

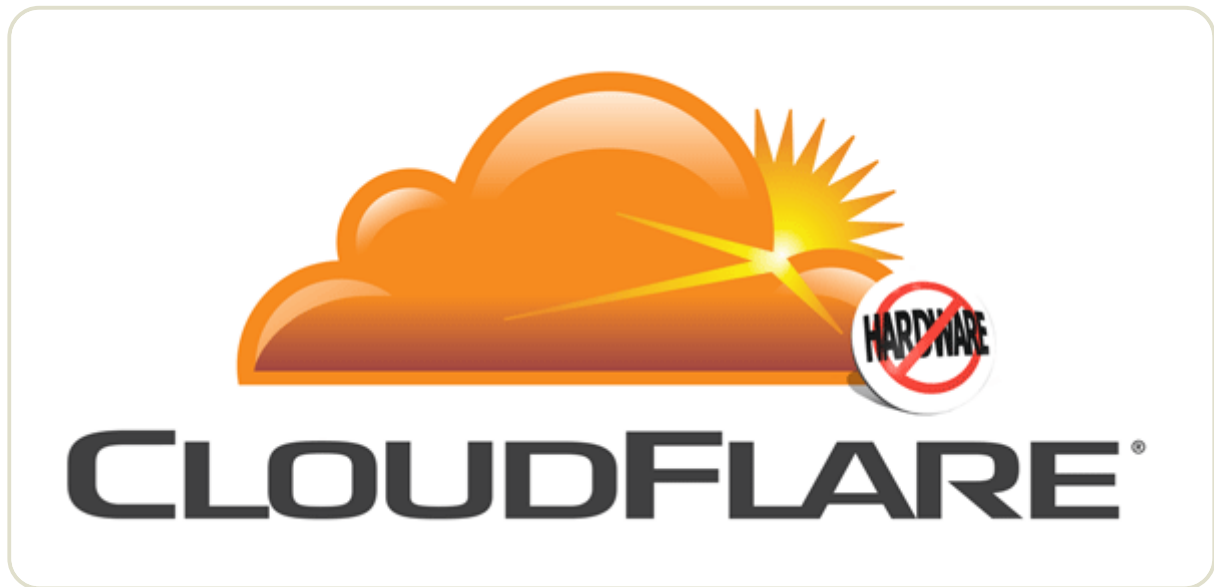
# 图解SSL/TLS协议

---

作者： 阮一峰

日期： 2014年9月20日

本周，[CloudFlare](#)宣布，开始提供Keyless服务，即你把网站放到它们的CDN上，不用提供自己的私钥，也能使用SSL加密链接。



我看了CloudFlare的说明（[这里](#)和[这里](#)），突然意识到这是绝好的例子，可以用来说明SSL/TLS协议的运行机制。它配有插图，很容易看懂。

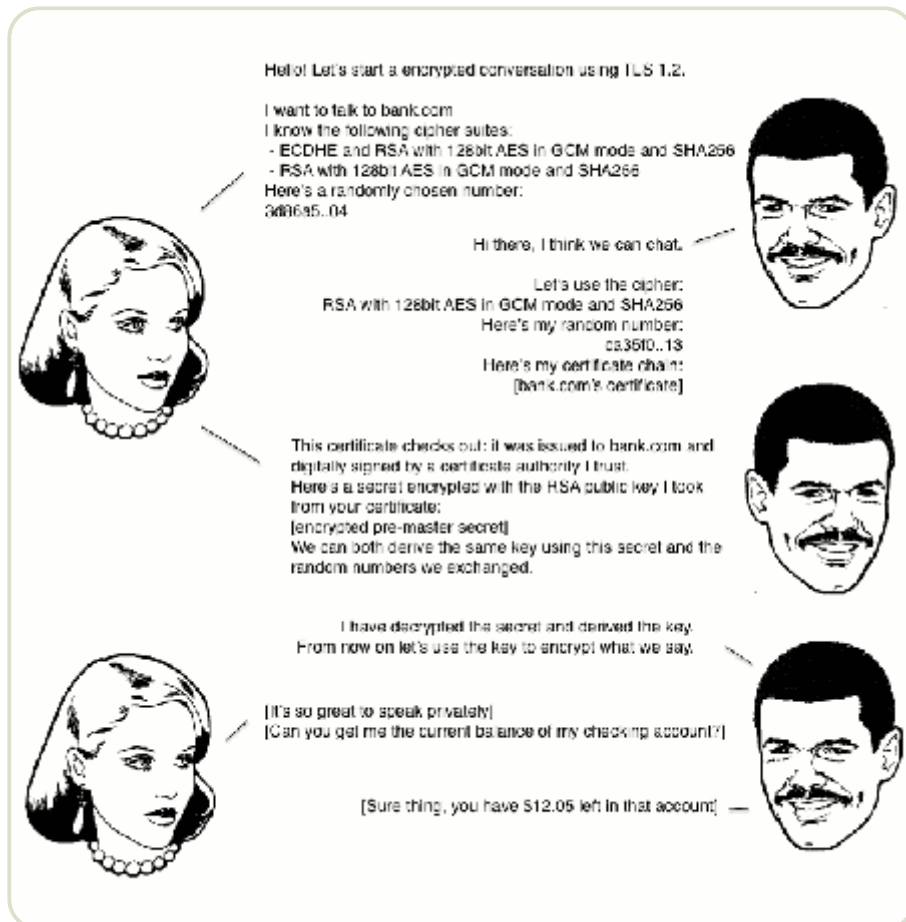
