

IPv6路由协议

3.1 IS-IS 路由协议



IS-IS(Intermediate System to Intermediate System)最初是由 Digital Equipment Corporation (DEC)和 Radian Corporation 共同开发的,作为 DECnet Phase V 网络协议套件的一部分。DECnet 是一种用于构建大型企业网络的协议套件,旨在支持分布式计算环境。IS-IS 协议最初用于 DECnet 网络,后来随着互联网的发展和广泛应用,被广泛用于大型企业和互联网服务提供商的网络中。它成为了一种内部网关协议(Internal Gateway Protocol,IGP),用于在自治系统(Autonomous System,AS)内部的路由选择和转发。

3.1.1 IS-IS 架构

IS-IS 架构包括 IS-IS 路由器、IS-IS 数据库、链路状态和链路状态数据库。IS-IS 路由器通过交换链路状态数据包来获取网络的拓扑结构,并使用链路状态数据库中的信息计算出最佳的路由路径。这种基于链路状态的路由算法和分布式的链路状态数据库使得 IS-IS 协议具有可扩展性、快速收敛和灵活的策略控制等特点。

1. IS-IS 路由器

IS-IS 路由器是网络中运行 IS-IS 协议的设备。它可以是路由器、三层交换机或者其他支持 IS-IS 协议的设备。IS-IS 路由器通过交换链路状态数据包(Link State PDU,LSP)来获取网络的拓扑结构信息,并使用这些信息计算出最佳的路由路径。IS-IS 路由器之间通过直接链路建立邻居关系,并通过链路状态数据库(Link State Database)来维护网络的拓扑信息。

2. IS-IS 数据库

IS-IS 数据库是每个 IS-IS 路由器都维护的一个存储路由信息的数据库。它包含了网络中的拓扑信息、链路状态和路由表等。IS-IS 数据库由多个链路状态数据库组成,每个链路状态数据库对应一个区域(Area)。每个 IS-IS 路由器都会维护自己所属区域的链路状态数据库,并与其他区域的 IS-IS 路由器交换链路状态信息,从而构建整个网络的拓扑图。

3. 链路状态

链路状态是指 IS-IS 路由器在网络中感知到的链路的状态信息,包括链路的状态、带宽、延迟、可靠性等。IS-IS 路由器会将链路状态信息封装为链路状态报文 PDU 并交换给邻居路由器,以便邻居路由器了解网络的拓扑结构。

4. 链路状态数据库

链路状态数据库是 IS-IS 路由器存储链路状态信息的地方。每个 IS-IS 路由器都会维护自己所属区域的链路状态数据库,其中包含了本区域内的链路状态信息。IS-IS 路由器通过交换链路状态信息与其他区域的 IS-IS 路由器建立全局的链路状态数据库,以便计算出最佳的

路由路径。

IS-IS 路由器使用链路状态数据库中的信息来计算出最佳的路由路径。它使用基于链路状态的路由算法,如 Dijkstra 算法,来计算从源节点到目的节点的最短路径。路由计算结果会存储在 IS-IS 路由器的路由表中,用于转发数据包。

IS-IS 架构具有下列特征:

(1) 支持路由器之间的动态路由。IS-IS 协议允许路由器之间动态地交换路由信息,以便实现自动路由选择和网络故障恢复。它能够动态适应网络拓扑的变化,并在需要时更新路由表。

(2) 可扩展性。IS-IS 协议设计用于大型网络,具有良好的可扩展性。它使用层次结构的设计,可以将网络划分为多个区域(Area),每个区域内部使用 IS-IS 协议进行路由,而区域之间使用区域边界路由器(Area Border Router)进行路由转发。

(3) 快速收敛。IS-IS 协议使用了快速收敛的机制,以便在网络故障或拓扑变化时能够迅速调整路由。它使用了基于链路状态的路由算法,通过交换链路状态报文 LSP 来更新路由信息,从而实现快速的网络收敛。

(4) 支持多种网络层协议。IS-IS 协议可以在不同的网络层协议(如 IPv4 和 IPv6)上运行,以支持不同版本的 IP 协议。它可以适应不同的网络环境,并提供一致的路由功能。

(5) 安全性。IS-IS 协议支持对路由信息进行认证和加密,以确保路由信息的安全性和完整性。它可以使用密钥和认证算法来验证和保护路由信息的传输。

(6) 灵活的策略控制。IS-IS 协议提供了灵活的策略控制功能,可以根据网络管理员的需求进行路由策略的配置和控制。它支持路由策略的过滤、重定向和优先级控制,以实现更精细的路由控制和管理。

3.1.2 控制机制

1. 邻居关系建立

IS-IS 协议使用 Hello 消息来建立和维护邻居关系,也称为 IIH(IS-to-IS Hello PDU)消息。当两个 IS-IS 路由器在相邻链路上收到对方的 Hello 消息时,它们将建立邻居关系。Hello 消息包含了路由器的标识和链路状态信息,用于验证邻居的可达性和链路的状态。通过建立邻居关系,路由器可以交换路由信息并了解相邻路由器的存在。

1) 广播网络

采用的是可靠的邻接建立过程,如果在接收的 IIH 报文中看到了自己接口的 MAC 地址,说明邻居已经收到并确认了自己发送的 IIH 报文,那么邻居的状态将变为 UP。

2) 点到点网络

两次握手没有可靠性保证。只要收到邻接发送的 IIH 报文,并通过检测,更新邻居的状态为 UP。如图 3-1 所示,3 次握手可保证邻接建立的可靠性。新增一种点到点邻居状态 TLV。

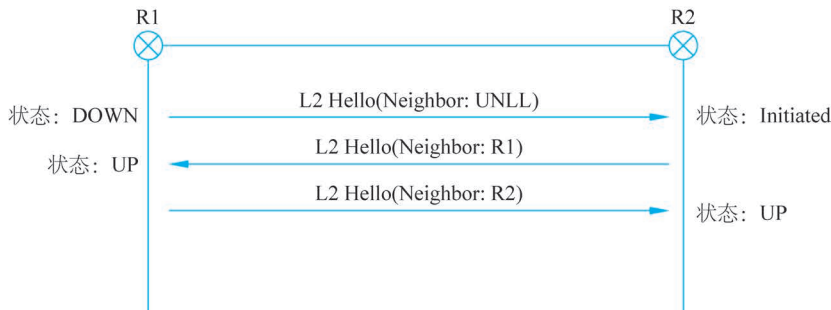


图 3-1 3 次握手

如图 3-2 所示,两次握手机制存在明显的缺陷。当路由器间存在两条及以上的链路时,如果某条链路上到达对端的单向状态为 DOWN,而另一条链路同方向的状态为 UP,则路由器之间能建立起邻接关系。



图 3-2 两次握手

3) 邻接关系建立条件

- 只有同一层次的相邻路由器才有可能成为邻居。
- 对于 Level-1 路由器来说,区域号必须一致。
- 链路两端 IS-IS 接口的网络类型必须一致。
- 链路两端 IS-IS 接口的地址必须处于同一网段

2. 路由信息交换

IS-IS 协议使用 LSP 来交换路由信息。每个 IS-IS 路由器维护一个链路状态数据库,其中包含了网络拓扑信息和路由信息。IS-IS 路由器通过交换 LSP 来更新链路状态数据库,并将更新的信息分发给其他路由器。LSP 包含了路由器的标识、邻居信息、链路状态和其他相关信息。

3. SPF 计算

SPF(Shortest Path First)计算是 IS-IS 协议的核心机制之一,如图 3-3 所示。当链路状态数据库发生变化时,IS-IS 路由器使用 SPF 算法计算出最短路径树,并确定到达目的地的最佳路径。SPF 计算考虑了链路的开销(通常是链路带宽)来选择最佳路径。SPF 算法通过逐步迭代的方式计算最短路径树,从而确定每个目的地的最佳路径。

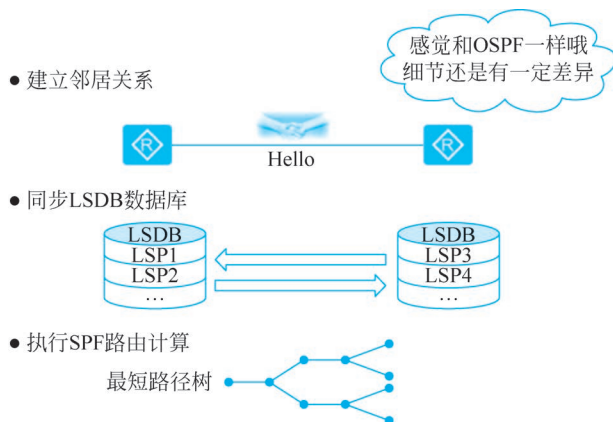


图 3-3 IS-IS 处理流程

4. 路由选择、聚合、控制

IS-IS 协议使用 SPF 计算来选择最佳路径,并将其作为路由表中的条目。IS-IS 支持等价路径的负载均衡,可以在多个等价路径之间进行流量分担。当存在多个等价路径时,IS-IS 路

由器可以根据配置或算法来决定如何分配流量。这种负载均衡的能力可以提高网络的性能和资源利用率。

IS-IS 支持路由聚合(Route Aggregation),可以将多个具体路由聚合成更具概括性的路由,从而减少路由表的大小和复杂性。路由聚合可以减少路由器之间交换的路由信息数量,降低网络的控制平面开销。聚合的路由可以代表一组具有相同目的地前缀的子网,从而简化路由表的管理。

IS-IS 协议支持路由策略控制,可以通过配置路由策略来影响路由选择的决策。路由策略控制允许网络管理员根据需要对路由进行优先级设置、过滤和筛选等操作。管理员可以配置路由策略来实现流量工程、安全性要求、服务质量(QoS)和其他策略目标。这种灵活的策略控制能力使得 IS-IS 协议适用于各种复杂网络环境。

5. 控制报文类型

IS-IS 协议使用不同类型的报文来实现邻居发现、链路状态同步和路由信息传播。常见的报文有邻居发现报文(IS-to-IS Hello PDU,IIH)、链路状态报文(Link State PDU,LSP)、完全序列号报文(Complete Sequence Number PDU,CSNP)和部分序列号报文(Partial Sequence Number PDU,PSNP)。

1) IIH 报文

IIH 报文的作用是邻居发现,协商参数并建立邻居关系,后期充当保活报文。IS-IS 建立邻居关系和 OSPF 一样,通过 IIH 报文的交互来完成。但是会根据场景分为 3 种类型的 IIH 报文。目前 IS-IS 只支持点对点网络和广播网络两种类型。IIH 报文需要通过填充字段用于在邻居两端协商发送报文的大小,如图 3-4 所示。

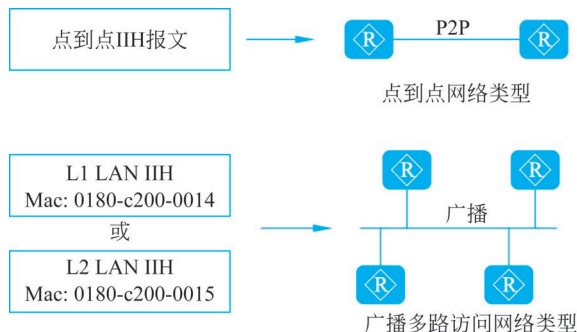


图 3-4 不同网络类型下的 IIH 报文

2) LSP 报文

LSP 分为 Level-1 LSP、Level-2 LSP,用于交换链路状态信息,描述路由器的接口及所连网络的信息,包括接口所连网络的子网、类型、开销等信息,触发即时更新或周期性更新。

3) CSNP 报文

CSNP 分为 Level-1 CSNP、Level-2 CSNP,包括 LSDB 中所有 LSP 的摘要信息,从而可以在相邻路由器间保持 LSDB 的同步。在广播网络上,CSNP 由 DIS 定期发送(默认发送周期为 10s);在点到点链路上,CSNP 只在第一次建立邻接关系时发送。

4) PSNP 报文

PSNP 分为 Level-1 PSNP、Level-2 PSNP,只列举最近收到的一个或多个 LSP 的序号,它能够一次对多个 LSP 进行确认,当发现 LSDB 不同步时,也用 PSNP 来请求邻居发送新的 LSP。

3.1.3 报文格式

1. 通用报文格式

IS-IS 协议报文可以分为两个部分：报文头和变长字段。其中报文头部又可以细分为通用头部和专用头部。对于所有的 IS-IS 报文来说，通用报文头部都是相同的，但是专用报文头部根据不同的 IS-IS 报文种类而不同。

IS-IS 报文格式如图 3-5 所示。

通用报文头部 (Common Header)
专用报文头部 (Specific Header)
变长字段 (Variable Length Fields)

图 3-5 IS-IS 报文格式

IS-IS 通用报文头格式如图 3-6 所示。

Interdomain Routing Protocol Discriminator	
Length Indicator	
Version/Protocol ID Extension	
ID Length	
Reserved	PDU Type
Version	
Reserved	
Maximum Area Address	

