Введение SSL с типовой транзакцией и обменом пакетами

Содержание

Введение

Этот документ описывает базовые понятия протокола Уровня защищенных сокетов (SSL) и предоставляет типовую транзакцию и захват пакета.

Обзор записи SSL

Базовая единица данных в SSL является записью. Каждая запись состоит из пятиразрядного рекордного заголовка, придерживавшегося данными.

Формат записи

- **Введите :** uint8 - значения упомянуты ниже
- **Version :** uint16
- **Длина:** uint16

Введите	**Version**		**Длина**	
T	СПИДОБАРОГРАФ	VL	LH	LL

Тип записи

В SSL существует четыре типа записи:

- **Квитирование** (22, 0x16)
- **Спецификация шифра изменения** (20, 0x14)
- **Предупреждение** (21, 0x15)
- **Данные прикладной программы** (23, 0x17)

Рекордная версия

Рекордная версия является 16 значениями в байтах и отформатирована в сетевом заказе.

> **Примечание:** Для Версии SSL 3 (SSLv3) версия является 0x0300. Для Версии 1 (TLSv1) Transport Layer Security версия является 0x0301. Устройство адаптивной защиты Cisco (ASA) не поддерживает Версию SSL 2 (SSLv2), которая использует версию 0x0002 или любую версию TLS, больше, чем TLSv1.

Рекордная длина

Рекордная длина является 16 значениями в байтах и отформатирована в сетевом заказе.

В теории это означает, что одиночная запись может быть до 65,535 (2^16 - 1) байты в длине. TLSv1 RFC2246 сообщает, что максимальная длина 16,383 (2^14 - 1) байты. Продукты Microsoft (Microsoft Internet Explorer и Internet Information Services), как известно, превышают эти пределы.

Типы записей

В этом разделе описываются четыре типа записей SSL.

Записи квитирования

Записи квитирования содержат ряд сообщений, которые используются для квитирования. Это сообщения и их значения:

- **Запрос Hello** (0, 0x00)
- **Сообщение приветствия клиента** (1, 0x01)
- **Приветствие сервера** (2, 0x02)
- **Сертификат** (11, 0x0B)
- **Exchange серверного ключа** (12, 0x0C)
- **Запрос сертификата** (13, 0x0D)
- **Приветствие сервера сделанный** (14, 0x0E)
- **Сертификат проверяет** (15, 0x0F)
- **Клиентский обмен ключами** (16, 0x10)

- **Законченный** (20, 0x14)

В простом случае не зашифрованы записи квитирования. Однако запись квитирования, которая содержит Законченное сообщение, всегда шифруется, как это всегда происходит после записи Спецификации шифра изменения (CCS).

Записи спецификации шифра изменения

Записи CCS используются для указания на изменение в crpytographic шифрах. Сразу после записи CCS, все данные зашифрованы с новым шифром. Записи CCS могли бы или не могли бы быть зашифрованы; в простом соединении с одиночным квитированием не зашифрована запись CCS.

Аварийные записи

Аварийные записи используются, чтобы указать к узлу, что произошло условие. В то время как другие являются фатальными и заставляют соединение отказывать, некоторые предупреждения являются предупреждениями. Предупреждения могли бы или не могли бы быть зашифрованы и могли бы произойти во время квитирования или во время передачи данных. Существует два типа предупреждений:

