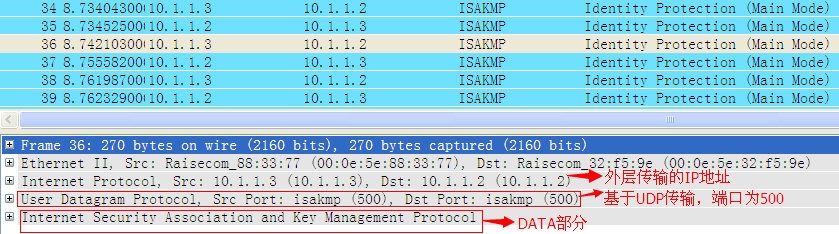
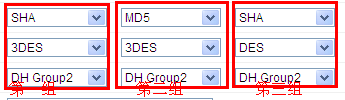
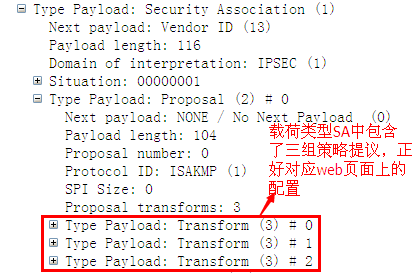
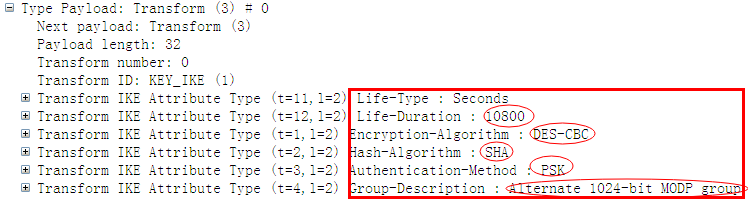
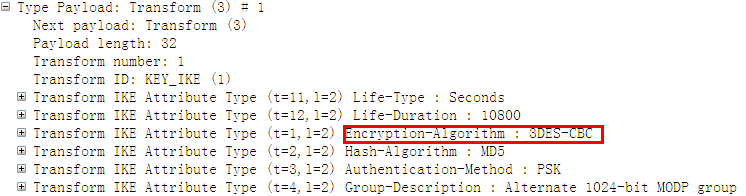
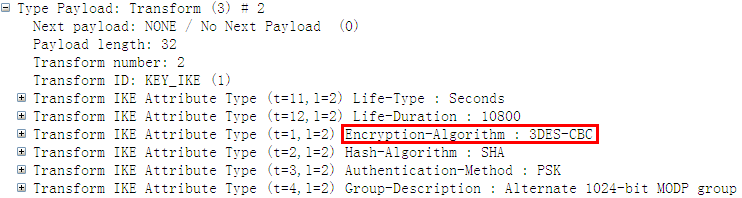
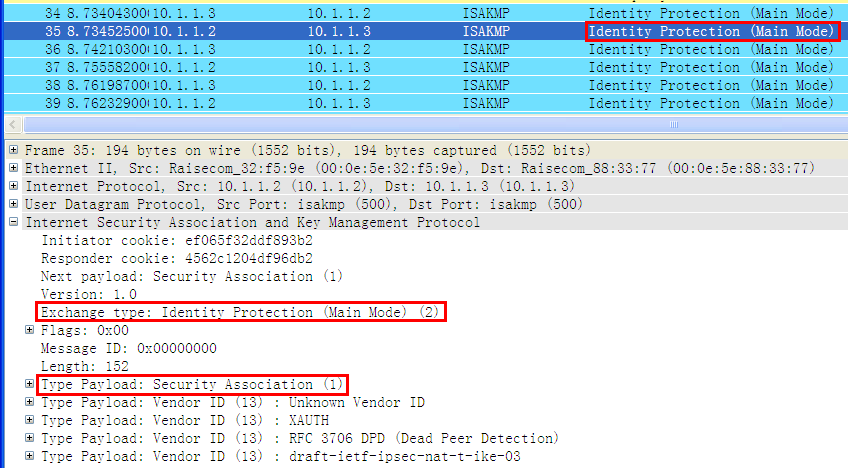
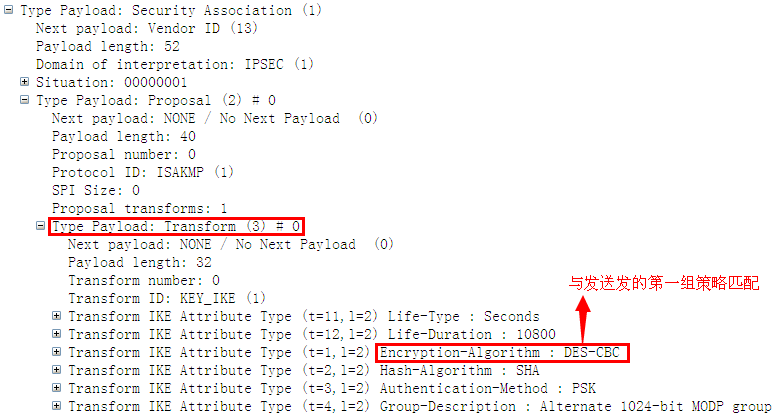
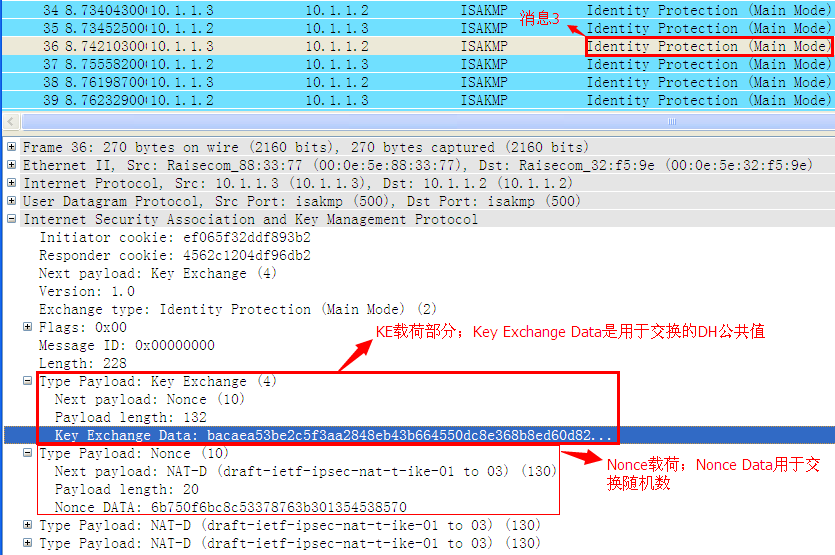
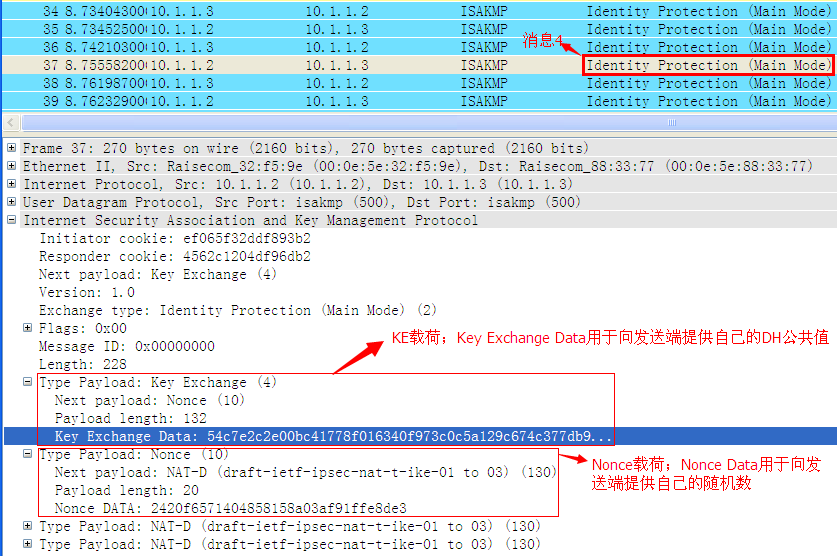
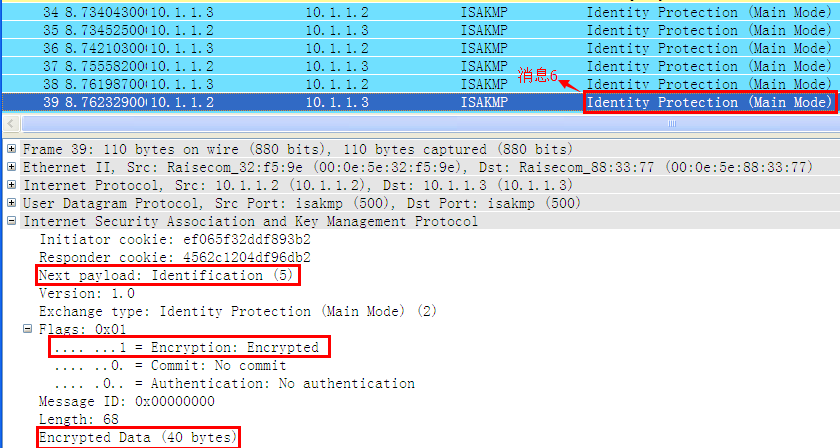
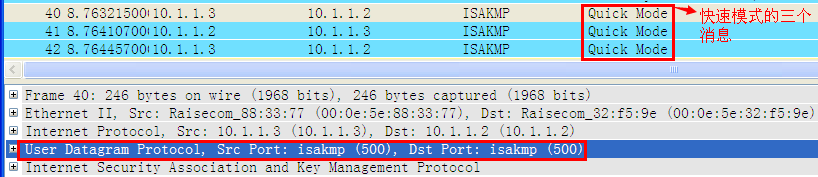
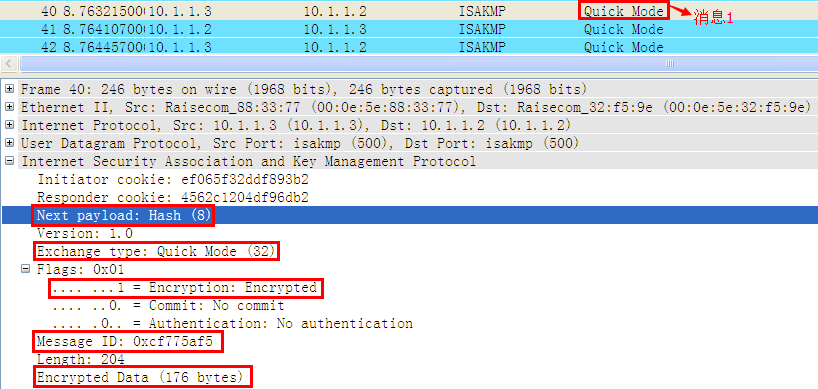
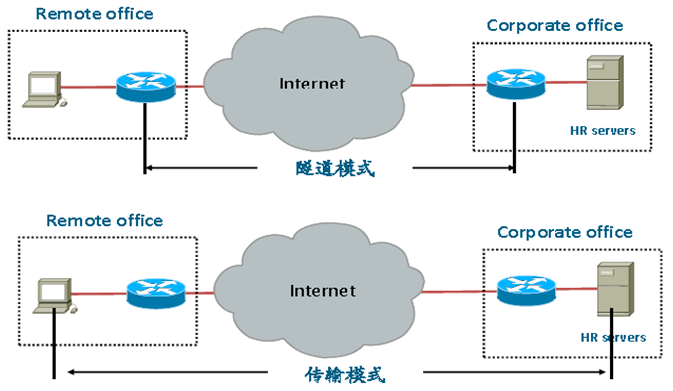
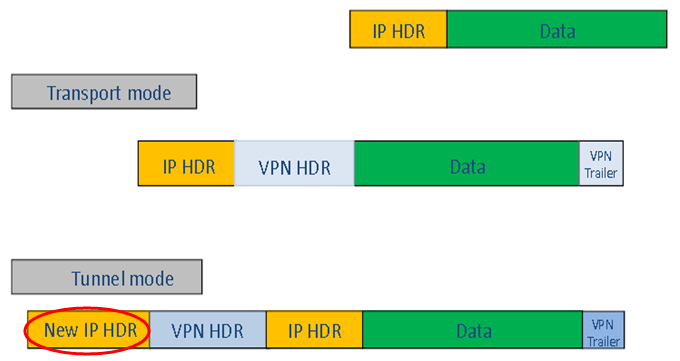
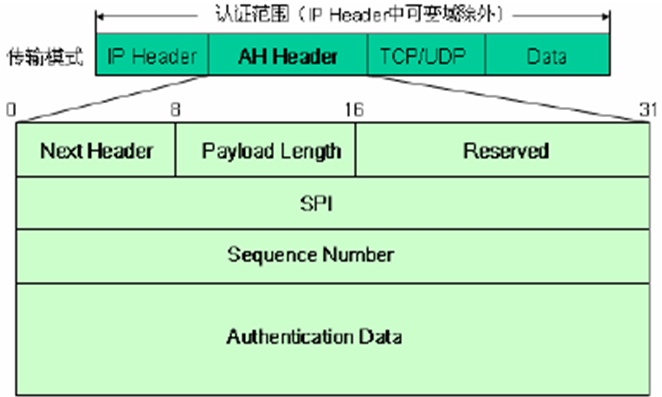
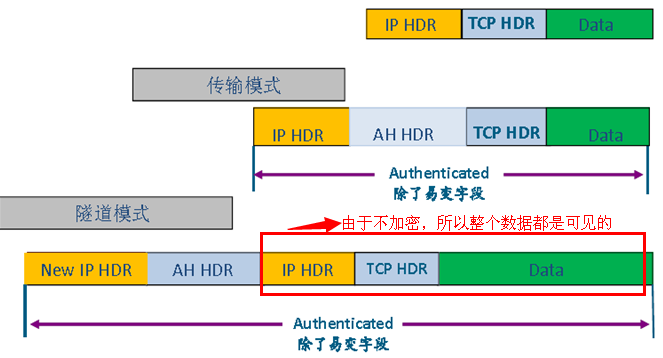
