Eth-Trunk 负载分担不均怎么办

文档版本 01

发布日期 2021-08-16

华为技术有限公司

商标声明


和其他华为商标均为华为技术有限公司的商标。
本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或默示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为技术有限公司

地址： 深圳市龙岗区坂田华为总部办公楼 邮编：518129

网址： https://www.huawei.com

客户服务邮箱： support@huawei.com

客户服务电话： 4008302118

目 录

1 什么是负载分担.....1
2 如何通过 Hash 算法实现负载分担.....2
3 如何处理 Eth-Trunk 负载分担不均.....4
3.1 成员数量是否为 2 的 N 次方.....4
3.2 是否存在跨设备堆叠.....4
3.3 是否存在 Hash 极化问题.....6
3.4 流量类型与负载分担模式是否匹配.....7

1 什么是负载分担

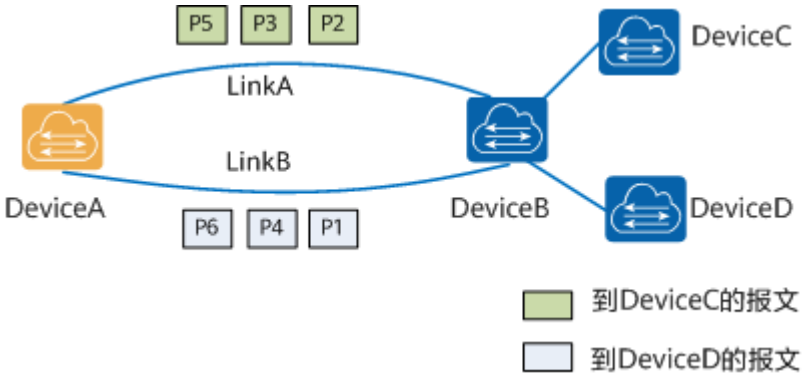
在网络部署当中，当存在多条转发路径的时候，常常会部署负载分担功能。通过部署负载分担，设备可以基于报文内容等进行逐流转发，或者基于随机数、轮转方式进行逐包转发，以达到充分利用链路，提高转发效率的目的。

基于逐包负载分担方式的实际部署较少，因为该方式可能导致同一个用户的流量经过网络的不同路径传输后，到达目的设备出现报文乱序或多份的情况。常见的负载分担方式一般选择基于逐流进行负载分担。

逐流负载分担即按照一定的规则，如根据五元组（源IP地址、目的IP地址、协议号、源端口号、目的端口号），将报文分成不同的流。同一条流的报文，经过Hash计算后，会在同一条链路上转发。

如图1-1所示，假设DeviceA上有6个报文要通过DeviceA和DeviceB之间的LinkA和LinkB进行负载分担，其发送顺序为P1、P2、P3、P4、P5、P6，其中P2、P3和P5去往DeviceC，P1、P4、P6去往DeviceD。假设负载分担采用逐流方式，则去往DeviceC的报文都通过LinkA发送，去往DeviceD的报文都通过LinkB发送；或者去往DeviceC的报文都通过LinkB发送，去往DeviceD的报文都通过LinkA发送。

