

【转载】通俗大白话，彻底看懂 https 原理本质（ssl证书）\\_证书序列号和证书指纹一样吗-CSDN博客  
=====

![Image 1](https://img-home.csdnimg.cn/images/20201124032511.png)](https://www.csdn.net/)

\* [博客](https://blog.csdn.net/)  
\* [下载](https://download.csdn.net/)  
\* [学习](https://edu.csdn.net/?utm\_source=zhuzhantoolbar)  
\* [社区](https://bbs.csdn.net/)  
\* [![Image 2](https://img-home.csdnimg.cn/images/20241022035258.png)C知道](https://so.csdn.net/chat?utm\_source=vip\_chatgpt\_common\_pc\_toolbar)  
\* [![Image 3](https://img-home.csdnimg.cn/images/20240829093757.png)GitCode](https://link.csdn.net/?target=https%3A%2F%2Fgitcode.com%3Futm\_source%3Dcsdn\_toolbar)  
\* [InsCode](https://inscode.csdn.net/?utm\_source=260232576)  
\* [会议](https://summit.csdn.net/)

搜索

登录

登录后您可以：

\* 免费复制代码  
\* 关注/点赞/评论/收藏  
\* 下载海量资源  
\* 写文章/发动态/加入社区

立即登录

[会员12.12 !][Image 4](https://img-home.csdnimg.cn/images/20241125050229.gif)](https://mall.csdn.net/vip)

[消息](https://i.csdn.net/#/msg/index)

[历史](https://i.csdn.net/#/user-center/history)

[创作中心](https://mp.csdn.net/ "创作中心")

![Image 5](https://img-home.csdnimg.cn/images/20230825101811.png)](https://mp.csdn.net/edit) !  
[Image 6](https://img-home.csdnimg.cn/images/20230815023238.png)

[发布](https://mp.csdn.net/edit)

【转载】通俗大白话，彻底看懂 https 原理本质（ssl证书）  
=====

最新推荐文章于 2023-03-27 19:25:20 发布

![Image 7](https://csdnimg.cn/release/blogv2/dist/pc/img/reprint.png)

[Blueeyedboy521](https://blog.csdn.net/Blueeyedboy521 "Blueeyedboy521") ![Image 8](https://csdnimg.cn/release/blogv2/dist/pc/img/newCurrentTime2.png) 最新推荐文章于 2023-03-27 19:25:20 发布

![Image 9](https://csdnimg.cn/release/blogv2/dist/pc/img/articleReadEyes2.png) 阅读量575 ![Image 10](https://csdnimg.cn/release/blogv2/dist/pc/img/tobarCollect2.png) ![Image 11](https://csdnimg.cn/release/blogv2/dist/pc/img/tobarCollectionActive2.png) 收藏 2

![Image 12](https://csdnimg.cn/release/blogv2/dist/pc/img/newHeart2023Active.png) ![Image 13](https://csdnimg.cn/release/blogv2/dist/pc/img/newHeart2023Black.png) 点赞数 3

分类专栏： [开发工具](https://blog.csdn.net/blueeyedboy521/category\_11722803.html) 文章标签：

[https](https://so.csdn.net/so/search/s.do?q=https&t=all&o=vip&s=&l=&f=&viparticle=&from\_tracking\_code=tag\_word&from\_code=app\_blog\_art) [ssl](https://so.csdn.net/so/search/s.do?q=ssl&t=all&o=vip&s=&l=&f=&viparticle=&from\_tracking\_code=tag\_word&from\_code=app\_blog\_art)

原文链接: [https://www.toutiao.com/article/7161592520668054051]  
(https://www.toutiao.com/article/7161592520668054051)

版权

[![Image 14](https://i-blog.csdnimg.cn/columns/default/20201014180756930.png?x-oss-process=image/resize,m\_fixed,h\_224,w\_224) 开发工具 专栏收录该内容]  
(https://blog.csdn.net/blueeyedboy521/category\_11722803.html "开发工具")

24 篇文章 6 订阅

订阅专栏

原文

[通俗大白话，彻底弄懂 https 原理本质](https://www.toutiao.com/article/7161592520668054051)

通俗大白话，彻底弄懂 https 原理本质

相信很多人，对 https 的过程弄不清楚，只是知道 https是安全加密的，背后的原理，过程并不清楚  
笔者曾经也是对https的过程并不清楚，一知半解，而且最可气的是每次面试，面试官很可能就问你这个问题  
每次都答不对或者答的面试官不满意，说来说去，还是自己没有真正理解  
其实https 的原理过程，并没有那么复杂，只是有些文章没有说清楚，这样的文章看多了，就迷糊了。  
在了解https原理的过程之前，我们先来了解一下加密的知识

一 加密知识

加密按照加密方式，可以分为以下三种方式

#### ### 1.1 单向加密

也叫做不可逆加密，对明文的加密产生一个密文，并不能再通过密文，解出来对应的明文  
一般用于产生消息摘要，密钥加密等，常见的单向加密有：

- \* MD5 ： 相信这个大家都熟悉，一个明文，md5以后，对应一个唯一的密文
- \* SHA ： 其中又分为 sha192 ， sha256

**\*\*特点\*\*：**

- \* 不可逆
- \* 输入一样，输出必然相同

#### ### 1.2 对称加密

对称加密，用一个密钥，对明文进行加密，同理，同这把密钥，也可以对密文进行解密

也就是说加密和解密，可以用同一个密钥

这种加密方法就是 对称加密

常用的对称加密方法有：

- \* DES
- \* 3DES
- \* AES

**\*\*特点\*\*：**

- \* 加密方和解密使用同一密钥

- \* 加密解密的速度比较快

### ### 1.3 非对称加密

我们知道，对称加密使用同一把密钥，相反，非对称加密，使用公钥和私钥进行加密解密

可以使用私钥加密，公钥进行解密，同理，也可以使用公钥加密，私钥进行解密

常见非对称加密方式的有：

- \* RSA
- \* DSA
- 我们平时最常用的就是 RSA

**\*\*特点\*\*：**

- \* 使用两把密钥进行加密和解密，即公钥和私钥
- \* 公钥加密私钥解密，私钥加密公钥可以解密
- \* 加密或者解密，速度非常慢
- \* 私钥和公钥是成对出现的

## 二 加密知识总结

**\\*\\*单向加密：**\\*\\*不可逆，只要输入的内容一样，输出的密文一定是一样的，有任何修改，产生的密文都是不同的

**\\*\\*对称加密：**\\*\\*加密和解密使用同一把密钥，加密解密速度特别快

非对称加密：使用公钥和私钥进行加密和解密，公钥加密私钥解，私钥加密公钥解。加密解密的过程非常慢

所谓公钥，就是可以公开给别人的

所谓私钥，就是不可以公开给别人，是自己私有保留的。

注：以上内容，纯粹是加密的知识，和https没有任何关系。下面我们开始讲解 https的过程。我们先看一个需求解决了这个需求，就明白了https的过程了。

### ### 从一个需求开始

假设有这样一个需求：小明和小花需要通信，少男少女写情书嘛，肯定不想让别人看到，所以需要安全的通信。

**### 问题一：**小明如何安全的把内容传给小花？

通过上面的加密知识的学习，我们很容易就想到，把通信的内容，给加密了就行了啊

答案是对的，把通信的内容给加密就行了。

**### 问题二：**使用哪种加密方式加密呢？

单向加密肯定不行，小花收到信，解不出来，这恋爱没法谈

对称加密可以，小花只要有密钥，就可以把内容解出来

非对称加密也可以，小明用自己的私钥加密，小花拿到小明的公钥，也可以把内容解出来

**### 问题三：**对称加密，非对称加密都可以，到底使用哪种呢？

通过上面的加密知识的学习，我们知道

对称加密速度快，非对称加密速度慢

那么对于小明，小花这两人来说，经常一聊就是几个小时，数据是非常多的

如果使用非对称加密，那估计得郁闷死，因为加密也慢，解密也慢，这两人肯定不会用非对称加密，要是我，我也不用，急死个人。

那么答案就是，使用对称加密方式，因为加密快啊，小明小花，都持有同一把密钥，双方互相都能解密出来对方发

的信。

总结：小明和小花通信，使用对称加密，假如密钥是 S，双方都使用同一把密钥 S 进行加密，解密

这样小明和小花就能愉快的通信了，而且内容是加密的，加密解密的速度也很快，这很美好。

但是这样有一个隐患，就是密钥 S，在传输的过程中，不小心被老王截获了

造成的后果就是：小明，小花以及老王，都有相同的密钥S了

那么，小明和小花之间没有秘密可言了，他们发的信，老王都能解开看，看完再加密，再发给小花，这还得了。

那么如何解决 密钥S 在传输的过程中，被别人截获的情况呢？

有人说，可以对称加密方式对密钥S 进行加密，再传输，那么此时的密钥S1 也是有被截获的风险啊

那就再对 S1 进行加密，再传输…，这样就无穷尽了。肯定是行不能的。

上面的方法肯定是不行了，现在的问题，变成了：小明如何把 密钥S 安全的传给小花，这是不是和之前的问题一  
小明如何安全的把内容传给小花？类似

所以，小明和小花如何要安全的通信，就需要使用对称加密 把信件内容加密传输

那么就得先解决一个问题：小明如何安全的把密钥S传输给小花？

### 问题四：小明如何安全的把密钥S传输给小花？

如果密钥S的传输过程不安全，那么后面的通信就是不安全的，反之，如何密钥S能安全的传输给小花，那么后面的通信就是安全的。

如果这是领导交待给我们这样一个活，我们使用自己学到的上面的加密知识，应该怎么解决呢？

通过上面的加密知识的学习，是不是有下面这样一个安全的加密传输方式

\* 小明使用非对称加密进行通信，首先小明生成了自己的一对私钥和公钥，为了方便，分别叫做 privateKey, publicKey  
\* 小明把 publicKey 给了小花  
\* \*\*方法一\*\*：小明用自己的privateKey，对 密钥S进行加密，加密后的密文S0传输给小花，小花用publicKey 对 S0解密出来 密钥S  
\* \*\*方法二\*\*：小花用publicKey对 密钥S 进行加密，加密后的密文 S0传输给小明，小明用 privatekey 对 S0解密出来 密钥S