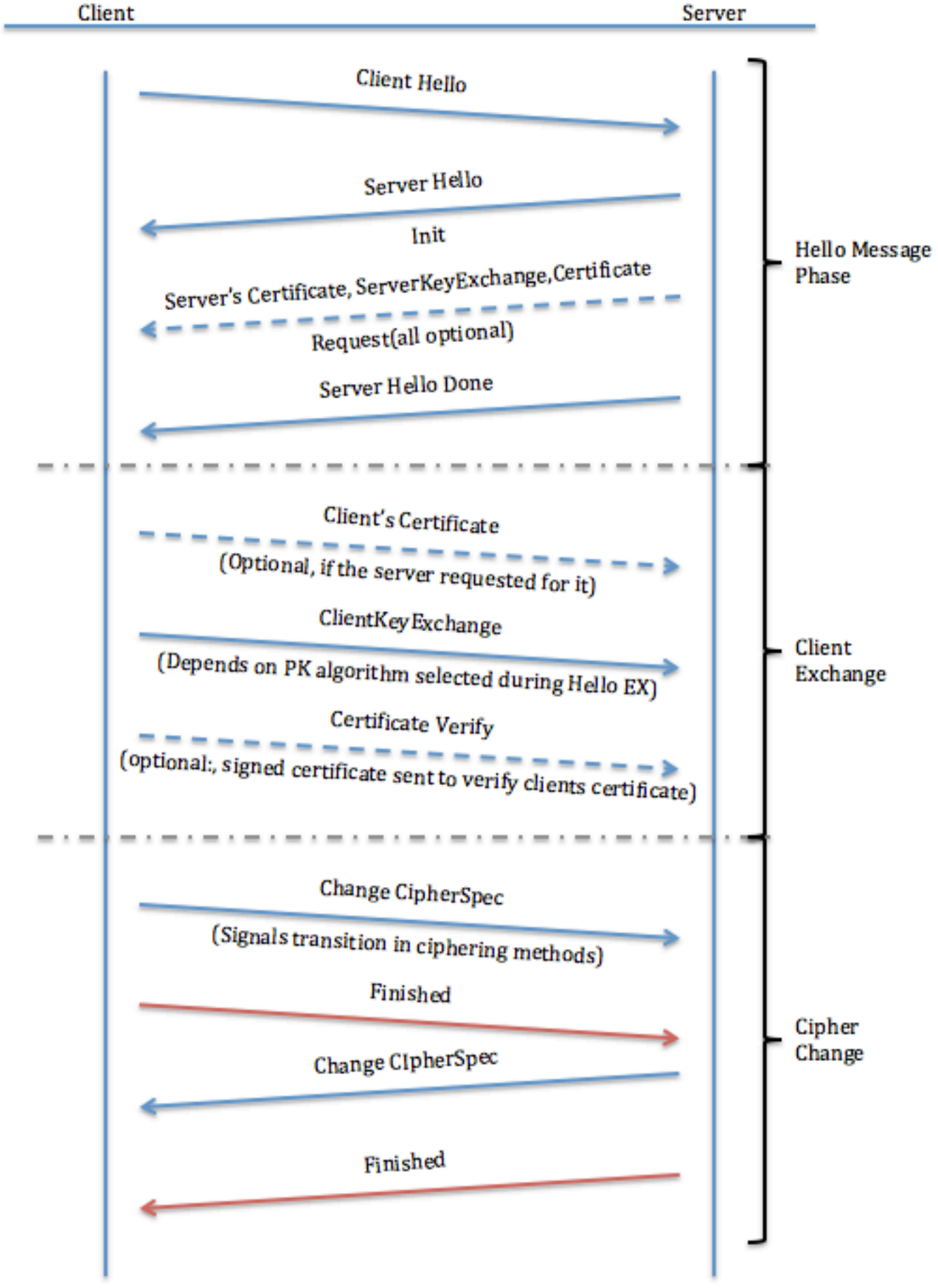
- **Предупреждения Закрытия:** соединение между клиентом и сервером должно быть должным образом закрыто во избежание любого вида атак усечения. Сообщение **close_notify** передается, который указывает получателю, что отправитель не будет больше передавать сообщения на том соединении.
- **Ошибочные Предупреждения:** Когда ошибка обнаружена, сторона обнаружения передает сообщение другой стороне. На передачу или получение фатального сигнального сообщения, обе стороны сразу закрывают соединение. Некоторые примеры ошибочных предупреждений:
  - (фатальный) **unexpected_message**
  - **decompression_failure**
  - **handshake_failure**

Запись данных прикладной программы

Эти записи содержат данные реального приложения. Эти сообщения несет рекордный уровень и фрагментируют, сжимают и шифруют, на основе состояния текущего соединения.

Типовая транзакция

В этом разделе описываются типовую транзакцию между клиентом и сервером.



