Introduzione a SSL con transazione di esempio e Packet Exchange

Sommario

Introduzione

In questo documento vengono descritti i concetti di base del protocollo SSL (Secure Sockets Layer) e viene fornito un esempio di transazione e acquisizione di pacchetti.

Panoramica record SSL

L'unità di base dei dati in SSL è un record. Ogni record è costituito da un'intestazione di record di cinque byte seguita da dati.

Formato record

- **Tipo:** uint8 - valori elencati
- **Version:** unità16
- **Lunghezza:** unità16

Tipo	Version		Lunghezza	
T	VH	VL	LH	L

Tipo di record

In SSL sono disponibili quattro tipi di record:

- **Handshake** (22, 0x16)
- **Modifica specifica crittografia** (20, 0x14)
- **Avviso** (21, 0x15)
- **Dati applicazioni** (23, 0x17)

Versione record

La versione del record è un valore a 16 bit e viene formattata in ordine di rete.

> **Nota:** Per SSL versione 3 (SSLv3), la versione è 0x0300. Per TLSv1 (Transport Layer Security Version 1), la versione è 0x0301. Cisco Adaptive Security Appliance (ASA) non supporta SSL versione 2 (SSLv2), che utilizza la versione 0x0002, o qualsiasi versione di TLS successiva a TLSv1.

Lunghezza record

La lunghezza del record è un valore di 16 byte e viene formattata in ordine di rete.

In teoria, ciò significa che un singolo record può avere una lunghezza massima di 65.535 (2^16 -1) byte. La RFC2246 di TLSv1 indica che la lunghezza massima è 16.383 byte (2^14 -1). È noto che i prodotti Microsoft (Microsoft Internet Explorer e Internet Information Services) superano questi limiti.

Tipi di record

In questa sezione vengono descritti i quattro tipi di record SSL.

Record handshake

I record di handshake contengono un set di messaggi utilizzati per eseguire l'handshake. Questi sono i messaggi e i loro valori:

- **Richiesta Hello** (0, 0x00)
- **Hello client** (1, 0x01)
- **Server Hello** (2, 0x02)
- **Certificato** (11, 0x0B)
- **Scambio chiavi server** (12, 0x0C)
- **Richiesta certificato** (13, 0x0D)
- **Fine sessione server** (14, 0x0E)
- **Verifica certificato** (15, 0x0F)
- **Scambio chiavi client** (16, 0x10)
- **Fine** (20, 0x14)

In questo caso i record handshake non vengono crittografati. Tuttavia, un record handshake che contiene un messaggio finito viene sempre crittografato, in quanto si verifica sempre dopo un record CCS (Change Cipher Spec).

Record CCS

I record CCS vengono usati per indicare una modifica nei cifrari crittografici. Subito dopo la registrazione su CCS, tutti i dati vengono crittografati con la nuova cifratura. i record CCS possono essere o non essere crittografati; in una connessione semplice con un singolo handshake, il record CCS non viene crittografato.

Record avvisi

I record di avviso vengono utilizzati per indicare al peer che si è verificata una condizione. Alcuni avvisi sono avvisi, altri sono irreversibili e impediscono la connessione. Gli avvisi possono essere crittografati o meno e possono verificarsi durante un handshake o il trasferimento di dati. Esistono due tipi di avviso:

