



中华人民共和国通信行业标准

YD/T 2710-2014

IPv6 路由协议 适用于低功耗有损网络的 IPv6 路由协议(RPL)技术要求

IPv6 routing protocol—technical requirements of IPv6 routing
protocol for low-power and lossy networks(RPL)

(IETF RFC6550, RPL: IPv6 Routing Protocol for Low-Power and Lossy
Networks, MOD)

2014-10-14 发布

2014-10-14 实施

中华人民共和国工业和信息化部 发布

目 次

前 言	II
1 范围	1
2 规范性引用文件	1
3 术语、定义和缩略语	2
4 RPL 协议概述	4
5 RPL 支持的通信模式	10
6 RPL 实例	11
7 ICMPv6 RPL 控制消息	12
8 序列计数器	26
9 向上路由	27
10 向下路由	33
11 安全机制	40
12 转发数据包和环路避免/探测	45
13 多播操作	47
14 路由邻接的维护	48
15 目标函数的指导方针	48
16 邻居发现互操作的建议	49
17 互操作性的需求摘要	49
18 RPL 常量和变量	50
19 可管理性的注意事项	51
20 安全考虑	58

前 言

本标准按照GB/T1.1-2009给出的规则起草。

本标准使用重新起草法修改采用IETF RFC6550《低功耗有损网络IPv6路由协议》，本标准与IETF RFC6550的技术性差异及其原因如下主要差异如下：

——按照汉语习惯对一些编排格式进行了修改；

——将一些适用于国际标准的表述改为适用于我国标准的表述；

——根据GB/T 1系列的要求，增加了第1章、第2章和第3章，其中第3.1节从技术内容上与IETF RFC6550第2章保持一致；

——本标准的第4章从技术内容上与IETF RFC6555的第3章保持一致，只是考虑到我国使用习惯重新进行了描述；

——本标准的第5章～第20章分别与RFC6550的第4章～第19章保持一致。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别这些专利的责任。

本标准由中国通信标准化协会提出并归口。

本标准起草单位：工业和信息化部电信研究院、中国科学院计算技术研究所。

本标准起草人：马军锋、宋 菲、王煜伟、常 城、高 飞、李红阳、黄春秀。

IPv6路由协议

适用于低功耗有损网络IPv6路由协议（RPL）技术要求

1 范围

本标准规定了低功耗有损网络IPv6路由协议（RPL）的技术要求，包括协议基本操作、协议消息格式、协议字段属性、及协议报文处理等方面。

本标准适用于支持低功耗有损网络IPv6协议网关及节点。

2 规范性引用文件

下列文件对于本文件的应用是必不可少的。凡是注日期的引用文件，仅所注日期的版本适用于本文件。凡是不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

IETF RFC1958 互联网架构原则

IETF RFC3447 公共密钥标准(PKCS)#1: RSA 密钥规范2.1版(Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1)

IETF RFC4191 默认路由器优先选择和更加明确路由 (Default Router Preferences and More-Specific Routes)

IETF RFC4302 IP认证头 (IP Authentication Header)

IETF RFC4443 符合IPv6要求的ICMPv6 (Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification)

IETF RFC4862 IPv6无状态地址自动配置 (IPv6 Stateless Address Autoconfiguration)

IETF RFC5548 城市应用低功耗有损网络路由需求 (Routing Requirements for Urban Low-Power and Lossy Networks)

IETF RFC5673 工业应用低功耗有损网络路由需求 (Industrial Routing Requirements in Low-Power and Lossy Networks)

IETF RFC5706 新协议和协议扩展的考虑运行和管理指南 (Guidelines for Considering Operations and Management of New Protocols and Protocol Extensions)

IETF RFC5826 家庭自动化应用低功耗有损网络路由需求 (Home Automation Routing Requirements in Low-Power and Lossy Networks)

IETF RFC5867 楼宇自动化应用低功耗有损网络路由需求 (Building Automation Routing Requirements in Low-Power and Lossy Networks)

IETF RFC6551 适用于低功耗有损网络路径计算的路由度量 (Routing Metrics Used for Path Calculation in Low-Power and Lossy Networks)

IETF RFC6552 适用于低功耗有损网络路由协议的目标函数零 (Objective Function Zero for the Routing Protocol for Low-Power and Lossy Networks (RPL))

IETF RFC6553 用于在数据平面数据报文中承载RPL信息的RPL选项 (The Routing Protocol for Low-Power and Lossy Networks (RPL) Option for Carrying RPL Information in Data-Plane Datagrams)

IETF RFC6554 适用于低功耗有损网络路由协议的一种IPv6路由头 (An IPv6 Routing Header for Source Routes with the Routing Protocol for Low-Power and Lossy Networks (RPL))

3 术语、定义和缩略词

3.1 术语和定义

下列术语和定义适用于本文件。

3.1.1

有向无循环图 Directed Acyclic Graph (DAG)

一个所有边上没有循环存在的有向图。所有边都包含在定向的路径上, 指向并终结于一个或者多个根节点。

3.1.2

DAG根节点 DAG Root

DAG内没有外出边的节点。因为图是非循环的, 所以按照定义所有的DAG至少应有一个DAG根, 且所有路径终止于一个根节点。

3.1.3

面向目的地的有向无循环图 Destination Oriented DAG (DODAG)

以唯一目的地为根的DAG, 没有外出边。

3.1.4

DODAG的根节点 DODAG Root

在DODAG内部充当一个边界路由器, 完成DODAG内部聚合路由, 并重新分配DODAG路由到其他路由协议内。

3.1.5

虚拟DODAG的根节点 Virtual DODAG Root

一个虚拟的DODAG根是两个或两个以上的RPL路由器的集合, 如6LoWPAN边界路由器(6LBRs)。它协调同步DODAG状态, 确保集合内的RPL路由器采取一致行动。

3.1.6

向上 Up

从叶节点沿着DODAG的边向根的方向。

3.1.7

向下 Down

从DODAG根反向沿着DODAG的边叶节点的方向。

3.1.8

序列 Rank

一个节点的序列定义了该节点相对于其他节点距离一个DODAG根节点的位置。序列在向下方向上严格增加, 在向上的方向上严格减少。序列的具体计算方法是由DAG的目标函数来决定的。序列可以近似地记录一个简单的拓扑距离, 依据链路度量计算出来, 或考虑其他属性作为约束。

3.1.9

目标函数 Objective Function (OF)

定义了路由度量、最佳目的以及相关函数如何被用来计算出Rank值。此外，OF指出了在DODAG内如何选择父节点从而形成DODAG。

3.1.10

目标编码点 Objective Code Point (OCP)

表明DODAG使用哪一个目标函数的标识符。

3.1.11

RPL实例标识 RPL InstanceID

一个网络的唯一标识。具有相同RPL实例标识的DODAG共享相同的OF。

3.1.12

RPL实例 RPL Instance

共享同一个RPL实例标识的一个或者多个DODAG的一个集合。一个RPL节点在一个RPL实例中只能最多属于一个DODAG。每个RPL实例都独立其他RPL实例运行。

3.1.13

DODAG根的标识符 DODAGID

DODAGID在低功耗有损网络一个RPL实例中具有唯一性。数组[RPL InstanceID DODAGID]唯一确定了一个DODAG。

3.1.14

DODAG版本 DODAG Version

给定DODAGID的DODAG的版本。

3.1.15

DODAG版本号 DODAG Version Number

一个连续的计数器，由根递增形成一个新的DODAG版本。DODAG版本由唯一数组[RPL InstanceID DODAGID DODAG Version Number]确定。

3.1.16

目标 Goal

RPL范围之外定义的特定应用目标。DODAG的任何节点都要了解该目标，以便确定能否满足该目标。一个典型的目标是根据特定目标函数构建DODAG图，确保它与一组主机之间的连通性。

3.1.17

固定的 Grounded

当一个DODAG根能够满足目标时该DODAG是固定的。

3.1.18

浮动的 Floating

当DODAG不是固定的，那么它是浮动的。一个浮动的DODAG被预期没有能力满足目标，但可以为DODAG内其他节点提供连通性。

3.1.19

DODAG父节点 DODAG Parent

DODAG内的一个节点的父节点，是该节点通向DODAG根路径上的直接后继。父节点的序列比该节点的序列低。

3.1.20

DODAG子集 Sub-DODAG

一个节点的DODAG子集是另外一些节点的集合，这些节点通向DODAG根都应经过该节点。

3.1.21

本地DODAG Local DODAG

本地DODAG包含唯一一个根节点，并允许该根节点分配和管理一个RPL实例。本地DODAG是由一个本地RPL实例标识，不需要其他节点的调整。

3.1.22

全局DODAG Global DODAG

全局DODAG使用一个全局的RPL实例标识，可能被其他几个节点调整。

3.2 缩略语

下列缩略语适用于本文件。

6LoWPAN 适用于低功耗有损无线个域网的 IPv6 协议 IPv6 over Low-Power Wireless Personal Area Networks

CC	Consistency Check	一致性检验
DAO	Destination Advertisement Object	目的地通告对象
DAG	Directed Acyclic Graph	有向无循环图
DIO	DODAG Information Object	DODAG 信息对象
DIS	DODAG Information Solicitation	DODAG 信息请求
DODAG	Destination Oriented DAG	面向目的地的有向无循环图
DODAGID	DODAG Identity	DODAG 标识
IANA	Internet Assigned Numbers Authority	互联网号码分配中心
NBMA	Non Broadcast Multi Access	非广播多路访问
OCP	Objective Code Point	目标编码点
OF	Objective Function	目标函数
RPL	IPv6 Routing Protocol for Low-Power and Lossy Networks	适用于低功耗有损网络的路由协议

4 RPL 协议概述

4.1 拓扑

描述RPL基本拓扑以及拓扑形成的基本规则。

4.1.1 构造拓扑

低功耗有损网络中，如无线电网络，没有一个预先定义的网络拓扑，RPL应当先发现链路，然后以节约模式选择对等体。

在大多数情况下，2层拓扑只是部分重叠，RPL形成一个非可传递的/非广播多路访问（NBMA）网络拓扑，并基于该拓扑计算路由。

RPL对发往或来自一个或多个作为拓扑汇聚节点的根节点流量进行路由优化。RPL构造一个DAG拓扑，该拓扑被分割成一个或多个DODAG拓扑。每个汇聚节点对应一个DODAG。如果DAG有多个根，彼此之间通过一个公共的骨干链路相连接。

4.1.2 RPL 标识符

RPL使用4个值来标识和维护拓扑。

a) RPL实例标识

标识一组或多组DODAG。一个网络可能有多个RPLInstanceID，每个都定义一个相互独立的DODAG，基于不同的目标函数或应用进行优化。RPL实例是由RPLInstanceID标识的一组DODAG。在同一个RPL实例中的所有DODAG使用相同的目标函数。

b) DODAG标识

DODAGID的作用域是一个RPL实例。RPL实例标识和DODAGID结合能够唯一标识网络中的一个DODAG。一个RPL实例可能有多个DODAG，每个都有唯一的DODAGID。

c) DODAG版本号

DODAG版本号的作用域是一个DODAG。一个DODAG会通过增加其版本号从根节点开始重建DODAG。RPL实例标识、DODAGID和DODAG版本号组合在一起唯一标识一个DODAG版本。

d) 序列

序列的作用域是一个DODAG版本。序列在DODAG版本中建立了一种序列关系，定义了单个节点相对于DODAG根的位置。

4.1.3 实例，DODAGs 和 DODAG 版本

RPL实例包含一个或多个DODAG根。一个RPL实例可以提供到某个目的地前缀的路由，通过DODAG根或者是DODAG内的替换路由可达。这些根节点可以独立运行，也可以通过一个网络协同工作。

RPL实例构成：

a) 单个DODAG，单个根：

例如，家居自动化应用中，DODAG优化以减少单个集中灯光控制器的时延。

b) 多个不协同的DODAG，相互独立的根（不同的DODAGID）：

例如，在城市数据采集应用中，多个数据采集点之间没有适当的连接来彼此协同，或者是使用多个DODAG格式作为一种方式来动态自主的划分网络。

c) 单个DODAG，使用虚拟的根通过一个骨干网络来协同低功耗有损网络汇聚节点，这些汇聚节点具有相同的DODAGID。

例如，在一条可靠的转发链路上运行的多个边界路由器，支持6LoWPAN应用，能够作为相同DODAG汇聚节点的等价逻辑接口。

d) 以上相结合，适用于一些应用场景

每个RPL分组都与一个特定的RPL实例标识和RPL实例相关联。

图1描述了RPL的一个实例，包括3个DODAG以及R1、R2、R3三个根。每个DODAG根通告相同的RPLInstanceID。连线描述父节点与子节点之间的连接。

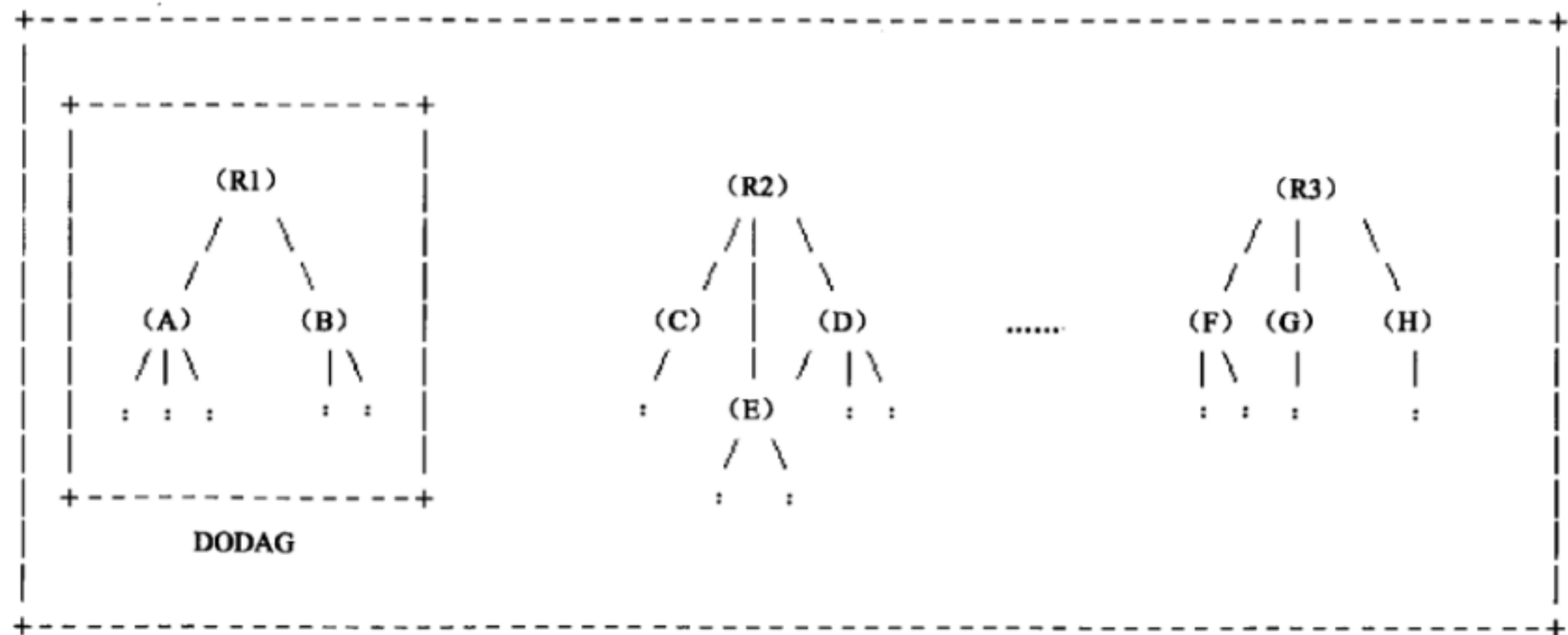


图 1 RPL 实例

图2描述了DODAG版本如何增加形成一个新的DODAG版本。以及一个新的DODAG拓扑。需要指出的是，新的DODAG版本并不意味着形成新的DODAG拓扑，但是新的DODAG拓扑要求一个新的DODAG版本。