Обмен приветствиями

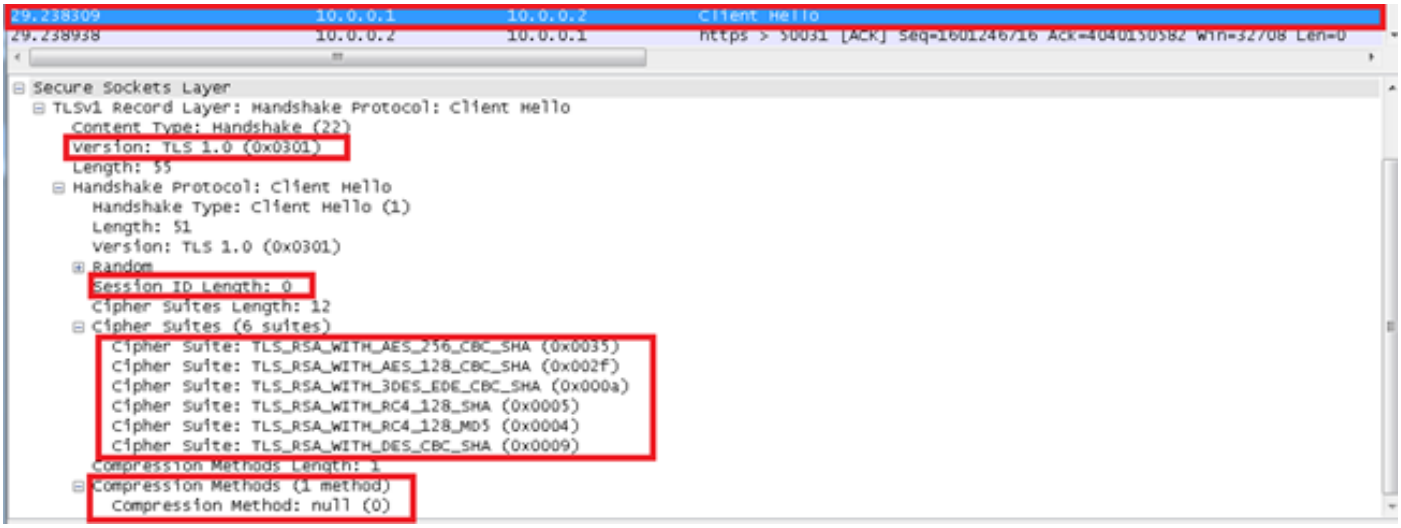
Когда клиент SSL и сервер начинаются к communitcate, они договариваются о версии протокола, выбирают криптографические алгоритмы, дополнительно аутентифицируют друг друга и используют способы шифрования с открытым ключом для генерации общих секретных ключей. Эти процессы выполнены в протоколе подтверждения связи. Таким образом, клиент передает сообщение Сообщения приветствия клиента к серверу, который должен ответить Приветствие сервера, сообщение или фатальная ошибка происходят и сбои соединения. Сообщение приветствия клиента и Приветствие сервера используется для установления возможностей улучшения безопасности между клиентом и сервером.

Сообщение приветствия клиента

Сообщение приветствия клиента передает эти атрибуты к серверу:

- **Версия протокола:** версия протокола SSL, которым клиент хочет связаться во время этого сеанса.
- **Идентификатор сеанса:** ID сеанса клиент хочет использовать для этого соединения. В первом Сообщении приветствия клиента обмена идентификатор сеанса пуст (обратитесь к снимку экрана захвата пакета после примечания ниже).
- **Набор шифров:** Это передают от клиента к серверу в сообщении Сообщения приветствия клиента. Это содержит комбинации криптографических алгоритмов, поддерживаемых клиентом в порядке предпочтения клиента (предпочтительный вариант сначала). Каждый набор шифров определяет и Key Exchange Algorithm и спецификацию шифра. Сервер выбирает набор шифров или, если никакие приемлемые выборы не представлены, возвращает предупреждение сбоя квитирования и закрывает соединение.
- **Метод сжатия:** Включает список алгоритмов сжатия, поддерживаемых клиентом. Если сервер не поддерживает метода, передаваемого клиентом, сбоями соединения. Метод сжатия может также быть пустым.

