**[Neutron 理解（10）：虚拟专用网（VPN）虚拟化 [How Neutron implements VPN Virtualization]](http://www.cnblogs.com/sammyliu/p/4677386.html)**

学习 Neutron 系列文章：

（1）[Neutron 所实现的虚拟化网络](http://www.cnblogs.com/sammyliu/p/4622563.html%20)

（2）[Neutron OpenvSwitch + VLAN 虚拟网络](http://www.cnblogs.com/sammyliu/p/4626419.html%20)

（3）[Neutron OpenvSwitch + GRE/VxLAN 虚拟网络](http://www.cnblogs.com/sammyliu/p/4627230.html%20)

（4）[Neutron OVS OpenFlow 流表 和 L2 Population](http://www.cnblogs.com/sammyliu/p/4633814.html%20)

（5）[Neutron DHCP Agent](http://www.cnblogs.com/sammyliu/p/4419195.html)

（6）[Neutron L3 Agent](http://www.cnblogs.com/sammyliu/p/4636091.html%20)

（7）[Neutron LBaas](http://www.cnblogs.com/sammyliu/p/4656176.html%20)

（8）[Neutron Security Group](http://www.cnblogs.com/sammyliu/p/4658746.html%20)

（9）[Neutron FWaas 和 Nova Security Group](http://www.cnblogs.com/sammyliu/p/4675991.html)

（10）[Neutron VPNaas](http://www.cnblogs.com/sammyliu/p/4677386.html%20)

（11）[Neutron DVR](http://www.cnblogs.com/sammyliu/p/4713562.html)

（12）[Neutron VRRP](http://www.cnblogs.com/sammyliu/p/4692081.html)

（13）[High Availability （HA）](http://www.cnblogs.com/sammyliu/p/4741967.html)

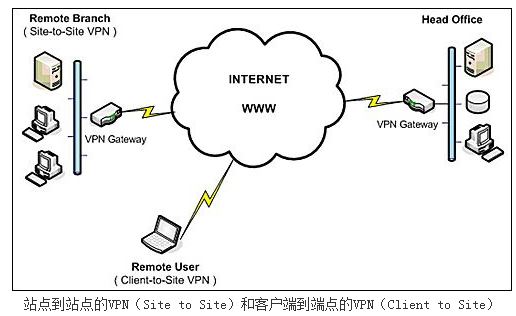
**1. 基础知识**

**1.1 虚拟专用网 （Virtual Private Network，VPN）**

**1.1.1 VPN 概念**

  每家公司都有自己的内部网络，但是这个网络是封闭的、有边界的。当一家公司拥有多个分布于不同地方的内部网络时，就需要一种方案将这些孤岛式的内部网络连接起来。一种解决方案是采用租用线路来将这些内部网络连接起来，但是这个方案成本高昂；另一种方案就是本文来谈到的 VPN 即虚拟专用网。我们可以把 VPN 理解成是虚拟出来的企业内部专线，它是一种建立在实际网络（或物理网络）基础上的功能性网络。对它的要求如下：

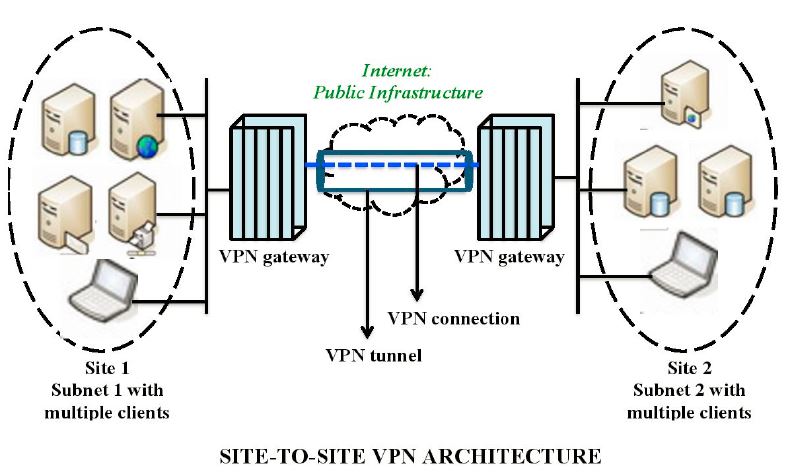
* 连通性：它通过实际网络（或物理网络）比如 Internet 将内部网络连在一起（site-to-site VPN），或者将 PC 和内部网络连在一起 （Client-to-site VPN）。
* 保密性：在 VPN 网络中，位于公共网络两端的网络在公共网络上传输信息时，其信息都是经过安全处理的，可以保证数据的完整性、真实性和私有性。
* 成本低：它利用低成本的公共网络做为企业骨干网，并不需要真正的去铺设光缆之类的物理线路。
* 透明性：对于用户来讲，VPN 的工作方式必须是完全透明的。