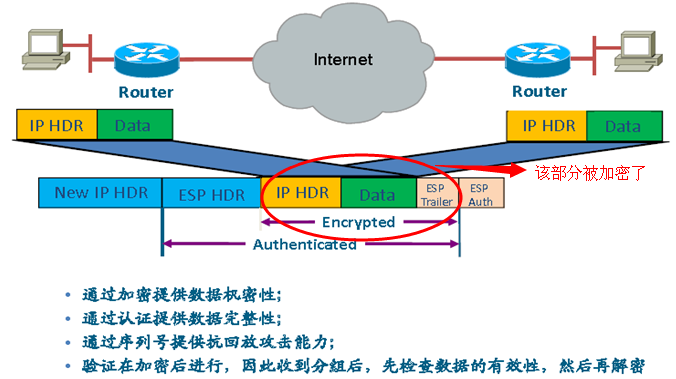
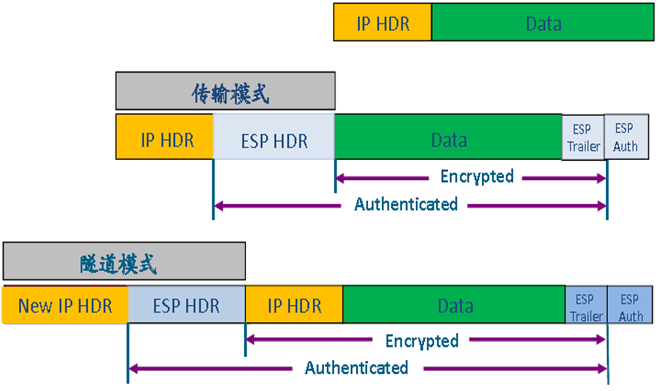
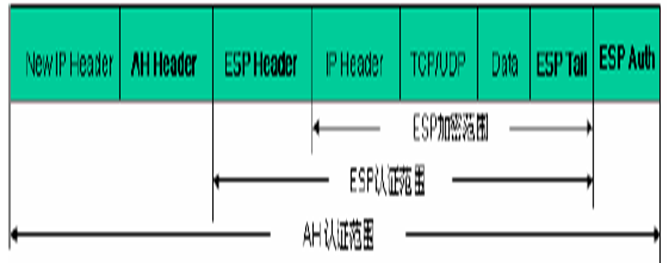
1. IPSec VPN隧道的建立过程分为两个阶段：  
   第一个阶段：分为两种模式主模式（Main Mode和野蛮模式（又称主动模式Aggressive）  
   第二个阶段：快速模式（Quick Mode）  
   区别：主模式与野蛮模式的区别：  
    （1）野蛮模式协商比主模式协商更快。  
    因为主模式需要交互6个消息，而野蛮模式只需要交互3个消息；  
    （2）主模式协商比野蛮模式协商更严谨、更安全。  
    因为主模式在“消息5&消息6”中对ID信息进行了加密。而野蛮模式由于受到交换次数的限制，ID消息在“消息1&消息2”中以明文的方式发送给对端。即主模式对对端身份进行了保护，而野蛮模式则没有。
2. 两个阶段分别完成任务：  
   （1）第一个阶段IKE设置，有三个任务需要完成：  
    （a）协商一系列算法和参数（这些算法和参数用于保护隧道建立过程中的数据）；  
    （b）必须计算出两边使用的加密KEY值，例如，两边使用3DES算法加密，3DES算法则需要一个密码，这个密码两端必须一样，但又不能在链路上传递。  
    （c）对等体的验证，如何才能知道对端就是我要与之通信的对端。这里验证有三种方法：预共享、数字签名和加密临时值。  