需要注意的是，下面这个类似树形的结构描述是出于简化，尽管DODAG结构允许每个节点有多个父节点。

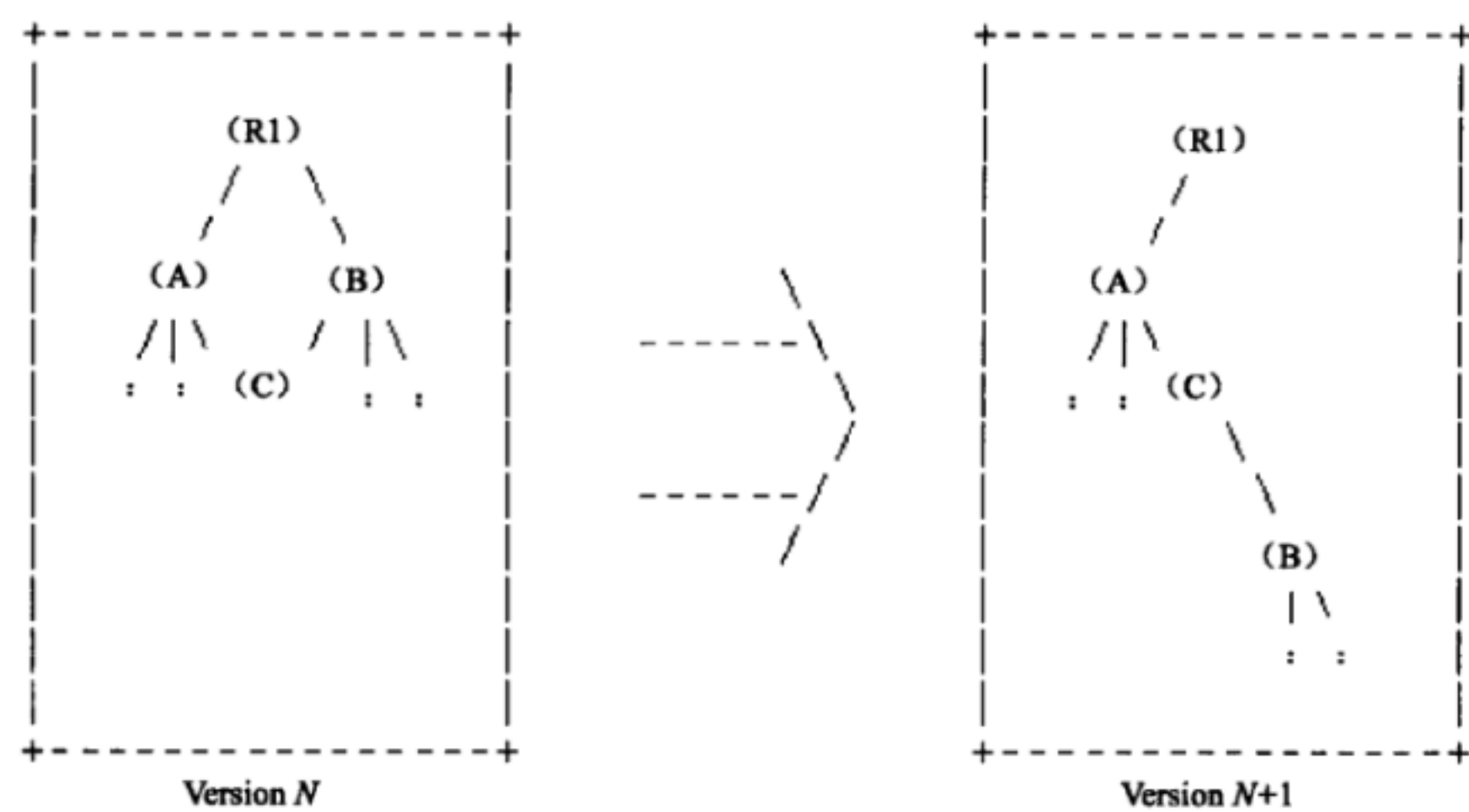


图 2 RPL 版本

4.2 上行路由和 DODAG 结构

RPL预置路由向上指向DODAG的根，根据一个目标函数（OF）形成一个优化的DODAG。RPL节点通过DODAG信息对象（DIO）消息构造和维护这些DODAG。

4.2.1 目标函数（Objective Function）

目标函数定义了RPL节点在一个RPL实例内如何选择和优化路由。目标函数通过DIO配置选项中的目标码点属性定义。一个目标函数定义节点如何将一个或多个路径值和约束转换成一个序列，该值近似于节点到DODAG根节点的距离，见IETF RFC6551。目标函数也定义了节点如何选择父节点，具体规定见IETF RFC6551和 IETF RFC6552。

4.2.2 DODAG 修复

DODAG根通过增加DODAG的版本号发起全局的修复操作。这将启动一个新的DODAG版本，该版本中的节点可以选择一个新的位置，其序列不会受到老DODAG版本序列值的影响。

RPL也支持在DODAG版本内的本地修复机制。DIO消息指定必要的参数依据DODAG根节点的配置和策略进行控制。

4.2.3 安全

RPL支持消息的机密性和完整性。RPL使用三种基本的安全模式。

a) 不安全，RPL控制消息在没有任何附加安全机制的情况下发送。不安全模式并不意味着RPL网络是不安全的，它可以使用其它现有的安全机制（如链路层安全）来满足应用安全需求。

b) 预置安装，节点加入RPL实例时有预安装的密钥，使得节点能够处理和产生安全RPL的消息。

c) 已认证，在已认证模式下，节点具有预安装的密钥，但是预安装的密钥只能作为叶节点加入RPL实例。作为一个路由器加入一个已认证的RPL实例，要求从一个已认证实体获得一个密钥。

4.2.4 固定和浮动的 DODAG

DODAG可能是固定的或者是浮动的。固定的DODAG提供到主机的连通性，要求满足预定义的应用目标。浮动的DODAG不要求满足目标，大多数情况下只提供到DODAG内的路由。例如，浮动DODAG可以用于在修复过程中保留内部连接。

4.2.5 本地 DODAG

RPL节点通过形成一个本地的DODAG，能够优化到低功耗有损网络内目的节点的路由。本地DODAG的根是期望到达的目的地。本地DAG有且仅有一个DODAG，因此也只有一个DODAG根。本地DODAG能够按需构造。

4.2.6 管理优先级

一个实现/部署通过管理优先级可以指定一些DODAG根节点优先于其他根节点被使用。管理优先级提供一种控制流量和DODAG工程信息的方式，以便更好的支持应用需求。

4.2.7 数据路径验证和环路检测

低功耗有损网络的低功耗、有损耗特性促使RPL使用数据分组进行按需环路检测。由于数据流量交互可能不频繁，所以维护路由拓扑与物理拓扑相一致会浪费能量。典型的低功耗有损网络呈现那些短暂的、不影响流量转发的物理连接变化，但是从控制平面来看成本是较高的。短暂的、不频繁的连接变化不需要RPL实时跟踪，直到有数据发送。

RPL分组信息通过数据分组传送，包含发送者的序列。一个分组（上行或下行）路由决策与两个节点之间的序列关系不一致表明一个潜在的环路。当接收到这样一个分组，节点触发一个本地修复操作。

例如，如果节点沿着上行方向收到一个带标签的分组，且分组记录发送者的序列比接收者的序列低，那么接收节点能够推断出分组不在上行方向，DODAG不一致。

4.2.8 分布式算法运行

构造DODAG图的分布式算法概述如下：

a) 一些节点配置成DODAG的根；

b) 发送本地链路组播DIO消息到所有RPL节点，用来通告节点的存在、所归属DODAG、路由代价、以及相关度量值；

c) 节点侦听DIO消息, 使用其信息加入一个新的DODAG (因此选择DODAG父节点), 或者根据指定的目标函数和其邻居的序列来维护已有的DODAG;

d) 节点通过DODAG版本中的父节点预置路由表项给DIO消息中指定的目的地。决定加入一个DODAG的节点可以预配置一个或多个DODAG父节点作为缺省路由的下一跳以及相关实例外部路由的下一跳。

4.3 下行路由和目的地通告

RPL使用目的通告对象(DAO)消息建立下行路由。DAO消息是点到多点或点到点通信应用的一个可选特征。RPL支持两种下行流量模式: 存储(完全状态)模式或非存储(完全源路由)模式。任一给定的RPL实例要么是存储模式, 要么是非存储模式。在两种模式下, 点到点的分组经由上行路由到DODAG根, 下行路由到最终的目的地(除非目的地在上行路由上)。在非存储模式下, 分组向下转发之前会沿各个方向到DODAG根。在存储模式下, 分组在到达DODAG根之前, 可以通过一个源和目的地的一般祖先节点直接向目的地转发。

4.4 本地 DODAG 路由发现

在一个RPL网络中, 到一个特定的目的地可选支持按需的DODAG发现。本地的DODAG行为与全局的DODAG行为存在细微差别: 本地DODAG是通过DODAGID和RPL实例标识相结合来唯一定义的。RPL实例标识指示一个DODAG是否是一个本地的DODAG。

4.5 Rank 属性

节点的序列是一个节点在DODAG版本内位置的层级表示。序列通常用于避免和检测环路, 并表明特定的属性。序列的准确计算依据目标函数, 并且应实现通用的属性。

特别是, 节点的序列必须依照到DODAG目的地的DODAG版本单向递减。在这种情况下, 序列能够被作为一个DODAG版本内节点位置的层级表示。

目标函数如何计算节点的序列超出了本标准。

序列不是一个路径的代价值, 尽管它的值受到路径权值的影响, 并能够根据该值推算出。序列有下述一些自身的属性:

- a) 类型: 是一个摘要数值。
- b) 功能: 序列是一个节点在一个DODAG版本中相对于邻居的相对位置表示, 并不是一个到根节点距离或者是路径代价的必要或适当表示。
- c) 稳定性: 序列的稳定性决定了路由拓扑的稳定性。
- d) 属性: 序列严格按照递增的模式增加, 用于验证来自或者是去向根节点的进程。
- e) 摘要: 序列没有一个物理单元, 而是每一跳增加的一个范围, 每次增加值的分配都是由目标函数来确定。

根据RPL环路避免策略, 序列值为DODAG父节点的选则提供依据。一旦增加一个父节点, 同时节点在DODAG内的序列值被通告, 与DODAG父节点选择相关的更多选项和节点在DODAG内的移动被限制, 以避免造成环路。

4.5.1 序列比较 (DAGRank ())

序列的值是定点数, 整数部分和小数部分之间小数点的位置由MinHopRankIncrease值决定。MinHopRankIncrease是节点和它的任一DODAG父节点之间序列的最小增加值。DODAG根节点规定

MinHopRankIncrease. MinHopRankIncrease会在跳的精确开销和网络能够支持的最大跳数之间做一个平衡。例如，一个非常大的MinHopRankIncrease允许序列上给定跳准确特性，但是不支持多跳。

当目标函数计算序列时，目标函数会运行于整个序列数。当序列被比较时，例如决定父子关系或环路检测，使用序列的整数部分。序列的整数部分按照DAGRank（）宏来计算：

$\text{DAGRank}(\text{rank}) = \text{floor}(\text{rank}/\text{MinHopRankIncrease})$

Floor(x) 函数计算小于或等于x的最大整数。

如果 $\text{DAGRank}(A) < \text{DAGRank}(B)$ ，那么节点A的序列小于节点B的序列。

如果 $\text{DAGRank}(A) = \text{DAGRank}(B)$ ，那么节点A的序列等于节点B的序列。

如果 $\text{DAGRank}(A) > \text{DAGRank}(B)$ ，那么节点A的序列大于节点B的序列。

4.5.2 序列关系

在低功耗有损网络中，任一有邻接关系的节点M和N，通过序列计算维护下面的属性：

a) $\text{DAGRank}(M) < \text{DAGRank}(N)$ ：在这种情况下，节点M的位置比节点N更接近于DODAG的根。节点M可以作为节点N的DODAG父节点，不会产生环路。而且，对于节点N，在DODAG所有父节点集中的父节点的序列一定比 $\text{DAGRank}(N)$ 小。换句话说，节点N的序列一定比它的父节点序列大；

b) $\text{DAGRank}(M) = \text{DAGRank}(N)$ ：在这种情况下，节点M和节点N在DODAG中与DODAG根节点的位置是相同的。经由相同序列节点的路由可能会引起路由环路；

c) $\text{DAGRank}(M) > \text{DAGRank}(N)$ ：在这种情况下，节点M距离DODAG的根比节点N要远。节点M实际上在节点N的DODAG子集中。如果节点N选择节点M作为DODAG的父节点，可能会产生路由环路。

4.6 RPL使用的路由权值和约束

在路由协议中使用路由权值计算最短路由。内部网关路由协议（IGP），如IS-IS和OSPF使用静态的链路权值。链路权值只是简单的反应链路的带宽或者是根据定义的不同链路属性值通过计算得到。一些路由协议支持多个权值：在大多数情况下，一个权值用于一个拓扑。有些情况，第二个权值用于等价多路径计算。多个权值的优化被公认为是一个NP完全问题，有时在集中路径计算引擎中能够支持。

相比较而言，低功耗有损网络要求支持静态和动态权值。而且，链路和节点的权值也是必须的。在RPL中，不大可能定义一个权值或一个组合的权值来满足所有的用户案例。

而且，RPL支持基于约束的路由，约束条件被应用于链路和节点。如果一个链路或节点不能满足要求的约束条件，它将从后备的邻居集中裁剪掉，因此生成一个基于约束的最短路径。

目标函数用于指定计算路径的目标。此外，节点通过配置支持某种度量和约束，并通过DIO消息中通告的度量和约束选择它们在DODAG中的父节点。上行和下行度量可以被合并或者分别通告，这取决于OF和度量。当它们被分别通告时，可能会出现DIO父节点集不同于DAO父节点集的情况。然而，所有父节点都是遵循序列计算的规则。

目标函数与RPL所使用路由度量和约束是松耦合关系。实际上，OF规定了一些规则例如DODAG父节点选择，负载平衡等，所使用度量和/或者约束集，从而决定首选路径，都是基于DIO消息中DAG容器选项携带的信息。

所支持的链路/节点约束和度量在IETF RFC6551中规定。

例1：最短路径：提供最短端到端延时的路径。

例2：最短约束路径：不穿过任何电池供电节点，优化路径可靠性的路径。

4.7 回路避免

当拓扑发生变化时RPL尝试避免生成回路，采用基于序列的数据链路确认机制检测回路。实际上，这意味着RPL既不能保证回路自由路径选择，也不能保证严格的延时收敛时间，但是能检测和修复回路。RPL使用这个回路检测机制来确保在DODAG版本内的数据包转发过程，并在必要时触发回路修复。

4.7.1 贪婪和不稳定性

如果一个节点试图在一个DODAG版本内部移动到更深的地方，以增加自己潜在父节点的数量，这种情况称该节点是“贪婪”的。一旦一个节点加入了一个DODAG版本，RPL禁止这种行为的发生，以保持DODAG版本的稳定性。

4.7.2 DODAG 循环

当一个节点从DODAG中分离出并重新附着于它的一个DODAG子集的设备上，会发生DODAG循环。当DIO消息丢包时也会发生这种情况。严格的使用DODAG版本号可以消除这种循环，但当使用一些本地修复时仍然可能发生。

4.7.3 DAO 循环

当一个父节点接收和处理来自于其子节点DAO消息建立路由时，但随后子节点清除了相关路由状态，这时会发生DAO循环。RPL的环路检测机制缓解了这种循环并触发修复机制。

5 RPL 支持的通信模式

RPL支持三种基本的通信模式：多点到点（MP2P）、点到多点（P2MP）和点到点（P2P）。

5.1 多点到点通信模式

多点到点（MP2P）通信在许多低功耗有损网络应用中占据着主导位置。MP2P通信的目的地是那些特定的具有一些应用意义的节点，例如提供到大规模网络或者核心私有IP网络的连通性。RPL支持MP2P通信，允许经过DODAG根到达MP2P的目的地。

多点到点（MP2P）通信在许多低功耗有损网络应用方式在IETF RFC5867、IETF RFC5826、IETF RFC5673、IETF RFC5548中有规定。

5.2 点到多点通信模式

点到多点通信（P2MP）是几个低功耗有损网络应用所需求的通信样式。RPL通过使用一种目的地通告机制支持P2MP通信，该机制提供朝向目的地且远离根的下行路由（前缀，地址，或多播组）。目的地通告能随着DODAG的拓扑改变更新路由表。

5.3 点到点通信模式

RPL DODAG为点到点（P2P）通信提供了一种基本结构。对于一个RPL网络要支持P2P通信，根节点必须能够路由到目的地。网络中的节点可能有同样的路由表。朝向一个根节点的数据流，直到到达它的祖先节点才能知道到达目的地的路由。在大多数受到约束的情况下（当节点不能存储路由时），那个普通的祖先节点可能成为DODAG根。其他情况下可能是一个靠近源和目的地的节点。

RPL同样支持P2P目的地是‘一跳’邻居的情况。

RPL既不指定也不排除额外的机制来计算和安装潜在的更多最佳路由以支持任意的P2P通信。

6 RPL 实例

一个给定的低功耗有损网络内，可能存在多个逻辑上独立的RPL实例。一个RPL节点可能属于多个RPL实例，可能作为一个路由器或者叶节点。

有两种类型的RPL实例：本地和全局的。RPL把RPL实例标识空间划分为全局和本地实例，以便允许协调分配或单边分配RPL实例标识。全局RPL实例是协调分配的，拥有一个或多个DODAG，是典型的长期存在的。本地RPL实例总是单一的一个DODAG，该DODAG的单一根拥有相应的DODAGID，并以一种单边的方式分配本地RPL实例标识。本地RPL实例能够被用在例如构建DODAG以便支持未来的按需路由解决方案。

RPL网络内的控制和数据包被明确标记它们所属的RPL实例。

每一条RPL控制消息都有一个RPL实例标识字段。一些RPL控制消息，当与如下面定义的一个本地RPL实例标识有关时，可能也包含一个DODAGID。

RPL网络中数据包显示RPL实例标识作为RPL所需求的RPL数据包信息。对于来自于RPL网络外部的数据包，入口路由器判断它的RPL实例标识并把它注入到RPL网络中的合成数据包。

6.1 RPL 实例标识

全局RPL实例标识必须在整个低功耗有损网络中是唯一的。整个网络中能够存在128个全局实例。本地实例通常用来和一个DODAGID互相协调，每个DODAGID中能支持64个本地实例。本地实例由属于DODAGID的节点分配和管理，与其他节点没有任何明确的协调，详细描述如下。