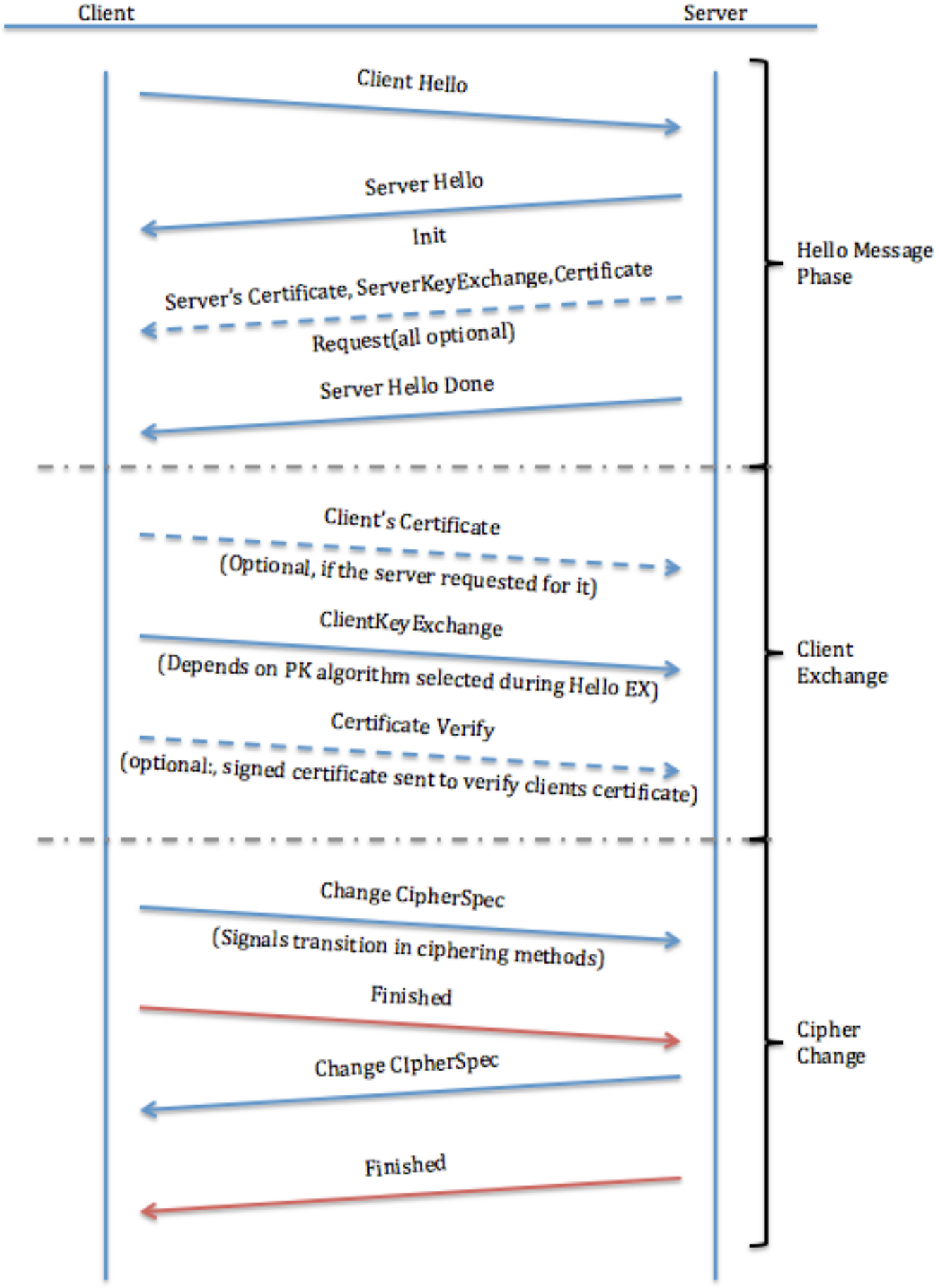
- **Avvisi di chiusura:** La connessione tra il client e il server deve essere chiusa correttamente per evitare qualsiasi tipo di attacco di troncamento. Viene inviato un messaggio **close_notice** che indica al destinatario che il mittente non invierà più messaggi su tale connessione.
- **Avvisi di errore:** Quando viene rilevato un errore, la parte che rileva l'errore invia un messaggio all'altra parte. Quando viene trasmesso o ricevuto un messaggio di avviso di errore irreversibile, entrambe le parti chiudono immediatamente la connessione. Di seguito sono riportati alcuni esempi di avvisi di errore:
  - **unexpected_message** (irreversibile)
  - **errore_decompressione**
  - **handshake_failure**

Record dati applicazione

Questi record contengono i dati effettivi dell'applicazione. Questi messaggi vengono trasportati dal livello di record e vengono frammentati, compressi e crittografati in base allo stato di connessione corrente.

Transazione di esempio

In questa sezione viene descritta una transazione di esempio tra il client e il server.