   上面一系列过程都是IKE（Internet 密钥交换协议，大多数厂商都把这个叫做VPNs Gateway）这个协议来实现。对于第一阶段需要注意以下几点：  
    （a1）只有remote vpn和easy vpn是积极模式的，其他都是用主模式来协商的；  
    （a2）让IKE对等体彼此验证对方并确定会话密钥，这个阶段用DH进行密钥交换，创建完IKE SA后，所有后续的协商都将通过加密和完整性检查来保护。  
    （a3）第一阶段帮助在对等体之间创建了一条安全通道，使后面的第二阶段过程协商受到安全保护。  
   （2）第二阶段：  
   协商IPSec SA使用的安全参数，创建IPSec SA（SA可以加密两个对等体之间的数据，这才是真正的需要加密的用户数据），使用AH或ESP来加密IP数据流。至此IPSec VPN隧道才真正建立起来。
3. 综上，有如下结论：  
   第一阶段作用：对等体之间彼此验证对方，并协商出IKE SA，保护第二阶段中IPSec SA协商过程；  
   第二阶段作用：协商IPSec单向SA，为保护IP数据流而创建；
4. 举例验证：以主模式，AH协议来简单分析一下IPSec VPN链接建立的过程（附带报文）：  
   第一个阶段三个任务，分别用6个消息来完成，每两个为一组，这些消息的具体格式取决于使用的对等体认证方法，使用预共享密钥进行验证的主模式（6条）协商过程使用ISAKMP消息格式来传递（基于UDP，端口号为500）。6条消息如下：  
     