上面，方法一 是不可行的，因为小明的 publicKey 是公开的，谁都可以下载，也就是说，老王也有小明的 publicKey，也可以对 S0进行解密出来 密钥S

方法二是可行的，因为privateKey只有小明有，小花用小明的公钥进行加密，只有小明能解开，其它任何人都解不开

所以上面的解决方案就是：

使用非对称加密 方式，对 密钥S 进行加密，进行传输

有人说，不对啊，非对称加密 性能不好，加密解密特别慢，要不刚开始，小明，小花直接使用非对称加密 进行通信，不就行了嘛

说的是对的，不过我们这里只是使用非对称加密 对 密钥S 进行加密，这个数据量很小的，而且密钥S安全的传输给对方之后

后面的通信就直接使用对称加密了，这样效率就高了，而非对称加密只是在开始协商怎么安全传输密钥S的阶段使用了，此阶段完成后，就不再需要使用了。

通过上面可知：非对称加密有这样的特性

我只要拿到谁的公钥，我和谁通信，就是安全的

比如，你有一对私钥和公钥，我只要拿到你的公钥，然后用你的公钥进行加密传输内容，只有你自己能解开，因为私钥只有你自己有

如下：

![Image 15: 在这里插入图片描述](https://i-blog.csdnimg.cn/blog\_migrate/cf70e28d5b0615d53796d897d89aa4ee.png)

反过来，小明用自己的私钥加密，其它人使用小明的公钥解密，这个过程的作用是什么呢？

答案是：验证身份的。

只要小明用自己的私钥加密，其它人用小明的公钥如果能解开，那么证明这封信一定以及肯定是小明写的

比如你需要发一个通知，但是又要确保这个通知一定是你发的，为了怕别人在中间涂改（比如古代假传圣旨，就是没有做好身份验证）

你可以用你的私钥对通知进行加密，其它人想看的话，通过下载你的公钥，进行解密，能解密出来，说明通知一定是你发的。

因为其它人如果在中间涂改，但是又没有你的私钥重新加密，所以是行不通的。

总结：通过以上的描述，我们解决了好几个问题，经过了以下几个过程。

- \* 小明和小花为了安全的通信，采用加密方式，对内容进行加密传输
- \* 对比来对比去，只能选对称加密这种加密方式，对内容进行加密传输
- \* 但是对称加密的密钥S，传输过程不安全，容易被老王窃取，怎么办呢
- \* 小明想到了非对称加密方式，于是就生成了一对私钥公钥，并且把公钥给了小花
- \* 小花就用公钥对密钥S进行加密，传给小明
- \* 因为是用小明的公钥加密的，又因为私钥只有小明自己有，所以，只有小明能解密。这个过程哪怕老王截获了密文，也解密不了
- \* 这样，小明用自己的私钥解密出来了 密钥S
- \* 此时 小明和小花就用对称加密， 密钥S， 进行愉快的通信了，比如商量彩礼给多少，酒席在哪办，蜜月在哪度
- \* 这样，这个通信过程就是安全的了。

上面的过程很完美，但是道高一尺，魔高一丈啊，老王脑子灵光特别好使啊，又想出来一招

既然你俩用非对称加密，我截取到密文也解密不了，那就换个法子。

如果小花在获取小明的公钥的过程，出了问题，比如小花获取的不是小明的公钥，而且老王的公钥呢（此时小花还以为手里的公钥是小明的呢）

会发生什么？先看一下图（也就是所谓的中间人攻击）

![Image 16: 在这里插入图片描述](https://i-blog.csdnimg.cn/blog\_migrate/6a4aa50efdfc5ed29bb9ea522410c23c.png)

根据上图，老王，也叫做中间人，上图就是中间人攻击，流程如下：

- \* 小花在获取小明公钥的过程中，被老王给掉包成了自己的公钥，发给了小花
- \* 小花误以为手里的公钥是小明的(其实是老王的公钥了)，所以就用老王的公钥对密钥S进行加密，得到密文S0
- \* 密文S0 发给小明的过程中，被老王拦截，老王就用自己的私钥解密，得到了密钥S
- \* 老王得到密钥S后，自己备份一份，再把此 密钥S，用小明的公钥加密，得到密文S1，发给小明
- \* 小明得到 密文S1后，用自己的私钥解密，得到 密钥S
- \* 以后，小明和小花，就用对称加密方式， 密钥S 进行通信了
- \* 他俩还以为很安全，其实通信的内容早就被老王先看了一遍了。还是不安全

啊啊啊，要疯了，为了通信安全，我们就加密，但是加密的密钥传输又不安全了

为了密钥传输安全，我们生产了私钥公钥对，把公钥给小花，小花用公钥对密钥加密再传输

这样就只有小明能解密了，没曾想，公钥的传输又不安全了。

谈个恋爱好难啊，老王啊，干的都叫啥事啊。。。

出了问题，总得解决啊，现在是传输公钥的过程，又不安全了

这和上面的问题 怎么把信件内容安全的传输给对方？以及怎么把密钥安全的传输给对方？ 是类似的

现在这个问题是： 怎么把公钥安全的传输给对方？

感觉进入到了死循环了，不管是把 信件内容安全传输，还是把密钥安全传输，还是把 公钥安全安全传输

本质都是类似的，只不过传输的东西不一样，采用的方法不一样

### 问题五： 小明如何安全的把自己的公钥传输给小花

经过上面我们解决的问题可以知道

\* 如何安全的把通信内容传输给对方？

解决方法：我们用对称加密的方式进行通信

- \* 如何安全的把密钥S安全的传输给对方？
- \* 解决方法：采用非对称加密方式，小明把自己的公钥给小花
- \* 小花用小明的公钥对密钥S加密传给小明，小明用自己的私钥解密
- \* 这个过程只有小明能解密，所以是安全的
- \* 现在新的问题是：公钥如何安全传输给对方？

难道再用对称或者非对称加密？都不对。这样已经行不通了。

想象一下，生活中，我们有个矛盾，有个问题，我们最相信的是谁，肯定是政府啊

现在从小明那下载公钥已经不靠谱了，已经不安全了

到底我应该相信谁呢？到底从谁那获取的公钥是小明真正的公钥呢？

所以，我们也搞一个机构，我们大家都相信这个机构，反正我就是无条件百分百相信这个机构，这是规定。

我们把这个机构起一个名字，叫做 CA 机构

好了，现在我们把问题抛给了 CA 机构，小花也好，小丽也好，小美也好，只要获取小明的公钥，都从CA那里获取

CA 机构哪来的小明的公钥呢？肯定是小明给的啊，对于小明来说，反正我已经把我的公钥给你CA了，你CA机构就得保证安全的传输给别人

这CA也是够倒霉的，你们搞不定的活，全抛给了我，又不是我和小花谈恋爱。。。

抱怨归抱怨，CA 是怎么解决的呢？

答案是`数字证书`，怎么又出来一个名字，数字证书是个什么鬼，是不是已经绕晕了，不要急，这个时候晕了，再回过过头再看看前面的写的

多看看几遍，别忘了，笔者也是看了N多遍，自己问自己问题，自己来尝试解决，才搞明白这个过程的。

先来说一个结论：数字证书就是解决公钥传输问题的

重要的事件重复三遍：数字证书就是解决公钥传输问题的，数字证书就是解决公钥传输问题的，数字证书就是解决公钥传输问题的

在说数字证书之前，我们先解决这样一个问题

### 问题六：信件的传输过程中，如何保证内容不被篡改，即信息的完整性？

结合前面学到的加密知识，我们可以用单向加密算法，我们以 md5 加密算法举例

- \* 小明给小花写完信后，用 md5 对信件的内容作一次加密运算，得到一个唯一的字符串，我们把这个字符串起个名，叫做摘要
- \* 小明在信件的底部，写上单向加密算法 md5，以及md5对信件内容运算出来的摘要，一块发给小花
- \* 小花收到信后，看到信件底部是md5算法，于是就用md5对信件内容进行加密算法，得到新的摘要
- \* 小花将新的摘要和信件底部附加的摘要进行对比，如果相等，说明信件没有被人改过
- \* 如果不相等，说明信件内容被别人改过了。

如下图表示此过程。

![Image 17: 在这里插入图片描述](https://i-blog.csdnimg.cn/blog\_migrate/51e7e2cf91e32a99da2cdef3b77053c8.png)

但就是上面这个过程，也是有问题的，如果老王又出现了呢

首先老王拿到信了，把信给改了

老王用md5算法，重新把信件内容给md5一下，得到新的加密串

老王把新的加密串，放在信件底部，发给了小花

此时小花收到信后，是没办法判断出来，信件是不是被篡改过的。

如下图表示：

![Image 18: 在这里插入图片描述](https://i-blog.csdnimg.cn/blog\_migrate/9d0ef8ac897463cc6247176fd4961f4a.png)

所以，单纯的使用单向加密算法，生成摘要，是不能保证内容的完整性的

那么如何才能保证信件的完整性，不被人篡改呢？

答案是，签名

又出来一个名词，签名，本文的名词太多了。

通过前面学习，我们知道，非对称加密，有2个作用，其中一个就是身份认证

还是上面的例子我，我们改一下：

小明用md5对信件内容进行运算，得到一个字符串，我们起名叫摘要

小明用自己的私钥对摘要进行加密运算，得到另一个字符串，我们起名叫签名

将 md5，摘要， 签名一块发给小花

小花用小明的公钥对签名进行解密，得到信件摘要，假如为 d1

小花用md5对信件内容进行运算，得到信件摘要，假如为 d2

对比 d1 和 d2 是否相等，相等说明信件内容没有被篡改过

d1 和 d2 不相等，说明信件内容被篡改过。

此时，这个过程就是安全的了

如果老王再次截取了信件，老王可以修改信件内容，再次用md5算出一个新的摘要出来

但是签名，老王是修改不了的。因为签名是用的小明的私钥加密的，就算老王能解密出来

老王是没有办法生成新的签名的，因为小明的私钥只有小明自己有。

而且小花收到信后，是用小明的公钥进行对签名解密的，老王假如用自己的私钥对摘要进行加密生成新的签名

小花用小明的公钥是解密不了的。

此时再来进行一些概念的定义

- \* **\*\*摘要\*\***：md5（或者其它单向加密算法），对内容进行加密出来的字符串，就叫做摘要
- \* **\*\*签名\*\***：小明用私钥对摘要进行加密，加密出来字符串，就叫做签名
- \* **\*\*验签\*\***：小花用小明的公钥，对签名进行解密操作，解密出来的摘要和原来的对比，就叫做验签