Hello Exchange

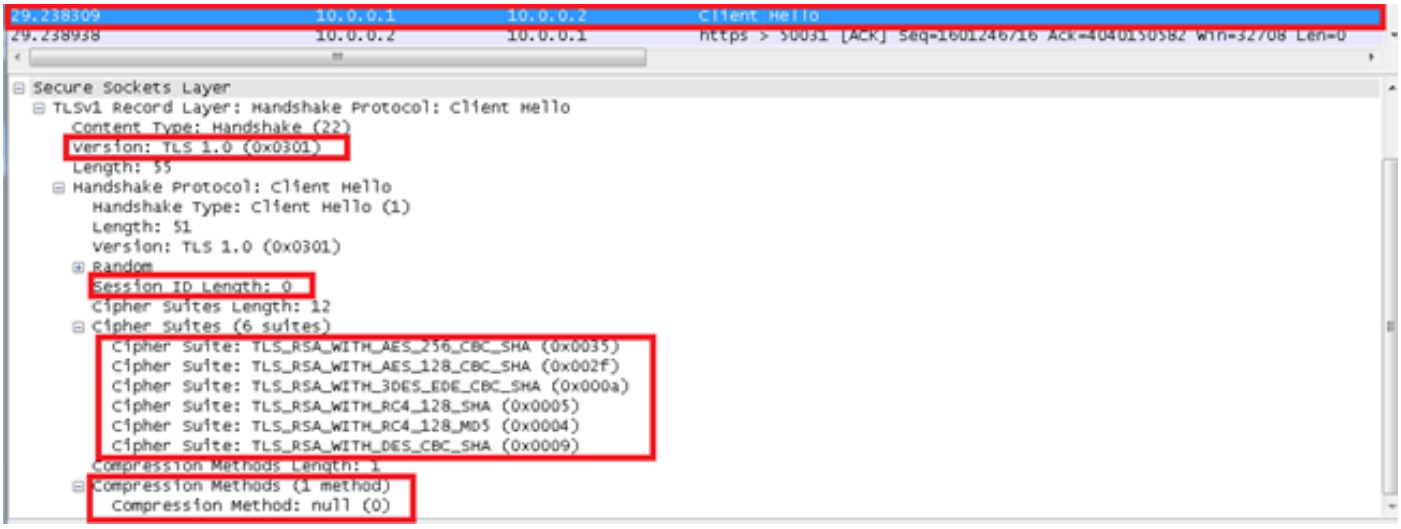
Quando un client e un server SSL iniziano a comunicare, concordano una versione del protocollo, selezionano gli algoritmi di crittografia, facoltativamente si autenticano a vicenda e utilizzano tecniche di crittografia a chiave pubblica per generare segreti condivisi. Questi processi vengono eseguiti nel protocollo handshake. In sintesi, il client invia un messaggio Hello al server, che deve rispondere con un messaggio Hello del server oppure si verifica un errore irreversibile e la connessione non riesce. Client Hello e Server Hello vengono utilizzati per stabilire funzionalità di miglioramento della sicurezza tra client e server.

Hello del client

Client Hello invia questi attributi al server:

- **Versione protocollo:** Versione del protocollo SSL con cui il client desidera comunicare durante la sessione.
- **ID sessione:** ID di una sessione che il client desidera utilizzare per questa connessione. Nel primo client Hello dello scambio, l'ID sessione è vuoto (fare riferimento alla schermata di acquisizione dei pacchetti ripresa dopo la nota).
- **Suite di crittografia:** Questo viene passato dal client al server nel messaggio Hello del client. Contiene le combinazioni di algoritmi di crittografia supportati dal client in ordine di preferenza del client (prima scelta). Ogni suite di cifratura definisce sia un algoritmo di scambio delle chiavi che una specifica di cifratura. Il server seleziona una suite di cifratura oppure, se non vengono presentate scelte accettabili, restituisce un avviso di errore dell'handshake e chiude la connessione.
- **Metodo di compressione:** Include un elenco di algoritmi di compressione supportati dal client. Se il server non supporta alcun metodo inviato dal client, la connessione non riesce. Anche il metodo di compressione può essere null.

