HansChen的博客

博观而约取，厚积而薄发

原 https安全在哪里，原理是什么？

标签： https rsa 安全 CA 加密

2016-08-15 21:19 1484人阅读 评论(0) 收藏 举报

分类：

https（1）

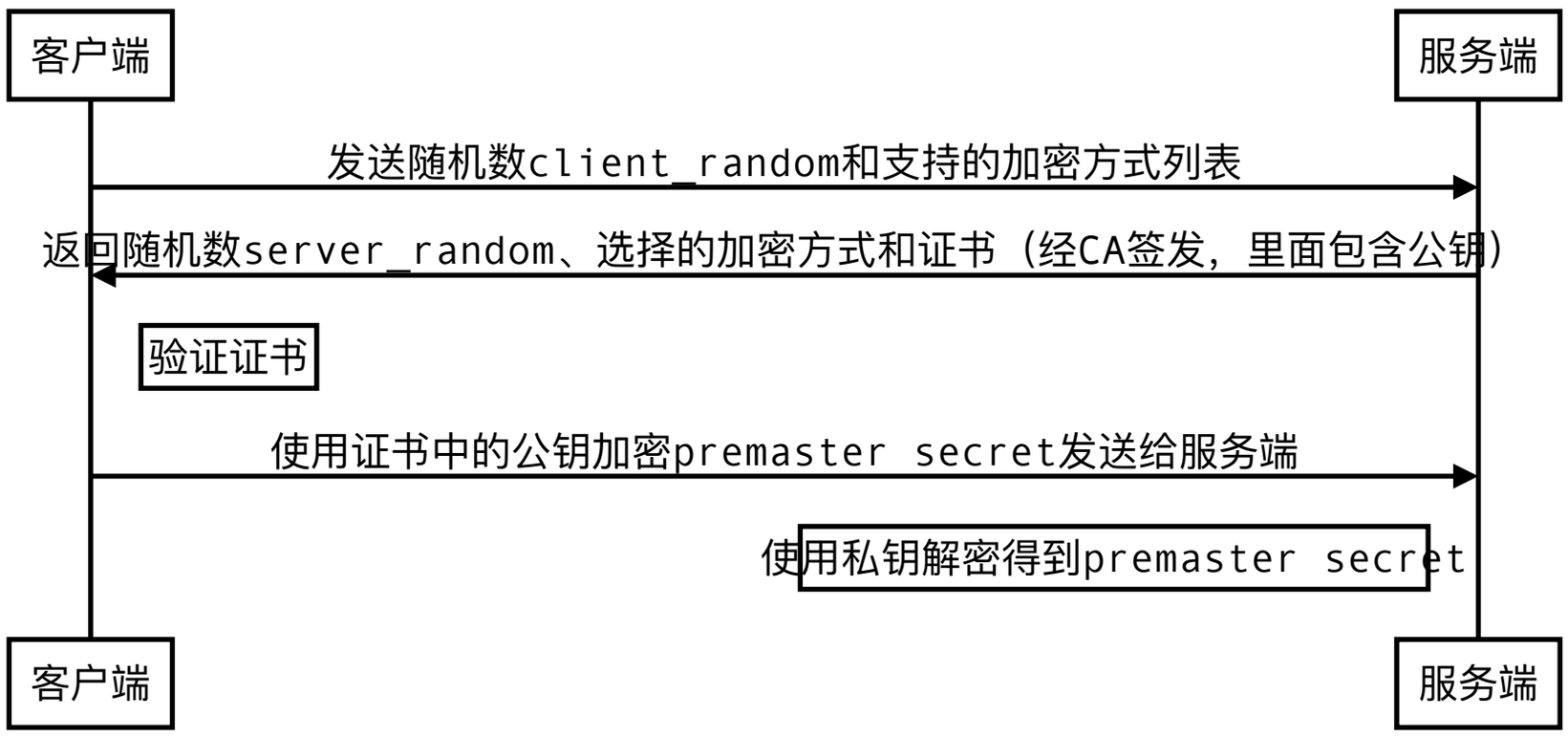
目录(?) [+]

