007-了解Vxlan

1. VXLAN知识

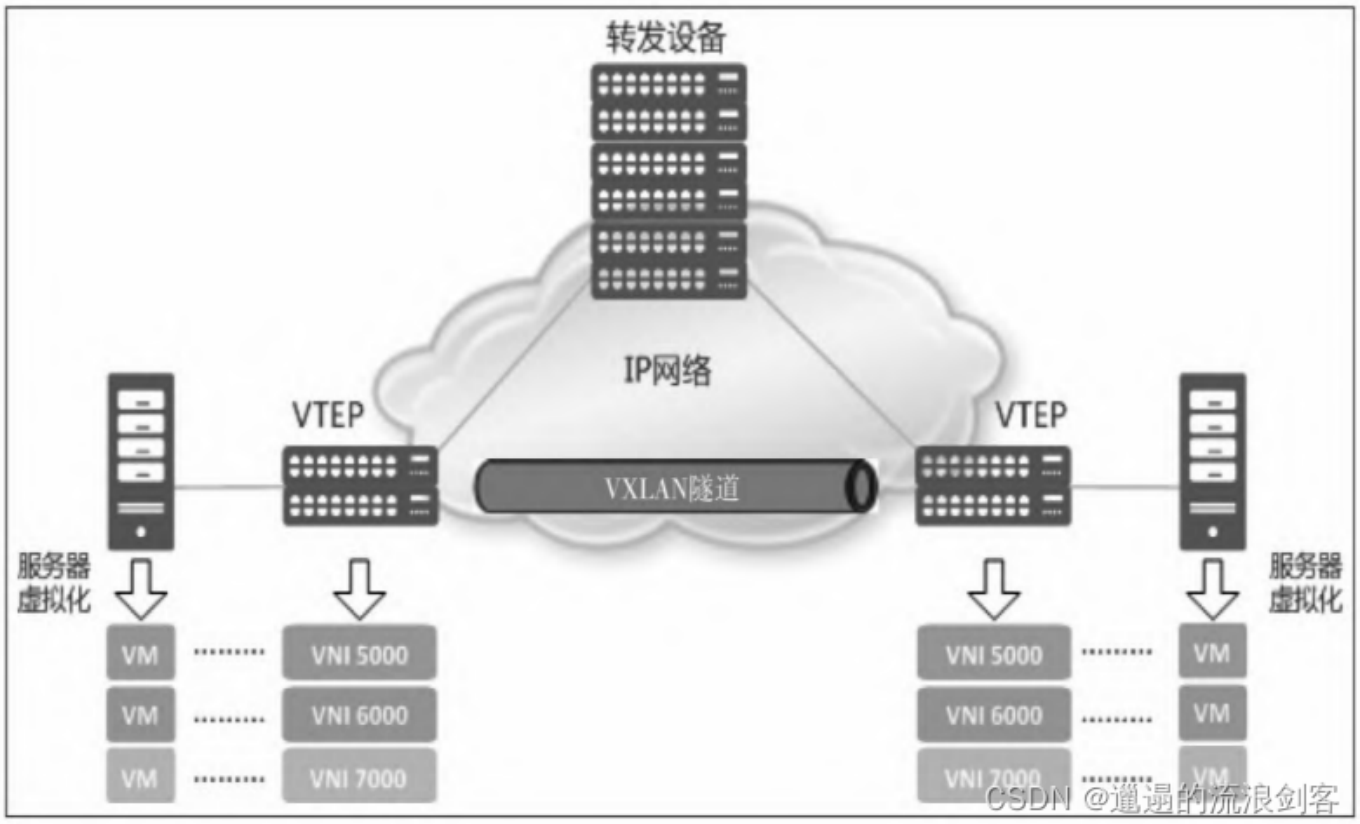
1.1 VXLAN 协议原理

1.2 VXLAN点对点通信

1. VXLAN知识

1.1 VXLAN 协议原理

VXLAN（虚拟可扩展的局域网）是一种虚拟化隧道通信技术。它是一种overlay（覆盖网络）技术，通过三层的网络搭建虚拟的二层网络；VXLAN是在底层物理网络（underlay）之上使用隧道技术，依托UDP层构建的overlay的逻辑网络，使逻辑网络与物理网络解耦，实现灵活的组网需求。



