

# 在 IEEE 802.15.4 网路上传送 IPv6 报文

## 文档声明

本文档为因特网社区指定一份因特网标准追踪协议，并请求讨论和改进建议。请参考当前版本的“因特网官方协议标准”（STD 1）对本协议的标准化陈述和状态陈述。本文档的分发不受限制。

## 摘要

本文描述了在 IEEE 802.15.4 网路上传送 IPv6 报文的帧格式，和生成 IPv6 链路地址和无状态自动配置地址的方法。另外还包括了一个简单的使用共享上下文的报头压缩方法和在 IEEE 802.15.4 meshs 网路中传送报方的方法。

## 1、简介

IEEE 802.15.4 标准[ieee802.15.4]的目标是低功耗个人局域网。本文定义了 IEEE 802.15.4 网路上传送 IPv6[RFC2460]报文的帧格式和 IPv6 链路地址和无状态自动配置地址等信息。由于 IPv6 的报文长度比 IEEE 802.15.4 最大帧长还要大得多，所以这里定义了一个适配层。为了使 IPv6 能在 IEEE 802.15.4 网路上运行和在 IEEE 802.15.4 meshs 上进行报文传送，本文还定义了报文压缩机制。然而，一个完整的 mesh 路由定义（使用特定的协议，集成邻居发现等）不在本文的讨论范围内。

### 1.1、符号说明

本文使用的关键词“MUST”、“MUST NOT”、“REQUIRED”、“SHALL”、“SHALL NOT”、“SHOULD”、“SHOULD NOT”、“MAY”和“OPTIONAL”请参考[RFC2119]的描述。

### 1.2、术语使用

AES：高级加密标准

CSMA/CA：载波感测多元存取 / 碰撞侦测

FFD：全功能设备

GTS：保证时间服务

MTU：最大传送单元

MAC：介质访问控制

PAN：个人局域网

RFD：简化功能设备

## 2、IP 的 IEEE 802.15.4 形式

IEEE 802.15.4 定义了 4 种帧：信标帧、MAC 指令帧、确认帧和数据帧。IPv6 报文必须放在数据帧里。数据帧要求进行确认是可选的。为了符合[RFC3819]，为了帮助链路恢复，建议

IPv6 报文放在需要确认的帧上进行传送。

IEEE 802.15.4 网路可以是非信标使能或信标使能的[ieee802.15.4]。后者是设备被叫作协调器信标同步的一个可选方式。这使得在一个非竞争的时间保证服务（GTS）内使用超级帧成为可能。本文并没要求 IEEE 网路运行在信标使能模式。在非信标使能网路中，数据帧（包括运载 IPv6 报文的）是通过基于竞争的无槽 CSMA/CA 通道访问方法来传送的。

在非信标使能网路中，同步过程不使用信标。然而，它们仍然对事件的关联和解除关联上的链路层设备发现是有用的。本文建议配置信标以实现这些功能。另一个建议是为了使这些事件在 IPv6 层的可用以便于检测网路附件，一个 IETF 正在研究的问题。

标准里允许帧省略源地址或目标地址（或两者）。本文定义的机制要求 IEEE 802.15.4 帧头里包含源地址和目标地址。同样包含源和目标的 PAN ID 域。

### 3、地址模式

IEEE 802.15.4 定义了几种地址模式：允许使用 IEEE64 位扩展地址或（在一个关联事件后）使用在 PAN 里唯一的 16 位地址[ieee802.15.4]。本文支持 64 位扩展地址和 16 位短地址。在 6LoWPANs 里使用 16 位短地址，本文在格式上增加了一些限制（除了 IEEE 802.15.4 要求的之外），具体描述见第 12 节。短地址实质上是瞬态的，需要注意的是：因为它们是由 PAN 协调器功能在一个关联事件中分发的，它们的有效性和唯一性只限制在这一个关联周期内。这个周期可能被关联的有效期或 PAN 协调器发生事故而缩短。由于集中分配和 PAN 协调器的单点故障而产生的可伸缩问题，开发者在使用短地址进行这样的网络部署时必须仔细权衡网络的增长（并实行必要的机制）。当然，IEEE64 位扩展地址就不会有这样的缺点，但仍然会有剩下的这些可扩展性问题，涉及到路由、发现、配置等。

本文假定一个 PAN 映射到一个特定的 IPv6 连接。这符合共享网络支持链路层子网[RFC3819]广播的建议。严格来说，在 IPv6 里是多播而不是广播。然而，IEEE 802.15.4 原本是不支持多播的。所以，IPv6 层多播报文必须放在 IEEE 802.15.4 网路的链路广播帧里。这是必须这样做的，广播帧只被一个特定链路 PAN 里的设备接收。对于[ieee802.15.4]的第 7.5.6.2 节，由以下方式来完成：

- 1) 帧里包含一个目的 PAN 标识符，它必须与链路层的 PAN ID 相符合。
- 2) 帧里包含一个目的短地址，它必须与广播地址（0xffff）相符合。

另外，第 9 节的 IPv6 多播地址映射的支持必须只用在 mesh 配置里。本文对这个功能不做更具体的描述。

通常，主机从路由器广告里学习 IPv6 前缀[RFC4861]。

### 4、最大传送单元

在 IEEE 802.15.4 上的 IPv6 报文最大传送单元是 1280 字节。然而，一个 IEEE 802.15.4 帧里容不下一个完整的 IPv6 报文。802.15.4 协议数据单元的大小取决于报头的大小[ieee802.15.4]。由于一个物理层的最大帧长是 127 字节（一个最大 PHY 报文长度）且一个最大帧头长度是

25 字节（一个最大帧头），所以介质访问控制层的最大帧长是 102 字节。链路安全协议还需要增加报头长度，最大的情况（对于 AES-CCM-128 来说是 21 字节，AES-CCM-32 和 AES-CCM-64 分别是 9 和 13 字节），只剩下 81 字节可用。这明显远不够 IPv6 报文的最小长度 1280 字节，为了和第 5 节的 IPv6 规格[RFC2460]一致，IP 层下必须提供一个分片和重组的适配层。适配层在第 5 节定义。

另外，由于 IPv6 报头长度是 40 字节，这就只剩下 41 字节给上层协议，如 UDP。UDP 使用 8 字节的报头，那应用数据就只剩下 33 字节了。并且，如上面提到的，还需要一个分片和重组的适配层，这将消耗更多的报头字节。

