

与示例处理和信息包交换的SSL简介

Contents

[Introduction](#)

[SSL记录概述](#)

[记录格式](#)

[记录类型](#)

[记录版本](#)

[记录长度](#)

[记录的类型](#)

[握手记录](#)

[CCS记录](#)

[戒备记录](#)

[应用数据记录](#)

[示例处理](#)

[Hello交换](#)

[客户端Exchange](#)

[密码更改](#)

[Related Information](#)

Introduction

本文描述安全套接字协议层(SSL)协议基本概念，并且提供示例处理和信息包获取。

SSL记录概述

数据基本的单元在SSL的是记录。每个记录包括一五字节记录首标，跟随由数据。

记录格式

- **类型** : uint8 -列出的值
- **版本** : uint16
- **长度** : uint16

类型 version 长度

T VH VL LH LL

记录类型

有在SSL的四种记录类型：

- **握手**(22, 0x16)
- **更改密码Spec** (20, 0x14)
- **警告**(21, 0x15)
- **应用数据**(23, 0x17)

记录版本

记录版本是16 BITS值和被格式化在网络秩序。

Note:对于SSL版本3 (SSLv3)，版本是0x0300。对于传输层安全版本1 (TLSv1)，版本是0x0301。Cisco可适应的安全工具(ASA)比TLSv1不支持SSL版本2 (SSLv2)，使用版本0x0002，或者TLS的任何版本极大。

记录长度

记录长度是16字节值和被格式化在网络秩序。

在理论上，这意味着单个记录可以是长度65,535个($2^{16} - 1$)字节。TLSv1 RFC2246阐明，最大长度是16,383个($2^{14} - 1$)字节。Microsoft产品(微软互联网探索者微软因特网资源管理器和互联网信息服务)知道超过这些限额。

记录的类型

此部分描述SSL记录的四种类型。

握手记录

握手记录包含是被使用的为了握手的一套消息。这些是消息和他们的值：

- Hello请求(0, 0x00)
- 客户端Hello (1, 0x01)
- 服务器问候(2, 0x02)
- 认证(11, 0x0B)
- 服务器密钥Exchange (12, 0x0C)
- 证书请求(13, 0x0D)
- 执行的服务器问候(14, 0x0E)
- Certificate verify (15, 0x0F)
- 客户端密匙交换(16, 0x10)
- 完成(20, 0x14)

在简单的案件，握手记录没有被加密。然而，包含一个完成的消息的握手记录总是被加密，因为总是发生，在更改密码Spec (CCS)后记录。

CCS记录

CCS记录用于为了指示在密码密码上的一个变化。在CCS记录，所有数据用新的密码之后加密。CCS记录可能或也许不被加密;在与单个握手的简单的连接，CCS记录没有被加密。

戒备记录

戒备记录用于为了表明到对等体情况发生了。而其他是致命的并且造成连接发生故障，一些戒备是警告。在数据传输期间，戒备可能或也许不被加密，并且也许发生在握手期间或。有戒备的两种类型：