Nota: L'indirizzo IP del server nelle clip è 10.0.0.2 e l'indirizzo IP del client è 10.0.0.1.



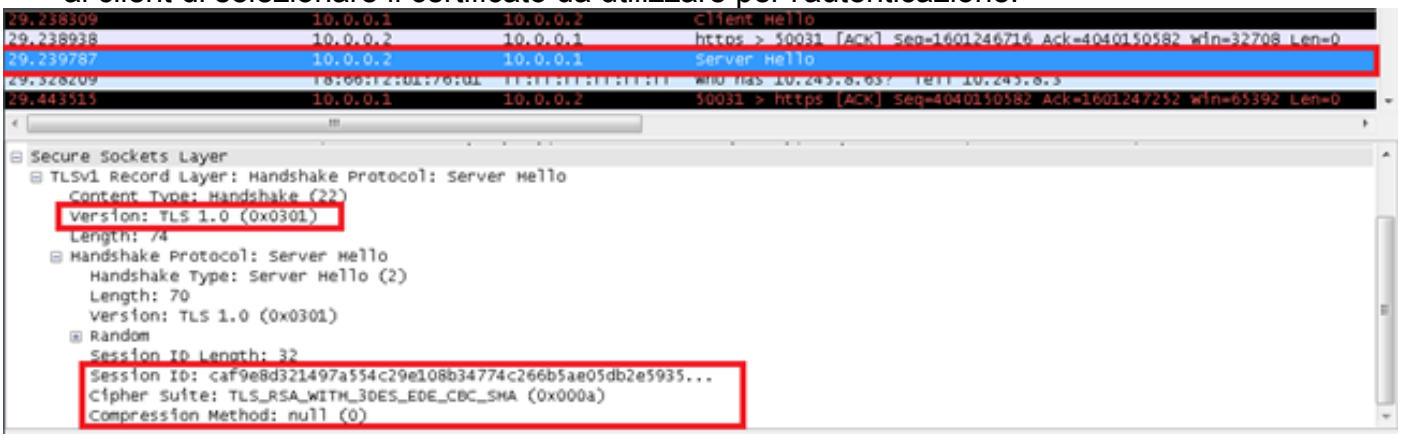
Server Hello

Il server invia al client i seguenti attributi:

- **Versione protocollo:** La versione scelta del protocollo SSL supportata dal client.
- **ID sessione:** Identità della sessione corrispondente alla connessione. Se l'ID sessione inviato dal client in Client Hello non è vuoto, il server cerca una corrispondenza nella cache della

sessione. Se viene trovata una corrispondenza e il server è disposto a stabilire la nuova connessione utilizzando lo stato sessione specificato, il server risponde con lo stesso valore fornito dal client. Indica una sessione ripresa e stabilisce che le parti devono procedere direttamente ai messaggi completati. In caso contrario, il campo contiene un valore diverso che identifica la nuova sessione. Il server potrebbe restituire un **session_id** vuoto per indicare che la sessione non verrà memorizzata nella cache e non potrà quindi essere ripresa.

- **Suite di crittografia:** Come selezionato dal server dall'elenco inviato dal client.
- **Metodo di compressione:** Come selezionato dal server dall'elenco inviato dal client.
- **Richiesta certificato:** Il server invia al client un elenco di tutti i certificati configurati e consente al client di selezionare il certificato da utilizzare per l'autenticazione.