上述的这些考虑导致了下面两个观察：

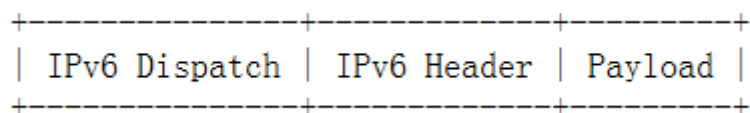
- 1) 适配层必须满足 IPv6 对于最小 MTU 的要求。然而，(a) 多数 IEEE 802.15.4 应用并不会使用这么大的报文，(b) 小量的应用数据和使用合理的报头压缩可以产生适合于一个 IEEE 802.15.4 帧的报文。使用适配层的理由不仅是为了满足 IPv6 报文，极有可能一些应用交换（如配置或服务）产生的报文也需要少量的分片。
- 2) 即使上述计算显示的是最坏情况的场景，它的确显示出了事实上报文压缩几乎是不可避免的。我们希望多数（如果不是全部）IEEE 802.15.4 使用 IP 的应用会用到报头压缩，这在它 10 节定义。

## 5、LoWPAN 适配层和帧格式

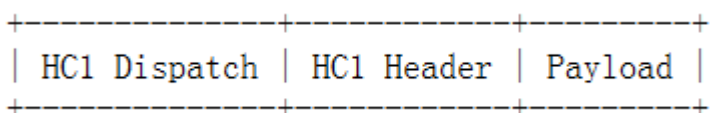
本节定义的封装格式（后面参考“LoWPAN 封装”）是 IEEE 802.15.4 MAC 协议数据单元（PDU）里的载荷。LoWPAN 载荷（例如，一个 IPv6 报文）在这个封装报头后面。

所有通过 IEEE 802.15.4 传送的 LoWPAN 封装报文都是由一个封装报头堆栈加载前缀的。在报头堆栈的每个报头包含一个报头类型，后面是 0 个或多个报头域。然而在一个 IPv6 报头里，堆栈依次包含地址、逐跳选项、路由、分片、目的选项和载荷[RFC2460]；在一个 LoWPAN 报头里，类似的报头序列是 mesh(L2)地址、逐跳选项（包括 L2 广播/多播）、分片和载荷。以下的例子展示了在 LoWPAN 网路中可能用到的典型的报头堆栈。

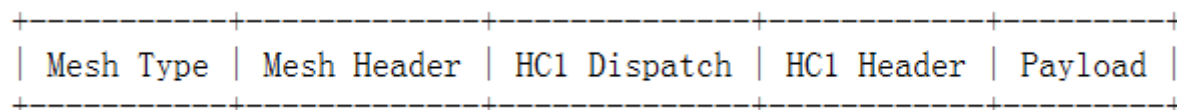
一个 LoWPAN 封装的 IPv6 报文：



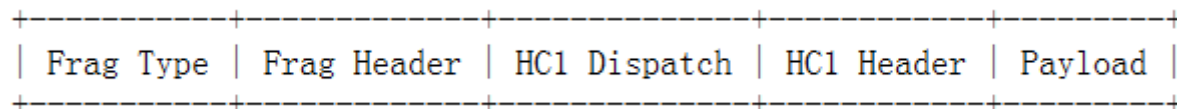
一个 LoWPAN 封装的 LOWPAN\_HC1 压缩 IPv6 报文



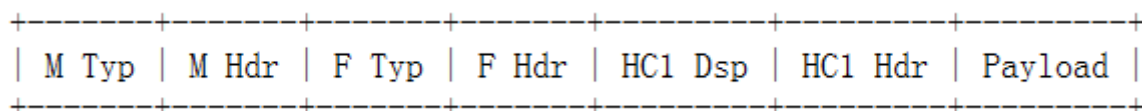
一个 LoWPAN 封装的 LOWPAN\_HC1 压缩并要求 mesh 寻址的 IPv6 报文



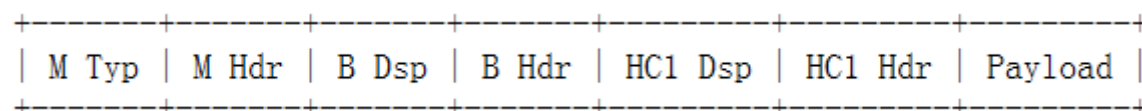
一个 LoWPAN 封装的 LOWPAN\_HC1 压缩并要求分片的 IPv6 报文



一个 LoWPAN 封装的 LOWPAN\_HC1 压缩并要求 mesh 寻址和分片的 IPv6 报文



一个 LoWPAN 封装的 LOWPAN\_HC1 压缩并要求 mesh 寻址和有广播报头支持 mesh 广播/多播的 IPv6 报文



当在一个报文里使用多个 LoWPAN 报头时，它们必须按以下顺序出现：

Mesh 寻址报头

广播报头

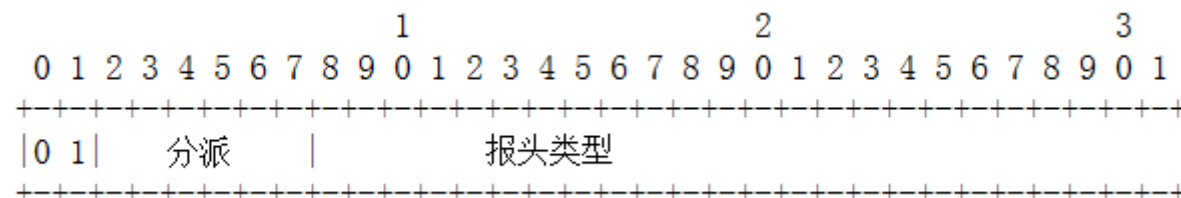
分片报头

所有协议报文（例如，IPv6、压缩的 IPv6 报头，等等）应该在一个有效的 LoWPAN 封装报头后面，如上面所给出的例子。这可以统一软件对报文的处理而不管传送方式是什么。

LoWPAN 报头的定义，除了 mesh 寻址和分片，还包含分派值、报头域的定义和与其他报头相关的顺序限制。尽管报头堆栈提供了一个处理在 LoWPAN 适配层上的未来需求的机制，但它不打算提供通用目标的可扩展性。这份格式文档为清晰性、简洁性和正交性指定了一个使用报头堆栈的报头类型集合。