     
   （1）准备工作：  
   在前2条消息发送之前，发送者和接受者必须先计算出各自的cookie（可以防重放和DOS攻击），这些cookie用于标识每个单独的协商交换消息。  
   cookie——RFC建议将源目的IP、源目的端口、本地生成的随机数、日期和时间进行散列操作。Cookie成为留在IKE协商中交换信息的唯一标识，实际上cookie是用来防止DOS攻击的，它把和其他设备建立IPSec所需要的连接信息不是以缓存的形式包存在路由器里，而是把这些信息HASH成个cookie值。  
   （2）1&2消息：  
   **消息1**：由发送方（协商发起端）发起，携带一些参数，发送方向接收方发送一条包含一组或多组策略提议（Raisecom工业路由器中是多组），在策略提议中包括5元组信息：  
    加密算法——DES；  
    散列算法——MD5-HMAC；  
    DH——Diffie-Hellman组-2；  
    认证方式——预共享；  
    IKE SA寿命。  
   如下是Raisecom中高级选项配置的策略：  
   （认证方式采用“预共享”方式）  
   （对于DPD，具体作用不知道，默认是关闭）  
   下面简要介绍一下上述五元组信息：  
    （a）协商模式：可以选择主模式（Main Mode）或者野蛮模式（Aggressive）。当选择主模式时，只能使用IP地址作为ID的类型。当用户端设备的IP地址为动态获取的情况时，需要选择野蛮模式。IKE野蛮模式相对于主模式来说更加灵活，可以选择根据协商发起端的IP地址或者ID来查找对应的身份验证字，并最终完成协商。  
    （b）验证方法AH（Authentication Header）：  
   身份验证确认通信双方的身份。目前在IKE提议中，仅可用pre-shared-key（预共享密钥）身份验证方法，使用该验证方法时必须配置身份验证字，并且两端的密钥要完全一致。  
    （c）加密算法：  
   包括DES和3DES加密算法；DES算法采用56bits的密钥进行加密，3DES算法采用168bits的密钥进行加密；AES128（Advanced Encryption Standard，即高级加密标准）采用Rijndael中的128bits的密钥进行加密；AES192（Advanced Encryption Standard，即高级加密标准）采用Rijndael中的192bits的密钥进行加密；AES256（Advanced Encryption Standard，即高级加密标准）采用Rijndael中的256bits的密钥进行加密；  
   一般来说，密钥越长的算法强度越高，受保护数据越难被破解，但消耗的计算资源会更多。  
    （d）Diffie-Hellman组标识（DH）：  
   用户可以选择Group 1即768bit 或 Group 2即1024bit。  
    （e）ISAKMP-SA生存周期：  
   IKE使用了两个阶段为IPSec进行密钥协商并建立安全联盟。第一阶段，通信各方彼此间建立了一个已通过身份验证和安全保护的通道，即ISAKMP安全联盟（ISAKMP SA）；第二阶段，用在第一阶段建立的安全通道为IPSec协商安全服务，即为IPSec协商具体的安全联盟，建立IPSec SA，IPSec SA用于最终的IP数据安全传送。ISAKMP-SA生存周期可以设定为60-604800之间的一个整数。  
    （f）定时发送keepalive报文（不是必须携带）：  
   IKE通过ISAKMP SA向对端定时发送KeepAlive报文维护该条ISAKMP SA的链路状态。当对端在配置的超时时间内未收到此KeepAlive报文时，如该ISAKMP SA带有timeout标记，则删除该ISAKMP SA及由其协商的IPSec SA；否则，将其标记为timeout。  
   如下是抓包获取到的信息（设备为Raisecom工业路由器）：  
   由上图可知，模式为主模式，载荷类型为SA。SA的数目和内容详见下图：  
   将载荷类型SA展开如下：  
   由下图可知，该SA中携带了三组策略，正好Raisecom中web页面配置的三组策略：  
     
