**问题：使用HTTPS协议，用户登录功能，在提交时，分析数据包前端加密和后端加密的优缺点。**

**前端加密的优点**

前端加密，即在数据发送前，将数据进行哈希或使用公钥加密。如果数据被中间人获取，拿到的则不再是明文。

1. 前端加密可以帮助后端分担计算压力
2. 增加撞库成本
3. 即使信息被截获，中间人也无法从截取的数据中得知用户的密码信息
4. 结合验证码进行前端加密较为安全，保证了密码在传输过程中的资料安全，即使攻击者拿到了数据也无法重放
5. 前端加密不是决定性的保护措施，但却是一种有意义的低成本安全增强方案。
6. 如果东西已经丢了，那么丢了密文总比丢了明文强
7. 保证后端日志等不会记录明文密码（也可以防止内鬼盗窃）
8. 保证后端内存中无用户明文密码，在 dump 等情况发生时不会出现泄露问题
9. 与 HTTPS 的流程相比，在前端散列一下几乎不影响网站响应速度和用户体验

**前端加密的缺点：**

1. 由于黑客知道密码是在前端进行哈希的，所以他不需要爆破出该md5对应的原文是什么，而是直接修改客户端向服务器发出的请求，把密码字段换成数据库中MD5就可以了，由于与数据库中记录一致，直接就会登录成功。这跟直接存储明文密码没有任何区别
2. 窃听者根本不需要原始密码，只要通过哈希结果就可以伪造请求登录系统
3. 并不能真正意义上改善网站的安全性
4. 前端进行加密会造成页面需要js脚本才能运行，那么假设你的系统需要兼容不能运行js的客户端，就必须再设计一个使用原文的登录接口。
5. 由于前端是不是加密，所有安全机制都必须照常应用，所以为系统增加这样的复杂性是完全没必要的，即使传输明文密码

**后端加密缺点：**

1. HTTPS只保证通信过程的安全，不保证服务器上数据的隐秘性
2. 使用TLS会导致客户端和服务端的性能下降。这些性能瓶颈在IoT设备，RFID设备上较为明显。
3. 我们无法完全信任服务器

**后端加密优点**：

1. 不管前端是不是加密了密码，使用HTTPS安全连接进行登录都是非常有必要的
2. 只要正确使用了HTTPS连接和服务器端安全的哈希算法，密码系统都可以是很安全的