### ### 问题七：数字证书是怎么由来的？

数字证书是由CA机构颁发的，首先小明如果想要有一个数字证书，就需要向CA机构申请

CA机构就会给小明颁发一张数字证书，里面包含了

- \* 公钥：小明的公钥
- \* 颁发者：CA（证书认证机构）
- \* 有效期：证书的使用期限
- \* 摘要算法：指定的摘要算法，用来计算证书的摘要
- \* 指纹：也就是证书的摘要，保证证书的完整性
- \* 签名算法：用于生成签名，确保证书是由CA签发
- \* 序列号：证书的唯一标识

知道了证书里面包含的内容，我们了解一下证书是如何产生的？

- \* 将小明的公钥，颁发者，有效期，摘要算法，哈希算法写入证书
- \* CA 根据证书中的指定的哈希算法，计算出整个证书的摘要，即 digest
- \* CA根据签名算法以及上一步计算出来的摘要，CA用自己的私钥对摘要进行加密，生成CA的签名，即 signature
- \* 最后把摘要，签名以及证书的基本信息，一起发布，就得到了小明的证书

### ### 问题八：数字证书的作用

从上面我们知道，数字证书就是解决公钥传输问题的，同时我们也知道，数字证书就是一个文件

既然数字证书是用来解决公钥的安全传输的，那么到底如何解决传输问题的呢

现在小明有了自己的证书了，我们就不会公开传输公钥了，只需要传输证书就行了

那么，小明和小花现在需要安全的通信，那么流程是怎么样的呢？如下

- \* 小明把自己的数字证书发送给小花
- \* 担心证书被老王掉包，小花需要对证书进行验证，验证什么呢？
- \* 其实就是验证此数字到底是不是CA机构颁发的，不是CA机构颁发的证书，我们就认为传输是不安全的。
- \* 验证数字证书是不是CA颁发的，需要有CA的公钥。。。 (为啥需要CA的公钥啊，因为证书上的签名，是CA的私钥加密的啊，只有CA的公钥才能解密啊)
- \* 啊啊啊，受不了啦，搞了半天怎么又需要公钥，我们讲了半天的数字证书，就是为了传输公钥的
- \* 所以，换成下面的描述会好点
- \* 验证数字证书是不是CA颁发的，需要CA的数字证书 (因为里面有CA的公钥)
- \* 那我们去哪里找CA的数字证书呢？从上面的描述，我们知道了，需要一个数字证书，就向CA申请，CA给我们颁发。
- \* 那么CA机构自己的数字证书哪来的呢？答案是也是自己给自己颁发的，那么我们从哪里获取呢？
- \* 如果从网上，或者从其它服务器下载，又有可能被掉包，又不安全了。
- \* 这真的是个伤心的故事，但是今天兔哥非要把这个故事讲完。
- \* 从网上下载或者从其它服务器下载数字证书，都不安全的，那么怎么样才是安全的呢？
- \* 答案就是：你的电脑安装操作系统的时候，操作系统里面，就已经内置了非常多的CA机构的数字证书了
- \* 也就是说，只要你安装了操作系统，不管是windows, linux, 或者 mac，或者你刚买的电脑，里面都已经有了CA机构的数字证书了
- \* 这个是可以相信的，是真的CA机构的数字证书，不会有假。(除非你安装的是盗版的操作系统，所以我们尽量用正版操作系统)

上面的过程真的是复杂啊，兔哥也是花了很久才搞明白的，知道这块面试会坑很多人，其实https 过程不知道，也没啥关系

也不影响你写代码，但是那些面试官就死爱问这块，好像他们能搞懂这个过程很了不起似的，你问点设计模式它不香嘛。

不过话说回来，兔哥在写自己的 HelloWorld技术社区 的时候，配置 https，数字证书，不懂这些，还真的不好搞啊

写文章不容易，尤其是写这篇文章，为了写的更容易懂点，花了不少精力，能看到这块的，帮忙给个关注吧

尤其是帮忙宣传一下兔哥的 HelloWorld技术社区，同一个世界，同一行代码，我们的域名是：  
www.helloworld.net

我们的电脑，天生就有CA的数字证书，而且是真的。天生的。上天定的，上天最大  
那么我们就可以对数字证书进行辨别真伪了。

### ### 问题九：对数字证书的验证

从上面可以知道：

小花收到了小明的数字证书，首先要对数字证书进行验证，就是验证此数字证书是不是CA颁发的

因为我们操作系统里面内置了所有CA机构的数字证书，所以，我们就可以对数字证书进行验证

在说流程之前，先来简单的复习一下前面的，摘要和签名怎么来的

```
```bash
```

摘要 = md5(证书内容)：单向加密算法，比如md5，对证书整个内容进行加密，得到摘要，也叫做证书的指纹

签名 = privateKey(摘要)：私钥对上一步摘要加密，产生签名

```
```
```

数字证书的验证流程如下：

- \* 小花用内置的CA的数字证书，得到CA的公钥
- \* 小明发过来的数字证书，我们假如叫做 C，小花用CA的公钥对C证书里面的签名进行解密，得到摘要D
- \* 小花根据C证书里面的摘要算法，假如是md5，小花用md5对证书整个内容进行计算，得到摘要D1
- \* 小花对比摘要D和摘要D1 是否相等
- \* 如果 D == D1，那么说明此证书就是CA颁发的
- \* 如果 D != D1，那么说明此证书不是CA颁发的，是有风险的，不安全的
- \* 假如证书验证通过，就说明此证书的确是CA颁发的，此时小花就可以从数字证书中拿到小明的公钥了



因为小明在申请数字证书时，数字证书中所有者是小明，CA是会验证小明的身份的，所以数字证书中小明的公钥是真实的

由至此，我们总算完成了一件事：小明正确的把自己的公钥安全的传输给了小花

这件事的成立，接下来我们的工作就好做多了。接下来，我们看一下具体的传输过程

### 问题十：完整的传输过程

下面我们看一下小明再次给小花通信，就和前面的不一样了，我们来看下：

- \* 小明把写完的信，在信的底部，附加上摘要算法，假如是MD5，以及通过MD5算出来的摘要
- \* 小明用自己的私钥，对上一步的摘要进行加密，得到签名
- \* 小明把摘要算法，摘要，签名都附加到信件底部以后，再把自己的数字证书，一起发送给小花
- \* 小花收到信后，首先用自己的CA数字证书，拿到CA公钥，再用CA公钥对数字证书进行验证（也就是上面我们讲的流程）
- \* 数字证书验证通过后，说明证书就是CA颁发的，没有被篡改
- \* 小花就从证书中拿到了小明的公钥
- \* 有了小明的公钥，接下来的过程，就是对信件内容进行验证了

对信件内容的验证流程如下（前面其实我们讲过）

- \* 小花用小明的公钥，对信件的签名进行解密，得到信件的摘要D1
- \* 小花用摘要算法，对信件进行运算，得到信件的摘要D2
- \* 小花对比 D1 是否等于D2
- \* 如果不相等，说明信件被人篡改过，不安全
- \* 如果相等，说明，信件内容没有被篡改过
- \* 相等的情况，小花就拿到了信件的内容

### 三、总结

以上所有的内容，是数字证书，加密解密，签名，验签的过程，还没有正式讲 https 的过程呢。

有了以上的知识，我们讲起来https 就容易的多了。下面我们看一张图

![Image 19: 在这里插入图片描述](https://i-blog.csdnimg.cn/blog\_migrate/5e408fa39f76632602aa7f751dac33b2.png)

我们以访问 [www.helloworld.net](http://www.helloworld.net) 网站为例，讲解https的过程

此过程分为3个阶段，我们在下面描述此3个阶段

访问 [www.helloworld.net](http://www.helloworld.net) 的过程 阶段如下

- \* 网站申请证书阶段
- \* 网站向CA机构申请数字证书（需要提交一些材料，比如域名）
- \* CA向证书中写入摘要算法，域名，网站的公钥等重要信息
- \* CA根据证书中写入的摘要算法，计算出证书的摘要
- \* CA用自己的私钥对摘要进行加密，计算出签名
- \* CA生成一张数字证书，颁发给了 [www.helloworld.net](http://www.helloworld.net)
- \* 网站的管理员，把证书放在自己的服务器上
- \* 浏览器验证证书阶段
- \* 浏览器在地址栏中输入<https://www.helloworld.net>，并回车
- \* 服务器将数字证书发送给浏览器
- \* 浏览器用操作系统内置的CA的数字证书，拿到CA的公钥
- \* 浏览器用CA公钥对 [www.helloworld.net](http://www.helloworld.net) 的数字证书进行验签
- \* 具体就是，浏览器用CA公钥，对helloworld的数字证书中的签名进行解密，得到摘要D1
- \* 浏览器根据helloworld数字证书中的摘要算法，计算出证书的摘要D2
- \* 对比 D1和D2 是否相等。
- \* 如果不相等，说明证书被掉包了
- \* 如果相等，说明证书验证通过了。
- \* 协商对称加密密钥阶段
- \* 浏览器验证数字证书通过后
- \* 浏览器拿到数字证书中的公钥，也就是 [www.helloworld.net](http://www.helloworld.net) 网站的公钥
- \* 浏览器有了网站的公钥后，就用公钥进行对密钥S进行加密，加密以后的密文发送给服务器
- \* 服务器收到密文后，用自己的私钥进行解密，得到密钥S
- \* 此后浏览器，服务器双方就用密钥S进行对称加密的通信了。

