OSPF 路由协议概述

1.1 内容简介

随着 Internet 技术在全球范围的飞速发展，OSPF 已成为目前 Internet 广域网和 Intranet 企业网采用最多、应用最广泛的路由协议之一。OSPF 是 Open Shortest Path First（开放最短路由优先协议）的缩写。它是 IETF 组织开发的一个基于链路状态的内部网关协议。目前针对 IPv4 协议使用的是 OSPF Version 2（RFC2328）。

OSPF 协议是由 Internet Engineering Task Force 的 OSPF 工作组所开发的，特别为 TCP/IP 网络而设计，包括明确的支持 CIDR 和标记来源于外部的路由信息。OSPF 也提供了对路由更新的验证，并在发送/接收更新时使用 IP 多播。此外，还作了很多的工作使得协议仅用很少的路由流量就可以快速地响应拓扑改变。

本文主要介绍 OSPF 路由协议的基本原理，包括：OSPF 的协议报文、邻居状态机、链路状态同步，以及 DR、BDR 选举和 OSPF 区域的划分。

本文来源于 H3C 网络学院教材，作为 OSPF 路由协议实验的主要参考资料。通过这个实验，学生应该能掌握 OSPF 路由协议的基本概念和基本原理，OSPF 路由计算过程，具备规划和配置 OSPF 路由协议的能力，并能处理一般的 OSPF 故障。

🕮 说明：

若没有特别说明，下文中所提到的 OSPF 均指 OSPFv2。

1.2 相关概念回顾

在理论课程中，我们对距离矢量和链路状态路由协议有了一定的了解。下面，我们将会回顾一些相关的概念，指出距离矢量算法和链路状态算法的一些区别。

1.2.1 路由表

路由器转发分组的关键是路由表。每个路由器中都保存着一张路由表，表中每条路由项都指明分组到某子网或某主机应通过路由器的哪个物理端口发送，然后就可到达该路径的下一个路由器，或者不再经过别的路由器而传送到直接相连的网络中的目的主机。

根据来源不同，路由表中的路由通常可分为以下三类：

1. 链路层协议发现的路由（也称为接口路由或直连路由）
2. 由网络管理员手工配置的静态路由
3. 动态路由协议发现的路由。

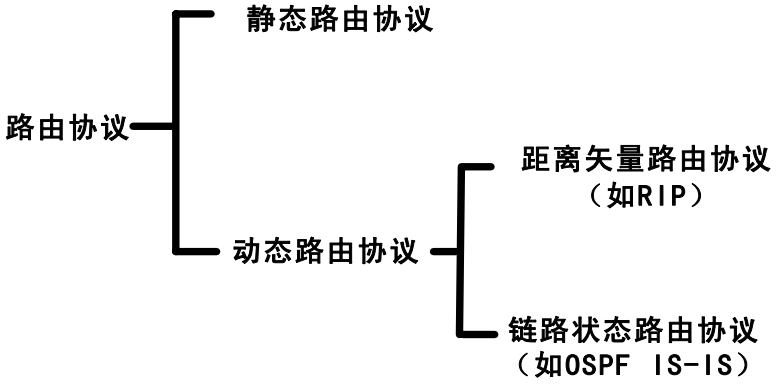


图1-1 如图 1-1 路由协议的分类

其中动态路由协议又包括有：TCP/IP 协议栈的 RIP（Routing Information Protocol，路由信息协议）协议、OSPF（Open Shortest Path First，开放式最短路径优先）协议；OSI 参考模型的 IS-IS（Intermediate System to Intermediate System）协议等。如图 1-1。

1.2.2 距离矢量算法和链路状态算法

动态路由协议有很多种，分类标准也很多。主要的分类标准是根据算法的不同来划分，不同的算法能适应的网络规模也不尽相同。目前常见的动态路由协议，根据使用的算法可分为：

- 距离矢量协议（Distance-Vector）：包括 RIP 和 BGP。其中，BGP 也被称为路径矢量协议（Path-Vector）。
- 链接状态协议（Link-State）：包括 OSPF 和 IS-IS。

以上两种算法的主要区别在于发现路由和计算路由的方法。

1. 距离矢量协议（Distance-Vector）：

距离矢量协议也称为 Bellman-Ford 协议，网络中路由器向相邻的路由器发送它们的整个路由表。路由器在从相邻路由器接收到的信息的基础之上建立自己的路由表。然后，将信息传递到它的相邻路由器。这样一级级的传递下去以达到全网同步。也就是说距离矢量路由表中的某些路由项有可能是建立在第 2 手信息的基础之上的，每个路由器都不了解整个网络拓

扑，它们只知道与自己直接相连的网络情况，并根据从邻居得到的路由信息更新自己的路由表，进行矢量行叠加后转发给其它的邻居，

距离矢量算法存在的一个重要的问题就是会产生路由环路。路由环路问题产生的原因和距离矢量算法的原理有关，正如前面所讲的，每个路由器根据从其它路由器接收到的信息来建立自己的路由表。如果某个路由器出现“故障”或者因为别的原因而无法在网上使用时，就会造成路由环路。

针对产生回路的问题，防止和解决的方法有：

- 定义最大路由权值
- 水平分割
- 毒性逆转
- 路由保持法
- 触发更新

距离矢量协议无论是实现还是管理都比较简单，但是它的收敛速度慢，报文量大，占用较多网络开销，并且会产生路由环路，为避免路由环路得提供特殊处理。路由环路是 DV 算法必须要解决的问题，只有处理好环路问题的路由协议才能应用在实际的系统中。常见的 D-V 路由协议一般都会采用上述阐述的多种方法解决路由环路的问题。

2. 链路状态算法

链路状态算法对路由的计算方法和距离矢量算法有本质的差别。距离矢量算法是一个平面式的,所有的路由表项学习完全依靠邻居,交换的是整个路由表项。链路状态是一个层次式的,执行该算法的路由器不是简单的从相邻的路由器学习路由，而是把路由器分成区域，收集区域内所有路由器的链路状态信息，根据链路状态信息生成网络拓扑结构，每一个路由器再根据拓扑结构图计算出路由。链路状态路由协议的一些注意事项如下：

- 网络中的设备并不向邻居传递“路由信息”，而是通告给邻居一些链路状态。
- 网络中的设备（路由器）最终都会得到网络的拓扑结果元素，只是详细程度并不相同。
- 各设备以自身为“树根”，根据 LSDB 中的数据计算到各个网断的“最短生成树”。
- 由于各设备有义务将网络拓扑信息向下传递,而具体的路由信息又是各设备自己计算所得，所以链路状态协议中，对是否“发出路由”不能进行控制，但可以控制路由的某些属性。

3. 距离矢量算法和链路状态算法的比较

采用链路状态算法的路由器，首先要得到整个网络的拓扑结构，再根据网络拓扑图计算出路由。这种路由的计算方法对路由器的硬件相对要求较高，但它计算准确，一般可以确保网络

中没有路由环路存在。由于路由不是在路由器间顺序传递的，网络动荡时，路由收敛速度较快。而且路由器不需要定期的将路由信息复制到整个网络中，网络流量相对较小。表 1-1 对这两种路由算法作一比较：

表1-1 距离矢量算法与链路状态算法

	距离矢量算法	链路状态算法
是否有环路	有	无
收敛速度	慢	快
对路由器 CPU、RAM 的要求	低	高
网络流量	大	小
典型协议	RIP、BGP	OSPF、IS-IS

1.3 OSPF 的基本概念

OSPF 是 Open Shortest Path First（即“开放最短路径优先协议”）的缩写。IETF(Internet Engineering Task Force)于 1988 年提出的 OSPF 是一个基于链路状态的动态路由协议。OSPF 是一类 Interior Gateway Protocol(内部网关协议 IGP)，它处理在一个自治系统中的路由表信息。当前 OSPF 协议使用的是第二版，最新的 RFC 是 2328。

OSPF 协议是特别为 TCP/IP 网络而设计，包括明确的支持 CIDR 和标记来源于外部的路由信息。OSPF 也提供了对路由更新的验证，并在发送/接收更新时使用 IP 多播。此外，还作了很多的工作使得协议仅用很少的路由流量就可以快速地响应拓扑改变。

O S P F 用链路状态算法来计算在每个区域中到所有目的的最短路径，当一个路由器首先开始工作，或者任一个路由变化发生，这个配备给 O S P F 的路由器将 L S A 扩散到同一级区域内所有路由器，这些 L S A 包含这个路由器的链接状态和它与邻居路由器联系的信息，从这些 L S A 的收集中形成了链路状态数据库，在这个区域中的所有路由器都有一个特定的数据库来描述这个区域的拓扑结构。

在 OSPF 路由协议中，可以通过划分区域来分割整个自治系统，每一个区域都有着该区域独立的网络拓扑数据库及网络拓扑图。对于每一个区域，其网络拓扑结构在区域外是不可见的，同样，在每一个区域中的路由器对其域外的其余网络结构也不了解。这意味着 OSPF 路由域中的网络链路状态数据广播被区域的边界挡住了，这样做有利于减少网络中链路状态数据包在全网范围内的广播，也是 OSPF 将自治系统划分成很多个区域的重要原因。

OSPF 仅通过在 IP 包头中的目标地址来转发 IP 包。IP 包在 AS 中被转发，而没有被其他协议再次封装。OSPF 是一种动态路由协议，它可以快速地探知 AS 中拓扑的改变（例如路由器接口的失效），并在一段时间的收敛后计算出无环路的新路径。收敛的时间很短且只使用很小的路由流量。

🕮 说明：

作为一种链路状态的路由协议，OSPF 将链路状态广播数据包 LSA（Link State Advertisement）传送给在某一区域内的所有路由器，这一点与距离矢量路由协议不同。运行距离矢量路由协议的路由器是将部分或全部的路由表传递给与其相邻的路由器。

1.3.1 OSPF 的基本特点

OSPF 是一种基于链路状态（Link-state）算法的协议，其核心思想是：每一台路由器将自己周边的链路状态（包括接口的直接路由、相连的路由器等信息）描述出来，发送给网络中所有的路由器。每台路由器在收到其他所有路由器的发送的链路状态信息之后，运行 Shortest Path First 算法计算路由。

随着 Internet 技术在全球范围的飞速发展，OSPF 已成为目前 Internet 广域网和 Intranet 企业网采用最多、应用最广泛的路由协议之一。

OSPF 协议具有如下特点：

- 适应范围 —— OSPF 支持各种规模的网络，最多可支持几百台路由器。
- 快速收敛 —— 如果网络的拓扑结构发生变化，OSPF 立即发送更新报文，使这一变化在自治系统中同步。
- 无自环 —— 由于 OSPF 通过收集到的链路状态用最短路径树算法计算路由，故从算法本身保证了不会生成自环路由。
- 子网掩码 —— 由于 OSPF 在描述路由时携带网段的掩码信息，所以 OSPF 协议不受自然掩码的限制，对 VLSM 提供很好的支持。
- 区域划分 —— OSPF 协议允许自治系统的网络被划分成区域来管理，区域间传送的路由信息被进一步抽象，从而减少了占用网络的带宽。
- 等值路由 —— OSPF 支持到同一目的地址的多条等值路由，即到达同一个目的地有多个下一跳，这些等值路由会被同时发现和使用。
- 路由分级 —— OSPF 使用 4 类不同的路由，按优先顺序来说分别是：区域内路由、区域间路由、第一类外部路由、第二类外部路由。

- 支持验证 —— 它支持基于接口的报文验证以保证路由计算的安全性。
- 组播发送 —— OSPF 在有组播发送能力的链路层上以组播地址发送协议报文，不仅达到了广播的作用，而且最大程度的减少了对其他网络设备的干扰。

1.3.2 Router ID

OSPF 协议使用一个被称为 Router ID 的 32 位无符号整数来唯一标识一台路由器。基于这个目的，每一台运行 OSPF 的路由器都需要一个 Router ID。这个 Router ID 一般需要手工配置，一般将其配置为该路由器的某个接口的 IP 地址。

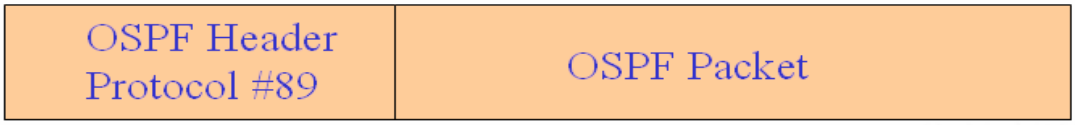
由于 IP 地址是唯一的，所以这样就很容易保证 Router ID 的唯一性。在没有手工配置 Router ID 的情况下，一些厂家的路由器（包括 Quidway 系列）支持自动从当前所有接口的 IP 地址自动选举一个 IP 地址作为 Router ID。

Router ID 选择注意点：

- 首先选取最大的 loopback 接口地址
- 如果没有配置 loopback 接口，那么就选取最大的物理接口地址
- 可以通过命令强制改变 Router ID：VRP 平台系统视图下，router id <ip address>
- 如果一台路由器的 Router ID 在运行中改变，则必须重启 OSPF 协议或重启路由器才能使新的 Router ID 生效

协议号：

OSPF 协议用 IP 报文直接封装协议报文，协议号是 89。



📖 说明：

通常 OSPF 的协议报文是不被转发的，只能传递一跳，即在 IP 报文头中 TTL 值被设为 1 (虚连接除外)。

为保证 OSPF 运行的稳定性，在进行网络规划时应该确定路由器 ID 的划分并手工配置。手工配置路由器的 ID 时，必须保证自治系统中任意两台路由器的 ID 都不相同。通常的做法是将路由器的 ID 配置为与该路由器某个接口的 IP 地址一致。

VRP3.4 平台的 OSPF 支持多进程，在同一台路由器上可以运行多个不同的 OSPF 进程，它们之间互不影响，彼此独立。OSPF 进程号是本地概念，不影响与其它路由器之间的报文交换。因此，不同的路由器之间，即使进程号不同也可以进行报文交换。不同 OSPF 进程之

间的路由交互相当于不同路由协议之间的路由交互。路由器的一个接口只能属于某一个 OSPF 进程。

1.3.3 SPF 算法和 COST 值

1. SPF 算法

SPF 算法是 OSPF 路由协议的基础。SPF 算法有时也被称为 Dijkstra 算法，这是因为最短路径优先算法 SPF 是 Dijkstra 发明的。SPF 算法将每一个路由器作为根（ROOT）来计算其到每一个目的地路由器的距离，每一个路由器根据一个统一的数据库会计算出路由域的拓扑结构图，该结构图类似于一棵树，在 SPF 算法中，被称为最短路径树。