### 5.1、分派类型和报头

分派类型定义为第 1 位为 0 第 2 位为 1。分派类型和报头如下图所示：



分派            一个 6 位选择器。确定报头类型，后面是分派报头。

报头类型      一个由分派报头确定的报头

图 1：分派类型和报头

分派值可以当成是一个无结构的命名空间。只需要少量的符号就可以表示当前的 LoWPAN

功能。虽然一些额外的存储可以通过在分派位附加功能来实现，但这种方法可能对未来扩展功能有所限制。

位序列	报头类型
00 xxxxxx	NALP - 不是一个LoWPAN帧
01 000001	IPv6 - 无压缩的IPv6地址
01 000010	LOWPAN_HC1 - LOWPAN_HC1 压缩的 IPv6
01 000011	reserved - 保留以后用
...	reserved - 保留以后用
01 001111	reserved - 保留以后用
01 010000	LOWPAN_BC0 - LOWPAN_BC0 广播
01 010001	reserved - 保留以后用
...	reserved - 保留以后用
01 111110	reserved - 保留以后用
01 111111	ESC - 后面有附加的分派位
10 xxxxxx	MESH - Mesh 报头
11 000xxx	FRAG1 - 分片报头（第一个）
11 001000	reserved - 保留以后用
...	reserved - 保留以后用
11 011111	reserved - 保留以后用
11 100xxx	FRAGN - 分片报头（随后的）
11 101000	reserved - 保留以后用
...	reserved - 保留以后用
11 111111	reserved - 保留以后用

图 2：分派值位序列

**NALP**：指明接下来的位序列不是 LoWPAN 封装的一部分，任何 LoWPAN 节点收到一个分派值为 00xxxxxx 的报文应该要忽略。其他非 LoWPAN 协议想要和 LoWPAN 节点共存的应该要在 802.15.4 报头后包含一个与这个序列相同的字节。

**IPv6**：指明接下来的报头是一个无压缩的 IPv6 报头[RFC2460]。

**LOWPAN\_HC1**：指明拉下来的报头是一个 LOWPAN\_HC1 压缩的 IPv6 报头。这个报头格式定义如图 9 所示。

**LOWPAN\_BC0**：指明接下来的报头是一个 LOWPAN\_BC0 报头，它支持 mesh 广播/多播，在 11.1 节进行描述。

**ESC**：指明接下来的报头是一个用于分派值的单独的 8 位域。它允许支持分派值大于 127。

## 5.2、Mesh 寻址类型和报头

Mesh 类型定义为前第一位是 1 第二位是 0。Mesh 类型和报头如下图所示：

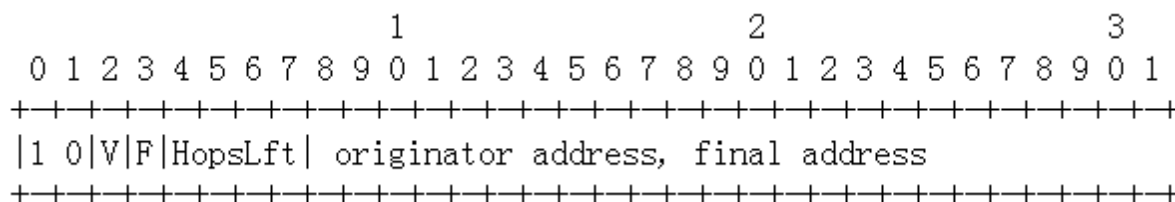


图 3: Mesh 寻址类型和报头

域定义如下所示：

**V:** 如果源（第一次）地址是一个 IEEE 扩展 64 位地址（EUI-64），这位应该是 0；如果是 16 位短地址，这位就为 1。

**F:** 如果最终目标地址是一个 IEEE 扩展 64 位地址（EUI-64），这位应该是 0；如果是 16 位短地址，这位就为 1。

**Hops Left:** 这个 4 位域的值应该由每一个转发节点在发送报文到下一跳前减 1。如果 Hops Left 值减到 0 报文将不再转发。值 0xF 是保留的并表示接下来的字节里有一个 8 位的 Deep Hops Left 域，这允许源节点指定一个大于 14 的跳数限制。

**Originator Address:** 这是源端的链路层地址。

**Final Destination Address:** 这是最终目标的链路层地址。

注意到 ‘V’ 和 ‘F’ 位允许 16 位和 64 位地址的混合。这至少对于允许 mesh 层“广播”是有用的，因为 802.15.4 广播地址定义的是 16 位短地址。

关于在 mesh 网路中传送帧的内容将会在第 11 节进行具体的讨论。

### 5.3、分片类型和报头

如果一个完整的载荷（如 IPv6）数据报装在一个单独的 802.15.4 帧里，它就是不分片的，那么 LoWPAN 封装就不必包含一个分片报头。如果数据报不是封装在一个单独的 IEEE 802.15.4 帧内，它应该在链路层进行分片。因为碎片偏移量只能表示为 8 字节的倍数，所以除了最后一片外的所有分片大小都必须为 8 字节的整数倍。第一个链路层分片应该包含第一个分片报头，其定义如下图所示：

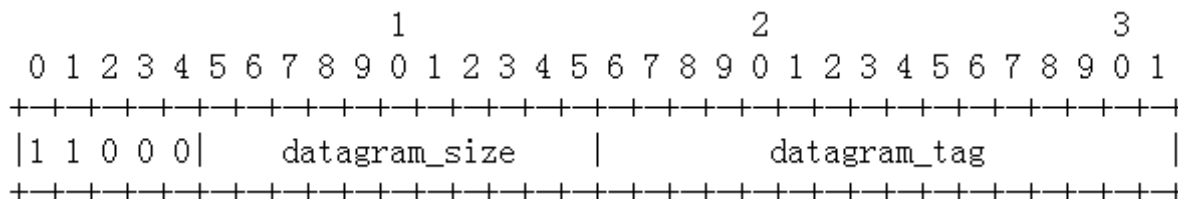


图 4: 第一个分片

第二个和以后的链路层分片（直到包含最后一个）应该包含的分片报头如下图所示：



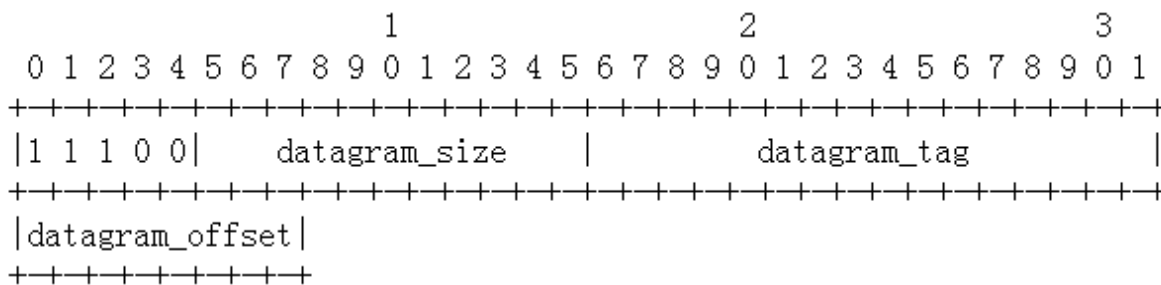


图 5：后面的分片

**datagram\_size:** 这个 11 位域编码表示在链路层分片之前（但在 IP 层分片之后）的整个 IP 报文的大小。对于一个 IP 报文来说所有的链路层分片的 **datagram\_size** 的值都应该是相同的。对于 IPv6，这个值应该是比报文里 IPv6 报头[RFC2460]的载荷长度多 40 字节（无压缩的 IPv6 报头的长度）。注意，这个报文可能已经由参与通信的主机进行分片了，也就是说，这个域需要设置最大长度是 1280 字节（IEEE 802.15.4 链路 MTU，如在本文所定义的那样）。

