

第 5 章 IPv6 与物联网网络层

物联网中用于将大量传感器和执行器接入 Internet 的接入网通常是低功耗无线个域网 (Low-Power Wireless Personal Area Networks, LoWPAN)。目前存在两种实现 LoWPAN 与 Internet 互联的方式,分别是网关模式和边缘路由器模式。边缘路由器模式使得互联网中结点与 LoWPAN 中结点之间可以直接传输 IPv6 分组,这将极大地方便互联网中后台与传感器和执行器之间的通信过程,但也对 LoWPAN 中结点的处理能力、存储能力和供电能力提出了新的要求。

5.1 网关模式与边缘路由器模式

网关模式下,接入网络中结点之间、结点与网关之间采用专用的链路层技术实现路由和通信过程。由网关实现专用链路层技术与互联网 TCP/IP 协议栈之间的转换过程。边缘路由器模式下,接入网络中结点之间、结点与边缘路由器之间采用网际层技术实现路由和通信过程,接入网络中结点与互联网中结点之间可以直接传输 IPv6 分组。

5.1.1 两种模式的区别

图 5.1 中的结点是智能物体,包含处理器、传感器或执行器、通信模块和电源。网关模式与边缘路由器模式是两种用于将智能物体接入互联网的方式。

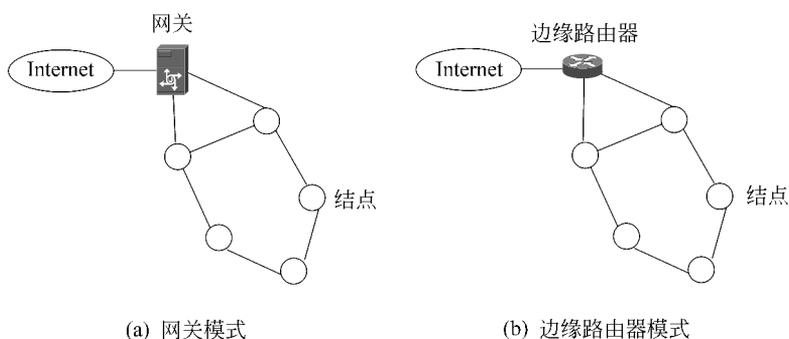


图 5.1 网关模式与边缘路由器模式

1. 网关模式

网关模式下,实现智能物体之间、智能物体与网关之间相互通信过程的接入网络是一个专用网络,智能物体之间、智能物体与网关之间通过专用网络的物理层和数据链路层协议实

现通信过程。网关是互联设备,一端联接互联网,一端联接专用网络。通过联接专用网络的一端接收智能物体发送给互联网的数据,将其封装成 IP 分组(IPv4 分组或 IPv6 分组)后,通过联接互联网的一端转发给互联网。同样,通过联接互联网的一端接收互联网发送给智能物体的数据,将其封装成专用网络的数据链路层对应的帧格式后,通过联接专用网络的一端转发给智能物体。对于 Internet,专用网络和智能物体都是透明的,需要由网关根据互联网发送给智能物体的数据确定数据的目的结点。

2. 边缘路由器模式

边缘路由器模式下,每一个结点(智能物体)分配 IP 地址,结点之间、结点与互联网之间传输的数据封装成 IP 分组(IPv4 分组或 IPv6 分组),边缘路由器的路由表中,既建立用于指明通往互联网的传输路径的路由项,又建立用于指明通往接入网络中各个结点的传输路径的路由项。由边缘路由器实现 IP 分组互联网与接入网络之间的转发过程,结点和实现结点互联的接入网络对于互联网都不是透明的,接入网络成为互联网的一个分支。

5.1.2 模式选择因素

网关模式与边缘路由器模式各有适用场景,选择模式时,需要考虑以下因素。

1. 通信方式

智能物体与互联网之间的通信方式可以分为单向通信方式和双向通信方式。单向通信方式下,只需要实现智能物体至互联网的通信过程。如传感器结点定时向互联网中的服务器发送采集到的数据。双向通信方式下,互联网与智能物体之间需要相互传输数据,如智能物体需要向互联网中的数据中心传输采集到的环境数据,数据中心根据传感器采集到的环境数据决定对环境做出的动作。数据中心通过向执行器发送命令,使得执行器完成数据中心决定的对环境做出的动作。