图 3-6 IS-IS 通用报文头格式

其中，各个字段含义如下。

- Interdomain Routing Protocol Discriminator：域内路由协议鉴别符，占用 1 字节，IS-IS 协议固定为 0x83。
- Length Indicator：IS-IS 报文头部长度的（单位为字节），包括通用报文和专用报文头部，不包括 TLV 字段，占用 1 字节。
- Version/Protocol ID Extension：版本/协议标识扩展，固定为 0x1，占用 1 字节。
- ID Length：NSAP 地址中，System ID 区域的长度，长度为 1 字节。注意，当该字段值为 0 时，表示 System ID 长度为 6，值为 255 时，表示 System ID 长度为 0。
- Reserved：保留字段，恒为 0。
- PDU Type：IS-IS PDU 报文类型（一共 9 种）。
- Version：IS-IS 版本，恒为 0x1（IS-IS 通用报文头部包含两个 Version 字段）。
- Maximum Area Address：同时支持的最大区域个数，占用 1 字节。该字段值默认值为 0，且此时的 0 表示支持的最大区域个数为 3。

2. Hello 报文专用报文头格式

广播链路 Hello 报文专用报文头格式如图 3-7 所示。

Reserved/Circuit Type	
Source ID	
Holding Time	
PDU Length	
R	Priority
LAN ID	

图 3-7 广播链路 Hello 报文专用报文头格式

点对点链路 Hello 专用报文头格式如图 3-8 所示。

Reserved/Circuit Type
Source ID
Holding Time
PDU Length
Local Circuit ID

图 3-8 点对点链路 Hello 专用报文头格式

相比之下,两者的区别在于优先级(Priority)字段和 LAN ID/Local Circuit ID 字段。其中,各个字段含义如下。

- Reserved/Circuit Type: 路由器级别,0x01 表示 L1 级别,0x02 表示 L2 级别,0x03 表示 L1/L2 级别(注:此字段与链路上 IS-IS 级别也有关系,必须当路由器级别和接口级别均为 L1/L2,此时此字段才为 L1/L2)。
- Source ID: 发送报文的 IS-IS 路由器 SID,占 6 字节。
- Holding Time: 保持计时器,单位为秒,占 2 字节,默认为 Hello-Interval 的 3 倍。本字段的含义是告诉邻居本端失效的时间。这里 IS-IS 协议与 OSPF 不同,IS-IS 允许建立邻居的相邻路由器之间 Hello-Interval 和 Holding Time 不一致。可以在路由器 IS-IS 视图下修改本字段时间,但是有一定的限制。
- PDU Length: 本字段表示整个 IS-IS 报文的长度,包括头部、Hello 报文以及后面的 TLV 字段的长度,占 2 字节。
- Priority: 表示优先级,占 1 字节,但是该字段值使用了 7 比特,因此该字段取值范围是 0~127。该字段主要用于 DIS 的选举,且该字段越大越优先。当该字段值相等时,则比较 MAC 地址,越大越优先。
- R: 预留字段,用 0 填充。
- LAN ID: 用于描述链路上的伪节点 DIS。
- Local Circuit ID: 表示本地链路 ID。

3. LSP 报文专用报文头格式

LSP 报文专用报文头格式如图 3-9 所示。

PUD Length			
Remaining Lifetime			
LSP ID			
Sequency Number			
CheckSum			
P	ATT	OL	IS Type

图 3-9 LSP 报文专用报文头格式

其中,各个字段含义如下。

- PDU Length: PDU 的总长度,单位为字节。
- Remaining Lifetime: LSP 生存时间,单位为秒。
- LSP ID: LSP ID,由 System id、伪节点标识符和分片标识符 3 部分组成,唯一标识一条 LSP 报文。
- Sequency Number: LSP 的序列号。
- Checksum: LSP 的校验和。
- P(Partition Repair): 仅与 L2-LSP 有关,表示路由器是否支持修复区域分割。
- ATT(Attachment): 一般由 L1/L2 路由器产生,用于控制 L1 区域对 L2 区域的路由学习情况。
- OL(LSDB Overload): 过载标志位。
- IS Type: 生成的 LSP 的路由器的级别,用来指明是 L1 的路由器还是 L2 的路由器。

4. CSNP 报文专用报文头格式

CSNP 报文专用报文头格式如图 3-10 所示。

其中,各个字段含义如下。

- PDU Length: PDU 的总长度。
- Source ID: 发出报文的路由器的 System ID。
- Start LSP ID: 报文中第一个 LSP 的 ID 值。
- End LSP ID: 报文中最后一个 LDP 的 ID 值。
- Variable Length Fields: 变长。

5. PSNP 报文专用报文头格式

PSNP 报文专用报文头格式如图 3-11 所示。

PDU Length
Source ID
Start LSP ID
End LSP ID
Variable Length Fields

图 3-10 CSNP 报文专用报文头格式

PDU Length
Source ID
Variable Length Fields

图 3-11 PSNP 报文专用报文头格式

其中,各个字段含义同 CSNP 报文。

3.2 面向 IPv6 的 IS-IS 协议扩展

3.2.1 IS-IS 的 IPv6 扩展

IETF 规范 RFC5308(Routing IPv6 with IS-IS)通过扩展 IS-IS 路由协议,使得该协议可以承载执行 IPv6 路由所需的信息,从而实现通过 IS-IS 路由协议交换 IPv6 路由信息,支持 IPv6、IPv4 和 OSI 共 3 种协议机制。

为了支持 IPv6 路由,规范定义了两个新的 TLV[“IPv6 可达性”(IPv6 Reachability)TLV、“IPv6 接口地址”(IPv6 Interface Address)TLV]以及一个新的 IPv6 协议标识符 NLPID 来扩展 IS-IS 协议。此外,仿照 RFC5305(IS-IS Extensions for Traffic Engineering)中对 RFC1195(Use of OSI IS-IS for routing in TCP/IP and dual environments)所做的更改,RFC5308 对 RFC5305 做出相同更改来处理新的前缀信息。

然而,上述规范中未明确说明 IPv4/IPv6 扩展可达性 TLV 的路由首选项。当前缀通告出现在 L2 LSP 中时,UP/DOWN 位如何应用于路由首选项的定义也存在不一致。因此,规范 RFC7775(IS-IS Route Preference for Extended IP and IPv6 Reachability)定义了 TLV135 的显式路由首选规则,修改了 TLV236 的路由首选规则,并澄清了当 UP/DOWN 位出现在 L2 LSP 的 TLV 中时的用法。RFC7775 是对 RFC5302(Domain-Wide Prefix Distribution with Two-Level IS-IS)和 RFC5305 的澄清;也是对 RFC5308 中定义的路由首选项规则的修正,以与 IPv4 的规则一致。同时,RFC7775 还明确指出,相同的规则适用于多拓扑(MT)等效 TLV235 和 TLV237。

3.2.2 报文格式

1. “IPv6 可达性”TLV

“IPv6 可达性”TLV 的 TLV 类型值为 236 (0xEC),它通过前缀、度量、标记等来描述可达的 IPv6 前缀信息。

RFC1195 中面向 IPv4 定义了两个可达性 TLV[“IP 内部可达性信息”(IP Internal Reachability Information)和“IP 外部可达性信息”(IP External Reachability Information)],而规范[RFC5308]使用一个“X”bit 来区分“内部”和“外部”。

“IPv6 可达性”TLV 通过指定路由前缀、度量信息、指示前缀是否从更高级别向下通告的标志位、指示前缀是否从另一个路由协议分发的标志位等字段来描述网络可达性,并且支持子 TLV 以允许以后的扩展。该数据结构如图 3-12 所示。

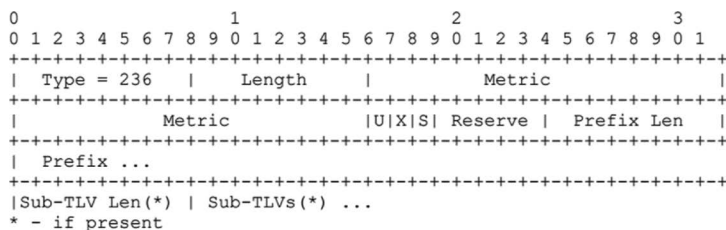


图 3-12 “IPv6 可达性”TLV 格式

各字段含义如下。

- Type: 8bit, TLV 类型, 此时值为 236(0xEC)。
- Length: 8bit, TLV 的 Value 部分长度。
- Metric: 32bit, 度量值。
- U: 1bit, UP/DOWN 位, 标识这个前缀是否是从较高级别通告下来的, 用于防止环路。
- X: 1bit, External Original 位, 标识这个前缀是否是从其他路由协议中引入的。
- S: 1bit, Sub-TLV Present 位, 子 TLV 标识位(可选)。
- Reserve: 5bit, 保留位。
- Prefix Len: 8bit, 前缀长度。
- Prefix: IPv6 地址前缀。
- Sub-TLV Len: 8bit, 子 TLV 长度。若 S 位置 1, 则存在。
- Sub-TLVs: 子 TLV。若 S 位置 1, 则存在。

上述“IPv6 可达性”TLV 可以在 LSP 中出现任意次数(包括 0 次), 但不能使用此 TLV 通告链路本地前缀。

如 RFC5305 中所述: 当前缀首次注入 IS-IS 时, UP/DOWN 标志位应设置为 0。如果前缀从较高级别通告到较低级别(如 level-2 到 level-1), 该位应设置为 1, 表示前缀已向下传播。将 UP/DOWN 位设置为 1 的前缀只能在层次结构中向下通告, 即向较低级别通告。

如果前缀是从另一个路由协议分发到 IS-IS, 则 External Original 位应设置为 1。当将前缀从 IS-IS 分发到其他协议时, 此信息很有用。

如果 Sub-TLV 位设置为 0, 则不存在 Sub-TLV 字段; 否则, 该位为 1, 前缀后面的 1 字节将包含结构的 Sub-TLV 部分的长度。

前缀在数据结构中“打包”, 即只存在所需数量的对应字节长度, 该数字可以从前缀长度字段计算: 前缀所占字节长度 = ((前缀长度 + 7) / 8) 的取整。

如 RFC5305 所述, 若使用大于 MAX_V6_PATH_METRIC(0xFE000000) 的度量来通告前缀, 则在正常的最短路径优先(Shortest Path First, SPF)计算期间不得考虑该前缀。这将允许为构建普通 IPv6 路由表以外的目的发布前缀。

2. “IPv6 接口地址”TLV

“IPv6 接口地址”TLV 是 TLV 类型 232(0xE8), 与 TLV132 相似, 相当于 RFC1195 中面向 IPv4 的“IP 接口地址”(IP Interface Address) TLV, 把原来的内容为 0~63 的 4 字节的 IPv4 地址改为内容为 0~15 的 16 字节 IPv6 接口地址。该数据结构如图 3-13 所示。

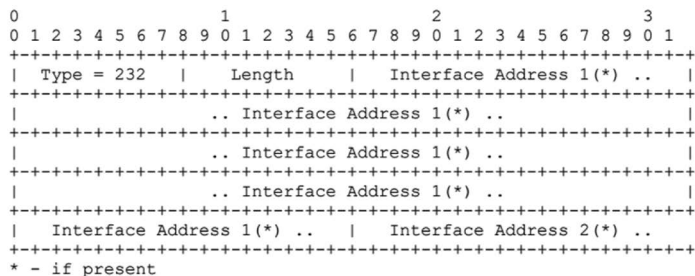


图 3-13 “IPv6 接口地址”TLV 格式

- Type: 8bit, TLV 类型, 此时值为 232(0xE8)。
- Length: 8bit, TLV 的 Value 部分长度。

- Interface Address: 128bit, IPv6 地址。

RFC5308 进一步限制了该 TLV 的语义,具体取决于它的通告位置。在不同的 PDU 中,这个字段的内容是不同的,在 Hello PDU 中,“接口地址 TLV”只能包含发送 hello 包的接口的 Link-local IPv6 地址;对于 LSP,“接口地址 TLV”只能包含中间系统(Intermediate System,IS)的 non-link-local IPv6 地址。

3. IPv6 NLPID

IPv6 网络层协议 ID(Network Layer Protocol ID,NLPID)的值为 142(0x8E)。为了支持 IPv6 路由的处理和计算,IS-IS 在 129 号 TLV 中新增了一个 NLPID。与 RFC1195 和 IPv4 一样,如果 IS 支持使用 IS-IS 的 IPv6 路由,那么在向外发布 IPv6 时它必须携带 NLPID 值。

3.2.3 可达性 TLV 扩展

1. TLV135 和 TLV235 支持的路由类型

下面分析使用 TLV135、TLV235 时 IPv4 支持的路由类型。

(1) L1 区域内路由:在 L1 LSP 的 TLV135 或 TLV235 中通告这些路由。UP/DOWN 位设置为 0。这些 IP 前缀直接连接到通告路由器。如果包含前缀属性标记子 TLV,则 X 标志和 R 标志都设置为 0。

(2) L1 外部路由:在 L1 LSP 的 TLV135 或 TLV235 中通告。UP/DOWN 位设置为 0。这些 IP 前缀来自其他协议,通常不直接连接到通告路由器。如果包含前缀属性标记子 TLV,则 X 标志设置为 1,R 标志设置为 0。