注意：不是每个报文都需要这个域的，如第一个发送的分片里含有这个域，后面的就可以不用了。然而，在每一个链路分片里包含这个域有利于目标主机对分片进行重组特别是在第二个（或更后面的）分片比第一个分片早到达的情况下。在这种情况下，接收者在收到任一个分片后就可以知道 **datagram\_size** 的值并在等待接收剩下的分片前知道设置多大的缓存空间。上述格式用简单换取了效率。

**datagram\_tag:** **datagram\_tag**（数据报标签）的值对于一个载荷（如 IPv6）数据报的所有链路分片应该是相同的。发送者应该对连续的分片数据报递增 **datagram\_tag** 的值。当 **datagram\_tag** 的值递增到大于 65535 时应该变为 0。这个域是 16 位的，它的初始值没有定义。

**datagram\_offset:** 这个域只出现在第二个和之后的链路分片里，并应该指示从载荷数据报开始的以 8 字节为增量的分片偏移量。数据报的第一个字节（例如，IPv6 报头的开始）的偏移量是 0；第一个链路分片的 **datagram\_offset** 隐含值是 0。这个域是 8 位的。

链路分片的接收者应该使用（1）发送者的 802.15.4 源地址（或 Originator Address 如果出现了 Mesh Addressing 域），（2）目标的 802.15.4 地址（或 Final Destination Address 如果出现了 Mesh Addressing 域），（3）**datagram\_size** 和（4）**datagram\_tag** 来确定一个给定数据报的所有链路分片。

在收到一个链路分片后，接收者开始构造大小为 **datagram\_size** 的未分片的原始报文。它使用 **datagram\_offset** 域来确定每一个分片在原始报文中的位置。例如，它把数据载荷（除了封装报头）放在一个位置由 **datagram\_offset** 确定的载荷数据报重组缓存里。重组缓存的大小应该由 **datagram\_size** 来确定。

如果收到与另一个分片重叠的链路分片，如上面所确定的，并且与重叠的分片的大小和 **datagram\_offset** 都不同，那么已经累积在重组缓存里的分片应该要丢弃。一个新的重组可能

从最近接收到的链路分片开始。分片重叠是由封装报头的 `datagram_offset` 和 802.15.4 物理层协议数据单元（PPDU）报文头部的“帧长”组合来确定的。

在检测到一个 IEEE 802.15.4 分裂事件时，分片接收者必须丢弃所有部分重组的载荷数据报所有的链路分片，分片发送者必须丢弃部分发送的载荷（如 IPv6）数据报的未发送的链路分片。类似地，当一个节点接收到第一个给定 `datagram_tag` 的分片，它开始一个重组定时器。当定时时间到，如果整个报文没有重组完成，现有的分片必须丢弃，并且重组状态必须重设。重组超时的最大时间必须为 60 秒（这也是 IPv6 重组程序的最大超时[RFC2460]）。

## 6、无状态地址自动配置

本节字义如何获得一个 IPv6 接口标识符。

IEEE 802.15.4 接口的接口标识符[RFC4291]可能是基于分配给 IEEE 802.15.4 设备的 EUI-64 标识符[EUI64]。在这种情况下，接口标识符依据“以太网上的 IPv6”说明文档[RFC2464] 来从 EUI-64 成生的。

所有的 802.15.4 设备都有一个 IEEE EUI-64 地址，但 16 位短地址（节 3 和节 12）也同样是可能的。在这些情况下，一个“假的 48 位地址”的产生过程如下。首先，左 32 位由 16 位 0 和 16 位 PAN ID（作为选择，如果不知道 PAN ID，就用 16 位 0）组成。这样产生的 32 位域如下所示：

16 位 PAN ID：16 位 0

然后，这个 32 位后面连接的是 16 位短地址。这样产生的 48 位地址如下所示：

如上所示的 32 位：16 位短地址

由这个 48 位地址产生的接口标识符如同“以太网上的 IPv6”说明[RFC2464]一样。然而，在合成的接口标识符里，为了符合这个不是全局唯一值这个事实，“全局/本地”（U/L）位应该设置为 0。对于这两种地址格式，不可以使用全 0 的地址。