一般情况下,如果仅需要实现单向通信方式,可以选择网关模式。如果需要进行双向通信方式,可以选择边缘路由器模式,在这种模式下,容易实现互联网中服务器与智能物体之间的端到端通信过程。

2. 传输开销

如果选择边缘路由器模式,智能物体发送的数据需要封装成传输层报文(UDP 报文或 TCP 报文),传输层报文需要封装成 IP 分组(IPv4 分组或 IPv6 分组)。IP 分组需要封装成接入网络的数据链路层对应的帧格式。如果选择网关模式,智能物体发送的数据直接封装成接入网络的数据链路层对应的帧格式。因此,接入网络边缘路由器模式下的传输开销比网关模式下的传输开销大。为了减少传输开销,即使采用边缘路由器模式,也要尽可能压缩传输层报文首部和 IP 首部开销。

3. 数据流模式

如果选择边缘路由器模式,任何智能物体和互联网中的任何主机(终端和服务器)之间都能实现端到端通信。由于端到端通信有着更好的安全和隐私保护机制,因此边缘路由器

模式能够提供更加安全的智能物体与互联网之间的通信过程。

4. 网络多样性

网关模式下的接入网络只能是单种类型的网络。边缘路由器模式下的接入网络可以由多种不同类型的网络互联而成的。实际上,由于网关的功能是与应用相关的,因此,网关模式不仅适用于单种类型网络构成的接入网络,而且只适用于特定应用下的场景。

5. 结点资源

结点资源主要体现在 CPU 处理能力、存储器容量和供电能力。如果结点的 CPU 处理能力和存储器容量不能支撑结点实现 IP 协议栈,这样的结点只能以网关模式接入 Internet。如果结点采用电池供电,应用场景又要求结点间隔较长时间(数月甚至数年)更换电池,这样的结点也不便采用边缘路由器模式接入 Internet。

6. 网络能力

如果接入网络的传输速率很慢,且传输可靠性不高,接入网络对应的数据链路层帧的长度就会受到限制。如果数据链路层帧的最大传输单元(Maximum Transmission Unit, MTU)很小,就会对封装 IP 分组带来困扰。IPv6 分组固定 40B 的首部和 UDP 报文固定 8B 的首部,使得 IPv6+UDP 的传输开销达到 48B,如果接入网络的数据链路层帧对应的 MTU 很小,48B 的传输开销就会是一个大问题。在这种情况下,如果采用边缘路由器方式,必须解决 IPv6+UDP 传输开销过大的问题。

5.1.3 物联网对边缘路由器模式的制约因素

边缘路由器模式下的结点需要实现完整的 IP 协议栈,需要具备 IP 分组转发功能。但物联网中结点处理和存储能力的限制以及接入网络带宽和可靠性的限制会对边缘路由器模式的实现过程带来制约。

1. 受限结点

物联网中,结点 CPU 处理能力、存储器容量和供电能力都是受限的,因此,即使采用边缘路由器模式,结点也不可能像互联网中的主机或路由器一样工作,必须针对受限结点特性,设计 IP 协议栈和路由协议。

2. 受限网络

接入网络大多是低功耗、低传输速率、低可靠性的无线通信网络,这种类型的网络,一是数据链路层帧对应的 MTU 较小;二是无法通过运行互联网路由协议生成路由表;三是结点的角色混杂,可能既有主机功能,又有路由器功能,即既有可能是数据的源或目的结点,又有可能是数据传输过程中的转发结点。

5.1.4 边缘路由器模式下的体系结构和适配层协议

1. 边缘路由器模式下的体系结构

边缘路由器模式下, 结点需要实现 IP 协议栈, 但受限结点和受限网络等制约因素又使得结点不能像互联网中的终端一样实现标准的 IP 协议栈。因此, 采用如图 5.2 所示的边缘路由器模式下的体系结构。该体系结构的特点有两个: 一是网络层协议采用 IPv6; 二是网络层与接入网络之间增加一个适配层——基于低功耗无线个域网的 IPv6 (IPv6 over Low-Power Wireless Personal Area Networks, 6LoWPAN)。