一个全局RPL实例标识在RPL实例标识字段中编码如图4所示。

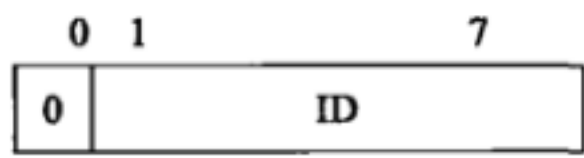


图 4 全局实例的 RPL 实例 ID 字段格式

一个本地RPL实例标识由拥有该DODAGID的节点自动配置，且对于该DODAGID来讲必须唯一。用来配置本地RPL实例标识的DODAGID必须是节点的一个可到达IPv6地址，且必须被当做所有本地实例中通信的终点。

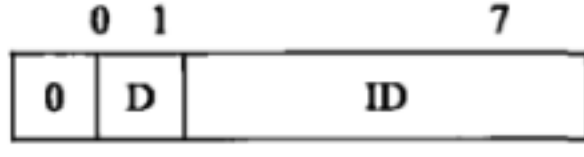


图 5 本地实例的 RPL 实例 ID 字段格式

图5中本地RPL实例标识中的D标志位在RPL控制消息中总是被设置为0.在数据包中它被用来表明该DODAGID是数据包的源或目的地址。如果被设置为1那么IPv6数据包的目的地址一定是该DODAGID。如果是0那么IPv6数据包的源地址一定是该DODAGID。

例如，假设节点A是一个本地RPL实例的DODAG根节点，并且已经分配了一个本地RPL实例标识。按照定义，所有穿过该本地RPL实例的流量将会起源或者终止于节点A。DODAGID在这种情况下将会是节点A的可到达的IPv6地址，且所有流量将会包含节点A的地址，DODAGID，源或者目的地址。因此本地RPL实例标识通过适当地设置D标志位，可能表示DODAGID等同于源地址或者目的地址。

7 ICMPv6 RPL 控制消息

本章定义了RPL控制消息，一种新的ICMPv6消息（见IETF RFC4443）。一个RPL控制消息由代码标识，并基于该代码和一系列选项组成。

大部分的RPL控制消息有一个链接范围。唯一的例外是非存储模式中的DAO/DAO-ACK消息，它们使用基于多跳的单播地址来交互，因此使用全局或者唯一本地地址作为源和目的地地址。对于其他的RPL控制消息，源地址为一个本地链路地址，目的地址为all-rpl-nodes多播地址或者本地链路地址。RPL多播地址是一个值要求为FF02::1A的新地址。

依照IETF RFC4443的规定，RPL控制消息由一个ICMPv6的头和消息身体组成。消息主体包含一个消息基础部分和可选项，如图6所示。

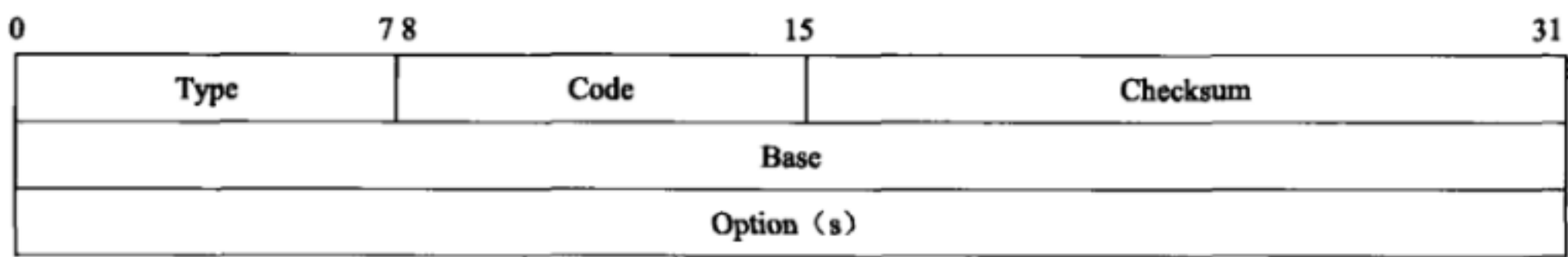


图 6 RPL 控制消息

RPL控制消息是一个类型值要求为155的ICMPv6信息消息。

code字段标识了RPL控制消息的类型。本标准定义了下面RPL控制消息类型的代码。

- 0x00: DODAG信息请求;
- 0x01: DODAG信息对象;
- 0x02: 目的地通告对象;
- 0x03: 目的地通告对象确认;
- 0x80: 安全DODAG信息请求;
- 0x81: 安全DODAG信息对象;
- 0x82: 安全目的地通告对象;
- 0x83: 安全目的地通告对象确认;
- 0x8A: 一致性检验。

如果一个节点收到了一条带有未知代码字段的RPL控制消息，节点必须丢掉该消息而不进行任何处理，可能提高一个管理的警报，一定不会发送任何回复消息。

校验和由IETF RFC 4443中规定计算。对于下面规定的RPL安全操作它被设置为0，而一旦RPL消息的剩余部分包括安全字段全被设置后它将被计算。

代码的高阶位（0x80）表示了RPL消息是否启用了安全。安全RPL消息有一个格式来支持机密性和完整性，如图7所示。

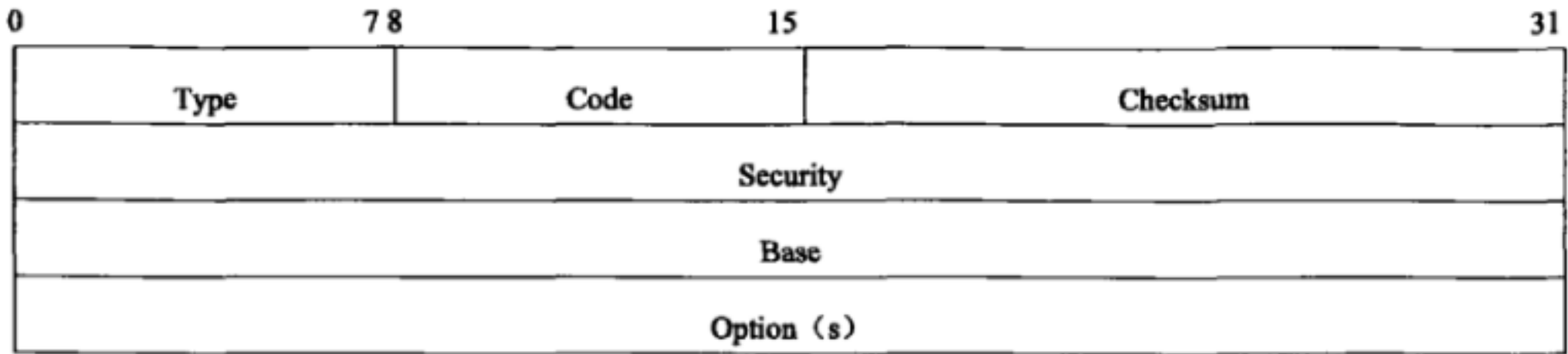


图 7 安全 RPL 控制消息

7.1 RPL 安全字段

每一条RPL消息都有一个安全变量。安全变量提供完整性和重播保护以及可选的机密性和延时性保护。因为安全覆盖了基础消息和选项，在安全消息中安全信息位于校验和与基础选项之间，如图7所示。所使用的安全级别和算法在协议消息中表明，如图8所示。

0	1	7	8	15	17	20	23	31
T	Reserved	Algorithm		KIM	Resvd	LVL	Flags	
Counter								
Key Identifier								

图8 安全部分

消息标识代码（MACs）和签名在整个非安全的ICMPv6 RPL控制消息中提供证明，包括所有定义字段的安全部分，但是ICMPv6校验和临时设置为0。加密提供了安全RPL ICMPv6控制消息的机密性，范围从安全部分之后的第一个字节开始到数据包的最后一个字节。安全转换生成了一条包含安全字段（MAC，签名等）的安全ICMPv6 RPL消息。换句话说，安全转换自身将会明确如何合并加密字段到安全数据包中去。安全本分自身没有明确的携带这些加密字段。

计数器是定时器（T）：如果计数器是定时器标志位被设置那么计数器字段是一个时间戳。如果标志位被清0那么是一个递增的计数器。

保留：7位非使用字段。字段必须被发送方初始化为0，并且被接收方忽略。

安全算法：安全算法字段规定了网络的加密，MAC，和签名计划。该字段支持的值如图9所示。

Algorithm	Encryption/MAC	Signature
0	CCM with AES-128	RSA with SHA-256
1~255	Unassigned	Unassigned

图9 安全算法编码

密钥标识符模式（KIM）：密钥标识符模式是一个2位的字段，用来表明用于数据包保护的密钥是含蓄还是直接地判定，以及详细表明密钥标识字段。密钥标识模式将会被设置成图10中的一个值。

Mode	KIM	Meaning	Key Identifier Length (octets)
0	00	Group key used, Key determined by key Index field, Key Source is not present, Key Index is present,	1
1	01	Per-pair key used, Key determined by source and destination of packet, Key Source is not present, Key Index is not present,	0
2	10	Group key used, Key determined by Key Index and Key Source Identifier, Key Source is present, Key Index is present,	9
3	11	Node's signature key used, If packet is encrypted, It uses a group key, Key Index and Key Source Specify key, Key Source may be present, Key Index may be present,	0/9

图10 密钥标识模式编码

在模式3中（KIM=11），密钥源和密钥标识符的存在和缺失由下面描述的安全级别决定。如果安全级别表明是加密的，那么字段存在；如果表明没有加密，那么字段不存在。

Resvd: 3位非使用字段。字段必须被发送方初始化为0，并且被接收方忽略。

安全级别: 安全级别是一个3位的字段用来表明提供数据保护。这个值能够适用于每一个数据包基础，允许数据真实性的级别的变化，可选择性的，数据机密性。KIM字段表明签名是否被使用和级别字段的含义。注意到安全级别的分配值未必是有序的，一个值高的LVL未必等同于高的安全。安全级别被设置成图11中的一个值。

KIM=0,1,2		
LVL	Attributes	MAC Len
0	MAC-32	4
1	ENC-MAC-32	4
2	MAC-64	8
3	ENC-MAC-64	8
4~7	Unassigned	N/A

KIM=3		
LVL	Attributes	Sig Len
0	Sign-3072	384
1	ENC-Sign-3072	384
2	Sign-2048	256
3	ENC-Sign-2048	256
4~7	Unassigned	N/A

图 11 安全级别编码

MAC属性表明消息有一个指定长度的消息标识代码。ENC属性表明消息是被加密的。签名属性表明消息有一个指定长度的签名。

标志位: 8位保留字段。字段必须被发送方初始化为0，并且被接收方忽略。

计数器: 计数器字段表明非重复4字节值用来构造加密机制，以便用来执行数据保护并考虑语义安全的规定。

密钥标识符: 密钥标识符字段表明使用哪一种密钥来保护数据包。该字段提供不同数据包保护级别，包括点到点密钥，群密钥，和签名密钥。这一字段如同密钥标识模式字段描述的那样并遵循如图12的格式。

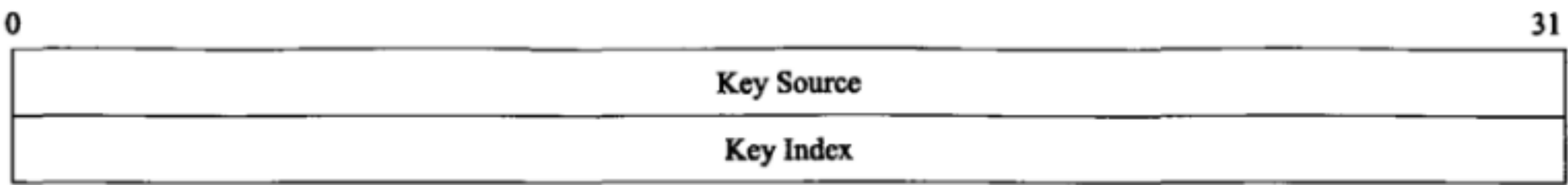


图 12 密钥标识

密钥源: 密钥源字段，当存在时，表明一个群密钥发起者的逻辑标识。当该字段存在时以8字节长度计数。

密钥索引：密钥索引字段，当存在时，允许同一发起者不同密钥使用唯一标识。每一个密钥发起者的责任是确保所使用的密钥的活跃性且有明确的索引，并且确保所有的密钥目录的值不为0x00。值0x00是预留给预置安装的共享密钥。当该字段存在时为1字节长度。

安全部分的未分配位是保留的。发送时它们必须被设置成0且接收时必须被忽略掉。

7.2 DODAG 信息请求 (DIS)

DODAG信息请求消息可以被用来向一个RPL节点请求一个DODAG信息对象。它的使用类似于IPv6邻居发现中的路由请求；一个节点可以使用DIS来探测附近DODAGs的邻居。

7.2.1 DIS 基础对象的格式

DIS 基础对象的格式如图 13 所示。

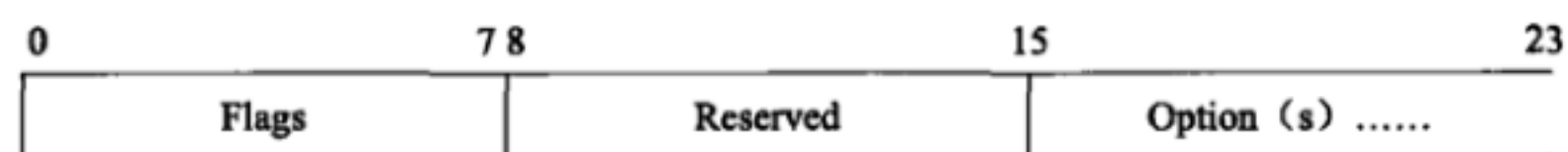


图 13 DIS 基础对象

标志位：8位保留字段。字段必须被发送方初始化为0，并且被接收方忽略。

保留位：8位保留字段。字段必须被发送方初始化为0，并且被接收方忽略。

DIS基础部分的未分配位是保留的。发送时它们必须被设置成0接收时必须被忽略掉。

7.2.2 安全 DIS

一条安全DIS消息遵循图7的格式，其中DIS消息中基础部分的格式如图13所示。

7.2.3 DIS 选项

DIS消息可能携带有效选项。

本标准考虑DIS消息携带下面选项：

- 0x00 Pad1；
- 0x01 PadN；
- 0x07 请求信息。

7.3 DODAG 信息对象 (DIO)

DODAG信息对象携带有关一个节点发现一个RPL实例，学习它的配置参数，选择一个DODAG父集，和维护DODAG的信息。

7.3.1 DIO 基本对象的格式

DIO 基本对象的格式如图 14 所示。

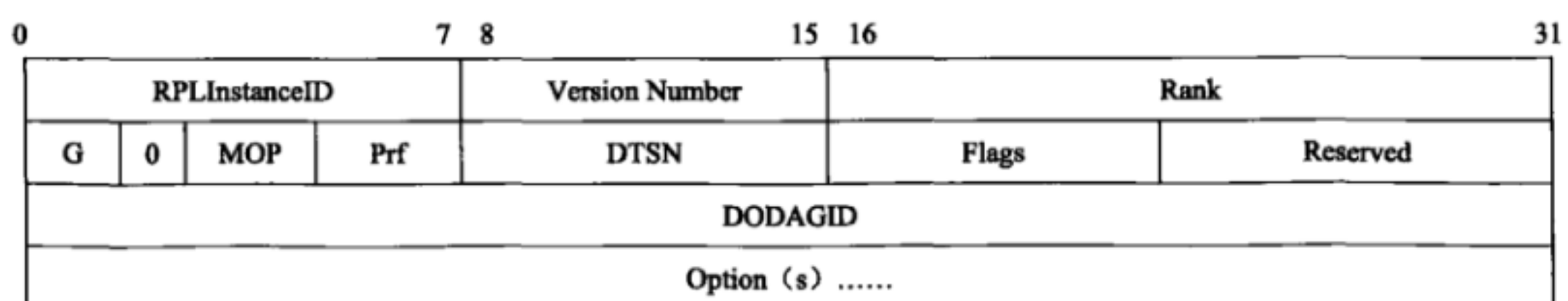


图 14 DIO 基本对象

Grounded (G) :表明通告的DODAG是否满足应用定义目标。如果设置，DODAG为grounded。如果清除，DODAG为floating。

Mode of Operation (MOP):表明RPL实例的操作模式是由DODAG根节点管理预分配和部署。所有加入DODAG的节点必须遵守 MOP以便充分地以路由器身份或者仅仅以叶子的身份参与。MOP编码如图 15所示。

MOP	Meaning
0	No downward routes maintained by RPL
1	Non storing mode
2	Storing without multicast support
3	Storing with multicast support
	All other values are unassigned

说明：值为0表明目的地通告消息是不可用的且DODAG仅仅维护向上路由

图 15 操作模式编码

DODAGPreference (Prf)：3位无符号整数。定义同一实例中如何选出本DODAG中最合适的根与其他DODAG的根做比较。范围从0x00到0x07，默认为0x00。

Version Number:8位无符号整数，由DODAG根设置DODAGVersionNumber。

Rank：16位无符号整数，表明发送DIO消息节点的DODAG的序列。

RPL实例标识:8位字段，由DODAG根设置表明属于DODAG的哪一个RPL实例。

Destination Advertisement Trigger Sequence Number (DTSN) :8位无符号整数，由节点设置发布DIO消息。DTSN用来作为维护向下路由的一个部分。