  目前，实现 VPN 的方法有多种。按照 VPN 所使用的隧道协议，VPN 主要可以分为三种，PPTP、L2TP 和 IPSec，其中PPTP和L2TP协议工作在OSI模型的第二层，又称为二层隧道协议；IPSec是第三层隧道协议。而 OpenStack 社区 Neutron 项目提供的是基于 IPSec 技术的一种参考实现。

**1.1.2 三种类型的 VPN**

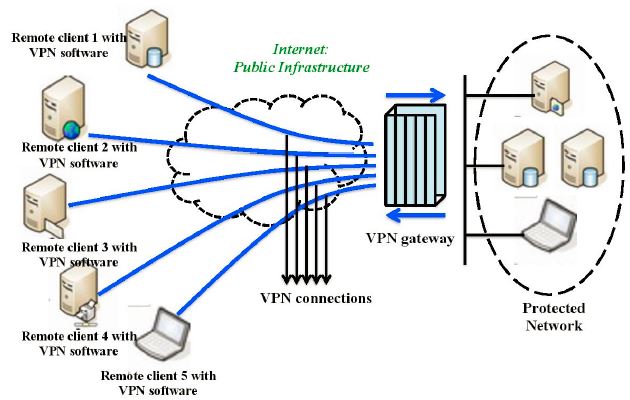
（1） Site-to-Site VPN (站点到站点VPN)



这种类型的VPN 常常用于公司的位于不同城市的分公司之间，或者私有云和公有云之间。它不需要在客户端或者主机上安装任何 VPN 客户端，而是由位于两个站点内的两个 VPN Gateway Gateway 负责 VPN 处理，发送端需要做封装，加密，身份验证，发送到对端 VPN Gateway等操作；接收端的 VPN Gateway 则负责解封，解密，将网络包发给内部网络中的目标主机等操作。

这种 VPN 可以提供更好的扩展性和灵活性，因为只需要安装两个 VPN Gateway，不需要安装客户端，因此，维护成本很低。但是，往往也会造成两个 VPN Gateway 的压力和损耗较高。

（2）Client-to-Site VPN （客户端到站点 VPN）

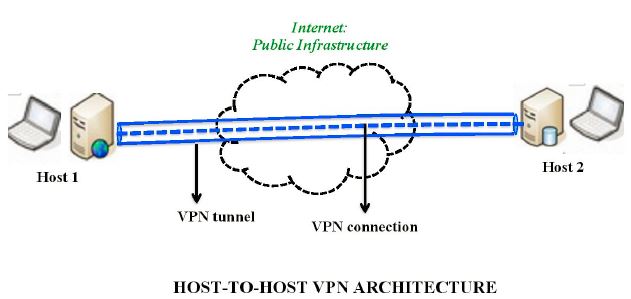


这种类型的 VPN 的一个典型例子是员工在公司外面办公时，他们需要在自己的电脑上安装一个 VPN 客户端，然后通过它连接到公司局域网。

因此，这种类型的 VPN 需要每个终端上都安装 VPN 客户端，它负责封装和加密数据，再把数据通过公网发到目标站点的 VPN Gateway。在站内侧的 VPN Gateway 的作用同 site-to-site 类型的 VPN 中的 Gateway。