图 5.2 边缘路由器模式下的体系结构

2. 选择 IPv6 的原因

物联网选择 IPv6 的原因如下。

一是解决 IPv4 的地址短缺问题。互联网中目前广泛使用的网络层协议 IPv4, 由于地址位数只有 32 位, 早已暴露出地址短缺问题。虽然网络地址转换 (Network Address Translation, NAT) 和无分类编址缓解了 IPv4 的地址短缺问题, 但没有从根本上解决这一问题。由于物联网需要将成千上万的结点 (智能物体) 接入互联网, 如果需要为每一个结点分配 IPv4 地址的话, IPv4 的地址短缺问题将更加严重。IPv6 的地址位数扩展到 128 位, 拥有庞大的地址空间, 在可以预见的未来, 不会发生 IPv6 的地址短缺问题。

二是灵活的地址分配机制。结点连接到网络时, 只有数据链路层地址, 网络层地址是需要配置的。目前存在两种配置 IPv4 地址的方式, 一是人工配置; 二是通过动态主机配置协议 (Dynamic Host Configuration Protocol, DHCP) 自动获取, 但需要设置和配置 DHCP 服务器。对于大多数物联网应用场景, 需要结点能够做到即插即用, 这是目前 IPv4 地址分配机制无法做到的。IPv6 提供了灵活的地址分配机制, 使得结点可以实现即插即用。

三是 IPv6 是未来趋势。虽然 IPv4 仍然是目前互联网的主流协议, 但由于 IPv4 的固有缺陷, 用 IPv6 代替 IPv4 是大势所趋。

四是存在 IPv6 适配层协议 6LoWPAN。受限结点和受限网络等制约因素使得结点不能像互联网中的终端一样实现标准的 IP 协议栈。只能实现适配于接入网络的 IP 协议栈, 这就需要在接入网络和网络层协议之间创建一个适配层, 通过适配层使得网络层协议满足接入网络具有的受限结点和受限网络的特性。目前 IETF 只定义了 IPv6 对应的适配层协议——6LoWPAN。

3. 6LoWPAN 的功能

为了使得 IPv6 能够适应接入网络具有的受限结点和受限网络的特性, IETF 定义了适配层协议 6LoWPAN。6LoWPAN 的功能是能够将 IPv6 分组封装成接入网络对应的数据链路层帧, 并使得接入网络能够实现 IPv6 分组源结点至目的结点的传输过程。根据接入网络具有的受限结点和受限网络的特性, 6LoWPAN 主要具有以下两个功能: 一是首部压缩, 尽可能减少传输 IPv6 首部和 UDP 首部带来的传输开销; 二是分片, 使得净荷长度较小的接

入网络对应的数据链路层帧能够封装分片后的 IPv6 分组。

边缘路由器连接接入网络的一端支持适配层协议 6LoWPAN,如图 5.3 所示,经过接入网络传输的完成首部压缩和分片操作后的 IPv6 分组,被边缘路由器还原成原始 IPv6 分组后,转发到互联网。同样,互联网发送给接入网络的 IPv6 分组,由边缘路由器完成首部压缩和分片操作后,转发给接入网络。

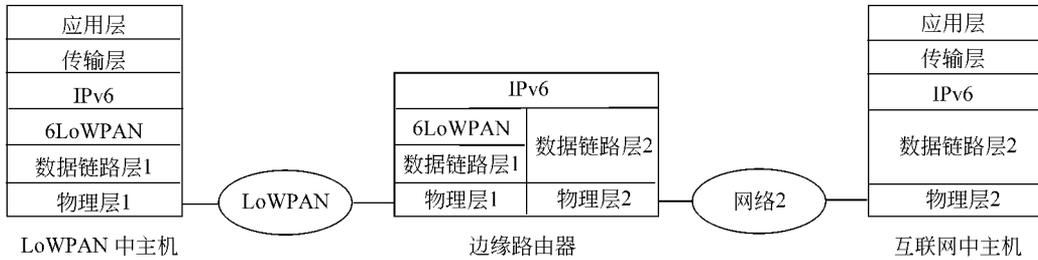


图 5.3 边缘路由器协议结构

5.2 IPv6

IPv6 广阔的地址空间和灵活的地址分配机制使得 IPv6 成为适合物联网的网际层协议。基于 IPv6 的适配层协议 6LoWPAN 使得 IPv6 成为 LoWPAN 的理想网际层协议。