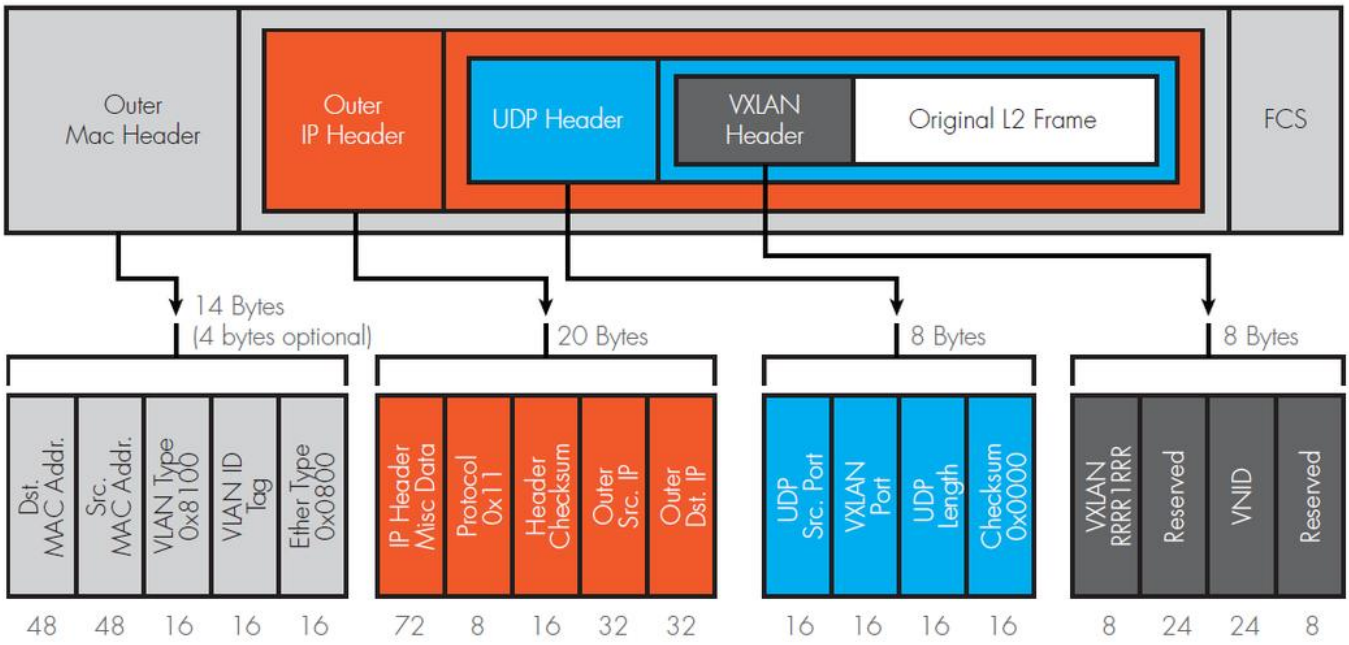
如上图所示为VXLAN的工作模型，它创建在原来的IP网络（三层）上，只要是三层可达（能够通过IP互相通信）的网络就能部署VXLAN

在VXLAN网络的每个端点都有一个VTEP设备，负责VXLAN协议报文的封包和解包，也就是在虚拟报文上封装VTEP通信的报文头部。物理网络上可以创建多个VXLAN网络，可以将这些VXLAN看作一个隧道，不同节点上的虚拟机/容器能够通过隧道直连。通过VNI标识不同的VXLAN网络，使得不同的VXLAN可以相互隔离

VXLAN的几个重要概念如下：

1. **VTEP（VXLAN Tunnel Endpoint ）VXLAN隧道端点，**负责VXLAN报文的封装与解封装。每个VTEP具备两个接口：一个是本地桥接接口，负责原始以太帧接收和发送，另一个是IP接口，负责VXLAN数据帧接收和发送。**VTEP可以是物理交换机或软件交换机。**
2. VNI（VXLAN Network Identifier）：VNI是每个VXLAN的标识，是个24位整数，因此最大值是2^{24}。 如果一个VNI对应一个租户，那么理论上VXLAN可以支撑千万级别的租户。
3. tunnel：隧道是一个逻辑上的概念，在VXLAN模型中并没有具体的物理实体相对应。隧道可以看作一种虚拟通道，VXLAN通信双方都认为自己在直接通信，并不知道底层网络的存在。从整体看，每个VXLAN网络像是为通信的虚拟机搭建了一个单独的通信通道，也就是隧道。

VXLAN其实是在三层网络上构建出来的一个二层网络的隧道。VNI相同的机器逻辑上处理同一个二层网络中。VXLAN封包格式如下图：



VXLAN的报文就是MAC in UDP，即在三层网络的基础上构建一个虚拟的二层网络。VXLAN的封包格式显示原来的二层以太网帧（包含MAC头部、IP头部和传输层头部的报文），被放在VXLAN包头里进行封装，再套到标准的UDP头部（UDP头部、IP头部和MAC头部）用来在底层网络上传输报文。

一个完整的VXLAN报文需要哪些信息?