(3) L2 区域内路由:在 L2 LSP 的 TLV135 或 TLV235 中通告这些路由。UP/DOWN 位设置为 0。这些 IP 前缀直接连接到通告路由器。如果包含前缀属性标记子 TLV,则 X 标志和 R 标志都设置为 0。

(4) L1->L2 区域间路由:在 L2LSP 的 TLV135 或 TLV235 中通告这些路由。UP/DOWN 位设置为 0。这些 IP 前缀通过 L1 路由学习,并在 L1 最短路径优先(SPF)计算期间从 TLV135 或 TLV235 中的 L1 LSP 中通告的前缀导出。如果包含前缀属性标记子 TLV,则 R 标志设置为 1。

(5) L2->L2 区域间路由:在 L2 LSP 的 TLV135 或 TLV235 中通告这些路由。UP/DOWN 位设置为 1,但会被忽略,并将其视为设置为 0。这些 IP 前缀是从通常在另一个区域运行的另一个 IS-IS 实例中学习的。如果包含前缀属性标记子 TLV,则 X 标志设置为 1,R 标志设置为 0。

(6) L2 外部路由:在 L2 LSP 的 TLV135 或 TLV235 中通告这些路由。UP/DOWN 位设置为 0。这些 IP 前缀来自其他协议,通常不直接连接到通告路由器。如果包含前缀属性标记子 TLV,则 X 标志设置为 1,R 标志设置为 0。

(7) L2->L1 区域间路由:在 L1 LSP 的 TLV135 或 TLV235 中通告这些路由。UP/DOWN 位设置为 1。这些 IP 前缀通过 L2 路由学习,并在 L2 SPF 计算期间从 TLV135 或 TLV235 中通告的前缀导出。如果包含前缀属性标记子 TLV,则 R 标志设置为 1。

(8) L1->L1 区域间路由:在 L1 LSP 的 TLV135 或 TLV235 中通告这些路由。UP/DOWN 位设置为 1。这些 IP 前缀是从通常在另一个区域运行的另一个 IS-IS 实例中学习的。如果包含前缀属性标记子 TLV,则 X 标志设置为 1,R 标志设置为 0。

TLV135 或 TLV235 中通告的 IPv4 路由由优选遵循以下顺序(同一级别不分先后):

- (1) L1 区域内路由; L1 外部路由。

- (2) L2 区域内路由；L2 外部路由；L1->L2 区域间路由；L2->L2 区域间路由。
- (3) L2->L1 区域间路由；L1->L1 区域间路由。

2. TLV236 和 TLV237 支持的路由类型

下面分析使用 TLV 236、TLV 237 时 IPv6 支持的路由类型。

(1) L1 区域内路由：在 L1 LSP 的 TLV236 或 TLV237 中通告这些路由。UP/DOWN 位设置为 0。外部标志位设置为 0。这些 IPv6 前缀直接连接到通告路由器。如果包含前缀属性标记子 TLV,则 R 标志设置为 0。

(2) L1 外部路由：在 L1 LSP 的 TLV236 或 TLV237 中通告。UP/DOWN 位设置为 0。外部标志位设置为 1。这些 IPv6 前缀来自其他协议,通常不直接连接到通告路由器。如果包含前缀属性标记子 TLV,则 R 标志设置为 0。

(3) L2 区域内路由：在 L2 LSP 的 TLV236 或 TLV237 中通告这些路由。UP/DOWN 位设置为 0。外部标志位设置为 0。这些 IPv6 前缀直接连接到通告路由器。如果包含前缀属性标记子 TLV,则 R 标志设置为 0。

(4) L1->L2 区域间路由：在 L2 LSP 的 TLV236 或 TLV237 中通告这些路由。UP/DOWN 位设置为 0。外部标志位设置为 0。这些 IPv6 前缀通过 L1 路由学习,并在 L1 最短路径优先 (SPF) 计算期间从 TLV236 或 TLV237 中的 L1 LSP 中通告的前缀导出。如果包含前缀属性标记子 TLV,则 R 标志设置为 1。

(5) L2 外部路由：在 L2 LSP 的 TLV236 或 TLV237 中通告。UP/DOWN 位设置为 0。外部标志位设置为 1。这些 IPv6 前缀来自其他协议,通常不直接连接到通告路由器。如果包含前缀属性标记子 TLV,则 R 标志设置为 0。

(6) L1->L2 外部路由：在 L2 LSP 的 TLV236 或 TLV237 中通告这些路由。UP/DOWN 位设置为 0。外部标志位设置为 1。这些 IPv6 前缀通过 L1 路由学习,并在 L1 最短路径优先 (SPF) 计算期间从 TLV236 或 TLV237 中的 L1 LSP 中通告的 L1 外部路由导出。如果包含前缀属性标记子 TLV,则 R 标志设置为 1。

(7) L2->L2 区域间路由：在 L2 LSP 的 TLV236 或 TLV237 中通告这些路由。UP/DOWN 位设置为 1,但会被忽略,并将其视为设置为 0。外部标志位设置为 1。这些 IP 前缀是从通常在另一个区域运行的另一个 IS-IS 实例中学习的。如果包含前缀属性标记子 TLV,则 R 标志设置为 0。

(8) L2->L1 区域间路由：这些路由在 L1 LSP、TLV236 或 TLV237 中通告。UP/DOWN 位设置为 1。外部标志位设置为 0。这些 IPv6 前缀通过 L2 路由学习,并在 L2 SPF 计算期间从 TLV236 或 TLV237 中通告的前缀派生而来。如果包含前缀属性标记子 TLV,则 R 标志设置为 1。

(9) L2->L1 外部路由：在 L1 LSP 的 TLV236 或 TLV237 中通告这些路由。UP/DOWN 位设置为 1。外部标志位设置为 1。这些 IPv6 前缀通过 L2 路由学习,并在 L2 SPF 计算期间从 TLV236 或 TLV237 中通告的前缀派生而来。如果包含前缀属性标记子 TLV,则 R 标志设置为 1。

(10) L1->L1 区域间路由：在 L1 LSP 的 TLV236 或 TLV237 中通告这些路由。UP/DOWN 位设置为 1。外部标志位设置为 1。这些 IP 前缀是从通常在另一个区域运行的另一个 IS-IS 实例中学习的。如果包含前缀属性标记子 TLV,则 R 标志设置为 0。

TLV236 或 TLV237 中通告的 IPv6 路由由优选遵循以下顺序(同一级别不分先后)：

- (1) L1 区域内路由；L1 外部路由。

(2) L2 区域内路由; L2 外部路由; L1-> L2 区域间路由; L1-> L2 外部路由; L2-> L2 区域间路由。

(3) L2-> L1 区域间路由; L2-> L1 外部路由; L1-> L1 区域间路由。

3.3 BGP 域间路由协议



边界网关协议(Border Gateway Protocol,BGP)是一种用来在路由选择域之间交换网络层可达性信息(Network Layer Reachability Information,NLRI)的路由选择协议。由于不同的管理机构分别控制着各自的路由选择域,因此,路由选择域经常被称为自治系统(Autonomous System,AS)。现在的 Internet 是由多个自治系统相互连接构成的大网络,BGP 作为事实上的 Internet 外部路由协议标准,被广泛应用于 ISP(Internet Service Provider)之间。早期发布的 3 个版本分别是 BGP-1、BGP-2 和 BGP-3,主要用于交换 AS 之间的可达路由信息,构建 AS 域间的传播路径,防止路由环路产生,并在 AS 级别应用一些路由策略。当前使用的版本是 BGP-4。

3.3.1 BGP 概念与原理

1. BGP 基本概念

1) 自治系统

在互联网中,AS 是指在一个或多个实体管辖下的所有 IP 网络和路由器构成的网络,它们对互联网执行共同的路由策略。每个 AS 可以支持多种内部网关路由协议。一个 AS 内的所有网络都被分配同一个 AS 号,属于一个行政单位管辖。AS 号分为 2B 和 4B,其中 2B AS 号的范围为 1~65 535。随着时间推进,可分配的 2B AS 号已经濒临枯竭,需要将 AS 号的范围扩展为 4B,取值范围扩展为 1~4 294 967 295。4B AS 号还可以用 X.Y 的形式表示,其中,X 的取值范围为 1~65 535,Y 的取值范围为 0~65 535。

2) EBGp 和 IBGP(External BGP /Internal BGP)

内部网关协议(IGP)被设计用来在单一的路由选择域内提供可达性信息,并不适合提供域间路由选择功能,因此 BGP 作为优秀的域间路由协议得以产生并发展。

当今的网络通常使用以下类型的 IGP:

- 距离矢量协议,如 RIP(Routing Information Protocol)。
- 链路状态协议,如 OSPF(Open Shortest Path First,OSPF)协议和 IS-IS(Intermediate System to Intermediate System,IS-IS)协议。

虽然这些协议是为实现不同目的设计的,并且具有不同的行为特征,但是它们共同的目标是解决在一个路由选择域内的路径最优化问题。IGP 并不适合提供域间路由选择功能。例如,一种域间路由选择协议应该能够提供广泛的策略控制,因为不同的域通常需要不同的路由选择策略和管理策略。从一开始,BGP 就被设计成一种域间路由选择协议,其设计目标就是策略控制能力和可扩展性。所以,BGP 也不适合替代 IGP,因为它们适用的场景不同。

如图 3-14 所示,BGP 有两种运行方式;当 BGP 运行于同一 AS 内部时,称为内部边界网关协议(Internal BGP,IBGP);当 BGP 运行于不同 AS 之间时,称为外部边界网关协议(External BGP,EBGP)。

2. BGP 与 IGP 交互

BGP 与 IGP 在设备中使用不同的路由表,为了实现不同 AS 间的相互通信,BGP 需要与

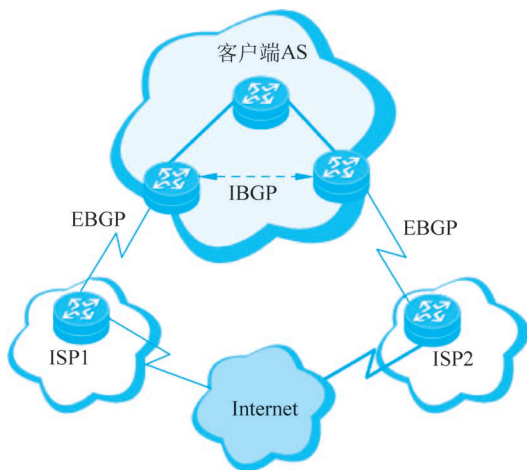


图 3-14 BGP 的运行方式

IGP 进行交互,即相互引入 BGP 路由表和 IGP 路由表。

BGP 引入 IGP 路由: BGP 协议本身不发现路由,因此需要将其他路由引入 BGP 路由表,实现 AS 间的路由互通。当一个 AS 需要将路由发布给其他 AS 时,AS 边界路由器会在 BGP 路由表中引入 IGP 的路由。为了更好地规划网络,BGP 在引入 IGP 的路由时,可以使用路由策略进行路由过滤和路由属性设置,也可以设置 MED 值指导 EBGP 对等体判断流量进入 AS 时的选路。当一个 AS 需要引入其他 AS 的路由时,AS 边界路由器会在 IGP 路由表中引入 BGP 的路由。为了避免大量 BGP 路由对 AS 内设备造成影响,当 IGP 引入 BGP 路由时,可以使用路由策略,进行路由过滤和路由属性设置。

BGP 引入路由时支持 Import 和 Network 两种方式:

- Import 方式是按协议类型,将 RIP 路由、OSPF 路由、IS-IS 路由等协议的路由引入 BGP 路由表中。为了保证引入的 IGP 路由的有效性,Import 方式还可以引入静态路由和直连路由。
- Network 方式是逐条将 IP 路由表中已经存在的路由引入 BGP 路由表中,该方式比 Import 方式更精确。

3. BGP 工作原理

BGP 是一种路径矢量协议,它使用 TCP 作为传输协议,使用 179 端口进行通信。以下从 4 个方面简要介绍 BGP 协议的基本工作原理。

1) BGP 报文交互中的角色

- Speaker: 发送 BGP 消息的路由器称为 BGP 发言者(Speaker),它接收或产生新的路由信息,并发布给其他 BGP Speaker。
- Peer: 相互交换消息的 BGP Speaker 之间互称对等体(Peer),若干相关的对等体可以构成对等体组(Peer Group)。

2) BGP 报文概述

BGP 的运行是通过报文驱动的,有 Open、Update、Notification、Keepalive 和 Route-refresh 五种报文类型。

- Open 报文: 它是 TCP 连接建立后发送的第一个报文,用于建立 BGP 对等体之间的连接关系。对等体在接收到 Open 报文并协商成功后,将发送 Keepalive 报文确认并保持连接的有效性。确认后,对等体间可以进行 Update、Notification、Keepalive 和

Route-refresh 报文的交换。

- Update 报文：用于在对等体之间交换路由信息。Update 报文可以发布多条属性相同的可达路由信息，也可以撤销多条不可达路由信息。
- Notification 报文：当 BGP 检测到错误状态时，就向对等体发出 Notification 报文，之后 BGP 连接会立即中断。
- Keepalive 报文：BGP 会周期性地向对等体发出 Keepalive 报文，用来保持连接的有效性。
- Route-refresh 报文：Route-refresh 报文用来请求对等体重新发送所有的可达路由信息。