5.2.1 IPv6 首部结构

设计 IPv6 首部结构时需要尽量避免 IPv4 存在的缺陷,因此,IPv6 首部中通过扩展地址字段位数解决地址短缺问题,通过设置流标签字段简化 QoS 实施过程,通过删除增加路由器转发 IP 分组操作步骤的字段提高路由器转发 IP 分组的速率。

1. IPv6 基本首部

IPv6 基本首部如图 5.4 所示,各字段的含义如下。

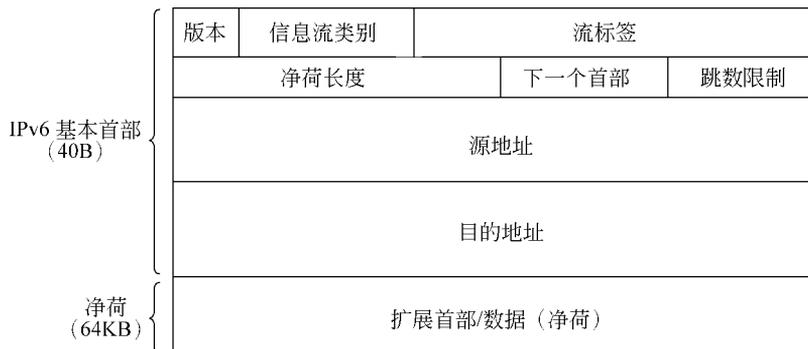


图 5.4 IPv6 基本首部

(1) 版本: 4b。给出 IP 的版本号, IPv6 的版本号为 6, 由于 IPv6 和 IPv4 的版本字段位于 IP 分组的同一位置, 可用该字段值区分 IP 分组所属的 IP 版本。

(2) 信息流类别: 8b。2b 是显式拥塞通知位(Explicit Congestion Notification, ECN), 6b 是区分服务码点(Differentiated Services Code Point, DSCP)。采用区分服务(Differentiated Services, DS)时, 用 DSCP 标识该 IP 分组的服务类别。

(3) 流标签: 20b。流是指一组具有相同的发送和接收进程的 IP 分组。分类服务分为两大类: 一类是区分服务(Differentiated Services, DiffServ), 另一类是综合服务(Integrated Services, IntServ)。区分服务定义若干服务类别, 路由器为不同的服务类别设置不同的服务质量, 当转发某个 IP 分组时, 根据 IP 分组的服务类别字段值确定该 IP 分组所属的类别, 并提供对应的服务质量。综合服务是将属于特定会话的一组 IP 分组作为流, 并为每一种流设置对应的服务质量。流标签用于支持综合服务, 用于唯一标识一组属于相同流的 IPv6 分组。

(4) 净荷长度: 16b。给出 IPv6 分组中净荷的字节数。

(5) 下一个首部: 8b。IPv6 取消了可选项, 增加了扩展首部, 且扩展首部作为净荷的一部分出现在净荷字段中, 使得扩展首部的长度只受净荷字段长度的限制。当存在扩展首部时, 用下一个首部给出扩展首部类型。当没有扩展首部时, 该字段等同于 IPv4 的协议字段, 用于指明净荷所属的协议。

(6) 跳数限制: 8b。给出 IPv6 分组允许经过的路由器数, IPv6 分组每经过一跳路由器, 该字段值减 1, 当该字段值减为 0 时, 如果 IPv6 分组仍未到达目的终端, 路由器将丢弃该 IPv6 分组, 以此避免 IPv6 分组在网络中无休止地漂荡。

(7) 源地址和目的地址: 128b。用于给出 IPv6 分组源终端和目的终端的 IPv6 地址。IPv6 的地址字段长度是 128b, 是 IPv4 的地址字段长度的 4 倍, IPv6 彻底解决了 IPv4 面临的地址短缺问题。

