WIRESHARK SSL实验报告

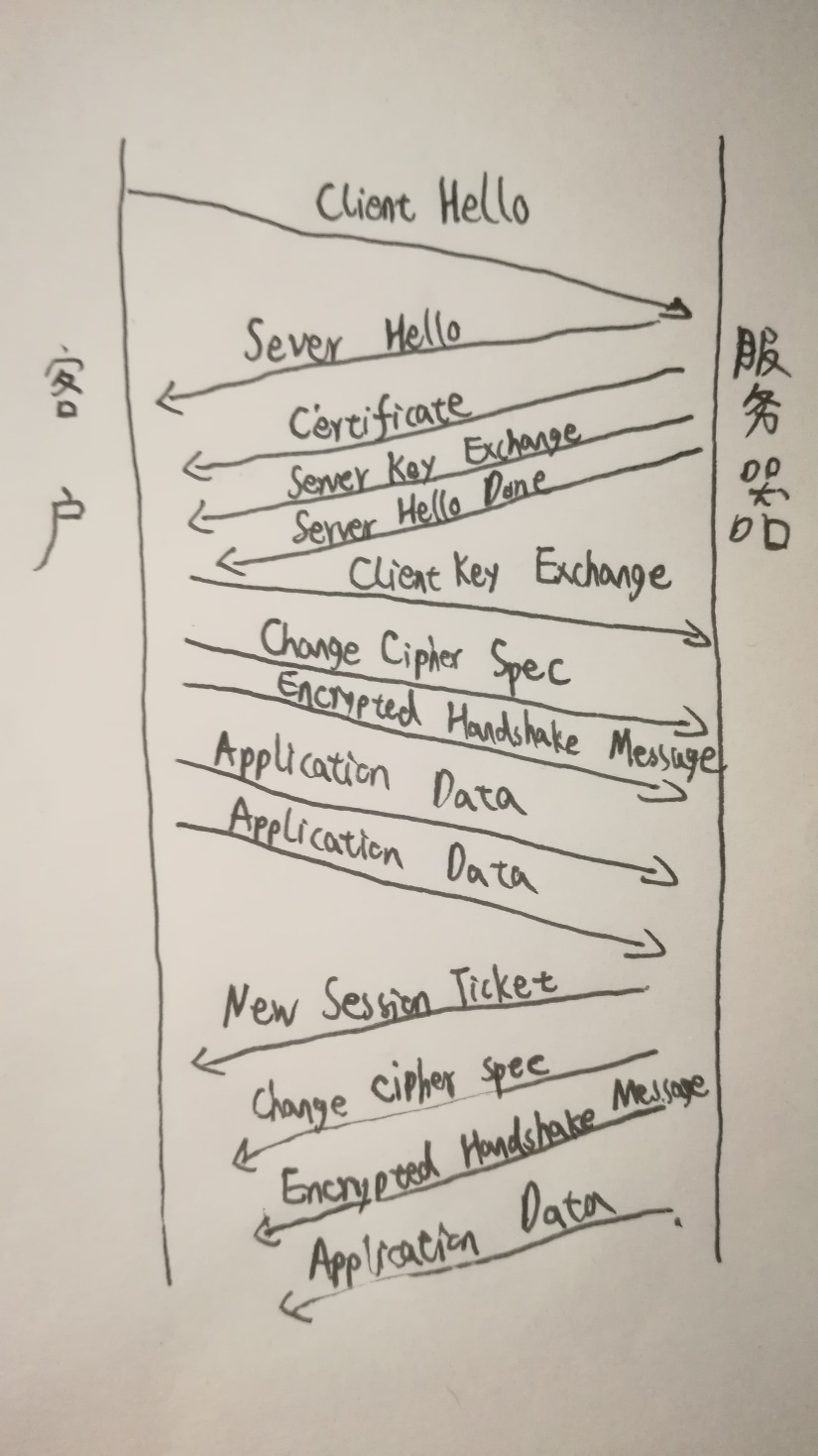
王嵘晟

PB17111614

1. 前八个帧为：



所以时序图为：



1. Content Type : 1 byte

Version: 2 bytes

Lenth: 2 bytes

1. Hand Shake(22)
2. 5ba9 205e b2d5 1f76 c072 38c3 3e66 9285 e000 b247 3a4f 0f95 5ec2 3423 0dd6 5678
3. AES公钥 GCM对称密钥 SHA256哈希
4. TLS\_ECDHE\_ECDSA公钥 AES GCM对称密钥 SHA256哈希
5. 有，长度为32，28字节给数据，4字节给时间，目的是防止replay attack
6. 没有，会话ID的目的是为SSL会话提供一个唯一的持久标识符，该标识符在clear中发送。客户端在发送ClientHello时可以使用服务器提供的会话ID来恢复同一会话。
7. 没有certificate certificate包含在一个单独的记录中，适用于一个以太网帧
8. 没有主密钥，服务器和客户端都使用预主密钥来生成主密钥，主密钥用于生成一组用于MAC和加密的会话密钥。这个秘密是使用服务器的公钥加密的，客户机已经从服务器发送给他们的证书记录中提取了公钥。加密的机密是128字节长。
9. 更改密码规范记录的目的是指示客户机发送的后续SSL记录（数据，而不是头）的内容将被加密。6字节长
10. 在加密握手记录中，生成从该客户端发送的所有先前握手消息的连接的MAC并发送到服务器。
11. 是的，服务器还向客户端发送更改密码规范记录和加密握手。服务器的加密握手记录与客户端发送的握手记录不同，因为它包含从服务器而不是从客户端发送的所有握手消息的串联。否则，记录与客户发送的记录相同。
12. 应用程序数据使用在握手阶段选择的对称密钥加密算法（在我的例子中是RC4）加密，使用使用使用预主密钥和nonce（来自客户端和服务器）生成的对称加密密钥。客户端加密密钥用于加密从客户端发送到服务器的数据，服务器加密密钥用于加密从服务器发送到客户端的数据。包含应用程序数据的记录确实包含MAC；但是，Wireshark不区分加密的应用程序数据和MAC
13. Nothing is interesting