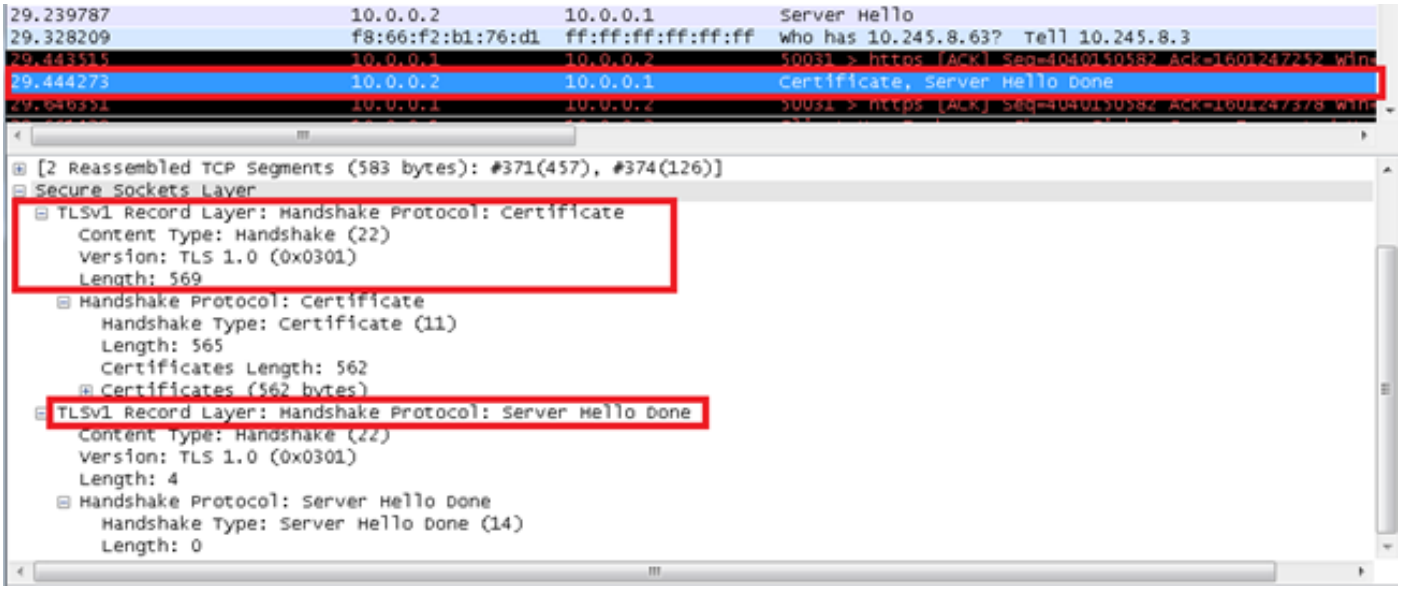


Per le richieste di ripresa delle sessioni SSL:

- Il server può inviare una richiesta Hello anche al client. Ciò serve solo per ricordare al client che dovrebbe avviare la rinegoziazione con una richiesta di Client Hello quando opportuno. Il client ignora la richiesta Hello del server se il processo di handshake è già in corso.
- I messaggi di handshake hanno maggiore precedenza sulla trasmissione dei dati delle applicazioni. La rinegoziazione deve iniziare al massimo tra una o due volte il tempo di trasmissione di un messaggio dati dell'applicazione di lunghezza massima.

Server pronto

Il messaggio Server Hello Done viene inviato dal server per indicare la fine del saluto del server e dei messaggi associati. Dopo l'invio del messaggio, il server attende una risposta del client. Dopo aver ricevuto il messaggio Server Hello Done, il client verifica che il server abbia fornito un certificato valido, se necessario, e controlla che i parametri di Server Hello siano accettabili.



Certificato server, scambio chiave server e richiesta certificato (facoltativo)

- **Certificato server:** se il server deve essere autenticato (in genere), il server invia il proprio certificato subito dopo il messaggio Server Hello. Il tipo di certificato deve essere appropriato per l'algoritmo di scambio chiavi della suite di cifratura selezionata e in genere è un certificato X.509.v3.
- **Scambio chiave server:** il messaggio Scambio chiave server viene inviato dal server se non dispone di un certificato. Se i parametri Diffie-Hellman (DH) sono inclusi nel certificato del server, questo messaggio non verrà utilizzato.
- **Richiesta certificato:** un server può facoltativamente richiedere un certificato al client, se appropriato per la suite di cifratura selezionata.