另一个由人工或软件设置的 MAC 地址可以用来成生接口标识符。如果使用了这个 MAC 地址，它的全局唯一性应该体现在 U/L 位的值上。

在 IEEE 802.15.4 接口上用于无状态自动配置[RFC4862]的 IPv6 地址前缀长度必须是 64 位的。

## 7、IPv6 链路本地地址

IEEE 802.15.4 接口的 IPv6 链路本地地址[RFC4291]由扩展接口标识符产生，如下定义，前缀是 FE80::/64。

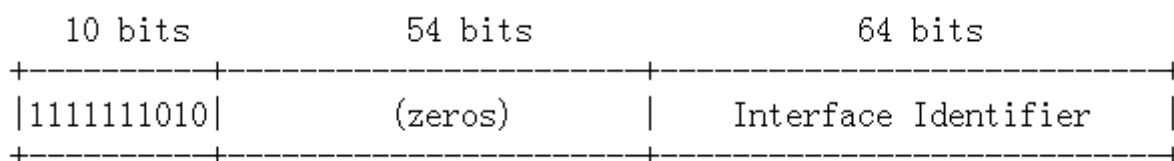




图 6

## 8、单播地址映射

用于映射非多播 IPv6 与 IEEE 802.15.4 链路层地址的地址解析程序服从[RFC4861]第 7.2 节的通用描述，除非有另外的规定。

当链路层是 IEEE 802.15.4 并地址是 EUI-64 或 16 位短地址时，源/目标链路层地址选项分别有以下的形式。

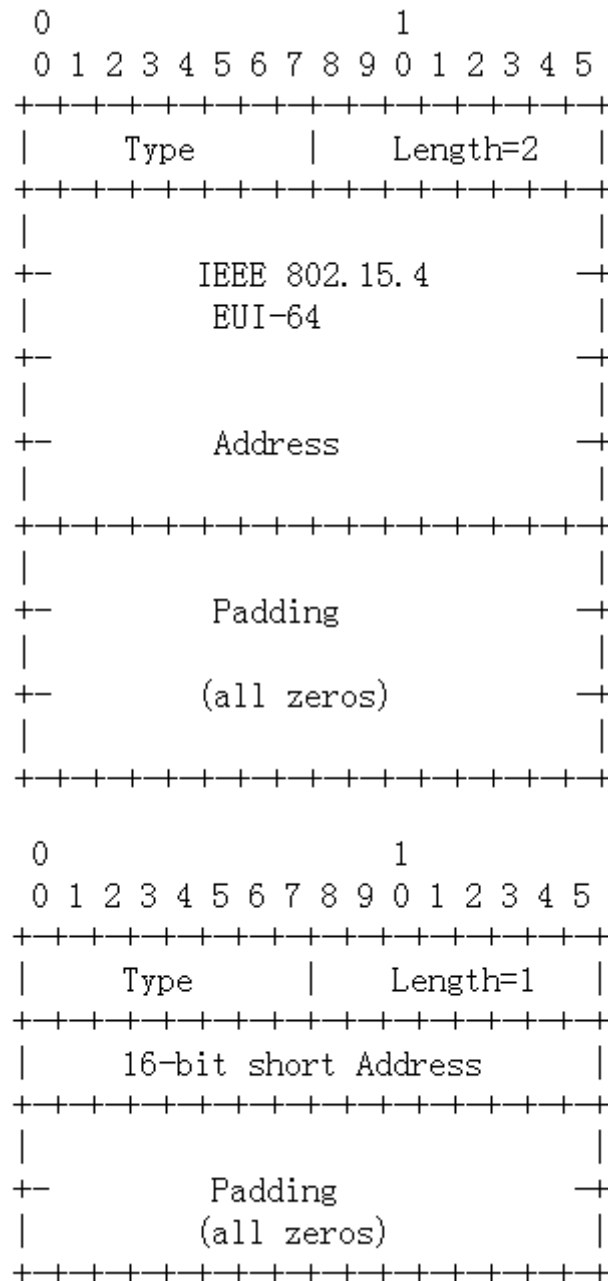


图 7

选项域:

Type:

- 1: 源链路层地址。
- 2: 目标链路层地址。

**Length:** 这是这个选项（包含 type 和 length 域）的长度，单位是字节。如果使用 EUI-64 地址，这个域的值为 2，如果使用 16 位短地址，这个域的值为 1。

**IEEE 802.15.4 Address:** 64 位的 IEEE 802.15.4 地址，或 16 位短地址（如节 9 所描述的格式），使用正则位顺序。这是接口目前可响应的地址。这个地址可能与用于产生接口标识符的内建的地址不同，缘于隐私或安全（如，邻居发现）的考虑。

## 9、多播地址映射

这一节里的功能必须在一个 mesh-使能的 LoWPAN 里使用。一个有多播目标地址（DST）的 IPv6 报文，包含 16 字节的 DST[1]到 DST[16]，将会传送到以下 802.15.4 的 16 位多播地址：

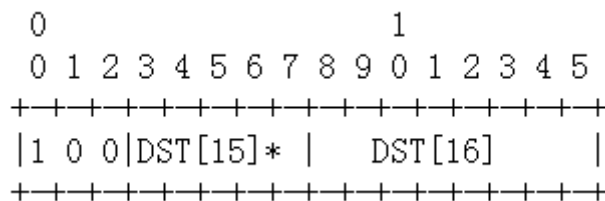


图 8

这里，DST[15]\*表示 DST[15]的后 5 位，那就是，DST[15]里的 3-7 位。开始的 3 位序列“100”紧跟在 16 位短地址格式后面用于多播地址（节 12）。

这允许在 6LoWPAN 网络中支持多播，但具体的多播方式不在本文的描述范围内。机制示例是：泛洪、受控泛洪、单播到 PAN 协调器等。这应该由不同的 mesh 路由机制来描述。

## 10、报头压缩

现有很多已发行的和正在制定的报头压缩标准。尽管如此，在 IEEE 802.15.4 上的 IPv6 报头压缩有许多不同的限制如下总结所示：

现有的标准的前提是两个设备之间有很多流量。在这里，我们假设有一个简单的并上下文不太相关的报头压缩。然而这运行独立于流量（可能有一些），这确实不使用流量相关的内容。因此，它的压缩效果没有单独为每个流量进行内容压缩的方法好。

由于报文大小的限制，对层 2 和层 3 进行压缩是非常有必要的，有些工作在传统上没有完成（虽然现在因为 ROHC（RObust 报头压缩）工作组面有所改变）。

IEEE 802.15.4 设备应该部署在多跳网路。然而，在传统的点对点链路场景下，压缩者和解压缩者是直接与对方通信并独占链路的，在 mesh 里的报头压缩方式显然是不同的。在一个 IEEE 802.15.4 网路中，一个设备有必要通过它的任一个邻居来发送压缩报头的报文，并尽可能减小预先内容处理。

报头压缩要求的任何新的报文格式，重用节 5 里使用不同的分派值来定义的基本报文格式。

报头压缩可能导致不能对齐 8 位组边界的情况。由于硬件通常不能传送小于一个字节的的数据单元，所以必须使用填充。填充按以下步骤：首先，排出整个邻接的压缩报头序列（本文只定义 IPv6 和 UDP 报头压缩机制，但其他的可能在别处定义）。然后，用 0 填充到合适的位置以便对齐 8 位位组的边界。这会消除任何由报头压缩所产生的不对齐，所以接下来的域（例如，非压缩报头或数据载荷）从一个 8 位位组边界开始并延续下去。