3) BGP 处理过程

因为 BGP 的传输层协议是 TCP，所以在 BGP 对等体建立之前，对等体之间首先需要建立 TCP 连接。BGP 对等体间会通过 Open 报文协商相关参数，建立起 BGP 对等体关系。建立连接后，BGP 对等体之间交换整个 BGP 路由表。BGP 会发送 Keepalive 报文来维持对等体间的 BGP 连接，BGP 协议不会定期更新路由表，但当 BGP 路由发生变化时，会通过 Update 报文增量地更新路由表。当 BGP 检测到网络中的错误状态时（如收到错误报文时），BGP 会发送 Notification 报文进行报错，BGP 连接会随即中断。图 3-15 为 BGP 邻接建立过程图。

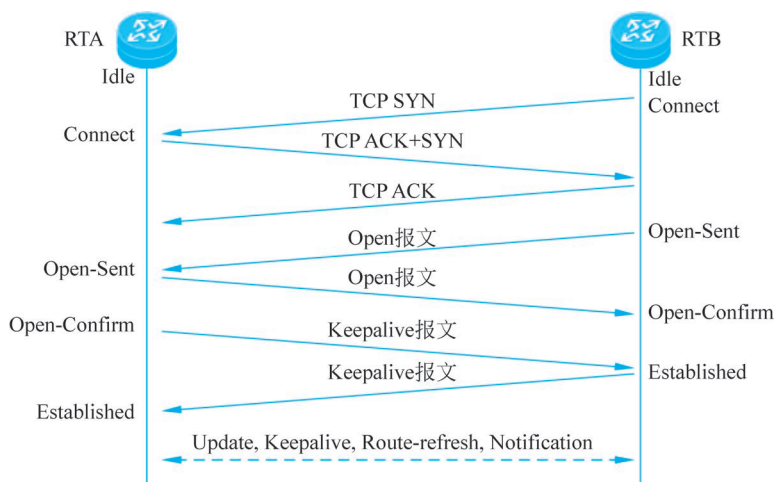


图 3-15 BGP 邻接建立过程图

BGP 对路由的处理过程如图 3-16 所示。BGP 路由来源包括从其他协议引入和从邻居学习两个部分，为了减少路由规模，可以对优选的 BGP 路由进行聚合。在引入路由、从邻居接收或发送路由的过程中，可以通过路由策略实现对路由的过滤，也可以修改路由的属性。

(1) 路由引入：BGP 自身不能发现路由，所以需要引入其他协议的路由（如 IGP 或者静态路由等）注入 BGP 路由表中，从而将这些路由在 AS 之内和 AS 之间传播。

(2) 路由选择：当到达同一目的地存在多条路由时，BGP 采取路由选择策略进行路由选择，例如，优选没有迭代到平滑下线处于延迟删除状态的 SRv6 TE-Policy 路由、在与 RPKI (Resource Public Key Infrastructure) 服务器进行连接的情景中，应用起源 AS 验证结果后的 BGP 路由优先级顺序为 Valid > Not Found > Invalid、优选没有误码的路由等。

(3) 路由聚合：在大规模的网络中，BGP 路由表十分庞大，使用路由聚合 (Routes Aggregation) 可以大大减小路由表的规模。路由聚合实际上是将多条路由合并的过程。这样 BGP 在向对等体通告路由时，可以只通告聚合后的路由，而不是通告所有的具体路由。

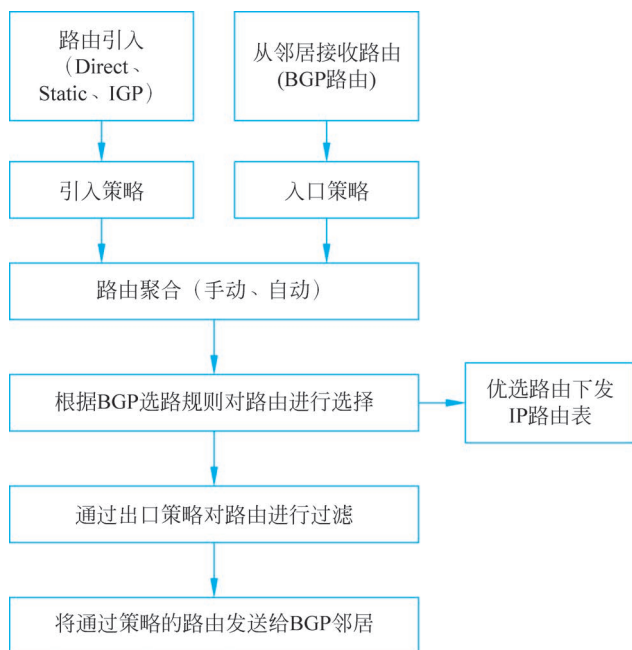


图 3-16 BGP 对路由的处理过程

(4) BGP 路由聚合支持两种方式,IPv4 支持自动聚合和手动聚合两种方式,而 IPv6 仅支持手动聚合。

- 自动聚合：对 BGP 引入的路由进行聚合。配置自动聚合后,对参加聚合的具体路由进行抑制。配置自动聚合后,BGP 将按照自然网段聚合路由(如 10.1.1.1/32 和 10.2.1.1/32 将聚合为 A 类地址 10.0.0.0/8),并只向对等体发送聚合后的路由。
- 手动聚合：对 BGP 本地路由进行聚合。手动聚合可以控制聚合路由的属性,以及决定是否发布具体路由。

(5) BGP 发布路由：BGP 发布路由时采用如下策略。

- 存在多条有效路由时,BGP Speaker 只将最优路由发布给对等体；
- BGP Speaker 从 EBGP 获得的路由会向它所有 BGP 对等体发布(包括 EBGP 对等体和 IBGP 对等体)；
- BGP Speaker 从 IBGP 获得的路由不向它的 IBGP 对等体发布；
- BGP Speaker 从 IBGP 获得的路由是否通告给它的 EBGP 对等体要依据 IGP 和 BGP 同步的情况确定；
- 连接一旦建立,BGP Speaker 将把自己可发布的 BGP 最优路由发布给新对等体。

4) BGP 协议有限状态机

BGP 协议的有限状态机共有 6 种状态,分别是 Idle、Connect、Active、Open-Sent、Open-Confirm 和 Established。在 BGP 对等体建立的过程中,通常可见的 3 个状态是 Idle、Active、Established。

(1) Idle 状态是 BGP 初始状态。在 Idle 状态下,BGP 拒绝邻居发送的连接请求。只有在收到本设备的 Start 事件后,BGP 才开始尝试和其他 BGP 对等体进行 TCP 连接,并转至 Connect 状态。

- Start 事件是由一个操作者配置一个 BGP 过程、重置一个已经存在的过程或者路由器软件重置 BGP 过程引起的。

- 任何状态中收到 Notification 报文或 TCP 拆除链路通知等 Error 事件后,BGP 都会转至 Idle 状态。

(2) 在 Connect 状态下,BGP 启动连接重传定时器(Connect Retry,默认为 32s),等待 TCP 完成连接。

- 若 TCP 连接成功,则 BGP 向对等体发送 Open 报文,并转至 Open-Sent 状态。
- 若 TCP 连接失败,则 BGP 转至 Active 状态。
- 若连接重传定时器超时,BGP 仍没有收到 BGP 对等体的响应,则 BGP 继续尝试和其他 BGP 对等体进行 TCP 连接,停留在 Connect 状态。
- 若发生其他事件(由系统或者操作人员启动的),则退回 Idle 状态。

(3) 在 Active 状态下,BGP 总是在试图建立 TCP 连接。

- 若 TCP 连接成功,那么 BGP 向对等体发送 Open 报文,关闭连接重传定时器,并转至 Open-Sent 状态。
- 若 TCP 连接失败,那么 BGP 停留在 Active 状态。
- 若连接重传定时器超时,BGP 仍没有收到 BGP 对等体的响应,那么 BGP 转至 Connect 状态。

(4) 在 Open-Sent 状态下,BGP 等待对等体的 Open 报文,并对收到的 Open 报文中的 AS 号、版本号、认证码等进行检查。

- 若收到的 Open 报文正确,那么 BGP 发送 Keepalive 报文,并转至 Open-Confirm 状态。
- 若发现收到的 Open 报文有错误,那么 BGP 发送 Notification 报文给对等体,并转至 Idle 状态。
- 在 Open-Confirm 状态下,BGP 等待 Keepalive 或 Notification 报文。若收到 Keepalive 报文,则转至 Established 状态;如果收到 Notification 报文,则转至 Idle 状态。

(5) 在 Established 状态下,BGP 可以和对等体交换 Update、Keepalive、Route-refresh 和 Notification 报文。

- 若收到正确的 Update 或 Keepalive 报文,则 BGP 认为对端处于正常运行状态,将保持 BGP 连接。
- 若收到错误的 Update 或 Keepalive 报文,则 BGP 发送 Notification 报文通知对端,并转至 Idle 状态。
- Route-refresh 报文不会改变 BGP 状态。若收到 Notification 报文,则 BGP 转至 Idle 状态。
- 若收到 TCP 拆链通知,则 BGP 断开连接,转至 Idle 状态。

BGP 对等体双方的状态必须都为 Established,BGP 邻居关系才能成立,双方通过 Update 报文交换路由信息,具体过程如图 3-17 所示。

3.3.2 BGP 消息格式

BGP 消息通过 TCP 进行可靠传输,端口号为 179,只有在完全收到后才进行消息的处理,最大的消息长度为 4096B,所有的实现必须支持这个最大值的限制,最小的消息为没有数据部分的 BGP 报头 19B。BGP 报文由 BGP 报文头和具体的报文内容两部分组成,其运行为消息驱动的,有 OPEN、UPDATE、NOTIFICATION、KEEPLIVE、REFRESH 共 5 种报文格式。

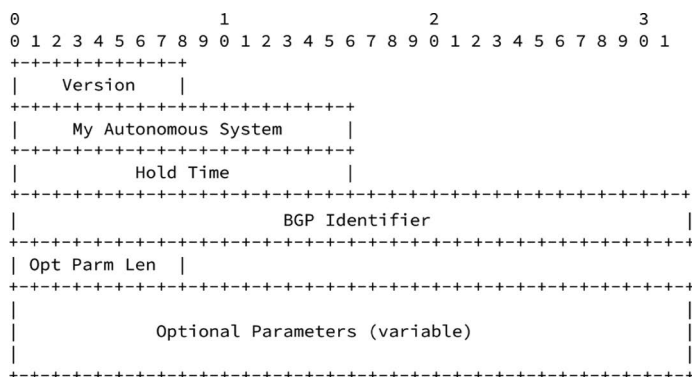


图 3-19 OPEN 消息的格式

其中,各字段的含义如下。

- Version: 1B,表示协议的版本号。
- My Autonomous System: 2B,发送者的 AS 域号。
- Hold Time: 2B,发送者自己设定的 hold time 值(单位:秒),用于协商 BGP 对等体之间保持建立连接关系,发送 KEEPLIVE 或 UPDATE 等消息的时间间隔。默认为 180s。
- BGP Identifier: 4B,发送者的 router id。
- Opt Parm Len: 1B,表示是可选参数的长度,为 0 表示没有可选参数。
- Optional Parameters: 可变长度。为 BGP 的可选参数列表,每个可选参数列表为 TLV(type length value)格式的单元。其中,Param. Type 占 1B,为可选参数的类型;Param. Length 占 1B,为 Param Value 包含可选参数的列表,每个参数被编码为 TLV 三元组。

3. UPDATE 消息格式

UPDATE 消息用于在 BGP 对等体之间传递路由信息,可以使用 UPDATE 消息中的信息构建描述各种 AS 关系图。通过规则,路由可以检测到路由环路和其他的异常情况并删除。UPDATE 消息用于通告可共享通用路径属性的可行路径到对等体,或撤销多个失效路由,其可以从服务中同时广播一条可行路由,撤出多条不可行路由。除了固定的 BGP 报文头外,UPDATE 消息的格式如图 3-20 所示。

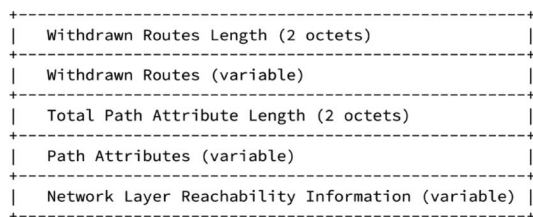


图 3-20 UPDATE 消息的格式

其中,各字段的含义如下。

- Withdrawn Routes Length: 2B,表示 Withdrawn Routes 部分的长度。
- Withdrawn Routes: 变长,包含要撤销的路由列表,列表中每个单元包含 1 字节 Length 域和可变长的 Prefix 区域。
- Total Path Attribute Length: 2B,表示 Path Attribute 的长度。
- Path Attributes: 变长,包含要更新的路由属性列表,每个属性单元为 TLV 格式。

UPDATE 消息的最小长度为 32B,其最多可以公告一组路径属性,但是可包含多个目的地。给定 UPDATE 消息中的所有路径属性适用于其 NLRI 字段中所携带的所有目的地。