目录(?) [+]

转载请标明出处：http://blog.csdn.net/shensky711/article/details/52214842
本文出自：【HansChen的博客】

Https通信基本过程

在通信过程中，https是如何保证通信的安全的？如何加密信息，如何防止中间人攻击？
以下是客户端发起https请求的时候的流程：



流程解析如下：

1. 客户端发送随机数client_random和支持的加密方式列表
2. 服务器返回随机数server_random，选择的加密方式和证书（经过ca颁发，或者自签名的证书，该证书包含公钥）
3. 客户端验证证书，使用证书中的公钥加密premaster secret发送给服务端
4. 服务端使用私钥解密得到premaster secret
5. 两端分别通过client_random，server_random和premaster secret生成master secret，用于对称加密后续通信内容

HansChen_

+ 加关注 发私信

访问： 42634次
积分： 684
等级： BLOG 3
排名： 千里之外

原创： 25篇 转载： 0篇
译文： 0篇 评论： 20条

阅读排行

Android Studio 2.2 NDK (9538)
Robolectric使用教程 (3856)
Fragment事务管理源码分 (3722)
Mock Server利器 - Moco (2755)
ProGuard代码混淆详细攻 (2505)
Android分包MultiDex源码 (2382)
Android最大方法数和解决 (2056)
MockWebServer使用指南 (1741)
Mockito使用指南 (1671)
依赖注入利器 - Dagger 2 (1590)

文章分类

Android (7)
Java (8)
Linux (0)
单元测试 (3)
工具使用 (2)
综合 (2)
杂七杂八 (2)
https (2)

联系方式

主页：http://blog.hanschen.site
GitHub：http://github.com/shensky7
Email：shensky711@gmail.com

整个过程主要的作用是让双方安全地协商出一个key，这个key会用于对称加密中。第三方即使截取了所有的通信数据，也是无法获取到这个key的。既然第三方无法获取这个key，自然也对加密过的数据无可奈何了。大家看到这里可能一脸懵逼，可能会有以下疑问：

- 这个过程是如何保证key不会被中间人窃取呢?
- 客户端/服务端如何确认对方就是“正确的人”，而不是其他中间人呢?

什么是RSA非对称加密

在解答上面的问题之前，首先我们得先了解一些基本的知识：

> RSA非对称加密：RSA分为公钥和私钥，从私钥可以生成公钥，但是不能通过公钥生成私钥。公钥加密过的信息，只有私钥能解开，私钥加密的信息，只有公钥能解开

https如何保证key不会被中间人窃取

在步骤（1）中，客户端的随机数client_random是完全可以被中间人窃取的，然后在步骤（2）中服务端返回的server_random也是完全可被中间人窃取的。关键是在步骤（3），客户端会把生成的premaster secret通过公钥进行加密，然后再发送给服务器，中间人当然也可以窃取加密后的premaster secret数据，但是中间人却不能解密出原始的premaster secret，这是为什么呢？**因为公钥加密的数据，只有私钥能解开**，而私钥是保存在服务端，不会外泄的！通过步骤1-4，服务器和客户端相互持有了client_random，server_random和premaster secret，而且只有客户端和服务端才有premaster secret，中间人是没有的。这时候通过前面三个key，生成master secret用于对称加密，确保通信安全。

为何最终使用对称加密，而不是全部通信都使用非对称加密呢？猜测是因为非对称加密效率和速度不如对称加密。而且对称加密的安全性并不是不高，对称加密的难点在于如何安全地交换key。

我一开始理解https的时候，遇到一个困惑：如果中间人从建立连接一开始就冒充服务器，转发客户端和服务端的所有数据，那么所有数据在中间人眼里应该都是透明的啊，中间人应该也能解密通信数据啊？是的，中间人确实是可以拿到所有数据，但是，中间人没有服务器的私钥！所以即使拿到了数据，也不能得到对称加密的key。其实说白了，一个https请求，不知会经过多少个中间人呢，所有路由转发都有可能是中间人，都有可能攻击你，但恰恰就是因为没有私钥而不能窃取数据，他们只能转发数据，但却不能解密数据。