2. IPv6 扩展首部

1) 扩展首部组织方式

IPv4 首部如果包含可选项, 中间经过的每一跳路由器都需要对可选项进行处理, 增加了路由器的处理负担, 降低了路由器转发 IPv4 分组的速率。IPv6 除了逐跳选项扩展首部外, 中间路由器将扩展首部作为分组净荷对待, 不对其做任何处理, 以此简化路由器转发 IPv6 分组所进行的操作, 提高路由器的转发速率。IPv6 目前定义的扩展首部有逐跳选项、路由、分片、鉴别、封装安全净荷、目的端选项这六种。当 IPv6 分组包含多个扩展首部时, 扩展首部按照以上顺序出现, 上层协议数据单元(PDU)放在最后面。如图 5.5 所示是上层协议数据单元(PDU)为 TCP 报文时, IPv6 分组的格式。

如图 5.5(a)所示的 IPv6 分组没有扩展首部, 净荷字段中只包含上层协议数据单元(TCP 报文), 因此, 基本首部中的下一个首部字段值给出上层协议类型 6, 指明上层协议为 TCP。如图 5.5(b)所示的 IPv6 分组中包含单个扩展首部, 净荷字段中首先出现的是路由扩展首部, 而基本首部中的下一个首部字段值给出扩展首部的类型。扩展首部中的下一个首部字段值给出上层协议类型。如图 5.5(c)所示的 IPv6 分组中包含两个扩展首部, 依次在净荷字段中出现的是路由和分片扩展首部。基本首部中的下一个首部字段值给出第 1 个扩展首部的类型(路由), 路由扩展首部中的下一个首部字段值给出第 2 个扩展首部的类型(分

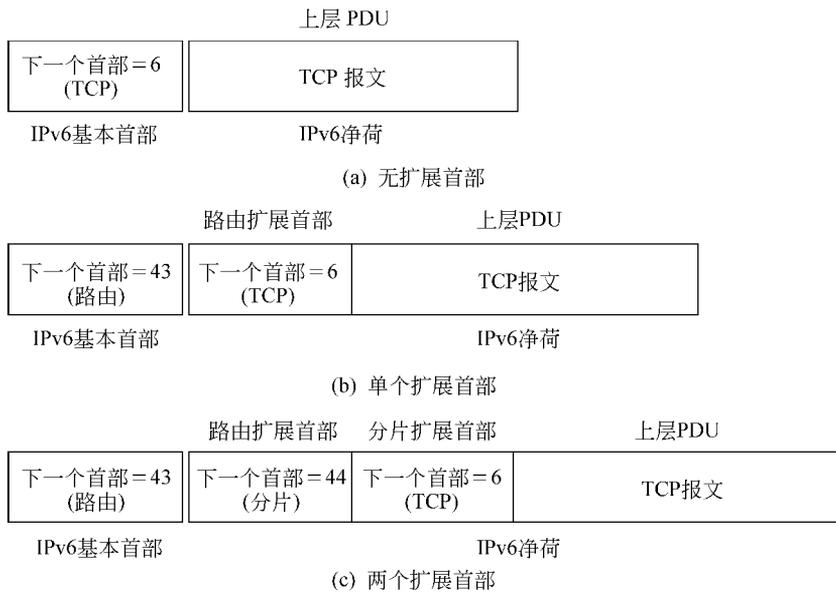


图 5.5 IPv6 基本首部、扩展首部和上层协议数据单元之间的关系

片),分片扩展首部中的下一个首部字段值给出上层协议类型(TCP)。当净荷字段中包含两个以上的扩展首部时,由前一个扩展首部中的下一个首部字段值给出下一个扩展首部的类型,最后一个扩展首部的下一个首部字段值给出上层协议类型。