2. Cost 值

在 OSPF 路由协议中，最短路径树的树干长度，即 OSPF 路由器至每一个目的地路由器的距离，称为 OSPF 的 Cost。Cost 值应用于每一个启动了 OSPF 的链路，它是一个 16 bit 的正数，范围是 1～65535。

Cost 值越小，说明路径越好。OSPF 选择路径是依靠整个链路 Cost 值的总和。那么 Cost 值是如何计算的呢？

OSPF 协议中，Cost 值的计算方法是用 10^8/链路带宽。在这里，链路带宽以 bps 来表示。也就是说，OSPF 的 Cost 与链路的带宽成反比，带宽越高，Cost 越小，表示 OSPF 到目的地的距离越近。举例来说，56k 的链路花费是 1785，10M 以太网链路花费是 10，64k 的链路花费是 1562，T1 的链路花费是 64。

缺省情况下，接口按照当前的波特率自动计算接口运行 OSPF 协议所需的开销。

1.3.4 OSPF 路由的计算过程

OSPF 是基于链路状态算法的路由协议，所有对路由信息的描述都是封装在 LSA 中发送出去。L S A（Link State Advertisement）用来描述路由器的本地状态， L S A 包括的信息是关于路由器接口的状态和所形成的邻接状态。

每台 OSPF 路由器都会收集其它路由器发来的 LSA，所有的 LSA 放在一起便组成了链路状态数据库 LSDB（Link State Database）。LSDB 则是对整个自治系统的网络拓扑结构的描述。到达某个目的网段的最短路径，可通过这些信息计算出来。

在图 1-2 中，描述了通过 OSPF 协议计算路由的过程。一个典型的网络，由四台运行 OSPF 的路由器组成，连线旁边的数字表示从一台路由器到另一台路由器所需要的花费。为简化问题，我们假定两台路由器相互之间发送报文所需花费是相同的

1. 每台路由器都根据自己周围的网络拓扑结构生成一条 LSA（Link State Advertisement），并通过相互之间发送协议报文将这条 LSA 发送给网络中其它的所有路由器。这样每台路由器都收到了其它路由器的 LSA，所有的 LSA 放在一起称作 LSDB（链路状态数据库）。显然，4 台路由器的 LSDB（Link State Database） 都是相同的。

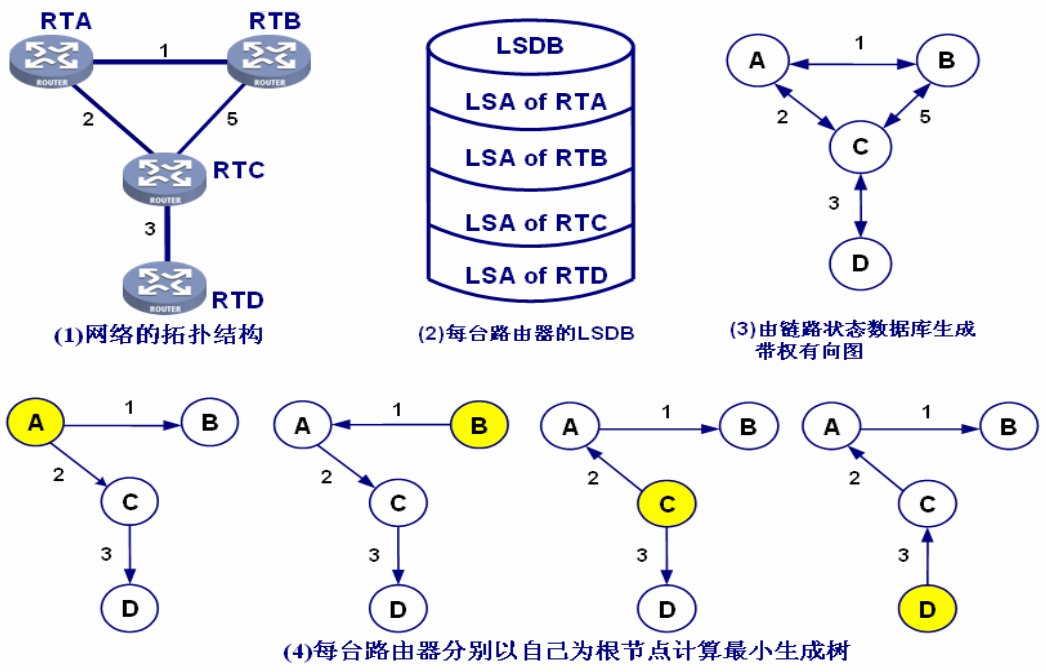


图1-2 图 1-2 OSPF 协议计算路由的过程

2. 由于一条 LSA 是对一台路由器周围网络拓扑结构的描述，那么 LSDB 则是对整个网络的拓扑结构的描述。路由器很容易将 LSDB 转换成一张带权的有向图，这张图便是对整个网络拓扑结构的真实反映。显然，4 台路由器得到的是一张完全相同的拓扑图。

3. 接下来每台路由器在图中以自己为根节点，使用 SPF 算法计算出一棵最短路径树，由这棵树得到了到网络中各个节点的路由表。显然，4 台路由器各自得到的路由表是不同的。

这样每台路由器都计算出了到其它路由器未知网段的路由。

1.3.5 OSPF 报文头

OSPF 有五种报文类型，他们有相同的报文头。如下图所示。

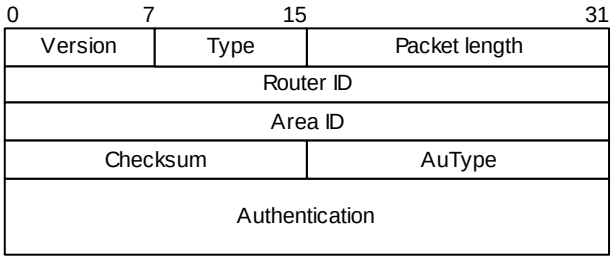


图1-3 OSPF 报文头格式

主要字段的解释如下：

- Version：OSPF 的版本号。对于 OSPFv2 来说，其值为 2。
- Type：OSPF 报文的类型。数值从 1 到 5，分别对应 Hello 报文、DD 报文、LSR 报文、LSU 报文和 LSAck 报文。
- Packet length：OSPF 报文的总长度，包括报文头在内，单位为字节。
  - Router ID 报文起源的 Router ID。
  - Area ID 一个 32 位的数，标识报文属于哪个区域，所有 OSPF 报文只属于单个区域，且只有一跳。当报文在虚链接上承载时，会打上骨干区域 0.0.0.0 的标签。
  - Checksum 包的整个内容的校验，从 OSPF 报文头部开始，但是除了 64 位的认证字段。
  - AuType 认证类型包括四种：0（无需认证），1（明文认证），2（密文认证）和其他类型（IANA 保留）。当不需要认证时，只是通过 Checksun 检验数据的完整性；当使用明文认证时，64 位的认证字段被设置成 64 位的明文密码；当使用密文认证时，对于每一个 OSPF 报文，共享密钥都会产生一个“消息位”加在 OSPF 报文的后面，由于在网络上从来不以明文的方式发送密钥，所以提高了网络安全性。
- Authentication：其数值根据验证类型而定。当验证类型为 0 时未作定义，为 1 时此字段为密码信息，类型为 2 时此字段包括 Key ID、MD5 验证数据长度和序列号的信息。

📖 **说明：**

MD5 验证数据添加在 OSPF 报文后面，不包含在 Authenticaiton 字段中。

1.3.6 OSPF 的五种报文类型

- HELLO 报文（Hello Packet）：

最常用的一种报文，周期性的发送给本路由器的邻居，使用的组播地址 224.0.0.5。DR 和 BDR 发送和接收报文使用的组播地址是 224.0.0.6。

HELLO 报文内容包括一些定时器的数值，DR，BDR，以及自己已知的邻居。

根据 RFC2328 的规定，要保持网络邻居间的 hello 时间间隔一致。需要注意的是，hello 时钟的值与路由收敛速度、网络负荷大小成反比。

缺省情况下，**p2p**、**broadcast** 类型接口发送 Hello 报文的时间间隔的值为 10 秒；**p2mp**、**nbma** 类型接口发送 Hello 报文的时间间隔的值为 30 秒。

📖 **说明：**

根据路由器使用链路层协议的不同，OSPF 将网络分为四种类型：
点到点 P2P（point-to-point）类型、广播（Broadcast）类型、NBMA（Non-Broadcast Multi-Access）类型、点到多点 P2MP（point-to-multipoint）类型。

注意：

没有一种链路层协议会被缺省的认为是 Point-to-Multipoint 类型。点到多点必须是由其他的网络类型强制更改的。常用做法是将非全连通的 NBMA 改为点到多点的网络。
具体内容，我们会在 OSPF 的网络类型章节讲解。

- DD 报文（Database Description Packet）：

路由信息(连接状态传送报文)只在形成邻接关系的路由器间传递。

首先，它们之间互发 DD（database description）报文，告之对方自己所拥有的路由信息，内容包括 LSDB 中每一条 LSA 的摘要（摘要是指 LSA 的 HEAD，通过该 HEAD 可以唯一标识一条 LSA）。

这样做是为了减少路由器之间传递信息的量，因为 LSA 的 HEAD 只占一条 LSA 的整个数据量的一小部分，根据 HEAD，对端路由器就可以判断出是否已经有了这条 LSA。

DD 报文有两种，一种是空 DD 报文，用来确定 Master/Slave 关系（避免 DD 报文的无序发送）。确定 Master/Slave 关系后，才发送有路由信息的 DD 报文。

收到有路由信息的 DD 报文后，比较自己的数据库，发现对方的数据库中有自己需要的数据，则向对方发送 LSR（Link State Request）报文，请求对方给自己发送数据。

- LSR 报文（Link State Request Packet）：

两台路由器互相交换过 DD 报文之后，知道对端的路由器有哪些 LSA 是本地的 LSDB 所缺少的或是对端更新的 LSA，这时需要发送 LSR 报文向对方请求所需的 LSA。内容包括所需要的 LSA 的摘要。

- LSU 报文（Link State Update Packet）：

用来向对端路由器发送所需要的 LSA，内容是多条 LSA（全部内容）的集合。

- LSAck 报文（Link State Acknowledgment Packet）

由于没有使用可靠的 TCP 协议，但是 OSPF 包又要求可靠的传输，所以就有了 LSAck 包。它用来对接收到的 LSU 报文进行确认。内容是需要确认的 LSA 的 HEAD（一个报文可对多个 LSA 进行确认）。

DD 报文、LSR 报文、LSU 报文发出后，在没有得到应有的对方相应的 LSR、LSU、LSAck 报文时，会重发。（例外：对 DD 报文若收到后发现没有必要产生连接状态请求报文，则不发连接状态请求报文。）同步后数据改变，则只向形成 Adjacency 关系的路由器发 LSU 报文。

1.3.7 LSA 头格式

所有的 LSA 都有相同的报文头，其格式如下图所示。

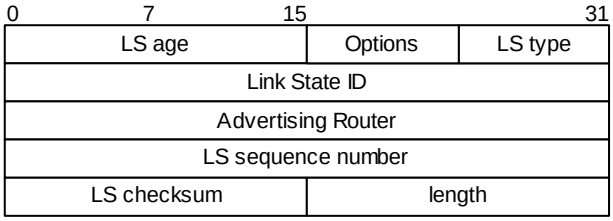


图1-4 LSA 的头格式

首先，我们分析一下 LSA 报文头：

LSA 头部都是 20 个字节，它包含了足够的信息来唯一标识一条 LSA（LS type，Link State ID，and Advertising Router ）。LSA 多实例在同一时候可以存在于路由域中，它被用来决定哪一个实例是最新的。LSA 头部还包括 LS 老化、LS 序列号和 LS 校验和等字段。

主要字段的解释如下：

LS age，用来标识 LS 产生的时间。生成 LSA 的路由器将 LS 时域初始化为 0，在洪泛过程中，每经过一个路由器，要按 InfTransDelay 的量增加，这个量表示传输 LSA 到下一个跳所需要的时间。当该时间达到所设定的 MaxAge 参数时，要撤消该 LSA。

Options，用来描述支持的路由域，主要包括 DC、EA、N/P、MC、E、T 等选项。DC 指的是始发路由器支持 Demand Circuits (按需拨号等)；EA 指的是始发路由器支持 External Attributes LSAs (现在未推广)；N/P 只用在 Hello 中 N=1，说明支持 NSSA，P 只用在 NSSA 中，通知 ABR 把 type7 的 LSA 翻译成 type5 的 LSA；MC 只在 MOSPF 中用到；E 表示可以接受外部路由（不是 stub 区），在一个 area 中的所有 router 此位必须一致，(Hello 中体现) 否则邻接关系无法建立；T 表示始发路由器支持 TOS。

LS type，链路类型。每种类型的 LSA 都有唯一的通告格式。

Link State ID，这个字段标识被描述的网络环境的一部分，Link State ID 的内容取决于 LSA 的类型，即不同类型的 LSA 其 Link State ID 也是不同的。比如，当 LSA 的类型是 Type 1 时， Link State ID 是始发路由器的 Router ID；当 LSA 的类型是 Type 2 时， Link State ID 是 DR 在该网段上接口的 IP 地址；当 LSA 的类型是 Type 3 时， Link State ID 是被通告的网络/子网的 IP 地址；当 LSA 的类型是 Type 4 时，Link State ID 是被通告 ASBR 的 Router ID；当 LSA 的类型是 Type 5 时， Link State ID 是目的地的 IP 地址。

Advertising Router，指始发此 LSA 的路由器的 Router ID。比如在 Network-LSAs 中，这个字段就是 DR 在该网段上接口的 IP 地址。

LS sequence number，用于识别 LSA 包是否是一个最新包。路由器每生成一个新的 LSA 时，将该序列号加 1。

LS checksum 用来检查 LSA 的完整性，包括除了 LS age 之外的 LSA 头部的内容。

Length，LSA 的长度，用 bytes 表示。LSA 的头部包括 20 字节。

LSA 头中的链路类型、链路状态 ID 和通告路由器的 Router ID 是一个 LSA 的唯一标识。一个 LSA 将有多个实例，不同的实例通过 LS 的序列号、LS 的校验和及 LS 的 Age 字段来描述。因此，必须要决定其实例是否是最近的，这要通过检查 LS 的序列号、LS 的校验和及 LS 的 Age 字段内容。