4. KEEPALIVE 消息格式

BGP 不适用于任何基于 TCP 的保活机制,而是一定要在保活定时器过期前与对等体交换 KEEPALIVE 消息,才能确定是否通信可达。其消息之间最大的合理时间将是保持时间间隔的 1/3。KEEPALIVE 消息仅包含 19B 的报文头。

5. NOTIFICATION 消息格式

如果 BGP 报文头中的 Type 为 3,则该报文为 NOTIFICATION 消息,当检测到错误条件时,发送 NOTIFICATION 消息,发送完 NOTIFICATION 消息后 BGP 立即关闭。除固定大小的 BGP 头外,其格式如图 3-21 所示。

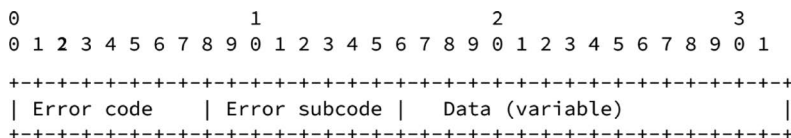


图 3-21 NOTIFICATION 消息格式

其中,各字段的含义如下。

- Error code: 1B,定义错误码的类型,非特定错误用 0 表示。
- Error subcode: 1B,指定错误细节编号,非特定错误细节编号用 0 表示。
- Data: 可变长度,指定错误数据内容。

6. REFRESH 消息格式

若 BGP 报文头中的报文类型为 5,则该报文为 REFRESH 消息。BGP 在建立邻居关系成功后才会发此报文,用来要求对等体重新发送指定地址族的路由信息,从而进行路由的更新。REFRESH 消息格式将一个 AFI-SAFI 组编码为如图 3-22 所示的三元组。



图 3-22 REFRESH 消息格式

其中,各字段的含义如下。

- AFI: 2B,表示地址族编号。
- Res.: 1B,所有位应该全为 0,在接收消息时此位被忽略。
- SAFI: 1B,表示子地址族编号。

3.3.3 BGP 路由优选规则

与 IGP 相比,BGP 最大的优势在于路径的选择策略非常丰富,通过调整 BGP 的路由属性来决定选路。在使用这些路由属性的时候应该考虑顺序和规则,尤其是当一台路由器有多条路径可到达目标的时候,BGP 需要根据下面的选路规则来优先选出一条最佳路径。BGP 选路的 14 条规则如下:

(1) 优选下一跳可达。下一跳可达是 BGP 选路规则中的第一条规则,如果 BGP 路由的下一跳 IP 地址不可达,那么该 BGP 路由将不会参与选路。

(2) 优选协议首选值(Pref_Val)数值最高的路由。协议首选值(Pref_Val)是设备的特有属性,在选路规则中位列第一位,因此最优先比较,该属性仅针对本路由器有意义,不会传递给任何 BGP 邻居。协议首选值越大越优先,默认首选值为 0,不同厂商有自己的实现方式。

(3) 优选本地优先级(Local_Pref)数值最高的路由。本地优先级属性在整个 AS 内传递,

不会传递到 AS 之外,默认值越大优先级越高,一般用作 AS 内路由器选择一个最优出口去往外部。如果路由没有设置本地优先级,那么 BGP 选路时将该路由按默认的本地优先级 100 来处理(BGP 优先级的默认值可以修改)。该属性只可以传递给 IBGP 邻居,但可以根据 EBGp 和 IBGP 邻居接收的路由修改本地优先级值然后加载到 BGP 路由表,修改之后该路由的本地优先级值同样只能继续传递给 IBGP 邻居。

(4) 优选本地生成的路由。该规则是指在当前路由器产生的 BGP 路由中,本地产生的路由要优于从 BGP 邻居学到的路由,本地产生的路由分为本地生成的聚合路由和 IP 路由注入的路由。其中聚合路由要优于不聚合的路由,聚合可分为手动聚合和自动聚合,前者要优于后者;通过 network 命令通告引入 BGP 的路由要优于通过 import 方式引入 BGP 的路由;最后是从邻居学到的路由(即优先顺序依次为:手动聚合路由>自动聚合路由>network 命令通告的路由>import-route 命令引入的路由>从对等体学习的路由)。BGP 中的路由有以上几种产生方式,如果同一个目标前缀的路由同时通过以上方式进入 BGP,那么将根据上述优先顺序进行裁决。

(5) 优选 AS 路径(AS_PATH)最短的路由。AS_PATH 属性是记录达到目标网络的 AS 路径列表,与距离矢量协议中的 hop 概念类似,AS-PATH 长度短的路由优先。

(6) 依次优选 Origin 类型为 IGP、EGP、incomplete 的路由。Origin 为 BGP 的起源属性,指 BGP 路由的起源,BGP 路由都会携带一个 Origin 属性。如果该路由是通过 network 的方式产生的,那么 Origin 类型为 IGP(标识为 i);如果是通过 EGP 协议学习到的路由,那么 Origin 类型为 EGP(标识为 e);如果路由信息是从其他渠道学习到的,如将外部路由引入进 BGP 的路由,那么 Origin 类型为 incomplete(标识为“?”),优先级顺序为 $i > e > ?$ 。该规则比较的是路由的注入方式(network 或 import)。

(7) 优选 MED(Multi Exit Discriminator)值最低的路由。该规则默认比较来自相同邻居的路由 MED 值,数值越小的路由越优先。如果是来自不同邻居 AS 的路由,MED 属性默认不参与比较。

(8) 依次优选 EBGp 路由、IBGP 路由。如果路由分别通过 EBGp 和 IBGP 同时学习,在其他规则都一样的情况下优先选择 EBGp 对等体。原因为 EBGp 连接外部 AS,而 IBGP 连接内部的 AS,路由器认为通过 EBGp 学到的路由必然是来自外部的 AS,因此直接选择 EBGp 对等体到达外部比穿越整个 AS 再到达外部要更加优先。

(9) 优选到 BGP 下一跳 IGP 度量值(Metric)最小的路由。BGP 的路由下一跳地址是通过 IGP 协议学到的,根据路由表可以计算到下一跳的度量值,值越小的越优先。

(10) 在上述规则一致的情况下,可以使用命令配置负载均衡。

(11) 优选最先学习到的路由。如果从多个 EBGp 邻居收到同样的 BGP 路由,最优选最先收到的,对 IBGP 无效同样对联邦 EBGp 也无效。

(12) 优选 Cluster_List 最短的路由:Cluster_List 为路由反射器中的属性,是由路由反射器添加的,将 Cluster_ID 添加到 Cluster_List 中,用于记录被反射的 BGP 路由在 AS 内经过的 Cluster 路径,类似 AS_PATH 属性,Cluster_List 越短的路由越优先;如果参与比较的路由没有 Cluster_List,则越过该规则直接比较后面的规则,如果某条路由没有 Cluster_List,而其他路由有 Cluster_List,则没有 Cluster_List 属性的路由优先。

(13) 优选 Router_ID 最小的设备发布的路由。Router_ID 最小的邻居通告的路由最优先。如果路由携带 Originator_ID 属性,则选路过程中将比较 Originator_ID 的大小,不再比较 Router_ID,其中 Originator_ID 最小的路由最优;如果参与比较的路由 Originator_ID 一样,

也不再比较 Router_ID,则直接越过该规则。

(14) 优选从具有最小 IP Address 的对等体学来的路由,即比较邻居的 IP 地址,最小的最优先。

3.3.4 BGP 错误处理

当检测到这里描述的任何条件时,一条指定错误码、错误子码、数据段的 NOTIFICATION 消息应该被发送,且关闭 BGP 连接(除非明确声明不会发送 NOTIFICATION 消息,BGP 连接不要关闭)。如果没有错误子代码指定,则必须使用零。

BGP 连接已关闭具体包括 TCP 连接已关闭,相关的 Adj-RIB-In 已被清除,该 BGP 连接的全部资源已被释放,与远程对等体相关联的 Loc-RIB 条目被标记为无效。本地系统将重新计算被标记为无效路由的目的地的最佳路由。无效路由被从系统中删除之前,它宣告给对等体,撤销标记为无效的路由,或在无效的路由将从系统中删除前生成新的最佳路由。

除非明确指定,否则指示错误的消息发送的 NOTIFICATION 的数据字段为空。

1. BGP 错误处理汇总

下面分析处理 BGP 消息时检测到错误时要采取的操作。BGP 中的错误码和错误子码如表 3-1 所示。

表 3-1 BGP 中的错误码和错误子码

错 误 码	错 误 子 码
1: 消息头错误	1 连接未同步 2 错误的消息长度 3 错误消息类型
2: OPEN 消息错误	1 不支持版本号 2 错误的对等 AS 3 错误的 BGP 标识符 4 不支持的可选参数 5 认证失败 6 不可接受的保持时间 7 不支持的能力
3: UPDATE 消息错误	1 不支持版本号 2 错误的对等 AS 3 错误的 BGP 标识符 4 不支持的可选参数 5 认证失败 6 不可接受的保持时间 7 不支持的能力
4: 保持定时器超时	0 没有特别意义的错误子码定义
5: 有限状态机错误	0 没有特别的错误子码定义
6: 终止	1 前缀超过最大值 2 管理关闭 3 删除邻居 4 管理重置 5 连接失败 6 其他配置改变 7 连接冲突 8 资源短缺 9 BFD 断开连接

2. 消息头错误处理

处理消息头时检测到的所有错误,必须通过发送带有消息头错误代码的 NOTIFICATION 消息来通告。消息头的 Marker 字段的期望值全部为 1。若消息头的 Marker 字段不如预期的那样,那么发生同步错误,并且错误子代码必须设置为连接不同步。

如果下面至少有一个为真,那么错误子代码必须设置为消息长度错误。数据字段必须包含错误的长度字段:

- 如果消息头的长度字段小于 19 或大于 4096;
- 如果 OPEN 消息的长度字段小于最小值 OPEN 消息的长度;
- 如果 UPDATE 消息的 Length 字段小于 UPDATE 消息的最小长度;
- 如果 KEEPALIVE 消息的 Length 字段不等于 19;
- 如果 NOTIFICATION 消息的 Length 字段小于 NOTIFICATION 消息的最小长度。

若消息头的类型字段无法识别,则错误子代码必须设置为消息类型错误。数据字段必须包含错误的类型字段

3. OPEN 消息错误处理

处理 OPEN 消息时检测到的所有错误,必须通过发送带有 OPEN 错误代码的 NOTIFICATION 消息指示 OPEN 消息错误。错误子代码详细阐述了具体错误内容的性质。

- 若接收到的 OPEN 消息的 version 字段中版本号不受支持,则必须将 NOTIFICATION 消息错误子代码设置为“不支持的版本号”。NOTIFICATION 消息的 DATA 字段是无符号的 2 字节整数,如果是较大值,则表示本地支持的版本小于从远端 BGP 对等体接收到的 OPEN 消息标识的版本号;如果是较小值,本地支持的版本号大于远端 BGP 对等体的版本号,那么 DATA 字段的值表示本地支持的最小版本号。
- 若 OPEN 消息的 AS 字段不可接受,那么错误子代码必须设置为 Bad Peer AS。判定可接受的 AS 号超出了协议范围。
- 若 OPEN 消息的保持时间字段是不可接受的,那么错误子码必须设置为不可接受的保持时间。一种实现是必须拒绝一秒或两秒的保持时间值。另一种实现可能是拒绝任何建议的保持时间。还有一种实现可能是接受协商的保持时间。
- 若 OPEN 消息的 BGP 标识符字段使用不正确的语法,错误子码必须设置为 Bad BGP Identifier。语法正确性意味着 BGP 标识符字段表示一个有效的单播 IP 主机地址。
- 若 OPEN 消息中的可选参数之一不识别,则错误子代码必须设置为不支持可选参数。
- 若 OPEN 消息中的可选参数之一被识别,但是格式错误,那么错误代码必须设置为 0。

4. NOTIFICATION 消息错误处理

若一个对等体发送一个 NOTIFICATION 消息,并且该消息的接收者检测到该消息中有错误,则接收方不能使用 NOTIFICATION 消息将此错误报告回对等体。任何这样的错误(如无法识别的错误代码或错误子代码)都应该被注意,在本地记录,并提醒对等体的管理机构注意。

5. 保持定时器超时错误处理

若系统在 OPEN 消息保持时间字段指定的周期内没有收到连续的 KEEPALIVE、UPDATE 和/或 NOTIFICATION 消息,然后发送错误代码为“保持定时器超时”的 NOTIFICATION 消息,并关闭 BGP 连接。

6. 有限状态机错误处理

BGP 有限状态机检测到的任何错误时,将通过发送错误代码设置为“有限状态机错误”的 NOTIFICATION 消息来通告。

7. 终止

BGP speaker 可以选择在任何给定时间,发送设置错误代码为 Cease 的 NOTIFICATION 消息关闭 BGP 连接。然而,如果存在致命错误时,不允许发送错误代码为 Cease 的 NOTIFICATION 消息。