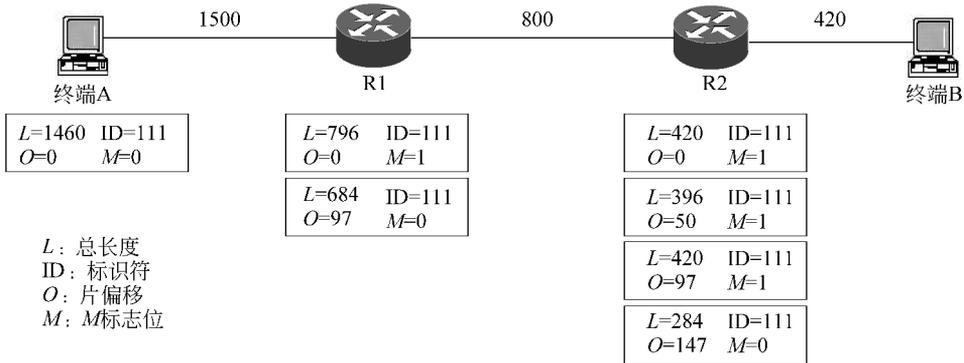
2) 扩展首部应用实例

下面通过分片扩展首部的应用,说明 IPv6 简化路由器转发操作的过程。分片扩展首部格式如图 5.6 所示。它的各个字段的含义和 IPv4 首部中与分片有关的字段的含义相同,片偏移给出当前数据片在原始数据中的位置。标识符用来唯一标识分片数据后产生的数据片序列,接收端通过标识符鉴别出因为分片数据后产生的一组数据片。 M 标志位用来标识最后一个数据片($M=0$)。如图 5.7 所示是一个互联网结构图,链路上标出的数字是链路 MTU。对于 IPv4 分组,由路由器根据输出链路 MTU 和 IPv4 分组的总长确定是否对 IP 分组分片,并在需要分片的情况下,完成分片操作。对于 IPv6 分组,由源终端通过路径 MTU 发现协议找出源终端至目的终端传输路径所经过链路的最小 MTU,该 MTU 称为路径 MTU,并由源终端完成分片操作,通过分片扩展首部给出各个数据片的片偏移及标识符。目的终端通过分片扩展首部中给出的信息,重新将各个数据片拼接成原始 IPv6 分组,整个操作过程如图 5.7 所示。

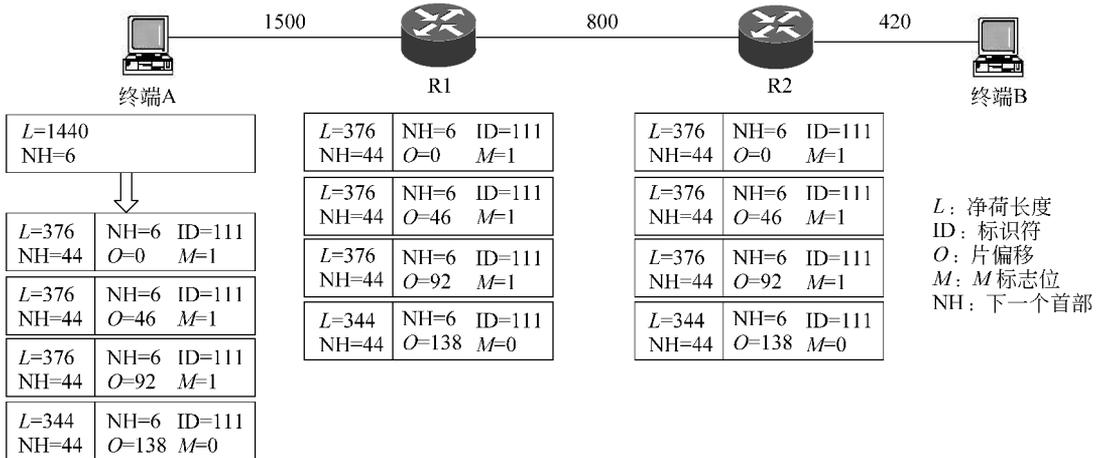
下一个首部	保留	片偏移	保留	M
标识符				

图 5.6 分片扩展首部格式

值得强调的是,IPv4 由路由器负责分片操作,而且可能由多个路由器对同一 IPv4 分组反复进行分片操作,如图 5.7(a)所示,这将严重影响路由器的转发速率。在 IPv6 中,改由源终端完成分片操作。源终端首先通过路径 MTU 发现协议获取源终端至目的终端传输路径所经过链路的最小 MTU(路径 MTU),然后对净荷进行分片。通常情况下,除最后一个数