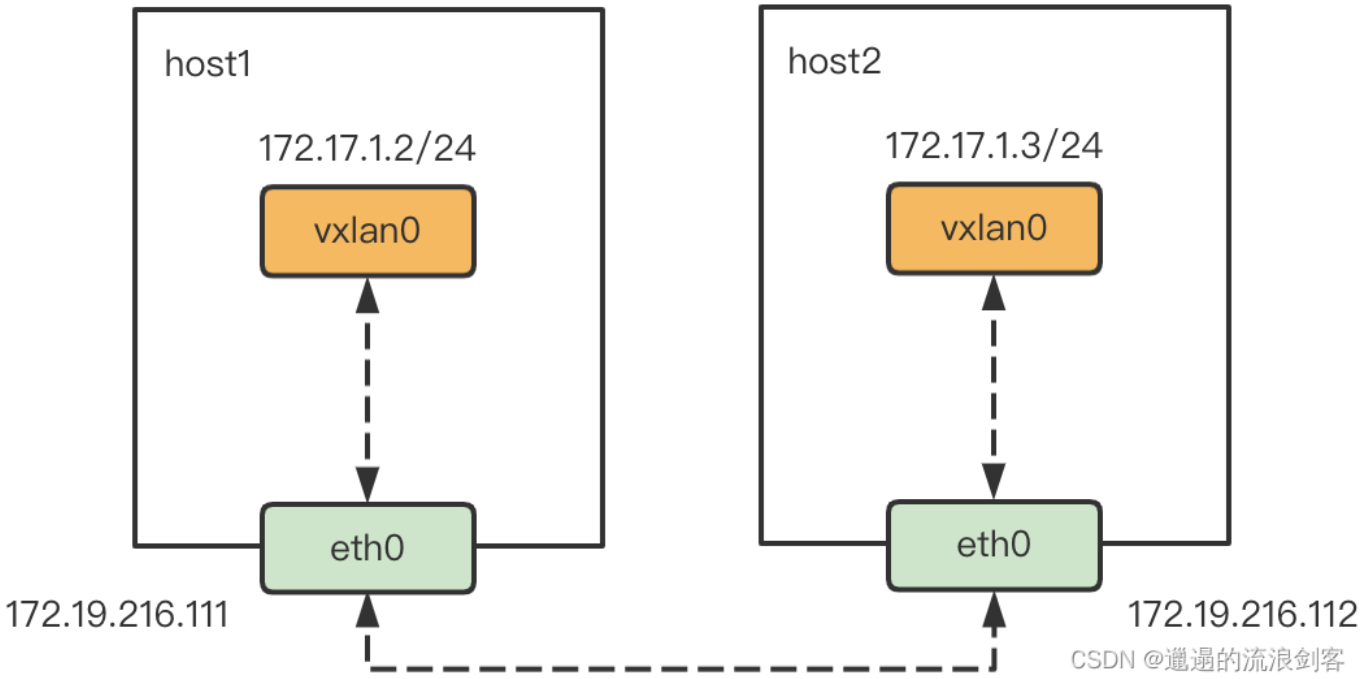
1. 内层报文：通信双方的IP地址已经确定，需要VXLAN填充的是对方的MAC地址，VXLAN需要一个机制实现ARP的功能

2. VXLAN头部：只需要知道VNI。它一般是直接配置在VTEP上的，即要么是提前规划的，要么是根据内部报文自动生成的
3. UDP头部：最重要的是源地址和目的地址的端口，源地址端口是由系统自动生成并管理的，目的端口一般固定位IANA分配的4789端口
4. IP头部：IP头部关系的是对端VTEP的IP地址，源地址可以用简单的方式确定，目的地址是虚拟机所在地址宿主机VTEP的IP地址，需要由某种方式来确定
5. MAC头部：确定了VTEP的IP地址，MAC地址可以通过ARP方式获取，毕竟VTEP在同一个三层网络内