Flags：8位无符号保留字段。字段必须被发送方初始化为0，并且被接收方忽略。

Reserved：8位非使用字段。字段必须被发送方初始化为0，并且被接收方忽略。

DODAGID:由DODAG根设置的128位IPv6地址，唯一标识一个DODAG。DODAGID必须是一个属于DODAG根的可路由的IPv6地址。

DIO基础的未分配位是保留的。它们必须被发送方初始化为0，并且被接收方忽略。

7.3.2 安全 DIO

一条安全DIO消息遵循图7的格式，其中DIO消息中的基础格式如图14所示。

7.3.3 DIO 选项

DIO消息可能携带有效选项。

本标准考虑DIO消息携带下面选项：

- 0x00 Pad1;
- 0x01 PadN;
- 0x02 度量约束;
- 0x03 路由信息;
- 0x04 DODAG 配置;
- 0x08 前缀信息。

7.4 目的地通告对象 (DAO)

DAO向上沿着DODAG用来传播目标节点信息。存储模式中DAO消息是由子节点单播到所选父节点。非存储模式中DAO消息是单播到DODAG根节点。DAO消息可能是可选的，根据明确的需求或错误，由目标节点返回一个目的地通告确认消息给DAO的发送者。

7.4.1 DAO 基础对象的格式

DAO 基础对象的格式如图 16 所示。

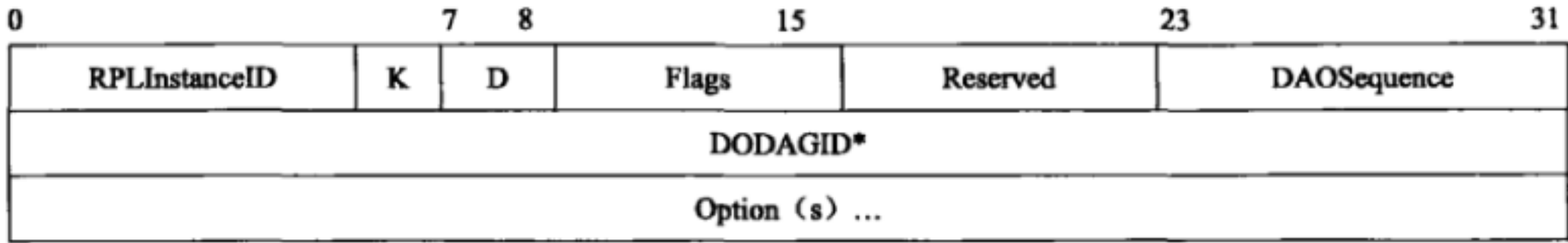


图 16 DAO 基本对象

RPL实例标识: 8位字段表明与DODAG关联的拓扑实例，从DIO消息中学习。

K: ‘K’ 标志位表明接收方被期望发送一个DAO-ACK回复消息。

D: ‘D’ 标志位表明DODAGID是存在的。当一个本地RPL实例标识被使用时该标志位必须被设置。

Flags: 6位未使用保留字段。字段必须被发送方初始化为0，并且被接收方忽略。

Reserved: 8位未使用字段。字段必须被发送方初始化为0，并且被接收方忽略。

DAOSequence: 在节点中每条唯一的DAO消息中递增并在DAO-ACK消息中重复。

DODAGID (可选): 由DODAG根设置的128位无符号整型，用来唯一标识一个DODAG。只有当 ‘D’ 标志位设置时该字段才存在。本字段只在当一个本地RPL实例标识使用时才存在，用来标识出与RPL实例标识关联的DODAGID。当使用一个全局的RPL实例标识时本字段没有必要存在。

DAO基础中未分配的位用来保留。它们必须被发送方初始化为0，并且被接收方忽略。

7.4.2 安全 DAO

一条安全DAO消息遵循图7的格式，其中DAO消息中的基础格式如图16所示。

7.4.3 DAO 选项

DAO消息可能携带有效选项。

本标准考虑DAO消息携带下面选项：

- 0x00 Pad1;
- 0x01 PadN;
- 0x05 RPL 目标;
- 0x06 传输信息;
- 0x09 RPL 目标描述符。

一个DAO消息的特殊情况，术语叫做No-path，出现在存储模式中用来清除向下路由状态，已经通过DAO操作预分配了。No-path携带一个目标选项和一个生存时间为0x00000000的关联传输信息选项，来表明对目标的可达性的丢失。

7.5 目的地通告对象确认

DAO-ACK消息由DAO接收方（一个DAO父节点或者DODAG根）以一个单播数据包的形式发送，用来响应单播DAO消息。

7.5.1 DAO-ACK 基础对象的格式

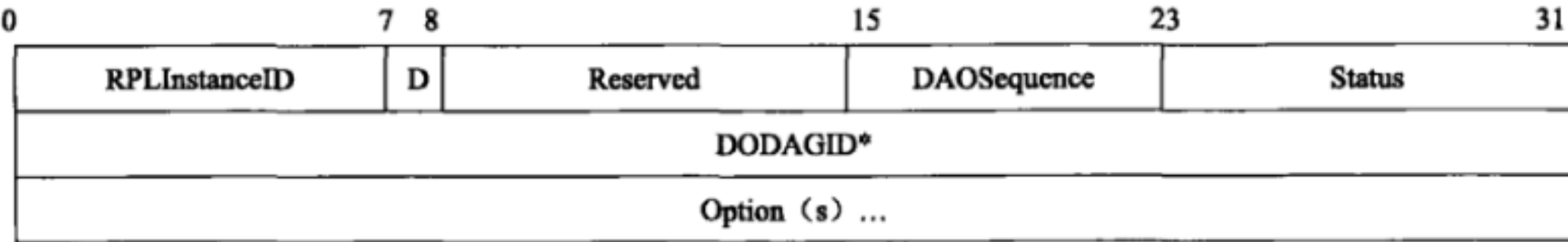


图 17 DAO-ACK 基本对象

RPL实例标识: 8位字段表明与DODAG关联的拓扑实例, 从DIO消息中学习。

D: ‘D’标志位表明DODAGID是存在的。当一个本地RPL实例标识被使用时该标志位必须被设置。

Reserved: 7位未使用字段。字段必须被发送方初始化为0, 并且被接收方忽略。

DAOSequence: 在节点中每条唯一的DAO消息中递增并在DAO-ACK消息中重复。DAOSequence用来关联一条DAO消息和一条DAO-ACK消息, 不会和与给定目标关联的传输信息选项路径序号混淆。

Status: 表明是否完成。状态0为不合格的接收, 1到127是未分配和未确定的, 128往上是拒绝代码, 用来表明节点应该选择一个交替的父节点。

DODAGID (可选): 由DODAG根设置的128位无符号整型, 用来唯一标识一个DODAG。只有当 ‘D’标志位设置时该字段才存在。本字段只在当一个本地RPL实例标识使用时才存在, 用来标识出与RPL实例标识关联的DODAGID。当使用一个全局的RPL实例标识时本字段没有必要存在。

DAO-ACK基础中未分配的位用来保留。它们必须被发送方初始化为0, 并且被接收方忽略。

7.5.2 安全 DAO-ACK

一条安全DAO-ACK消息遵循图7的格式, 其中DAO-ACK消息中的基础格式如图17所示。

7.5.3 DAO-ACK 选项

本标准不定义任何DAO-ACK消息携带的选项。

7.6 一致性检验

CC消息用来检验消息计数和发布质疑/响应。CC消息必须以一条安全RPL消息的方式发送。

7.6.1 CC 基础对象的格式

CC 基础对象的格式如图 18 所示。

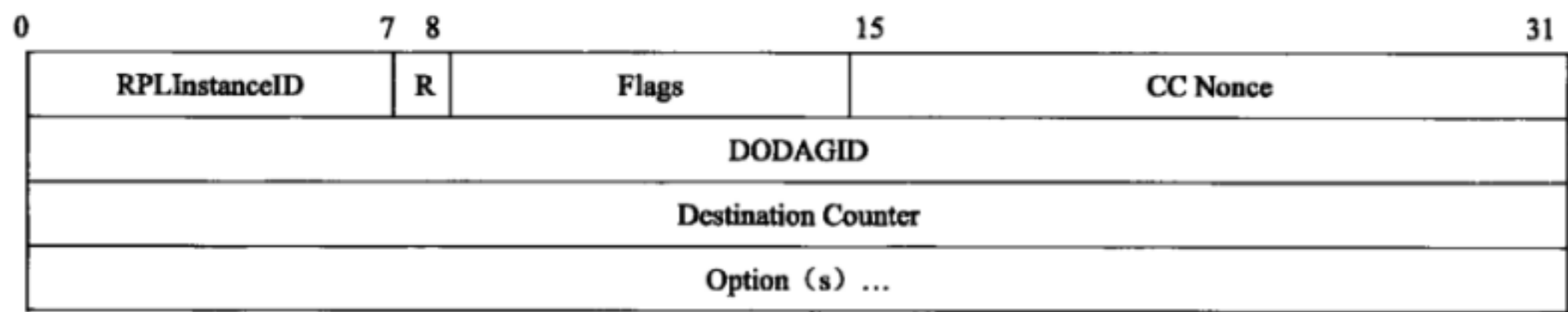


图 18 CC 基本对象

RPL实例标识: 8位字段表明与DODAG关联的拓扑实例, 从DIO消息中学习。

R: ‘R’标志位表明CC消息是否是一个响应。‘R’标志位被清除则是一个请求; ‘R’标志位被设置则是一个响应。

Flags: 7位未使用字段。字段必须被发送方初始化为0, 并且被接收方忽略。

CC Nonce: 由CC请求设置的16位无符号整型。相应的CC响应具有与请求相同的CC nonce。

Destination Counter: 32位无符号整型值表明发送者对目的地当前安全计数器值的估计。如果发送方没有一个估计值, 则应该设置目的地计数器字段为0。

CC基础中未分配的位用来保留。它们必须被发送方初始化为0, 并且被接收方忽略。

目的地计数器值允许新的或者已修复的节点通过CC消息交互重新同步。确定一个计数器的值对于一个给定安全密钥不是重复使用的是非常重要的事情, 即使是在设备从故障中恢复事件创建了一个计数器状态损失。例如, 当一条CC请求或者其他消息安全部分中带有初始化的计数器的RPL消息, 规定CC相应消息中的传入计数器允许请求节点重置其传出计数器的值, 该值比接收节点收到的最后一个值要大; 传入计数器将会同样从接收到的CC响应中更新。

7.6.2 CC 选项

本标准考虑CC消息携带下面选项：

- 0x00 Pad1；
- 0x01 PadN。

7.7 RPL 控制消息选项

7.7.1 RPL 控制消息选项生成格式

RPL控制消息选项遵循图19的格式。

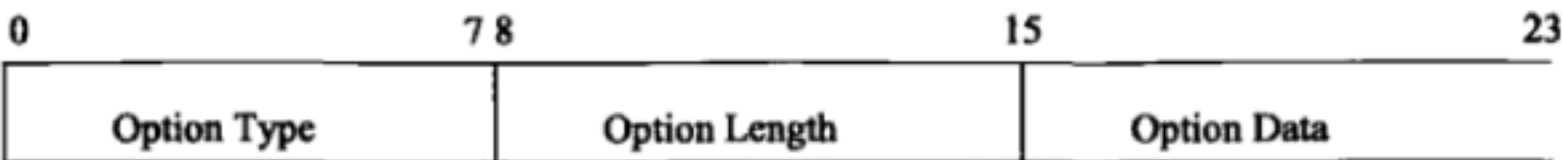


图 19 RPL 选项生成格式

Option Type: 8位类型选项的标识符。

Option Length: 8位无符号整数，代表选项的字节长度，不包括选项类型和长度字段。

Option Data: 包含特定于选项的数据的可变长度字段。

当所处理的RPL消息所包含的选项类型值不被接收方识别时，接收方必须默默的忽略未识别选项并继续处理下面的选项，正确的处理消息中任何剩余的选项。

RPL消息选项可能有对齐请求。按照IPv6惯例，带有对齐要求的选项，在数据包中满足每个选项的选项数据字段的自然边界多字节值处对齐。

7.7.2 Pad1

Pad1选项可能存在于DIS、DIO、DAO、DAO-ACK和CC消息中，格式如图20所示。

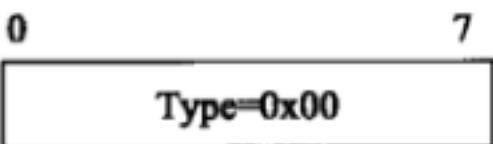


图 20 Pad 1 选项格式

Pad1选项是用来在消息中插入一个单字节以确保选项对齐。如果需要多于一个字节的padding，PadN选项应该被使用而不是使用多个Pad1选项。

注：Pad1选项的格式是特殊情况，它没有选项长度和选项数据字段。

7.7.3 PadN

PadN选项可能存在于DIS、DIO、DAO、DAO-ACK和CC消息中，格式如图21所示。

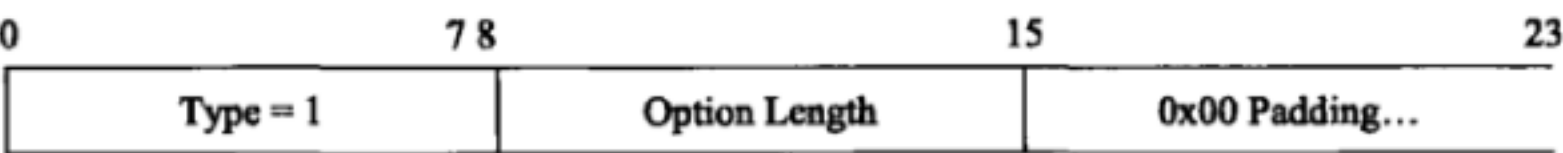


图 21 Pad N Option 的格式

Pad N选项是用来在消息中插入二个或多个字节以确保选项对齐。Pad N选项数据必须被接收方忽略。

Option Type: 0x01。

Option Length: padding的N字节， $2 \leq N \leq 7$ ，选项长度字段包含值N-2。选项长度0表明总共2字节的padding。选项长度5表明总共7字节的padding，且是Pad N选项允许的最大padding大小。

Option Data: 对于N (N>1) 字节的padding，选项数据包含了N-2字节的0值。

7.7.4 度量约束

度量约束选项可能存在于DIO或者DAO消息中，格式如图22所示。

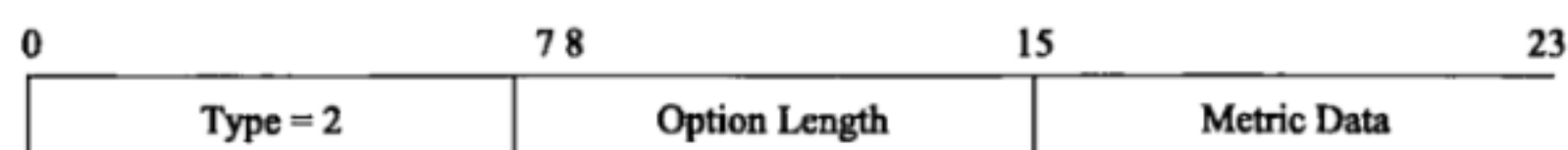


图 22 度量约束选项格式

度量约束用来报告DODAG中的度量。度量约束可能包含许多的离散节点，链接，和聚合路径度量和约束，这些由实施者选择。

度量约束可能同一条RPL控制消息中出现不止一次，例如调解一个度量数据长于256字节的使用用例。

注：更多信息见IETF RFC6551。

处理和传播度量约束是由实现特定策略函数来管理的。

Option Type: 0x02。

Option Length: 选项长度字段包含度量数据的字节长度。

Metric Data: 度量约束的顺序，内容和编码在IETF RFC6551中规定。

7.7.5 路由信息

路由信息选项可能存在于DIO消息中，并且携带和由IETF RFC4191定义的IPv6邻居发现路由信息选项相同的信息。DODAG的根节点有权来设置信息，并且该信息沿着DODAG的传播是不变的。一个RPL路由器可能通过传递ND选项以便在其自己的RAs中通告路由信息，所以一个附着于RPL路由器的节点将会结束使用DODAG，因为该DODAG的根拥有对于数据包目的地最佳路径。除了现存的ND语义，目标函数使用该信息来选择一个拥有到数据包目的地最佳路径的DODAG。选项的格式如图23所示。

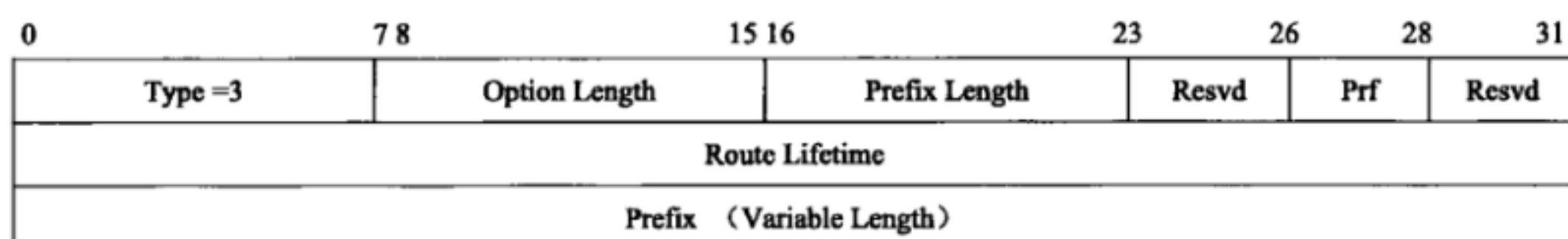


图 23 路由信息选项格式

路由信息选项用来表明从DODAG根到指定目的地前缀的连通性。

当一条RPL控制消息需要指定连通性多于一个目的地时，路由信息选项可能会重复。字段描述为方便起见抄入如下：

Option Type: 0x03。

Option Length: 变量，选项的字节长度不包括类型和长度字段。注意本长度用单字节单元来表述，不同于IPv6 ND。

Prefix Length: 8位无符号整数。前缀中主导位的数量是有效的。值的范围是0到128.前缀字段的字节数是从选项长度字段推断而来，必须至少是前缀长度。注意在RPL中这意味着前缀字段可能的长度不同于0,8或16。