(a) IPv4分片过程



(b) IPv6分片过程

图 5.7 IPv4 和 IPv6 分片过程

据片,其他数据片长度的分配原则是:须是 8 的倍数,且加上 IPv6 首部和分片扩展首部后尽量接近路径 MTU。假定路径 $MTU=M$,净荷长度= L ,将净荷分成 N 个数据片,则 N 是满足条件“ $L + N \times 48 \leq M \times N$ ”的最小整数。48B 包括 40B IPv6 基本首部和 8B 分片扩展首部。在本例中, $M=420B$, $L=1440B$,根据 $1440 + N \times 48 \leq 420 \times N$,得出 $N \geq 1440 / (420 - 48) = 3.87$, N 取满足上述等式的最小整数 4。前 3 个数据片长度应该是满足条件“小于或等于 $(420 - 48)$ 且是 8 的倍数”的最大值,这里是 368B。加上 8B 的分片扩展首部后,得出净荷长度=376B,最后 1 个数据片的长度是 $1440 - 3 \times 368 = 336B$,得出净荷长度=344B。4 个数据片的片偏移分别是 0、 $368/8=46$ 、 $736/8=92$ 和 $1104/8=138$ 。值得说明的是,在每个会话存在期间,源终端和目的终端之间都有大量 IP 分组传输,因此,源终端先通过路径 MTU 发现协议获取源终端至目的终端传输路径所经过链路的最小 MTU(路径 MTU)是值得的,否则,对每一个 IP 分组都进行如图 5.7(a)所示的分片操作会对路由器的转发速率造成巨大影响。

5.2.2 IPv6 地址结构

开发 IPv6 的主要原因是为了解决 IPv4 的地址短缺问题,因此,IPv6 的地址字段长度是

IPv4 的 4 倍,即 128b。有人计算过, 2^{128} 的 IPv6 地址空间可以为地球表面每平方米的面积提供 8.65×10^{23} 个不同的 IPv6 地址,这么多的 IPv6 地址可以为地球上的每一粒沙子分配唯一的 IPv6 地址。如此巨大的地址空间,为使用 IPv6 地址提供了非常大的灵活性。

1. IPv6 地址表示方式

1) 基本表示方式

基本表示方式是将 128b 以 16 位为单位分段,每一段用 4 位十六进制数表示,各段用冒号分隔。下面是两个用基本表示方式表示的 IPv6 地址。

```
2001:0000:0000:0410:0000:0000:0001:45FF
0000:0000:0000:0000:0001:0765:0000:7627
```

2) 压缩表示方式

基本表示方式中可能出现很多 0,甚至可能整段都是 0,为了简化地址表示,可以将不必要的 0 去掉。不必要的 0 是指去掉后,不会错误理解段中 16 位二进制数的那些 0。如 0410 可以压缩成 410,但不能压缩成 41 或 041。上述用基本表示方式表示的 IPv6 地址可以压缩成如下表示方式。

```
2001:0:0:410:0:0:1:45FF
0:0:0:0:1:765:0:7627
```

用压缩表示方式表示的 IPv6 地址仍然可能出现相邻若干段都是 0 的情况,为了进一步缩短压缩表示方式表示的 IPv6 地址,可用一对冒号::表示连续的一串 0,当然,一个 IPv6 地址只能出现一个::。这种用::表示连续的一串 0 的压缩表示方式就是 0 压缩表示方式,如下是用 0 压缩表示方式表示上述地址的结果。

```
2001::410:0:0:1:45FF
::1:765:0:7627
```

2001:0:0:410:0:0:1:45FF 也可表示成 2001:0:0:410::1:45FF,但不能表示成 2001::410::1:45FF,因为后一种表示无法确定每一个::表示几个相邻的 0。