Примечание: IP-адрес сервера в перехватах 10.0.0.2, и IP-адрес клиента 10.0.0.1.

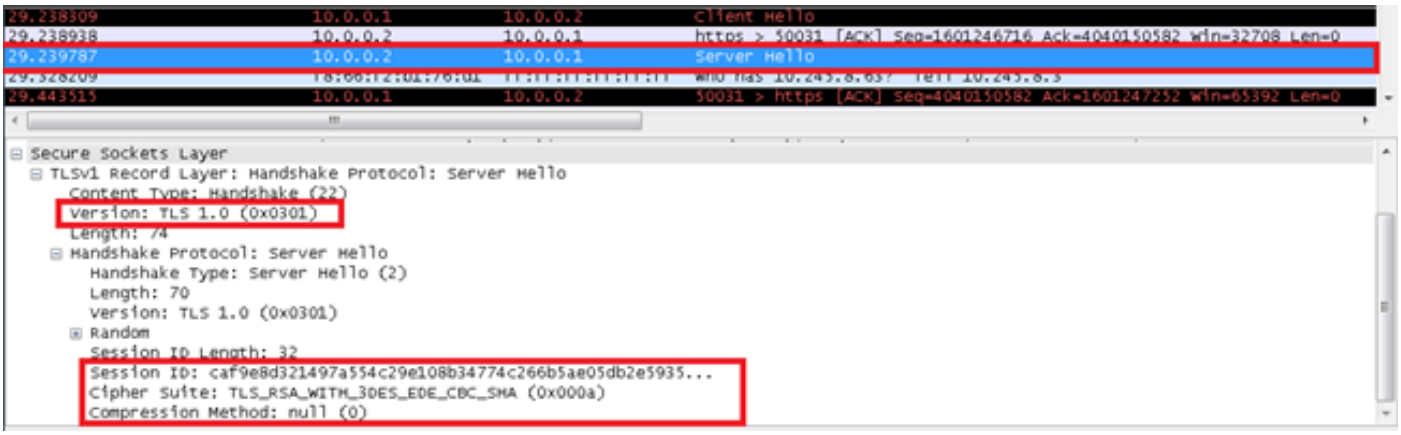


Приветствие сервера

Сервер передает эти атрибуты обратно клиенту:

- **Версия протокола:** выбранная версия протокола SSL, что поддержки клиентов.

- **Идентификатор сеанса:** Это - идентичность сеанса, который соответствует этому соединению. Если идентификатор сеанса, передаваемый клиентом в Сообщении приветствия клиента, не пуст, виды сервера в кэше сеанса для соответствия. Если соответствие найдено, и сервер готов установить новое соединение с помощью указанного состояния сеанса, сервер отвечает тем же значением, которое было предоставлено клиентом. Это указывает на возобновленный сеанс и диктует, что стороны должны продолжиться непосредственно к законченным сообщениям. В противном случае это поле содержит другое значение, которое определяет новый сеанс. Сервер мог бы возвратить пустой **session_id**, чтобы указать, что сеанс не будет кэшироваться, и поэтому не может быть возобновлен.
- **Набор шифров:** Как выбрано сервером из списка, который передавался от клиента.
- **Метод сжатия:** Как выбрано сервером из списка, который передавался от клиента.
- **Запрос сертификата:** сервер передает клиенту список всех сертификатов, которые настроены на нем, и позволяет клиенту выбирать, какой сертификат он хочет использовать для аутентификации.