如何确认对方就是“正确的人”，而不是其他中间人呢

但是问题又来了，虽然通信内容不会被第三个人窃取了，但是我如何保证对方就是我想要找的人呢？比如我要访问www.baidu.com，确实有一个服务器给了我回复，但我怎么确定这个是真的“百度”给我的回复呢？万一我的请求被劫持了呢？

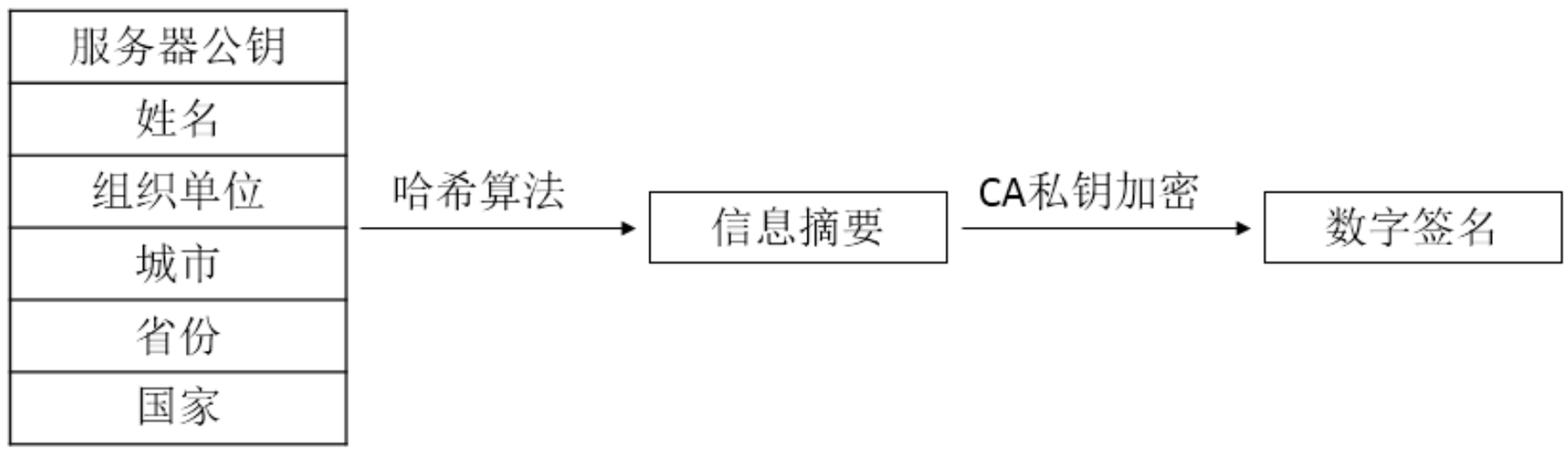
这个就得依靠验证步骤（2）里的证书了。

什么是证书呢？数字证书就是一个人或者组织在网络世界中的身份证，其发证机关是证书管理机构(certificate authority,CA)，在这里CA是一个权威的机构，我们可以信任他，他信任的站点，我们也会认为是可信任的。个人电脑上无法对每一个网站都进行验证，因为这样几乎不可能，也不方便。在日常生活中，如果我们要验证一个人的身份，通常的做法是查看他的身份证。我们信任身份证颁发机构即政府机构的公信力，因此只要验证一个人的身份证不是伪造的，我们就相信这个人的身份和身份证上所描述的是一致的。