BGP speaker 可以通过配置限制从邻居接受地址前缀数量的上限。当达到上限时, speaker 在本地配置的控制下,丢弃来自邻居的新地址前缀(同时维护 BGP 与邻居的连接),或终止其与邻居的 BGP 连接。如果 BGP speaker 决定终止其与邻居的 BGP 连接(因为从邻居接收的地址前缀的数量超过本地配置的上限),那么 speaker 必须向邻居发送一个错误代码为 Cease 的 NOTIFICATION 消息。speaker 也可以本地记录相关日志。

3.4 面向 SRv6 的 BGP 协议扩展

SRv6 是指在 IPv6 数据平面实例化的分段路由。BGP 用于从出接口 PE(Provider Edge) 节点向入接口 PE 节点通告特定服务前缀的可达性。

基于 SRv6 的 BGP 业务是指以 BGP 作为控制平面,SRv6 作为数据平面执行 Layer3 和 Layer2 封装的数据转发业务。本节主要分析面向 SRv6、基于 BGP 的 Ethernet VPN(EVPN),同时介绍了一个新的 BGP-LS 协议标识,以及新的 BGP-LS 节点和链接描述符 TLV,以促进广播中的 BGP-LS Link NLRI 来表示 BGP 对等拓扑。此外,还介绍了在同一 BGP-LS Link NLRI 中发布的 BGP 对等段(即 Peer-Node SID 和 Peer-Set SID)的 BGP-LS 属性 TLV。

3.4.1 基于 BGP 的 EVPN 控制协议

本节主要介绍构建 EVPN(Ethernet VPN)Overlay 路由层的可扩展方法,具体的路由类型及其含义如表 3-2 所示。

表 3-2 路由类型及其含义

类型(Type)	含 义
1	以太网自动发现路由[Ethernet Auto-Discovery (A-D) route]
2	MAC/IP 广播路由(MAC/IP Advertisement route)
3	兼容性多播以太网标记路由(Inclusive Multicast Ethernet Tag route)
4	以太网段路由(Ethernet Segment route)
5	IP 前缀路由(IP Prefix route)
6	选择性多播以太网标记路由(Selective Multicast Ethernet Tag route)
7	多播成员身份报告同步路由(Multicast Membership Report Synch route)
8	组播成员离开同步路由(Multicast Leave Synch route)

为了支持基于 SRv6 的 EVPN 覆盖,一个或多个 SRv6 服务 SID 使用路由类型 1、2、3 和 5。每个路由类型的 SRv6 服务 SID 在 BGP Prefix-SID 属性中的 SRv6 L3/L2 Service TLV 中发布。设计该 TLV 有两个目的:第一,表示 BGP 出口设备支持 SRv6 Overlay,接收此路由的 BGP 入口 PE 必须执行 IPv6 封装,并在需要时插入 SRH;第二,表示封装中使用的 Service SID 的值。

SRv6 Service SID 可在出口 PE 的 AS 内路由,提供入接口 PE 和出接口 PE 之间的可达性,同时也对 SRv6 端点行为进行编码。

当 SRv6 服务的转向基于最短路径转发(如 Best effort 或 IGP Flex 算法)时,入口 PE 将 IPv4 或 IPv6 用户数据包封装在外部 IPv6 报头中(使用[RFC8986]中指定的 H. Encaps 或 H. Encaps. Red 风格),其中目标地址是与相关 BGP 路由更新关联的 SRv6 Service SID。因此,在考虑对接收前缀执行 BGP 最佳路径计算之前,入口 PE 必须对 SRv6 Service SID 执行可解析性检测。如果可以通过多个转发表访问 SRv6 SID,则将使用本地策略来确定要使用哪个表。若入口 PE 存在一个局部策略,允许备用重路由机制到达入口 PE,则 SRv6 Service SID 可解析性检测的结果可以被忽略。

对于路由器自身而言,出口 PE 将 BGP 的下一跳设置为它的一个 IPv6 地址。该地址可以 SRv6 Locator 覆盖,同时 SRv6 Service SID 也从中分配。BGP 下一跳用于基于现有的 BGP 处理流程跟踪出口 PE 的可达性。

当在入口 PE 接收到的 BGP 路由用 Color Extended Community 字段着色,并且当存在有效的 SRv6 策略可用时,将按照中分段路由策略协议规范所述执行服务流程的指导。当入口 PE 确定,Service SID 与 SR 策略段列表中的最后一个 SRv6 SID(出口 PE 的)属于同一个 SRv6 Locator 时,它将在引导服务流程时排除最后一个 SRv6 SID。例如,与 SID 列表< S1, S2, S3 >关联的 SRv6 策略的有效段列表为< S1, S2, S3-Service-SID >。

1. 以太网自动发现路由(Ethernet Auto-Discovery (A-D) route)

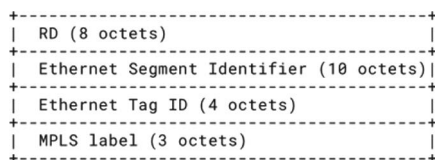


图 3-23 EVPN 路由类型 1 的编码

以太网自动发现路由为 1 号路由类型,可用于实现水平分割过滤、快速收敛和混叠。EVPN 路由类型 1 也用于 EVPN-VPWS 和 EVPN-flexible cross-connect,主要用于通告点对点的 Service ID。其编码如图 3-23 所示。

1) Ethernet A-D per ES Route

当 ESI 过滤方法与编码的转置方案被一起使用时,ESI 标签扩展社区的 24 位 ESI 标签字段携带 SRv6 SID 的全部或部分参数;否则,它在高 20 位中设置为 Implicit NULL(即 0x000030)。在任何一种情况下,该值都被设置为 24 位。当使用换位方案时,换位长度应小于或等于 24,并小于或等于 AL。

在 BGP Prefix-SID 属性中的 SRv6 L2 Service TLV 中包含的 Service SID 与 A-D 路由一起发布。SRv6 端点行为应使用 End. DT2M。当使用 ESI 过滤方法时,Service SID 用于发出 Arg. FE2 SID 参数以匹配 END. DT2M 行为。

2) Ethernet A-D per EVI Route

面向 SRv6 的 per-EVI(EVPN Instance)路由的编码类似于 RFC7432 和 RFC8214 中描述的内容,但有以下变化。

MPLS Label: 使用编码转换方案时,24 位标签字段携带 SRv6 SID 的功能部分,否则设置为 20 位(即 0x000030)。在任何情况下,该值都被设置为 24 位。当使用换位方案时,换位长度应小于或等于 24,并小于或等于 FL。

在 BGP Prefix-SID 属性中的 SRv6 L2 Service TLV 中包含的 Service SID 与 A-D 路由一起发布。SRv6 SID 的端点行为是 End. DX2、End. DX2V 或 End. DT2U 之一。

2. MAC/IP 通告路由(MAC/IP Advertisement route)

EVPN 路由类型 2 用于通过 MP-BGP 协议向给定的 EVPN 实例中的所有其他 PE,通告

单播流量 MAC 和 IP 地址的可达性。图 3-24 展示了 EVPN 路由类型 2 的报文格式。

MPLS Label1: 与 SRv6 L2 Service TLV 相关联。当使用编码转换方案时,此 24 位字段携带 SRv6 SID 的功能部分的全部或部分;否则,高 20 位将隐式设置为 NULL(即 0x000030)。当使用换位方案时,换位长度应小于或等于 24,并小于或等于 FL。

MPLS Label2: 与 SRv6 L3 Service TLV 相关联。当使用编码转换方案时,此 24 位字段携带 SRv6 SID 的功能部分的全部或部分;否则,高 20 位将隐式设置为 NULL(即 0x000030)。当使用换位方案时,换位长度应小于或等于 24,并小于或等于 FL。

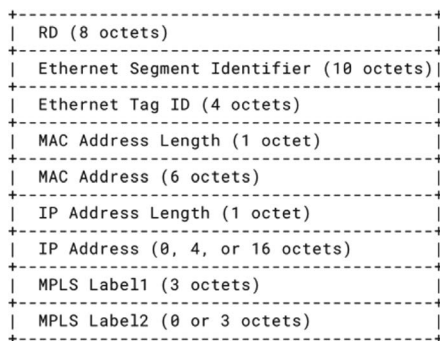


图 3-24 EVPN 路由类型 2 的报文格式

包含在 SRv6 L2 Service TLV 中和 SRv6 L3 Service TLV(可选)中的 Service SID 与 MAC/IP 通告路由一起发布。下面介绍不同路由类型 2 通告。

1) MAC/IP Advertisement Route with MAC Only

Service SID 包含于 BGP Prefix-SID 属性中的 SRv6 L2 Service TLV 中,它们将与路由一起发布。该 SRv6 SID 的端点行为是 End. DX2 或 End. DT2U 中的一种。

2) MAC/IP Advertisement Route with MAC+IP

L2 Service SID 包含在 BGP Prefix-SID 属性中的 SRv6 L2 Service TLV 中,将与路由一起发布。同时,包含在 BGP Prefix-SID 属性中的 SRv6 L3 Service TLV 中的 L3 Service SID 也会随路由一起发布。对于 L2 Service SID,SRv6 端点行为是 End. DX2 或 End. DT2U 中的一种,对于 L3 Service SID,SRv6 端点行为是 End. DT46、End. DT4、End. DT6、End. DX4 或 End. DX6 中的一种。

3. 兼容性多播以太网标记路由(Inclusive Multicast Ethernet Tag route)

EVPN 路由类型 3 用于通过 MP-BGP 向给定 EVPN 实例中的所有其他 PE 发布组播流量可达性信息。EVPN 路由类型 3 的报文格式如图 3-25 所示。

P-PMSI(组播服务接口)隧道属性用于识别发送广播、未知单播或组播(BUM)流量的供应商隧道(P-tunnel)。PMSI 隧道属性的格式的编码如图 3-26 所示。

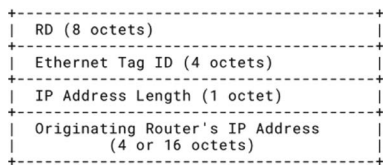


图 3-25 EVPN 路由类型 3 的报文格式

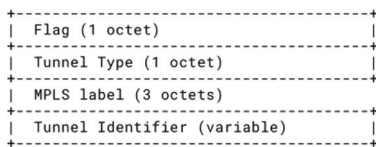


图 3-26 PMSI 隧道属性在 SRv6 实现上的编码

其中,各字段的含义如下。

- Flag: 此字段的值为 0。
- MPLS label: 当使用入口复制时,该 24 位字段携带 SRv6 SID 功能部分的全部或部分,并使用编码的转换方案;否则,按 RFC6514 中的定义设置。当使用换位方案时,换位长度应小于或等于 24,且应小于或等于 FL。
- Tunnel Identifier: 该字段是出口 PE 的 IP 地址。

包含在 BGP Prefix-SID 属性中 SRv6 L2 Service TLV 中的 Service SID 将与路由一起发布。

4. 以太网段路由(Ethernet Segment route)

EVPN 路由类型 4 的编码如图 3-27 所示。

BGP Prefix-SID 属性中的 SRv6 Service TLV 不会随此路由一起发布。

5. IP 前缀路由(IP Prefix route)

EVPN 路由类型 5 用于通过 MP-BGP 协议向给定的 EVPN 实例中的所有其他 PE 通告 IP 地址的可达性。该 IP 地址可以包括一个主机 IP 前缀或任何特定的子网。EVPN 路由类型 5 的编码如图 3-28 所示。

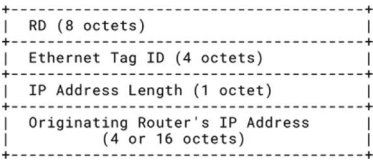


图 3-27 EVPN 路由类型 4 的编码

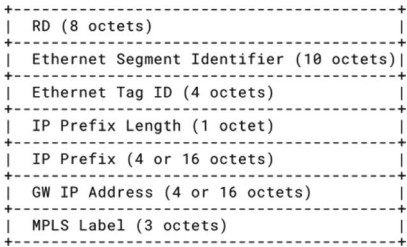


图 3-28 EVPN 路由类型 5 的编码

MPLS Label: 当使用编码转换方案时,此 24 位字段携带 SRv6 SID 的功能部分的全部或部分;否则,高 20 位将隐式设置为 NULL(即 0x000030)。当使用换位方案时,换位长度应小于或等于 24,并小于或等于 FL。

SRv6 Service SID 被编码为 SRv6 L3 Service TLV 的一部分。SRv6 的端点行为是 End, DT4、End, DT6、End, DT46、End, DX4 或 End, DX6 中的一种。

6. EVPN 多播路由(类型 6、类型 7 和类型 8)

这些路由不需要 SRv6 服务 TLV 的通告,与 EVPN 路由类型 4 类似,BGP 下一跳等于出口 PE 的 IPv6 地址。

3.4.2 BGP 对等段 BGP-LS 属性

首先介绍与以下 BGP 对等段 SID 对应的 BGP-LS 属性。

- 对等节点段标识符(PeerNode SID)。
- 对邻接段标识符(PeerAdj SID)。
- 对等集段标识符(PeerSet SID)。

以下新的 BGP-LS 链路属性 TLV 被定义为使用 BGP-LS Link NLRI,用于通告 BGP 对等的 SID。表 3-3 展示了 BGP-LS TLV 类型与其描述。