这种类型的 VPN 使用也很方便，但是它要求安装和维护 VPN 客户端，以及对站点内的 VPN Gateway 也有兼容性要求。

（3）Host-to-Host VPN （主机到主机 VPN）



这种类型的 VPN 要求在两台主机上都安装 VPN 软件，通过它建立隧道和连接，从而传送数据。

**1.2 IPSec VPN**

**1.2.1 概念**

    IPSec（Internet Protocol Security）是 IETF 制定的为保证在 Internet 上传送数据的安全保密性能的三层隧道加密协议。IPSec 在 IP 层对 IP 报文提供安全服务。IPSec 协议本身定义了如何在 IP 数据包中增加字段来保证 IP 包的完整性、私有性和真实性，以及如何加密数据包。使用 IPsec，数据就可以安全地在公网上传输。简单地说，IPsec 提供了两个主机之间、两个安全网关之间或主机和安全网关之间的保护。

IPSec 协议：IPSec 包括报文验证头协议AH（协议号51）和报文安全封装协议ESP（协议号50）两个协议。AH 可提供数据源验证和数据完整性校验功能；ESP 除可提供数据验证和完整性校验功能外，还提供对IP报文的加密功能。

* AH 协议（IP协议号为51，使用较少）提供数据源认证、数据完整性校验和防报文重放功能，它能保护通信免受篡改，但不能防止窃听，适合用于传输非机密数据。AH的工作原理是在每一个数据包上添加一个身份验证报文头，此报文头插在标准IP包头后面，对数据提供完整性保护。可选择的认证算法有MD5（Message Digest）、SHA-1（Secure Hash Algorithm）等。MD5算法的计算速度比SHA-1算法快，而SHA-1算法的安全强度比MD5算法高。
* ESP 协议（IP协议号为50，使用较广）提供加密、数据源认证、数据完整性校验和防报文重放功能。ESP的工作原理是在每一个数据包的标准IP包头后面添加一个ESP报文头，并在数据包后面追加一个ESP尾。与AH协议不同的是，ESP将需要保护的用户数据进行加密后再封装到IP包中，以保证数据的机密性。常见的加密算法有DES、3DES、AES等。同时，作为可选项，用户可以选择MD5、SHA-1算法保证报文的完整性和真实性。这三个加密算法的安全性由高到低依次是：AES、3DES、DES，安全性高的加密算法实现机制复杂，运算速度慢。对于普通的安全要求，DES算法就可以满足需要。

IPSec 还支持 IKE 协议：

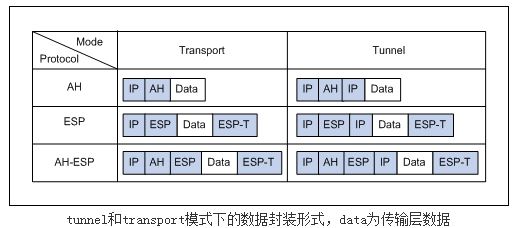
* IKE 协议（Internet Key Exchange ，RFC2409）：IKE 使两个地点能够建立安全的连接，方法是使用事先共享的密钥或由一家认证机构管理的公钥基础结构(PKI)数字证书，后者是一种进行公钥登记的内部或外购服务。通过使用签名数字ID来确认端点的身份，IKE能够将VPN的规模扩展到数以千计的端点。 为确保安全的数据加密，Cisco在路由器和PIX上对IPSec的实施过程中既支持数据加密标准(DES) ，也支持三重DES算法。详情请参见 [IKE 工作原理](http://www.kxzjzx.com/szzy/zykc201106/zykc201106_2/Courseware/content/SCOs/zykc201106_2_8_1_5/zykc201106_2_8_1_5.html)。

IPSec 和 IKE 的关系：

* IKE是 UDP 之上的一个应用层协议，是 IPSEC 的信令协议。
* IKE为 IPSEC 协商建立安全联盟，并把建立的参数及生成的密钥交给 IPSEC。 IPSEC 使用 IKE 建立的安全联盟对 IP 报文加密或验证处理。
* IPSEC 处理做为 IP 层的一部分，在 IP 层对报文进行处理。AH 协议和 ESP 协议有自己的协议号，分别是51和50。