下面，我就用这些图片作为例子，配合我半年前写的[《SSL/TLS协议运行机制的概述》](#)，来解释SSL协议。

## 一、SSL协议的握手过程

---

开始加密通信之前，客户端和服务端首先必须建立连接和交换参数，这个过程叫做握手（handshake）。

假定客户端叫做爱丽丝，服务器叫做鲍勃，整个握手过程可以用下图说明（点击看大图）。



握手阶段分成五步。

第一步，爱丽丝给出协议版本号、一个客户端生成的随机数（Client random），以及客户端支持的加密方法。

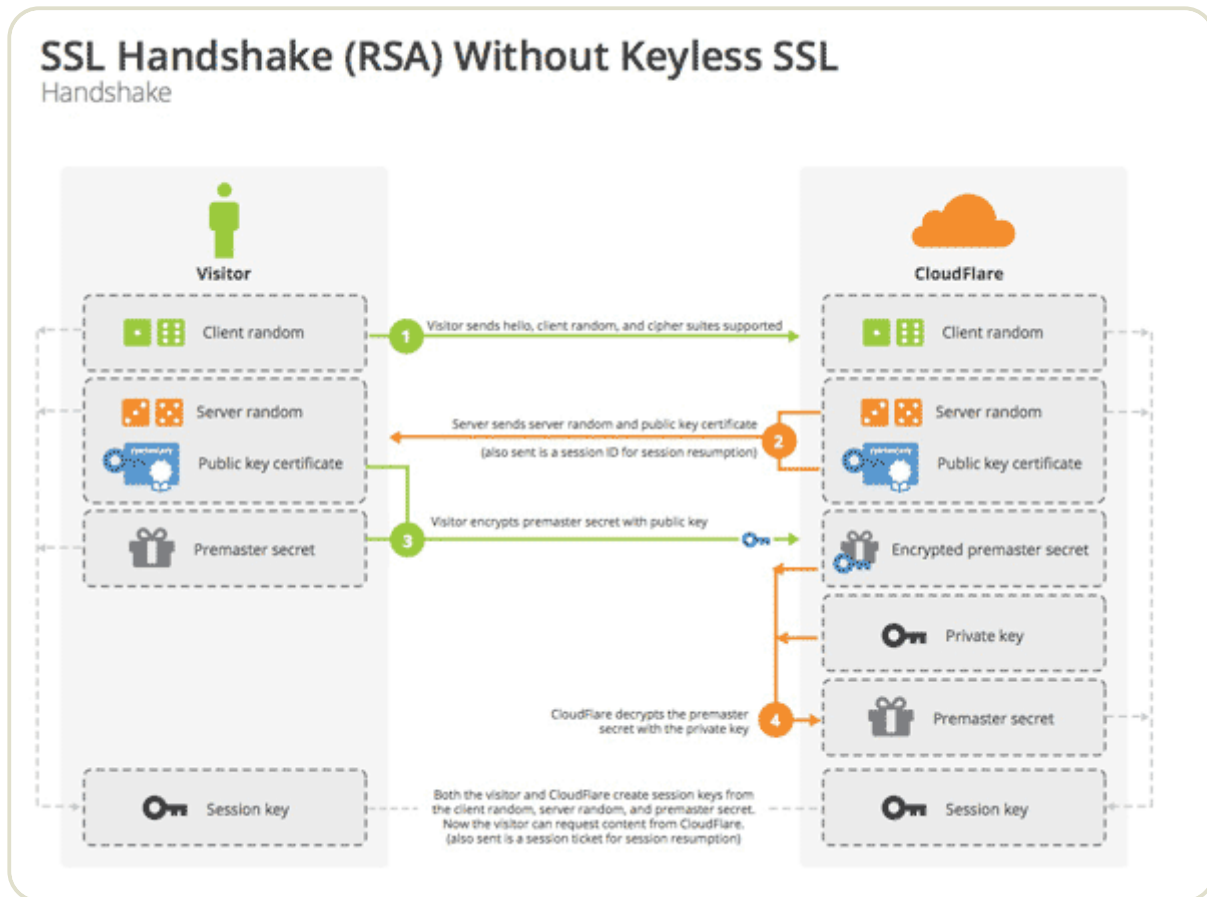
第二步，鲍勃确认双方使用的加密方法，并给出数字证书、以及一个服务器生成的随机数（Server random）。