1.3.8 LSA 的类型

OSPF 是基于链路状态算法的路由协议，所有对路由信息的描述都是封装在 LSA 中发送出去。当路由器初始化或当网络结构发生变化（例如增减路由器，链路状态发生变化等）时，路由器会产生链路状态广播数据包 LSA（Link-State Advertisement），该数据包里包含路由器上所有相连的链路，也即为所有端口的状态信息。

LSA 根据不同的用途分为不同的种类，主要有如下类型的 LSA：

- Router LSA（Type = 1）：

是最基本的 LSA 类型，所有运行 OSPF 的路由器都会生成这种 LSA。主要描述本路由器运行 OSPF 的接口的连接状况，花费等信息。

对于 ABR，它会为每个区域生成一条 Router LSA。这种类型的 LSA 传递的范围是它所属的整个区域。

- Netwrok LSA（Type = 2）：

本类型的 LSA 由 DR 生成。对于广播和 NBMA 类型的网络，为了减少该网段中路由器之间交换报文的次数而提出了 DR 的概念。

一个网段中有了 DR 之后不仅发送报文的方式有所改变，链路状态的描述也发生了变化。

在 DROther 和 BDR 的 Router LSA 中只描述到 DR 的连接，而 DR 则通过 Network LSA 来描述本网段中所有已经同其建立了邻接关系的路由器。（分别列出它们 Router ID）。

同样， 这种类型的 LSA 传递的范围是它所属的整个区域。

- Network Summary LSA（Type = 3）：

本类型的 LSA 由 ABR 生成。当 ABR 完成它所属一个区域中的区域内路由计算之后，查询路由表，将本区域内的每一条 OSPF 路由封装成 Network Summary LSA 发送到区域外。

LSA 中描述了某条路由的目的地址、掩码、花费值等信息。

这种类型的 LSA 传递的范围是 ABR 中除了该 LSA 生成区域之外的其他区域。

- ASBR Summary LSA（Type = 4）：

本类型的 LSA 同样是由 ABR 生成。内容主要是描述到达本区域内部的 ASBR 的路由。

这种 LSA 与 Type3 类型的 LSA 内容基本一样，只是 Type4 的 LSA 描述的目的地址是 ASBR，是主机路由，所以掩码为 0.0.0.0。

这种类型的 LSA 传递的范围与 Type3 的 LSA 相同。

- AS External LSA（Type = 5）：

本类型的 LSA 由 ASBR 生成。主要描述了到自治系统外部路由的信息，LSA 中包含某条路由的目的地址、掩码、花费值等信息。

本类型的 LSA 是唯一一种与区域无关的 LSA 类型，它并不与某一个特定的区域相关。

这种类型的 LSA 传递的范围整个自治系统（STUB 区域除外）。

- Multicast OSPF LSA（Type =6）：

使用在 OSPF 多播应用程序里。

- Not-So-Stubby Area（Type =7）：

本类型的 LSA 由 Not-So-Stubby area(NSSA) 区域中 ASBR 生成。

为了解决 ASE 路由（自治系统外部路由）单向传递的问题，Not-So-Stubby area(NSSA) 中重新定义了一种 LSA——Type 7 类型的 LSA，作为区域内的路由器引入外部路由时使用。

该类型的 LSA 除了类型标识与 Type 5 不相同之外，其它内容基本一样。这样区域内的路由器就可以通过 LSA 的类型来判断是否该路由来自本区域内。

但由于 Type 7 类的 LSA 是新定义的，对于不支持 NSSA 属性的路由器无法识别，所以协议规定：在 NSSA 的 ABR 上将 NSSA 内部产生的 Type 7 类型的 LSA 转化为 Type 5 类型的 LSA 再发布出去，并同时更改 LSA 的发布者为 ABR 自己。这样 NSSA 区域外的路由器就可以完全不用支持该属性。

在 NSSA 区域内的所有路由器必须支持该属性（包括 NSSA 的 ABR），而自治系统中的其他路由器则不需要。

- External-Attributes-LSA（Type =8）：

特殊的 LSA，还没有实现。当 BGP 信息需要在 OSPF 上承载时，需要用到此 LSA。

- opaque LSA（Type =9～11）：

用于 MPLS 流量工程，有关此 LSA 的详细应用请参考网络学院 MPLS 流量工程培训教材或 RFC2370 文档。

当一台路由器向它的邻居发送一条 LSA 后，需要等到对方的确认报文。若在重传间隔时间内没有收到对方的确认报文，就会向邻居重传这条 LSA。

1.3.9 邻居和邻接

在 OSPF 中，邻居（Neighbors）和邻接（Adjacencies）是两个不同的概念。

OSPF 路由器启动后，便会通过 OSPF 接口向外发送 Hello 报文。收到 Hello 报文的 OSPF 路由器会检查报文中所定义的一些参数，如果双方一致就会形成邻居关系。

形成邻居关系的双方不一定都能形成邻接关系，这要根据网络类型而定。只有当双方成功交换 DD 报文，并能交换 LSA 之后，才形成真正意义上的邻接关系。

为了交换路由信息，邻居路由器之间首先要建立邻接关系，并不是每两个邻居路由器之间都能建立邻接关系。

HELLO Interval：接口上发送报文的时间间隔，以秒为单位。OSPF 邻居之间的 Hello 定时器的时间间隔要保持一致。Hello 定时器的值与路由收敛速度、网络负荷大小成反比。

如果两路由器不具有相同的呼叫周期，则不能成为邻接关系。

DEAD Interval：如果在 DEAD TIME 指定的秒数内没有从已建立的邻居处收到报文，那么，邻居被宣布为故障状态。

如果 Hello 报文中的 Dead Interval 与接收端口所设置的 DeadInterval 值不相同，则丢弃该报文。因此，要确保两邻居路由器具有相同的参数。

在同一接口上失效时间应至少为 Hello 间隔时间的 4 倍。

📖 **说明：**

Hello 报文主要负责建立和维护邻接关系，周期性的在路由器的接口上发送。当路由器发现自己被列在邻居路由器的 hello 报文中，双向通信就建立起来。
在不同类型的链路上，hello 报文工作的方式也不同。邻居建立后，还需要通过 HELLO 报文进行邻居关系的维持，有两个定时器来进行这项工作：

HELLO TIME：缺省为 10 秒（对于 NBMA 网络为 30 秒）
DEAD TIME：缺省为 4 倍的 HELLO TIME

邻接关系形成之后，接下来就是同步链路状态数据库。

1.3.10 OSPF 的邻居状态机

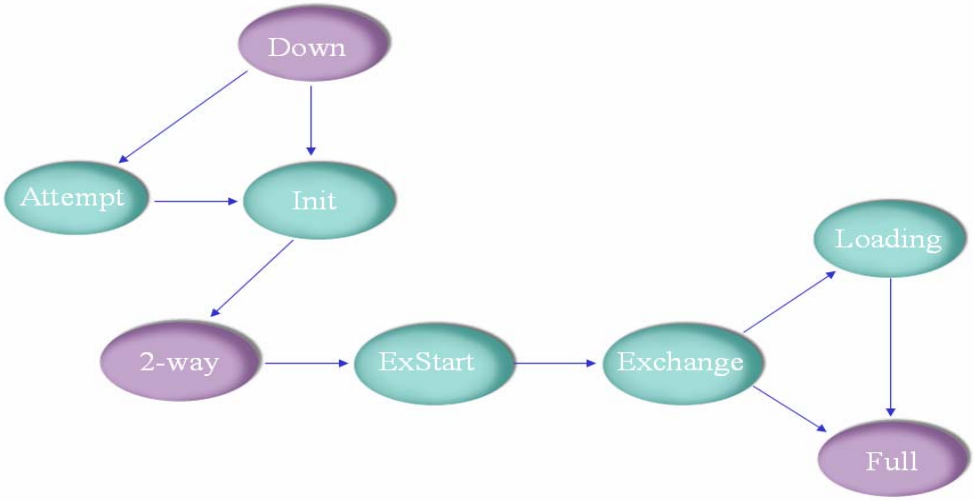


图1-5 OSPF 的邻居状态机

- Down：

邻居状态机的初始状态，是指在过去的 Dead-Interval 时间内没有收到对方的 Hello 报文。

- Attempt：

只适用于 NBMA 类型的接口，处于本状态时，定期向那些手工配置的邻居发送 HELLO 报文。

- Init：

本状态表示已经收到了邻居的 HELLO 报文，但是该报文中列出的邻居中没有包含我的 Router ID（对方并没有收到我发的 HELLO 报文）。

- 2-Way：

本状态表示双方互相收到了对端发送的 HELLO 报文，建立了邻居关系。在广播和 NBMA 类型的网络中，两个接口状态是 DROther 的路由器之间将停留在此状态。其他情况状态机将继续转入高级状态。

- ExStart：

在此状态下，路由器和它的邻居之间通过互相交换 DD 报文（该报文并不包含实际的内容，只包含一些标志位）来决定发送时的主/从关系。建立主/从关系主要是为了保证在后续的 DD 报文交换中能够有序的发送。

- Exchange：

路由器将本地的 LSDB 用 DD 报文来描述，并发给邻居。

- Loading：

路由器发送 LSR 报文向邻居请求对方的 LSU 报文。

- Full：

在此状态下，邻居路由器的 LSDB 中所有的 LSA 本路由器全都有了。即，本路由器和邻居建立了邻接（adjacency）状态。

🕮 说明：

Down、2-Way、Full 的状态是指稳定的状态，其他状态则是在转换过程中瞬间（一般不会超过几分钟）存在的状态。
本路由器和状态可能与对端路由器的状态不相同。例如本路由器的邻居状态是 Full，对端的邻居状态可能是 Loading。

1.3.11 链路状态数据库的同步过程

链路状态数据库同步过程的主要步骤：

- HELLO 报文发现邻居
- 主从关系协商
- DD 报文交换
- LSA 请求
- LSA 更新

(1) LSA 应答

如图 1-6 显示了两台路由器之间如何通过发送 5 种协议报文来建立邻接关系，以及邻居状态机的迁移。

- RT1 的一个连接到广播类型网络的接口上激活了 OSPF 协议，并发送了一个 HELLO 报文（使用组播地址 224.0.0.5）。由于此时 RT1 在该网段中还未发现任何邻居，所以 HELLO 报文中的 Neighbor 字段为空。

- RT2 收到 RT1 发送的 HELLO 报文后，为 RT1 创建一个邻居的数据结构。RT2 发送一个 HELLO 报文回应 RT1，并且在报文中的 Neighbor 字段中填入 RT1 的 Router ID，表示已收到 RT1 的 HELLO 报文，并且将 RT1 的邻居状态机置为 Init。

- RT1 收到 RT2 回应的 HELLO 报文后，为 RT2 创建一个邻居的数据结构，并将邻居状态机置为 Exstart 状态。下一步双方开始发送各自的链路状态数据库。

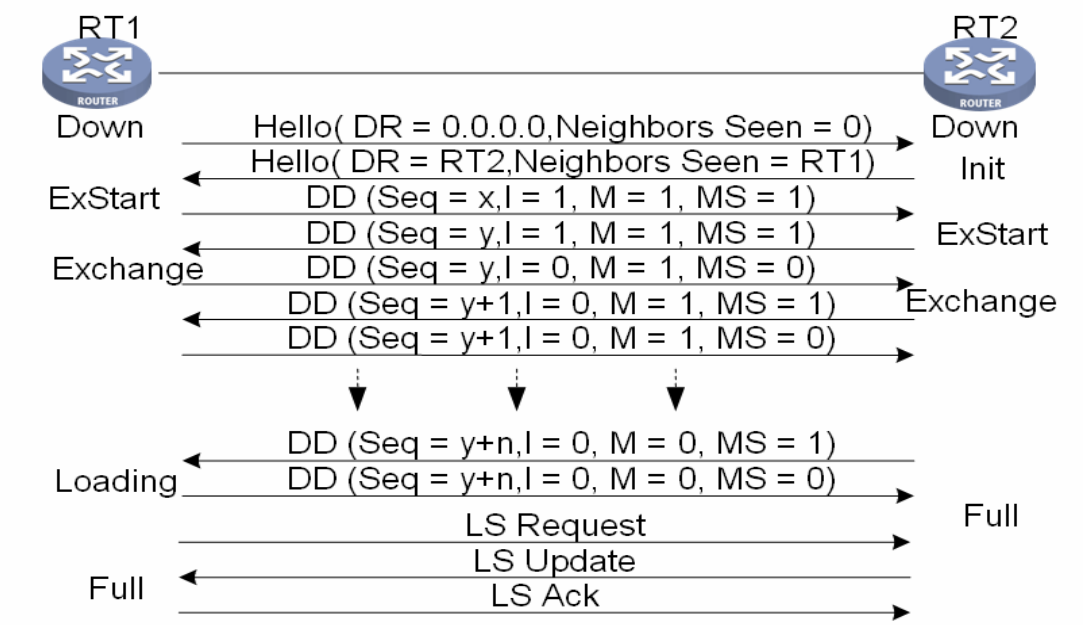


图1-6 两台路由器之间建立邻接关系的过程

为了提高发送的效率，双方需先了解一下对端数据库中那些 LSA 是自己所需要的（如果某一条 LSA 自己已经有了，就不再需要请求了）。

方法是先发送 DD 报文，DD 报文中包含了对本地数据库中 LSA 的摘要描述（每一条摘要可以唯一标识一条 LSA，但所占的空间要少得多）。由于 OSPF 直接用 IP 报文来封装自己的协议报文，所以在传输的过程中必须考虑到报文传输的可靠性。

为了做到这一点，在 DD 报文的发送过程中需要确定双方的主从关系。作为 Master 的一方定义一个序列号 seq，每发送一个新的 DD 报文将 seq 加一。作为 Slave 的一方，每次发送 DD 报文时使用接收到的上一个 Master 的 DD 报文中的 seq。实际上这种序列号机制是一种隐含的确认方法。如果再加上每个报文都有超时重传，就可以保证这种传输是可靠的。

RT1 首先发送一个 DD 报文，宣称自己是 Master（MS=1），并规定序列号为 x。I=1 表示这是第一个 DD 报文，报文中并不包含 LSA 的摘要，只是为了协商主从关系。M=1 说明这不是最后一个报文。