### 10.1、IPv6 报头域编码

由于加入到同一个 6LoWPAN 网路，设备共享一些状态。这使得可以压缩报头而不用明确创建任何内容状态的压缩。因此，6LoWPAN 报头压缩并不保留任何流量状态；反而，它依赖于与整个链路相关的信息。以下的 IPv6 报头值是针对 6LoWPAN 网路是通用的，所以 HC1 报头从一开始用于高效压缩：版本是 IPv6；IPv6 源和目标地址都是链路本地地址；源和目标地址的 IPv6 接口标识符（后 64 位）可以从第二层的源和目标地址（当然，这是只能由一个 802.15.4 MAC 地址来产生接口标识符）产生的；报文长度可以从第二层（IEEE 802.15.4 PPDU 里的“帧长”）或分片报头（如果有）的“datagram\_size”域来获得；交通等级和流量标签都是 0；下一个报头是 UDP，ICMP 或 TCP。IPv6 报头里总是需要完整安装的域是跳数限制（8 位）。一些不可压缩的域同样需要“嵌入的”安装（节 10.3.1），取决于报文与这种通用情况有多相近。这个通用 IPv6 报头（如上所述）能从 40 字节压缩到 2 字节（1 字节 HC1 编码和 1 字节跳数限制）。这样的报文是可通过 LOWPAN\_HC1 格式来进行压缩的，通过使用一个分派值 LOWPAN\_HC1 后面的 LOWPAN\_HC1 报头里“HC1 编码”域（8 位）来编码不同的组合，如下所示。这个报头前面可能是一个分片报头，再前面可能是一个 mesh 报头。

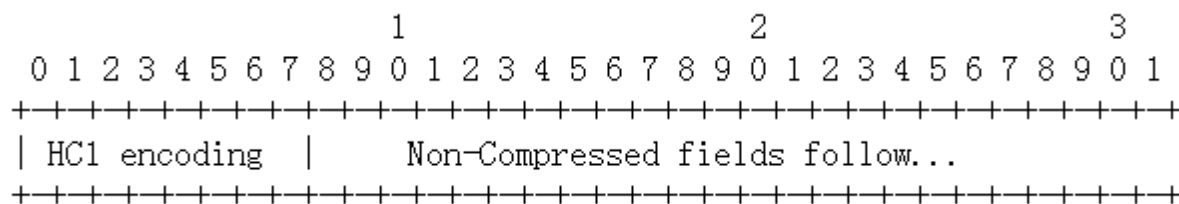


图 9：LOWPAN\_HC1（通用压缩报头编码）

如下面所示（位 7），一个 HC2 编码后面可能是一个 HC1 字节。在这种情况下，HC2 编码域后面是一个非压缩域（节 10.3）。

由“HC1 编码”产生的地址域解释如下所示：

PI: 内嵌的前缀（节 10.3.1）。

PC: 压缩的前缀（假定是链路本地前缀）。

II: 内嵌的接口标识符（节 10.3.1）。

IC: 省略的接口标识符（从相应的链路层地址获得的）。如果在一个 mesh 路由中使用源或目标地址的接口标识符，相应的链路层地址是“Mesh Addressing”域里的地址（节 5.2）。

“HC1 编码”如下所示（从位 0 到位 7）：

IPv6 源地址（位 0 和 1）：

- 00: PI, II
- 01: PI, IC
- 10: PC, II
- 11: PC, IC

IPv6 目标地址（位 2 和 3）：

- 00: PI, II
- 01: PI, IC
- 10: PC, II
- 11: PC, IC

交通等级和流量标签（位 4）：

- 0: 非压缩的；全 8 位的交通等级和 20 位的流量标签都会发送
- 1: 交通等级和流量标签都为 0

下一报头（位 5 和 6）：

- 00: 非压缩的；全 8 位都会发送
- 01: UDP
- 10: ICMP
- 11: TCP

HC2 编码（位 7）：

- 0: 没有更多的报头压缩位
- 1: HC1 编码后面有更多 HC2 编码格式的报头压缩位。位 5 和 6 确定了 HC2 编码的用途（例如，UDP，ICMP 或 TCP 编码）。

## 10.2、UDP 报头域编码

LOWPAN\_HC1 的位 5 和 6 允许对 IPv6 报头的下一报头域进行压缩（UDP，TCP 和 ICMP）。对这些协议的每一个报头进行进一步的压缩同样是可能的。本节解释 UDP 报头本身是如何被压缩的。本节的 HC2 编码就是 HC\_UDP 编码，并它只在 HC1 的位 5 和 6 指定的 IPv6 报

头后面的协议是 UDP 时才是有效的。HC\_UDP 编码（图 10）允许对 UDP 报头的以下域进行压缩：源端口，目标端口和长度。UDP 报头的检验和域是不可压缩的，因此它必须进行完整安装。下面的方法定义了如何把 UDP 报头从 8 字节压缩到 4 字节。

UDP 报头里唯一一个域值可以从其他可用信息里推导出来的是长度。其他域必须内嵌安装，以完整的或部分压缩的形式（节 10.3.2）。

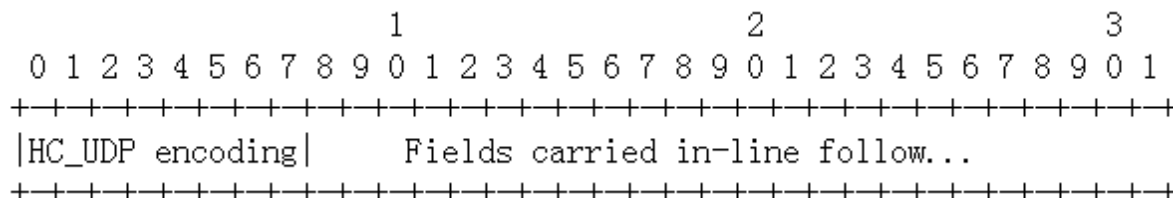


图 10: HC\_UDP (UDP 通用压缩报头编码)

UDP 的“HC\_UDP”编码如下所示（从位 0 到位 7）：

UDP 源端口（位 0）：

- 0: 非压缩的，“内嵌”安装（节 10.3.2）
- 1: 压缩到 4 位。实际的 16 位源端口通过计算得到： $P + \text{short\_port}$  值。 $P$  的值是 61616 (0xF0B0)。 $\text{short\_port}$  是一个“内嵌”安装的 4 位值（节 10.3.2）。

UDP 目标端口（位 1）：

- 0: 非压缩的，“内嵌”安装（节 10.3.2）
- 1: 压缩到 4 位。实际的 16 位目标端口通过计算得到： $P + \text{short\_port}$  值。 $P$  的值是 61616 (0xF0B0)。 $\text{short\_port}$  是一个“内嵌”安装的 4 位值（节 10.3.2）。

长度（位 2）：

- 0: 非压缩的，“内嵌”安装（节 10.3.2）
- 1: 压缩的，从 IPv6 报头长度信息来计算长度。UDP 长度域的值与 IPv6 报头的载荷长度再减去出现在 IPv6 报头和 UDP 报头之间的任何扩展报头的长度相等。

保留（位 3 到 7）

## 10.3、非压缩域

### 10.3.1、非压缩的 IPv6 域

这种方法允许把 IPv6 报头压缩到不同的程度。所以，只有非压缩域将会发送，而不是整个（标准）IPv6 报头。紧接着的报头（如原始 IPv6 报头里的下一个报头所指定的）跟在 IPv6 非压缩域后面。

