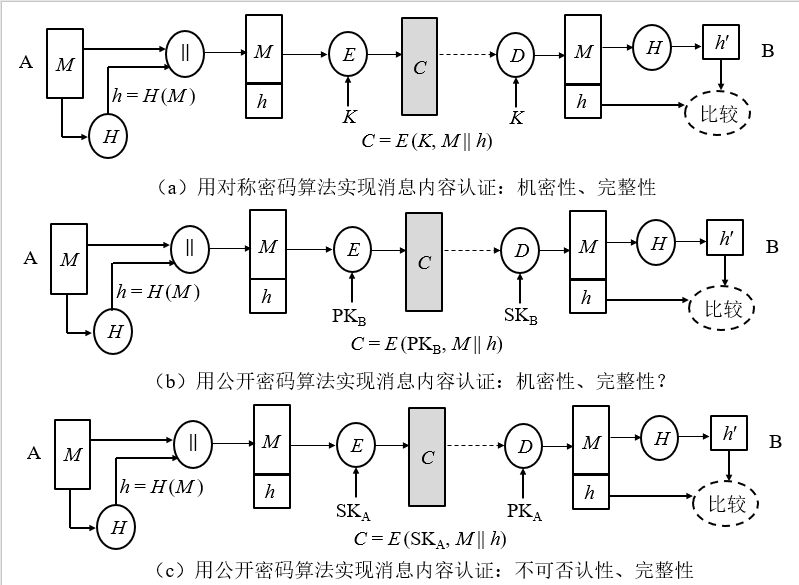
简答

**1.DES计算 p64**

**2.一个密文实现图用公式表示**

消息认证利用MAC码 p105

**对明文和散列值的整体加密的方法进行消息认证**



E代表加密，C代表密文。PKb代表B的公钥。SKa代表A的私钥

问题1：方案(b)不能保证完整性。虽然攻击者不能解密消息后再计算消息后的散列码，但因为攻击者也知道B的公钥，完全可以把原始消息截下来后，再伪造（而不是篡改）一条消息，计算散列值并用B的公钥加密后发送给B，B是无法知道这是一条做伪造的消息。但可以保证机密性，解密需要B的私钥。实际应用中，用公钥算法加密就是用对方的公钥加密的。

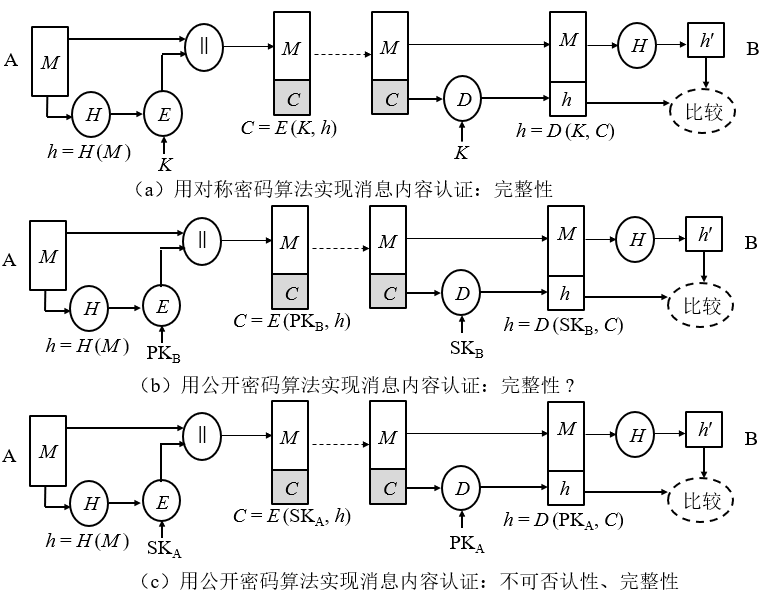
问题2：性能问题，实际应用中，一般是对散列值用公钥算法（自己的私钥）进行加密，再用对称密码算法对消息（可带也可不带上加密后的散列值）进行加密，来保证机密性、完整性和真实性

