Beijing Forest Studio

北京理工大学信息系统及安全对抗实验中心

HTTPS浅析

王子文　硕士研究生

2018年03月18日

内容提要

- 背景简介
- 基本概念
- HTTPS的发展
- TLS/SSL原理
- 参考文献

背景简介

- 我们经常接触到的就有邮箱登录，网上购物，电子银行等等，大部分均基于 HTTP 协议。
- 但HTTP[RFC2616]最初应用于INTERNET时没有使用密码，安全性很低，因此随着人们对安全性需求的提高，为用户提供面向通道安全的HTTPS协议应运而生。

基本概念

- HTTP协议

  HTTP是Hyper Text Transfer Protocol（超文本传输协议）的缩写。是用于从WWW服务器传输超文本到本地浏览器的传送协议，其规定了客户端和服务器请求和应答的标准。

- SSL/TLS协议

  SSL（Secure Sockets Layer，安全套接层），及其继任者 TLS（Transport Layer Security，传输层安全）是为网络通信提供安全及数据完整性的一种安全协议。

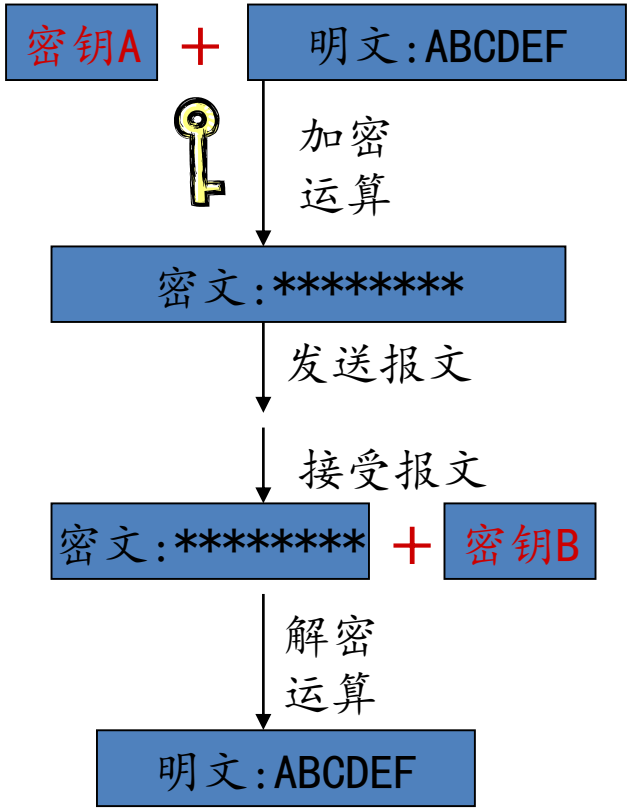
基本概念

- HTTPS

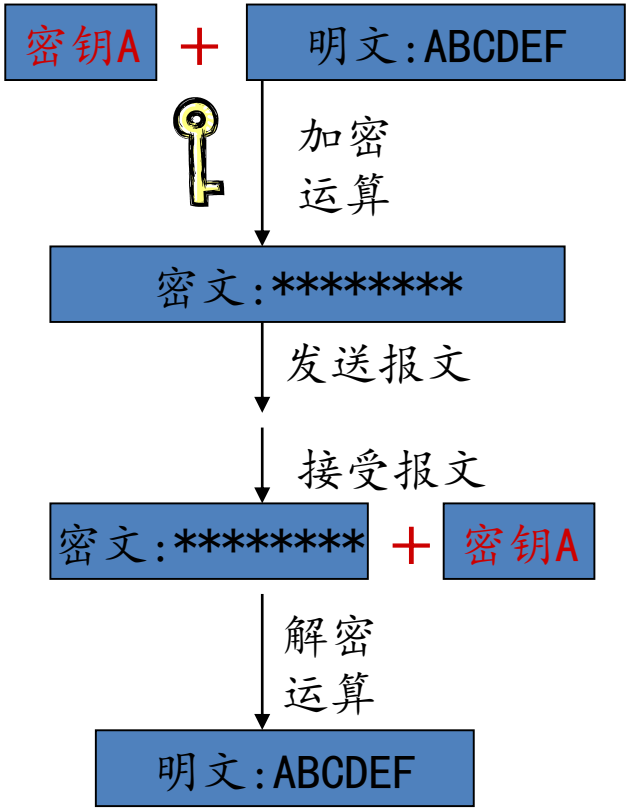
HTTPS：安全超文本传送协议，（HTTPS，Hyper Text Transfer Protocol over Secure Socket Layer），是以安全为目标的 HTTP 通道，简单讲是 HTTP 的安全版。即 HTTP 下加入 TLS 层，HTTPS 的安全基础是 TLS，因此加密的详细内容就需要 TLS.