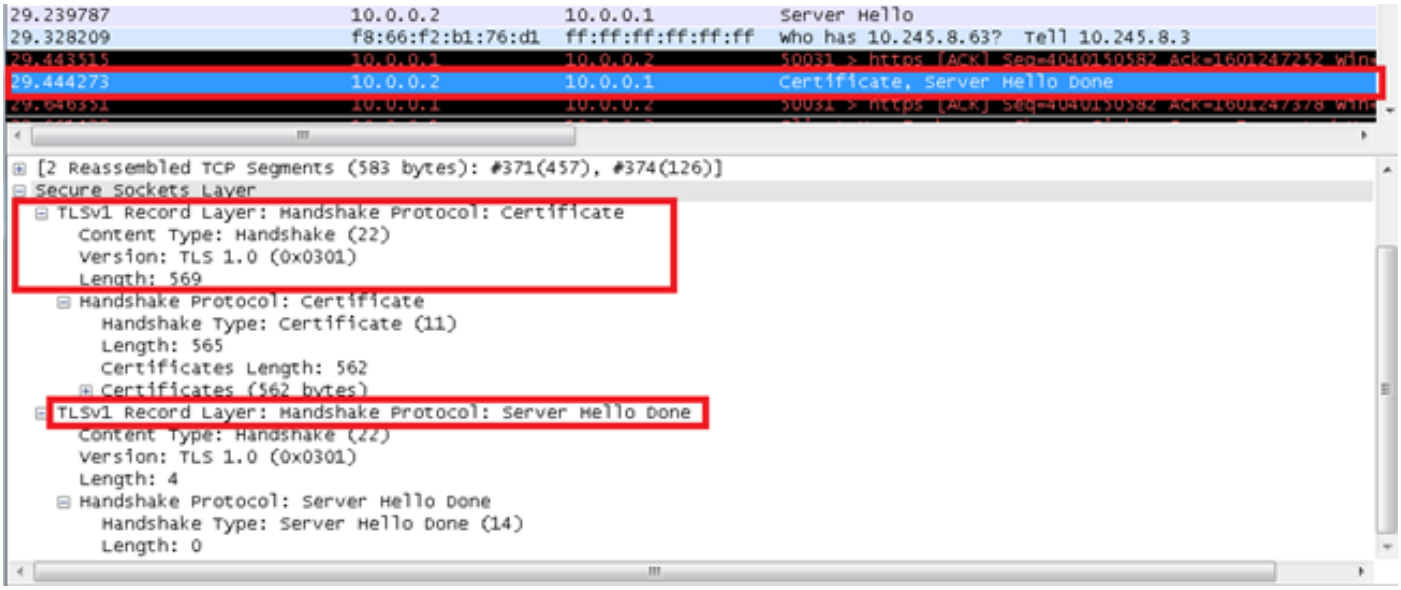


Для запросов возобновления сеанса SSL:

- Сервер может отправить запрос Hello клиенту также. Это только, чтобы напомнить клиенту, что это должно начать пересмотр с запроса Сообщения приветствия клиента, когда удобный. Если процесс квитирования идет уже полным ходом, клиент игнорирует запрос Hello от сервера.
- Сообщения квитирования имеют большие приоритеты по передаче данных прикладной программы. Пересмотр должен начаться в не больше, чем один или два раза времени передачи сообщения данных прикладной программы максимальной длины.

Приветствие сервера сделанный

Приветствие сервера Сделанное сообщение передается сервером для указания на конец приветствие сервера и привязанные сообщения. После того, как это передаст это сообщение, сервер ждет клиентского ответа. По получении Приветствие сервера Сделанного сообщения клиент проверяет, что сервер предоставил подтвержденный сертификат при необходимости и проверяет, что Приветствие сервера параметры приемлемы.



Серверный сертификат, Exchange серверного ключа и запрос сертификата (Необязательно)

- **Серверный сертификат:** Если сервер должен аутентифицироваться (который обычно имеет место), сервер сразу передает свой сертификат после Приветствие сервера сообщение. Тип сертификата должен быть соответствующим выбранному Key Exchange Algorithm набора шифров и обычно является сертификатом X.509.v3.
- **Exchange Серверного ключа:** Сообщение Exchange Серверного ключа передается сервером, если это не имеет никакого сертификата. Если Диффи? Хеллмен (DH), параметры включены с серверным сертификатом, это сообщение, не используется.
- **Запрос сертификата:** сервер может дополнительно запросить сертификат от клиента, в подходящих случаях выбранному набору шифров.

