçoive un message fait de serveur bonjour. Ce message est seulement envoyé si le serveur demande un certificat. Si aucun certificat approprié n'est disponible, le client envoie une alerte de **no_certificate** à la place. Cette alerte est seulement un avertissement ; cependant, le serveur pourrait répondre avec une alerte mortelle de panne de prise de contact si l'authentification client est exigée. Les Certificats CAD de client doivent apparier les paramètres CAD spécifiés par serveur.

Key Exchange de client

Le contenu de ce message dépend de l'algorithme de clé publique sélectionné entre le client bonjour et les messages Hello de serveur. Le client utilise une clé de premaster chiffrée par l'algorithme de Rivest-Shamir-Addleman (RSA) ou le CAD pour l'accord et l'authentification principaux. Quand la RSA est utilisée pour l'authentification de serveur et l'échange de clé, un **pre_master_secret** 48-byte est généré par le client, chiffré sous la clé publique de serveur, et envoyé au serveur. Le serveur emploie la clé privée afin de déchiffrer le **pre_master_secret**. Les deux interlocuteurs convertissent alors le **pre_master_secret** en **master_secret**.



Le certificat vérifient (facultatif)

Si le client envoie un certificat avec la capacité de signature, un certificat digital-signé vérifient le message est envoyé afin de vérifier explicitement le certificat.

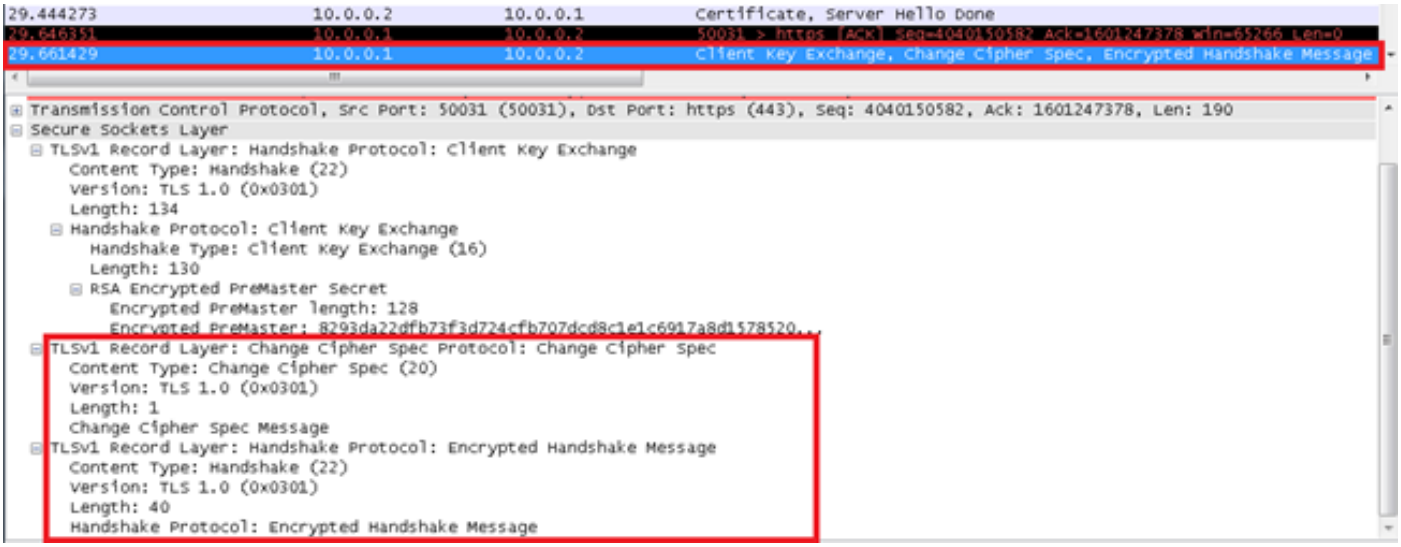
Modification de chiffrement

Messages de spécification de chiffrement de modification

Le message de spécification de chiffrement de modification est envoyé par le client, et le client copie la spécification en attente de chiffrement (le neuf) dans la spécification en cours de chiffrement (celle qui a été précédemment utilisée). Le protocole de spécification de chiffrement de modification existe des transitions de signal dans des stratégies de chiffrage. Le protocole se compose d'un message simple, qui est chiffré et compressé sous la spécification en cours de chiffrement (pas l'en suspens). Le message est envoyé par chacun des deux le client et serveur afin d'informer l'interlocuteur de réception que des enregistrements ultérieurs sont protégés sous la spécification et les clés récemment négociées de chiffrement. La réception de ce message fait copier le récepteur « a lu en attendant » l'état dans » l'état en cours « lu. Le client envoie un échange suivant de clé de prise de contact de message de spécification de chiffrement de modification et le certificat vérifient des messages (le cas échéant), et le serveur envoie un après qu'il traite avec succès le message d'échange principal qu'il a reçu du client. Quand une session précédente est reprise, le message de spécification de chiffrement de modification est envoyé après les messages Hello. Dans les captures, l'échange de client, le chiffrement de modification, et les messages de finition sont envoyés comme message simple du client.

Messages de finition

Un message de finition est toujours envoyé juste après qu'un message de spécification de chiffrement de modification afin de vérifier que l'échange et les procédures d'authentification principaux étaient réussis. Le message de finition est le premier paquet protégé avec les algorithmes, les clés, et les secrets récemment négociés. Aucun accusé de réception du message de finition n'est prié ; les interlocuteurs peuvent commencer à envoyer des données cryptées juste après qu'ils envoient le message de finition. Les destinataires des messages de finition doivent vérifier que le contenu est correct.



Informations connexes

- RFC 6101 - La version 3.0 de Secure Sockets Layer Protocol
- Support et documentation techniques - Cisco Systems