基本概念

- 加密过程



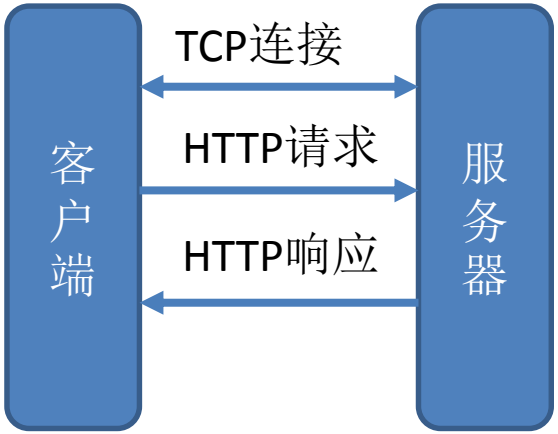
非对称加密



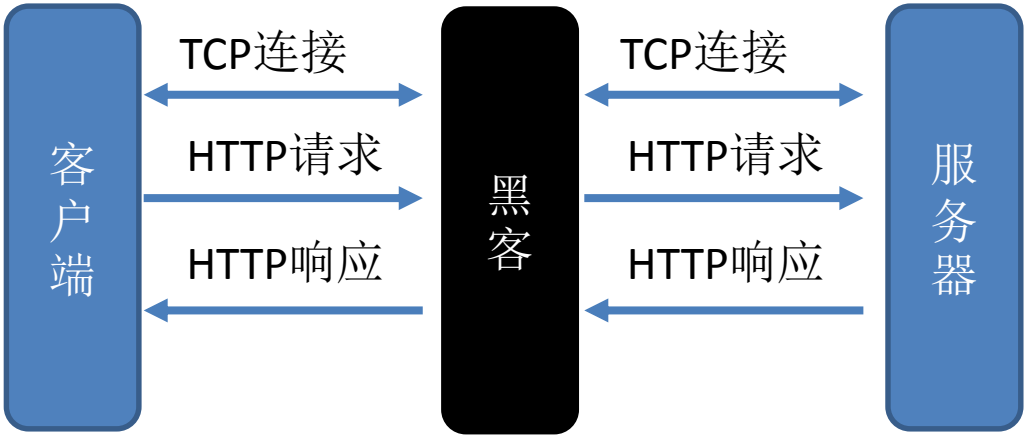
对称加密

HTTPS的发展

- HTTP的访问过程：



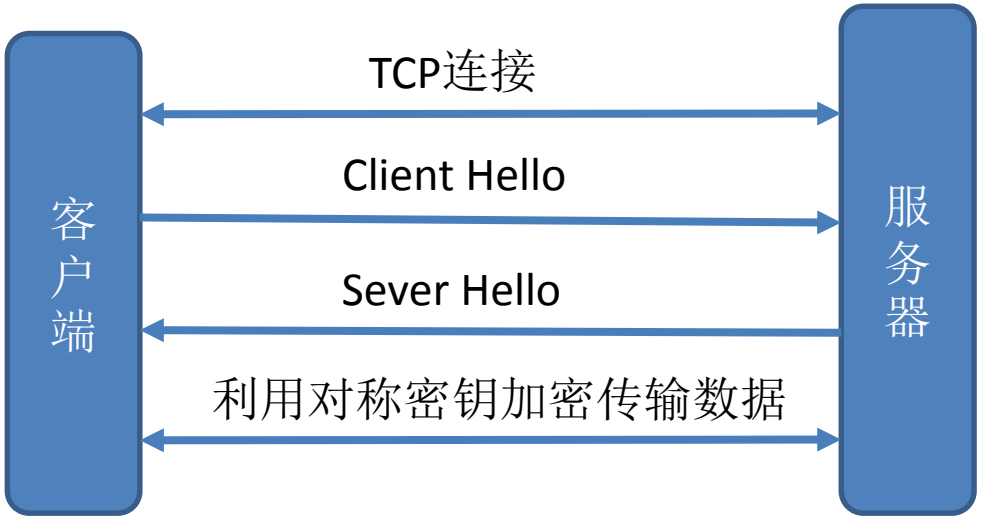
HTTP请求过程中，客户端和服务器无需身份验证。并且数据是以明文的方式传输的。