Prf: 2位有符号整数。路由优先权表明当已经收到多个完全相同的前缀时，是否优先选择与此前缀关联的路由器而不是其他路由器，（对于不同的路由器）。如果接收到保留值（10），路由信息选项必须被忽略。依照IETF RFC4191的规定保留值（10）不能被发送。

Resvd: 2个3位的非使用字段。它们必须被发送方初始化为0，并且被接收方忽略。

Route Lifetime: 32位无符号整数。以秒为单位的时间长度（相对于发送数据包的时间）前缀对于路由检测是有效的。值0xffffffff代表无限大。

Prefix: 可变长度字段, 包含一个IP地址或者IPv6地址的前缀。前缀长度字段包含了前缀中有效主导位的数量。前缀长度之后的位被保留且必须被发送方初始化为0, 并且被接收方忽略。

注: RPL中本字段可能的长度不同于0、8或16。

路由信息选项中未分配的位是保留的。它们必须被发送方初始化为0, 并且被接收方忽略。

7.7.6 DODAG 配置

DODAG配置选项可能存在于DIO消息中, 格式如图24。

0	7 8	15 16	19 21	23	31
Type =4	Option Length = 14	Flags	A	PCS	DIOIntDoubl
DIOIntMin	DIORedun	MaxRankIncrease			
Reserved	Def.Lifetime	Lifetime Unit			

图 24 DODAG 配置选项格式

DODAG配置选项用来描述通过DODAG的DODAG操作的配置信息。

选项中用于沟通的信息普遍是静态和不可改变的, 因此没有必要在每一条DIO中都包括。该信息配置在DODAG的根节点并且通过DODAG伴随DODAG配置选项分发出去。DODAG根节点之外的节点在传播DODAG配置选项的时候一定不能更改信息。该选项可能偶尔地被包含在DODAG根内(由DODAG根决定), 一定会被包含在一条单播请求的回复中, 例如一条单播DODAG信息请求消息(DIS)。

Option Type: 0x04 (在IANA中确定)。

Option Length: 14。

Flags: 4位非使用保留字段。字段必须被发送方初始化为0, 并且被接收方忽略。

Authentication Enabled (A): 一位的标志位, 描述网络的安全模式。该位描述一个节点在它以路由器的身份加入网络之前是否必须使用密钥权利鉴权。如果是非安全的DIO, 'A'位必须是0。

Path Control Size (PCS): 3位无符号整数, 用来配置可能分配给路径控制字段的位数量。注意当PCS被咨询用来决定路径控制字段的宽度, 一个为1的值被添加时, 即一个PCS值为0导致了路径控制字段一个活动位。PCS默认值为DEFAULT_PATH_CONTROL_SIZE。

DIOIntervalDoublings: 8位无符号整数, 用来配置DIO trickle定时器的Imax。DIOIntervalDoublings的默认值是DEFAULT_DIO_INTERVAL_DOUBLINGS。

DIOIntervalMin: 8位无符号整数, 用来配置DIO trickle定时器的Imin。DIOIntervalMin的默认值为DEFAULT_DIO_INTERVAL_MIN。

DIORedundancyConstant: 8位无符号整数, 用来配置DIO trickle定时器的k。默认值为DEFAULT_DIO_REDUNDANCY_CONSTANT。

MaxRankIncrease: 16位无符号整数, 用来配置DAGMaxRankIncrease, 允许的本地修复所支持的rank增量。如果值为0那么该机制为禁用的。

MinHopRankInc: 16位无符号整数, 用来配置4.4.1节中规定的MinHopRankIncrease。默认值为DEFAULT_MIN_HOP_RANK_INCREASE。

Default Lifetime: 8位无符号整数。所有RPL路由的默认生存周期。它用生存周期单位表达, 例如以秒为单位的默认生存周期是(Default Lifetime) * (Lifetime Unit)。

Lifetime Unit: 16位无符号整数。提供RPL中以秒为单位用来表达路由生存周期的单元。对于稳定的网络, 可能是小时甚至是天为单位。

Objective Code Point (OCP): 16位无符号整数。OCP字段确定OF并由IANA管理。

7.7.7 RPL 目标

RPL目标选项可能存在于DAO消息中，格式如图25所示。

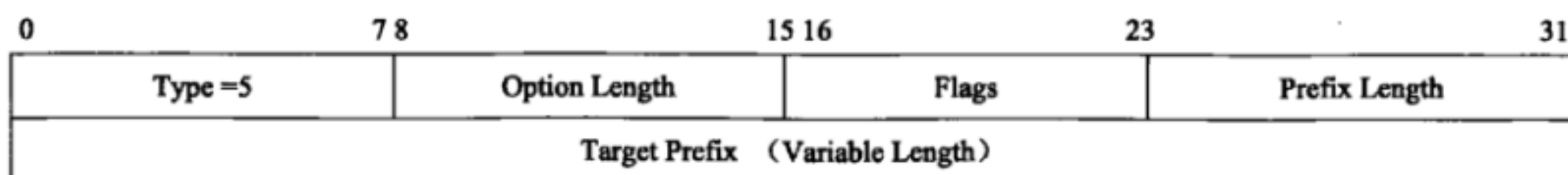


图 25 RPL 目标选项格式

RPL目标选项用来表明一个目标IPv6地址，前缀，或者沿着DODAG的可到达的或者询问的多播组。DAO消息中，RPL目标选项表明可达性。

一个RPL目标选项可以选择与RPL目标描述符选项配对，以使取得到达目标资格。

在一条DAO消息中，一个或多个传输信息选项集合可能直接跟随一个或多个目标选项集合。

如果有必要的话RPL目标选项可能会被重复使用来表明多个目标。

Option Type: 0x05（在IANA中确定）。

Option Length: 变量，选项的字节长度但不包括类型和长度字段。

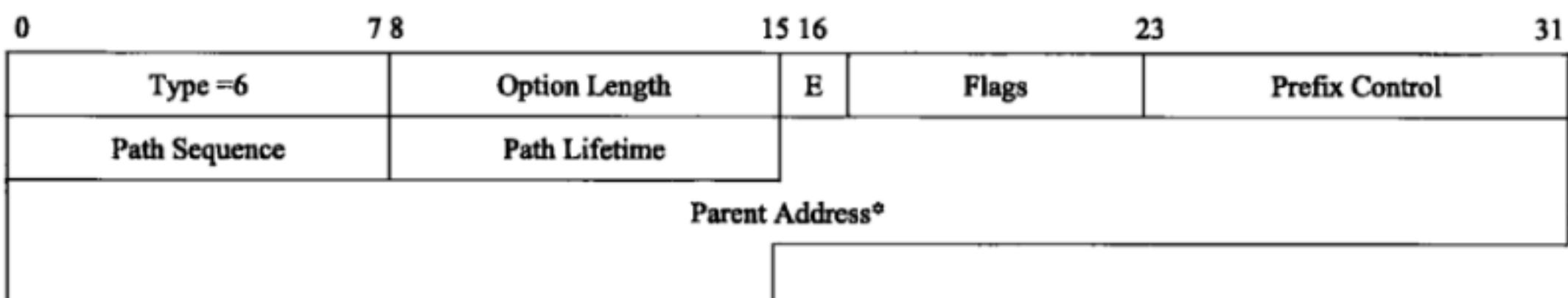
Flags: 8位非使用保留字段。字段必须被发送方初始化为0，并且被接收方忽略。

Prefix Length: 8位无符号整数。IPv6前缀中有效主导位的数量。

Target Prefix: 可变长字段识别一个IPv6目的地地址，前缀，或者多播组。前缀长度字段包含了前缀的有效主导位的数量。前缀长度之后的前缀位是保留的且必须被发送方初始化为0，并且被接收方忽略。

7.7.8 传输信息

传输信息选项可能存在于DAO消息中，格式如图26所示。



‘*’ 表示父节点地址选项不是经常存在的

图 26 传输信息选项格式

传输信息选项是节点用来表明到一个或多个目的地路径的属性。目的地由传输信息选项之前的一个或多个目标选项表明。

传输信息选项能够被一个节点用来表明它的DODAG父节点到祖先节点所收集DODAG路由信息，典型目的是构造源路由。在非存储模式中该祖先可能是DODAG的根，并且这个选项由DAO消息携带。在存储模式中父节点地址是不需要的，因为DAO消息是直接发送给父节点的。选项长度用来决定父节点地址是否存在。

对于拥有多个DAO父节点的非存储节点，每一个DAO父节点都包括一个传输信息选项作为非存储目的地通告。节点可能把路径控制字段的位分配到不同的DAO父节点组中，以便标记父节点的优先级。构建向下路由时，优先级用于从交替的父节点/路径中选择DODAG根节点。

一个或多个传输信息选项必须在前面加上一个或多个RPL目标选项。如此RPL目标选项表明子节点，传输信息选项枚举DODAG父节点。

一个典型的非存储模式节点将会使用多个传输信息选项，它将会发送DAO消息从而直接形成根。一个典型的存储节点将会使用一个不带父节点选项的传输信息选项，将会发送DAO消息从而形成，随着而外的路径控制调整稍后在说，到一个或多个父节点。

例如，非存储模式中用T表示一个目标选项。用P表示一个传输信息选项且包含一个父节点地址P。考虑这种情况，一个非存储节点N通告自身拥有的目标N1和N2以及拥有父节点P1，P2和P3。那种情况下DAO消息将会被期望包含序列((N1)，(N2))，((P1)，(P2)，(P3))，如此目标选项组{N1，N2}由传输信息选项描述同时拥有父节点{P1，P2，P3}。非存储节点将会寻址DAO消息直接到DODAG的根，并且通过DODAG父节点P1，P2，或者P3中的一个转发DAO消息。

Option Type: 0x06（将在IANA中确定）。

Option Length: 变量，取决于父节点地址是否存在。

External (E): 1位标志位。‘E’标志位被设置用来表明父节点路由器重新分配额外的目标到RPL网络中去。一个外部的目标是指通过交替协议已经学习到的目标。外部目标在传输信息选项之前的目标选项中列出。一个外部目标是不被期望支持RPL消息和选项的。

Flags: 7位非使用保留字段。它们必须被发送方初始化为0，并且被接收方忽略。

Path Control: 8位位字段。路径控制字段限制DAO父节点的数量。一条DAO消息通告连通性到一个指定的目的地可能发送的，连同提供一些相关参数的象征。限制提供了低功耗有损网络中输出端所有DAO消息的一些边界。路径控制位的分配和顺序同样服务于通讯优先级。并不是所有的这些位都是依照DODAG配置中的PCS来激活的。路径控制字段被分割成2位的4个子字段：PC1，PC2，PC3和PC4，如图27所示。子字段由优先权排序，PC1优先权最高，PC4最低，如图27所示。子字段内部没有优先权顺序。通过分组父节点和排序父节点，父节点可能与路径控制字段指定的位关联在某种程度上通讯优先权。

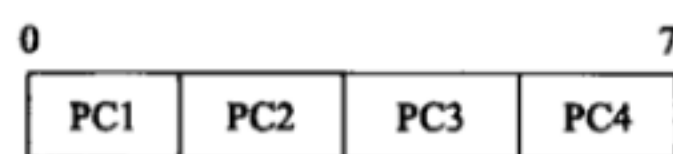


图 27 路径控制参数子字段格式

Path Sequence: 8位无符号整数。当一个RPL目标选项由一个属于目标前缀的节点发行时，节点设置路径序号并且每一次通过更新信息质疑一个RPL目标选项的时候都会增加路径序号。

Path Lifetime: 8位无符号整数。以单元寿命为单位的时间的长度，其前缀对于路由探测是有效的。当一个新的路径序列被发现时周期开始。值0xFF代表无限大。值0x00不可到达。一条DAO消息包含的传输信息选项带有的Path Lifetime值为0x00对于一个目标在本文中被看做是No-path。

Parent Address:（可选）发行传输信息选项的源节点的DODAG父节点的IPv6地址。该字段可能不存在，依照DODAG的操作模式并由传输信息选项长度表明。

传输信息选项的未分配位是保留的。它们必须被发送方初始化为0，并且被接收方忽略。

7.7.9 请求信息

请求信息选项可能存在于DIS消息中，格式如图28所示。

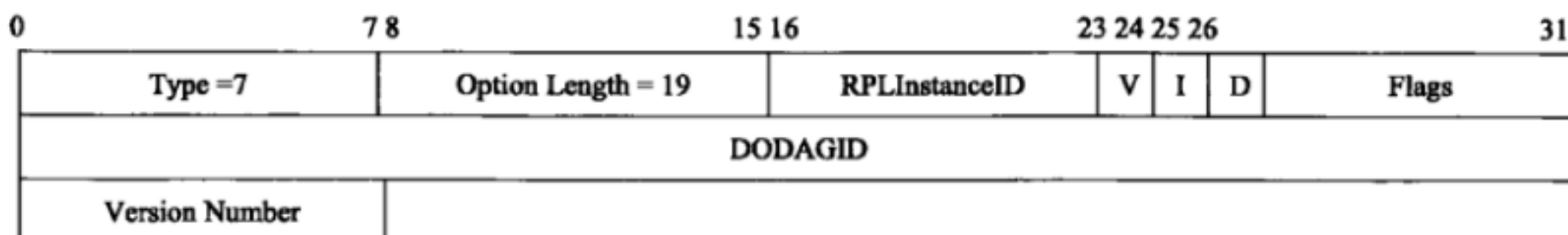


图 28 请求信息选项格式

请求信息选项是节点用来从子网的一个邻居节点请求一条DIO消息用的。请求信息选项可能会制定一些术语的条件来与一个接受节点匹配。这由请求者使用来限制从无兴趣节点应答的数量。这些判断影响一个节点是否重置它的细流计时器。

请求信息选项包含标志位，用来判断一个节点是否应该发起检验以决定是否重置它的细流定时器。当所有的判断为真时节点重置它的细流定时器。如果标志位被设置，那么RPL节点必须检验相关的谓语。如果标志位被清除，那么RPL节点不必检验。

Option Type: 0x07

Option Length: 19

V: 为版本判断标志位。如果接收方的DODAGVersionNumber匹配请求的版本号，那么该位为真。如果被清除那么版本字段不是有效的，并且版本字段必须在发送方设置为0在接收方忽略。

I: 为InstanceID判断标志位。如果RPL节点当前的RPL实例标识匹配请求者得RPL实例标识那么InstanceID判断标志位为真。如果被清除那么RPL实例标识字段不是有效的，并且必须在发送方设置为0在接收方忽略。

D: 为DODAGID判断标志位。如果RPL节点的父集与DODAGID字段设置相同的DODAGID那么DODAGID判断标志位为真。如果被清除那么DODAGID字段不是有效的并且必须在发送方设置为0在接收方忽略。

Flags: 5位非使用保留字段。字段必须由发送方初始化为0由接收方忽略。

Version Number: 8位无符号整数，包含DODAGVersionNumber值当正在请求有效时。

RPL实例标识: 8位无符号整数，包含DODAGVersionNumber值当正在请求有效时。

DODAGID: 128位无符号整数，包含DODAGVersionNumber值当正在请求有效时。

请求信息选项的未分配位用来保留。它们必须在发送方初始化为0在接收方忽略。

7.7.10 前缀信息

前缀信息选项可能存在于DIO消息中，并携带RPL节点和IPv6主机所使用的IPv6 ND前缀信息选项中的信息。尤其是，一个RPL节点可以使用这个选项从一个父节点通告的前缀中获得State-Less Address Auto-Configuration，并通告它自己的地址。DODAG的根节点有权设置该信息。该信息向下无变化的沿着DODAG传播，例外是一个RPL路由器可能通过附加它自己的后缀来更新前缀。该选项的格式是修改的（类型，长度，前缀）以便携带一个RPL选项如图29所示。

0	7 8	15 16	23 24 25 26	31
Type =8	Option Length = 30	Prefix Length	L A R	Reserved1
Valid Lifetime				
Preferred Lifetime				
Reserved2				
Prefix				

图 29 前缀信息选项格式

前缀信息选项用来分配DODAG内部使用的前缀，例如地址自动配置。字段描述如下：

Option Type: 0x08（将在IANA中确定）。

Option Length: 30.注意该长度以单字节单元表示，不像IPv6 ND。

Prefix Length: 8位无符号整数。前缀中主导位的有效数量。值的范围是0到128。前缀长度字段提供在线决策的必要信息。它同样协助如同在IETF RFC4862中规定的地址自动配置，在前缀长度可能存在更多的限制。

L: 1位在线标志位。当设置时，表明该前缀能够用来做在线决策。当不设置时通告不做有关在线和下线的前缀属性声明。换句话说，如果L标志位没有被设置，一个RPL节点一定不能推断出那个地址来源于下线的前缀。就是说，它不能更新一个地址是在线的象征。一个担当路由器的RPL节点不能传播一条L标志位被设置的PIO消息。一个担当路由器的RPL节点可以传播一条L标志位被设置的PIO消息。