方案(c)还可以保证不可否认性，但不可保证机密性，因为它是用自己的私钥对M、h进行加密，是后面要介绍的数字签名

感觉消息完整性和身份认证性是挂钩的

**用仅加密消息的散列值的方法散列值进行消息认证**，这时没有机密性，方案2甚至不能保证完整性，方案3还有不可否认性，是数字签名。

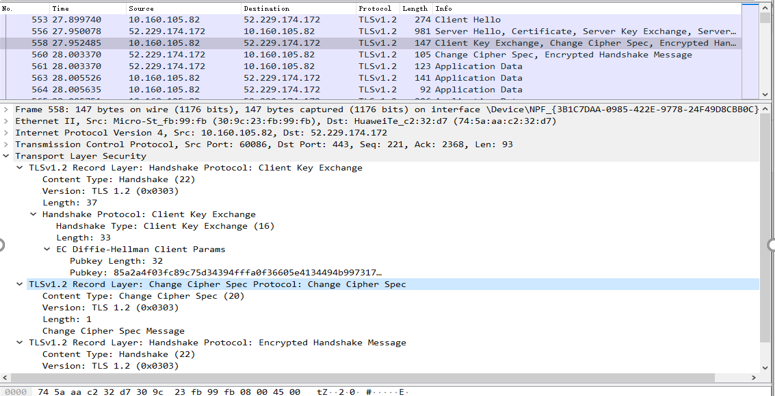
**要想有机密性，M/C在传输给B是还要使用对称密钥加密，或用B的公钥加密**



**3.wireshark抓SSL、TLS包**

* SSL采用的是链式加密（数字信封）方法：采用对称加密算法加密消息（SSL记录协议），用公开密码算法交换对称加密算法的对称密钥（SSL握手协议）

**密码变更规格协议**

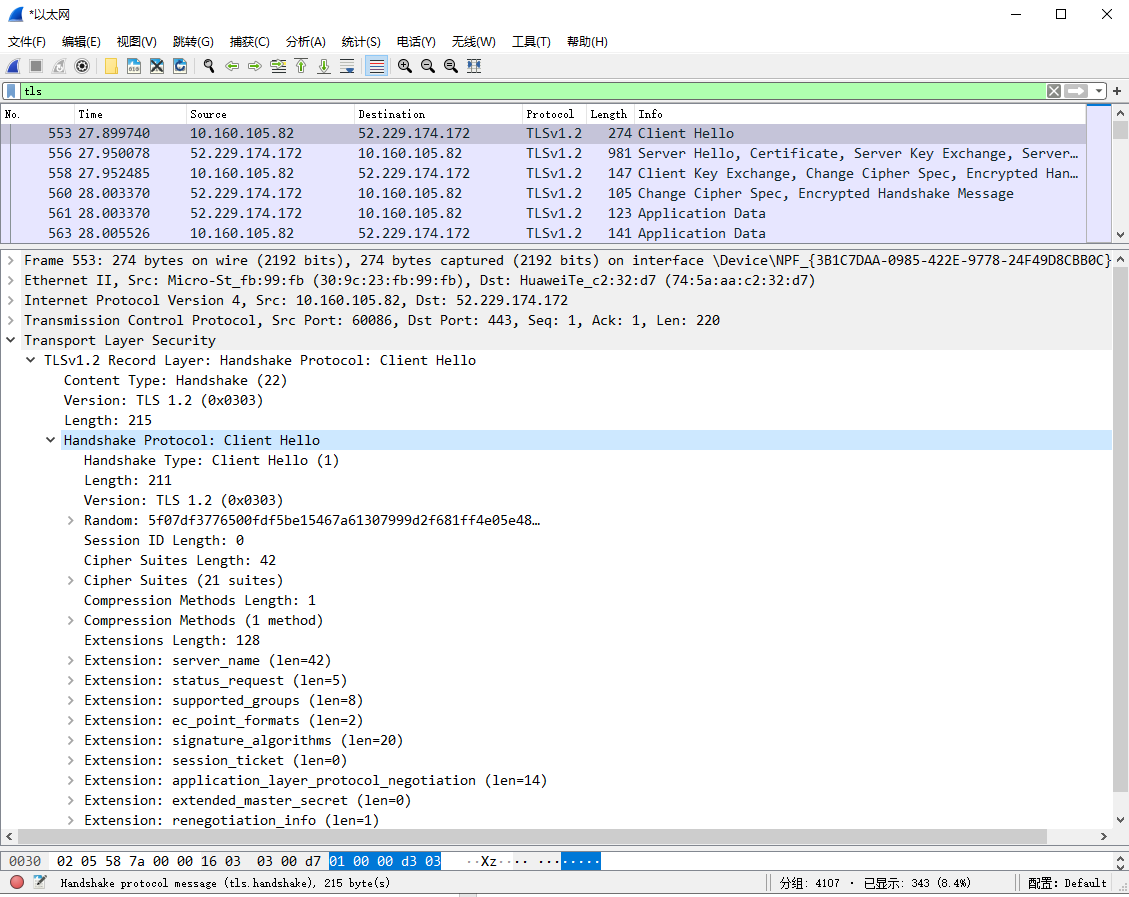


一旦握手商定了一组新的密钥，都会发送一条变更规格协议报文指示启用新的密钥

问题：为什么不作为其它协议（如握手协议）的一条报文而要独立成一个协议？

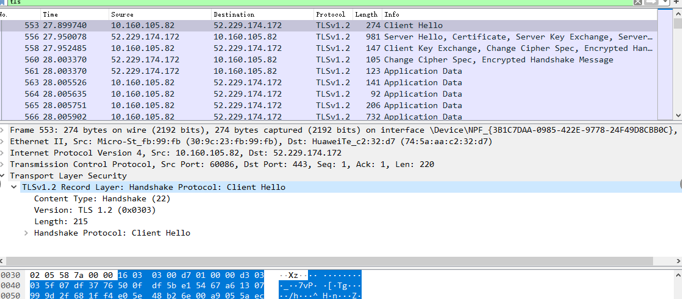
答：为了保障SSL传输过程的安全性，SSL协议要求客户端或服务器端每隔一段时间必须改变其加解密参数。当某一方要改变其加解密参数时，就发送一个简单的消息通知对方下一个要传送的数据将采用新的加解密参数，也就是要求对方改变原来的安全参数。因此，无论是从功能上还是从可扩展性来讲，将其独立出来，而不是做为握手协议的一部分更合适。

**握手协议 Clienthello**

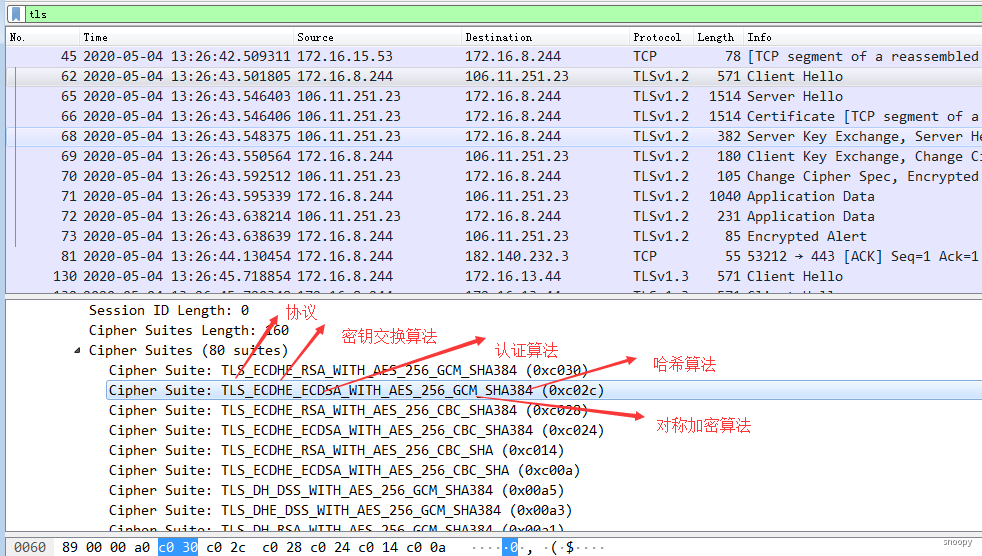