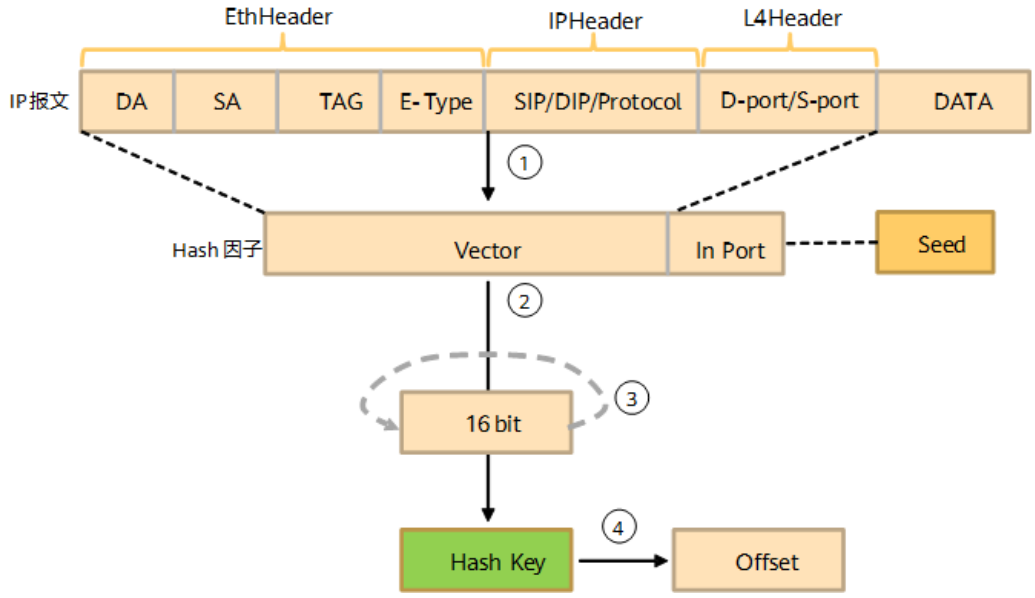
图 1-1 逐流负载分担示意图



2 如何通过 Hash 算法实现负载分担

常见的负载分担处理过程包含输入（流量、报文的有效字段）、处理（通过Hash算法进行计算）和输出（根据计算结果将流量通过相应的出接口转发）。其中，通过Hash计算的结果会直接影响负载分担的效果，因此如何利用好Hash算法进行计算，在负载分担部署当中就显得尤为重要。

图 2-1 Hash 算法流程



如图2-1所示，Hash计算的流程如下：

1. 获取报文的信息，对于普通的IP报文，为源MAC、目的MAC、源IP、目的IP、VLAN、四层端口号等。按照配置的流量模型，将这些信息作为Hash因子的参考值。
2. 根据Hash算法和Seed对Hash因子进行计算，得到Hash Key。

   其中，Hash算法是芯片提供固定种类的算法，不同的算法对于不同的流量模型计算的效果不同，有多种算法以供选择，可以通过**hash-mode** *hash-mode-id*参数选择Hash算法。

   另外，Seed是一个数值，用于参与计算。在相同Hash因子的情况下，Seed值会影响计算出的Hash Key的值，通过命令**seed** *seed-data*设置Seed值。

3. 为了让同一个Hash因子衍生更多的变化，以更加灵活地适应不同的Hash场景，设备芯片可以对Hash Key的值进行0~15位的偏移，可以通过**universal-id** *universal-id*参数进行设置Hash Key的值。
4. 将Hash Key转化为Offset值，Offset值与出口个数进行取余运算，根据结果决定报文从设备的哪一个接口发出去。

3 如何处理 Eth-Trunk 负载分担不均

正常情况下，流量在Eth-Trunk负载分担后，会被分配到多条链路上传输。Eth-Trunk负载分担不均是指流量通过负载分担后，仅被集中分配到一条或者某几条链路上传输，而其他链路无流量或者流量较少的情况。如果单条链路的流量较大，可能会影响业务的正常运行。

如何解决Eth-Trunk成员出现负载不均的情况，可以从以下几个方面考虑：

- **成员数量是否为2的N次方**
- **是否存在跨设备堆叠**
- **是否存在Hash极化问题**
- **流量类型与负载分担模式是否匹配**

3.1 成员数量是否为2的N次方
3.2 是否存在跨设备堆叠
3.3 是否存在Hash极化问题
3.4 流量类型与负载分担模式是否匹配

3.1 成员数量是否为 2 的 N 次方

Eth-Trunk每个发送周期有16个发送报文的时隙，Eth-Trunk成员口轮流使用这16个时隙发送报文。

当Eth-Trunk成员口数量是2的N次方时，可以使得负载分担更均匀。例如，如果Eth-Trunk成员口数量是2、4或8（可以整除16），每个接口得到的发送报文的时隙是整数，负载分担就均匀；如果成员口数量不是2的N次方（比如3个），在16个时隙里有1个接口得到了6次发送报文的机会，另外两个接口只得到5次发送报文的机会，负载分担就不均匀。因此，Eth-Trunk成员口数量最好是2的N次方，保证负载分担更均匀。

