交换机和Linux服务器网卡绑定模式（bond）对接经典配置

以太网接口 二层链路聚合 **陆世叶** 2018-02-06 发表

交换机与Linux服务器多网卡bond模式对接

交换机多端口和服务器对接时，需要确定是否需要配置聚合或者不配置聚合，并且配置聚合的时候还需要确认是静态聚合还是动态聚合，当然这和当前服务器网卡的bond模式有关。下面我们了解下Linux服务器的7种bond模式，说明如下：

第一种模式：mod=0，即：(balance-rr) Round-robin policy（平衡抡循环策略）

特点：传输数据包顺序是依次传输（即：第1个包走eth0，下一个包就走eth1….一直循环下去，直到最后一个传输完毕），此模式提供负载平衡和容错能力；但是我们知道如果一个连接或者会话的数据包从不同的接口发出的话，中途再经过不同的链路，在客户端很有可能会出现数据包无序到达的问题，而无序到达的数据包需要重新要求被发送，这样网络的吞吐量就会下降。这种模式需要接入交换机配置静态链路聚合配置。

V3平台交换机侧的静态典型配置

[H3C] link-aggregation group 1 mode manual
[H3C] interface ethernet2/1/1
[H3C-Ethernet2/1/1] port link-aggregation group 1
[H3C-Ethernet2/1/1] interface ethernet2/1/2
[H3C-Ethernet2/1/2] port link-aggregation group 1
[H3C-Ethernet2/1/2] interface ethernet2/1/3
[H3C-Ethernet2/1/3] port link-aggregation group 1

V5/V7交换机侧的静态典型配置

[DeviceA] interface Bridge-Aggregation 1　　//默认静态
[DeviceA-Bridge-Aggregation1] quit
[DeviceA] interface GigabitEthernet 4/0/1
[DeviceA-GigabitEthernet4/0/1] port link-aggregation group 1
[DeviceA] interface GigabitEthernet 4/0/2
[DeviceA-GigabitEthernet4/0/2] port link-aggregation group 1

第二种模式：mod=1，即：(active-backup) Active-backup policy（主-备份策略）

特点：只有一个设备处于活动状态，当一个宕掉另一个马上由备份转换为主设备。mac地址是外部可见得，从外面看来，bond的MAC地址是唯一的，以避免switch(交换机)发生混乱。此模式只提供了容错能力；由此可见此算法的优点是可以提供高网络连接的可用性，但是它的资源利用率较低，只有一个接口处于工作状态，在有 N 个网络接口的情况下，资源利用率为1/N。交换机侧无需任何配置，但是会存在MAC漂移的记录。

第三种模式：mod=2，即：(balance-xor) XOR policy（平衡策略）

特点：基于指定的传输HASH策略传输数据包。缺省的策略是：(源MAC地址 XOR 目标MAC地址) % slave数量。其他的传输策略可以通过xmit_hash_policy选项指定，此模式提供负载平衡和容错能力。交换机侧无需配置任何链路模式

第四种模式：mod=3，即：broadcast（广播策略）

特点：在每个slave接口上传输每个数据包，此模式提供了容错能力。交换机侧无需配置任何链路模式

第五种模式：mod=4，即：(802.3ad) IEEE 802.3ad Dynamic link aggregation（IEEE 802.3ad 动态链接聚合）

特点：创建一个聚合组，它们共享同样的速率和双工设定。根据802.3ad规范将多个slave工作在同一个激活的聚合体下。外出流量的slave选举是基于传输hash策略，该策略可以通过xmit_hash_policy选项从缺省的XOR策略改变到其他策略。需要注意的 是，并不是所有的传输策略都是802.3ad适应的，尤其考虑到在802.3ad标准43.2.4章节提及的包乱序问题。不同的实现可能会有不同的适应性。交换机侧需要动态链路聚合配置对接。

必要条件：

条件1：ethtool支持获取每个slave的速率和双工设定
条件2：switch(交换机)支持IEEE 802.3ad Dynamic link aggregation
条件3：大多数switch(交换机)需要经过特定配置才能支持802.3ad模式