A: 1位的自动地址配置标志位。当设置时表明该前缀能够被用做无状态地址配置，见IETF RFC4862中的规定。当ND RAs和RPL DIOs协议都用来在同一条链接上携带PIO时，使用SLAAC中的任何一个都是有可能的。同样的通过强行设置PIO消息A标志位为0令SLAAC中任何协议不合格也是可能的。

R: 1位的路由地址标志位。当设置时，表明前缀字段包含了一个分配给发送方路由器的完整的IPv6地址，该地址能够作为目标选项的父节点使用。所声明的前缀是前缀字段的第一个前缀长度位。路由器IPv6地址拥有同样的范围并且如同通告前缀一样依照同样的声明周期值。前缀字段的使用与通告前缀自身相配合，因为前缀通告仅使用主导位。该标志位的解释因此不依赖于在线的处理请求和自动地址配置标志位。

Reserved1: 5位非使用字段。它必须由发送方初始化为0而在接收方忽略掉。

Valid Lifetime: 32位无符号整数。以秒为单位的在线决策的前缀的时间的长度。Oxffffffff代表无限。**Valid Lifetime**同样在IETF RFC4862中规定。

Preferred Lifetime: 32位无符号整数。以秒为单位的从前缀经过无状态地址自动配置生成的首选地址的时间长度。

注：该字段的值一定不能超出有效周期字段以避免首选地址变成无效的。

Reserved2: 未使用字段。它必须被发送方初始化为0且被接收方忽略。

Prefix: 一个IPv6地址或者IPv6地址的前缀。前缀长度字段包含前缀中有效主导位的数量。紧跟着前缀长度之后的前缀位是保留的，且必须被发送方初始化为0被接收方忽略。一个路由器不应该发送一个本地链路前缀的前缀选项，一个主机应该忽略这个前缀选项。一个非存储节点应该直到拥有一个该前缀的地址才通告该前缀，然后它应该通告这个字段的全地址，且‘R’标志位被设置。一个通告全地址且‘R’标志位被设置的节点的子节点可能使用那个地址来决定传输信息选项中父节点地址字段的内容。

前缀信息选项的未分配位用来保留。它们必须在发送方初始化为0在接收方忽略。

7.7.11 RPL 目标描述符

RPL目标选项后面紧跟着一个不透明特定目标的描述符，如图30所示。

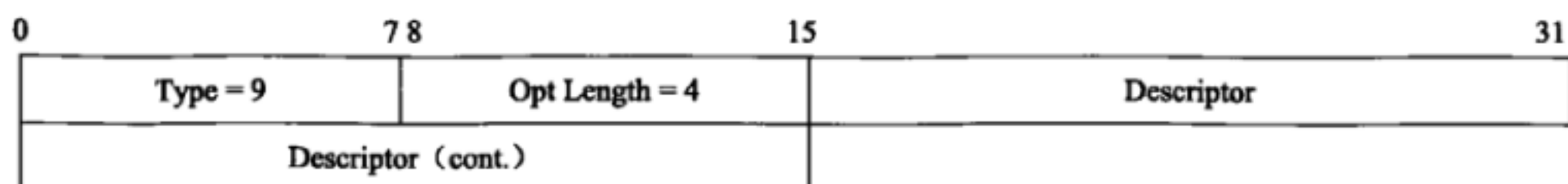


图 30 RPL 目标描述符选项格式

RPL目标描述符选项用来限制一个目标，有时候被称为标记。

每一个目标至多有一个描述符。描述符由节点设置并注射到RPL网路中的目标去。它必须被复制但是不能被路由器修改，并且在DAO消息中向上沿着DODAG传输到目标中。

Option Type: 0x09。

Option Length: 4。

Descriptor: 32位无符号整数, 不透明。

8 序列计数器

描述RPL中序列计数器的引导和操作的总体方案, 例如DIO消息中的DODAGVersionNumber、DAO消息中的DAOSequence和传输信息选项中的路径序列。

8.1 序列计数器概述

使用三种不同的序列号来验证协议信息是否最新和同步性:

DODAGVersionNumber: 该序列计数器在DIO消息中用来确定DODAG形成的版本。为了确保完整性和允许全局修复的发生, 每一次当根节点决定生成一个DODAG的新版本时, DODAGVersionNumber就单调递增。DODAGVersionNumber向下传播到DODAG直到路由加入的新的DODAG版本时才生改变。DODAGVersionNumber在DODAG中是全局有效地, 它表明了一个路由器运行中的DODAG版本。一个过期的(小的)值表明始发的路由器没有更新到新的DODAG版本, 所以一旦接收节点更新到新的DODAG版本时, 该节点不能作为父节点使用。

DAOSequence: 该序列计数器在DAO消息中用来关联一个DAO消息和一个DAO应答消息。DAOSequence数对于节点来说具有局部的意义, 它表明一个DAO消息用来探测DAO消息丢包和允许重传时自身的消耗。

Path Sequence: 该序列计数器存在于DAO消息中的传输信息选项里。该计数器的作用是用来从对于同一目标并行存在多条路由的路由冗余场景中, 区分出一条新的路由取代旧路由的运动。Path Sequence具有全局的意义, 在DAO消息中用来表明关联目标路由的新鲜性。从源路由器接受到的一个旧的(小的)值, 标明源路由器拥有的是陈旧的路由所以源路由器不能再被认为是目标的可能的下一条节点。Path Sequence由节点计算得出并通告给目标节点, 可能是目标本身或者是一个路由器通告目标主机来取代主机, 直到DAO消息被父节点路由器传播给根节点之前内容都不会改变。如果主机没有通过一个计数到它的路由器, 那么路由器负责计算代表主机的Path Sequence, 该主机只能注册一个为此目的的路由器。如果以路由冗余为目的, 在特定的时间点向多个父节点发送包含同一目标节点的DAO消息, 那么在所有相同目标节点的所有DAO消息中Path Sequence都是相同的。

8.2 序列计数器运算

RPL序列计数器以‘lollipop’样式细化([Perlman83]), 值大于等于128被用做线性序列标明重新开始计数和引导程序, 值小于等于127被用作循环序列使用, 空间大小为128。所关心的问题是线性区域向循环区域过度时运算模式。最后, 当在循环区域里面运算时, 如果序列数被检测到相差甚远, 它们不是可比较的, 如下面所描述。

一个窗口的比较值, $SEQUENCE_WINDOW = 16$, 基于 $2 \sim N$ 的值来形成, 在本标准中 N 的值为4。

一个给定的序列计数器,

a) 该序列计数器应该被初始化为 128 或者是更大值才能被使用。推荐值为 240 ($256 - SEQUENCE_WINDOW$);

b) 当一个序列计数器的值增加将导致其超出最大值的时候, 则必须重新变成 0。当递增序列计数器 ≥ 128 时, 最大值为 255。当递增序列计数器 < 128 时, 最大值为 127;

c) 当比较两个序列计数器时, 必须遵守以下规则:

1) 第一序列计数器 A 在区间 $[128 \cdots 255]$, 第二个序列计数器在区间 $[0 \cdots 127]$:

— 如果 $(256+B-A) \leq \text{SEQUENCE_WINDOW}$, 那么 $B > A$, $A < B$, 2 个值不相等。

— 如果 $(256+B-A) > \text{SEQUENCE_WINDOW}$, 那么 $A > B$, $B < A$, 2 个值不相等。

例如, $A=240$, $B=5$, $(256+B-A)$ 值为 21, $21 > \text{SEQUENCE_WINDOW}$ (16), 所以 $240 > 5$ 。另一个例子, 如果 $A=250$, $B=5$, 那么 $(256+B-A)$ 值为 11, $11 > \text{SEQUENCE_WINDOW}$ (16), 所以 $250 < 5$ 。

2) 一种情况两个计数器都 ≤ 127 , 另一种情况两个计数器都 ≥ 128 :

— 如果两个序列计数器绝对量值的差异小于或者等于 SEQUENCE_WINDOW , 那么在 IETF RFC1982 中规定了两者的关系, 大于、小于或者等于。

— 如果两个序列计数器绝对量值的差异大于 SEQUENCE_WINDOW , 那么将会发生同步化, 两个序列数值将不可比较。

d) 如果两个序列号被确定为没有可比性, 即比较的结果没有定义, 那么一个节点应该考虑对最近被观察到发生增量变化的序列号给予优先考虑的方式来评价两者。如果做不到这一点, 节点应该考虑这样一种比较方式, 以最小改变自己状态的结果来评价两者。

9 向上路由

描述RPL如何发现和维持向上路由。描述了DODAG信息对象(DIO)消息的用途, 发现和维持这些路由的消息。定义了RPL如何生成和回应DIO消息。还描述了DODAG信息请求(DIS)消息, 用来触发DIO传输。

节点如果决定加入一个DODAG必须准备至少一个DODAG父节点做为关联实例的一个默认路由。该默认路由向上转发一个数据包直到最终到达一个共同的祖先, 然后通过向下路由到目的地。如果目的地不在DODAG中, 那么DODAG根节点也许可以利用连通性转发数据包到DODAG外; 如果它不能转发数据包到DODAG外, 那么DODAG根节点必须丢弃它。

DIO消息同样能够传输明确的路由信息:

DODAGID DODAGID是一个全局或者本地唯一的根节点ipv6地址。一个节点加入DODAG应该经过DODAG父节点提供一条到根使用的地址的主机路由, 作为DODAGID。

RIO Prefix 根节点可能会在一个DIO消息中放置一个或者多个路由信息选项。RIO用来通告一条额外的路由, 该路由经由根节点且与优先权相关, 最终和由IETF RFC4191规定的RIO消息合并。一个目标函数可能会使用RIO中的路由通告或者这些路由的优先权从而选出同一实例的最好的DODAG。

9.1 DIO 基本规则

a) 对于下面的 DIO 相应的字段, 一个节点如果不是 DODAG 相应的根则必须与其首选 DODAG 的父节点通告相同的值。这样这些值将会沿着 DODAG 不做更改的传播并且被每一个拥有连接到该 DODAG 根的路由的节点通告。这些字段是:

- Grounded (G)
- Mode of Operation (MOP)
- DAGPreference (Prf)
- Version
- RPLInstanceID

— DODAGID

b) 节点必须在每一条更新下面的相应字段:

— Rank

— DTSN

c) 每个根集合中的 DODAGID 字段在 RPL 实例中必须唯一并且必须为根的可路由的 IPV6 地址。

9.2 向上路由发现和维护

向上路由发现允许节点通过发现感兴趣的 DODAG 成员邻居节点来加入 DODAG 并且最终标识一个父节点集合。选择邻居和父节点确切的执行策略是由 OF 来依赖实现和驱动的。

9.2.1 DODAG 版本内的邻居和父节点

RPL 的向上路由发现算法和处理是三种本地链路节点的逻辑集合。第一, 候选的邻居集是一些可以通过本地链路组播到达的节点子集。集合的选择依赖于执行和 OF。第二, 父节点集合是候选邻居集合的受限子集。最后, 首选的父节点是向上路由中首选下一条的父节点集合的成员。首选的父节点概念是一个单独节点; 也可能是一些优先权相同并且具有相同序列的多个父节点的集合。

更多细化:

a) DODAG 父节点集合必须是一个候选邻居集合的子集;

b) DODAG 根必须有一个大小为 0 的 DODAG 父节点集;

c) 一个节点如果不是 DODAG 的根可能会维持一个大小大于或等于根的父节点集;

d) 一个节点的首选 DODAG 父节点必须是它的 DODAG 父节点集的一个成员;

e) 节点的序列必须大于它的 DODAG 父节点集中所有节点序列;

f) 当邻居不可到达探测机制 (NUD), 或者一个等效机制, 确定一个邻居节点为不可到达时, 当计算和路由通告时 RPL 节点不再考虑该节点作为候选邻居集, 直到其被再次确定为可以到达。通过一个不可到达的节点的路由必须从路由表中移除。

这些规则确保了 DODAG 内节点的一致有序性。只要这些节点的序列不变, 遵循以上规则将会确保每个节点到一个 DODAG 根的路由是 loopfree 的, 向根的每一条序列都会递减的。

OF 可以用来候选邻居集和父节点集的选择, 见 IETF RFC 6552。

9.2.2 跨 DODAG 版本的邻居和父节点

上面的规则管理单一的一个 DODAG 版本。本节的规则定义了当存在多个 DODAG 版本时 RPL 是怎样操作的。

9.2.2.1 DODAG 版本

a) 组元 (RPLInstanceID, DODAGID, DODAGVersionNumber) 唯一的定义了一个 DODAG 版本号。一个节点的 DODAG 父节点集中的每一个成员, 如同被最后听到的从每个 DODAG 父节点消息的传输一样, 必须属于同一 DODAG 版本;

b) 如果一个节点的 DODAG 父节点的每一个成员都属于该 DODAG 版本, 或者该节点是相应的 DODAG 的根, 那么该节点是该 DODAG 版本的一个成员;

c) 节点不能向他不属于的 DODAG 版本发送 DIOs 消息;

d) DODAG 的根可能会增加它们所通告的 DODAGVersionNumber, 从而移动到一个新的 DODAG 版本;

e) 在一个给定的 DODAG 内, 一个非根节点所通告的 DODAGVersionNumber 一定不能比他所听到的最大的 DODAGVersionNumber 的值还要高;

f) 一旦一个节点已经通过发送 DIO 消息通告了 DODAG 版本, 那么它必将不再是同一个 DODAG 中老的 DODAG 版本中的一个成员 (即同样的 RPL 实例标识, 同样的 DODAGID, 但是比较小的 DODAGVersionNumber)。

当 DODAG 父节点集在一个非根节点上变为空的时候 (即最后的父节点已经移走, 导致节点不再与该 DODAG 相关联), DODAG 信息应当不在被压抑直到其特定的本地执行时间终止, 用来观察 DODAGVersionNumber 是否已经增加, 是否需要新的 DODAG 父节点集。这将有助于预防环路的产生, 如果节点非故意的在其先前的 DODAG 子集中重新加入了一个旧的 DODAG 版本, 则回路会发生。

随着 DODAGVersionNumber 的增加, 新的 DODAG 版本从 DODAG 根向外传播。一个通告新的 DODAGVersionNumber 的父节点不能从属于该节点通告旧的 DODAGVersionNumber 的 DODAG 子集。因此一个节点能安全的添加任何一个序列带有一个新 DODAGVersionNumber 的父节点而不会形成回路。

例如, 假设一个节点以 DODAGVersionNumber 值为 N 离开一个 DODAG。假设该节点有一个 DODAG 子集, 并且试图通过通告一个 INFINITE_RANK 的序列来妨碍 DODAG 子集, 但是这些通告或许已经在低功耗有损网络中丢失。如果节点确实在源 DODAG 中位置为 DODAGVersionNumber N 的地方观察一个候选邻居通告, 这个候选邻居很可能在该节点之前的 DODAG 子集中并且很可能在加入候选邻居作为父节点时产生一个环路。如果那个候选邻居在这个情况下被观察到通告的 DODAGVersionNumber 为 N+1, 那么这个候选邻居确定是安全的, 因为他确定不在源节点的 DODAG 子集中, 当他已经能够通过收听 DODAG 根的消息来增加 DODAGVersionNumber, 于此同时源节点是分离出去的。为此对于分离节点来说记住源节点的 DODAG 信息, 且包含 DODAGVersionNumber N 是非常有用的。

在一个节点转变到和通告一个新的 DODAG 版本, 以上所做的规则保证他一旦决定通告新的 DODAG 版本就不能够再通告以前的 DODAG 版本 (旧的 DODAGVersionNumber)。

9.2.2.2 DODAG 根

- a) 一个没有满足应用定义目标可能性的 DODAG 根必须不能设定接地位;
- b) 一个 DODAG 根必须通告 ROOT_RANK 消息;
- c) 一个父节点集是空的节点可能会成为一个浮动 DODAG 的 DODAG 根。

该节点也可以设置其 DAGPreference 使得他不是首选

在采用非 RPL 链接来联合大量低功耗有损网络根节点的部署中, 运行 RPL 在这些非 RPL 链接并且用一个路由器作为主干根节点是有可能的。这种主干根节点是个虚拟的 DODAG 根节点, 并且显示了主干的 BASE_RANK 的序列。所有的低功耗有损网络根节点都是主干根的父节点, 其中包括如果骨干根也是低功耗有损网络根本身, 显示了低功耗有损网络的 ROOT_RANK 的序列。这些虚拟根是同一 DODAG 内的一部分并且通告同样的 DODAGID。它们协调 DODAGVersionNumbers 和其他的 DODAG 参数在骨干与虚拟根之间。

9.2.2.3 DODAG 选择

目标函数和一组通告路由权值以及 DAG 的约束条件决定了节点怎样选择它的邻居集, 父节点集和首选父节点。这样的选择还含蓄的决定了 DAG 内的 DODAG。这样的选择可能包含了管理优先级及权值或者其他考虑。

如果一个节点可以选择加入一个比较偏爱的 DODAG，同时又有其他的优化目标，那么该节点一般会寻求加入 OF 所决定的比较偏爱的 DODAG。其他条件相同，决定哪些 DODAG 是最优先考虑的要留到执行的时候才会决定（因为，作为一个提醒，一个节点在每个实例中只能加入一个 DODAG 中）。