客户端发出的请求很容易被黑客截获，如果此时黑客冒充服务器，则其可返回任意信息给客户端，而不被客户端察觉。

HTTPS的发展

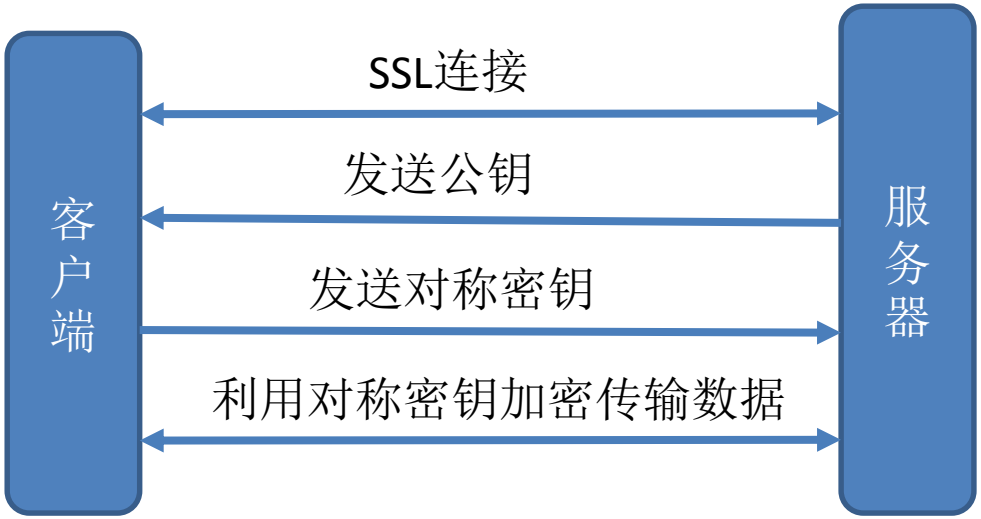
- 对称密钥对信息加密



此种方式属于对称加密，双方拥有相同的密钥，信息得到安全传输，但此种方式的缺点是：
（1）不同的客户端、服务器数量庞大，所以双方都需要维护大量的密钥，维护成本很高
（2）因每个客户端、服务器的安全级别不同，密钥极易泄露