总结一下，一个VXLAN报文需要确定两个地址信息：内层报文（对应目的虚拟机/容器）的MAC地址和外层报文（对应目的虚拟机/容器所在宿主机的VTEP）IP地址。

1.2 VXLAN点对点通信

点对点的VXLAN即两台机器构成一个VXLAN网络，每台机器上有一个VTEP，VTEP之间通过它们的IP地址进行通信。点对点VXLAN网络拓扑如下图：



使用ip link命令创建VXLAN接口：

Shell | 复制代码

```
[root@aliyun ~]# ip link add vxlan0 type vxlan id 42 dstport 4789 remote 17
2.19.216.112 local 172.19.216.111 dev eth0
```

上面这条命令创建了一个名为vxlan0，类型为vxlan的网络接口，一些重要的参数如下：

- id 42：指定VNI的值，有效值在1到2^{24}之间
- dstport：VTEP通信的端口，IANA分配的端口是4789
- remote 172.19.216.112：对端VTEP的地址
- local 172.19.216.111：当前节点VTEP要使用的IP地址，即当前节点隧道口的IP地址
- dev eth0：当前节点用于VTEP通信的网卡设备，用来获取VTEP IP地址

Shell | 复制代码

```
[root@aliyun ~]# ip -d link show dev vxlan0
3: vxlan0: <BROADCAST,MULTICAST> mtu 1450 qdisc noop state DOWN mode DEFAULT group default qlen 1000
    link/ether 26:1a:4d:9b:a3:ed brd ff:ff:ff:ff:ff:ff promiscuity 0 minmtu 68 maxmtu 65535
    vxlan id 42 remote 172.19.216.112 local 172.19.216.111 dev eth0 srcport 0 0 dstport 4789 ttl auto ageing 300 udpcsum noudp6zerocsumtx noudp6zerocsumrx addrgenmode eui64 numtxqueues 1 numrxqueues 1 gso_max_size 65536 gso_max_segs 65535
```

为刚创建的VXLAN网卡配置IP地址并启用它：

```shell
[root@aliyun ~]# ip addr add 172.17.1.2/24 dev vxlan0
[root@aliyun ~]# ip link set vxlan0 up

[root@aliyun ~]# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 172.19.216.111  netmask 255.255.240.0  broadcast 172.19.223.255
        inet6 fe80::216:3eff:fe29:7a94  prefixlen 64  scopeid 0x20<link>
        ether 00:16:3e:29:7a:94  txqueuelen 1000  (Ethernet)
        RX packets 3146  bytes 780076 (761.7 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 3495  bytes 549248 (536.3 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 2  bytes 140 (140.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 2  bytes 140 (140.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

vxlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1450
        inet 172.17.1.2  netmask 255.255.255.0  broadcast 0.0.0.0
        inet6 fe80::241a:4dff:fe9b:a3ed  prefixlen 64  scopeid 0x20<link>
        ether 26:1a:4d:9b:a3:ed  txqueuelen 1000  (Ethernet)
        RX packets 45  bytes 3254 (3.1 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 45  bytes 3258 (3.1 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

执行成功后会发现路由表项多了下面的内容，所有目的地址是172.17.1.0/24网段的包要通过vxlan0转发：

```shell
[root@aliyun ~]# ip route
172.17.1.0/24 dev vxlan0 proto kernel scope link src 172.17.1.2
```

同时，vxlan0的FDB表项中标的内容如下：

```shell
[root@aliyun ~]# bridge fdb
00:00:00:00:00:00 dev vxlan0 dst 172.19.216.112 via eth0 self permanent
```

这个表项的意思是，默认的VTEP对端地址为172.19.216.112。换句话说，原始报文经过vxlan0后会被内核添加上VXLAN头部，而外部UDP头的目的IP地址会被带上172.19.216.112

在另外一台机器上（172.19.216.112）进行配置：

```shell
# 创建VXLAN接口
[root@aliyun ~]# ip link add vxlan0 type vxlan id 42 dstport 4789 remote 172.19.216.111 local 172.19.216.112 dev eth0
# 为刚创建的VXLAN网卡配置IP地址并启用它
[root@aliyun ~]# ip addr add 172.17.1.3/24 dev vxlan0
[root@aliyun ~]# ip link set vxlan0 up
```

测试两个VTEP的连通性：

```shell
[root@aliyun ~]# ping 172.17.1.3
PING 172.17.1.3 (172.17.1.3) 56(84) bytes of data.
64 bytes from 172.17.1.3: icmp_seq=1 ttl=64 time=0.720 ms
64 bytes from 172.17.1.3: icmp_seq=2 ttl=64 time=0.346 ms
64 bytes from 172.17.1.3: icmp_seq=3 ttl=64 time=0.343 ms
```