IPSec 传输模式：隧道（tunnel）和传送（transport）两种工作方式。

* 在隧道方式中，用户的整个 IP 数据包被用来计算AH或ESP头，且被加密。AH 或 ESP 头和加密用户数据被封装在一个新的IP数据包中；
* 在传送方式中，只是传输层数据被用来计算AH或ESP头，AH或ESP头和被加密的传输层数据被放置在原IP包头后面。



而 IPSec VPN 即指采用 IPSec 协议来实现远程接入的一种 VPN 技术。

**1.2.2 IPSec 连接**

  IPsec 的两个端点被称为是 IPsec 对等体，要在两个对等体之间实现数据的安全传输就要在两者之间建立安全关联（Security Association，SA）。SA 是 IPsec 的基础，也是 IPsec 的本质。SA 是通信对等体间对某些要素的约定，例如，使用哪种协议（AH、ESP还是两者结合使用）、协议的封装模式（传输模式和隧道模式）、加密算法（DES、3DES和AES）、特定流中保护数据的共享密钥以及密钥的生存周期等。SA 是单向的，在两个对等体之间的双向通信，最少需要两个 SA 来分别对两个方向的数据流进行安全保护。同时，如果两个对等体希望同时使用AH和ESP来进行安全通信，则每个对等体都会针对每一种协议来构建一个独立的SA。SA是具有生存周期的，且只对通过IKE协商建立的SA有效，手工方式建立的SA永不老化。

  IPSec 两种传输模式的区别：

1. 传输模式在 AH、ESP 处理前后IP头部保持不变，主要用于 End-to-End 的应用场景。
2. 隧道模式则在 AH、ESP 处理之后再封装了一个外网 IP 头，主要用于 Site-to-Site 的应用场景。

**1.2.3 IPSec 的优点**

1. 支持IKE（Internet Key Exchange，因特网密钥交换），可实现密钥的自动协商功能，减少了密钥协商的开销。可以通过IKE建立和维护安全关联（Security Association，SA）的服务，简化了IPsec的使用和管理。
2. 所有使用IP协议进行数据传输的应用系统和服务都可以使用IPsec，由于IPSec工作在OSI的第3层，低于应用程序直接涉及的层级，所以对于应用程序来讲，利用IPSec VPN所建立起来的隧道是完全透明的，无需修改既有的应用程序，并且，现有应用程序的安全解决方法也不会受到任何影响。
3. 对数据的加密是以数据包为单位的，而不是以整个数据流为单位，这不仅灵活而且有助于进一步提高IP数据包的安全性，可以有效防范网络攻击。

以上内容都引用自互联网，主要来自：

* <http://security.pconline.com.cn/0910/1866954_all.html>
* <http://www.h3c.com.cn/Service/Channel_Service/Operational_Service/ICG_Technology/201005/675214_30005_0.htm>

**2. Neutron IPSec VPNaas**

OpenStack 社区 Neutron 项目的 VPNaaS 是一种 site-to-site 类型的 VPN 解决方案。它向用户提供 RESETS API， CLI 和 Horizon GUI 去操作 IPSec 所使用的各种协议和对象。



