使用PHP，JS，RSA实现安全登录

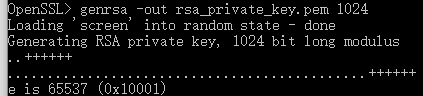
首先需要生成RSA公玥和私玥，下载openssl通过命令行生成：

进入openssl的bin目录下，打开openssl.exe命令行输入一下命令：

1. 生成RSA私钥

genrsa -out rsa\_private\_key.pem 1024 ；

得到exponent: 10001



1. 生成modulus

rsa -in rsa\_private\_key.pem -noout -modulus



1. 生成RSA公钥

rsa -in rsa\_private\_key.pem -pubout -out rsa\_public\_key.pem

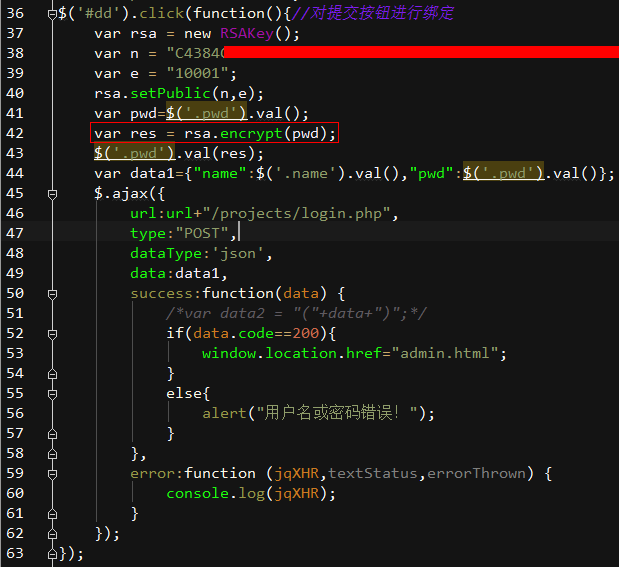


则当前文件夹下的rsa\_private\_key.pem和rsa\_public\_key.pem文件即RSA私玥和公玥文件。

以下展示如何在客户端和服务器端使用公私玥

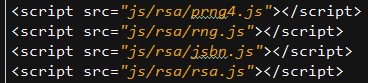
为了实现安全登录，采用公玥加密，私玥解密的方式，在前端将登录密码通过RSA加密后传输，在服务器端进行解密比较，实现登录验证。

前端JS代码



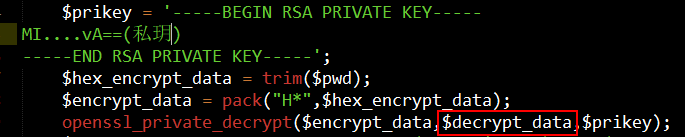
38行的n即代表此前生成的modulus，e即10001，42行对代码密码进行加密，44行对用户名和密码进行传输，由此实现密码的安全传输。

PS：改加密过程需要使用到4个JS库，需要在html文件中引入库，分别如下：



由于库之间存在依赖关系，需要注意引用顺序（上图顺序尚可）。

服务器端代码



其中$pwd即传输到后台的加密后的密钥，首先对其进行十六进制数据转换，然后使用私玥解密，得到的$decrypt\_data即解密后的数据。

其后对数据进行判断操作即可。

其实真的过程很简单，网上页有很多教程，但是在实验过程中，由于自己多此一举的将一长串私玥缩进成一行，导致解密数据不正确，无法完成验证，找了很久的原因才解决此问题，切记：直接将rsa\_private\_key.pem中文本拷贝到私玥变量中即可！切记切记！

后记：这个问题真的想杀自己的心都有了，浪费自己时间，不过还好这么一折腾倒是对这个加密过程更加熟悉了。