终止所述，终于讲完了，花了整整一天的时间

过程那么多，其实抓住几个关键的问题是很简单的，本质上还是两个人，如何安全高效的进行通信

#### ### 四、面试

我们再次简单的总结一下，采用一问一答的方式，我觉得比较好

### 问题一：小明和小花安全的通信，怎么做？

答：通过加密

### 问题二：通过哪种加密方式通信，更高效？

答：对称加密

因为，单向加密，没办法解密，不行

非对称加密，太慢，也不行

只有对称加密，速度快

### 问题三：采用对称加密，密钥S 怎么安全传输？

答：小花使用小明的公钥，对密钥S进行加密，传给小明

小明用自己的私钥解密

### 问题四：小明如何安全的把自己的公钥传输给小花？

答：使用数字证书

具体就是 小明向CA申请一个自己的数字证书，把自己的公钥放在证书中

小明将数字证书发送给小花

### 问题五：小花如何验证数字证书的真实性？

答：小花用操作系统内置的CA的数字证书，拿到CA的公钥，用CA的公钥，对数字证书进行验签

验签通过，说明数字证书是真的。

关注博主即可阅读全文 ![Image 20](https://csdnimg.cn/release/blogv2/dist/pc/img/arrowDownAttend.png)

![Image 21](https://csdnimg.cn/release/blogv2/dist/pc/img/vip-limited-close-newWhite.png)

确定要放弃本次机会？

福利倒计时

\_:\_ \_:\_

![Image 22](https://csdnimg.cn/release/blogv2/dist/pc/img/vip-limited-close-roup.png) 立减 ¥

普通VIP年卡可用

[立即使用](https://mall.csdn.net/vip)

![Image 23](https://profile-avatar.csdnimg.cn/b44265b0cd214a31a3c66f3d2261ad39\_blueeyedboy521.jpg!1) Blueeyedboy521  
(https://blog.csdn.net/Blueeyedboy521)

[关注](javascript:;) 关注

\* ![Image 24](https://csdnimg.cn/release/blogv2/dist/pc/img/tobarThumbUpactive.png) ![Image 25]  
(https://csdnimg.cn/release/blogv2/dist/pc/img/toolbar/like-active.png) ![Image 26]  
(https://csdnimg.cn/release/blogv2/dist/pc/img/toolbar/like.png) 3

点赞

\* ![Image 27](https://csdnimg.cn/release/blogv2/dist/pc/img/toolbar/unlike-active.png) ![Image 28](https://csdnimg.cn/release/blogv2/dist/pc/img/toolbar/unlike.png)

踩

\* [![Image 29](https://csdnimg.cn/release/blogv2/dist/pc/img/toolbar/collect-active.png) ![Image 30](https://csdnimg.cn/release/blogv2/dist/pc/img/toolbar/collect.png) ![Image 31](https://csdnimg.cn/release/blogv2/dist/pc/img/newCollectActive.png) 2](javascript:;)

收藏

觉得还不错? 一键收藏 ![Image 32](https://csdnimg.cn/release/blogv2/dist/pc/img/collectionCloseWhite.png)

\* ![Image 33](https://csdnimg.cn/release/blogv2/dist/pc/img/guideRedReward01.png) 知道了  
[![Image 34](https://csdnimg.cn/release/blogv2/dist/pc/img/toolbar/comment.png) 0](https://blog.csdn.net/Blueeyedboy521/article/details/127938799?spm=1001.2101.3001.4242.1#commentBox)

评论

\* [![Image 35](https://csdnimg.cn/release/blogv2/dist/pc/img/toolbar/share.png) 分享](javascript:;)

复制链接

分享到 QQ

分享到新浪微博

![Image 36](https://csdnimg.cn/release/blogv2/dist/pc/img/share/icon-wechat.png) 扫一扫

\* ![Image 37](https://csdnimg.cn/release/blogv2/dist/pc/img/toolbar/more.png)

![Image 38](https://csdnimg.cn/release/blogv2/dist/pc/img/toolbar/report.png) 举报

![Image 39](https://csdnimg.cn/release/blogv2/dist/pc/img/toolbar/report.png) 举报

专栏目录

[X. 509数字\_证书\_的签名和\_指纹\_](https://jasonj333.blog.csdn.net/article/details/138181683)

[jasonj333](https://blog.csdn.net/wjz110201)

04-25 ![Image 40](https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png) 428

[X. 509 是一种非常普遍的数字\_证书\_标准, 由国际电信联盟 (ITU) 制定。它定义了\_证书\_的格式和一种验证\_证书\_有效性的方法。X. 509 \_证书\_的结构遵循特定的语法和编码规则, 通常使用 ASN.1 (Abstract Syntax Notation One) 进行描述和编码。](https://jasonj333.blog.csdn.net/article/details/138181683)

[\_https\_证书\_原理\_是什么](https://blog.csdn.net/wecloud1314/article/details/120738552)

[蔚可云的博客](https://blog.csdn.net/wecloud1314)

10-13 ![Image 41](https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png) 2136

[有多少人对于\_https\_是不是特别的茫然呢, 实际上这个就是安全\_证书\_, 安全\_证书\_对于我们网络来说是必不可少的一个证件, 所以说大家一定要重视起来, 多多学习这方面的知识, 那么今天小编给大家讲一下\_https\_证书\_原理\_是什么? 一、\_https\_证书\_原理\_ 现在给大家说一下\_https\_证书\_原理\_, \_https\_在传输出去之前就需要和客户端有。呃。握手, 在握手的过程中将确立双方加密传输数据的密钥信息。就是保持这个信息的安全性。浏览器将自己支持的一套加密规则发送给网站之后, 网站就会选出一种加密算法。与hash算法, 将自己的身份信息和\_证书\_] (https://blog.csdn.net/wecloud1314/article/details/120738552)

参与评论 您还未登录, 请先 登录 后发表或查看评论

[这么多年被第三方接入坑的那些事。。。关于md5签名和sha1\_证书\_的坑]  
([https://blog.csdn.net/weixin\\_33923762/article/details/91944154](https://blog.csdn.net/weixin_33923762/article/details/91944154))

[weixin\\_33923762的博客]([https://blog.csdn.net/weixin\\_33923762](https://blog.csdn.net/weixin_33923762))

04-24 ![Image 42](<https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png>) 1172

[2019独角兽企业重金招聘Python工程师标准\>\>\> ...]  
([https://blog.csdn.net/weixin\\_33923762/article/details/91944154](https://blog.csdn.net/weixin_33923762/article/details/91944154))

[\_SSL\_原理\_](<https://blog.csdn.net/gaodemin/article/details/1584816>)

[踏雪无痕的专栏](<https://blog.csdn.net/gaodemin>)

04-25 ![Image 43](<https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png>) 1217

[\_SSL\_ 是一个安全协议，它提供使用 TCP/IP 的通信应用程序间的隐私与完整性。因特网的 超文本传输协议（HTTP）使用 \_SSL\_ 来实现安全的通信。在客户端与服务器间传输的数据是通过使用对称算法（如 DES 或 RC4）进行加密的。公用密钥算法（通常为 RSA）是用来获得加密密钥交换和数字签名的，此算法使用服务器的 \_SSL\_数字\_证书\_中的公用密钥。有了服务器的\_SSL\_数字\_证书\_，客户端也可以验证服]  
(<https://blog.csdn.net/gaodemin/article/details/1584816>)

[【\_SSL\_】\_HTTPS\_和 \_SSL\_证书\_\_原理\_](<https://blog.csdn.net/bandaoyu/article/details/106554600>)

[bandaoyu的note](<https://blog.csdn.net/bandaoyu>)

06-04 ![Image 44](<https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png>) 1306

[问题 阐述\_HTTPS\_和\_SSL\_证书\_的原理\_。解答 \_HTTPS\_（Hyper Text Transfer Protocol Secure）对比HTTP而言，数据传输更安全，通常用来传输私密性的数据，如在线交易/订单处理/银行数据等，不过随着安全意识的普及，非传统安全领域的网站也在转向\_HTTPS\_。如苹果要求App Store内的应用都要支持\_HTTPS\_。\_HTTPS\_通过安全协议如 \_SSL\_/TLS 来保证数据传输，\_SSL\_和TLS都是基于非对称加密算法，而RSA算法是非对称加密算法中的佼佼者。RSA的公](<https://blog.csdn.net/bandaoyu/article/details/106554600>)

[浅析\_HTTPS\_与\_SSL\_原理\_]([https://blog.csdn.net/tengxy\\_cloud/article/details/52808163](https://blog.csdn.net/tengxy_cloud/article/details/52808163))

[tengxy\\_cloud的专栏]([https://blog.csdn.net/tengxy\\_cloud](https://blog.csdn.net/tengxy_cloud))

10-13 ![Image 45](<https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png>) 2857

[\_HTTPS\_是指结合HTTP和\_SSL\_来实现网络浏览器和服务器之间的安全通信。\_HTTPS\_被融合到当今网络操作系统和网络浏览器中，他依赖于网络服务器是否支持\_HTTPS\_协议。本文重点是阐述\_HTTPS\_和\_SSL\_证书\_的原理\_，不涉及具体实现。在开始叙述\_原理\_之前，先明确几个网络安全方面的基础概念。一.基础概念 1. 对称加密算法一个对称加密算法由五个部分组成：明文：原始消息或数]  
([https://blog.csdn.net/tengxy\\_cloud/article/details/52808163](https://blog.csdn.net/tengxy_cloud/article/details/52808163))

[\_HTTPS\_协议工作\_原理\_( \_SSL\_数字\_证书\_)](<https://xie1997.blog.csdn.net/article/details/84395154>)

[谢公子的博客]([https://blog.csdn.net/qz\\_36119192](https://blog.csdn.net/qz_36119192))