- RT2 在收到 RT1 的 DD 报文后，将 RT1 的邻居状态机改为 Exstart，并且回应了一个 DD 报文（该报文中同样不包含 LSA 的摘要信息）。由于 RT2 的 Router ID 较大，所以在报文中 RT2 认为自己是 Master，并且重新规定了序列号为 y。

- RT1 收到报文后，同意了 RT2 为 Master，并将 RT2 的邻居状态机改为 Exchange。RT1 使用 RT2 的序列号 y 来发送新的 DD 报文，该报文开始正式地传送 LSA 的摘要。在报文中 RT1 将 MS=0，说明自己是 Slave。
- RT2 收到报文后，将 RT1 的邻居状态机改为 Exchange，并发送新的 DD 报文来描述自己的 LSA 摘要，需要注意的是：此时 RT2 已将报文的序列号改为 y+1 了。
- 上述过程持续进行，RT1 通过重复 RT2 的序列号来确认已收到 RT2 的报文。RT2 通过将序列号加 1 来确认已收到 RT1 的报文。当 RT2 发送最后一个 DD 报文时，将报文中的 M=0，表示这是最后一个 DD 报文了。
- RT1 收到最后一个 DD 报文后，发现 RT2 的数据库中有许多 LSA 是自己没有的，将邻居状态机改为 Loading 状态。此时 RT2 也收到了 RT1 的最后一个 DD 报文，但 RT1 的 LSA，RT2 都已经有了，不需要再请求，所以直接将 RT1 的邻居状态机改为 Full 状态。
- RT1 发送 LS Request 报文向 RT2 请求所需要的 LSA。RT2 用 LS Update 报文来回应 RT1 的请求。RT1 收到之后，需要发送 LS Ack 报文来确认。上述过程持续到 RT1 中的 LSA 与 RT2 的 LSA 完全同步为止。此时 RT1 将 RT2 的邻居状态机改为 Full 状态。

🕮 **说明：**

以上过程是两台路由器由相互没有发现对方的存在到建立邻接关系的过程。或者可以理解为网络中新加入一台路由器时的处理情况。当两台路由器之间的状态机都已经达到 Full 状态之后，如果此时网络中再有路由变化时，就无须重复以上的所有步骤。只由一方发送 LS Update 报文通知需要更新的内容，另一方发送 LS Ack 报文予以回应即可。双方的邻居状态机在此过程中不再发生变化。

1.3.12 OSPF 的四种网络类型

OSPF 协议计算路由是以本路由器周边网络的拓扑结构为基础的。每台 OSPF 路由器根据自己周围的网络拓扑结构生成链路状态通告 LSA（Link State Advertisement），并通过更新报文将 LSA 发送给网络中的其它 OSPF 路由器。

根据路由器使用链路层协议的不同，OSPF 将网络分为四种类型：

- 广播（Broadcast）类型：

当链路层协议是 Ethernet、FDDI 时，OSPF 缺省认为网络类型是 Broadcast。在该类型的网络中，通常以组播形式（224.0.0.5 和 224.0.0.6）发送协议报文，选举 DR /BDR。

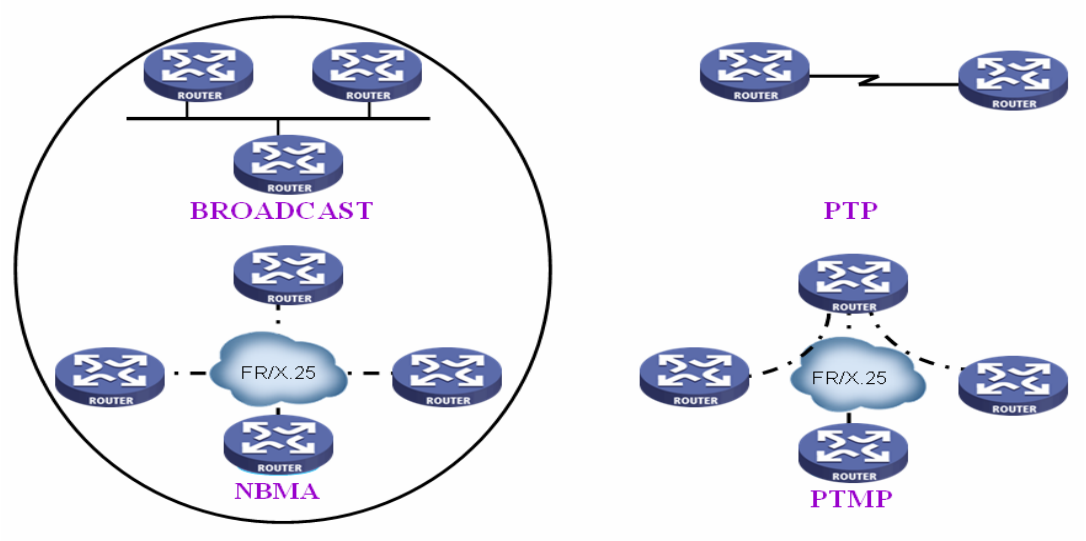


图1-7 OSPF 的四种网络类型

- NBMA（Non-Broadcast Multi-Access）类型：

当链路层协议是帧中继、ATM 或 X.25 时，OSPF 缺省认为网络类型是 NBMA。在该类型的网络中，以单播形式发送协议报文。手工指定邻居，选举 DR/BDR，DR/BDR 要求和 DROTHER 完全互连。

- 点到多点 P2MP（point-to-multipoint）类型：

没有一种链路层协议会被缺省的认为是 Point-to-Multipoint 类型。点到多点必须是由其他的网络类型强制更改的。常用做法是将非全连通的 NBMA 改为点到多点的网络。在该类型的网络中，以组播形式（224.0.0.5）发送协议报文。多播 hello 包自动发现邻居，不要求 DR/BDR 的选举。

- 点到点 P2P（point-to-point）类型：

当链路层协议是 PPP、HDLC 和 LAPB 时，OSPF 缺省认为网络类型是 P2P。在该类型的网络中，以组播形式（224.0.0.5）发送协议报文。无需选举 DR BDR，当只有两个路由器的接口要形成邻接关系的时候才使用。

📖 说明：

在 OSPF 协议中 NBMA 和点到多点都是指非广播多点可达的网络，但 NBMA 网络必须满足全连通（full meshed）的要求，即任意两点都可以不经转发而使报文直达对端。否则，我们称该网络是点到多点网络。

1.3.13 DR（Designated Router）和 BDR（Backup Designated Router）

1. 广播及 NBMA 网段中的 N2 连接问题

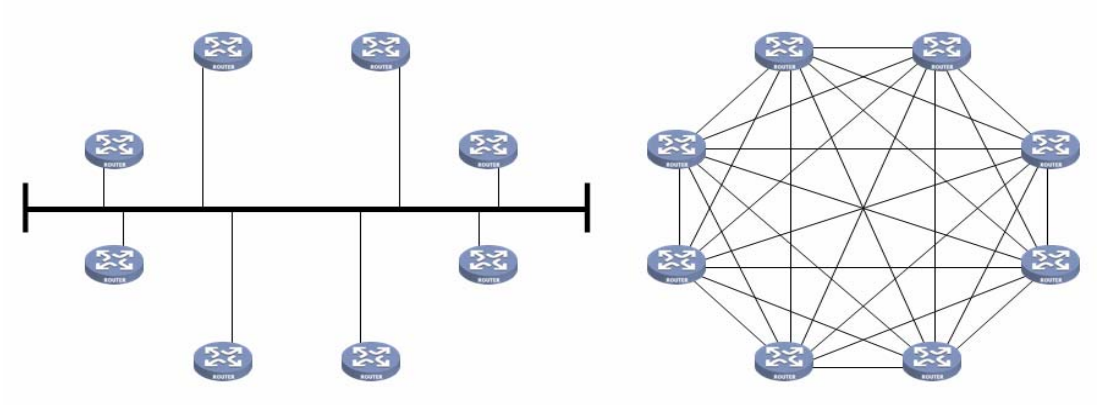


图1-8 广播及 NBMA 网段中的 N2 连接问题

在广播和 NBMA 类型的网络上，任意两台路由器之间都需要传递路由信息（flood），如果网络中有 N 台路由器，则需要建立 N *（N-1）/2 个邻接关系。

任何一台路由器的路由变化，都需要在网段中进行 N*（N-1）/2 次的传递。

这些邻居关系要定期更新链路状态数据库 LSDB，不仅浪费了宝贵的带宽，也会消耗大量的系统资源。应该怎么处理呢？

2. DR（Designated Router）概念的提出：

为了解决这个问题，OSPF 协议指定一台路由器 DR（Designated Router）来负责传递信息。所有的路由器都只将路由信息发送给 DR，再由 DR 将路由信息发送给本网段内的其他路由器。

两台不是 DR 的路由器（DROther）之间不再建立邻接关系，也不再交换任何路由信息。

这样在同一网段内的路由器直谏只需建立 N 个邻接关系，每次路由变化只需进行 2N 次的传递即可。

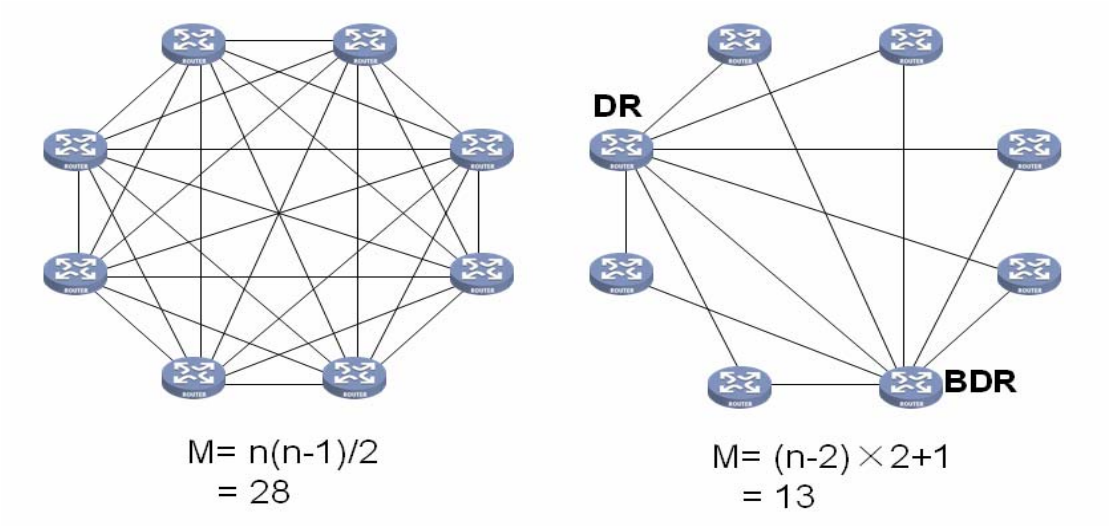


图1-9 DR（Designated Router）概念的提出

如上图，在一个广播的网段中，存在 N＝8 台路由器，则需要建立 M=n(n-1)/2 = 28 个邻接关系。网络中有 每 台路由器，则需要建立 28 个邻接关系。

选举 DR 后，两台不是 DR 的路由器（DROther）之间不再建立邻接关系，也不再交换任何路由信息。

这样在同一网段内的路由器之间只需建立 8 个邻接关系，每次路由变化只需进行 16 次的传递即可。

3. DR（Designated Router）的选举过程：

通过 Hello 报文的所带 priority 位，和 DR、BDR 信息，可以选出该网段的 DR。所有路由器认可一个优先级最高的路由器作为 DR，优先级次高的作为 BDR，所有这个网段的路由器与 DR，BDR 构成邻接关系。

哪台路由器会成为本网段内的 DR 并不是人为指定的，而是由本网段中所有的路由器共同选举出来的。

DR 的选举过程如下：

- 登记选民

  本网段内的运行 OSPF 的路由器；（本村内的 18 岁以上公民）

- 登记候选人

  本网段内的 Priority>0 的 OSPF 路由器；Priority 是接口上的参数，可以配置，缺省值是 1；（本村内的 30 岁以上公民且在本村居住 3 年以上）

- 竞选演说

  一部分 Priority>0 的 OSPF 路由器自己是 DR；（所有的候选人都自认为应该当村长）

- 投票

  Priority 大于 0 的路由器都可作为“候选者”，选票就是 Hello 报文。

  每台路由器将自己选出的 DR 写入 Hello 报文中，发给网段上的其它路由器。

  （选年纪最大若年龄相等按姓氏笔划排序）

当同一网段的两台路由器都宣布自己是 DR 时，选择 Priority 高的。如果 Priority 相等，选择 Router ID 大的。DR 由本网段中所有路由器共同选举。

4. DR 选举中的指导思想

选举制

DR 是各路由器选出来的，而非人工指定的，虽然管理员可以通过配置 priority 干预选举过程。

终身制

DR 一旦当选，除非路由器故障，否则不会更换，即使后来的路由器 priority 更高。

世袭制

DR 选出的同时也选出 BDR 来，DR 故障后，由 BDR 接替 DR 成为新的 DR。

5. 稳定压倒一切:

由于网段中的每台路由器都只和 DR 建立邻接关系。如果 DR 频繁的更迭，则每次都要重新引起本网段内的所有路由器与新的 DR 建立邻接关系。

这样会导致在短时间内网段中有大量的 OSPF 协议报文在传输，降低网络的可用带宽。所以协议中规定应该尽量的减少 DR 的变化。

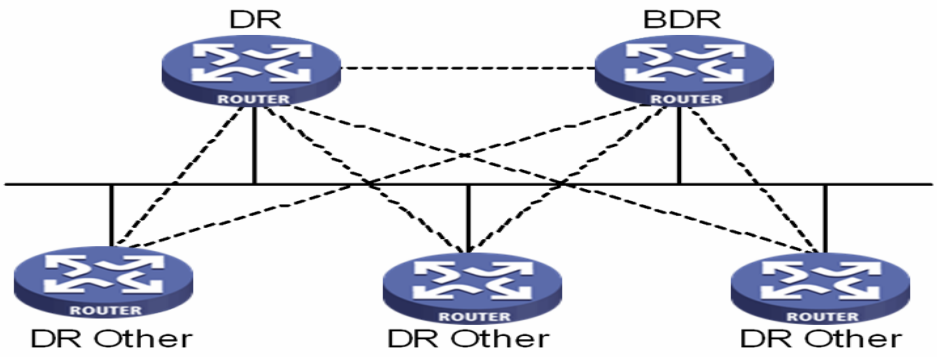


图1-10 DR 和 BDR

具体的处理方法是，每一台新加入的路由器并不急于参加选举，而是先考察一下本网段中是否已有 DR 存在。如果目前网段中已经存在 DR，即使本路由器的 priority 比现有的 DR 还高，也不会再声称自己是 DR 了。而是承认现有的 DR。