Клиентский Exchange

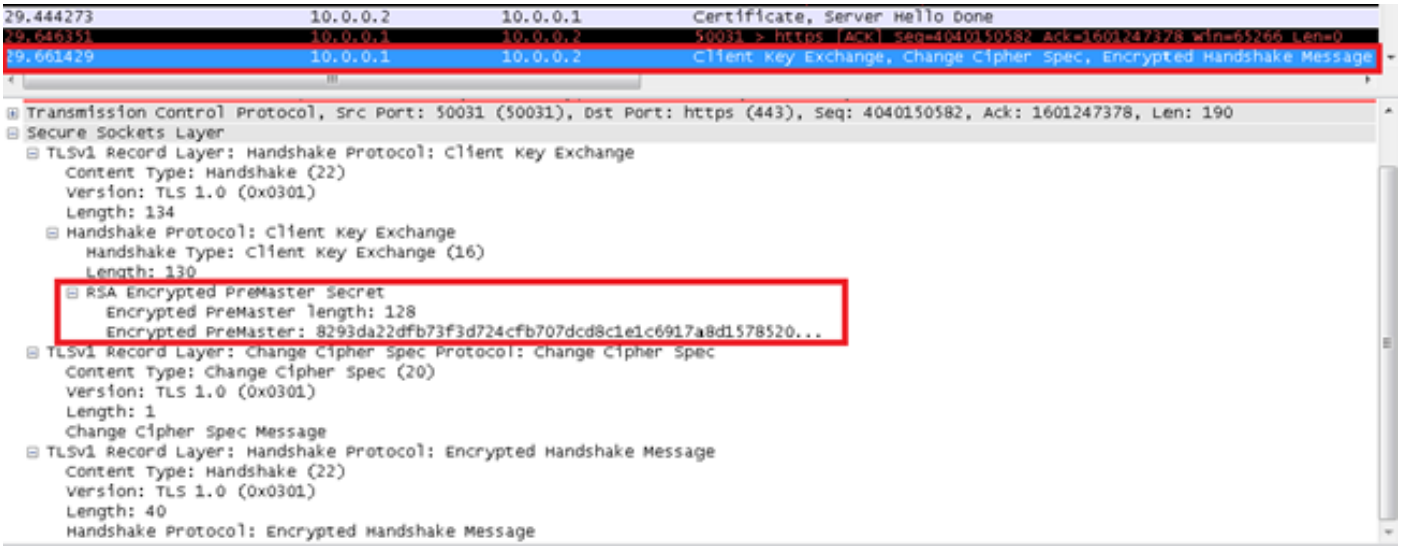
Сертификат клиента (Необязательно)

Это - первое сообщение, что клиент передает после того, как он/она получает Приветствие сервера Сделанное сообщение. Это сообщение только передается если запросы к серверу сертификат. Если никакой подходящий сертификат не доступен, клиент передает предупреждение **no_certificate** вместо этого. Это предупреждение является только предупреждением; однако, если аутентификация клиента требуется, сервер мог бы ответить фатальным предупреждением сбоя квитирования. Клиентские сертификаты DH должны совпасть, сервер задал параметры DH.

Клиентский обмен ключами

Содержание этого сообщения зависит от алгоритма с открытым ключом, выбранного между Сообщением приветствия клиента и Приветствие сервера сообщения. Клиент использует или предосновной ключ, зашифрованный алгоритмом алгоритма цифровой подписи райвеста шамира адлемана (RSA) или DH для согласования ключей и аутентификации. Когда RSA используется для проверки подлинности сервера, и обмен ключами, 48 байтов **pre_master_secret** генерируются клиентом, зашифровали под открытым ключом сервера и передали к серверу. Сервер использует секретный ключ для дешифрования

pre_master_secret. Обе стороны тогда преобразовывают **pre_master_secret** в **master_secret**.



Сертификат проверяет (Необязательно)

Если клиент передает сертификат с подписанием способности, снабженный цифровой подписью Сертификат Проверяют, что сообщение передается для явной проверки сертификата.