**2.1 受 VPNaas 管理 IPSec 对象**

**2.1.1 IKE Policy**

VPNaas 提供 vpn-ikepolicy-create CLI 来创建 IKE Policy。创建时可以指定各种参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 解释 | VPNaas 支持值 | 默认值 | 说明 |
| auth-algorithm | Authentication algorithm. 验证算法 | {sha1} | sha1 | IKE 认证算法。 |
| encryption-algorithm | Encryption algorithm 加密算法 | {aes-128,aes-256,aes-192} | aes-128 | IKE 加密算法。 |
| phase1-negotiation-mode | IKE Phase1 negotiation mode （Phase 1 交互模式） |  | main | 在IKE 第一阶段，VPN 网关或者客户端互相认证，通过确认对方的网关使用的是一个被认可的 Pre-Shared Key， certificates 或者 public key encryption。Pre-Shared Key 是用得最多的。该模式包括 Main 和aggressive。在 IKE 第二阶段，双方确定 IPSec 连接的参数。 |
| ike-version | IKE version （版本） | {v1,v2} | v1 | IKE版本，包括 V1 和 V2. |
| pfs | Perfect Forward Secrecy | {group2,group5,group14} | group5 | Diffie-Hellman 键的计算方法。详细可参考 [这篇文章](https://sc1.checkpoint.com/documents/R77/CP_R77_VPN_AdminGuide/13847.htm)。 |
| lifetime units=UNITS,value=VALUE | IKE lifetime attributes. 'units'-seconds。'value'-non  negative integer |  | units 默认值 为seconds  value 默认值为 3600 | IKE negociation 的时间间隔，一般来说，越短越安全，但是成本也越高。详细可参考 [这篇文章](https://sc1.checkpoint.com/documents/R77/CP_R77_VPN_AdminGuide/13847.htm)。 |

**2.1.2 IPSec Policy**

VPNaas 提供 REST API 和 CL 来操作 IPSec Policy：

* vpn-ipsecpolicy-create Create an IPsec policy.
* vpn-ipsecpolicy-delete Delete a given IPsec policy.
* vpn-ipsecpolicy-list List ipsecpolicies that belongs to a given tenant connection.
* vpn-ipsecpolicy-show Show information of a given IPsec policy.
* vpn-ipsecpolicy-update Update a given IPsec policy.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 解释 | VPNaas 支持的值 | 默认值 | 说明 |
| transform-protocol | Transform protocol in lowercase | {esp,ah,ah-esp} | esp | IPSec 协议，见上面的基础知识部分。 |
| auth-algorithm | Authentication algorithm in lowercase | {sha1} | sha1 | AH 认证头摘要或者 ESP 摘要的计算方法，包括 sha1 和 md5。见上面的基础知识部分。 |
| encryption-algorithm | Encryption algorithm in lowercase, | {3des，aes-128，aes-256，aes-192} | aes-128 | ESP 对传输的数据的加密算法。见上面的基础知识部分。 |
| encapsulation-mode | Encapsulation mode | {tunnel,transport} | tunnel | IPSec 传输模式。见上面的基础知识部分。 |
| pfs | Perfect Forward Secrecy | {group2,group5,group14} | group5 | 详细可参考 [这篇文章](https://sc1.checkpoint.com/documents/R77/CP_R77_VPN_AdminGuide/13847.htm)。 |
| lifetime units=UNITS,value=VALUE | IPsec lifetime attributes. 'units'-seconds， | 'value'-non negative integer | units 默认值 seconds  value 默认值 3600 | 一个IPSec 连接的有效时间长度。可以设置 时间长度(seconds) 和传输的数据量 (kilobytes)。超过该 lifetime 后，IPSec 重走 加密和认证过程。 |

**2.1.3 VPN Service**

VPN Service 的操作包括：

* vpn-service-create Create a VPN service.
* vpn-service-delete Delete a given VPN service.
* vpn-service-list List VPN service configurations that belong to a given tenant.
* vpn-service-show Show information of a given VPN service.
* vpn-service-update Update a given VPN service.

|  |  |  |
| --- | --- | --- |
| 参数 | 解释 | 说明 |
| ROUTER | Router unique identifier for the VPN service | VPN 隧道在本地的终端 Router |
| SUBNET | Subnet unique identifier for the VPN service deployment. | VPN 服务所面向的子网。该子网内的虚机可以使用 VPN 服务。 |

一般来说，VPN 是向两个子网（site-to-site）提供的，因此 VPN 的两头是 Router 或者 Gateway。 VPNaas 使用 Router 作为一个端点，还需要使用一个 subnet 指定可以使用该 VPN Connection 的目标虚机网段。

（[图片来源](https://sc1.checkpoint.com/documents/R77/CP_R77_VPN_AdminGuide/13847.htm)）