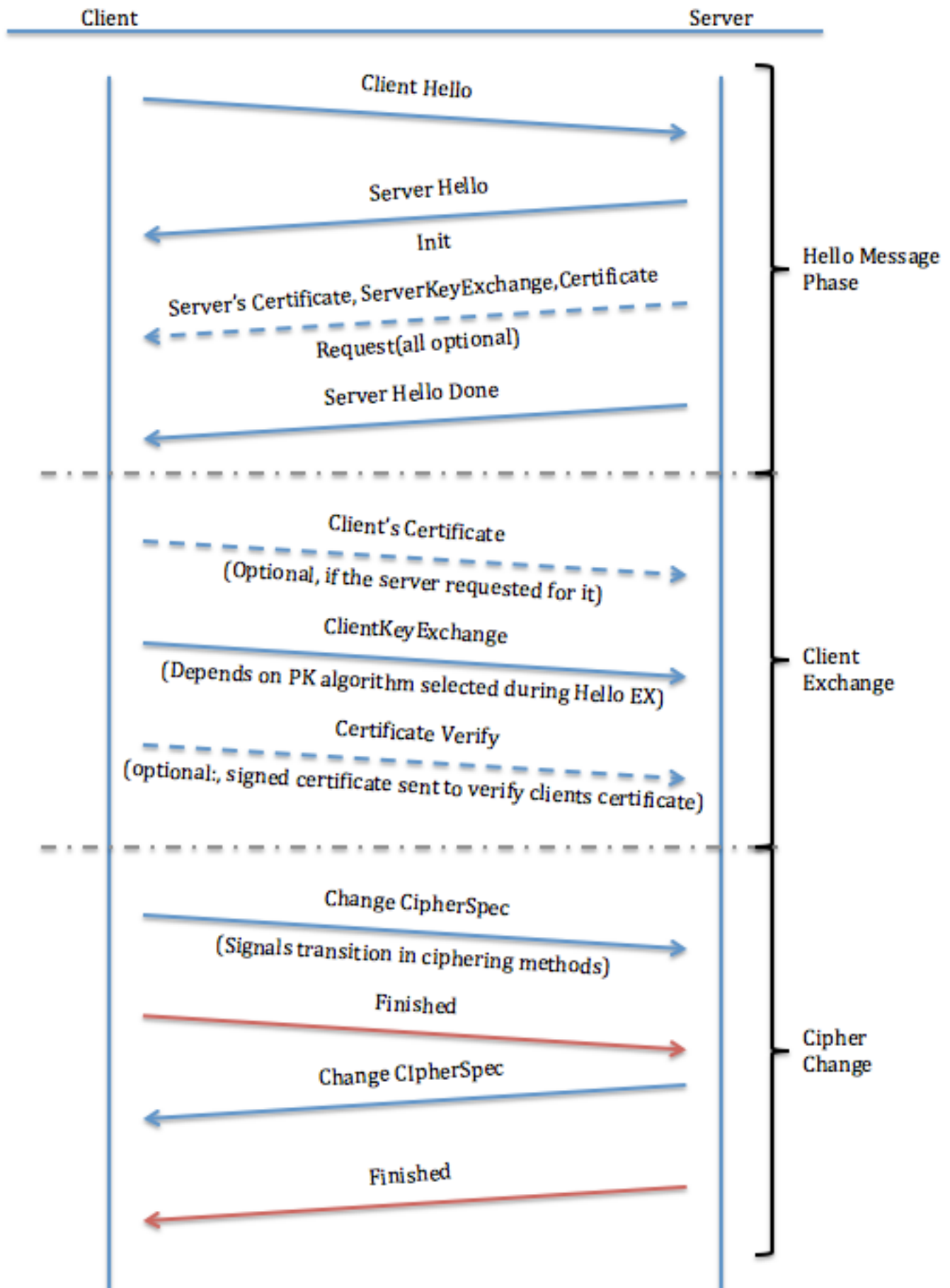
- **关闭戒备**：必须适当地断开客户端和服务端之间的连接以避免任何截断攻击。的 `close_notify` 信息表明给接收人传送发送方不会再将传送在该连接的信息。
- **错误戒备**：当发现时错误，发现的当事人传送信息到另一个当事人。在一个致命警报消息的发射或收据，两个当事人立即断开连接。错误戒备某些示例是：
 - `unexpected_message` (致命)
 - `decompression_failure`
 - `handshake_failure`