6. BDR（Backup Designated Router）

如果 DR 由于某种故障而失效，这时网络中必须重新选举 DR，并与之新的 DR 同步。这需要较长的时间，在这段时间内，路由计算是不正确的。

为了能够缩短这个过程，进行快速响应，OSPF 提出了 BDR（Backup Designated Router）的概念。

与 DR 同时被选举出来。BDR 也与本网段内的所有路由器建立邻接关系并交换路由信息。DR 失效后，BDR 立即成为 DR。

由于不需要重新选举，并且邻接关系已经建立，所以这个过程可以很快完成。

这时，当然还需要重新选举出一个新的 BDR，虽然一样需要较长的时间，但并不会影响路由计算。

7. 选举 DR 和 BDR 的注意事项

1. 只有在广播和 NBMA 类型的接口上才会选举 DR，在 point-to-point 和 point-to-muiltipoint 类型的接口上不需要选举。
2. 路由器接口的优先级 Priority 将影响接口在选举 DR 时所具有的资格。优先级为 0 的路由器不会被选举为 DR 或 BDR。
3. 网段中的 DR 并不一定是 priority 最大的路由器；同理，BDR 也并不一定就是 priority 第二大的路由器。若 DR、BDR 已经选择完毕，即使有一台 Priority 值更大的路由器加入，它也不会成为该网段中的 DR。
4. DR 是指某个网段中概念，是针对路由器的接口而言的。某台路由器在一个接口上可能是 DR，在另一个接口上可能是 BDR，或者是 DROther。
5. 两台 DROther 路由器之间不进行路由信息的交换，但仍旧互相发送 HELLO 报文。他们之间的邻居状态机停留在 2-Way 状态。
6. 在广播的网络上必须存在 DR 才能够正常工作，但 BDR 不是必需的。

📖 **说明：由于 DR 的出现带来协议的变化**

为了减少在广播和 NBMA 网段内带宽的占用，提出了 DR 的概念。为协议本身带来如下变化：

- 将广播和 NBMA 网段内 LSDB 同步的次数由 O(N)2 减少为 O(N)。
- 在广播和 NBMA 网段中，路由器的角色划分为 DR、BDR、DROther。
- 路由器之间的关系分为 Unknown、Neighbor、Adjacency。两台 DROther 路由器之间只建立 Neighbor 关系，邻居状态机停留在 2-Way 状态。DR 及 BDR 与本网段内的所有路由器建立 Adjacency 关系，邻居状态机会达到 Full 状态。
- 增加了一种接口类型：Point-to-Multipoint。
- 增加了一种新的 LSA 类型：Network-LSA，由 DR 生成，描述了本网段的链路状态信息。

1.3.14 NBMA（NonBroadcast MultiAccess）和 PTMP（point-to-multipoint）

1. 为什么 point-to-multipoint 的网络中不能选举 DR

NBMA（NonBroadcast MultiAccess）是指非广播多点可达的网络，比较典型的有 ATM、X.25 和 Frame Relay。在这种网络中，为了减少路由信息的传递次数，需要选举 DR，其他的路由器只与 DR 交换路由信息。

在上述描述中有一个缺省的条件：这个 NBMA 网络必须是全连通的（Full Meshed）。但这在实际情况中并不一定总能得到满足：例如一个 X.25 网络出于花费方面的考虑，并不一定在任何两台路由器之间都建立一条 map；即使是一个全连通的网络，也可能由于故障导致某条 map 中断，使该网络变成不是全连通的。

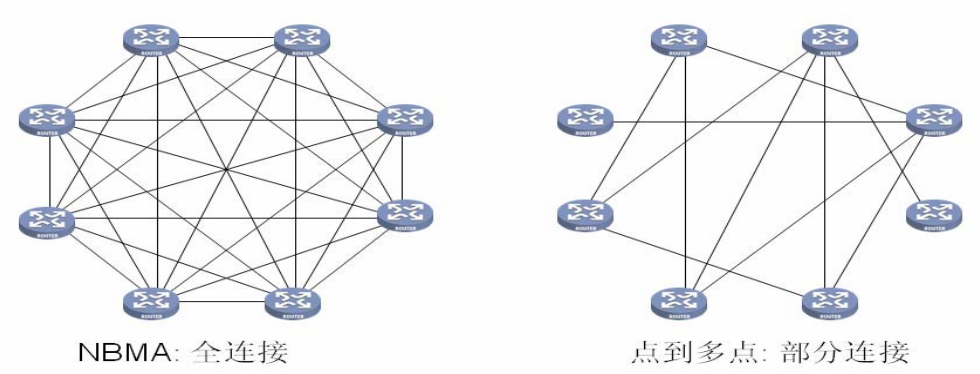


图1-11 为什么 point-to-multipoint 的网络中不能选举 DR

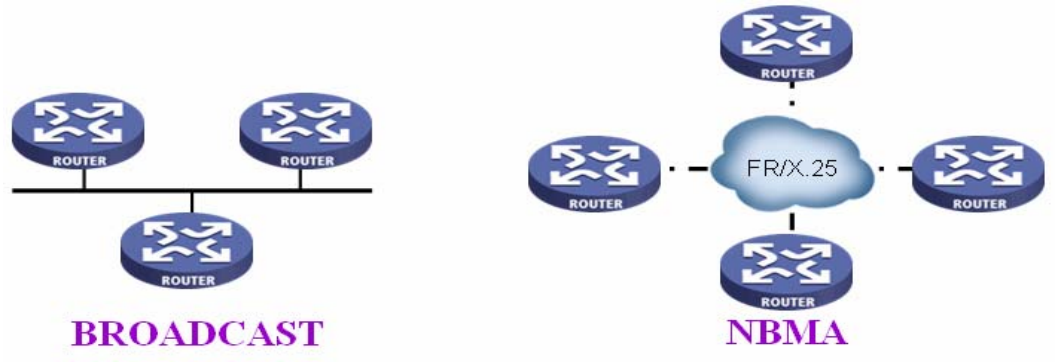
在这种情况下会有什么问题呢？图中是一个非全连通的 X.25 网络，但其中 A、B、E 三者是全连通的，假设 E 被选举为 DR，其他为 DROther（这里先不考虑 BDR）。A、C、D 三者也是全连通的，D 是其中的 DR。由于 D、E 之间不连通，所以 DR 的选举算法不能正确运行，D、E 都坚持宣称自己是 DR。对于 A，则只能根据选举算法确定一个 DR，假设是 E，则 A 与 E 之间交换路由信息。A 不承认 D 是 DR，D 无法与 A 交换路由信息，A，C 之间也无法交换路由信息（两者都是 DROther）。这样 D、C 就无法与网络中其他路由器交换路由信息。导致路由计算不正确。

由上述分析可知：错误产生的原因是因为在非全连通的网络中选举 DR 所至。为了解决这个问题，OSPF 协议定义了一种新的网络类型：point-to-multipoint（点到多点）。点到多点与 NBMA 最本质的区别是：在点到多点的网络中不选举 DR、BDR，即这种类型的网络中任意两台路由器之间都交换路由信息。在上面的图二中 B、C 可以通过 A 与网段中的其他路由器交换路由信息。一个 NBMA 的网络是否是全连通的需要网络管理人员去判断，如果不是，则需要更改配置，将网络的类型改为点到多点。

2. NBMA 网络的配置原则

对于接口类型为 NBMA 的网络需要进行一些特殊的配置。由于无法通过广播 Hello 报文的形式发现相邻路由器，必须手工为该接口指定相邻路由器的 IP 地址，以及该相邻路由器是否有选举权等。可通过配置轮询间隔来指定路由器在与相邻路由器构成邻接关系之前发送轮询 Hello 报文的时间周期。

NBMA 网络必须是全连通的，即网络中任意两台路由器之间都必须直接可达。如果部分路由器之间没有直接可达的链路时，应将接口配置成 p2mp 方式。如果路由器在 NBMA 网络中只有一个对端，也可将接口类型改为 p2p 方式。



3. NBMA 与 p2mp 之间的区别：

NBMA 与 p2mp 之间的区别：

- 在 OSPF 协议中 NBMA 是指那些全连通的、非广播、多点可达网络。而点到多点的网络，则并不需要一定是全连通的。
- 在 NBMA 上需要选举 DR 与 BDR，而在点到多点网络中没有 DR 与 BDR。
- NBMA 是一种缺省的网络类型，例如：如果链路层协议是 ATM，OSPF 会缺省的认为该接口的网络类型是 NBMA（不论该网络是否全连通）。点到多点不是缺省的网络类型，没有哪种链路层协议会被认为是点到多点，点到多点必须是由其它的网络类型强制更改的。最常见的做法是将非全连通的 NBMA 改为点到多点的网络。
- NBMA 用单播发送协议报文，需要手工配置邻居。点到多点是可选的，即可以用单播发送，又可以用多播发送报文。

1.4 OSPF Multi-Area 原理

1.4.1 Autonomous System 、Area 和 Area ID

OSPF 路由协议是一种典型的链路状态（Link-state）的路由协议，一般用于同一个路由域内。多个路由域 组成了一个自治系统（Autonomous System），即 AS。

- 自治系统(autonomy system)：

由同一机构管理，使用同一组选路策略的路由器的集合，包括一个单独的管理实体下所控制的一组路由器，缩写为 AS。

例如，属于一个特定公司的所有路由器，同时这些路由器上也可能运行了多种的路由协议。

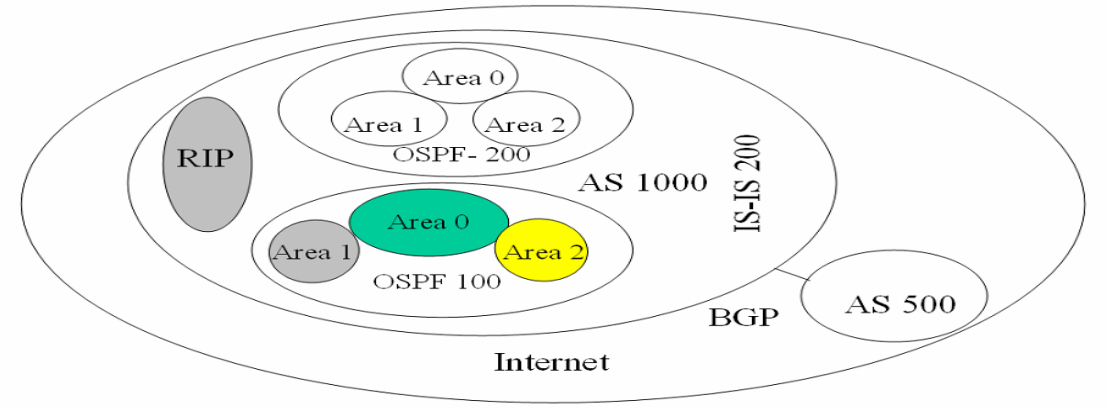


图1-12 自治系统，区域和区域 ID

- 区域（Area）：

区域是指一个路由器的集合，相同的区域有着相同的拓扑结构数据库。 OSPF 用区域把一个 A S 分成多个链路状态域，因为一个区域的拓扑结构对另一个区域是不可见的。

- 区域 ID（Area ID）

区域号用一个 32bit 的整数来标识，可以定义为 IP address 格式，也可以用一个十进制整数表示(ie. Area 0.0.0.0, or Area 0)。

其中，区域 0.0.0.0 保留为骨干区域，非骨干区域一定要连接到骨干区域。

🕮 注意：

图中 OSPF 100、OSPF 200 指的是 OSPF 的进程号

1.4.2 区域划分

1. 为什么需要划分区域：

随着网络规模日益扩大，网络中的路由器数量不断增加。当一个巨型网络中的路由器都运行 OSPF 路由协议时，就会遇到如下问题：

- 每台路由器都保留着整个网络中其他所有路由器生成的 LSA，这些 LSA 的集合组成 LSDB，路由器数量的增多会导致 LSDB 非常庞大，这会占用大量的存储空间。
- LSDB 的庞大会增加运行 SPF 算法的复杂度，导致 CPU 负担很重。
- 由于 LSDB 很大，两台路由器之间达到 LSDB 同步会需要很长时间。
- 网络规模增大之后，拓扑结构发生变化的概率也增大，网络会经常处于“动荡”之中，为了同步这种变化，网络中会有大量的 OSPF 协议报文在传递，降低了网络的带宽利用率。更糟糕的是：每一次变化都会导致网络中所有的路由器重新进行路由计算。

解决上述问题的关键主要有两点：减少 LSA 的数量；屏蔽网络变化波及的范围。

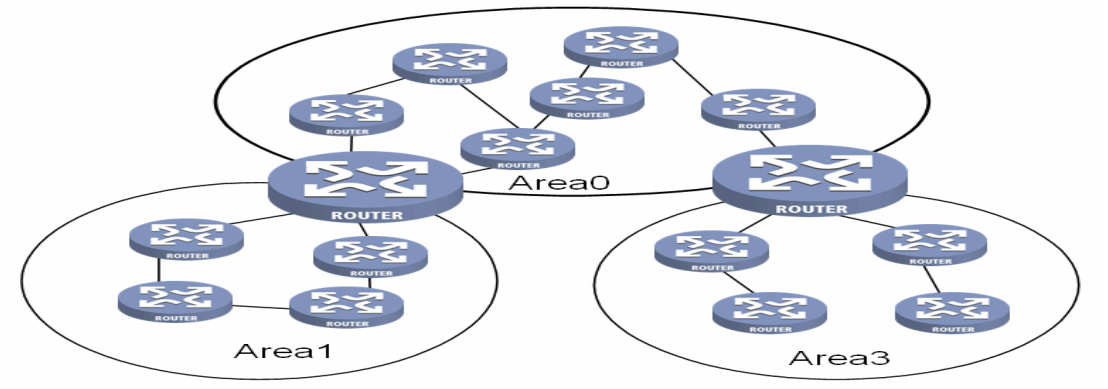


图1-13 多区域的 OSPF 网络

OSPF 协议通过将自治系统划分成不同的区域（Area）来解决上述问题。区域是在逻辑上将路由器划分为不同的组。区域的边界是路由器，这样会有一些路由器属于不同的区域，（这样的路由器称作区域边界路由器——ABR），而一个网段只能属于一个区域。