HTTPS的发展

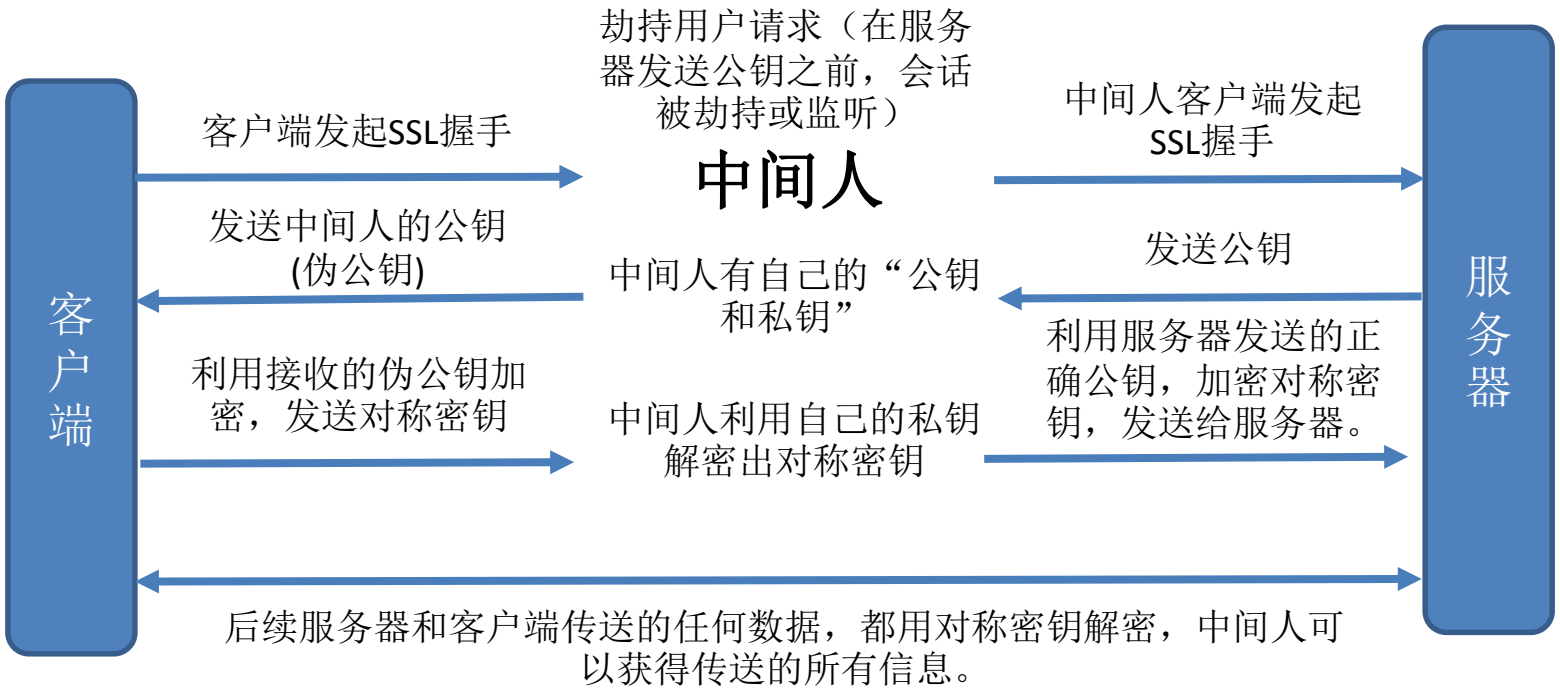
- 非对称密钥对信息加密



在非对称加解密算法里，公钥加密的数据，有且只有唯一的私钥才能够解密，而私钥只部署在服务器中，其他任何人都没有私钥，因此，只有服务器才能够解密。

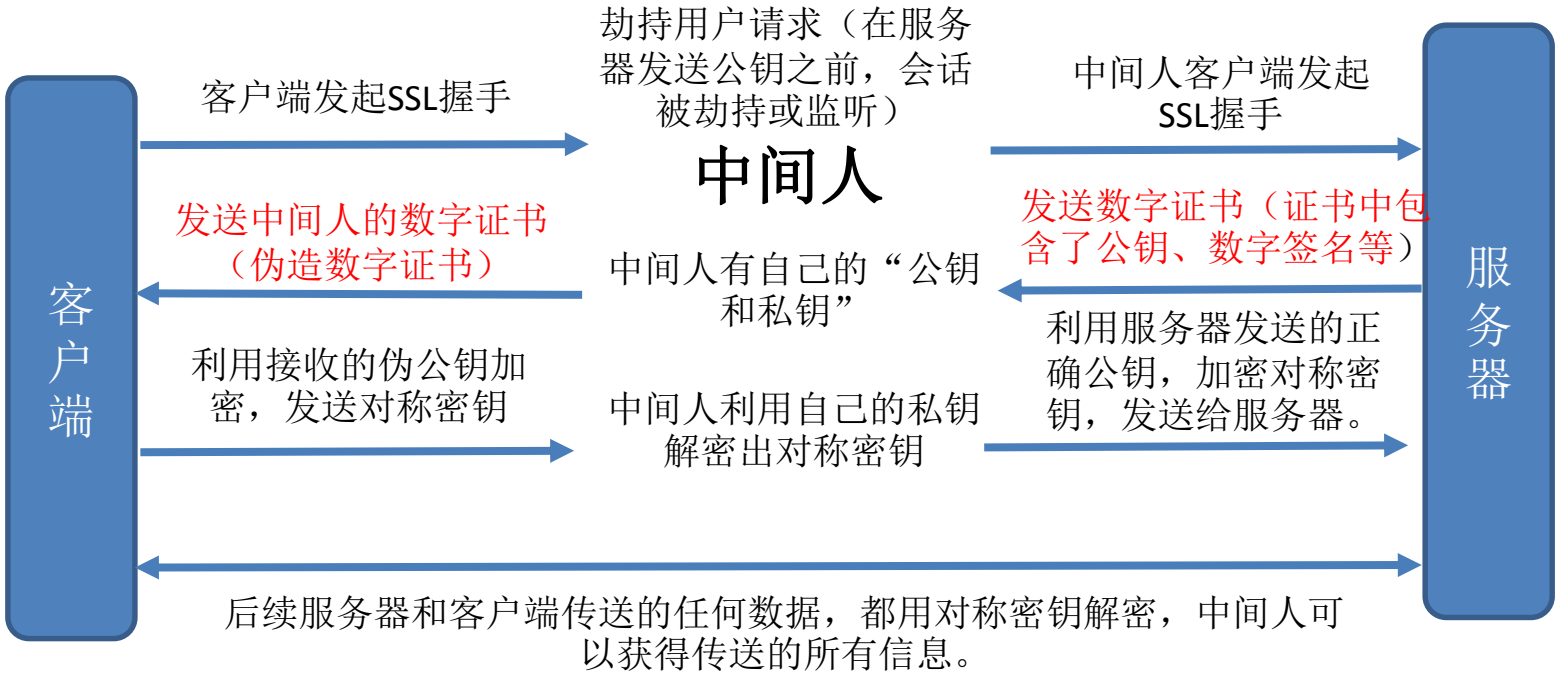
HTTPS的发展

- 中间人攻击



HTTPS的发展

引入数字证书



客户端对数字证书进行CA校验：

1. 校验成功，则表明是服务器正确的数字证书，客户端利用公钥加密发送对称密钥。

2. 校验失败，会话终止。

非对称加密算法（公钥和私钥）交换对称密钥+数字证书验证身份（验证公钥是否是伪造的）+利用对称密钥加解密后续传输的数据=安全。

TLS/SSL工作原理

- TLS握手协议

TLS 握手协议是TLS协议中最重要同时也是最复杂的协议。TLS 握手协议主要负责算法协商、身份验证和确定密钥。

TLS握手过程分为四种:

- Full Handshake：全流程握手，C/S双方从无到有建立TLS连接；
- Resume session Handshake： C/S双方曾经建立过连接，但中途断了，TLS会话信息还有保留，只要执行部分握手流程就可建立TLS连接；
- Server Re-negotiation Handshake：已经建立了TLS连接，但server端为了某些原因(比如安全性)要求重新对密钥进行协商，也只需要执行部分握手流程；
- Client Re-negotiation Handshake：已经建立TLS连接，但client端为了某些原因要求重新协商，只需执行部分握手流程。

TLS/SSL工作原理

- TLS/SSL握手协议

TLS/SSL工作原理

- Client Hello

```
▲ Handshake Protocol: Client Hello
    Handshake Type: Client Hello (1)
    Length: 210
    Version: TLS 1.2 (0x0303)
  ▲ Random: e5342044f1829eae2ac306050cc942eed26b80574d8e7b0d...
      GMT Unix Time: Nov  9, 2091 06:16:36.000000000 中国标准时间
      Random Bytes: f1829eae2ac306050cc942eed26b80574d8e7b0df19ec4a5...
    Session ID Length: 0
    Cipher Suites Length: 36
  ▲ Cipher Suites (18 suites)
      Cipher Suite: Reserved (GREASE) (0xeaea)
      Cipher Suite: TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256 (0xcca9)
      Cipher Suite: TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256 (0xcca8)
      Cipher Suite: TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256 (0xcc14)
      Cipher Suite: TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256 (0xcc13)
      Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 (0xc02b)
      Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xc02f)
      Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (0xc02c)
      Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xc030)
      Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA (0xc009)
      Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (0xc013)
      Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA (0xc00a)
      Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014)
```