     
   第一组Type Payload：Transform（3）# 0展开如下：  
   SA生存时间为10800；加密机制为DES；认证算法为SHA；认证方法选择PSK（预共享密钥）；DH为Group 2；  
   第二组Type Payload：Transform（3）# 1展开如下：  
   第三组  
   Type Payload：Transform（3）# 2展开如下：  
   报文中的组顺序和web页面上组顺序不一致，这个无所谓，只要能对上即可，因为实际中只要这三个组能匹配上即可。  
   **消息2**：由响应者（即对端设备）回应，内容基本一样，主要与发起者比较，是否与发起者的IKE策略匹配，不匹配则进行下一组比较，如果最终都找不到匹配，隧道就停止建立；  
   （note：发起者将其所有IKE策略发给接受者，接受者则在自己的策略中寻找与之匹配的策略；对比顺序从优先级号小的到大的；默认策略实际就是个模板没作用，如果认证只配置预共享的话，其他参数就会copy默认策略里的）  
   报文如下：  
    由上图可知，接受端回应的消息中，匹配了发送端的一条策略，如果有一条匹配，则不需要匹配其他策略。  
   在消息1和消息2中报错可能出现的原因：  
    （a）peer路由不通（即，外层的IP地址不通，这里对应的是发送发10.1.1.3和接收方10.1.1.2这两个地址不通，这里配置简单属于直连，而实际大型组网中，中间会有很多其他网元，往往是通过配置动态路由）；  
    （b）crypto iskmp key没有设置（即，没有配置预共享密钥）；  
    （c）一阶段的策略不匹配（这时需要检查两端设备的策略有不一致地方么）  
     