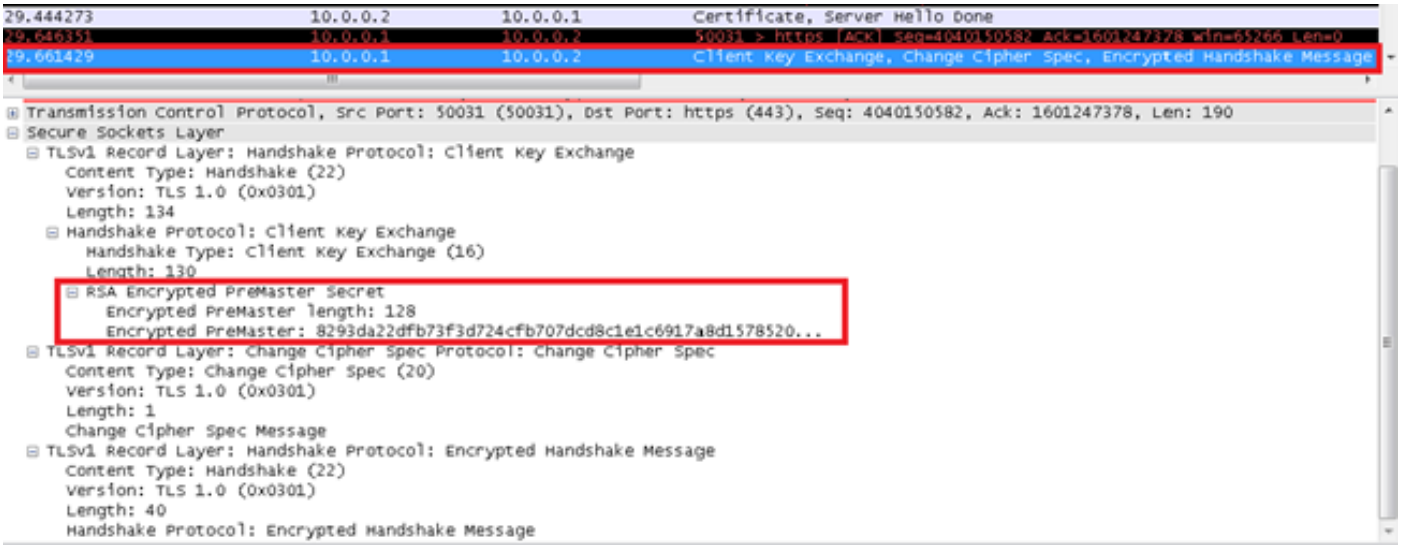
Scambio client

Certificato client (facoltativo)

Questo è il primo messaggio che il client invia dopo aver ricevuto un messaggio di completamento del server. Questo messaggio viene inviato solo se il server richiede un certificato. Se non è disponibile alcun certificato appropriato, il client invia un avviso **no_certificate**. Questa segnalazione è solo un'avvertenza; tuttavia, il server potrebbe rispondere con un avviso di errore irreversibile dell'handshake se è necessaria l'autenticazione del client. I certificati DH client devono corrispondere ai parametri DH specificati dal server.

Scambio chiavi client

Il contenuto di questo messaggio dipende dall'algoritmo a chiave pubblica selezionato tra i messaggi Hello del client e Hello del server. Il client utilizza una chiave premaster crittografata dall'algoritmo RSA (Rivest-Shamir-Addleman) o DH per l'autenticazione e l'accordo della chiave. Quando RSA viene utilizzato per l'autenticazione del server e lo scambio di chiavi, il client genera un **pre_master_secret di** 48 byte, lo cripta con la chiave pubblica del server e lo invia al server. Il server utilizza la chiave privata per decrittografare il **pre_master_secret**. Entrambe le parti convertono quindi il **pre_master_secret** nel **master_secret**.



Verifica certificato (facoltativa)

Se il client invia un certificato con funzionalità di firma, viene inviato un messaggio di verifica del certificato con firma digitale per verificare esplicitamente il certificato.