版本号：TLS 1.2

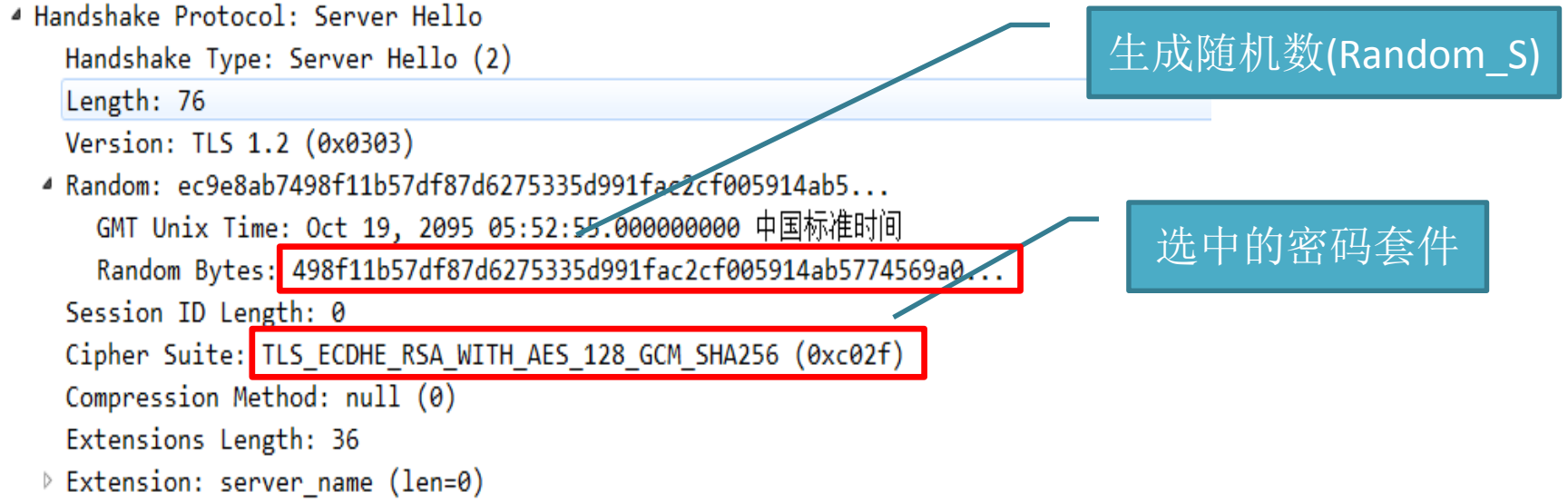
生成随机数(Random_C)

会话ID

密码套件列表

TLS/SSL工作原理

- Sever Hello



Sever Hello 消息的意义是将服务器选择的连接参数传送回客户端。消息结构与Client Hello类似，只是每个字段只包含一个选项。

TLS/SSL工作原理

- Certificate

```
▲ Handshake Protocol: Certificate
    Handshake Type: Certificate (11)
    Length: 3666
    Certificates Length: 3663
  ▲ Certificates (3663 bytes)
      Certificate Length: 2524
    ▲ Certificate: 308209d8308208c0a003020102020c4f58ebff10c9da8ce5... (id-at-commonName=baidu.com,id-at-organi
      ▲ signedCertificate
          version: v3 (2)
          serialNumber: 0x4f58ebff10c9da8ce591c22e
        ▷ signature (sha256WithRSAEncryption)
        ▷ issuer: rdnSequence (0)
        ▷ validity
        ▷ subject: rdnSequence (0)
        ▲ subjectPublicKeyInfo
          ▲ algorithm (rsaEncryption)
              Algorithm Id: 1.2.840.113549.1.1.1 (rsaEncryption)
          ▲ subjectPublicKey: 3082010a0282010100c8990f0b42debfa2f4b21358dce4ce...
              modulus: 0x00c8990f0b42debfa2f4b21358dce4cee49c0ef20c739b65...
              publicExponent: 65537
```