   （3）3&4消息：密钥交换过程  
   **消息3**：由发起者（即，隧道建立的发起者）发出，但是在发出消息3之前，有个过程必须要完成，就是Diffie-Hellman算法过程。  
   Diffie-Hellman算法过程目的：在消息1和消息2中所协商的算法，它们必须需要一个KEY（即，共享密钥中设置的密码），这个KEY在两个对等体上必须一样，但同时这个KEY不能在链路中传递，因为传递KEY是一个不安全的手段。所以，该过程的目的是分别在两个对等体间独立地生成一个DH公共值，该公共值有什么作用？因为两个对等体上都生成该DH公共值后，它们会在接下来的消息3和消息4中传送给对方，打个比方，A收到了B的DH公共值，B收到了A的DH公共值。当A、B都收到了对方的该公共值后，问题就好解决了。因为有一个公式在数学中被论证成立，那么现在借助公式，就可以在两个对等体上生成一个只有它们两个对等体知道的相同的KEY，该公式为：  
   发起者密钥=(Xb)amod p = (Xa)bmod p=响应者密钥  
   note：这个密钥不是最终算法中使用的KEY，但两个对等体通过该KEY材料来生成另外三个密钥，分别是：  
   SKEYID\_d——此密钥被用于计算后续IPSec密钥资源；  
   SKEYID\_a——此密钥被用于提供后续IKE消息的数据完整性以及认证；  
   SKEYID\_e——此密钥被用于对后续IKE消息进行加密；  
   所以，由发起者发起的第三条消息主要是向对等体发送自己的DH公共值和Nonce随机数；  
   实际报文如下：  
   由上述报文可知，发送方开始向接收方发送自己的DH公共值以及随机数；  
   对端收到后，可以根据“消息1&消息2”中协商的DH算法，以及发送端在消息3中给出的DH和nonce值来生成SKEYID\_d、SKEYID\_a、SKEYID\_e三个密钥；  
   **消息4**：同消息3，告知发送端自己的DH公共值和Nonce随机数；  
   报文如下：  
   由上述报文可知，接受方开始向发送方发送自己的DH公共值以及随机数；  
   对端收到后，可以根据“消息1&消息2”中协商的DH算法，以及接受端在消息4中给出的DH和nonce值来生成SKEYID\_d、SKEYID\_a、SKEYID\_e三个密钥；  
     