V3交换机的动态聚合典型配置

[H3C] link-aggregation group 1 mode static
[H3C] interface ethernet2/1/1
[H3C-Ethernet2/1/1] port link-aggregation group 1
[H3C-Ethernet2/1/1] interface ethernet2/1/2
[H3C-Ethernet2/1/2] port link-aggregation group 1

[H3C-Ethernet2/1/2] interface ethernet2/1/3
[H3C-Ethernet2/1/3] port link-aggregation group 1
V5/V7平台交换机的动态聚合典型配置
[DeviceA] interface Bridge-Aggregation 1
[DeviceA-Bridge-Aggregation1] link-aggregation mode dynamic
[DeviceA] interface GigabitEthernet 4/0/1
[DeviceA-GigabitEthernet4/0/1] port link-aggregation group 1
[DeviceA-GigabitEthernet4/0/1] quit
[DeviceA] interface GigabitEthernet 4/0/2
[DeviceA-GigabitEthernet4/0/2] port link-aggregation group 1

第六种模式：mod=5，即：(balance-tlb) Adaptive transmit load balancing（适配器传输负载均衡）
特点：不需要任何特别的switch(交换机)支持的通道bonding。在每个slave上根据当前的负载（根据速度计算）分配外出流量。如果正在接受数据的slave出故障了，另一个slave接管失败的slave的MAC地址。
该模式的必要条件：ethtool支持获取每个slave的速率。交换机侧目前无需配置任何链路模式。
第七种模式：mod=6，即：(balance-alb) Adaptive load balancing（适配器适应性负载均衡）
特点：该模式包含了balance-tlb模式，同时加上针对IPV4流量的接收负载均衡(receive load balance, rlb)，而且不需要任何switch(交换机)的支持。接收负载均衡是通过ARP协商实现的。bonding驱动截获本机发送的ARP应答，并把源硬件地址改写为bond中某个slave的唯一硬件地址，从而使得不同的对端使用不同的硬件地址进行通信。
来自服务器端的接收流量也会被均衡。当本机发送ARP请求时，bonding驱动把对端的IP信息从ARP包中复制并保存下来。当ARP应答从对端到达 时，bonding驱动把它的硬件地址提取出来，并发起一个ARP应答给bond中的某个slave。使用ARP协商进行负载均衡的一个问题是：每次广播 ARP请求时都会使用bond的硬件地址，因此对端学习到这个硬件地址后，接收流量将会全部流向当前的slave。这个问题可以通过给所有的对端发送更新 （ARP应答）来解决，应答中包含他们独一无二的硬件地址，从而导致流量重新分布。当新的slave加入到bond中时，或者某个未激活的slave重新 激活时，接收流量也要重新分布。接收的负载被顺序地分布（round robin）在bond中最高速的slave上。交换机侧目前无需任何链路模式对接。
当某个链路被重新接上，或者一个新的slave加入到bond中，接收流量在所有当前激活的slave中全部重新分配，通过使用指定的MAC地址给每个 client发起ARP应答。下面介绍的updelay参数必须被设置为某个大于等于switch(交换机)转发延时的值，从而保证发往对端的ARP应答 不会被switch(交换机)阻截。
必要条件：
条件1：ethtool支持获取每个slave的速率；
条件2：底层驱动支持设置某个设备的硬件地址，从而使得总是有个slave(curr_active_slave)使用bond的硬件地址，同时保证每个bond 中的slave都有一个唯一的硬件地址。如果curr_active_slave出故障，它的硬件地址将会被新选出来的 curr_active_slave接管
其实mod=6与mod=0的区别：mod=6，先把eth0流量占满，再占eth1，….ethX；而mod=0的话，会发现2个口的流量都很稳定，基本一样的带宽。而mod=6，会发现第一个口流量很高，第2个口只占了小部分流量。

交换机侧有两种链路捆绑模式，一种是静态聚合，一种是动态聚合。静态对应服务器侧的bond 0，动态对应服务器侧的bond 4。