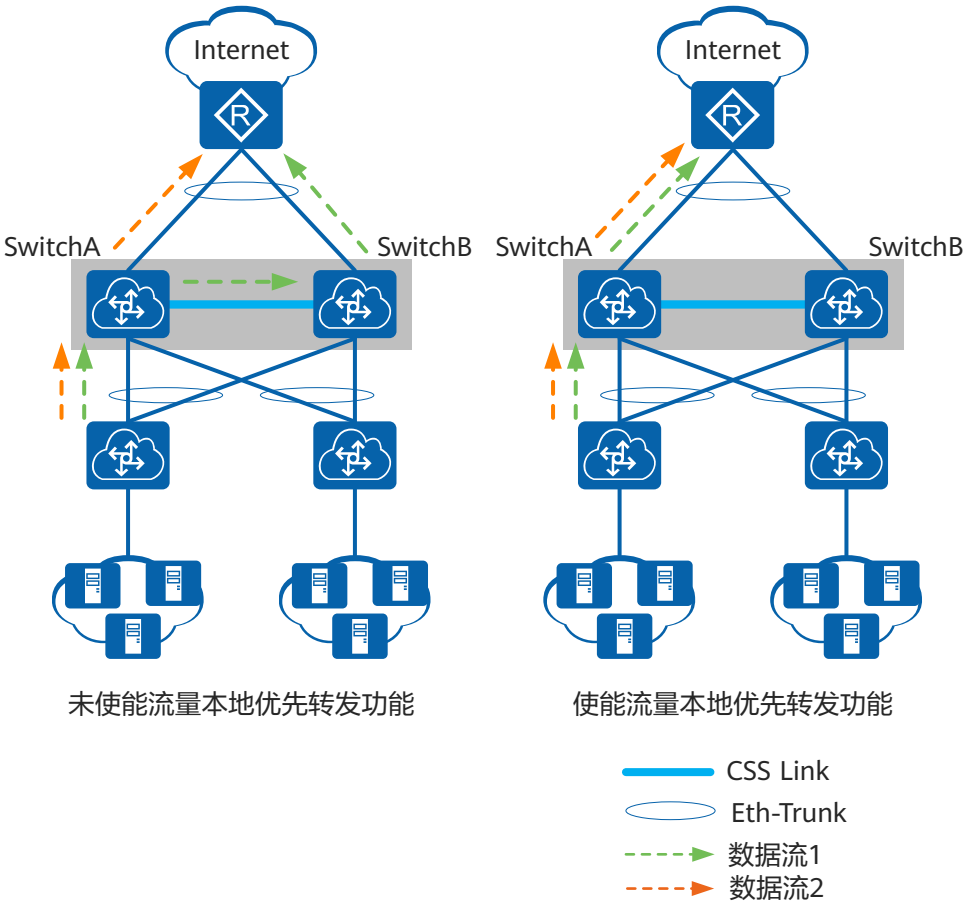
3.2 是否存在跨设备堆叠

CE交换机在堆叠场景下默认开启Eth-Trunk本地优先转发功能，即从本设备进入的流量，优先从本设备的出接口转发出去。本地优先转发可以降低转发延时，降低堆叠链路的利用率。

如**图3-1**所示，SwitchA与SwitchB组成堆叠，上下行加入到Eth-Trunk。如果没有本地优先转发，则从SwitchA进入的流量，会有一部分经过堆叠线缆，从SwitchB的物理接

口转发出去。设备启用本地优先转发之后，从SwitchA进入的流量，优先从SwitchA的接口转发。

图 3-1 流量本地优先转发示意图



默认开启本地优先转发的情况下，同框的Eth-Trunk成员口负载分担均匀，不同框的Eth-Trunk成员口负载分担不均匀。要解决跨设备堆叠场景中的不同设备的成员口负载分担不均问题，可以按照以下步骤处理。

- 在任意视图下，执行命令**display interface eth-trunk** [*trunk-id* [*.subnumber*] | **main**]，检查Eth-Trunk成员口是否跨设备，如堆叠场景下不同设备上的接口捆绑成Eth-Trunk端口。
- 如果Eth-Trunk成员口是跨设备的，去使能Eth-Trunk接口流量本地优先转发功能。即在Eth-Trunk接口视图下，执行命令**local-preference disable**。

```
<HUAWEI> system-view
[~HUAWEI] interface eth-trunk 10
[*HUAWEI-Eth-Trunk10] local-preference disable
[*HUAWEI-Eth-Trunk10] commit
```