9.2.2.4 DODAG 版本的序列和移动

- a) 节点在 DODAG 版本中所通告的序列必须不能小于或者等于它的父节点集的任何成员；
- b) 节点在 DODAG 版本中所通告的序列可能会比先前通告中的低；
- c) 设定 L 为一个 DODAG 版本中给定节点所通告的最低序列。在同一 DODAG 版本中该节点所通告的序列一定不能高于 $L + \text{DAGMaxRankIncrease}$ 。INFINITE_RANK 是该规则的一个例外：在 DODAG 版本中节点可以没有约束的通告 INFINITE_RANK。如果一个节点的序列高于所允许的 $L + \text{DAGMaxRankIncrease}$ 值，那么当他通告序列的时候要用 INFINITE_RANK 作为它的序列；
- d) 节点可以在任何时间选择一个在 RPL 实例中的 DODAG 来加入。这样的加入没有序列约束，除非所加入的这个不同的 DODAG 是这个节点之前就曾经加入过的 DODAG 版本，在这种情况下先前的规则 bullet (3) 必须得遵守。直到一个节点传输一个 DIO 消息来声明它的新的 DODAG 资格，他必须沿着之前的 DODAG 来传输数据包；
- e) 节点可以在听到来自合适父节点的下一个 DODAGVersionNumber 通告的任何时候，选择迁移到 DODAG 内的下一个 DODAG 版本。

从概念上来讲，一个执行是维持 DODAG 父集在 DODAG 版本内部。移动需要改变到 DODAG 父集。向上移动不会出现回路的风险但是向下移动可能会，所以操作受到额外约束的支配。

当节点迁移到下一个 DODAG 版本，DODAG 父集需要为新版本重组。迁移的执行可能会推迟一段合理的时间，看看是否有其他更好的衡量标准并且更高的级别的邻居在广播自己。类似的，当一个节点越入到一个新的 DODAG 当中时需要重新为这个 DODAG 构造新的 DODAG 父集。

如果节点需要向下移动到他喜爱的 DODAG 中，增加它的序列，那么还会在他移动之前破坏它的路由并延时如 9.2.2.5 所描述。

一个节点被允许没有任何约束的加入任何其之前没有加入过的 DODAG 版本，但是如果这个节点之前是该 DODAG 版本中的成员那么必须继续遵守规则，不可以在该 DODAG 版本的任何点通告级别高于 $L + \text{DAGMaxRankIncrease}$ 的序列。此规则必须被遵守以免造成漏洞，将允许节点有效的增加它的序列一直到 INFINITE_RANK，这将会影响到其他节点并创建一个资源浪费计数到无穷大的情况。

9.2.2.5 Poisoning

- a) 节点通过通告 INFINITE_RANK 来 Poison 路由；
- b) 节点的父节点集中的任何节点都不能有 INFINITE_RANK 序列。

尽管执行可能会为了 Poisoning 而通告 INFINITE_RANK，但是如此做和设定序列为 INFINITE_RANK 时是不一样的。例如，一个节点可能会继续发送数据包而它的 RPL 数据包信息包含的序列却不是 INFINITE_RANK，但是还会通告 INFINITE_RANK 在它的 DIO 消息中。

当一个（前）父节点被观察到通告 INFINITE_RANK，那么这个（前）父节点被从 DODAG 中分离出来并且不再会被当做父节点，也没有其他任何原因的另一个节点可能被认为序列大于 INFINITE_RANK。因此该（前）父节点不再被当做父节点并从父集中移走。

9.2.2.6 分离

一个节点不能够保持连接到一个给定 DODAG 版本的 DODAG 中去，即不能保留非空的父集而不违反本标准的规定，可能会从该 DODAG 版本中分离出去。分离出去的节点成为他自己浮动 DODAG 的根，并将会在一条 DIO 消息中立即通告这个新的地点作为备用 Poisoning。

9.2.2.7 跟随一个父集

如果一个节点收到了来自一个 DODAG 父节点的一条 DIO 消息，表明该父节点已经离开了这个 DODAG，如果可能的话这个节点应该用一个备用的父节点继续留在当前的 DODAG 中。他也可能会跟随那个离开的父节点。

DODAG 父节点可能移动，迁移到下一个 DODAG 版本，或者直接跳入一个不同的 DODAG 中去。如果可能的话一个节点应该用一个备用的父节点尽可能的保留在当前的 DODAG 中，如果没有其他选项他应该跟随那个离开的父节点。

9.2.3 DIO 消息通信

当接收到一条 DIO 消息时，接收节点必须首先决定这条 DIO 消息是否会被进一步的处理，如果合适的话并随后为进一步处理呈现这条 DIO 消息。

- a) 如果这条 DIO 消息是畸形的，那么这条 DIO 消息是不合适做进一步处理的，节点必须丢掉它；
- b) 如果发送 DIO 消息的节点是候选邻居集中一员，并且这条 DIO 消息不是畸形的，节点必须处理。

9.2.3.1 DIO 消息处理

当从候选邻居中收到 DIO 消息时，候选邻居可能会遵循 9.2 规定的 DODAG 发现规则被提升为 DODAG 父节点。当节点放置一个邻居节点到 DODAG 父集中时，节点通过新的 DODAG 父节点变成对 DODAG 有依赖。

最佳首选的父节点应该用来约束其他节点成为 DODAG 父节点。DODAG 集中的一些节点的序列可能会少于或者等于最佳 DODAG 父节点。

9.3 DIO 传输

RPL 节点用 Trickle 定时器来传输 DIO 消息。发送节点发出的具有较小 DAGRank 的 DIO 消息将不会导致接受节点的父集发生变化，首选父节点或者序列对于 Trickle 定时器应该考虑的一致。

下面的数据包和事件对于 Trickle 定时器的考虑必须不能一致，并且导致 Trickle 定时器重置：

- a) 当节点转发数据包时检测不一致；
- b) 当节点接收了一条没有关联信息选项的多播 DIS 消息时，除非一个 DIS 标志位限制这一行为；
- c) 当节点接收了一条含有关联信息选项的多播 DIS 消息时，并且节点满足所有关联信息选项的判断，除非一个 DIS 标志位限制这一行为；
- d) 当节点加入到一个新的 DODAG 版本（例如，通过更新它的 DODAGVersionNumber，加入一个新的 RPL 实例等）。

节点不应该重置它的 DIO 消息 Trickle 定时器来回应单播的 DIO 消息。当节点收到了一条没有信息关联选项的单播 DIS 消息时，他必须单播的回复一条 DIO 消息给发送者。这条单播 DIO 消息必须包含 DODAG 配置选项。当节点收到了一条带有信息关联选项的单播 DIS 消息并且满足信息关联选项的判断，那么他必须单播的回复一条 DIO 消息给发送者。这条单播 DIO 消息必须包含 DODAG 配置选项。因此节点可能向潜在的 DODAG 父节点传送一条单播 DIS 消息为了探测 DODAG 配置选项和其他参数。

9.3.1 Trickle 参数

Trickle 定时器的配置参数如下面定义:

Imin:从 DIO 消息中学习当做 $(2^{\text{DIOIntervalMin}})$ ms。DIOIntervalMin 的默认值 DEFAULT_DIO_INTERVAL_MIN。

Imax:从 DIO 消息中学习当做 DIOIntervalDoublings。DIOIntervalDoublings 的默认值为 DEFAULT_DIO_INTERVAL_DOUBLINGS。

K:从 DIO 消息中学习当做 DIORedundancyConstant。DIORedundancyConstant 的默认值为 DEFAULT_DIO_REDUNDANCY_CONSTANT。在 RPL 中,当 k 的值为 0x00 时将被看做是冗余无限大常量,即 Trickle 从不抑制消息。

9.4 DODAG 选择

DODAG 选择依赖于实施部署和 OF。为了限制不稳定运动,并且所有的权值是相等的,节点应该保持其先前的选择。并且,节点应该提供一种方法来过滤出一个父节点,该父节点的可用性被探测是变动的,至少这时候还有更多稳定的选择可供使用的情况下。

出于安全或者其他原因当连接到一个固定 DODAG 时是不合理的,分散的 DODAGs 可能会为了允许低功耗有损网络中的连通性尽可能多的聚集到大的 DODAGs 当中去。

节点应该验证双向的连通性和充足的链路质量对于一个候选邻居是否可用,在他确认该候选者作为它的 DODAG 父节点之前。

9.5 作为叶节点的操作

某些情况下一个 RPL 节点可能只会以叶节点的身份归属于一个 DODAG。这种情况的一个例子是一个节点不理解或者不支持 RPL 实例的 OF 或者通告的权值/约束。按照有关的方针函数的描述,节点可能会以叶节点的身份加入 DODAG 中或者不加入 DODAG。

一个叶节点不会扩展 DODAG 的连通性,但是在某种情况下叶节点可能仍然需要偶尔传输 DIO 消息,尤其是当叶节点不会总是被当做叶节点并且不一致性被检测出来。

节点当做一个叶节点操作必须遵循下面的规则:

- a) 必须不能传输包含 DAG 权值容器的 DIO 消息;
- b) 其 DIO 消息必须通告值为 INFINITE_RANK 的 DAGRank;
- c) 它可以禁止 DIO 消息的传输,除非当一个数据包被转发或者回复一个单播的 DIS 消息时, DIO 消息的传输由于不一致的检测已经被触发,这种情况下 DIO 消息的传输不能被禁止;
- d) 它可能传输单薄的 DAO 消息,如 10.2 规定;
- e) 他可能传输单播的 DAO 消息到距离他“一跳”范围的邻居节点,如 10.10 规定。

一种特殊的情况中要求叶节点发送 DIO 消息如果那个叶节点是另一个 DODAG 中的优先成员并且,另外一个节点以旧的拓扑转发一条消息,引发了不一致。叶节点需要传输一条 DIO 消息来修复这种不一致。注意由于低功耗有损网络的松散本质,即便叶节点已经乐观地 poison 了它的路由通过通告一条 INFINITE_RANK 的序列在旧的 DODAG 中并且在成为叶节点之前,这条通告可能会被丢掉然后一个叶节点必须有能力在之后发送一条 DIO 消息为了修复这种不一致。

通常的情况下,叶节点不能够以路由的方式通告其本身。

9.6 管理的序列

某些情况下修改节点通告的序列可能是有益的,除了以一些实现特定方针和节点的属性为基础通过

OF 计算出来的之外。例如，一个只有有限电量的节点应该是一个叶节点除非没有其他选择，并且可能会增加通过 OF 来获得的序列值计算来显示一个夸张的序列。

10 向下路由

描述 RPL 如何发现和维护向下路由。RPL 使用 DAO 消息构建和维护向下路由。向下路由支持 P2MP 流，范围从 DODAG 根到叶子。向下路由同样也支持 P2P 流：P2P 消息能通过一条向上路由流向一个 DODAG 根，然后通过一条向下路由从 DODAG 根奔向一个目的节点。

本标准描述了一个 RPL 实例可能从维护的路由中选择的两种模式。第一种模式，称为“存储模式”，节点为它的 DODAG 子集存储向下路由表。在存储模式网络中，一条向下路由中的每一跳要通过检查它的路由表来决定下一跳。第二种模式，称为“非存储”，节点不存储向下路由表。向下数据包按照 DODAG 根中的源路由进行发送，见 IETF RFC6554。

RPL 允许一个简单的一跳 P2P 来最佳化存储和非存储网络。一个节点可能会对距离他自身只有一跳的邻居发送 P2P 包。

10.1 目标通告父节点

为了确定向下路由，RPL 节点发送向上的 DAO 消息。这些 DAO 消息的下一跳目的节点被称为父节点。一个节点的 DAO 父节点的聚集被称为 DAO 父集。

a) 一个节点可以使用 all-rpl-nodes 多播地址发送 DAO 消息，这样做是用来提供一条最优化的一跳路由。传输多播 DAO 消息时 ‘K’ 位必须被清除；

b) 一个节点的 DAO 父集必须是它的 DODAG 父集的一个子集；

c) 存储模式中，节点一定不能对非 DAO 父节点的节点配置单播 DAO 消息；

d) 存储模式中，一条 DAO 消息中的 IPv6 的源和目的地址必须是本地链路地址；

e) 非存储模式中，节点一定不能对非 DAO 根节点配置单播 DAO 消息；

f) 非存储模式中，一条 DAO 消息中的 IPv6 源和目的地址必须是唯一本地地址或者全局地址。

DAO 父集的选择依赖于目标函数。

10.2 向下路由的发现和维护

目的通告可能会被配置成完全地不可用，或者配置成存储或非存储模式，正如 DIO 消息中 MOP 记录的那样。

a) 所有节点加入一个 DODAG 必须遵守根的 MOP 设定。节点没有像路由器一样完全分担的能力，即节点不满足通告的 MOP，可能会以叶子的身份加入 DODAG；

b) 如果 MOP 为 0，表示没有向下路由，节点一定不能传输 DAO 消息，可能会忽略 DAO 消息；

c) 在非存储模式下，DODAG 根应该存储从 DAO 消息学习来的要进入目标地址的路由表。如果根存储某些信息失败了，那么一些目的地址将会不可到达；

d) 在存储模式下，所有非根、非叶节点必须存储从 DAO 消息学习来的要进入目标地址的路由表。

一个 DODAG 能拥有一个或者几个可能的运行方式，就像 MOP 字段中定义的一样。或者他不支持向下路由，支持通过从 DODAG 根的源路由的向下路由，或者支持通过 in-networking 路由表中的向下路由。

当向下路由是通过 in-networking 路由表所支持的，本标准中所定义的多播操作可能或者可能不被支持，同样是由 MOP 字段来显示。

当向下路由是通过 in-networking 路由表所支持的如同本规格中所描述的, 节点被假设当作路由器并且能够充分地维持请求路由表的状态的能力是很被期望的。如果被当作路由器的节点不能够维持整个路由表的状态, 那么路由表状态是不完整的, 消息可能会因此被丢掉, 同时会有一个错误被记录。未来对于 RPL 的扩展将会详细重新定义行为/动作来处理这种情况。

本标准中 RPL 不支持混合模式操作, 即一些节点源路由和其他存储路由表: 未来对于 RPL 的扩展可能会支持这种模式。

10.2.1 路径顺序的维护

对于每一个节点关联的目标, 该节点负责发送 DAO 消息以便 provision 向下路由。这些 DAO 消息中所包含的 Target+Transit 信息随后向上传播到 DODAG。传输信息选项中的路径顺序计数器用来指示新鲜和更新旧的向下路由信息。

对于每一个节点关联的目标, 该节点必须增加路径顺序计数器, 并生成一个新的 DAO 消息, 当:

- a) 路径周期将要被更新 (即刷新或者 no-path);
- b) 父节点地址表将会被改变。

对于每一个节点关联的目标, 该节点可能增加路径顺序计数器, 并生成一个新的 DAO 消息, 偶尔为了刷新向下路由信息。在存储模式中, 节点向它的每一个 DAO 父节点生成这种 DAO 消息以便允许多路径。所有在同一时间对同一目标的生成的 DAO 消息在传输信息中必须具有相同的路径顺序。

10.2.2 DAO 消息的生成

当一个节点收到 DAO 消息时他可能会发送 DAO 消息, 在其 DAO 父集中导致改变的结果, 或者在相应另一个事件例如一个相关前缀周期的到期。当接受 DAO 消息时, 节点关心 DAO 消息是否是“新的”, 或者包是否含新的信息。在非存储模式中, 每一条 DAO 消息都是“新的”。在存储模式下, 如果一条 DAO 消息满足下面条件中的任何一条, 那么这条消息是“新的”:

- a) 有一个更新的路径序列号;
- b) 有额外的路径控制位, 或者;
- c) 是一条 NO-Path DAO 消息, 清除最后的向下路由到一个前缀。

节点从它的 DODAG 子集收到一条 DAO 消息, 如果该 DAO 消息不是“新的”, 该节点可能会禁止继续传输 DAO 消息。

10.3 DAO 基本规则

a) 如果一个节点发送一条比先前传输 DAO 消息更新或者带有不同信息的 DAO 消息, 它一定会增加 DAOSequence 字段且至少加 1。如果一条 DAO 消息传输与他之前的 DAO 消息传输相同则可能加 DAOSequence 字段;

b) DAO 消息中的 RPL 实例标识和 DODAGID 字段的值必须相等作为节点父集中的一员和所传输的 DIO 消息;

c) 一个节点可以在一条单播 DAO 消息中设置 ‘K’ 标志位来请求一个单播的 DAO-ACK 消息作为回应以便确认请求;

d) 节点接受了一条 ‘K’ 标志位置位的单播 DAO 消息应该回应 DAO-ACK 消息。节点接受了一条 ‘K’ 标志位没有被置位的单播 DAO 消息可能会回应一条 DAO-ACK 消息, 尤其是报告一条错误条件;

e) 一个节点在一条单播 DAO 消息中设置了 ‘K’ 标志位, 但是没有收到回应的 DAO-ACK 消息可

以重新发送 DAO 消息再次尝试，直到一个规定的重试次数；

f) 节点应该忽略没有新序列号的 DAO 消息并不再处理该消息。

与 DIO 消息中的版本字段不同，版本字段仅仅只会被 DODAG 根增加而对其他的节点响应不变，DAOSequence 值对每个节点都是唯一的。序列号空间对于单播和多播的 DAO 消息来说可以相同也可以不同。推荐使用相同的序列号空间。