第三步，爱丽丝确认数字证书有效，然后生成一个新的随机数（Premaster secret），并使用数字证书中的公钥，加密这个随机数，发给鲍勃。

第四步，鲍勃使用自己的私钥，获取爱丽丝发来的随机数（即Premaster secret）。

第五步，爱丽丝和鲍勃根据约定的加密方法，使用前面的三个随机数，生成"对话密钥"（session key），用来加密接下来的整个对话过程。

上面的五步，画成一张图，就是下面这样。



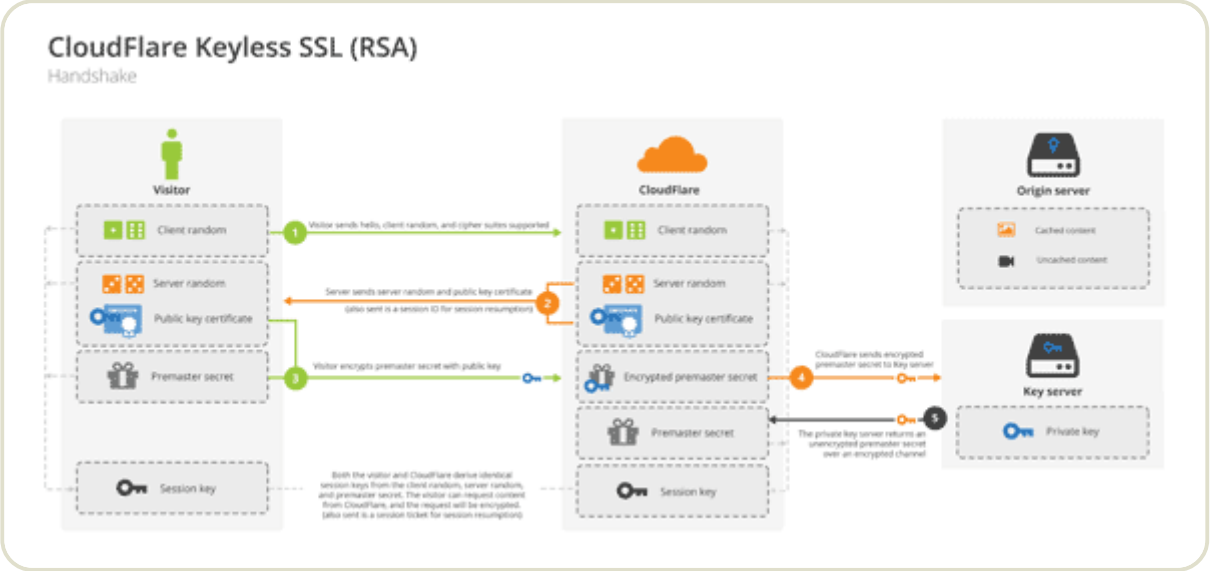
## 二、私钥的作用

握手阶段有三点需要注意。

- (1) 生成对话密钥一共需要三个随机数。
- (2) 握手之后的对话使用"对话密钥"加密（对称加密），服务器的公钥和私钥只用于加密和解密"对话密钥"（非对称加密），无其他作用。
- (3) 服务器公钥放在服务器的数字证书之中。

从上面第二点可知，整个对话过程中（握手阶段和其后的对话），服务器的公钥和私钥只需要用到一次。这就是CloudFlare能够提供Keyless服务的根本原因。

某些客户（比如银行）想要使用外部CDN，加快自家网站的访问速度，但是出于安全考虑，不能把私钥交给CDN服务商。这时，完全可以把私钥留在自家服务器，只用来解密对话密钥，其他步骤都让CDN服务商去完成。