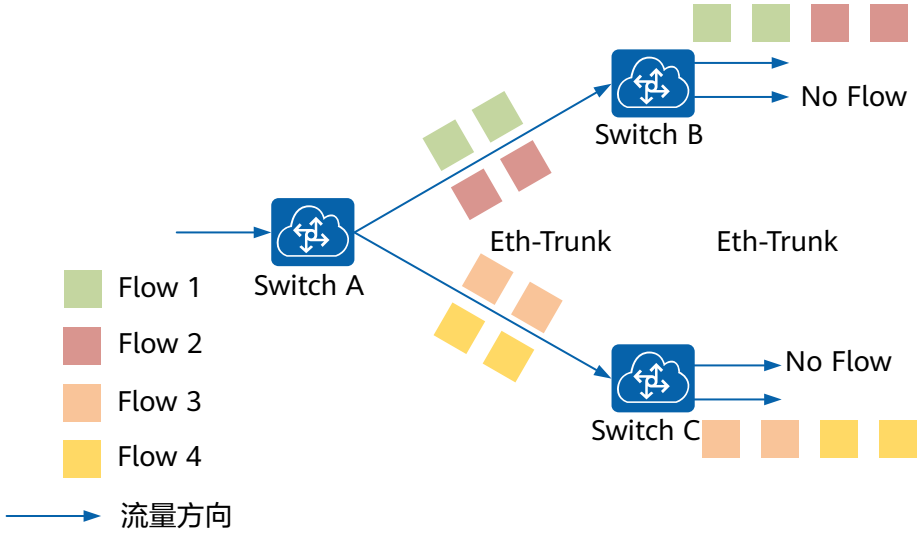
说明

关闭本地优先转发功能后，部分流量会跨设备之间的堆叠链路，需要保证该链路带宽足够。

3.3 是否存在 Hash 极化问题

Hash极化，也被称为Hash不均，是指流量经过2次或2次以上Hash后出现的负载分担不均匀的现象。常见于跨设备的多次Hash场景，即第一级进行Eth-Trunk Hash，第二级再进行ECMP Hash或者Eth-Trunk Hash。在同一设备上，若存在ECMP的出接口为多个Eth-Trunk也可能会出现Hash极化。

图 3-2 Hash 极化示意图



如图3-2所示，Switch A的入接口有4种流量，出接口为2条等价链路，经Hash计算，流量1和流量2走上面的链路到Switch B；流量3和流量4走下面的链路到Switch C。在Switch B出接口同样为2条等价链路，若采用与Switch A相同或者类似的Hash算法，其Hash的结果将为流量1和流量2走上面的链路，而下面的链路没有流量。Switch C的情况类似。这种经过多次Hash后，ECMP或者Eth-Trunk各成员口之间流量极度不均匀的现象称为Hash极化。

实际上，交换机Hash功能的实现很大程度上取决于芯片，所以当使用同类型芯片的交换机位于网络中相邻的层级时，就可能会出现Hash极化问题。因此，在多级网络中部署ECMP或者Eth-Trunk负载分担，需要考虑出现Hash极化问题的风险。

解决两级负载分担场景下的哈希极化问题，就是要避免两级设备使用相同的负载分担参数。

1. 如果流量有多个特征有较大变化时，可以让两级设备采用不同的哈希因子，比如第一级Eth-Trunk使用源IP进行哈希，第二级使用目的IP进行哈希。

   # 第一级Eth-Trunk使用源IP进行哈希。

   ```
   <HUAWEI> system-view
   [~HUAWEI] load-balance profile default
   [~HUAWEI-load-balance-profile-default] ip src-ip
   [*HUAWEI-load-balance-profile-default] commit
   ```

   # 第二级Eth-Trunk使用目的IP进行哈希。

   ```
   <HUAWEI> system-view
   [~HUAWEI] load-balance profile default
   ```

```
[~HUAWEI-load-balance-profile-default] ip dst-ip
[*HUAWEI-load-balance-profile-default] commit
```

2. 如果调整哈希因子后效果不明显，通过执行命令**eth-trunk hash-mode** *hash-mode-id*，调整两级的哈希算法为不同的算法。
3. 如果调整哈希算法任然没有效果，通过执行命令**eth-trunk universal-id** *universal-id*，调整Eth-Trunk的偏移量universal-id。