**2.1.4 IPSec Site Connection**

IPSec Site Connection 操作包括：

* ipsec-site-connection-create Create an IPsec site connection.
* ipsec-site-connection-delete Delete a given IPsec site connection.
* ipsec-site-connection-list List IPsec site connections that belong to a given tenant.
* ipsec-site-connection-show Show information of a given IPsec site connection.
* ipsec-site-connection-update Update a given IPsec site connection.

|  |  |  |
| --- | --- | --- |
| 参数 | 解释 | 说明 |
| vpnservice-id | VPN service instance ID associated with this connection | ID |
| ikepolicy-id | IKE policy ID associated with this connection. | 指定待使用的 IKE policy |
| ipsecpolicy-id | IPsec policy ID associated with this connection | 指定待使用的 IPSec Policy |
| peer-address | Peer gateway public IPv4/IPv6 address or FQDN | 指定对方 Router的 公网IP 地址 |
| peer-id | Peer router identity for authentication. Can be IPv4/IPv6 address, e-mail address, key id, or FQDN | 用于验证的对方 router 的 ID，可以同 peer-address。 |
| peer-cidr | Remote subnet(s) in CIDR format. | 对端的 subnet |
| psk | Pre-shared key string | 用于IKE 第一阶段双方的认证 |

一个例子：

host 1 ------ NEURON SERVER （-- INTERNET GATEWAY --） <--- IPSEC ---> （INTERNET GATEWAY） --Neutron Server---- host 2

| | | | | |

| | | | | |

192.168.64.0/24 10.10.20.2/1.2.3.10 1.2.3.9 4.3.2.9 4.3.2.10/10.10.20.10 192.168.63.0/24  
  
左边的环境中，VPN Service 和 IPSec site connection 的配置为：

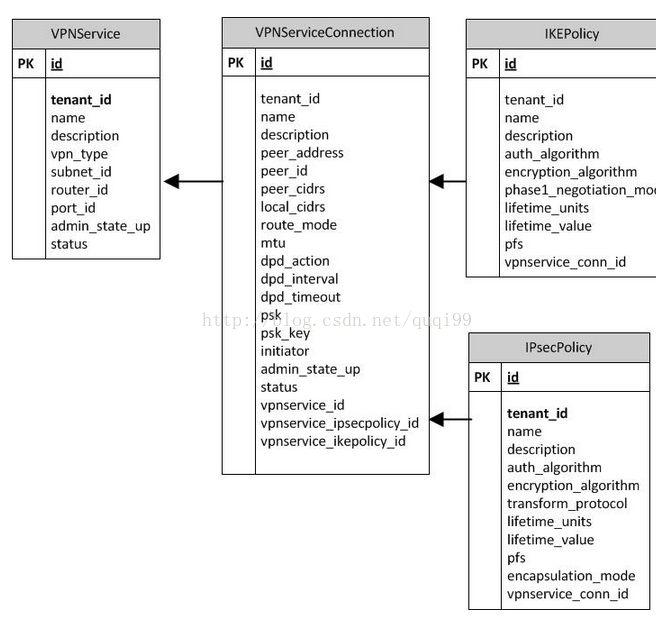
* ROUTER = router 1 （运行在 10.10.20.2 上的名为 “router1” 的 Router 的 UUID 或者 name。对于 IPSec tunnel 来说，本环境中的端点地址为 router 的 external port 的 IP 即 1.2.3.10）
* SUBNET = subnet1 （192.168.64.0/24 的 subnet UUID 后者 name）
* peer-cidr = 192.168.63.0/24
* peer-id = 4.3.2.10
* peer-address = 4.3.2.10

右边的环境中，VPN Service 和 IPSec site connection 的配置为：

* ROUTER = router 1 （运行在 10.10.20.10 上的名为 “router1” 的 Router 的 UUID 或者 name。对于 IPSec tunnel 来说，本环境中的端点地址为 router 的 external port 的 IP 即 4.3.2.10）
* SUBNET = subnet1 （192.168.63.0/24 的 subnet UUID 后者 name）
* peer-cidr = 192.168.64.0/24
* peer-id = 1.2.3.10
* peer-address = 1.2.3.10