应用数据记录

这些记录包含实际应用数据。这些消息由记录层传播和根据当前连接状态被分段，被压缩，并且被加密。

示例处理

此部分描述在客户端和服务端之间的示例处理。



Hello交换

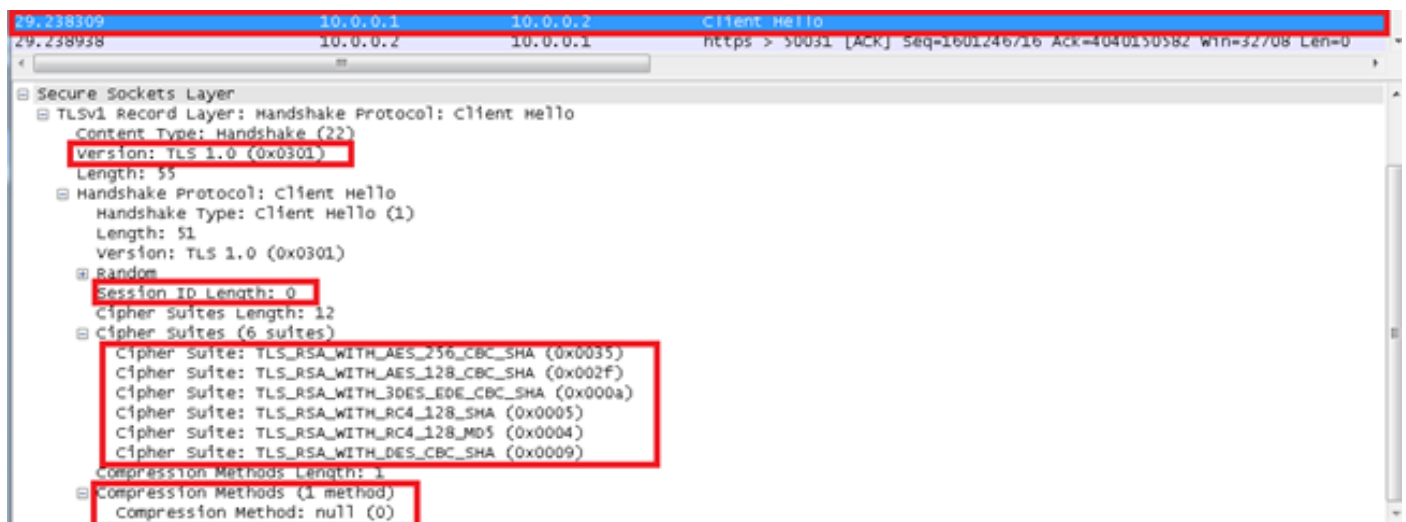
当SSL客户端和服务端开始沟通时，他们对协议版本达成协议，挑选加密算法，可选地互相验证，并且使用公共密钥加密技术为了生成共有的秘密。这些进程在握手协议执行。总之，客户端派遣Hello信息一个的客户端到服务器，必须回应服务器问候消息或致命错误生成，并且连接发生故障。客户端Hello和服务器问候用于设立在客户端和服务端之间的安全性增强功能。

客户端Hello

客户端Hello发送这些属性到服务器：

- **协议版本**：客户端希望沟通SSL协议的版本在此会话期间。
- **会话ID**：客户端希望使用此连接会话的ID。在交换的第一个客户端Hello，会话ID是空的(在附注以后请参见信息包获取屏幕画面)。
- **密码套件**：这从客户端通过到在Hello信息的客户端的服务器。它包含客户端支持的加密算法的组合按照客户端首选(首先第一个选择的顺序)。每个密码套件定义了一个密钥交换算法和一个密码spec。服务器选择一个密码套件或，如果没有提交可接受的选择，返回握手故障戒备并且断开连接。
- **压缩方法**：包括客户端支持的压缩算法列表。如果服务器不支持客户端发送的任何方法，连接发生故障。压缩方法可以也空。

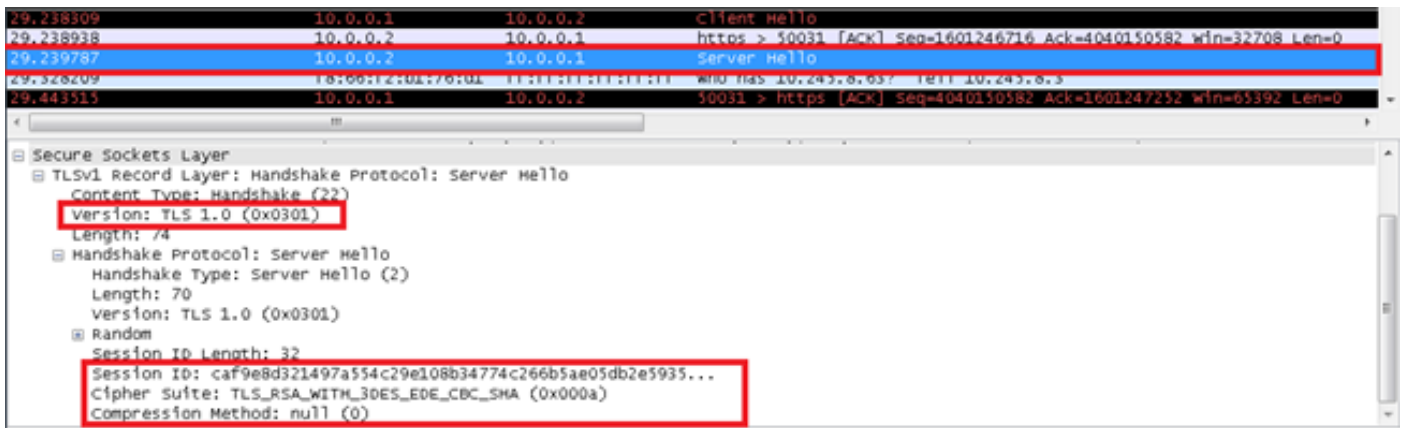
Note:在捕获的服务器IP地址是10.0.0.2，并且客户端IP地址是10.0.0.1。