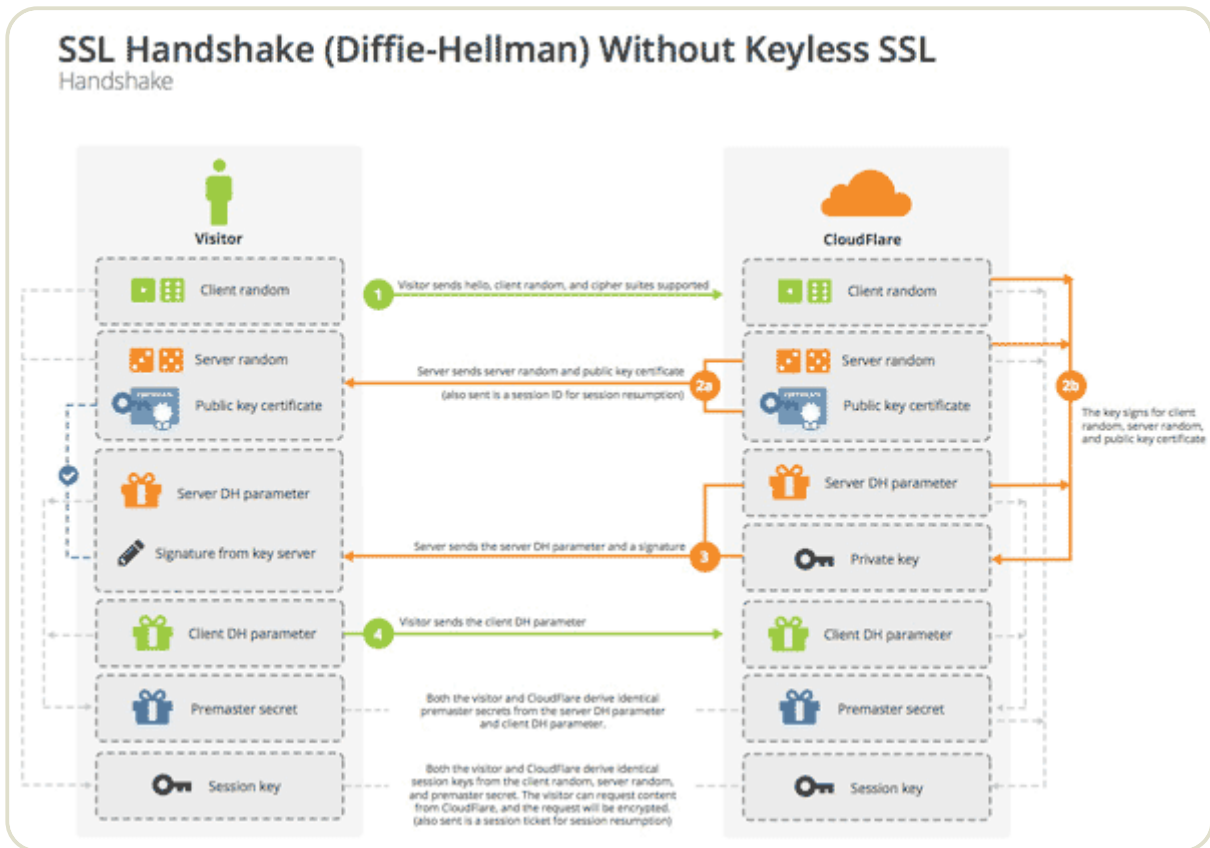
上图中，银行的服务器只参与第四步，后面的对话都不再会用到私钥了。

### 三、DH算法的握手阶段

整个握手阶段都不加密（也没法加密），都是明文的。因此，如果有人窃听通信，他可以知道双方选择的加密方法，以及三个随机数中的两个。整个通话的安全，只取决于第三个随机数（Premaster secret）能不能被破解。

虽然理论上，只要服务器的公钥足够长（比如2048位），那么Premaster secret可以保证不被破解。