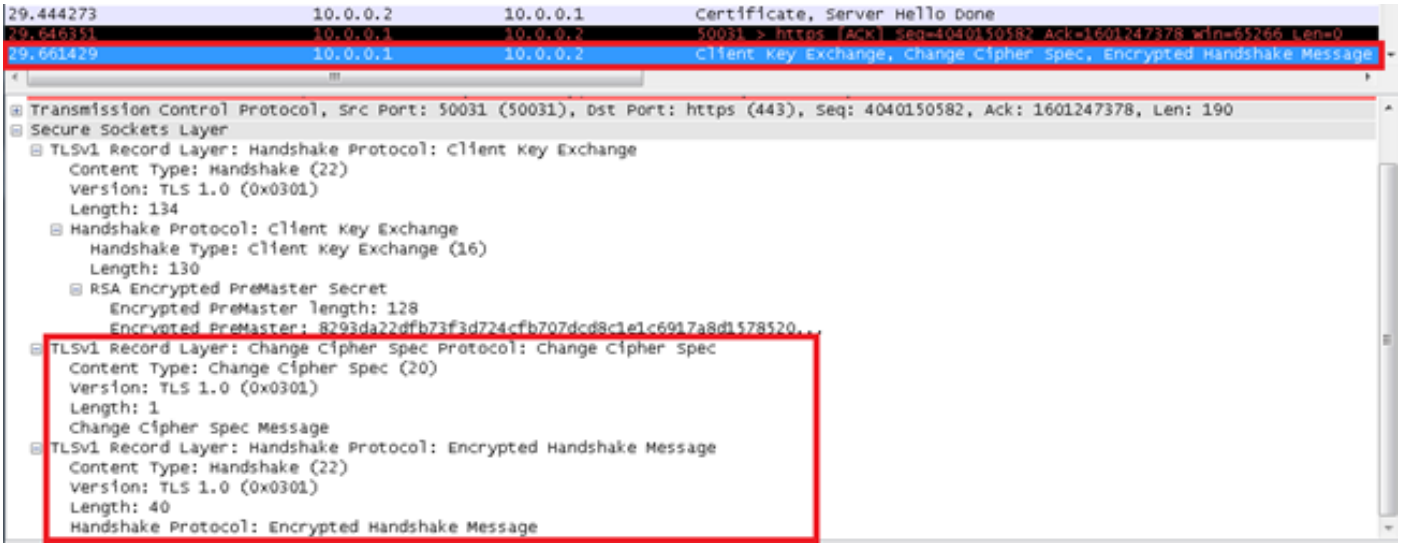
Cifratura

Messaggi di modifica delle specifiche di crittografia

Il messaggio Modifica specifica di cifratura viene inviato dal client, che copia la specifica di cifratura in sospeso (la nuova) nella specifica di cifratura corrente (quella utilizzata in precedenza). Il protocollo Modifica specifica cifratura esiste per segnalare transizioni nelle strategie di cifratura. Il protocollo è costituito da un singolo messaggio, cifrato e compresso in base alla specifica di cifratura corrente (non in sospeso). Il messaggio viene inviato sia dal client che dal server per notificare alla parte ricevente che i record successivi sono protetti in base alla specifica di cifratura e alle chiavi negoziate più di recente. La ricezione di questo messaggio determina la copia da parte del destinatario dello stato di lettura in sospeso nello stato di lettura corrente. Il client invia un messaggio Modifica specifica crittografia dopo lo scambio di chiave dell'handshake e gli eventuali messaggi di verifica del certificato e il server ne invia uno dopo aver elaborato correttamente il messaggio di scambio chiave ricevuto dal client. Quando viene ripresa una sessione precedente, il messaggio Modifica specifica crittografia viene inviato dopo i messaggi Hello. Nelle clip, i messaggi Scambio client, Cambia crittografia e Fine vengono inviati come un unico messaggio dal client.

Messaggi completati

Il messaggio Fine viene sempre inviato subito dopo il messaggio Modifica specifica di crittografia per verificare che i processi di scambio chiave e autenticazione abbiano avuto esito positivo. Il messaggio Finished è il primo pacchetto protetto con gli algoritmi, le chiavi e i segreti negoziati più di recente. Non è richiesto alcun riconoscimento del messaggio Finished. le parti possono iniziare a inviare i dati crittografati subito dopo aver inviato il messaggio Finished. I destinatari dei messaggi Finiti devono verificare che il contenuto sia corretto.



Informazioni correlate

- RFC 6101 - Protocollo Secure Sockets Layer versione 3.0
- Wireshark SSL wiki - decrittografa i pacchetti SSL con Wireshark
- Documentazione e supporto tecnico – Cisco Systems