服务器问候

服务器退还这些属性到客户端：

- **协议版本**：客户端支持SSL协议的选择的版本。
- **会话ID**：这是对应于此连接会话的身份。如果客户端的Hello客户端发送的会话ID不是空的，服务器在会话高速缓冲存储器查找为匹配。如果找到匹配使用指定的会话状态，并且服务器是愿意建立新连接，服务器回应客户端提供的同一值。这指示一次恢复的会话并且指明当事人必须进行直接地到完成的消息。否则，此字段包含识别个新会话的不同的值。服务器也许返回一空 `session_id` 为了表明不会缓存会话，并且不可能恢复。
- **密码套件**：如选择由服务器从从客户端被发送的列表。
- **压缩方法**：如选择由服务器从从客户端被发送的列表。
- **证书请求**：服务器发送客户端被配置对此所有证书的列表，并且允许认证它要使用认证的客户端选择。

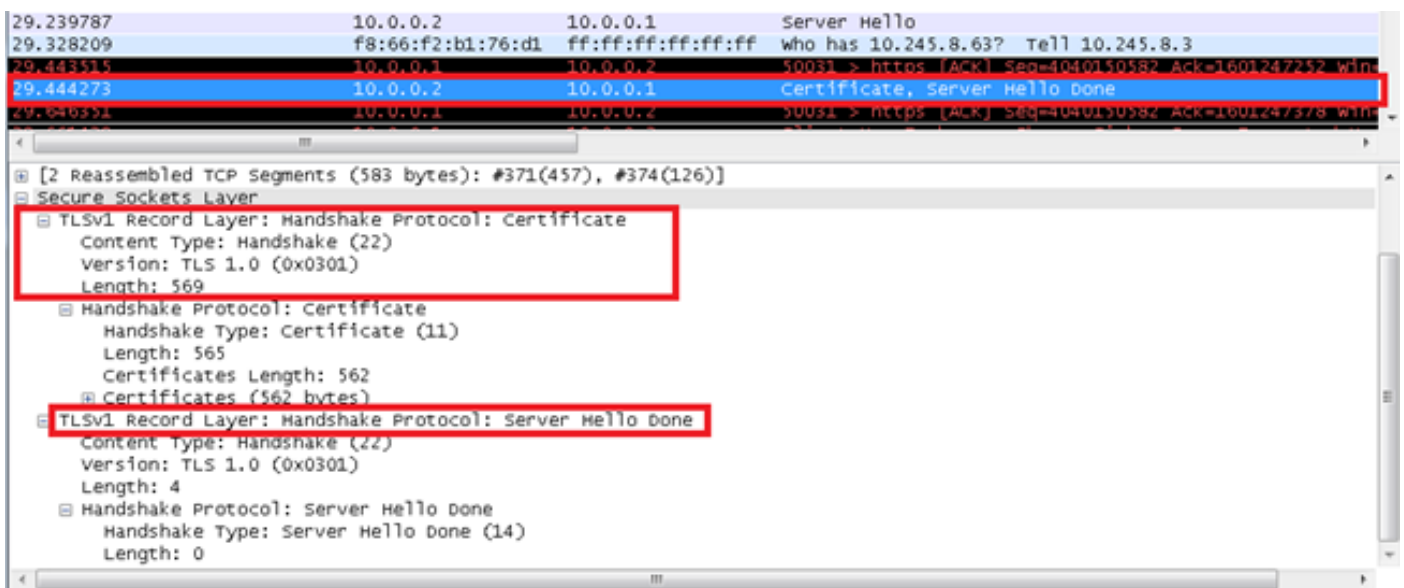


SSL会话恢复请求：

- 服务器能发送Hello请求到客户端。这是为了只提醒客户端应该开始与客户端Hello请求的重新协商，当方便。如果握手进程已经进行中，客户端忽略自服务器的Hello请求。