非压缩的 IPv6 寻址是由一个包含 IPv6 分派值的分派类型来描述的，后面接的是非压缩 IPv6 报头。这个分派类型前面可能有附加的 LoWPAN 报头。

非压缩 IPv6 域里必须总是出现的是跳数限制（8 位）。这个域必须总是跟在编码域后面（如图 9 的“HC1 编码”），还可能包含其他未来的编码域）。其他非压缩域必须跟在跳数限制后面，如前面（节 10.1）所述的“HC1 编码”的顺序一样：源地址前缀（64 位）和/或接口标识符（64 位），交通等级（8 位），流量标签（20 位）和下一报头（8 位）。实际的下一报头（如 UDP，TCP，ICMP，等）跟在非压缩域后面。

### 10.3.2、非压缩的和部分压缩的 UDP 域

这个方法允许对 UDP 报头压缩到不同程度。因此，只有非压缩的或部分压缩的域需要发送，而不是整个（标准）UDP 报头。

UDP 报头里的非压缩或部分压缩域必须总是跟在 IPv6 报头和任何相关的内嵌域后面。任何 UDP 报头内嵌域必须和一个通用 UDP 报头[RFC0768]里的顺序一样，那就是，源端口，目标端口，长度和校验和。如果源或目标端口是“short\_port”格式的（如压缩 UDP 报头里指定的），内端口号只占 4 位，而不是占 16 位。

## 11、链路层 Mesh 的帧传送

即使 802.15.4 网路应该尽量使用 mesh 路由，但是 IEEE 802.15.4—2003 标准[ieee802.15.4]并没有定义这种功能。在这种情况下，全功能设备（FFDs）以 ad hoc 或 mesh 路由协议方式运行，形成自己的路由表（不在本文讨论范围内）。在这样的 mesh 场景下，两个设备不需要直接连接才能进行通信。对于这些设备，发送者表示为“发起端”，接收都表示为“目标端”。一个发起端设备可能通过其他中间设备来转发数据到目标端。为了使用单播完成这样的帧传送，除了跳与跳的源地址和目标地址外，还有必要包括发起端和目标端的链路层地址。

本节定义如何有效地在一个 mesh 中传送第二层帧，使用一个目标“目标端”链路层地址。

Mesh 传送成为可能要达到以下条件，一个 LoWPAN 封装（节 5）里的在其他报头之前的 Mesh 寻址报头，一个非分片的和分片的报头；一个完整的 IPv6 报头；或一个压缩的 IPv6 报头如节 10 或其他地方定义的那样。

如果一个节点希望使用一个默认的 mesh 转发节点来传送报文（也就是，它不能与目标节点直接通信），它必须包含一个 Mesh 寻址报头，报头发起端链路层地址是它自己，目标端链路层地址是报文的最终目标。它设置 IEEE802.15.4 报头里源地址为自己的链路层地址，目标地址为转发节点的链路层地址。最后就可传送报文。

类似地，如果一个节点接收到一个包含 Mesh 寻址报头的帧，它必须检查 Mesh 寻址报头里的“目标端”域来确定真正的目标。如果节点是最终目标，它如正常传送那样处理报文。如果它不是最终目标，节点会对“Hops Left”域减 1，如果结果是 0，丢弃报文。否则，节点查询自己的链路层路由表，以确定到达最终目标的下一跳节点是什么，然后把下一跳地址放到 802.15.4 报头的目标地址域里。最后，节点把 802.15.4 报头的源地址变成自己的链路层地址然后发送报文。



鉴于节点必须在一个 mesh 路由协议里充当转发者，不是所有节点都要求进行 mesh 转发。只有“全功能设备”（FFDs）才需要在 mesh 网路中充当路由器。“简化功能设备”（RFDs）只限于发现 FFDs 然后使用它们进行转发，形式上类似于 IP 主机通常使用默认路由来转发所有的无连接数据包。对于一个使用 mesh 传送的 RFD，“转发者”总是某一个恰当的 FFD。

### 11.1、LoWPAN 广播

在 mesh 报头后是一个具有 mesh 路由功能的路由报头。特别地，一个广播报头包含一个 LOWPAN\_BC0 分派和一个序列号。序列号用于检测重复报文（并抑制它们）。

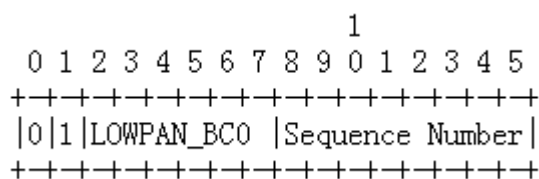


图 11：广播报头

域定义如下所示：

**Sequence Number:** 这个 8 位域由发起端进行递增操作，不管是发送一个新的 mesh 广播或多播报文。具体的操作方法不在本文范围内。

这个 mesh 层广播的进一步描述，如映射到流量控制机制，拓扑发现等都不在本文的讨论范围内。

另外的 mesh 路由功能，如具体的 mesh 路由协议，源路由等可以在增加额外的路由报头来实现分片和寻址功能。具体的 mesh 路由功能不在本文的讨论范围内。

## 12、IANA 事项

本文创建 2 个新的 IANA 注册，如下所示。这些注册的未来任务是与 IANA 协调，并遵守“必要说明” [RFC2434] 里的规则。这些规则允许其他（非 IETF）组织更容易得到任务分配。

本文为第 5 节定义的报头中分派类型域创建了一个新的 IANA 注册。本文定义了 IPv6 值，LOWPAN\_HC1 报头压缩，BC0 广播和两个逃脱模式（NALP 指示不是一个 LOWPAN 帧，ESC 允许额外的分派字节）。本文定义这个域为 8 位长。值 00xxxxxx 是保留的，总共有 192 个不同的值可用，这应该足够用了。如果定义了对于 HC1 报头压缩格式，或额外的 TCP，ICMP HC2 格式，那么这时候就应该在 LOWPAN\_HC1 之后使用保留的分派值。如果定义了额外的 mesh 传送格式，这就会使用 LOWPAN\_BC0 后的保留值。

本文为 6LoWPAN 报文中的 16 位短地址域创建了一个新的 IANA 注册。

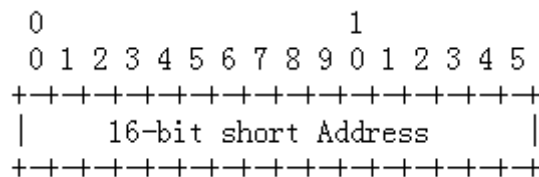


图 12

这个注册必须包含地址 0xffff（目前当监听通道时所有设备都接受的 16 位广播地址）和在 [ieee802.15.4]中定义的 0xfffe。另外，在 6LoWPAN 网路中，16 位短地址必须使用这种格式（从 0 到 7 的位序列），这里“x”是一位未指定的值：