****

**HTTPS**

**TLS的用户首先向服务器的指定端口（443）发起TCP连接请求。成功建立TCP连接后，向服务器发送TLS ClientHello，开始TLS握手过程。当TLS握手过程完成后，用户将发起第一次HTTP请求。此时，所有HTTP协议数据都要以TLS应用数据的形式通过TLS记录协议加密传输。在此过程中，要遵守HTTP协议规范，包括保留连接**



**Clienthello补充**



SSL密钥交换方法有多种，如利用RSA 公钥算法（一般称为数字信封，在PGP那一节讲过），利用DH算法等，教材里那段表述采用的是RSA公钥算法来交换预主密钥。如果是用DH算法则交换的是DH参数，然后根据参数各自计算出预共享密钥。目前更常见的方法是用DH来交换（通常是基于椭圆曲线的DH，即ECDH）。采用哪种交换算法，是在SSL第一阶段里协商的（如果支持DH，则DH算法的全局参数也在这个阶段交换的）

课本上的SSL密钥交换是RSA密钥交换（P198），最大问题是一旦私钥外泄，那么第三方就能从之前所有截获的被加密的密钥中解密出所有之前的key，使用key就可以破解所有之前监听得到的密文。A与B之前所有的保密通信都归零了。这就是所谓的不支持前向安全（forward secrecy），私钥参与了密钥交换，安全性也随之取决于私钥的安全。

鉴别：实体之间建立身份认证的过程，包括通信实体鉴别和通信内容鉴别。

鉴别易受**重放攻击**：攻击者发送一个目的主机已接收的包，来达到欺骗目的主机目的。

• 最坏情况下：冒充合法方；

• 其他情况下：扰乱正常操作。

**对付重放攻击的现时（Nonce）**

鉴别消息中增加一项：现时（仅使用一次）：

1）**随机数：不可猜测，性质最好，但不适合无连接应用。SSL握手中交换的随机数用于生成主密钥！**

2）时间戳：需要时钟同步，协议必须能容错。

3）序列号：每一方都要记住其它各方与其通信时的最后一个序列号，难以实现；还要求系统抗毁。