- 握手消息有在应用数据发射的更多优先次序。重新协商必须在不大于一两倍内开始最大长度应用数据消息的传输时间。

执行的服务器问候

服务器传送服务器问候完成的信息为了指示服务器问候和被关联的消息的末端。在它传送此信息后，服务器等待客户端回应。收到执行的服务器问候后消息，客户端验证服务器提供了一个有效证书，如果必须，并且检查服务器问候参数是可接受的。



服务器证明、服务器密钥Exchange和证书请求(可选)

- **服务器证明**：如果必须验证服务器(通常是实际情形)，服务器发送其在服务器问候消息之后的认证。证书类型一定是适当的为所选的密码套件密钥交换算法，并且通常是X.509.v3认证。
- **服务器密钥Exchange**：如果没有认证，服务器发服务器密钥交换消息。如果Diffie-Hellman (DH)参数被包括在服务器证明中，没有使用此消息。
- **证书请求**：服务器能为所选的密码套件可选地请求从客户端的一个认证，如果适当。

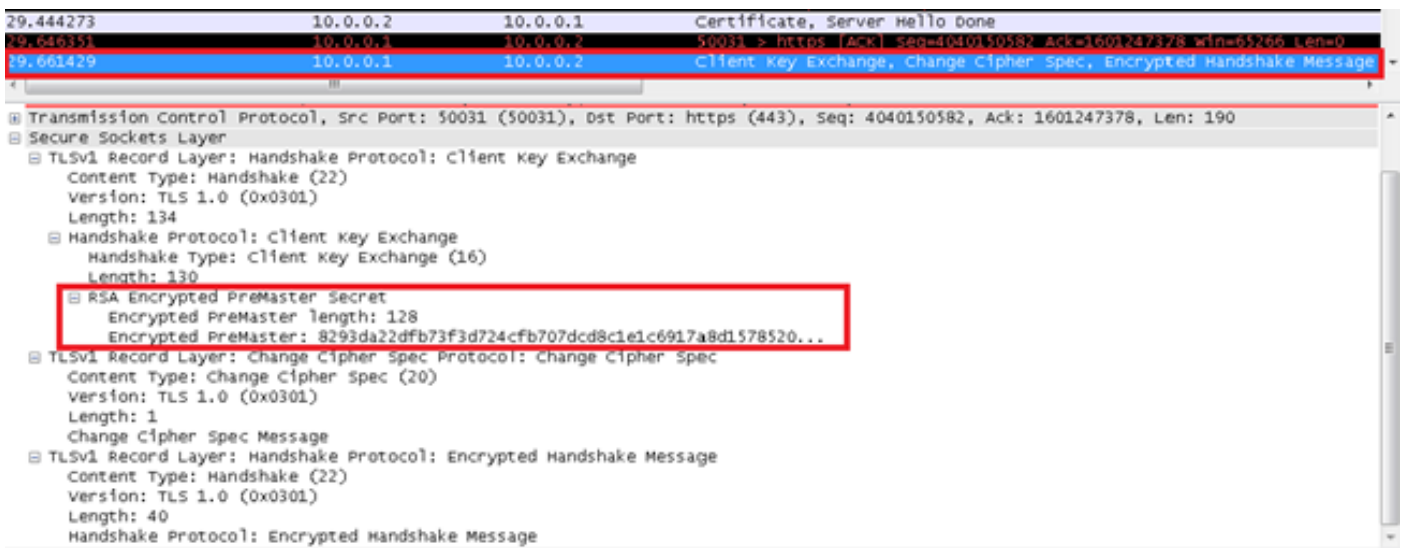
客户端Exchange

客户端证书(可选)