范围 1，0xxxxxxxxxxxxxx：第一位（位 0）应该为 0 如果 16 位短地址是一个单播地址。这样有 15 位用于真实地址。

范围 2，100xxxxxxxxxxxxxx：位 0，1 和 2 应该使用这个格式如果 16 位短地址是一个多播地址（见节 9）。这样有 13 位用于真实多播地址。

范围 3，101xxxxxxxxxxxxxx：这个格式的位 0，1 和 2 是保留的。任何未来的分配应该遵守上面提到的规则。

范围 4，110xxxxxxxxxxxxxx：这个格式的位 0，1 和 2 是保留的。任何未来的分配应该遵守上面提到的规则。

范围 5，111xxxxxxxxxxxxxx：这个格式的位 0，1 和 2 是保留的。任何未来的分配应该遵守上面提到的规则。

### 13、安全事项

从 EUI-64MAC 地址生成的接口标识符的方法是为了保留一个全局唯一的地址当有需要的时候使用。然而，没有方法来预防故障或伪装产生的重复。

在 IEEE 802.15.4 链路的邻居发现可能容易受到[RFC3756]里所描述的一些威胁。在 IEEE 802.15.4 网路中 mesh 路由是很常用的。这意味着会有另外的威胁因为使用 ad hoc 路由 [KW03]。IEEE 802.15.4 提供了一些链路层的安全功能。用户应该尽可能的使用这些安全功能。这样有助于减少上面提到的威胁。

IEEE 802.15.4 设备大多数都应该在一个 PAN 进行通信（既是，在自己的链路内，用 IPv6 的方式来说）。考虑到成本和功耗问题，并保持 IEEE 802.15.4 模型的“简化功能设备”(RFDs)，这些设备通常只有最小部分必需的功能。因此，这些设备的安全性能很大程度上依赖于 IEEE 802.15.4 链路层的安全机制。后者，然而，只在 IEEE 802.15.4 帧上定义了高级加密算法(AES)用于认证或加密，但不提供密钥管理（大概是基于组的）。其他部署上的问题涉及到安全配置和管理。然而，一个完整的描述已经超出了本文的范围了，部署 IEEE 802.15.4 网路还是需要认真考虑安全问题的。当然，对于一些 IEEE 802.15.4 设备（被称为“全功能设备”，或“FFDs”）还是要有相应的安全功能。这些通常是使用非连接的 IPv6 进行通信的（相对于

更通用的在连接的交换)。这些 IPv6 设备使用平常的机制 (如 IPsec, TLS, 等) 来确保端到端的通信安全。

#### 14、致谢

感谢 RFC 2464 和 RFC 2734 的作者, 本文引用了部分他们的图案。感谢 Geoff Mulligan 的讨论, 帮助完成了本文。Erik Nordmark 的建议对报头压缩部分有很大帮助。同样感谢 Shoichi Sakane, Samita Chakrabarti, Vipul Gupta, Carsten Bormann, Ki-Hyung Kim, Mario Mao, Phil Levis, Magnus Westerlund, 和 Jari Arkko。

#### 15、参考文献

(注: 参考文献请看原文 <http://www.rfc-editor.org/rfc/rfc4944.txt>)

#### 附录 A、可选的 Mesh 传送帧

在最后选择一个 mesh 网路 (节 11) 的传送机制之前, 先考虑一下其他的可选方法。除了 hop-by-hop 源和目标链路层地址, 在 LoWPAN mesh 上传送一个报文还需要端到端的发起端和目标端的地址。这同样适合于第 2 层和第 3 层 (也就是 IP)。在后者情况, 本文不需要提供额外的报头支持 (也就是, 它本身包含在 LoWPAN 报头内)。链路层的目标地址会指向下一跳目标地址而 IP 报头的目标地址会指向最终目标 (IP) 地址 (可能距离源主机有很多跳), 源地址也是类似的。因此, 当转发数据时, 单跳源和目标地址会在每一跳后改变 (分别指向要转发的节点, 和 “最好” 的链路层下一跳), 而源和目标 IP 地址依然没有改变。注意, 如果一个 IP 报文被分片, 单个的分片可能不按顺序到达任一节点。如果第一个分片 (包含 IP 报头) 由于某种原因延迟了, 接收到其他分片的节点会缺少必要的信息。它会强制等待直到接收到 IP 报头 (在第一个分片内) 才能把报文进行转发。这要求中间节点增加额外的缓存空间。此外, 这一个规格只对一种类型的 LoWPAN 载荷有用: IPv6。通常, 它必须增加适配器以适用于其他载荷, 并要求载荷提供自己的端到端地址信息。

另一方面, 这种方法最后 (节 11) 在 LoWPAN 层 (层 3 以下) 建立了一种 Mesh。因此, 链路层的发起端和目标端地址都包含在 LoWPAN 报文里。这使得在 LoWPAN 适配层上 (节 5) 进行任何协议或应用数据的 mesh 传送。对于本文中支持的 IPv6, 在层 2 上使用发起端和目标端地址还有另一个好处, 那就是 IPv6 地址能够压缩, 如节 10 所述的报头压缩。此外, 维护路由表所需要的空间由于 802.15.4 地址 (64 位或 16 位) 的大小而减小, 相对于 IPv6 地址 (128 位)。一个不好的地方是, 在 IP 顶层的应用不使用链路层目标地址来封装报文。因此, 给定一个 IP 地址, 有必要进行转化为相应的链路层地址。于是, 一个 mesh 路由协议需要指定邻居发现方法, 虽然对于一些特殊的情况, 可以从层 3 地址导出层 2 的设备地址 (并可反向)。具体的说明不在本文的范围内。

#### 作者地址

(请参考原文 <http://www.rfc-editor.org/rfc/rfc4944.txt>)

版权声明

(请参考原文 <http://www.rfc-editor.org/rfc/rfc4944.txt>)

知识产权

(请参考原文 <http://www.rfc-editor.org/rfc/rfc4944.txt>)

本文由 zucheng10 翻译,

Email:[mzucheng10@qq.com](mailto:mzucheng10@qq.com)

<http://blog.csdn.net/zucheng10>

2014 年 9 月 1 日

PS:仅以学习目的翻译此文档, 由于水平有限, 翻译难免有错漏之处, 标准文档请参考英文原文。