   （3）5&6消息：用于双方彼此验证。由“于消息1&消息2”的算法，以及“消息3&消息4”生成的三个KEY，所以在后续的“消息5&消息6”就能被加密传送，这个过程是受SKEYID\_e加密保护的。  
    预共享密钥的作用：为了正确生成密钥，每一个对等体必须找到与对方相对应的预共享密钥，当有许多对等体连接时，每一对对等体两端都需要配置预共享密钥，每一对等体都必须使用ISAKMP分组的源IP来查找与其对等体对应的预共享密钥（此时，由于ID还没到，彼此先用HASH来彼此验证对方）HASH认证成分——SKEYID\_a、cookieA、cookieB、preshare\_key、SA payload、转换集和策略。  
   **消息5**：由发起者向响应者发送，主要是为了验证对端自己就是自己想要与之通信的对端。这可以通过预共享、数字签名、加密临时值来实现。  
     
   **消息6**：由响应者向发起者发送，主要目的和第五条一样：  
     
   在消息5和消息6中报错可能出现的原因：  
   （1）crypto iskmp key设置错了；（即，两端的预共享密钥值设置的不一样）
5. 第二阶段：  
   第2阶段用三个消息来完成，目标是协商IPSec SA，而且只有一种模式，快速模式（Quick Mode），快速模式的协商是受IKE SA保护的。  
     
   对应设备上需要配置的参数（以R202i-VM为例）：  
     
   （1）1&2消息：发送IPSec SA的属性，协商IPSec SA  
   **消息1**：发起者会在第一条消息中发送IPSec SA的转换属性。其中包含：HASH、IPSec策略提议、Nonce可可选的DH以及身份ID。  
    （a）HASH：是用于给接受方作为完整性检验的，用于再次认证对等体（必须）HASH的成分和5-6阶段一样；  
    （b）IPSec策略提议：其中包括了安全协议（AH、ESP或AH-ESP）、SPI、散列算法、模式（隧道模式或传输模式）、IPSec SA生命周期（必选）；  
    （c）Nonce：用于防重放攻击，还被用作密码生成的材料，仅当启用PFS时用到；  
    （d）ID：描述IPSec SA是哪些地址、协议和端口建立的，即感兴趣流中的IP地址；  
     
    （e）PFS（利用DH交换，可选）：用了PFS后，就会在第二阶段重新DH出一个数据加密KEY，这个KEY和以前IKE协商出来的KEY没有任何关系，然后由这个新KEY来加密数据，只有到这个IPSec SA的生命周期后，会再次DH出新的KEY，这样，安全性就提高了（普通IPSec SA过期或密钥超时时，重新生成的数据加密密钥还是根据第一阶段DH出来的SKEYID\_d衍生出来的），PFS启用后，数据加密部分使用的密钥就没有了衍生的过程。  
    （f）DH：重新协商IPSec SA时使用的密钥（正常情况下，IPSec阶段使用的密钥都是由SKEYID\_d衍生而来的，密钥之间都有一定的关系，就算IPSec SA超时，新的KEY还是和SKEYID\_d有一定的关系）。  
   以上数据均被加密处理；  
     