划分成区域之后，给 OSPF 协议的处理带来了很大的变化。

每一个网段必须属于一个区域，或者说每个运行 OSPF 协议的接口必须指名属于某一个特定的区域，区域用区域号（Area ID）来标识。区域号是一个从 0 开始的 32 位整数。不同的区域之间通过 ABR 来传递路由信息。

2. 区域间路由计算的变化：

OSPF 将自治系统划分为不同的区域后，路由计算方法也发生了很多变化：

• 只有同一个区域内的路由器之间会保持 LSDB 的同步，网络拓扑结构的变化首先在区域内更新。

• 区域之间的路由计算是通过 ABR 来完成的。ABR 首先完成一个区域内的路由计算，然后查询路由表，为每一条 OSPF 路由生成一条 Type3 类型的 LSA，内容主要包括该条路由的目的地址、掩码、花费等信息。然后将这些 LSA 发送到另一个区域中。

• 在另一个区域中的路由器根据每一条 Type3 的 LSA 生成一条路由，由于这些路由信息都是由 ABR 发布的，所以这些路由的下一跳都指向该 ABR。

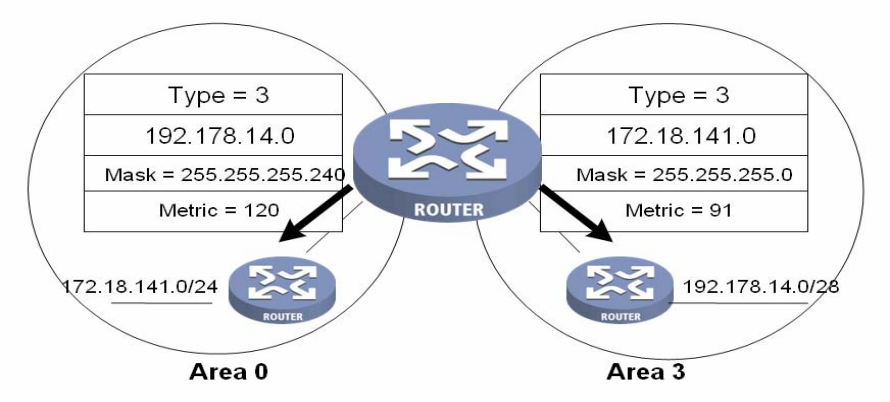


图1-14 区域间路由计算的变化

3. 划分区域后的好处：

• 由于划分区域后 ABR 是根据本区域内的路由生成 LSA，则可以根据 IP 地址的规律先将这些路由进行聚合后再生成 LSA，这样做可以大大减少自治系统中 LSA 的数量。

• 划分区域之后，网络拓扑的变化首先在区域内进行同步，如果该变化影响到聚合之后的路由，则才会由 ABR 将该变化通知到其他区域。大部分的拓扑结构变化都会被屏蔽在区域之内了。

📖 说明：

OSPF 在区域内计算路由时，使用的是链路状态算法。
但当 OSPF 划分为不同的区域之后，ABR 通过将区域内已计算好的路由封装成 Type3 类型的 LSA 发送出去，此时的 OSPF 计算区域间路由时使用的是 D-V 算法。
相关内容，我们会在的后面详细讲解。

1.4.3 路由器的类型

OSPF 路由器根据在 AS 中的不同位置，可以分为以下四类：

- IAR（Internal Area Router）：

区域内路由器，是指该路由器的所有接口都属于同一个 OSPF 区域。这种路由器只生成一条 Router LSA，只保存一个 LSDB。

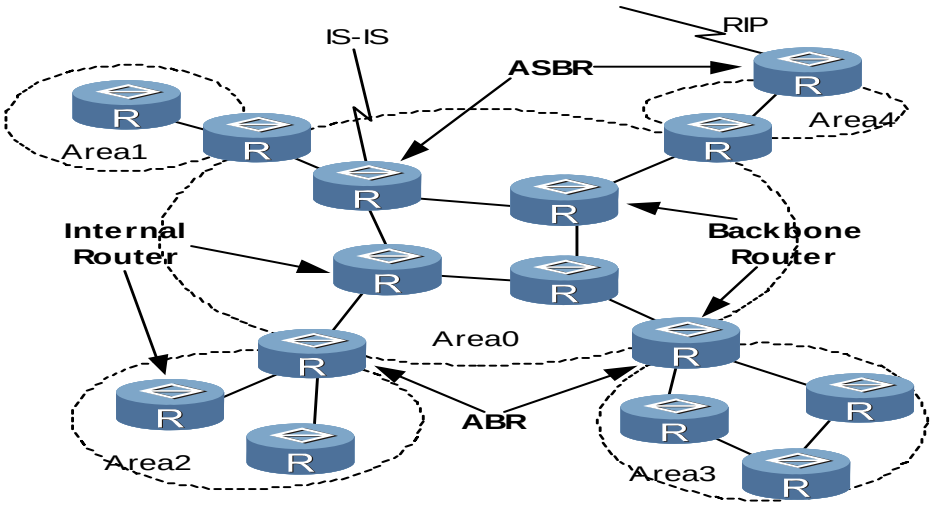


图1-15 OSPF 路由器的类型

- ABR（Area Border Router）：

区域边界路由器，该路由器同时属于两个以上的区域（其中必须有一个是骨干区域，也就是区域 0）。

该类路由器可以同时属于两个以上的区域，但其中一个必须是骨干区域。ABR 用来连接骨干区域和非骨干区域，它与骨干区域之间既可以是物理连接，也可以是逻辑上的连接。

该路由器为每一个所属的区域生成一条 Router LSA，为每一个所属的区域保存一个 LSDB。并根据需要生成 Network Summary LSA（Type = 3）和 ASBR Summary LSA（Type = 4）。

- BBR （BackBone Router）：

该类路由器至少有一个接口属于骨干区域（也就是 0 区域）。因此，所有的 ABR 和位于 Area0 的内部路由器都是骨干路由器。

- ASBR（AS Boundary Router）：

自治系统边界路由器，是指该路由器引入了其他路由协议（也包括静态路由和接口的直接路由）发现的路由。需要注意的是 ASBR 并不一定在拓扑结构中位于自治系统的边界。ASBR 生成 AS External LSA（Type = 5）。

ASBR 并不一定位于 AS 的边界，它有可能是区域内路由器，也有可能是 ABR。只要一台 OSPF 路由器引入了外部路由的信息，它就成为 ASBR。

🕮 **说明：**

路由器的各种类型之间是可以“兼职”的（除了不能同时是 IAR 和 ABR），例如：一台路由器可以同时是 IAR、BBR、ASBR。而接口状态是指路由器的其中一个接口的状态，两者之间没有任何关系。

1.4.4 骨干区域与虚连接

1. **为何需要骨干区域：**

OSPF 划分区域之后，并非所有的区域都是平等的关系。其中有一个区域是与众不同的，它的区域号（Area ID）是 0，通常被称为骨干区域（Backbone Area）。

由于划分区域之后，区域之间是通过 ABR 将一个区域内的已计算出的路由封装成 Type3 类的 LSA 发送到另一个区域之中来传递路由信息。需要注意的是：此时的 LSA 中包含的已不再是链路状态信息，而是纯粹的路由信息了。

或者说，此时的 OSPF 是基于 D-V 算法，而不是基于链路状态算法的了。这就涉及到一个很重要的问题：路由自环。因为 D-V 算法无法保证消除路由自环。如果无法解决这个问题，则区域概念的提出就是失败的。

通过分析 D-V 算法中路由环的产生的原因可知，自环的产生主要是因为生成该条路由信息的路由器没有加入生成者的信息，即每一条路由信息都无法知道最初是由谁所生成。OSPF 协议在生成 LSA 时首先将自己的 Router ID 加入到 LSA 中，但是如果该路由信息传递超过两个区域后，就会丧失最初的生成者的信息。

解决的方法是：所有 ABR 将本区域内的路由信息封装成 LSA 后，统一的发送给一个特定的区域，再由该区域将这些信息转发给其他区域。在这个特定区域内，每一条 LSA 都确切的知道生成者信息。在其他区域内所有的到区域外的路由都会发送到这个特定区域中，所以就不会产生路由自环。

这个“特定区域”就是骨干区域。由上面的分析可知：所有的区域必须和骨干区域相连，也就是说，每一个 ABR 连接的区域中至少有一个是骨干区域。而且骨干区域自身也必须是连通的。

骨干区域负责区域之间的路由，非骨干区域之间的路由信息必须通过骨干区域来转发。对此，OSPF 有两个规定：

- 所有非骨干区域必须与骨干区域保持连通；
- 骨干区域自身也必须保持连通。

2. 虚连接：

但在实际应用中，可能会因为各方面条件的限制，无法满足 OSPF 规定的这两个要求。这时可以通过配置 OSPF 虚连接予以解决。

虚连接是指在两台 ABR 之间通过一个非骨干区域而建立的一条逻辑上的连接通道。它的两端必须是 ABR，而且必须在两端同时配置方可生效。为虚连接两端提供一条非骨干区域内部路由的区域称为运输区域（Transit Area）。

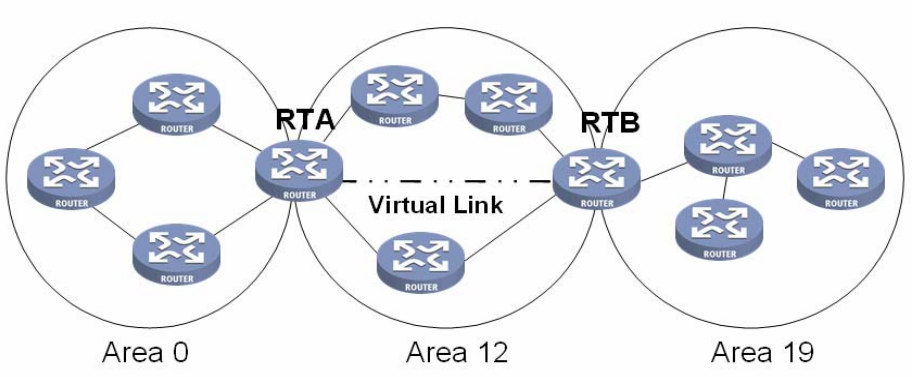


图1-16 虚连接示意图之一

- 当非骨干区域与 Area 0 之间没有直连链路时

由于网络的拓扑结构复杂，有时无法满足每个区域必须和骨干区域直接相连的要求，例如图中的 Area 19。在上图中，Area19 与骨干区域之间没有直接相连的物理链路，但可以在 ABR 上配置虚连接，使 Area12 通过一条逻辑链路与骨干区域保持连通。

OSPF 通过提出虚连接的概念解决此问题。虚连接在 RTA 和 RTB 两台 ABR 之间，穿过一个非骨干区域 Area 12（转换区域——transit Area），建立的一条逻辑上的连接通道。可以理解为两台 ABR 之间存在一个点对点的连接。

“逻辑通道”是指两台 ABR 之间的多台运行 OSPF 的路由器只是起到一个转发报文的作用（由于协议报文的目的地址不是这些路由器，所以这些报文对于它们是透明的，只是当作普通的 IP 报文来转发），两台 ABR 之间直接传递路由信息。这里的路由信息是指由 ABR 生成的 type3 的 LSA，区域内的路由器同步方式没有因此改变。

虚连接相当于在两个 ABR 之间形成了一个点到点的连接，因此，在这个连接上，和物理接口一样可以配置接口的各参数，如发送 HELLO 报文间隔等。

- 当骨干区域不连续时

OSPF 路由协议要求骨干区域 Area 0 必须是连续的，但是，骨干区域也会出现不连续的情况。例如，当我们想把两个运行 OSPF 路由协议的网络混合到一起，并且想要使用一个骨干区域时；或者当某些路由器出现故障引起骨干区域不连续的情况。在这些情况下，我们可以采用虚连接将两个不连续的 Area 0 连接到一起。这时，虚拟链路的两端必须是两个 Area 0 的边界路由器，并且这两个路由器必须都有处于同一个区域的端口。

虚连接的在实际应用中主要是提供冗余的备份链路，当骨干区域因链路故障将被分割时，通过虚连接仍然可以保证骨干区域在逻辑上的连通性。如下图所示。

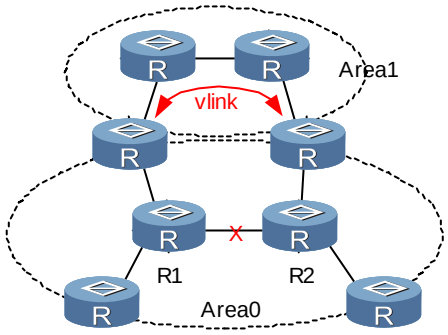


图1-17 虚连接示意图之二

上图为我们提供了一种通过虚链接提高网络可靠性的方法。Area 1 与骨干区域通过两个 ABR 连接，但是当 Area 0 中的 R1 和 R2 物理链路断开时，骨干区域会因链路故障将被分割为两半。如果我们在 Area 1 的两个 ABR 上配置了虚链接，当 R1 和 R2 物理连接失效后，那么 Area 0 通过虚链接经过 Area 1 同样可以保持骨干区域的连续性。

📖 注意：：

如果自治系统被划分成一个以上的区域，则必须有一个区域是骨干区域，并且保证其它区域与骨干区域直接相连或逻辑上相连，且骨干区域自身也必须是连通的。

1.4.5 与自治系统外部通讯

- 自治系统：

OSPF 是自治系统内部路由协议，负责计算同一个自治系统内的路由。在这里“自治系统”是指彼此相连的运行 OSPF 路由协议的所有路由器的集合。对于 OSPF 来说，整个网络只有“自治系统内”和“自治系统外”之分。需要注意的是：“自治系统外”并不一定在物理上或拓扑结构中真正的位于自治系统的外部，而是指那些没有运行 OSPF 的路由器或者是某台运行 OSPF 协议的路由器中没有运行 OSPF 的接口。

- ASBR（Autonomous System Boundary Router）：

作为一个 IGP，OSPF 同样需要了解自治系统外部的路由信息，这些信息是通过 ASBR（自治系统边界路由器）获得的，ASBR 是那些将其他路由协议（也包括静态路由和接口的直接路由）发现的路由引入（import）到 OSPF 中的路由器。同样需要注意的是：ASBR 并不一定真的位于 AS 的边界，而是可以在自治系统中的任何位置。