表 3-3 BGP-LS TLV 类型与其描述

TLV 类型	描 述
1101	对等节点段标识符(PeerNode SID)
1102	对邻接段标识符(PeerAdj SID)
1103	对等集段标识符(PeerSet SID)

对等节点段标识符、对邻接段标识符和对等集段标识符 SID 的格式如图 3-29 所示。

- 类型(Type): 1101、1102、1103,已在表 3-3 中列出。
- 长度(Length): 根据编码是 SID 索引还是标签,有效取值是 7 或 8。
- 标志(Flags): 一个具有如图 3-30 所定义的 1 字节标志。

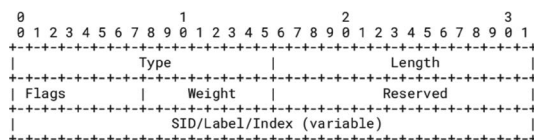


图 3-29 BGP 对等段 SID TLV 格式

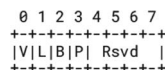


图 3-30 对等 SID TLV 标志格式

① V: 值标志。如果置 1, 则 SID 携带一个标签值。默认情况下, 该标志置 1。

② L: 本地标志。如果置 1, 则 SID 所携带的值/索引具有局部显著性。默认情况下, 该标志置 1。

③ B: 备份标志。如果置 1, SID 引用适合使用快速重路由(FRR)进行保护的路径。备份转发路径的计算及其与 BGP 对等 SID 转发条目的关联是特定于具体实现的。RFC9087 讨论了为 BGP 对等段识别备份路径的一些方法。

④ P: 持久标志。如果置 1, SID 被持续分配, 即 SID 值在路由器重新启动和会话/接口保护期间保持一致。

⑤ Rsvd: 保留为将来使用, 发送时置为 0, 接收时忽略。

- 权重(Weight): 8 字节。该值表示用于负载均衡的 SID 的权重。

- SID/Label/Index。根据 TLV 长度和 V-标志和 L-标志设置, 它包含:

① 一个 3 字节的本地标签, 其中最右边的 20 位用于编码标签值。在这种情况下, 将设置 V-标志和 L-标志。

② 一个 4 字节路由器索引, 定义段路由全局块(SRGB)的偏移量。

PeerNodeSID、PeerAdj SID 和 PeerSet SID Sub-TLV 的值在路由器重新启动后也是不变的。

1. PeerNode SID 发布

PeerNode SID TLV 包含与 BGP 对等节点关联的 SID, 该节点由特定的 BGP-LS Link NLRI 描述。

(1) 本地节点描述符包括:

- 启用 BGP-EPE 的出口 PE 的本地 BGP 路由器 ID(TLV 516);
- 本地 ASN(TLV 512)。

(2) 远程节点描述符包括:

- 对等 BGP 路由器 ID(TLV 516)(即在 BGP 会话中使用的对等 BGP ID);
- 对等体 ASN(TLV 512)。

(3) 链路描述符包括使用 RFC7752 中定义的 TLV 编码的 BGP 会话所使用的地址:

- IPv4 接口地址(TLV 259)包含 BGP 会话 IPv4 本地地址;
- IPv4 邻居地址(TLV 260)包含 BGP 会话 IPv4 对等地址;
- IPv6 接口地址(TLV 261)包含 BGP 会话 IPv6 本地地址;
- IPv6 邻居地址(TLV 262)包含 BGP 会话 IPv6 对等地址。

(4) 链路属性 TLV 包括图 3-29 中定义的对等段 SID TLV。

2. PeerAdj SID 发布

PeerAdj SID TLV 包括一个与到 BGP 对等体的底层链路关联的 SID, 该 SID 由特定的 BGP-LS Link NLRI 描述。

(1) 本地节点描述符包括:

- 启用 BGP-EPE 的 Egress PE 的本地 BGP 路由器 ID(TLV 516);

- 本地 ASN(TLV 512)。

(2) 远程节点描述符包括：

- 对等 BGP 路由器 ID(TLV 516)(即在 BGP 会话中使用的对等 BGP ID)；
- 对等体 ASN(TLV 512)。

(3) 链路描述符必须包括以下 TLV,见 RFC7752 中的定义：

链路本地/远程标识符(TLV 258)包含 4 字节链路本地标识符,后跟 4 字节链路远程标识符。当链路远程标识符未知时,默认使用值 0。

(4) 附加的链路描述符 TLV,也包括在 RFC7752 中的定义 TLV,用于描述与 BGP 路由器之间链路的对应地址：

- IPv4 接口地址(Sub-TLV 259)包含用来建立 BGP 会话的本地接口的地址。
- IPv6 接口地址(Sub-TLV 261)包含用来建立 BGP 会话的本地接口的地址。
- IPv4 邻居地址(Sub-TLV 260)包含 BGP 会话所使用的对等接口的 IPv4 地址。
- IPv6 邻居地址(Sub-TLV 262)包含 BGP 会话所使用的对等接口的 IPv6 地址。

3. PeerSet SID 发布

PeerSet SID TLV 包括一个在 BGP 对等体之间共享的 SID,或由特定的 BGP-LS Link NLRI 描述的底层链路。

3.5 路由操作系统



FRR(Free Range Routing)是一款全功能、高性能的开源软件 IP 路由套件,实现了所有标准路由协议及其扩展,如 BGP、RIP、OSPF、IS-IS 等。它主要用 C 语言编写,能够轻松处理完整的互联网路由表,并适用于多种硬件。FRR 基于 GPLv2 许可协议分发,以类似于 Linux 内核的方式开发。作为 Quagga 项目的延续,FRR 旨在进一步改进 Quagga 的基础,以创建最佳的路由协议堆栈。FRR 的架构包括多个守护进程,每个主要协议都在自己的守护进程中实现,并与中间人守护进程 Zebra 通信,后者负责协调路由决策和与数据平面的通信。

3.5.1 FRR 概述

FRR 是一个功能齐全、高性能、开源的软件 IP 路由套件。它提供了广泛的功能和协议支持,使其成为网络路由方面的强大工具。

FRR 实现了包括 BGP、RIP、OSPF、IS-IS 等标准路由协议,以及它们的许多扩展。无论是基本的路由协议还是复杂的扩展功能,FRR 都能满足需求。用户可以根据需要选择和配置适合自己网络环境的路由协议。

作为一个主要用 C 编写的高性能套件,FRR 能够轻松处理完整的互联网路由表,适用于从廉价单板计算机(SBC)到商业级路由器的各种硬件。它的性能优势使得它成为许多公司、大学、研究实验室和政府机构在生产环境中的首选。

FRR 是以 GPLv2 许可协议开源分发的,其开发模型类似于 Linux 内核。这意味着任何人都可以参与其中,贡献新功能、修复错误、改进工具和文档等。FRR 是 Quagga 路由套件的一个分支项目,它在 Quagga 的基础上进行了扩展和改进。

作为 IP 路由套件,FRR 在网络堆栈中扮演着重要的角色。它与其他路由器交换路由信息,做出路由和策略决策,并将这些决策传播给其他层。在实际应用中,FRR 能够将路由决策安装到操作系统内核中,使得内核网络堆栈能够根据这些决策进行转发。

除了支持动态路由功能,FRR 还提供了全面的 L3 配置选项,包括静态路由、地址管理、路由器通告等。此外,它还具备一些轻量级的 L2 功能,但其具体支持取决于不同的系统。

这使得 FRR 适用于各种部署场景,从小型家庭网络中的静态路由到运行完整互联网路由表的互联网路由器。

FRR 可在包括 Linux 和 BSD 在内的所有现代 UNIX 操作系统上运行。尽管不同平台上的功能支持可能有所差异,但它在各个平台上都能提供强大的路由功能。

3.5.2 进程架构

FRR 提供了一组不同功能的守护进程,其内部架构采用模块化和多线程设计,以实现各种路由协议和功能,并保证网络的可靠性和性能。它的设计模式和线程管理使得它能够灵活适应不同的网络环境和需求。FRR 守护进程中使用的基本模式是事件循环。一些守护进程使用内核线程。在这些守护进程中,每个内核线程运行自己的事件循环。事件循环是线程安全的,并允许其所属线程以外的线程进行事件调度。

1. 事件架构

核心事件系统在 lib/event.c 和 lib/lib/event.h 中实现。主结构是 struct event_loop,以后称之为线程管理器。线程管理器(Threadmaster)是一个全局状态对象或上下文,它包含所有当前等待执行的任务以及已经执行的任务的统计信息。事件系统是通过向该数据结构添加任务,然后调用一个函数来检索下一个要执行的任务来驱动的。在初始化时,守护进程通常会创建一个线程管理器,添加一组初始任务,然后运行一个循环来获取每个任务并执行。

要使用事件系统,在创建线程管理器之后,程序会添加一组初始任务。当这些任务执行时,它们会添加更多的任务,这些任务将在未来的某个时间点执行。这个任务序列驱动着程序的生命周期。当没有更多的任务可用时,程序结束。通常在启动时添加的第一个任务是 VTYSH 的 I/O 任务以及通信双方或 IPC 所需的任何网络套接字。

为了获取下一个任务,程序调用 event_fetch()函数,该函数在内部根据基本优先级逻辑计算下一个要执行的任务。

事件(类型 EVENT_EVENT)以最高优先级执行,其次是过期计时器和最后的 I/O 任务(类型 EVENT_READ 和 EVENT_WRITE)。在调度任务时,需要提供一个函数和一个参数。然后任务通过 event_fetch()返回,随后基于 event_call()执行。

该基础架构的简化版本如图 3-31 所示。一系列“任务”(task)表示当前就绪的任务队列。这里没有显示其他类型的各种队列。

2. 内核线程架构

为了提高性能和稳定性,FRR 已引入内核线程。很自然,内核线程体系结构一直被认为与事件驱动体系结构是正交的,两者在设计选择方面确实有很大的重叠。由于事件模型被紧密地集成到 FRR 中,因此对如何引入 pthread、它们扮演什么角色以及它们如何与事件模型互操作等方面的工作需要进行仔细的考虑。

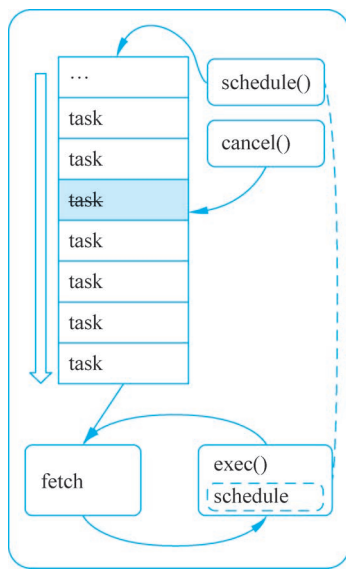


图 3-31 使用单个线程管理程序的生命周期

每个内核线程都表现为 FRR 中的轻量级进程,共享相同的进程内存空间。另外,事件系统被设计成在单个进程中运行,并驱动一组任务的串行执行。考虑到这一点,自然的选择是在每个内核线程中实现事件系统,这允许利用事件驱动的执行模型和当前存在的任务和上下文原语。通过这种方式,FRR 执行模型不仅获得了同时执行任务的能力,还保留了现有的并发模型。

图 3-32 展示了具有多个 pthread 的体系结构,每个 pthread 运行它们自己的基于线程管理器的事件循环。每个圆角矩形表示事件体系结构中运行相同事件循环的单个 pthread。注意,从右边的 exec()框到中间线程中的 schedule()框的箭头,说明在一个 pthread 中运行的代码将一个任务调度到了另一个 pthread 的线程管理器上。每个线程管理器都有一个全局锁来同步这些操作。

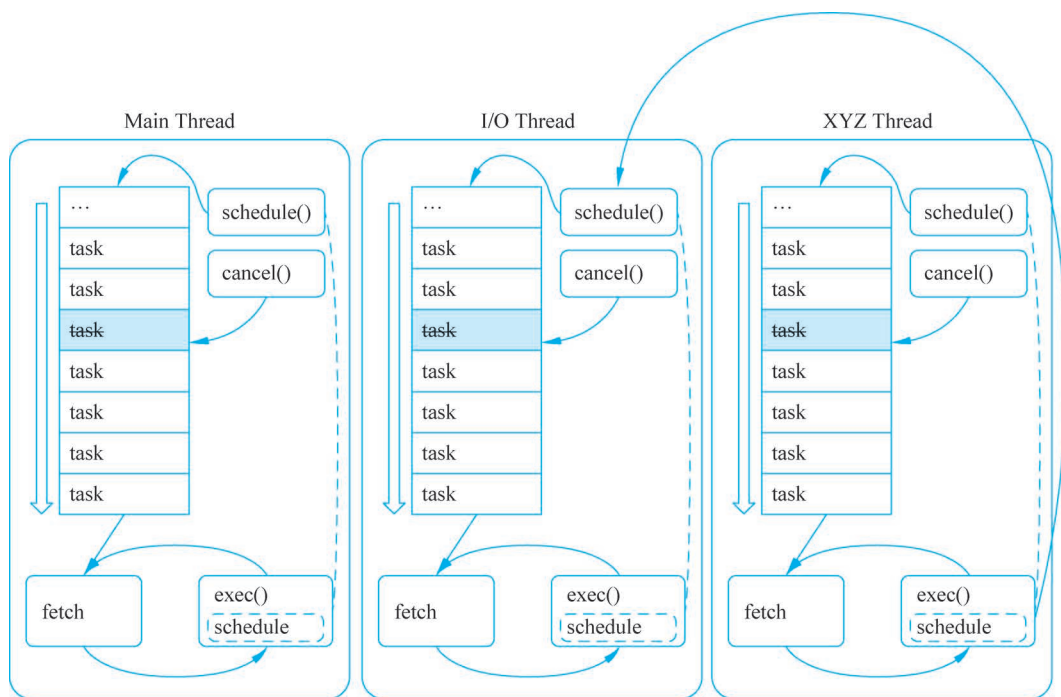


图 3-32 使用多个线程的程序生命周期,每个线程运行自己的线程管理器

3. 内核线程包装器

pthread 和事件系统集成基础是在 lib/frr_pthread.[ch]中实现的两个系统的轻量级包装器(wrapper)。头文件提供了一个核心数据结构 struct frr_pthread,它封装了来自 POSIX 线程和 event.c、event.h 的结构。具体而言,此数据结构具有指向在 pthread 中运行的线程管理器的指针。它还具有名为 start()和 stop()的函数,这些函数的签名类似于 pthread_create()的 POSIX 参数。