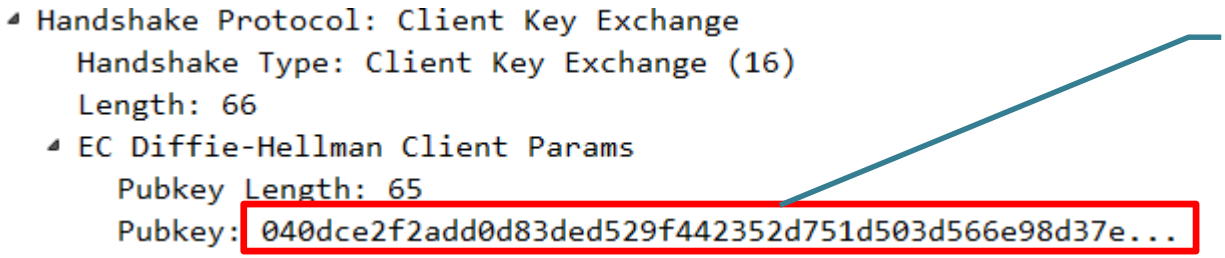
将证书中的公钥发送给客户端

- Sever Hello Done

```
▲ Handshake Protocol: Server Hello Done
    Handshake Type: Server Hello Done (14)
    Length: 0
```

TLS/SSL工作原理

- Client Key Exchange



将生成的随机数用公钥加密

解析证书：

1. 验证证书是否有效，如验证颁发机构、过期时间等。
2. 生成一个随机值（预主密钥Pre-Master）。
3. 将这个随机值用从服务器获得的公钥进行加密。

至此，服务器和客户端都拥有了计算私钥所用的的信息：
两个明文随机数：Random_C和Random_S，预主密钥Pre-Master。
最终的密钥（对称密钥）key = Func(Random_C, Random_S, Pre-Master)。

TLS/SSL工作原理

- Change Cipher Spec

```
TLSv1.2 Record Layer: Change Cipher Spec Protocol: Change Cipher Spec
    Content Type: Change Cipher Spec (20)
    Version: TLS 1.2 (0x0303)
    Length: 1
    Change Cipher Spec Message
```

- Finished

```
TLSv1.2 Record Layer: Handshake Protocol: Encrypted Handshake Message
    Content Type: Handshake (22)
    Version: TLS 1.2 (0x0303)
    Length: 40
    Handshake Protocol: Encrypted Handshake Message
```

Change Cipher Spec：发送一个不加密的信息，客户端使用该信息通知服务器后续的通信都采用协商的通信密钥和加密算法进行加密通信。

Finished：此消息的发送意味着握手已经完成，消息内容将加密。把从client hello一直到现在的数据摘要，可以让server验证握手过程是否被第三方篡改。

对比分析

- HTTP与HTTPS对比

1. https协议需要到CA申请证书，一般免费证书较少，因而需要一定费用。
2. http是超文本传输协议，信息是明文传输，https则是具有安全性的TLS加密传输协议。
3. http和https使用的端口不一样，前者是80，后者是443。
4. http的连接很简单，是无状态的；HTTPS协议是由SSL+HTTP协议构建的可进行加密传输、身份认证的网络协议，比http协议安全。

参考文献

[1]Ivan Ristic. HTTPS权威指南 [M]. 人民邮电出版社, 2016
[2]http://blog.csdn.net/tenfyguo/article/details/5802682
[3]https://baijiahao.baidu.com/s?id=1570143475599137&wfr=spider&for=pc.
[4]http://blog.csdn.net/tenfyguo/article/details/5802682

谢谢！

大成若缺，其用不弊。大盈若冲，其用不穷。大直若屈。大巧若拙。大辩若讷。静胜躁，寒胜热。清静为天下正。