11-23 ![Image 46](<https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png>) 5452

[目录 \_HTTPS\_ \_SSL\_协议的工作过程 \_SSL\_数字\_证书\_的查看 \_HTTPS\_ 我们都知道HTTP协议是明文传输的，并且不能验证对方的身份，而且不能保证数据的完整性。而我们在网络上进行购物电子交易时，电子网银转账时，这种方式就显得很不安全了。如果黑客截取了我们的服务器端的通信数据，那么黑客就能获取我们的一些敏感信息了。所以，\_HTTPS\_应运而生！\_HTTPS\_是在HTTP协议的基础上加入了SS...]  
(<https://xie1997.blog.csdn.net/article/details/84395154>)

[（转）\_https\_/ \_SSL\_ \_原理\_](<https://blog.csdn.net/zujiety/article/details/84003393>)

[Android Fans](<https://blog.csdn.net/zujiety>)

08-17 ![Image 47](<https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png>) 116

[此前我一直对\_https\_/\_ssl\_如何保护数据不被监听有点疑问，因为服务器的\_证书\_是公开的，只能实行上行方向的数据加密，下行数据的加密我一直认为是浏览器会自动生成一个客户端的密钥对并将公钥发给服务器。今天仔细研究了\_https\_/\_ssl\_，发现其实并不像我想的那样，这里面既有非对称加密，又因为性能原因使用了对称加密。其\_原理\_大概是这样的：客户端浏览器连接到\_https\_/\_ssl\_服务器，并发送\_ssl\_版本号等...]

(<https://blog.csdn.net/zujiety/article/details/84003393>)

[通俗\_易懂讲解, \_彻底\_弄懂\_ \_https\_ \_原理\_本质\_ (五) \_https\_ 的过程 最新发布]  
([https://blog.csdn.net/csdn\\_aiyang/article/details/129801225](https://blog.csdn.net/csdn_aiyang/article/details/129801225))

[csdn\\_aiyang的博客]([https://blog.csdn.net/csdn\\_aiyang](https://blog.csdn.net/csdn_aiyang))

03-27 ![Image 48](<https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png>) 557

[摘要: 单向加密算法, 比如md5对\_证书\_整个内容加密的结果就叫摘要, 也叫做\_证书\_的\_指纹\_; 签名: 小明用私钥🔑对摘要进行加密出来签字串, 就叫做签名。 验签: 用公钥对签名进行解密出来的摘要和原来的对比, 就叫做验签。]([https://blog.csdn.net/csdn\\_aiyang/article/details/129801225](https://blog.csdn.net/csdn_aiyang/article/details/129801225))

[通俗\_易懂的带你了解\_HTTPS\_的\_原理\_] (<https://blog.csdn.net/codeyanbao/article/details/111569150>)

[吴延宝](<https://blog.csdn.net/codeyanbao>)

12-22 ![Image 49](<https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png>) 6713

[序言 在绝大部分人的印象中, 技术知识都是很枯燥无味的, 读起技术文章来, 感觉脑细胞都要死一片。今天我们就尝试换一种新的方式, 目的是可以让大家在轻松愉快的氛围中还能把技术学了。记得关注我哦, 让我们一起将技术变的有趣起来。《新编二战回忆录》之\_HTTPS\_不讲武德 二战前 二战前夕, 德国励精图治, 军备急速扩张, 收到情报的丘桑立马联系了远在东边的斯大林同志: 由于HTTP协议是明文传输, 丘桑和大林同志的微信聊天被元首轻而易举的截获并篡改, 以此迷惑住大林同志, 元首更加大胆的筹划着下一步动作: 闪击波兰 友情提示: 看出](<https://blog.csdn.net/codeyanbao/article/details/111569150>)

[【\_HTTPS\_】一文带你全面了解\_SSL\_] (<https://xyf0628.blog.csdn.net/article/details/99691301>)

[Dream it Possible](<https://blog.csdn.net/u013034223>)

08-17 ![Image 50](<https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png>) 1844

[【引言】\_HTTPS\_是众所周知的, 但要是问到“S”的含义是什么? 也许能回答上来的人并不多。\_HTTPS\_其实是HTTP + \_SSL\_, S的含义也就是Secure Socket Layer (简称\_SSL\_)。前段时间看了下\_SSL\_的相关文档, 所以本篇博客以\_SSL\_为主题, 对相关知识做个总结。...] (<https://xyf0628.blog.csdn.net/article/details/99691301>)

[\_大白话\_彻底\_搞懂HBase RowKey详细设计 热门推荐]  
(<https://ropladata.blog.csdn.net/article/details/105994960>)

[静待花开]([https://blog.csdn.net/qz\\_26803795](https://blog.csdn.net/qz_26803795))

05-08 ![Image 51](<https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png>) 2万+

[本文从RowKey的\_原理\_, 可能出现的问题, 如何优化及各个优化措施对应的缺点和适用的场景, 设计原则等角度对RowKey进行了详细全面的解析, 相信一定能对你有帮助。]  
(<https://ropladata.blog.csdn.net/article/details/105994960>)

[\_https\_之\_SSL\_\_证书\_] ([https://blog.csdn.net/ocean\\_fan/article/details/79201996](https://blog.csdn.net/ocean_fan/article/details/79201996))

[小范的博客]([https://blog.csdn.net/ocean\\_fan](https://blog.csdn.net/ocean_fan))

01-30 ![Image 52](<https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png>) 603

[内容大纲 1、\_SSL\_\_证书\_是由受信任的数据\_证书\_颁发机构颁发的\_证书\_, 主要是帮助用户识别服务器身份以及数据传输过程加密防止数据被篡改, 防劫持、防篡改、防监听。 2、一份\_SSL\_\_证书\_包含一个公钥和一个私钥。公钥用于数据加密私钥用于数据解密。 3、\_SSL\_连接产生会话服务器会将它的\_证书\_传到客户端, 客户端再根据传回的\_证书\_生成会话密钥, 这个会话密钥会对传输数据进行加密。 4、理解三个协议, 握]([https://blog.csdn.net/ocean\\_fan/article/details/79201996](https://blog.csdn.net/ocean_fan/article/details/79201996))

[加密、签名和\_SSL\_握手机制细节]([https://blog.csdn.net/weixin\\_30559481/article/details/98601266](https://blog.csdn.net/weixin_30559481/article/details/98601266))

[weixin\\_30559481的博客]([https://blog.csdn.net/weixin\\_30559481](https://blog.csdn.net/weixin_30559481))

11-22 ![Image 53](<https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png>) 186

[open\_ssl\_系列文章: <http://www.cnblogs.com/f-ck-need-u/p/7048359.html> 1.1 背景知识 对称加密: 加密解密使用同一密钥, 加解密速度快。随着人数增多, 密钥数量急增 $n(n-1)/2$ 。 非对称加密: 使用公私钥配对加解密, 速度慢。公钥是从私钥中提取出来的, 一般拿对方公钥加密来保证数据安全性, 拿自己的私钥加密来证明数据来源的身份...] ([https://blog.csdn.net/weixin\\_30559481/article/details/98601266](https://blog.csdn.net/weixin_30559481/article/details/98601266))

[数字\_证书\_\_原理\_]([https://blog.csdn.net/weixin\\_30737363/article/details/99334208](https://blog.csdn.net/weixin_30737363/article/details/99334208))

[weixin\\_30737363的博客]([https://blog.csdn.net/weixin\\_30737363](https://blog.csdn.net/weixin_30737363))

06-24 ![Image 54](<https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png>) 1895

[文中首先解释了加密解密的一些基础知识和概念，然后通过一个加密通信过程的例子说明了加密算法的作用，以及数字\_证书\_的出现所起的作用。接着对数字\_证书\_做一个详细的解释，并讨论一下windows中数字\_证书\_的管理，最后演示使用makecert生成数字\_证书\_。如果发现文中有错误的地方，或者有什么地方说得不够清楚，欢迎指出！ 1、基础知识 这部分内容主要解释一些概念和术语，最好是先理解这部分内容...]([https://blog.csdn.net/weixin\\_30737363/article/details/99334208](https://blog.csdn.net/weixin_30737363/article/details/99334208))

[\_https\_\_原理\_](<https://blog.csdn.net/liulpan2min3/article/details/83868833>)

[程序媛的一生](<https://blog.csdn.net/liulpan2min3>)

11-09 ![Image 55](<https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png>) 4932

[我们都知道\_https\_能加密信息，以免敏感信息被第三方获取。所以很多银行网站活着电子邮箱等安全级别比较高的服务都会采用\_https\_协议\_https\_简介\_https\_其实有两部分组成：HTTP + \_SSL\_ / TLS，也就是在HTTP上又加了一层处理加密信息的模块。服务端和客户端的信息传输都会通过TLS加密，所以传输的数据都是加密后的数据。具体是符合进行加密的，解密，验证的，且看下图 ...](<https://blog.csdn.net/liulpan2min3/article/details/83868833>)

[\_https\_ \_SSL\_流程\_原理\_](<https://blog.csdn.net/u012957549/article/details/105331956>)

[u012957549的博客](<https://blog.csdn.net/u012957549>)

04-05 ![Image 56](<https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png>) 390

[前面两章介绍了，我们了解对称加密与非对称加密的特点和优缺点，以及数字\_证书\_的作用。\_https\_没有采用单一的技术去实现，而是根据他们的特点，充分的将这些技术整合进去，以达到性能与安全最大化。这套整合的技术我们称之为\_SSL\_(Secure Socket Layer 安全套接层)。所以\_https\_并非是一项新的协议，它只是在http上披了一层加密的外壳。\_https\_的建立 先看一下建立的流程图： 这里把h...](<https://blog.csdn.net/u012957549/article/details/105331956>)

[一文读懂\_Https\_\_原理\_](<https://blog.csdn.net/hyhlmy/article/details/104181847>)

[hyhlmy的博客](<https://blog.csdn.net/hyhlmy>)

02-05 ![Image 57](<https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png>) 1819