这是客户端发送的第一条消息，在他或她收到一个服务器问候完成的消息后。此信息，如果服务器请求认证，只传送。如果适当的认证不是可用的，客户端发送一次no_certificate戒备。此戒备是仅警告;然而，如果需要，服务器也许回应一次致命握手故障戒备客户端验证。客户端DH证书必须匹配服务器指定的DH参数。

客户端密匙交换

此消息内容取决于公共密钥算法选择在客户端Hello和服务器问候消息之间。客户端使用Rivest沙米尔Addleman (RSA)算法加密的一个premaster键或DH关键协议和认证。当RSA使用服务器验证和密匙交换时，48字节pre_master_secret是由客户端生成的，被加密在服务器公共密钥下，并且被发送到服务器。服务器使用专用密钥为了解码pre_master_secret。两个当事人然后转换pre_master_secret成master_secret。



Certificate verify (可选)

如果客户端发送与签署的能力的一个认证，传送数字式地签字的Certificate verify信息为了明确地验证认证。

密码更改

更改密码Spec消息

客户端传送更改密码Spec信息，并且客户端复制待定密码Spec (新的)到当前密码Spec (以前使用)的那个。更改密码Spec协议存在为了在加密的策略的信号转换。协议包括单个消息，被加密并且被压缩在当前(不是待定)密码Spec下。两个传送信息客户端和服务器为了通知接收方随后的记录保护在最近协商的密码Spec和键下。此消息的接收造成接受器复制被读取等待状态到读的当前状态。客户端在握手密匙交换和Certificate verify消息以后传送更改密码Spec信息(若有)，并且服务器发送一，在从客户端收到的顺利地处理密匙交换消息后。当上次会话恢复时，更改密码Spec信息在hello消息以后传送。在捕获，客户端Exchange，更改密码和完成的信息传送作为单个消息从客户端。

完成的消息

总是传送完成的信息，在更改密码Spec消息为了验证之后密匙交换和认证过程是成功的。完成的消息是与最近协商的算法、键和秘密的第一个保护的信息包。没有需要完成的消息的确认;在他们传送完成的信息之后，当事人能开始发送加密的数据。完成的消息的接收人必须验证内容是正确的。

29.444273	10.0.0.2	10.0.0.1	Certificate, Server Hello done
29.646351	10.0.0.1	10.0.0.2	50031 > https [ACK] Seq=4040150582 Ack=1601247378 win=65766 len=0
29.661429	10.0.0.1	10.0.0.2	client key exchange, change cipher spec, Encrypted Handshake Message

Transmission Control Protocol, Src Port: 50031 (50031), Dst Port: https (443), Seq: 4040150582, Ack: 1601247378, Len: 190			
Secure Sockets Layer			
TLSv1 Record Layer: Handshake Protocol: Client Key Exchange			
Content Type: Handshake (22)			
Version: TLS 1.0 (0x0301)			
Length: 134			
Handshake Protocol: Client Key Exchange			
Handshake Type: Client Key Exchange (16)			
Length: 130			
RSA Encrypted PreMaster Secret			
Encrypted PreMaster length: 128			
Encrypted PreMaster: 8293da22dfb73f3d724cfb707dc08c1e1c6917a8d1578520			
TLSv1 Record Layer: Change Cipher Spec Protocol: Change Cipher Spec			
Content Type: Change Cipher Spec (20)			
Version: TLS 1.0 (0x0301)			
Length: 1			
Change Cipher Spec Message			
TLSv1 Record Layer: Handshake Protocol: Encrypted Handshake Message			
Content Type: Handshake (22)			
Version: TLS 1.0 (0x0301)			
Length: 40			
Handshake Protocol: Encrypted Handshake Message			

Related Information

- [RFC 6101 -安全套接字协议层协议版本3.0](#)
- [Wireshark SSL wiki -请解码与Wireshark的SSL信息包](#)
- [Technical Support & Documentation - Cisco Systems](#)