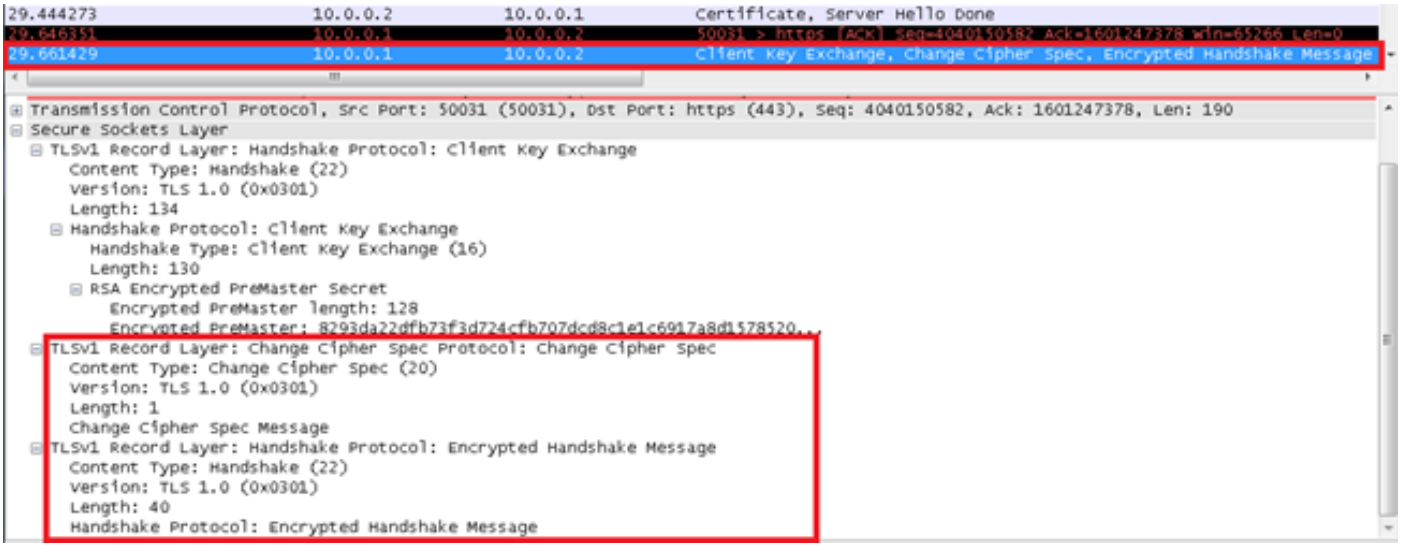
Изменение шифра

Сообщения спецификации шифра изменения

Сообщение Спецификации Шифра Изменения передается клиентом, и клиент копирует Спецификацию Шифра в состоянии ожидания (новая) в текущую Спецификацию Шифра (тот, который ранее использовался). Протокол Спецификации Шифра изменения существует чтобы к сигнальным переходам в считающих стратегиях. Протокол состоит из одиночного сообщения, которое зашифровано и сжато под током (не ожидание) Спецификация Шифра. Сообщение передается обоими клиента и сервер, чтобы уведомить принимающую сторону, что последующие записи защищены под последний раз согласованной Спецификацией Шифра и ключами. Прием этого сообщения заставляет получатель копировать "чтение, ожидающее" состояние в "чтение текущее" состояние. Клиент передает сообщение Спецификации Шифра Изменения после обмена ключами квитирования, и Сертификат Проверяют сообщения (если таковые имеются), и сервер передает тот после того, как это успешно обрабатывает сообщение обмена ключами, которое это получило от клиента. Когда предыдущий сеанс возобновлен, сообщение Спецификации Шифра Изменения передается после Приветственных сообщений. В перехватах Клиентский Exchange, Шифр Изменения и Законченные сообщения передаются как одиночное сообщение от клиента.

Законченные сообщения

Законченное сообщение всегда сразу передано после сообщения Спецификации Шифра Изменения, чтобы проверить, что обмен ключами и процессы проверки подлинности были успешны. Законченное сообщение является первым защищенным пакетом с последний раз согласованными алгоритмами, ключами и тайнами. Никакое подтверждение Законченного сообщения не требуется; стороны могут начать передавать зашифрованные данные сразу после того, как они передадут Законченное сообщение. Получатели Законченных сообщений должны проверить, что содержание корректно.



Дополнительные сведения

- RFC 6101 - версия протокола 3.0 уровня защищенных сокетов
- Cisco Systems – техническая поддержка и документация