但是为了足够安全，我们可以考虑把握手阶段的算法从默认的RSA算法，改为 Diffie-Hellman算法（简称DH算法）。

采用DH算法后，Premaster secret不需要传递，双方只要交换各自的参数，就可以算出这个随机数。



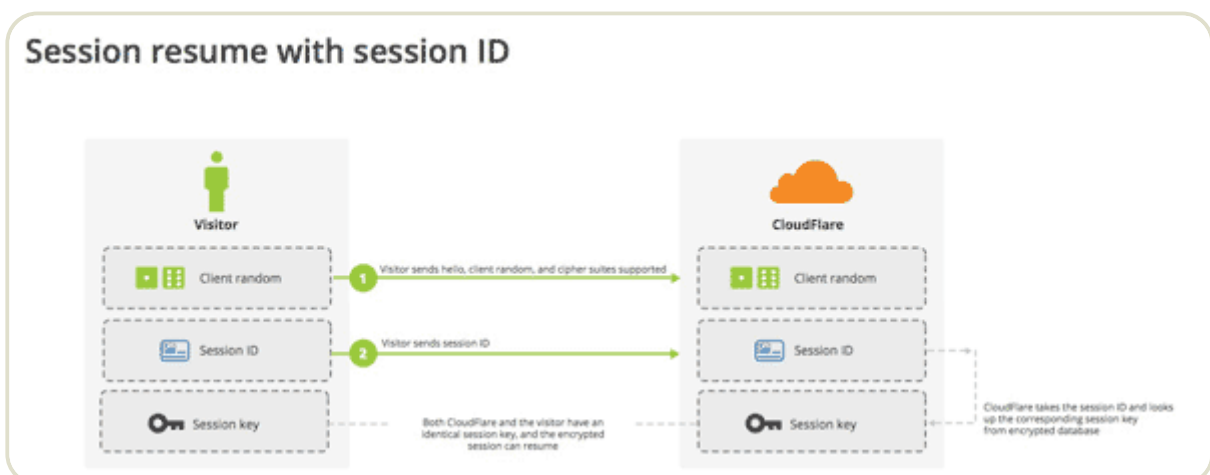
上图中，第三步和第四步由传递Premaster secret变成了传递DH算法所需的参数，然后双方各自算出Premaster secret。这样就提高了安全性。