[一文读懂\_Https\_\_原理\_前言\_https\_\_原理\_浅析\_https\_是什么如何加密数据公钥分发问题服务器身份认证\_证书\_链\_SSL\_/TLS完整握手过程Fiddler代理\_https\_\_原理\_前言 对于\_https\_大部分人应该都不陌生，相信大家也都知道\_https\_是安全的，但是如果问你为什么\_https\_是安全的，可能还是会有一大部分人是答不上来的。因为本人在之前面试和实际项目中也经常碰到\_https\_相关的问题，所以最近花了些时间梳理...](<https://blog.csdn.net/hyhlmy/article/details/104181847>)

[\_Https\_之\_SSL\_\_原理\_]([https://blog.csdn.net/weixin\\_30381317/article/details/99121614](https://blog.csdn.net/weixin_30381317/article/details/99121614))

[weixin\\_30381317的博客]([https://blog.csdn.net/weixin\\_30381317](https://blog.csdn.net/weixin_30381317))

03-04 ![Image 58](<https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png>) 144

[一、\_HTTPS\_和\_SSL\_ HTTP (Hyper TEXT Transfer Protocol超文本传输协议)是目前互联网上应用最为广泛的一种网络协议，用于在Web浏览器和网站服务器之间传递信息，但是HTTP协议以明文的方式发送内容，不提供任何数据加密，攻击者能够很轻易通过抓包的方式截取传输内容并读懂其中的信息，所以HTTP不适合传输一些比较私密的信息，为了解决HTTP这一...]([https://blog.csdn.net/weixin\\_30381317/article/details/99121614](https://blog.csdn.net/weixin_30381317/article/details/99121614))

[\_HTTPS\_( \_SSL\_) 工作\_原理\_](<https://download.csdn.net/download/fisonlee/9951260>)

08-26

[\_HTTPS\_( \_SSL\_) 工作\_原理\_](<https://download.csdn.net/download/fisonlee/9951260>)

[![Image 59]([https://profile-avatar.csdnimg.cn/b44265b0cd214a31a3c66f3d2261ad39\\_blueeyeb0y521.jpg!1](https://profile-avatar.csdnimg.cn/b44265b0cd214a31a3c66f3d2261ad39_blueeyeb0y521.jpg!1))](<https://blog.csdn.net/Blueeyeb0y521>)

[Blueeyedboy521] (<https://blog.csdn.net/Blueeyedboy521> "Blueeyedboy521") CSDN认证博客专家 CSDN认证企业博客

码龄7年 [![Image 60] (<https://csdnimg.cn/identity/nocErtification.png>) 暂无认证] ([https://i.csdn.net/#/uc/profile?utm\\_source=14998968](https://i.csdn.net/#/uc/profile?utm_source=14998968) "暂无认证")

[262 原创] (<https://blog.csdn.net/Blueeyedboy521>)

[1995 周排名] (<https://blog.csdn.net/rank/list/weekly>)

[8547 总排名] (<https://blog.csdn.net/rank/list/total>)

60万+

访问

[![Image 61] (<https://csdnimg.cn/identity/blog5.png>)] (<https://blog.csdn.net/blogdevteam/article/details/103478461>)

等级

3563

积分

6万+

粉丝

714

获赞

130

评论

2300

收藏

[![Image 62: 五一创作勋章] ([https://csdnimg.cn/medal/51\\_create.png](https://csdnimg.cn/medal/51_create.png))

[![Image 63: 持续创作] (<https://csdnimg.cn/medal/chizhiyiheng@240.png>)

[![Image 64: 1024勋章] ([https://csdnimg.cn/medal/up2\\_1024@240.png](https://csdnimg.cn/medal/up2_1024@240.png))

[![Image 65: 勤写标兵] (<https://csdnimg.cn/f10c883cfd674a3e9217683e8cd55511.png>)

[![Image 66: 笔耕不辍] (<https://csdnimg.cn/c151d54288e14ceba2ee6595d3dec3c7.png>)

[![Image 67: 创作能手] (<https://csdnimg.cn/medal/qixiebiaobing4@240.png>)

[私信] (<https://im.csdn.net/chat/Blueeyedboy521>)

关注

[![Image 68] (<https://blog.csdn.net/Blueeyedboy521/article/details/127938799?spm=1001.2101.3001.4242.1>)

[![Image 69] (<https://csdnimg.cn/cdn/content-toolbar/csdn-sou.png?v=1587021042>)

[![Image 70] (<https://kuny.csdn.net/1.png?p=56&adId=1048397&adBlockFlag=0&a=1048397&c=0&k=%E3%80%90%E8%BD%AC%E8%BD%BD%E3%80%91%E9%80%9A%E4%BF%97%E5%A4%A7%E7%99%BD%E8%AF%9D%EF%BC%8C%E5%BD%BB%E5%BA%95%E5%BC%84%E6%87%82%20https%20%E5%8E%9F%E7%90%86%E6%9C%AC%E8%B4%A8%EF%BC%88ss1%E8%AF%81%E4%B9%A6%EF%BC%89&spm=1001.2101.3001.5000&articleId=127938799&d=1&t=3&u=049e90ec7d9345ff925b23ed52ad65ed>)

### ### 热门文章

\* [谷歌浏览器提示客户端和服务端不支持一般 SSL 协议版本或加密套件（亲测有效）！[Image 71]  
(<https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png>) 126751]  
(<https://blog.csdn.net/Blueeyedboy521/article/details/124963562>)

\* [ArduinoUNO实战-第十七章-火焰传感器！[Image 72]  
(<https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png>) 11199]  
(<https://blog.csdn.net/Blueeyedboy521/article/details/127377566>)

\* [docker安装Prometheus+Grafana监控系统（SpringBoot集成监控）！[Image 73]  
(<https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png>) 10798]  
(<https://blog.csdn.net/Blueeyedboy521/article/details/126409204>)

\* [docker容器的时间和宿主机时间不一致相差8小时！[Image 74]  
(<https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png>) 9576]  
(<https://blog.csdn.net/Blueeyedboy521/article/details/125223595>)

\* [SpringBoot集成RabbitMQ！[Image 75]  
(<https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png>) 8387]  
(<https://blog.csdn.net/Blueeyedboy521/article/details/124001397>)

### ### 分类专栏

\* [[Image 76] ([https://i-blog.csdnimg.cn/columns/default/20201014180756923.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756923.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) 数据库]  
([https://blog.csdn.net/blueeyedboy521/category\\_11772980.html](https://blog.csdn.net/blueeyedboy521/category_11772980.html)) 10篇

\* [[Image 77] ([https://i-blog.csdnimg.cn/columns/default/20201014180756922.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756922.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) java]  
([https://blog.csdn.net/blueeyedboy521/category\\_11767600.html](https://blog.csdn.net/blueeyedboy521/category_11767600.html)) 27篇

\* [[Image 78] ([https://i-blog.csdnimg.cn/columns/default/20201014180756918.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756918.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) 深度学习]  
([https://blog.csdn.net/blueeyedboy521/category\\_11871122.html](https://blog.csdn.net/blueeyedboy521/category_11871122.html)) 10篇

\* [[Image 79] ([https://i-blog.csdnimg.cn/columns/default/20201014180756925.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756925.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) 物联网]  
([https://blog.csdn.net/blueeyedboy521/category\\_12074657.html](https://blog.csdn.net/blueeyedboy521/category_12074657.html)) 4篇

\* [[Image 80] ([https://i-blog.csdnimg.cn/columns/default/20201014180756754.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756754.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) 云计算]  
([https://blog.csdn.net/blueeyedboy521/category\\_12010685.html](https://blog.csdn.net/blueeyedboy521/category_12010685.html)) 1篇

\* [[Image 81] ([https://i-blog.csdnimg.cn/columns/default/20201014180756754.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756754.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) 前端]  
([https://blog.csdn.net/blueeyedboy521/category\\_11767917.html](https://blog.csdn.net/blueeyedboy521/category_11767917.html)) 1篇

\* [[Image 82] ([https://i-blog.csdnimg.cn/columns/default/20201014180756922.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756922.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) 音视频]  
([https://blog.csdn.net/blueeyedboy521/category\\_11767552.html](https://blog.csdn.net/blueeyedboy521/category_11767552.html)) 13篇

\* [[Image 83] ([https://i-blog.csdnimg.cn/columns/default/20201014180756738.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756738.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) SpringCloud]  
([https://blog.csdn.net/blueeyedboy521/category\\_11689598.html](https://blog.csdn.net/blueeyedboy521/category_11689598.html)) 31篇

\* [[Image 84] ([https://i-blog.csdnimg.cn/columns/default/20201014180756925.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756925.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) 大数据]  
([https://blog.csdn.net/blueeyedboy521/category\\_11722577.html](https://blog.csdn.net/blueeyedboy521/category_11722577.html)) 28篇

\* [[Image 85] ([https://i-blog.csdnimg.cn/columns/default/20201014180756930.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756930.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) 开发组件]  
([https://blog.csdn.net/blueeyedboy521/category\\_11740291.html](https://blog.csdn.net/blueeyedboy521/category_11740291.html)) 54篇

\* [[Image 86] ([https://i-blog.csdnimg.cn/columns/default/20201014180756913.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756913.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) RabbitMQ]  
([https://blog.csdn.net/blueeyedboy521/category\\_11767722.html](https://blog.csdn.net/blueeyedboy521/category_11767722.html)) 9篇

\* [[Image 87] ([https://i-blog.csdnimg.cn/columns/default/20201014180756928.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756928.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) linux]  
([https://blog.csdn.net/blueeyedboy521/category\\_11717413.html](https://blog.csdn.net/blueeyedboy521/category_11717413.html)) 29篇

\* [[Image 88] ([https://i-blog.csdnimg.cn/blog\\_column\\_migrate/93b4cecf8455f24625573e18a314ca7.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/blog_column_migrate/93b4cecf8455f24625573e18a314ca7.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) linux嵌入式]  
([https://blog.csdn.net/blueeyedboy521/category\\_11708612.html](https://blog.csdn.net/blueeyedboy521/category_11708612.html)) 31篇