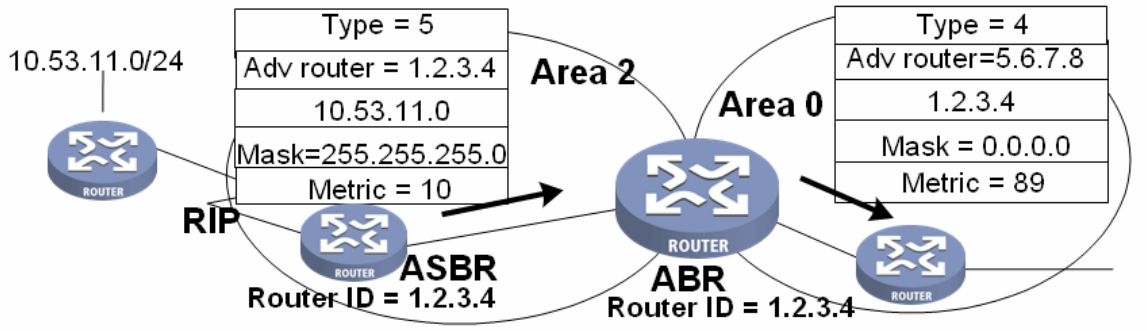


图1-18 自治系统外部路由计算

- 计算自治系统外部路由：

ASBR 为每一条引入的路由生成一条 Type5 类型的 LSA，主要内容包括该条路由的目的地址、掩码和花费等信息。这些路由信息将在整个自治系统中传播（STUB AREA 除外）。计算路由时先在最短路径树中找到 ASBR 的位置，然后将所有由该 ASBR 生成的 Type5 类型的 LSA 都当作叶子节点挂在 ASBR 的下面。

以上的方法在区域内部是可行的，但是由于划分区域的原因，与该 ASBR 不处于同一个区域的路由器计算路由时无法知道 ASBR 的确切位置（该信息被 ABR 给过滤掉了，因为 ABR 是根据区域内的已生成的路由再生成 Type3 类型的 LSA）。

为了解决这个问题，协议规定如下：如果某个区域内有 ASBR，则这个区域的 ABR 在向其他区域生成路由信息时必须单独为这个 ASBR 生成一条 Type 4 类型的 LSA，内容主要包括这个 ASBR 的 Router ID 和到他所需的花费值。

1.4.6 OSPF 的路由类型和优选顺序：

AS 区域内和区域间路由描述的是 AS 内部的网络结构，外部路由则描述了应该如何选择到 AS 以外目的地址的路由。OSPF 将引入的 AS 外部路由分为两类：Type1 和 Type2。

第一类外部路由是指接收的是 IGP 路由（例如静态路由和 RIP 路由）。由于这类路由的可信程度高一些，所以计算出的外部路由的开销与自治系统内部的路由开销是相同的，并且和 OSPF 自身路由的开销具有可比性，即到第一类外部路由的开销=本路由器到相应的 ASBR 的开销+ASBR 到该路由目的地址的开销。

第二类外部路由是指接收的是 EGP 路由。由于这类路由的可信度比较低，所以 OSPF 协议认为从 ASBR 到自治系统之外的开销远远大于在自治系统之内到达 ASBR 的开销。所以计算路由开销时将主要考虑前者，即到第二类外部路由的开销=ASBR 到该路由目的地址的开销。如果两条路由计算出的开销值相等，再考虑本路由器到相应的 ASBR 的开销

如果加上前面所述的两种路由类型，OSPF 一共将路由分为四级，按优先级从高到低排列：

- 优选区域内的路由

  同为区域内的路由则比较 Cost 值，小的优先

- 优选区域间的路由

  同为区域间的路由则优选通过骨干区域的，然后比较 Cost 值，小的优先。

- 优选自治系统 Type1 类外部路由

  同为 Type1 类的路由，则比较（Type1 类路由的 Cost＋到发布该路由的 ASBR 的自治系统内部的 Cost）之和，小的优先。

- 优选自治系统 Type2 类外部路由

  同为 Type2 类的路由，则比较 Type2 类路由的 Cost，小的优先，如果相等，则比较到发布该路由的 ASBR 的自治系统内部路由的 Cost，小的优先。

- 若都相等，则填加等值路由。

📖 注意：：

其中前两种路由在路由表中的优先级是一样的，缺省值为 10; 后两种路由在路由表中的优先级是相同的，缺省值是 150。

1.4.7 OSPF 与路由自环

路由自环是指到某一个目的地址的路由在网络中形成了环路，一个最简单的例子：路由器 A 上有一条路由 10.0.0.0/8 下一跳是路由器 B；而在路由器 B 的路由表中该路由的下一跳指向 A；如果 A 收到一条到 10.0.0.1 的报文，它会转发给 B，而 B 根据路由表又将该报文转发给 A。于是该报文会在 A、B 之间不停的震荡，直至 TTL=0 才会将该报文丢弃，最坏的情况可能会震荡 255 次。

路由自环对网络的危害是极大的，不仅导致路由不可达，而且浪费了大量的网络的带宽。路由自环是所有路由协议必须解决的问题，也是衡量一个路由协议好坏的重要标志。

1. D-V 算法与路由自环：

D-V 算法又称为“距离—矢量”算法，其核心思想是网络中的每台路由器都将自己已知的路由表发送给相邻的路由器，每台路由器都会根据收到的所有路由确定最优路径的下一跳。

这种算法（主要指 RIP 所实现的 D-V 算法）的缺陷是：

- 每台路由器对接收到的路由的可信度完全依赖于相邻的路由器。而一台路由器只能保证自己本地路由的正确性（指本路由器的接口路由），而对由其他路由器发送来的路由则无法保证。

- 每一条路由信息中没有标明生成者的信息，该路由信息经过网络中多次传递之后，可能被传回给最初的生成者，而生成者无法知道该信息是否由自己所发布，这就为自环的产生埋下了隐患。

当网络拓扑结构发生变化时，一条已经无效的路由在未能彻底清除之前，可能仍旧在网络中传递，当它传递给该路由的生成者时，此时下一跳的计算可能会发生错误，导致路由自环的产生。

2. OSPF 与路由自环：

OSPF 是一种基于链路状态算法的协议，其核心思想是：每一台路由器将自己周边的链路状态（包括接口的直接路由、相连的路由器等信息）描述出来，发送给网络中所有的路由器。每台路由器在收到其他所有路由器的发送的链路状态信息之后，运行 SPF 算法计算路由。

OSPF 计算出的路由不会有自环，主要有以下原因：

- 每台路由器描述的是自己能够确保正确的信息——自己周边的网络拓扑结构。并且在生成的 LSA 中标记了该信息的生成者——写入自己的 Router ID。其他的路由器只负责在网络中传输该信息，而不会有任何的更改。这一点保证了无论网络的拓扑结构如何，无论路由器位于网络中的什么位置，都可以准确无误的接收到全网的拓扑结构图。

• 路由计算的算法是 SPF 算法。计算的结果是一棵树，路由是树上的叶子节点。从根节点到叶子节点是单向不可回复的路径。

• 当网络的拓扑结构发生变化时（此时最易产生路由自环），会有一台（或多台）路由器感知到这一变化，重新描述网络拓扑结构，并将其通知给其他路由器。每个路由器接收到更新信息后，都会立即重新运行 SPF 算法，得到新的路由。

OSPF 真的没有路由自环吗？

上文中描述的是指没有划分区域时的情况，或者说是在同一区域内 OSPF 的运行情况。当 OSPF 划分为不同的区域之后，ABR 通过将区域内已计算好的路由封装成 Type3 类型的 LSA 发送出去。

需要特别注意的是：此时 ABR 只是单纯的对路由进行描述，而不是描述链路的状态了。或者说，此时的 OSPF 计算区域间路由时使用的是 D-V 算法。这时就可能会产生路由自环，OSPF 通过骨干区域解决了这个问题（详情见“骨干区域与虚连接”）。

但是当 OSPF 引入自治系统外部路由时，ASBR 封装在 Type5 类型的 LSA 中的同样是路由信息，而不是链路状态。这时也同样会产生路由自环。这一次 OSPF 没有采取任何措施来避免自环的生成，因为引入的自治系统外部路由其本身就是不可靠的。它可能来源于 RIP 等不可靠的路由，或者是配置错误的静态路由等等，可能这些路由本身就存在路由环。

划分区域时区域内计算出的路由都是正确无误的，所以应极力避免由于划分区域而生成新的环路。而引入的外部路由由于其来源即不可靠，所以就没必要再进行其他的操作了。

“OSPF 协议不会产生路由自环”这句话的严格定义应该是：OSPF 协议生成的自治系统内部路由是无自环的，引入的自治系统外部路由则无法保证。

1.4.8 OSPF 协议与 RIP 路由协议的比较

我们已经知道了 OSPF 路由协议是一种链路状态的路由协议，为了更好地说明 OSPF 路由协议的基本特征，我们将 OSPF 路由协议与距离矢量路由协议之一的 RIP（Routing Information Protocol）作一比较，归纳为如下几点：

- RIP 路由协议中用于表示目的网络远近的唯一参数为跳（HOP），也即到达目的网络所要经过的路由器个数。在 RIP 路由协议中，该参数被限制为最大 15，也就是说 RIP 路由信息最多能传递至第 16 个路由器；对于 OSPF 路由协议，路由表中表示目的网络的参数为 Cost，该参数为一虚拟值，与网络中链路的带宽等相关。也就是说 OSPF 路由信息不受物理跳数的限制，在某些特殊情况下 RIP 可能会产生环路，生成次优路由。因此，OSPF 比较适合应用于大型网络中。

- RIPv1 路由协议不支持变长子网屏蔽码（VLSM），这被认为是 RIP 路由协议不适用于大型网络的又一重要原因。采用变长子网屏蔽码可以在最大限度上节约 IP 地址。OSPF 路由协议不仅支持 VLSM，而且支持 CIDR。

- RIP 路由协议路由收敛较慢。RIP 路由协议周期性地将整个路由表作为路由信息广播至网络中，该广播周期为 30 秒。在一个较为大型的网络中，RIP 协议会产生很大的广播信息，占用较多的网络带宽资源；并且由于 RIP 协议 30 秒的广播周期，影响了 RIP 路由协议的收敛，甚至出现不收敛的现象。当网络状态比较稳定时，网络中传递的链路状态信息是比较少的，或者可以说，当网络稳定时，网络中是比较安静的。因此，OSPF 路由协议即使是在大型网络中也能够较快地收敛，这也正是链路状态路由协议区别与距离矢量路由协议的一大特点。

- 在 RIP 协议中，网络是一个平面的概念，并无区域及边界等的定义。在 OSPF 路由协议中，一个网络，或者说是一个路由域可以划分为很多个区域 area，每一个区域通过 OSPF 边界路由器相连，区域间可以通过路由汇总（Summary）来减少路由信息，减小路由表，提高路由器的运算速度。

- OSPF 路由协议支持路由验证，只有互相通过路由验证的路由器之间才能交换路由信息。并且 OSPF 可以对不同的区域定义不同的验证方式，提高网络的安全性。

- OSPF 路由协议对负载分担的支持性能较好。OSPF 路由协议支持多条 Cost 相同的链路上的负载分担，目前我们 VRP3.4 平台的路由器支持 3 条链路的负载分担。

- OSPF 占用的实际链路带宽比 RIP 少，因为它的路由表是有选择的广播（只在建立邻接关系的路由器间），OSPF 使用的 CUP 时间比 RIP 少，因为 OSPF 达到平衡后的主要工作只是发 HELLO 报文，RIP 是发送路由表，OSPF 用的路由器的内存比 RIP 大，因为 OSPF 相对有一个大的路由表。

1.5 OSPF 的报文格式详解

OSPF 用 IP 报文直接封装协议报文，协议号为 89。一个比较完整的 OSPF 报文（以 LSU 报文为例）结构如下图所示：

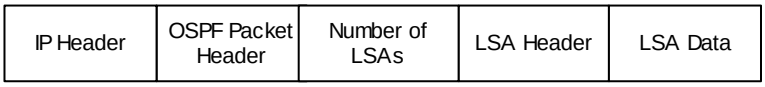


图1-19 OSPF 报文结构

1.5.2 OSPF 报文头

OSPF 有五种报文类型，他们有相同的报文头。如下图所示。

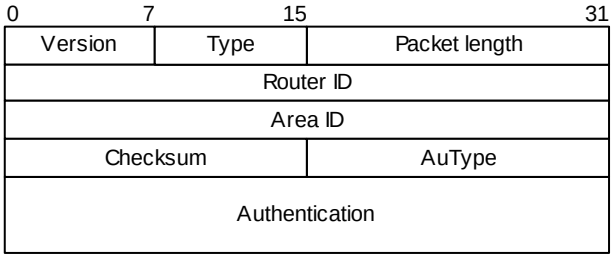


图1-20 OSPF 报文头格式

主要字段的解释如下：

- Version：OSPF 的版本号。对于 OSPFv2 来说，其值为 2。
- Type：OSPF 报文的类型。数值从 1 到 5，分别对应 Hello 报文、DD 报文、LSR 报文、LSU 报文和 LSAck 报文。
- Packet length：OSPF 报文的总长度，包括报文头在内，单位为字节。
- AuType：验证类型。可分为不验证、简单验证和 MD5 验证，其值分别为 0、1、2。
- Authentication：其数值根据验证类型而定。当验证类型为 0 时未作定义，为 1 时此字段为密码信息，类型为 2 时此字段包括 Key ID、MD5 验证数据长度和序列号的信息。