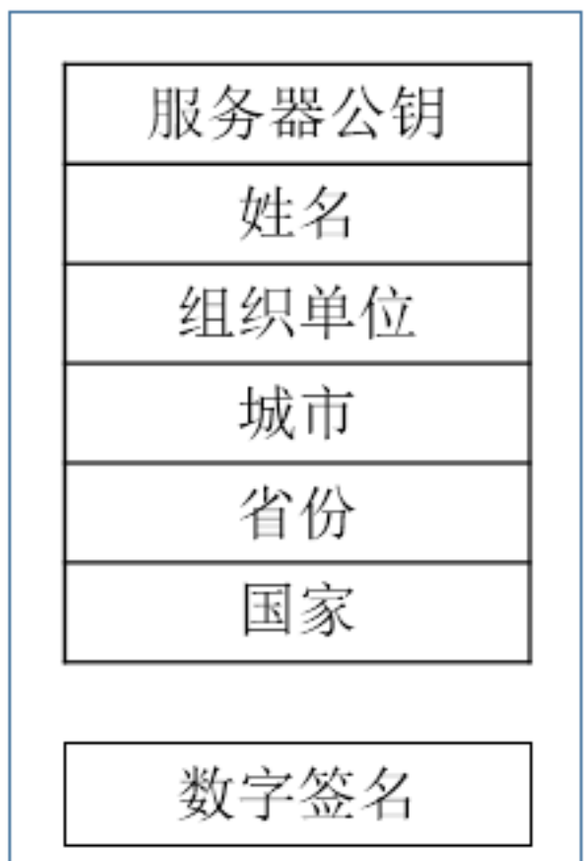
说到这里，又有同学可能要懵逼了，通俗点讲，就是所有网站都要去CA机构那里去登记，然后CA会发给那么网站一个“身份证”。但是我们如何验证一个人身份证的真伪呢？CA机构也会提供一个工具给我们，我们用那个工具就可以验证身份证的真伪。