\* [[Image 89] ([https://i-blog.csdnimg.cn/columns/default/20201014180756930.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756930.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) 开发工具]  
([https://blog.csdn.net/blueeyedboy521/category\\_11722803.html](https://blog.csdn.net/blueeyedboy521/category_11722803.html)) 24篇

\* [[Image 90] ([https://i-blog.csdnimg.cn/blog\\_column\\_migrate/586dad44b33c654c8f94450ee15b9265.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/blog_column_migrate/586dad44b33c654c8f94450ee15b9265.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) idea]  
([https://blog.csdn.net/blueeyedboy521/category\\_10180275.html](https://blog.csdn.net/blueeyedboy521/category_10180275.html)) 1篇

\* [[Image 91] ([https://i-blog.csdnimg.cn/columns/default/20201014180756916.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756916.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) window]



([https://blog.csdn.net/blueeyebboy521/category\\_11722057.html](https://blog.csdn.net/blueeyebboy521/category_11722057.html)) 3篇  
\* [[Image 92]([https://i-blog.csdnimg.cn/columns/default/20201014180756922.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756922.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) C组件库]  
([https://blog.csdn.net/blueeyebboy521/category\\_11723424.html](https://blog.csdn.net/blueeyebboy521/category_11723424.html)) 12篇  
\* [[Image 93]([https://i-blog.csdnimg.cn/columns/default/20201014180756724.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756724.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) activiti]  
([https://blog.csdn.net/blueeyebboy521/category\\_11696767.html](https://blog.csdn.net/blueeyebboy521/category_11696767.html)) 1篇  
\* [[Image 94]([https://i-blog.csdnimg.cn/columns/default/20201014180756916.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756916.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) vue]  
([https://blog.csdn.net/blueeyebboy521/category\\_9388169.html](https://blog.csdn.net/blueeyebboy521/category_9388169.html)) 1篇  
\* [[Image 95]([https://i-blog.csdnimg.cn/columns/default/20201014180756757.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756757.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) 工作]  
([https://blog.csdn.net/blueeyebboy521/category\\_8947356.html](https://blog.csdn.net/blueeyebboy521/category_8947356.html)) 8篇  
\* [[Image 96]([https://i-blog.csdnimg.cn/columns/default/20201014180756925.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756925.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) StringBoot]  
([https://blog.csdn.net/blueeyebboy521/category\\_8947361.html](https://blog.csdn.net/blueeyebboy521/category_8947361.html)) 14篇  
\* [[Image 97]([https://i-blog.csdnimg.cn/columns/default/20201014180756926.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756926.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) SpringMVC]  
([https://blog.csdn.net/blueeyebboy521/category\\_8947366.html](https://blog.csdn.net/blueeyebboy521/category_8947366.html))  
\* [[Image 98]([https://i-blog.csdnimg.cn/columns/default/20201014180756922.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756922.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) docker]  
([https://blog.csdn.net/blueeyebboy521/category\\_9022032.html](https://blog.csdn.net/blueeyebboy521/category_9022032.html)) 14篇

[[Image 99](<https://csdnimg.cn/release/blogv2/dist/pc/img/arrowDownWhite.png>)

### ### 最新评论

\* [[【转载】通俗大白话，彻底弄懂 https 原理本质（ssl证书）]  
([https://blog.csdn.net/Blueeyebboy521/article/details/127938799#comments\\_35233520](https://blog.csdn.net/Blueeyebboy521/article/details/127938799#comments_35233520)) [凉汽水:]  
(<https://blog.csdn.net/yinpengl86>) 写的相当棒了~

\* [ubuntu安装Espeak实现tts文字转语音]  
([https://blog.csdn.net/Blueeyebboy521/article/details/128733872#comments\\_34920033](https://blog.csdn.net/Blueeyebboy521/article/details/128733872#comments_34920033)) [Blueeyebboy521:]  
(<https://blog.csdn.net/Blueeyebboy521>) 我也发现了，这个效果不行，得找找别的方案

\* [ArduinoUNO实战-第二十一章-通过调整电位器来改变9克舵机的角度]  
([https://blog.csdn.net/Blueeyebboy521/article/details/127460250#comments\\_34876585](https://blog.csdn.net/Blueeyebboy521/article/details/127460250#comments_34876585)) [Alex-東:]  
([https://blog.csdn.net/2401\\_87816138](https://blog.csdn.net/2401_87816138)) 报错了呀

\* [ubuntu安装Espeak实现tts文字转语音]  
([https://blog.csdn.net/Blueeyebboy521/article/details/128733872#comments\\_34874957](https://blog.csdn.net/Blueeyebboy521/article/details/128733872#comments_34874957)) [晓晓的独白:]  
(<https://blog.csdn.net/T15272794984>) 有没有好听的中文包呀，这个中文太不标准了![[Image 100: 表情包]  
(<https://g.csdnimg.cn/static/face/emoji/010.png>)

\* [Windows下给瑞芯微ROC-3588-PC烧录Ubuntu20.04.4]  
([https://blog.csdn.net/Blueeyebboy521/article/details/127350531#comments\\_34796739](https://blog.csdn.net/Blueeyebboy521/article/details/127350531#comments_34796739)) [Blueeyebboy521:]  
(<https://blog.csdn.net/Blueeyebboy521>) 教程中有链接地址，去他们官网可以下载

### ### 大家在看

\* [python毕设 旅游景区智能门票系统规划设计论文+程序 ![[Image 101]  
(<https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png>) 810]  
(<https://blog.csdn.net/sheji301/article/details/144501598>)

\* [如何选对产品推广渠道？五大实用建议来啦！]  
([https://blog.csdn.net/d\\_cln/article/details/144529759](https://blog.csdn.net/d_cln/article/details/144529759))

\* [SAM大模型在卫星遥感中得应用实践](<https://blog.csdn.net/1hr123789/article/details/144513417>)

\* [配置使用 catalog 恢复目录做数据库备份]  
([https://blog.csdn.net/baidu\\_17517885/article/details/144524042](https://blog.csdn.net/baidu_17517885/article/details/144524042))

\* [STM32 水质水位检测项目（硬件架构）]  
([https://blog.csdn.net/qg\\_64219867/article/details/144529989](https://blog.csdn.net/qg_64219867/article/details/144529989))

### ### 最新文章

\* [linux下c++调用opencv3.4.16实战技巧]  
(<https://blog.csdn.net/Blueeyebboy521/article/details/128382183>)

\* [大数据Hadoop、Hive实战第一章 配置环境]  
(<https://blog.csdn.net/Blueeyebboy521/article/details/133634419>)

\* [linux入门到精通-第十七章-本地套接字]

(<https://blog.csdn.net/Blueeyedboy521/article/details/131230016>)

[2024年27篇] (<https://blog.csdn.net/Blueeyedboy521?type=blog&year=2024&month=12>)

[2023年45篇] (<https://blog.csdn.net/Blueeyedboy521?type=blog&year=2023&month=12>)

[2022年191篇] (<https://blog.csdn.net/Blueeyedboy521?type=blog&year=2022&month=12>)

[2020年1篇] (<https://blog.csdn.net/Blueeyedboy521?type=blog&year=2020&month=07>)

[2019年9篇] (<https://blog.csdn.net/Blueeyedboy521?type=blog&year=2019&month=10>)

! [Image 102] (<https://kunyu.csdn.net/1.png?p=57&adId=1048398&adBlockFlag=0&a=1048398&c=0&k=%E3%80%90%E8%BD%AC%E8%BD%BD%E3%80%91%E9%80%9A%E4%BF%97%E5%A4%A7%E7%99%BD%E8%AF%9D%EF%BC%8C%E5%BD%BB%E5%BA%95%E5%BC%84%E6%87%82%20https%20%E5%8E%9F%E7%90%86%E6%9C%AC%E8%B4%A8%EF%BC%88ss1%E8%AF%81%E4%B9%A6%EF%BC%89&spm=1001.2101.3001.5001&articleId=127938799&d=1&t=3&u=5f76fd35af9e46d78dac3de4ccce4cb8>)

### ### 目录

! [Image 103] (<https://kunyu.csdn.net/1.png?p=530&adId=1057732&adBlockFlag=0&a=1057732&c=0&k=%E3%80%90%E8%BD%AC%E8%BD%BD%E3%80%91%E9%80%9A%E4%BF%97%E5%A4%A7%E7%99%BD%E8%AF%9D%EF%BC%8C%E5%BD%BB%E5%BA%95%E5%BC%84%E6%87%82%20https%20%E5%8E%9F%E7%90%86%E6%9C%AC%E8%B4%A8%EF%BC%88ss1%E8%AF%81%E4%B9%A6%EF%BC%89&spm=1001.2101.3001.4647&articleId=127938799&d=1&t=3&u=a7fab7ba7f224e98971711c39e071534>)

### ### 目录

! [Image 104] (<https://kunyu.csdn.net/1.png?p=479&adId=1049278&adBlockFlag=0&a=1049278&c=0&k=%E3%80%90%E8%BD%AC%E8%BD%BD%E3%80%91%E9%80%9A%E4%BF%97%E5%A4%A7%E7%99%BD%E8%AF%9D%EF%BC%8C%E5%BD%BB%E5%BA%95%E5%BC%84%E6%87%82%20https%20%E5%8E%9F%E7%90%86%E6%9C%AC%E8%B4%A8%EF%BC%88ss1%E8%AF%81%E4%B9%A6%EF%BC%89&spm=1001.2101.3001.4834&articleId=127938799&d=1&t=3&u=3f97cd8305a543aca0d7cdd0d9255c16>)

### ### 分类专栏

\* [[Image 105] ([https://i-blog.csdnimg.cn/columns/default/20201014180756923.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756923.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) 数据库] ([https://blog.csdn.net/blueeyedboy521/category\\_11772980.html](https://blog.csdn.net/blueeyedboy521/category_11772980.html)) 10篇