两边的 IPSec Site Connection 需要使用同样的 IKE Policy 和 IPSec Policy。这样，两个 Router 之间的IPSec 隧道就可以建立起来了，192.168.63.0/24 和 192.168.64.0/24 两个网段内的虚机就可以直接通信了。

**2.1.5 这些对象之间的关系**

（[图片来源](http://blog.csdn.net/quqi99/article/details/9734251)）

**2.2 IPSec VPNaas：基于 OpenSwan （IPSec在Linux下的实现）的实现**

**2.2.1 OpenSwan**

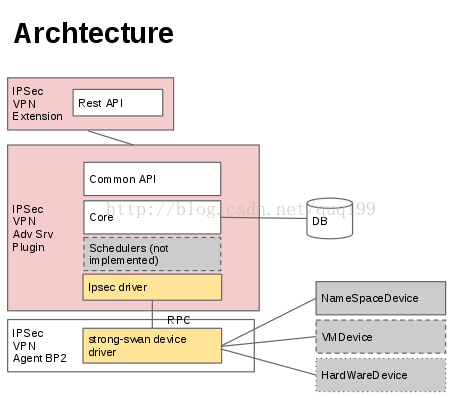
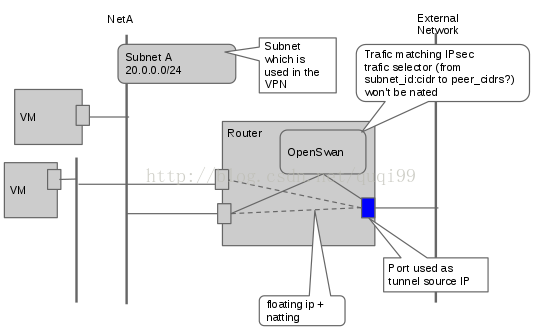
    OpenSwan 是 IPSec 在 Linux 上的开源实现。因此，要使用 IPSec VPNaas，你需要在 L3 Agent 节点上安装它。它的配置保存在 /etc/ipsec.conf 中。[这篇文章](http://mos1989.blog.51cto.com/4226977/1109450" \t "_blank) 详细阐述了使用 OpenSwan 构造 VPN 的方法。

**2.2.2 IPSec VPNaas 的实现**

    同 LBaas 的实现方式一样，VPNaas 主要是实现了一个让用户通过 REST API 操作 OpenSwan 的框架：

* Extention 提供 REST API
* Plugin 实现了 DB 操作，以及通过 RPC 调用 VPNaas 的 OpenSwan driver。
* Driver 层实现了具体的功能

    通过这些代码，Neutron 就能够通过配置 OpenSwan 为在指定 Router 上的指定子网提供 VPN 服务。

        （[图片来源](http://blog.csdn.net/quqi99/article/details/9734251)）

Driver 的实现代码主要在 neutron/services/vpn/device\_drivers/ipsec.py 中。其start方法调用 ipsec 命令来配置 IPSec，包括：

（1）初始化，其中包括在 router 的 iptables nat 表中添加一条规则，来使得目的为 ipsec 隧道的网络包不会做 DNAT：

-A POSTROUTING -s <local\_cidr> -d <peer\_cidr> -m policy --dir out --pol ipsec -j ACCEPT

（2）Neutron Server 在 VPN Server 有变化时，通过 RPC 通知该 VPN Service 的 Router 的 L3 Agent 上的 VPNaas Agent。当有新的 VPN Server 被创建时，它执行以下操作：

（2.1）通过 RPC 获取新建的 VPN Service，并通过 \_make\_vpnservice\_dict 函数将其配置转化为 IPSec 配置文件中的配置项，并初始化待启动的 IPSec 进程的配置。

（2.2）启动一个  pluto IKE keying daemon

self.\_execute([self.binary, 'pluto', '--ctlbase', self.pid\_path, '--ipsecdir', self.etc\_dir, '--use-netkey', '--uniqueids', '--nat\_traversal','--secretsfile', self.secrets\_file, '--virtual\_private', virtual\_private ])

