通过锐捷RG-NBR-6120E的SSL VPN访问内网Web资源有两种方式。一种是通过Web经身份认证登录后，点击管理员创建并授权给用户访问的Web资源链接进行访问，该种方式的原理基于Web反向代理技术；另一种是登录锐捷SSL VPN客户端后，在浏览器中直接键入内网网址进行访问，这种方式主要依赖于隧道技术。二者之间最显著的不同在于客户端机器上是否被安装了锐捷的虚拟网卡。

隧道型SSL VPN是基于虚拟网卡技术和SSL协议实现的，需要通信双方安装VPN软件，以便在计算机上创建虚拟网卡。隧道行SSL VPN进程运行在用户空间，通过虚拟网卡提供的接口，将应用程序发送到内网的网络层数据包，从操作系统内核空间转发给VPN进程，经过SSL建立的安全机制加密后，再通过物理网卡将数据发送出去。所以，隧道行SSL VPN的隧道数据封装过程是一个网络层二次封装的过程（通常是IP-to-IP），其提供的是一个工作于网络二、三层的VPN。

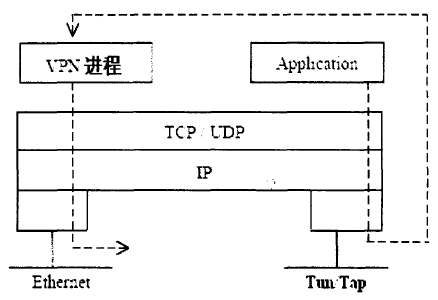
## 虚拟网卡技术简介

常用的虚拟网卡技术是TUN/TAP。TUN/TAP设备驱动采用一种巧妙的方式，对数据进行处理。在内核空间，TUN/TAP设备注册了两种设备接口，一种为字符设备接口，一种为网络设备接口。字符设备接口的主要作用是使得处于内核空间的数据包可以传送到处于用户空间的VPN进程中，便于VPN进程可以读取并加密数据包。

TUN/TAP网络设备提供了两种形式的驱动方式：YUN和TAP。实际上，在设备模块内部，两种设备封装好的应用数据没有什么差别。在网络协议栈中，它们的不同行为方式源于提交给上层协议栈的封装数据中，TAP设备指明它是一个类似点到点设备的数据包。这种标记导致的结果仅仅影响接下来进行的数据链路层处理。如果数据包需要转发，TAP设备的数据包支持数据链路层的特有功能，如使用网桥设备进行的冲突域划分等。而TUN设备仅支持IP层以上的功能。如果目的网络依靠IP层路由寻址，则YUN和TAP没有本质区别。

与真实的网络接口设备不同在于，TUN/TAP虚拟设备不是从物理媒介收发数据包，而是从应用程序接收发送数据包。在真实的网络接口设备中，从物理链路获取该网络接口数据包，然后产生一个中断，通知系统接收数据包，提交上层协议处理；当系统的上层网络模块，把一个数据包交付给该网路设备时，该网络设备首先为该数据包添加网络接口的协议头部，然后选择合适的时机，把数据包发送到物理链路上去，这通常是由硬件功能部件完成的。在TUN/TAP设备中，数据的收发来自用户空间，用户空间的应用程序通过TUN/TAP字符设备的接口，把数据包写入该设备，设备将识别该数据包，进行必要的检验，然后把该数据包转交到上层协议处理；而通过内核空间的网络模块传来的数据包，则排入TUN/TAP字符设备的接收队列中，等待用户空间的应用程序读取。

当应用程序通过VPN向外发送数据时，应用数据经过网络协议栈处理后到达虚拟网卡时，虚拟网卡上收到的是一个目的地址为内网IP地址的完整网络数据包。此时虚拟网卡通过中断策略通知VPN进程，而VPN进程通过字符设备驱动以类似读取终端设备的方法将虚拟网卡收到的网络数据包取出，进行加密处理后，以对方的外网IP地址为目的地址通过物理网卡发送出去，如下图所示。对方物理网卡收到数据包，则按照相反的流程处理，先交给VPN进程解密，再由VPN进程将目的地址为内网IP地址的网络数据包写入本方虚拟网卡的数据缓冲区，等待应用程序接收数据。



## TCP over TCP的隧道数据封装带来的问题及解决办法

隧道型SSL VPN在进行隧道数据封装时，是将内网的整个网络层数据包（通常是IP数据包）作为载荷，封装进外网的IP数据包中。这种封装在传输层有两种方式，既可以用TCP封装，也可以用UDP封装。

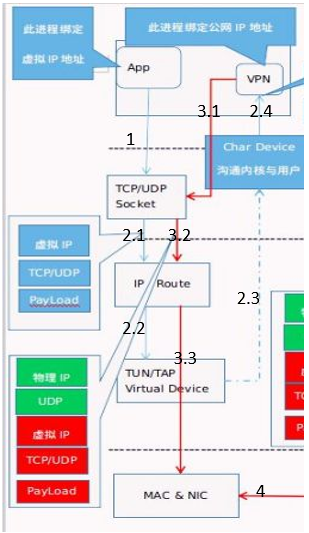
如果隧道型SSL VPN使用TCP协议对内网数据进行封装，而恰恰内网数据包也使用了TCP协议时，就给用户增加了对数据包进行重复控制检查的负担。这不仅不会给系统带来额外的可靠性，而且还会产生严重的数据延迟问题。其原因在于，TCP协议中通过设置一个计时器来决定数据包是否需要重传。如果把一个TCP协议封装在另一个TCP协议中，在一个不可靠网络中的数据包有可能在第一个计时器的时间内到达而没有在第二个计时器时间内到达外层的TCP协议，这样一个原本已经成功传输的数据包就不得不进行重传处理。在这种情况下，重传的数据包将可能会急剧增加，数据包传输的速度会急剧下降，网络的传输速度将会受到严重影响。

因此，隧道型SSL VPN在进行外层封装时，大多采用UDP协议。由于IP协议本来就是为不可靠的网络环境设计的，采用UDP协议恰好模拟了这种条件。使用UDP协议的另外一个好处是，VPN服务器与多个VPN客户端之间通信时，仅需要多路复用一个UDP端口，不仅可以为系统节省资源，且不需要由于对外开放更多端口而给系统带来的更多不安全因素。

SSL协议是建立在可靠的传输层之上的，因此SSL并不能直接工作在UDP协议上。这主要由以下两个原因造成：一是SSL握手协议必须按顺序进行；二是CBC模式的对称加密要求数据的连续性。这两点都是UDP协议无法保证的。为此，有人提供了一个TLS协议在UDP上的实现，也就是DTLS协议。隧道型SSL VPN采取了与DTLS协议类似的机制来保证SSL可以安全的工作在UDP协议之上。

隧道型SSL VPN采用与TCP类似的应答与重传机制来保证SSL握手协议的正常进行。对于握手过程中的每一个UDP数据包都要进行唯一编号，接收方必须对接收到的数据包做出应答。如果发送方一定时间内没有接收到已发出数据的应答消息，它将重发这个数据包。双方都要维护一个接收窗口以便对接收到的数据包进行排序。

### 客户端数据发送的包转发流程



当用户启动并成功登录SSL VPN客户端后，服务器首先会为客户端的虚拟网卡分配一个虚拟地址（即内网地址），并且给客户端的路由表推送关于内网IP的路由信息，使得一切目的地址为内网IP的数据包都经过虚拟网卡。

当用户访问内网网站时，http请求经过tcp和IP层完成初次封装后，在进行IP路由的过程中