\* [[Image 106] ([https://i-blog.csdnimg.cn/columns/default/20201014180756922.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756922.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) java] ([https://blog.csdn.net/blueeyedboy521/category\\_11767600.html](https://blog.csdn.net/blueeyedboy521/category_11767600.html)) 27篇

\* [[Image 107] ([https://i-blog.csdnimg.cn/columns/default/20201014180756918.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756918.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) 深度学习] ([https://blog.csdn.net/blueeyedboy521/category\\_11871122.html](https://blog.csdn.net/blueeyedboy521/category_11871122.html)) 10篇

\* [[Image 108] ([https://i-blog.csdnimg.cn/columns/default/20201014180756925.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756925.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) 物联网] ([https://blog.csdn.net/blueeyedboy521/category\\_12074657.html](https://blog.csdn.net/blueeyedboy521/category_12074657.html)) 4篇

\* [[Image 109] ([https://i-blog.csdnimg.cn/columns/default/20201014180756754.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756754.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) 云计算] ([https://blog.csdn.net/blueeyedboy521/category\\_12010685.html](https://blog.csdn.net/blueeyedboy521/category_12010685.html)) 1篇

\* [[Image 110] ([https://i-blog.csdnimg.cn/columns/default/20201014180756754.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756754.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) 前端] ([https://blog.csdn.net/blueeyedboy521/category\\_11767917.html](https://blog.csdn.net/blueeyedboy521/category_11767917.html)) 1篇


\* [[Image 111] ([https://i-blog.csdnimg.cn/columns/default/20201014180756922.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756922.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) 音视频] ([https://blog.csdn.net/blueeyedboy521/category\\_11767552.html](https://blog.csdn.net/blueeyedboy521/category_11767552.html)) 13篇


\* [[Image 112] ([https://i-blog.csdnimg.cn/columns/default/20201014180756738.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756738.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) SpringCloud] ([https://blog.csdn.net/blueeyedboy521/category\\_11689598.html](https://blog.csdn.net/blueeyedboy521/category_11689598.html)) 31篇


\* [[Image 113] ([https://i-blog.csdnimg.cn/columns/default/20201014180756925.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756925.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) 大数据] ([https://blog.csdn.net/blueeyedboy521/category\\_11722577.html](https://blog.csdn.net/blueeyedboy521/category_11722577.html)) 28篇


\* [[Image 114] ([https://i-blog.csdnimg.cn/columns/default/20201014180756930.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756930.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) 开发组件] ([https://blog.csdn.net/blueeyedboy521/category\\_11740291.html](https://blog.csdn.net/blueeyedboy521/category_11740291.html)) 54篇


\* [[Image 115] ([https://i-blog.csdnimg.cn/columns/default/20201014180756913.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756913.png?x-oss-process=image/resize,m_fixed,h_64,w_64)) RabbitMQ] ([https://blog.csdn.net/blueeyedboy521/category\\_11767722.html](https://blog.csdn.net/blueeyedboy521/category_11767722.html)) 9篇


\*  [https://i-blog.csdnimg.cn/columns/default/20201014180756928.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756928.png?x-oss-process=image/resize,m_fixed,h_64,w_64) linux] [https://blog.csdn.net/blueeyedboy521/category\\_11717413.html](https://blog.csdn.net/blueeyedboy521/category_11717413.html)) 29篇


\*  [https://i-blog.csdnimg.cn/blog\\_column\\_migrate/93b4cecf8455f24625573e18a314ca7.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/blog_column_migrate/93b4cecf8455f24625573e18a314ca7.png?x-oss-process=image/resize,m_fixed,h_64,w_64) linux嵌入式] [https://blog.csdn.net/blueeyedboy521/category\\_11708612.html](https://blog.csdn.net/blueeyedboy521/category_11708612.html)) 31篇


\*  [https://i-blog.csdnimg.cn/columns/default/20201014180756930.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756930.png?x-oss-process=image/resize,m_fixed,h_64,w_64) 开发工具] [https://blog.csdn.net/blueeyedboy521/category\\_11722803.html](https://blog.csdn.net/blueeyedboy521/category_11722803.html)) 24篇


\*  [https://i-blog.csdnimg.cn/blog\\_column\\_migrate/586dad44b33c654c8f94450ee15b9265.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/blog_column_migrate/586dad44b33c654c8f94450ee15b9265.png?x-oss-process=image/resize,m_fixed,h_64,w_64) idea] [https://blog.csdn.net/blueeyedboy521/category\\_10180275.html](https://blog.csdn.net/blueeyedboy521/category_10180275.html)) 1篇


\*  [https://i-blog.csdnimg.cn/columns/default/20201014180756916.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756916.png?x-oss-process=image/resize,m_fixed,h_64,w_64) window] [https://blog.csdn.net/blueeyedboy521/category\\_11722057.html](https://blog.csdn.net/blueeyedboy521/category_11722057.html)) 3篇


\*  [https://i-blog.csdnimg.cn/columns/default/20201014180756922.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756922.png?x-oss-process=image/resize,m_fixed,h_64,w_64) C组件库] [https://blog.csdn.net/blueeyedboy521/category\\_11723424.html](https://blog.csdn.net/blueeyedboy521/category_11723424.html)) 12篇


\*  [https://i-blog.csdnimg.cn/columns/default/20201014180756724.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756724.png?x-oss-process=image/resize,m_fixed,h_64,w_64) activiti] [https://blog.csdn.net/blueeyedboy521/category\\_11696767.html](https://blog.csdn.net/blueeyedboy521/category_11696767.html)) 1篇

\*  [https://i-blog.csdnimg.cn/columns/default/20201014180756916.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756916.png?x-oss-process=image/resize,m_fixed,h_64,w_64) vue] [https://blog.csdn.net/blueeyedboy521/category\\_9388169.html](https://blog.csdn.net/blueeyedboy521/category_9388169.html)) 1篇


\*  [https://i-blog.csdnimg.cn/columns/default/20201014180756757.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756757.png?x-oss-process=image/resize,m_fixed,h_64,w_64) 工作] [https://blog.csdn.net/blueeyedboy521/category\\_8947356.html](https://blog.csdn.net/blueeyedboy521/category_8947356.html)) 8篇



\*  [https://i-blog.csdnimg.cn/columns/default/20201014180756925.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756925.png?x-oss-process=image/resize,m_fixed,h_64,w_64) StringBoot] [https://blog.csdn.net/blueeyedboy521/category\\_8947361.html](https://blog.csdn.net/blueeyedboy521/category_8947361.html)) 14篇


\*  [https://i-blog.csdnimg.cn/columns/default/20201014180756926.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756926.png?x-oss-process=image/resize,m_fixed,h_64,w_64) SpringMVC] [https://blog.csdn.net/blueeyedboy521/category\\_8947366.html](https://blog.csdn.net/blueeyedboy521/category_8947366.html))

\*  [https://i-blog.csdnimg.cn/columns/default/20201014180756922.png?x-oss-process=image/resize,m\\_fixed,h\\_64,w\\_64](https://i-blog.csdnimg.cn/columns/default/20201014180756922.png?x-oss-process=image/resize,m_fixed,h_64,w_64) docker] [https://blog.csdn.net/blueeyedboy521/category\\_9022032.html](https://blog.csdn.net/blueeyedboy521/category_9022032.html)) 14篇

### ### 目录

评论  <https://csdnimg.cn/release/blogv2/dist/pc/img/closeBt.png>)

 <https://csdnimg.cn/release/blogv2/dist/pc/img/commentArrowLeftWhite.png>) 被折叠的 条  
评论 [为什么被折叠?] <https://blogdev.blog.csdn.net/article/details/122245662>)  <https://csdnimg.cn/release/blogv2/dist/pc/img/iconPark.png>) 到【灌水乐园】发言]  
<https://bbs.csdn.net/forums/FreeZone>)

查看更多评论!  <https://csdnimg.cn/release/blogv2/dist/pc/img/commentArrowDownWhite.png>)

添加红包

祝福语

请填写红包祝福语或标题

红包数量

个

红包个数最小为10个

红包总金额

元

红包金额最低5元

余额支付

当前余额3.43元 [前往充值 \>] (<https://i.csdn.net/#/wallet/balance/recharge>)

需支付: 10.00元

取消 确定

![Image 132] (<https://csdnimg.cn/release/blogv2/dist/pc/img/guideRedReward02.png>) 下一步

![Image 133] (<https://csdnimg.cn/release/blogv2/dist/pc/img/guideRedReward03.png>) 知道了

![Image 134] (<https://blog.csdn.net/Blueeyedboy521/article/details/127938799?spm=1001.2101.3001.4242.1>)

成就一亿技术人!

领取后你会自动成为博主和红包主的粉丝 规则

![Image 135] (<https://profile-avatar.csdnimg.cn/default.jpg!2>)

hope\\_wisdom

发出的红包

实付元

[使用余额支付] (javascript:;)

![Image 136] (<https://csdnimg.cn/release/blogv2/dist/pc/img/pay-time-out.png>) 点击重新获取

![Image 137] (<https://csdnimg.cn/release/blogv2/dist/pc/img/weixin.png>) ![Image 138] (<https://csdnimg.cn/release/blogv2/dist/pc/img/zhifubao.png>) ![Image 139] (<https://csdnimg.cn/release/blogv2/dist/pc/img/jingdong.png>) 扫码支付

钱包余额 0

![Image 140] (<https://csdnimg.cn/release/blogv2/dist/pc/img/pay-help.png>)

抵扣说明:

1. 余额是钱包充值的虚拟货币, 按照1:1的比例进行支付金额的抵扣。
2. 余额无法直接购买下载, 可以购买VIP、付费专栏及课程。

![Image 141] (<https://csdnimg.cn/release/blogv2/dist/pc/img/recharge.png>) 余额充值] (<https://i.csdn.net/#/wallet/balance/recharge>)

![Image 142] (<https://blog.csdn.net/Blueeyedboy521/article/details/127938799?spm=1001.2101.3001.4242.1>)