调用 frr_pthread_new()函数会创建并注册一个新的 frr_pthread。返回的结构体包含一个预初始化的线程管理器,它的 start()和 stop()函数初始化为默认值,并使用给定的线程管理器运行一个基本事件循环。

调用 frr_pthread_run()函数会使用 start()函数启动线程。从这里开始,该模型与常规事件模型相同。要在特定的 pthread 上调度任务,只需像往常一样使用 event.c 文件中的常规函数,并提供从 frr_pthread 指向的线程管理器。作为实现包装台的一部分,event.c 中的函数是

线程安全的。因此,在属于调用线程和任何其他 pthread 的线程管理器上调度事件是安全的。这是线程间通信的基础,可以归结为稍微复杂一点的消息传递方法,其中的消息是事件驱动模型中使用的常规任务事件。唯一的区别是线程取消,它需要调用 `event_cancel_async()` 而不是 `event_cancel()` 来取消 threadmaster 上调度的任务,该任务不属于同一个 pthread 的情况。若一个 pthread 想要保证另一个 pthread 上的任务在继续之前被取消,则需要避免线程间的竞态条件问题。

3.5.3 模块组成

1. Zebra 协议

1) 协议概述

Zebra 协议通常称为 ZAPI,是一个关键的网络协议,用于实现协议守护程序和 Zebra 守护程序之间的通信。这一协议在网络管理中发挥着重要作用,特别是在路由信息的交换和处理方面。

该协议允许不同的网络协议守护程序与 Zebra 守护程序进行有效的通信。通过这种方式,它们能够共享和更新关于网络接口状态、路由表、下一跳验证等关键信息。这种通信机制确保了网络的高效和稳定运行,同时也使得路由决策能够根据网络状况的实时变化而动态调整。

Zebra 协议支持多种路由协议,如 RIP、OSPF 等,通过与 Zebra 守护程序的交互,它们能够在路由表中添加、删除或修改路由。这种灵活性使得 Zebra 协议在复杂的网络环境中尤为重要,能够适应多变的网络需求和配置。除了基本的路由信息交换,Zebra 协议还提供了一系列高级功能,如数据平面批处理,这进一步优化了信息处理和传输的效率。此外,它的设计允许多个协议守护进程同时与 Zebra 守护程序通信,从而增加了网络管理的灵活性和扩展性。

2) 功能与特点

Zebra 协议提供了一系列强大的功能和特点,使其在网络管理和路由决策中发挥关键作用。首先,其核心功能之一是支持多种路由信息的交换和处理。这包括对接口状态、路由状态以及下一跳的有效验证。通过这些功能,Zebra 协议能够实时更新网络状态,确保路由决策反映当前的网络环境。

Zebra 协议支持数据平面批处理。这一特性优化了信息处理的效率,降低了网络延迟,确保在高负载情况下网络的稳定性。数据平面批处理是网络性能优化的关键,特别是在数据中心和大型企业网络中。

此外,Zebra 协议的另一特点是其灵活性和可扩展性。它允许多个协议守护进程与 Zebra 守护程序进行通信,这为网络管理员提供了更多的控制和定制选项。这种多守护进程通信机制使得 Zebra 协议能够适应不同的网络需求和配置,增强了网络应对复杂场景的能力。

Zebra 协议还具有高度的兼容性,能够与多种现有的网络协议和平台无缝集成。这种兼容性确保了 Zebra 协议能够在各种网络环境中有效运行,不论是传统的网络架构还是最新的云基础设施。

总的来说,Zebra 协议凭借其强大的功能和灵活性,在网络管理和路由优化中发挥着不可或缺的作用。它不仅提高了网络的性能和效率,还大大增强了网络管理员在面对日益复杂网络挑战时的能力。

Protobuf 是众多新的序列化格式之一,其中消息架构以专用语言表示。用于对 wire 格式进行编码/解码的代码是从架构生成的。Protobuf 消息可以轻松扩展,同时保持与旧代码的兼容性。Protobuf 与 Netlink 相比具有以下优势:

- 序列化/反序列化的代码是自动生成的。这降低了出现问题的可能性,允许快速集成第三方程序,并且易于添加字段。
- 消息格式不依赖操作系统(Linux),可以独立发展。

目前 Zebra 中有两个 FPM 模块: FPM 和 dplane_fpm_nl。

(1) FPM。

第一个使用 Zebra 路由处理函数中的钩子构建的 FPM 实现。它使用自己的 netlink/protobuf 编码函数将 ZebraRoute 数据结构转换为格式化的二进制数据。

(2) dplane_fpm_nl。

使用 Zebra 的数据平面框架作为插件构建的较新的 FPM 实现。它仅支持 Netlink,并共享 Zebra 的 Netlink 功能,以将路由事件快照转换为格式化的二进制数据。

在协议规范方面,FPM 使用 TCP 连接与外部应用程序通信。它作为 TCP 客户端运行,并使用 CLI 配置的地址/端口连接到 FPM 服务器(默认为 2620 端口)。

FPM 使用标头对所有数据进行帧处理,以帮助外部读者计算必须读取多少字节才能读取完整消息(这有助于模拟数据报,就像在原始 Linux Netlink 内核使用中一样)。

dplane_fpm_nl 能够从底层 FPM 读取路由 Netlink 消息,该消息可以告诉 Zebra 路由是否已卸载/失败或捕获。最终开发人员必须将数据发送到已创建的同一套接字,以侦听来自 Zebra 的 FPM 消息。发送的数据必须具有 Version 设置为 1、MessageType 设置为 1 的 Frame Header 和相应的消息长度。消息数据必须包含一条 RTM_NEWROUTE Netlink 消息,该消息发送与路由关联的前缀和下一跃点。最后,rtm_flags 必须包含 RTM_F_OFFLOAD、RTM_F_TRAP 和/或 RTM_F_OFFLOAD_FAILED,以表示 ASIC 中的路由操作。

3. PCEplib 库

1) PCEplib 概述

PCEplib 是一个为路径计算元素(PCE)和路径计算客户端(PCC)设计的路径计算元素协议(PCEP)实现库。它在网络计算中扮演着重要角色,主要用于优化和简化网络路径的计算过程。通过提供高效且可扩展的解决方案,PCEplib 支持网络管理员和工程师更好地管理和优化网络资源。PCEplib 的设计目的是增强网络路径计算的效率和准确性,同时提供灵活的网络管理功能,可适用于各种复杂和动态变化的网络环境。

2) PCEP 及其扩展

PCEP 是一种用于通信路径计算元素和路径计算客户端之间的网络协议。PCEplib 支持多种 PCEP 的扩展,如 RFC8281、RFC8231 和 RFC8232 等。这些扩展提高了 PCEplib 的灵活性和功能,使其能够适应不同的网络需求和环境,更有效地处理复杂的路径计算和网络优化任务。通过这些扩展,PCEplib 可以实现诸如状态报告、请求取消、路径计算反馈等高级功能,为用户提供更全面的网络管理工具。这些功能的集成,使 PCEplib 成为网络路径计算和优化的强大工具,可适用于多种复杂的网络场景。

3) PCEplib 的架构

PCEplib 采用了模块化的架构设计,以增强其灵活性和可扩展性,主要包括 pcep_messages、pcep_pcc 和 pcep_session_logic 模块。pcep_messages 模块负责处理 PCEP 消息的编码和解码,涉及创建和解析消息对象及其 TLV 结构;pcep_pcc 模块管理路径计算客户端的功能,提

供了 PCEplib 的公共 PCC API 以及示例 PCC 二进制文件,这些 API 封装了其他 PCEplib 库的功能提供 PCC 功能的接口,简化了与其他 PCEplib 模块的交互过程;pcep_session_logic 模块则控制 PCEP 会话的逻辑流程,包括消息接收、会话对象管理、定时器设置和计数器管理。其他模块,如 pcep_socket_comm 和 pcep_timers 分别负责套接字通信和定时器处理,pcep_utils 则提供了内部工具。

总的来说,每个模块在 PCEplib 中都承担着特定的职责。pcep_messages 模块是整个库的核心,负责所有与 PCEP 消息相关的操作。pcep_pcc 模块则提供了一套机制,用于管理和控制 PCC 的行为和状态。pcep_session_logic 模块的职责是维护 PCEP 会话的稳定性和效率,确保消息的正确传输和处理。这些模块相互协作,共同提升了 PCEplib 的可靠性和其他性能。PCEplib 的架构如图 3-34 所示。

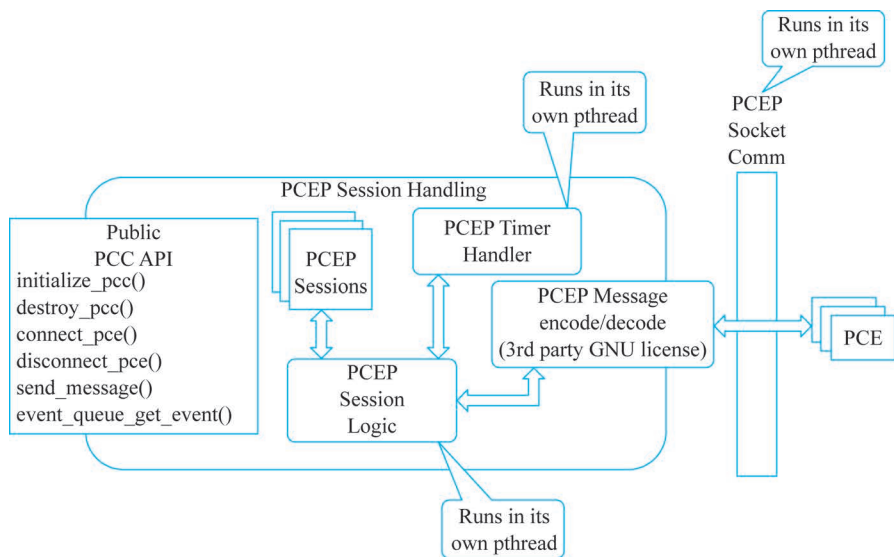


图 3-34 PCEplib 架构图

4) API 和配置

PCEplib 提供了一套详细的 API,使用户能够轻松地集成和使用这个库。这些 API 涵盖了从基本配置到高级功能的各个方面,包括 PCC 初始化、配置参数设定、与 PCE 的连接与断开管理。API 还涉及 PCEP 消息、对象和 TLV 的创建,管理及在会话中的应用,事件队列的管理以及 PCEP 事件和消息的处理。通过这些 API,PCEplib 可以适应不同网络环境和需求,提高在实际应用中的灵活性和效率。

5) 应用

PCEplib 的多场景应用能力是其核心优势之一,具体体现在其对复杂路径计算的处理能力,特别是在数据中心或广域网络等特定场景中。它通过优化路径选择,有效减少延迟并提高数据传输效率。在网络拥塞管理方面,PCEplib 利用高级路径计算能力识别并规避高流量区域,减少数据包的延迟和丢失。此外,PCEplib 能够动态调整路径和优化网络流量分布,从而提高网络的整体效率和稳定性。这些能力的综合应用使得 PCEplib 在不同网络环境中能够提供高效和灵活的网络管理解决方案。

FRR 的模块化架构使得用户可以根据需求选择和配置所需的路由协议,从而满足不同网络环境的需求。它提供了丰富的功能,如路由过滤、路由重分发、路由聚合、路由策略、路由跟踪和路由监控等,为网络管理员和运维人员提供了灵活性和可定制性。作为开源软件,FRR

得到了广泛的开发者社区支持和贡献。开放的代码和社区合作使得 FRR 能够持续改进和发展,不断适应新的网络需求和技术发展。

综上所述,FRR 是一个强大、可靠且灵活的 IP 路由套件,适用于各种网络设备和系统。它在提供高性能、多协议支持和丰富功能方面表现出色,为构建可靠和高效的网络架构提供了可靠的解决方案。