## 四、session的恢复

握手阶段用来建立SSL连接。如果出于某种原因，对话中断，就需要重新握手。

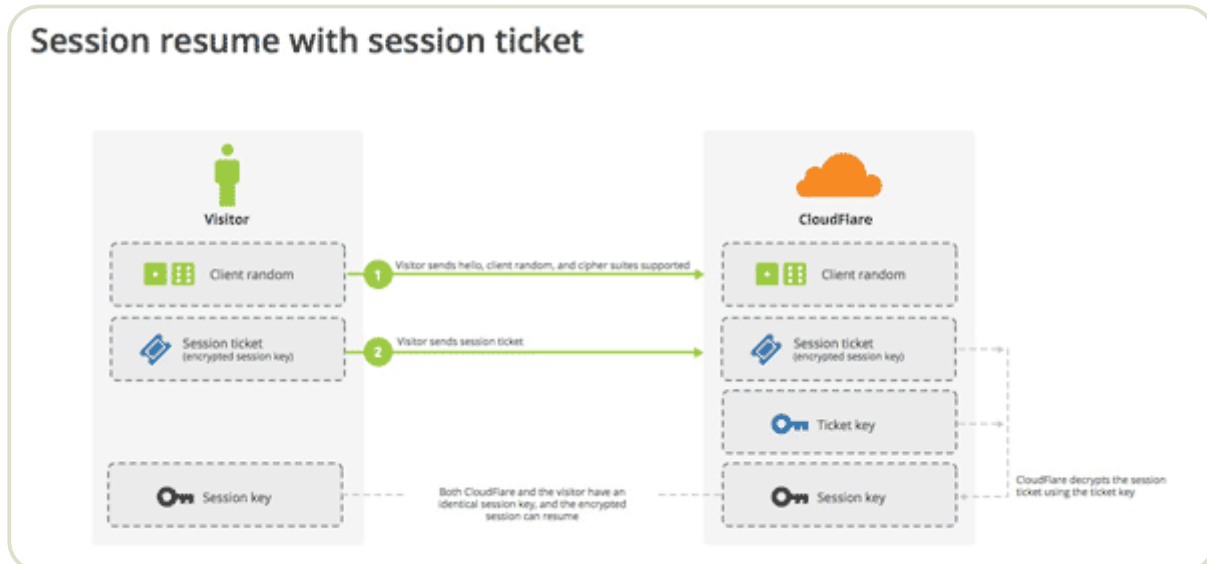
这时有两种方法可以恢复原来的session：一种叫做session ID，另一种叫做session ticket。

session ID的思想很简单，就是每一次对话都有一个编号（session ID）。如果对话中断，下次重连的时候，只要客户端给出这个编号，且服务器有这个编号的记录，双方就可以重新使用已有的"对话密钥"，而不必重新生成一把。



上图中，客户端给出session ID，服务器确认该编号存在，双方就不再进行握手阶段剩余的步骤，而直接用已有的对话密钥进行加密通信。




session ID是目前所有浏览器都支持的方法，但是它的缺点在于session ID往往只保留在一台服务器上。所以，如果客户端的请求发到另一台服务器，就无法恢复对话。session ticket就是为了解决这个问题而诞生的，目前只有Firefox和Chrome浏览器支持。



上图中，客户端不再发送session ID，而是发送一个服务器在上一次对话中发送过来的session ticket。这个session ticket是加密的，只有服务器才能解密，其中包括本次对话的主要信息，比如对话密钥和加密方法。当服务器收到session ticket以后，解密后就不必重新生成对话密钥了。

(完)

## 文档信息

- 版权声明：自由转载-非商用-非衍生-保持署名（创意共享3.0许可证）
- 发表日期：2014年9月20日
- 更多内容：档案 » 开发者手册
- 博客文集：《寻找思想之路》，《未来世界的幸存者》
- 社交媒体： twitter,  weibo
- Feed订阅：



## 相关文章

- **2017.04.13:** [Emoji 简介](#)

一、含义 Emoji 是可以插入文字的图形符号。

- **2017.04.05:** [CSS in JS 简介](#)

1、以前，网页开发有一个原则，叫做"关注点分离"（separation of concerns）。

- **2017.03.03:** [技术的热门度曲线](#)

全球最大的 IT 咨询公司高德纳（Gartner），有一个"技术热门度曲线"模型（Gartner Hype Cycle）。

- **2017.02.22:** [函数式编程入门教程](#)

你可能听说过函数式编程（Functional programming），甚至已经使用了一段时间。

## 广告（购买广告位）