- 网站身份证：网站证书，需要CA机构签发
- 真伪辨认工具：CA证书

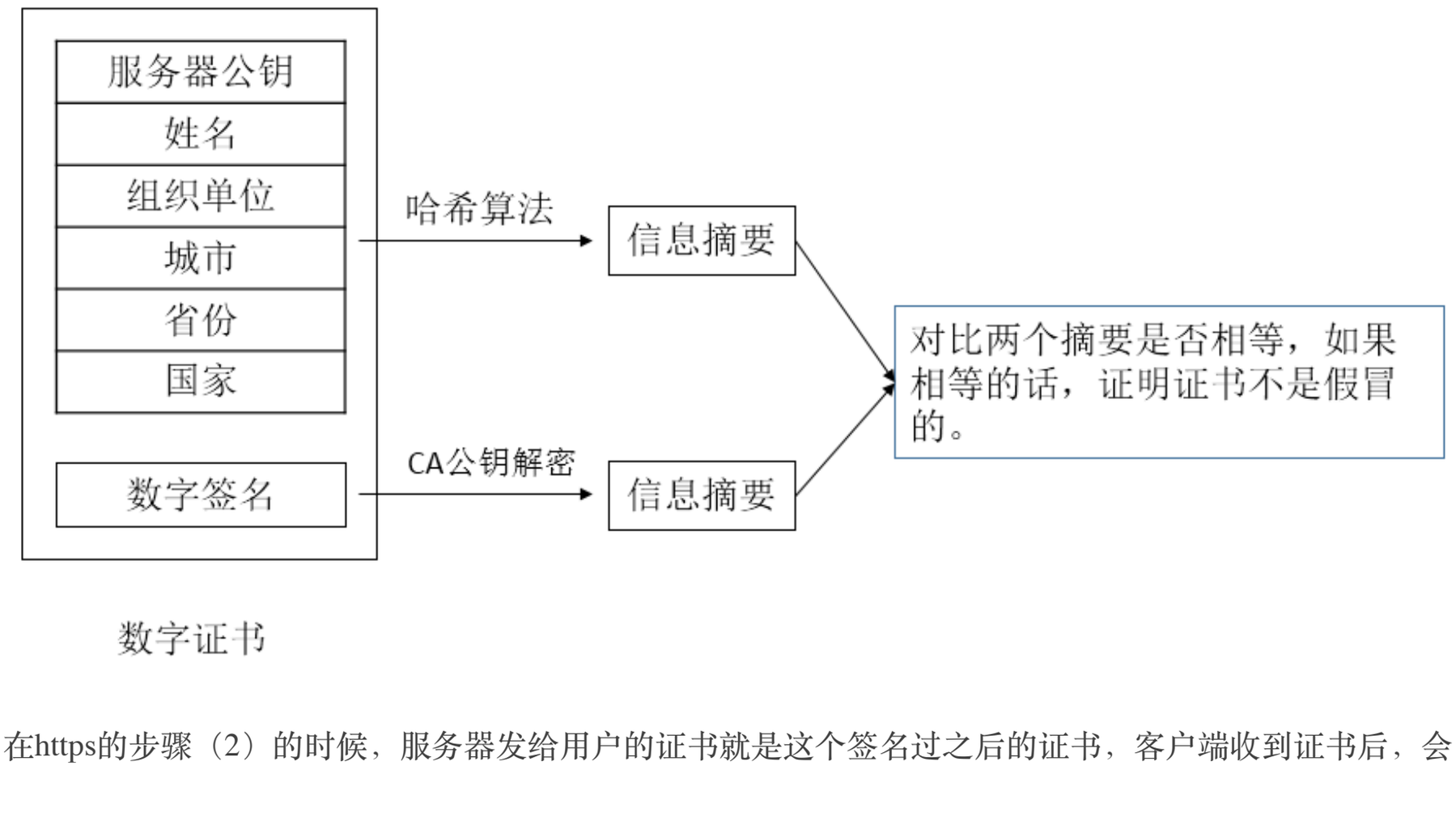
那么，什么是CA颁发的“身份证”呢？



1. 服务端生成自己的证书请求文件（尚未被CA签名），里面包含了姓名、服务器私钥对应的公钥等信息
2. CA机构对该证书进行签名，也就是生成数字签名，注意，这个签名是用**CA的私钥加密过的**
3. 把原始的证书和生成的数字签名合并在一起，形成证书



数字证书



数字证书

在https的步骤（2）的时候，服务器发给用户的证书就是这个签名过之后的证书，客户端收到证书后，会

使用CA的公钥（这个是内置在浏览器的）对数字签名进行解密得出一个信息摘要，然后用哈希算法自己算出信息摘要，对比摘要，一致的话，证明该证书是CA机构颁发的。因为公钥只能解开私钥加密的数据，如果信息摘要是匹配的，那么证明该加密数据是由CA机构用私钥加密的，证书是可靠的。

到现在，我们终于可以愉快地确定对方的身份，愉快地通信了。主要是依赖一个CA公钥来判别对方证书的真伪。

但是会有同学问了，万一冒牌网站把正牌网站copy下来，转发给我了怎么办，转发的证书是由正牌网站copy的，肯定是真的，所以客户端可以验证通过的，那怎么办？确实是的，如果冒牌网站用正版网站的证书来忽悠客户端，那么客户端确实是会被“忽悠”过去的，但是不用担心，客户端是依靠证书上的公钥来生成premaster secret的，而公钥对应的私钥，冒牌网站是不可能拿得到的，也就不可能解密出正确的premaster secret，自然也无法正常和客户端正常沟通了。

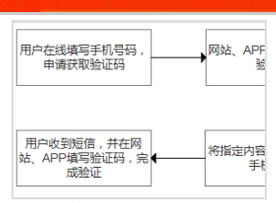
现在很多**Android**应用的服务端虽然采用的是https，但是却是没有经过ca机构认证的（因为要花钱），所以一般会自己给自己颁发数字证书（自己充当CA）。但国内很多开发者在**android**应用里面采用的做法是信任所有证书，这样是很不安全的，正确的做法应该是导入CA的证书，这样才能在拿到证书后，判断证书的真伪。

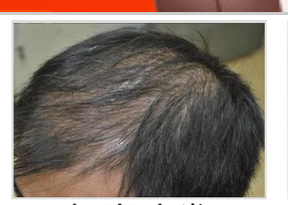
上一篇 Git使用和介绍-基础指令
下一篇 利用keytool、openssl生成证书文件