3.4 流量类型与负载分担模式是否匹配

判断Eth-Trunk接口转发的报文特征和配置的负载分担方式是否匹配。如果不匹配，例如转发报文的MAC地址变化，而设置的负载分担方式为src-ip，则无法负载分担。

识别报文类型

- 确定报文的类型。

  确定报文为IP报文、MPLS报文、TRILL报文、FCoE报文等。

- 不同报文的负载分担方式。

  针对不同类型的报文，可以分别配置负载分担模式。以CE6856HI为例，负载分担方式如**表1 不同报文的负载分担方式**所示。例如，对于IPv4报文，默认情况下根据源IP、目的IP、目的端口号、源端口号进行负载分担，也可以通过命令行配置负载分担模式。需要获取其他款型的负载分担方式，请参考**配置负载分担方式**进行配置。

表 3-1 不同报文的负载分担方式

报文（以入接口为准）	默认负载分担模式	可配置的负载分担模式
IPv4报文	**src-ip**、**dst-ip**、**l4-src-port**、**l4-dst-port**	**src-ip**、**dst-ip**、**l4-src-port**、**l4-dst-port**、**protocol**
IPv6报文	**src-ip**、**dst-ip**、**l4-src-port**、**l4-dst-port**	**src-ip**、**dst-ip**、**protocol**、**l4-src-port**、**l4-dst-port**
MPLS报文	Ingress/Egress/Transit节点：**top-label**、**2nd-labeltop-label**、**2nd-label**	Ingress/Egress/Transit节点：**top-label**、**2nd-label**、**dst-ip**、**src-ip**
非以上类别的L2报文	**src-mac**、**dst-mac**	**src-mac**、**dst-mac**、**src-interface**、**eth-type**
TRILL报文	Ingress节点：二层报文基于内层的**src-mac**和**dst-mac**；三层报文基于**src-ip**、**dst-ip**、**l4-src-port**和**l4-dst-port**	**src-mac**、**dst-mac**、**src-ip**、**dst-ip**、**src-interface**、**l4-src-port**、**l4-dst-port**
	Transit/Egress节点：二层报文基于内层的**src-mac**和**dst-mac**；三层报文基于**src-ip**、**dst-ip**、**l4-src-port**和**l4-dst-port**	**src-mac**、**dst-mac**、**src-ip**、**dst-ip**、**l4-src-port**、**l4-dst-port**

报文（以入接口为准）	默认负载分担模式	可配置的负载分担模式
FCoE报文	**dst-fcid**、**src-fcid**	**dst-fcid**、**src-fcid**

配置负载分担方式

- 配置已知单播的负载分担方式
  a. 执行命令**interface eth-trunk** *trunk-id*，进入Eth-Trunk接口视图。
  b. 执行命令**load-balance** { **dst-ip** | **dst-mac** | **random** | **round-robin** | **src-ip** | **src-mac** | **src-dst-ip** | **src-dst-mac** | **enhanced** [**resilient**] **profile** *profile-name* }，配置Eth-Trunk负载分担方式。

     用户可以根据流量模型设置不同的负载分担方式来选择各种负载分担模式，流量中该参数变化越频繁，选择此负载分担模式的流量就越均衡。例如，在网络中，如果报文的IP地址变化较频繁，那么选择基于**dst-ip**、**src-ip**或**src-dst-ip**的负载分担模式更有利于流量在各物理链路间合理的负载分担；如果报文的MAC地址变化较频繁，IP地址比较固定，那么选择基于**dst-mac**、**src-mac**或**src-dst-mac**的负载分担模式更有利于流量在各物理链路间合理的负载分担。
  c. 执行命令**commit**，提交配置。
- 配置未知单播的负载分担方式
  a. 执行命令**load-balance unknown-unicast** { **mac** | **enhanced** }，配置未知单播的负载分担方式。

     在网络中，对于未知单播，如果IP报文的源MAC地址或者目的MAC变化较频繁，而IP地址比较固定，那么选择参数**mac**对未知单播进行负载分担；如果IP报文的源IP地址或者目的IP地址变化较频繁，而MAC地址比较固定，那么选择参数**enhanced**对未知单播进行负载分担。
  b. 执行命令**commit**，提交配置。