📖 **说明：**

MD5 验证数据添加在 OSPF 报文后面，不包含在 Authenticaiton 字段中。

1.5.3 Hello 报文（Hello Packet）：

最常用的一种报文，周期性的发送给本路由器的邻居。内容包括一些定时器的数值、DR、BDR 以及自己已知的邻居。Hello 报文格式如下图所示。

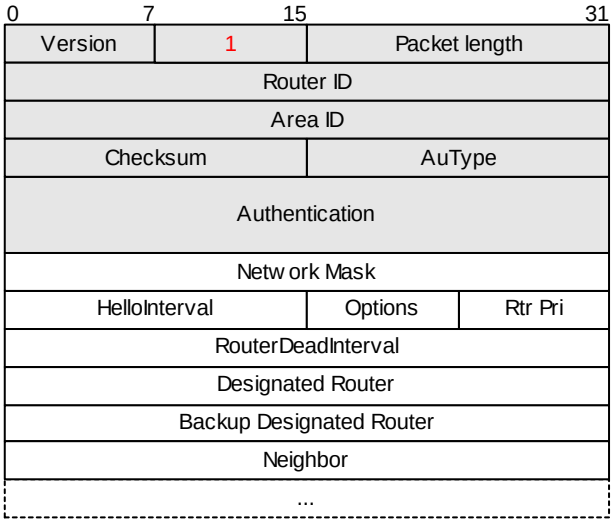


图1-21 Hello 报文格式

主要字段解释如下：

- Network Mask：发送 Hello 报文的接口所在网络的掩码。
- HelloInterval：发送 Hello 报文的时间间隔。如果相邻两台路由器的 Hello 间隔时间不同，则不能建立邻居关系。
- Rtr Pri：DR 优先级。如果设置为 0，则路由器不能成为 DR/BDR。
- RouterDeadInterval：失效时间。如果在此时间内未收到邻居发来的 Hello 报文，则认为邻居失效。如果相邻两台路由器的失效时间不同，则不能建立邻居关系。

1.5.4 DD 报文（Database Description Packet）：

两台路由器进行数据库同步时，用 DD 报文来描述自己的 LSDB，内容包括 LSDB 中每一条 LSA 的 Header（LSA 的 Header 可以唯一标识一条 LSA）。LSA Header 只占一条 LSA 的整个数据量的一小部分，这样可以减少路由器之间的协议报文流量，对端路由器根据 LSA Header 就可以判断出是否已有这条 LSA。

DD 报文格式如下图所示。

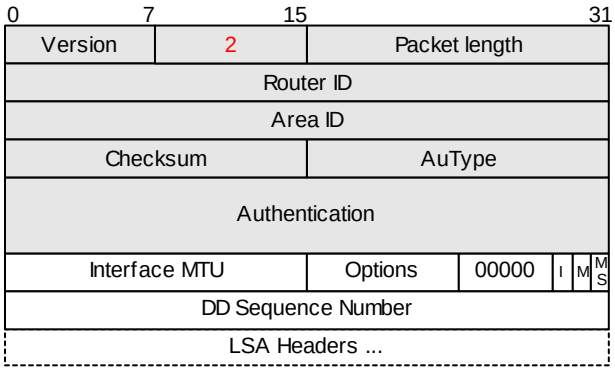


图1-22 DD 报文格式

主要字段的解释如下：

- Interface MTU：在不分片的情况下，此接口最大可发出的 IP 报文长度。
- I（Initial）：当发送连续多个 DD 报文时，如果这是第一个 DD 报文，则置为 1，否则置为 0。
- M（More）：当发送连续多个 DD 报文时，如果这是最后一个 DD 报文，则置为 0。否则置为 1，表示后面还有其他的 DD 报文。
- MS（Master/Slave）：当两台 OSPF 路由器交换 DD 报文时，首先需要确定双方的主从关系，Router ID 大的一方会成为 Master。当值为 1 时表示发送方为 Master。
- DD Sequence Number：DD 报文序列号，由 Master 方规定起始序列号，每发送一个 DD 报文序列号加 1，Slave 方使用 Master 的序列号作为确认。主从双方利用序列号来保证 DD 报文传输的可靠性和完整性。

1.5.5 LSR 报文（Link State Request Packet）：

两台路由器互相交换过 DD 报文之后，知道对端的路由器有哪些 LSA 是本地的 LSDB 所缺少的，这时需要发送 LSR 报文向对方请求所需的 LSA。内容包括所需要的 LSA 的摘要。LSR 报文格式如下图所示。

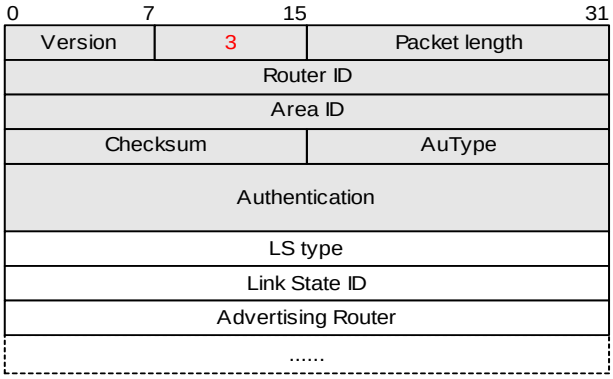


图1-23 LSR 报文格式

主要字段解释如下：

- LS type：LSA 的类型号。例如 Type1 表示 Router LSA。
- Link State ID：即 LSA 头格式中的字段，根据 LSA 的类型而定。
- Advertising Router：产生此 LSA 的路由器的 Router ID。

1.5.6 LSU 报文（Link State Update Packet）：

用来向对端路由器发送所需要的 LSA，内容是多条 LSA（全部内容）的集合。LSU 报文格式如下图所示。

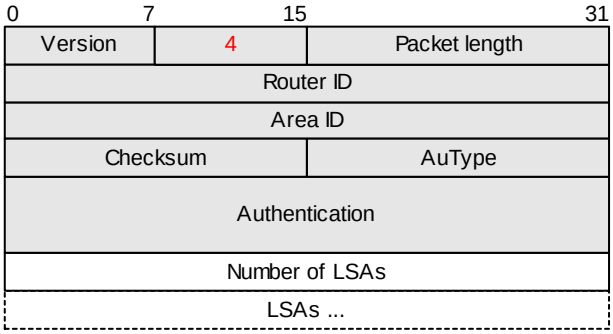


图1-24 LSU 报文格式

1.5.7 LSAck 报文（Link State Acknowledgment Packet）

用来对接收到的 LSU 报文进行确认。内容是需要确认的 LSA 的 Header（一个 LSAck 报文可对多个 LSA 进行确认）。报文格式如下图所示。

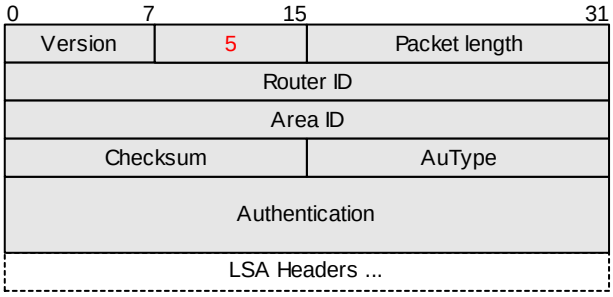


图1-25 LSAck 报文格式

1.5.8 LSA 头部

首先，我们分析一下 LSA 报文头：

所有 LSA 头部都有 20 个字节，它包含了足够的信息来唯一标识一条 LSA（LS type，Link State ID，and Advertising Router ）。LSA 多实例在同一时候可以存在于路由域中，它被用来决定哪一个实例是最新的。LSA 头部还包括 LS 老化、LS 序列号和 LS 校验和等字段。

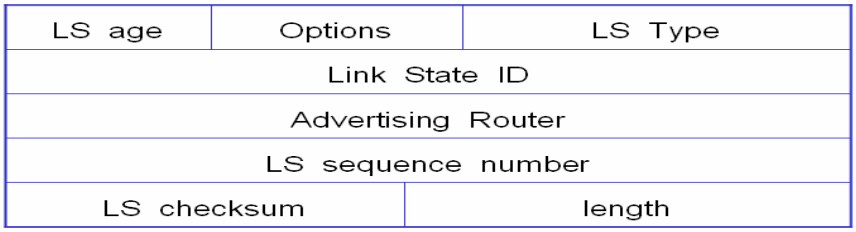


图1-26 LSA 头部

LS age，用来标识 LS 产生的时间。生成 LSA 的路由器将 LS 时域初始化为 0，在洪泛过程中，每经过一个路由器，要按 InfTransDelay 的量增加，这个量表示传输 LSA 到下一个跳所需要的时间。当该时间达到所设定的 MaxAge 参数时，要撤消该 LSA。

Options，用来描述支持的路由域，主要包括 DC、EA、N/P、MC、E、T 等选项。DC 指的是始发路由器支持 Demand Circuits (按需拨号等)；EA 指的是始发路由器支持 External Attributes LSAs (现在未推广)；N/P 只用在 Hello 中 N=1，说明支持 NSSA，P 只用在 NSSA 中，通知 ABR 把 type7 的 LSA 翻译成 type5 的 LSA；MC 只在 MOSPF 中用到；E 表示可以接受外部路由（不是 stub 区），在一个 area 中的所有 router 此位必须一致，(Hello 中体现) 否则邻接关系无法建立；T 表示始发路由器支持 TOS。

LS type，链路类型。每种类型的 LSA 都有唯一的通告格式。

Link State ID，这个字段标识被描述的网络环境的一部分，Link State ID 的内容取决于 LSA 的类型，即不同类型的 LSA 其 Link State ID 也是不同的。比如，当 LSA 的类型是 Type 1 时， Link State ID 是始发路由器的 Router ID；当 LSA 的类型是 Type 2 时， Link State ID 是 DR 在该网段上接口的 IP 地址；当 LSA 的类型是 Type 3 时， Link State ID 是被通告的网络/子网的 IP 地址；当 LSA 的类型是 Type 4 时，Link State ID 是被通告 ASBR 的 Router ID；当 LSA 的类型是 Type 5 时， Link State ID 是目的地的 IP 地址。

Advertising Router，指始发此 LSA 的路由器的 Router ID。比如在 Network-LSAs 中，这个字段就是 DR 在该网段上接口的 IP 地址。

LS sequence number，用于识别 LSA 包是否是一个最新包。路由器每生成一个新的 LSA 时，将该序列号加 1。

LS checksum 用来检查 LSA 的完整性，包括除了 LS age 之外的 LSA 头部的内容。

Length，LSA 的长度，用 bytes 表示。LSA 的头部包括 20 字节。

LSA 头中的链路类型、链路状态 ID 和通告路由器的 Router ID 是一个 LSA 的唯一标识。一个 LSA 将有多个实例，不同的实例通过 LS 的序列号、LS 的校验和及 LS 的 Age 字段来描述。因此，必须要决定其实例是否是最近的，这要通过检查 LS 的序列号、LS 的校验和及 LS 的 Age 字段内容。

1.5.9 LSA 的类型

OSPF 是基于链路状态算法的路由协议，所有对路由信息的描述都是封装在 LSA 中发送出去。当路由器初始化或当网络结构发生变化（例如增减路由器，链路状态发生变化等）时，路由器会产生链路状态广播数据包 LSA（Link-State Advertisement），该数据包里包含路由器上所有相连的链路，也即为所有端口的状态信息。

LSA 根据不同的用途分为不同的种类，主要有如下类型的 LSA：

- Router LSA（Type = 1）：

是最基本的 LSA 类型，所有运行 OSPF 的路由器都会生成这种 LSA。主要描述本路由器运行 OSPF 的接口的连接状况，花费等信息。对于 ABR，它会为每个区域生成一条 Router LSA。这种类型的 LSA 传递的范围是它所属的整个区域。

- Netwrok LSA（Type = 2）：

本类型的 LSA 由 DR 生成。对于广播和 NBMA 类型的网络，为了减少该网段中路由器之间交换报文的次数而提出了 DR 的概念。

一个网段中有了 DR 之后不仅发送报文的方式有所改变，链路状态的描述也发生了变化。

在 DROther 和 BDR 的 Router LSA 中只描述到 DR 的连接，而 DR 则通过 Network LSA 来描述本网段中所有已经同其建立了邻接关系的路由器。（分别列出它们 Router ID）。

同样， 这种类型的 LSA 传递的范围是它所属的整个区域。

- Network Summary LSA（Type = 3）：

本类型的 LSA 由 ABR 生成。当 ABR 完成它所属一个区域中的区域内路由计算之后，查询路由表，将本区域内的每一条 OSPF 路由封装成 Network Summary LSA 发送到区域外。

LSA 中描述了某条路由的目的地址、掩码、花费值等信息。

这种类型的 LSA 传递的范围是 ABR 中除了该 LSA 生成区域之外的其他区域。

- ASBR Summary LSA（Type = 4）：

本类型的 LSA 同样是由 ABR 生成。内容主要是描述到达本区域内部的 ASBR 的路由。

这种 LSA 与 Type3 类型的 LSA 内容基本一样，只是 Type4 的 LSA 描述的目的地址是 ASBR，是主机路由，所以掩码为 0.0.0.0。这种类型的 LSA 传递的范围与 Type3 的 LSA 相同。

- AS External LSA（Type = 5）：

本类型的 LSA 由 ASBR 生成。主要描述了到自治系统外部路由的信息，LSA 中包含某条路由的目的地址、掩码、花费值等信息。

本类型的 LSA 是唯一一种与区域无关的 LSA 类型，它并不与某一个特定的区域相关。这种类型的 LSA 传递的范围整个自治系统（STUB 区域除外）。

- Multicast OSPF LSA（Type =6）：

使用在 OSPF 多播应用程序里。

- Not-So-Stubby Area（Type =7）：

本类型的 LSA 由 Not-So-Stubby area(NSSA) 区域中 ASBR 生成。

为了解决 ASE 路由（自治系统外部路由）单向传递的问题，Not-So-Stubby area(NSSA) 中重新定义了一种 LSA——Type 7 类型的 LSA，作为区域内的路由器引入外部路由时使用。

该类型的 LSA 除了类型标识与 Type 5 不相同之外，其它内容基本一样。这样区域内的路由器就可以通过 LSA 的类型来判断是否该路由来自本区域内。

但由于 Type 7 类的 LSA 是新定义的，对于不支持 NSSA 属性的路由器无法识别，所以协议规定：在 NSSA 的 ABR 上将 NSSA 内部产生的 Type 7 类型的 LSA 转化为 Type 5 类型的 LSA 再发布出去，并同时更改 LSA 的发布者为 ABR 自己。这样 NSSA 区域外的路由器就可以完全不用支持该属性。

在 NSSA 区域内的所有路由器必须支持该属性（包括 NSSA 的 ABR），而自治系统中的其他路由器则不需要。

- External-Attributes-LSA（Type =8）：

特殊的 LSA，还没有实现。当 BGP 信息需要在 OSPF 上承载时，需要用到此 LSA。

- opaque LSA（Type =9～11）：

用于 MPLS 流量工程，有关此 LSA 的详细应用请参考网络学院 MPLS 流量工程培训教材或 RFC2370 文档。当一台路由器向它的邻居发送一条 LSA 后，需要等到对方的确认报文。若在重传间隔时间内没有收到对方的确认报文，就会向邻居重传这条 LSA。