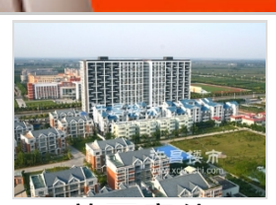
相关文章推荐

- https比http到底那里安全?
- 【Bugly干货分享】“HTTPS”安全在哪里?
- Https为什么是安全的
- 网站安全之HTTPS部署解决方案
- 详解Https是如何确保安全的?
- 八大免费SSL证书-给你的网站免费添加Https安全…
- HTTPS为什么安全 &分析 HTTPS 连接建立全过程
- 带你了解HTTPS和HTTP的区别，数据安全时代的…
- HTTPS 互联网世界的安全基础
- HTTPS和HTTP区别和联系?


三级分销平台


短信验证码接


短信接口

加密头发

验证码短信接

美国房价

新西兰留学3+

在线听英语


英语入门学习

做网站

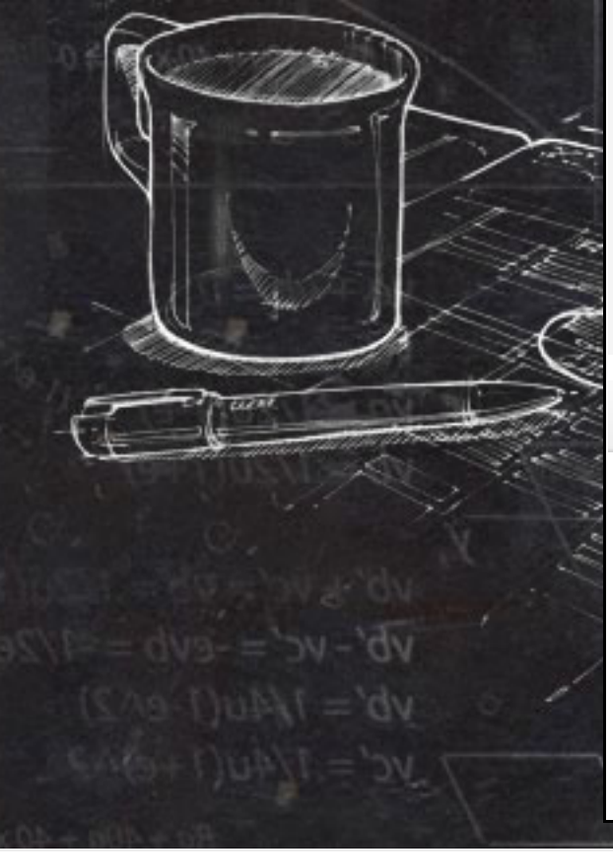

英语学习班

自动贴标签机

开发一个app多

页游排行

猜你在找



- 【直播】计算机视觉原理及实战—屈教授
- 【套餐】Hadoop生态系统零基础入门--侯勇蛟
- 【套餐】2017软考系统集成项目——任铄
- 【直播】广义线性模型及其应用——李科
- 【直播】机器学习之凸优化——马博士
- 【套餐】深度学习入门视频课程—唐宇迪
- 【套餐】嵌入式Linux C编程基础--朱有鹏
- 【套餐】Android 5.x顶级视频课程——李宁
- 【直播】从0到1 区块链的概念到实践
- 【套餐】微信订阅号+服务号Java版 v2.0--翟东平

查看评论

暂无评论

您还没有登录,请[登录]或[注册]

* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

公司简介 | 招贤纳士 | 广告服务 | 联系方式 | 版权声明 | 法律顾问 | 问题报告 | 合作伙伴 | 论坛反馈

网站客服 杂志客服 微博客服 webmaster@csdn.net 400-660-0108 | 北京创新乐知信息技术有限公司 版权所有 | 江苏知之为计算机有限公司 |

江苏乐知网络技术有限公司

京 ICP 证 09002463 号 | Copyright © 1999-2017, CSDN.NET, All Rights Reserved