   基于以上，第二阶段有几个概念需要理清：  
   **（a）封装模式：**包括传输模式（Transport）和隧道模式（Tunnel）。  
   传输模式：不使用新的IP头部，IP头部中的源/目的IP为通信的两个实点（当通信点等于加密点时，使用传输模式）；  
   **隧道模式：**需要封装一个新的IP头部，新的IP头部中源/目的IP为中间的VPN网关设备地址（当通信点不等于加密点时使用隧道模式）；  
   二者比较：  
   从安全性来讲，隧道模式优于传输模式，隧道模式可以完全地对原始IP数据报进行验证和加密以及可以使用IPSec对等体的IP地址来隐藏客户机的IP地址；  
   从性能来讲，隧道模式比传输模式占用更多带宽，一个额外的IP头；  
   因此，到底使用哪种模式需要按照实际的应用场景进行权衡。  
     
   **（b）安全联盟生存周期：**  
   所有在安全策略视图下没有单独配置生存周期的安全联盟，都采用全局生存周期。IKE（因特网密钥交换协议）为IPSec协商建立安全联盟（SA）时，采用本地设置的和对端提议的生存周期中较小的一个（即，当两端配置的生存周期不一致时，那么就用最小的那个值）。安全联盟生存周期的输入范围：30～604800；  
   所以，两端设备配置的生存周期不一致不会导致隧道无法建立。  
   **（c）采用的安全协议：**  
   安全提议中需要选择所采用的安全协议，用于为IP数据包提供安全。目前可选的安全协议有AH（验证报头）和ESP（封装安全有效负载），也可以指定同时使用AH和ESP（AH-ESP）。安全隧道两端所选择的安全协议必须一致。  
   所以，第二阶段协商不起来，两端协议是否一致是一个排查重点。  
   **AH协议**：类似于ICMP、TCP、UDP的IP协议，分配给它的协议号为51。提供如下安全功能：数据完整性服务、提供抗数据回放攻击、不提供数据加密性（不加密）。  
     
   （note：AH是不提供数据的加密的，所以在报文中可以看到完整的DATA部分）  
   AH报文头格式：  
     
   AH在两种模式下的封装：  
     
   **ESP协议**：协议号为50，提供如下功能：提供数据加密性（支持加密）、提供数据完整性、提供抗回放攻击能力；  
   ESP的数据验证和完整性服务只包括ESP的头和有效载荷（不包括外部的IP头部）  
   （note：ESP是提供加密的，所以抓取的ESP报文，是看不到原来被封装的数据部分）  
     
     
   ESP在两种模式下的封装：  
     
   **AH-ESP共用：**  
   隧道模式下：  
   （d）ESP协议加密算法：  
   ESP能够对IP报文内容进行加密保护，防止报文内容在传输过程中被窥探。加密算法的实现主要通过对称密钥系统，即使用相同的密钥对数据进行加密和解密。  
   一般来说IPSec使用两种加密算法：DES和3DES。  
   （e）ESP协议即AH协议的验证算法：  
   AH和ESP都能够对IP数据包的完整性进行验证，以判别报文在传输过程中是否被篡改。  
   一般来说IPSec使用两种验证算法：MD5和SHA-1  
   MD5：MD5输入任意长度的消息，产生128bit的消息摘要；  
   SHA-1：SHA-1输入长度小于2的64次方比特的消息，产生160bit的消息摘要。SHA-1的摘要长于MD5，因而是更安全的。  
   （f）使用NAT穿越：  
   在IPSec/IKE组建的VPN隧道中，若存在NAT安全网关设备，则必须配置IPSec/IKE的NAT穿越功能。  
   **消息2**：响应者向发起者发送第二条消息，同意第一条消息中的属性，同时，也能起到确认收到对端消息的作用。  
     
   在消息1和消息2中报错可能出现的原因：  
   （1）双方的模式不匹配（即，可能一端用传输模式，另一端用隧道模式）；  
   （2）感兴趣流不对称（如上述消息1中的（d））；  
     
   **消息3**：发送方发送第三条消息，其中包含一个HASH，其作用是确认接收方的消息以及证明发送方处于Active状态（表示发送方的第一条消息不是伪造的）  
     
   这一步一旦完成，隧道就建立起来了，用户的数据就能被放入隧道中传递。  
     
   本文参考资料：  
   <http://www.360doc.com/content/11/0517/14/706976_117422649.shtml>

<http://www.docin.com/p-549203149.html>