（2.3）添加已配置的 IPSec 连接

for ipsec\_site\_conn in self.vpnservice['ipsec\_site\_connections']:

nexthop = self.\_get\_nexthop(ipsec\_site\_conn['peer\_address'])

self.\_execute([self.binary, 'addconn', '--ctlbase', '%s.ctl' % self.pid\_path, '--defaultroutenexthop', nexthop, '--config', self.config\_file,

ipsec\_site\_conn['id']])

（2.4）初始化 IPSec 连接

for ipsec\_site\_conn in self.vpnservice['ipsec\_site\_connections']:

if not ipsec\_site\_conn['initiator'] == 'start':

continue

#initiate ipsec connection

self.\_execute([self.binary, 'whack', '--ctlbase', self.pid\_path,'--name', ipsec\_site\_conn['id'],'--asynchronous', '--initiate'])

而这些命令的指定都是在用于 VPN Service 的那个 router 的 namespace 中执行的：

def \_execute(self, cmd, check\_exit\_code=True):

"""Execute command on namespace."""

ip\_wrapper = ip\_lib.IPWrapper(self.root\_helper, self.namespace) #self.namespace 是 Router 的 namespace

return ip\_wrapper.netns.execute(cmd, check\_exit\_code=check\_exit\_code)

**3. 其它 Neutron VPNaas 实现**

[这篇文章](http://www.wowrk.com/vpn-tunneling-protocol/) 阐述了各种不同的  VPN 实现技术。目前，Neutron 项目中有几个不同的 blueprint 在实现基于不同技术的 VPNaas。

**3.1 MPLS VPNaas**

    这个 [blueprint](https://wiki.openstack.org/wiki/Neutron/MPLSVPNaaS)实现了 MPLS VPN。其出发点主要是  QoS 支持，以及目前 MPLS 已经在骨干网络中已经很普及了。根据 [百度百科](http://baike.baidu.com/view/50920.htm)，MPLS VPN 是一种基于MPLS技术的IP-VPN，是在网络路由和交换设备上应用MPLS技术，简化核心路由器的路由选择方式，利用结合传统路由技术的标记交换实现的IP虚拟专用网络（IP VPN），可用来构造宽带的Intranet、Extranet，满足多种灵活的业务需求。

**3.2 PPTP VPNaas**

    这个 [blueprint](https://blueprints.launchpad.net/neutron/+spec/pptp-vpn-support)致力于实现 PPTP VPNaas，这是一种 client-to-site 类型的 VPN 解决方案。根据 [百度百科](http://baike.baidu.com/view/32771.htm)，点对点隧道协议 (PPTP) 是由包括微软和3Com等公司组成的PPTP论坛开发的一种点对点隧道协，基于拨号使用的PPP协议使用PAP或CHAP之类的加密算法，或者使用 Microsoft的点对点加密算法MPPE。其通过跨越基于 TCP/IP 的数据网络创建 VPN 实现了从远程客户端到专用企业服务器之间数据的安全传输。PPTP 支持通过公共网络(例如 Internet)建立按需的、多协议的、虚拟专用网络。PPTP 允许加密 IP 通讯，然后在要跨越公司 IP 网络或公共 IP 网络(如 Internet)发送的 IP 头中对其进行封装。

（引用自 [百度百科](http://baike.baidu.com/view/32771.htm)）

到目前为止，这个 blueprint 都还没开始做。

**3.3 其它 VPNaas**

* BGP VPNaas：<https://blueprints.launchpad.net/neutron/+spec/neutron-bgp-vpn>
* VPN 相关的 blueprint：<https://blueprints.launchpad.net/neutron?searchtext=VPN>

其它参考文档：

* <http://www.amaranten.com/support/user%20guide/VPN/IPSec_Basics/Overview.htm>
* <https://www.diva-portal.org/smash/get/diva2:861531/FULLTEXT02.pdf>

分类: [Neutron](http://www.cnblogs.com/sammyliu/category/677824.html)