3) 特殊地址

(1) 内嵌 IPv4 地址的 IPv6 地址。

这种地址是为了解决 IPv4 和 IPv6 共存时期配置不同版本的 IP 地址的终端之间通信问题而设置的,128b 的地址中包含 32b 的 IPv4 地址,32b 的 IPv4 地址仍然采用 IPv4 的地址表示方式,以 8b 为单位分段,每一段用对应的十进制值表示,段之间用点分隔。地址的其他部分采用 IPv6 的地址表示方式。以下是常用的两种内嵌 IPv4 地址的 IPv6 地址的表示方式。

```
0000:0000:0000:0000:0000:FFFF:192.167.12.16 或是::FFFF:192.167.12.16
0000:0000:0000:0000:FFFF:0000:192.167.12.16 或是::FFFF:0:192.167.12.16
```

(2) 环回地址。

::1 是 IPv6 的环回地址,等同于 IPv4 的 127.X.X.X。

(3) 未确定地址。

全 0 地址(表示成::)作为未确定地址,当某个没有分配有效 IPv6 地址的终端需要发送 IPv6 分组时,可用该地址作为 IPv6 分组的源地址。该地址不能作为 IPv6 分组的目的地址。

4) 地址前缀

IPv6 采用无分类编址方式,将地址分成前缀部分和主机号部分,用前缀长度给出地址中表示前缀的二进制数位数,用下述表示方式表示地址前缀。

IPv6 地址/前缀长度

IPv6 地址必须是用基本表示方式或 0 压缩表示方式表示的完整地址,前缀长度是一个 0~128 的整数,给出 IPv6 地址的高位中作为前缀的位数。以下是正确的前缀表示方式。

::FE80:0:0:0/68

::1:765:0:7627/60

2001:0000:0000:0410:0000:0000:0001:45FF/64

2. IPv6 地址分类

IPv6 地址分为单播、组播和任播这三种类型。

(1) 单播地址:唯一标识某个接口,以该种类型地址为目的地址的 IPv6 分组,到达目的地址标识的唯一的接口。

(2) 组播地址:标识一组接口,而且,大部分情况下,这组接口分属于不同的结点(终端或路由器),以该种类型地址为目的地址的 IPv6 分组,到达所有由目的地址标识的接口。

(3) 任播地址:标识一组接口,而且,大部分情况下,这组接口分属于不同的结点(终端或路由器),以该种类型地址为目的地址的 IPv6 分组,到达由目的地址标识的一组接口中的其中一个接口,该接口往往是这一组接口中和源终端距离最近的那个接口。

1) 单播地址

(1) 链路本地地址。

这里的链路不是指物理线路,它指的是实现连接在同一网络上的两个结点之间通信过程的传输网络,如以太网。链路本地地址指的是在同一传输网络内作用的 IPv6 地址,它的作用有两个:一是用于实现同一传输网络内两个结点之间的网际层通信过程;二是用于标识连接在同一传输网络上的接口,并用该 IPv6 地址解析接口的链路层地址。一旦某个接口被定义为 IPv6 接口,该接口自动生成链路本地地址。链路本地地址格式如图 5.8 所示。

10b	54b	64b
1111111010	0	接口标识符

图 5.8 链路本地地址格式

链路本地地址的高 64b 是固定不变的,低 64b 是接口标识符。接口标识符用于在传输网络内唯一标识某个连接在该传输网络上的接口。存在两种常用的导出接口标识符的方法,一种由接口的链路层地址导出;另一种是随机生成的 64 位随机数。

不同类型的传输网络导出接口标识符的过程不同,下面是通过以太网的 MAC 地址导出接口标识符的过程。

48 位 MAC 地址由 24 位的公司标识符和 24 位的扩展标识符组成,公司标识符由 IEEE 负责分配。公司标识符最高字节的第 0 位是 I/G(单播地址/组地址)位,该位为 0,表明是单播地址;该位为 1,表明是组地址。第 1 位是 G/L(全局地址/本地地址)位,该位为 0,表明是全局地址;该位为 1,表明是本地地址。一般情况下,MAC 地址都是全局地址,G/L 位为 0。MAC 地址导出接口标识符的过程如图 5.9 所示,首先将 MAC 地址的 G/L 位置 1,然后在公