10.4 DAO 消息的结构

存储模式网络和非存储模式网络中的 DAO 消息遵循共同的结构。按照大多数通用形式，一条 DAO 消息可能包含几组选项，每一组是由一个或多个目标选项和跟随着的一个或多个传输信息选项组成。整个组的传输信息选项应用于整个组的目标选项。

a) RPL 节点在其所传输的每一条 DAO 消息中必须包含一个或多个 RPL 目标选项。如果该节点需要 DODAG 为其提供向下路由的话，它的 RPL 目标选项必须包含节点 IPv6 地址的前缀。RPL 目标选项后面可能会紧随一个验证它的不透明的 RPL 目标描述符选项；

b) 当一个节点为一个目标选项而更新其传输信息选项中的信息而覆盖了其中的一个地址，他必须在该传输信息选项中增加路径序列号。路径序列号可能会不定期的增加以便刷新向下路由；

c) 一条单播 DAO 消息中的一个或多个 RPL 目标选项必须紧跟着一个或多个传输信息选项。所有传输信息选项应用于所有紧挨着在它们之前的目标选项；

d) 多播的 DAO 消息中的传输信息选项中一定不能包含有父节点地址；

e) 节点接受和处理包含特定目标信息的一条 DAO 消息时，目标含有之前的信息，必须使用传输信息选项中的路径序列号与目标关联来决定 DAO 消息是否包含更新信息；

f) 如果一个节点不按照上述规则接受 DAO 消息，那么必须丢弃该 DAO 消息而不做进一步处理。

非存储模式中，根节点构建一个严谨的源路由头部，逐跳地，递归地查询绑定到一个目标每一跳信息（地址和前缀）以及一个传输地址。某种情况下，当一个子节点从父节点所通告的前缀中获取地址时，这种父节点与子节点关系会被根推测出来，目的是为了构建源路由头部。其他的情况下非常有必要通知根节点这种可到达目标节点的传输-目标关系，以便稍后允许递归构建路由头部。一条 DAO 消息中作为目标通告的地址必须在同一路由器中是 collocated，或者是路由器拥有的可以到达的 onlink，这些都在相关传输信息表示出来。下面规则适用于确保端对端源路由路径的连通性：

a) 父节点在传输选项中所用的地址必须是从一条‘R’置位的父节点 PIO 消息中获得。PIO 消息中‘R’标志位表明前缀字段实际包含全部父节点地址但是子节点不应该假设父节点地址是 onlink。

b) 一条‘A’标志位被置位的 PIO 消息表示 RPL 子节点可能会用前缀自动配置地址。一个父节点在一条‘A’标志位被置位的消息中通告了一个前缀，一定要确保 PIO 消息中的地址或者整个前缀是被根作为一个 DAO 目标并通告出去从而可以到达的。如果父节点同样也置位了‘L’标志位表明前缀是 onlink 的，那么他必须在一条 DAO 消息中通告这个前缀作为目标。

c) 一条 DAO 消息中作为目标通告的地址必须在同一路由器中是 collocated，或者是路由器拥有的可以到达的 onlink，这些都在相关传输信息表示出来。

d) 为了使路由头部能够更加优化地压缩，父节点应该在所有 PIO 消息中设置‘R’标志位和‘A’标志位，‘L’位清除，子节点应该会偏向使用从 PIO 消息中发现的父节点传输地址，用来自动配置 DAO 消息中作为通告目标的地址。

e) 一个路由器可能会有一个不被父节点知道的 onlink 的目标, 或者是因为它们被地址分配在交替的接口上, 或者是因为它们属于 RPL 外部的节点, 例如连接到主机上。为了在 RPL 网络中加入这个目标, 路由器必须在传输信息选项中以父节点地址向那个目标通告自己, 并使用一个对那个节点 DAO 父节点来说是 onlink 的地址。如果目标属于一个额外的节点那么路由器必须在传输信息中设置额外的 ‘E’ 标志位。

一个子节点从一条 ‘L’ 标志位被置位的 PIO 消息已经自动配置了地址就没有必要通告该地址作为一个 DAO 目标, 因为父节点确认整个前缀是被根可以到达的。但是如果 ‘L’ 标志位没有被置位子节点就非常有必要在非存储模式中通知根这种父子关系, 利用一条可以到达的父节点地址, 以便确保路由头部能循环的构建。这是由一条 DAO 消息中关联一个父节点的地址作为传输和一个子节点的地址作为目标来完成的。

10.5 DAO 传输行程

因为 DAO 消息向上传输, 所以当接收到一条单播的 DAO 消息时, 能发送一条单播 DAO 消息给一个 DAO 父节点。

a) 当接收一条带有更新信息选项的单播 DAO 消息时, 如包含一个带有一个新路径序列的传输信息选项, 节点应该发送 DAO 消息。不必立即发送这条 DAO 消息, 而应该延期发送 DAO 消息为了从其他 DAO 父节点那里聚合 DAO 消息;

b) 节点应该使用定时器 (DelayDAO) 来延期发送 DAO 消息。接收一条 DAO 消息同时开启了定时器。DAO 消息的接收是在 DelayDAO 定时器活跃时, 并不会重置定时器。当 DelayDAO 定时器到期时, 节点发送 DAO 消息;

c) 当一个节点增加了一个新节点到其 DAO 父集中, 应该传输一条 DAO 消息的。

DelayDAO 的值和计算依赖于具体实施。本标准定义的默认值为 DEFAULT_DAO_DELAY。

10.6 触发 DAO 消息

节点能触发它的 DODAG 子集发送 DAO 消息。每一个节点持有一个 DAO 触发序列号 (DTSN), DTSN 是通过 DIO 消息中传递过来的。

a) 如果一个节点知道了它的一个父节点增加其 DTSN, 节点必须使用 10.3 和 10.5 节中的规则传输 DAO 消息;

b) 非存储模式中, 如果一个节点知道了它的一个父节点增加其 DTSN, 节点必须增加他自己的 DTSN。

在存储模式中, 作为日常路由表的更新和维护, 一个存储节点可能会增加它的 DTSN, 从而可靠地从他直接的子节点触发一组 DAO 更新消息。在存储模式的模式中, 没有必要从整个 DODAG 子集中触发 DAO 更新消息, 因为状态信息将会逐跳地在 DODAG 中传播。

在非存储模式中, DTSN 的增加同样也会导致一个节点的直接子节点依次地增加它们的 DTSN, 从它们的子 DODAG 中触发一组 DAO 更新消息。在非存储模式的中一个典型的特征是仅仅根节点会独立地增加 DTSN, 当一条 DAO 消息需要被刷新时但是一个整个的修订 (例如增加 DODAGVersionNumber) 是不被渴望的。在非存储模式中, 另一个典型的特征是所有的非根节点将会增加它们的 DTSN, 仅仅当它们的父节点也被观察到增加了 DTSN。一般来说, 节点可能触发 DAO 消息更新取决于执行细节逻辑, 例如向下路由不一致性的探测或者偶尔地根据一个内部定时器。

在触发 DAO 消息的情况下，选择一个合适的 DAODelay 能够大大地减少 DAO 消息传输的数目。触发向下流动 DODAG；最好的情况下 DAO 消息向上流动 DODAG 使得叶子首先发送 DAO 消息，每个节点仅仅只发送一次 DAO 消息。这样地安排约等于设置 DAODelay 与 Rank 成反比。注意这条建议被打算当做允许有效集合体的最优化。

10.7 非存储模式

非存储模式中，向下的 RPL 路由消息使用 IP 源路由。

a) 父节点传输信息选项中的地址字段必须包含一个或多个地址。所有这些地址必须是发送者 DAO 父节点的地址；

b) DAO 消息会沿着一条默认路由被直接发送给根节点；

c) 当一个节点从它的 DAO 父集中移掉一个节点时，会生成一条新的带有更新传输选项的 DAO 消息。

非存储模式中，节点用 DAO 消息来报告其 DAO 父节点给 DODAG 根节点。DODAG 根能够通过使用路由中每个节点的 DAO 父集拼接出一条向下路由给一个节点。路径序列信息可能会被用来探测旧的 DAO 信息。这样逐跳路由计算的目的是当 DAO 父节点改变时最小化流量。如果节点报告完整的源路由，那么当一个 DAO 父节点改变时整个的 DODAG 子集将会不得不向 DODAG 根发送一条新的 DAO 消息。因此，在非存储模式中，节点可能发送一条单独的 DAO，也可能会选择发送多条 DAO 消息给多个 DAO 父节点。

节点通过发送一条单独的带有多个 RPL 目标选项的 DAO 消息包装 DAO。每一个 RPL 目标选项有其自己的传输信息选项。

10.8 存储模式

在存储模式中，RPL 路由消息向下是通过 IPv6 目的地址，下面的规则应用于存储模式中的节点：

a) 传输信息选项中的父节点地址字段必须为空；

b) 当接收一条单播的 DAO 消息时，节点必须计算 DAO 是否将会改变节点自己通告的前缀集合。这个计算应该包含与 DAO 关联的传输信息选项中路径序列信息的商议，来决定是否该 DAO 消息包含新的信息来取代已经存储于节点中的信息。如果是这样，节点必须生成一条新的 DAO 消息并传输。这样的变化包括接收到一条 NO-PATH 的 DAO 消息；

c) 当节点生成了一条新的 DAO，应该对其每一个 DAO 父节点单播出去。不能对不是其 DAO 父节点的节点单播 DAO 消息；

d) 当一个节点从其 DAO 父集中移除一个节点时，应该向那个被移除的 DAO 父节点发送一条 no-path DAO 消息以便使已经存在的路由无效；

e) 如果发向一个被通告的向下地址的消息遭受了一个传输错误，邻居不可到达检测 (NUD)，或者类似的失败，节点会标记该地址为不可到达并且生成一条合适的 no-path DAO 消息。

DAO 消息通告一个节点所路由到的目的地址和前缀。不同于非存储模式，这些 DAO 消息自己不交流有关路由的信息：路由信息存储于网络中并且从 IPv6 源地址中暗示出来。当一个存储节点生成一条 DAO 消息时，使用所接收到的 DAOs 的存储状态来生成一组 RPL 目标选项以及所关联的传输信息选项。

因为这条信息存储于每一个节点路由表中，存储模式中 DAOs 是直接与 DAO 父节点沟通的，这些父节点存储这些信息。

10.9 路径控制

DAO 消息包含一个或多个目标选项。每一个目标选项或者指定了节点的前缀通告, 一个低功耗有损网络外部可到达的前缀地址, 节点的 DODAG 子集中的目的地址, 或者指定 DODAG 子集中节点所监听的一个多播组。传输信息选项中路径控制字段允许节点请求或者顾及到多播向下路由。节点构造传输信息选项中的路径控制字段要遵循如下规则:

- a) 路径控制字段的位的宽度必须等于 $(PCS+1)$ 的值, PCS 的值是由 DODAG 配置选项中的控制字段指定。数据位大于或者等于 $(PCS+1)$ 的值必须在传输中被清除并且在接收中必须被忽略。低于 $(PCS+1)$ 值的位被认为是“活动”位;
- b) 节点必须逻辑的构造其 DAO 父节点组当 populating 路径控制字段时, 每一组由相等优先权的 DAO 父节点组成。这些组必须依照优先权排序, 从而顾及到 DAO 父节点到路径控制子域的一个逻辑映射。组可能会重复地以便伸展到整个路径控制字段的位宽度, 但是顺序, 包括重复组, 必须被保留以便优先权能被适当的传达;
- c) 因为一个 RPL 目标选项描述了一个节点自己的地址或者低功耗有损网络外部的一个前缀, 路径控制字段中的至少一个活动位被用来设置。更多的路径控制字段活动位可能会被设置;
- d) 如果一个节点接收到了多个具有相同 RPL 目标选项的 DAO 消息, 它必须把所接收到的路径控制字段进行按位或运算。这个聚合的按位或操作代表了向下路由前缀请求的数目;
- e) 当一个节点向其 DAO 父节点发送一条 DAO 消息时, 它必须选择一个或者多个位被设置成活动的在子字段中, 该位是一组从聚合路径控制字段中包含 DAO 父节点的映射。一个给定的位只能仅仅对一个父节点设置成活动的。所传递给父节点的 DAO 消息必须使这些活动位被置位而其他活动位被清零;
- f) 对于 RPL 目标选项和 DAOSequence 号来说, 节点发送给不同父节点的不同的 DAO 消息必须有不相交的活动路径控制位。节点对于两个不同的 DAO 父节点一定不能设置相同的 DAO 消息活动位;
- g) 路径控制位应该根据路径控制子域中 DAO 父节点的优先权映射来分配, 例如活动路径控制位, 或者组群位, 这些属于特殊的路径控制子域被分配给该子域对应映射的组群中的 DAO 父节点;
- h) 非存储模式的操作模式中, 一个节点可能会透传 DAO 消息而对路径控制字段不做任何处理;
- i) 一个节点一定不能单播一条路径控制字段中活动位没有被设置的 DAO 消息。有可能, 对于一个给定的目标选项, 节点没有足够聚合路径控制位来发送包含目标的 DAO 消息给它的每一个 DAO 父节点, 这种情况下具有最小优先权的 DAO 父节点可能不会从目标获得一条 DAO 消息。

路径控制字段允许节点绑定指向其的向下路由。在路径控制字段中设置一些位等值于其所偏爱的向下路由的最大数。每一位至多被发送给一个 DAO 父节点; 一簇位能被发送给一个单独的 DAO 父节点来在它自己的父节点中划分。

为目标提供了一条拥有关联路径控制字段的 DAO 路由的节点, 应该使用路径控制字段中的内容以便来决定到目标众多可选择地 DAO 路由的优先顺序。路径控制字段的分配取决于优先级 (DAO 父节点的), 该优先级是以节点的每一个目标函数在“向下”方向上“端到端”合计权值的基础根据的。在非存储模式中根节点能从每条接收的 DAO 消息中聚合信息来决定向下的路由, 其中包含首选 DAO 父节点的路径控制显示。

10.9.1 路径控制事例

假设有一个存储模式的低功耗有损网络包含一个节点 N 和 4 个父节点 ($P1$ 、 $P2$ 、 $P3$ 和 $P4$)。 N 有 3

个子节点，DODAG 子集中的 C1、C2 和 C3。PCS 为 7，如此的话，路径控制字段将会有 8 位的活动位：1111111b。考虑下面的例子：

路径控制字段被划分成 4 个子域，PC1 (11000000b)，PC2 (00110000b)，PC3 (00001100b)，(00000011b)，如此的话，这 4 个子域代表了 4 种不同优先级别如同图 27 所示。本事例中节点 N 上的操作，组 {P1, P2} 相互之间具有相同优先级别和最优先考虑的集团整体。{P3} 落后于 {P1, P2}，但是优先于 {P4}。让节点 N 像下面这样执行它的路径控制映射：

{P1, P2} -> PC1 (11000000b) in the Path Control field

{P3} -> PC2 (00110000b) in the Path Control field

{P4} -> PC3 (00001100b) in the Path Control field

{P4} -> PC4 (00000011b) in the Path Control field

注意到实际中重复了 {P4} 以便获得完整的路径控制字段范围。

a) 假设 C1 发送了一条包含目标 T 的路径控制 10000000b 的 DAO。节点 N 存储了关联 10000000b 的条目通过 C1 和目标 T 的路径控制字段；

b) 假设 C2 发送了一条包含目标 T 的路径控制 00010000b 的 DAO。节点 N 存储了关联 00010000b 的条目通过 C2 和目标 T 的路径控制字段；

c) 假设 C3 发送了一条包含目标 T 的路径控制 00001100b 的 DAO。节点 N 存储了关联 00001100b 的条目通过 C3 和目标 T 的路径控制字段；

d) 经过一段时间后，节点 N 生成了一条目标为 T 的 DAO 消息。节点 N 将会构建一个聚合的路径控制字段通过或运算它的每一个子节点向目标 T 所发送 DAO 消息中的路径控制字段。聚合的路径控制字段因此具有的活动位为：10011100b；

e) 接下来节点 N 分发这个聚合的路径控制位给它的父节点 P1、P2、P3 和 P4 以便准备 DAO 消息；

f) P1 和 P2 是最有资格从首选子域接收到活动位的 (11000000b)。在聚合的路径控制字段中这些位的值为 10000000b。节点 N 必须只能从 2 个父节点中选一个出来。这种情况下节点 P1 被分配到了这个位，获得了路径控制字段 10000000b 为它的 DAO。没有为节点 P2 留下合适的配置位，因此节点 P2 将会有个值为 00000000b 路径控制字段，并且因为没有活动位不会生成 DAO 消息给节点 P2；

g) 第二优先选的子域 (00110000b) 有活动位 00010000b。节点 N 已经映射了 P3 到它的子域中。节点 N 可能会分配活动位给 P3，构建一条 P3 的包含目标 T 并带有路径控制为 00010000b 的 DAO 消息；

h) 第三优先选的子域 (00001100b) 有活动位 00001100b。节点 N 已经映射了 P4 到它的子域中。节点 N 可能会分配活动位给 P4，构建一条 P4 的包含目标 T 并带有路径控制为 00011000b 的 DAO 消息；

i) 最后选的子域 (00000011b) 没有活动位。如果有活动位，这些位将会被添加到 P4 构建的 DAO 消息中的路径控制字段；

j) 传播去往 P1、P2、P3、P4 的并带有其他目标收益的 DAO 消息的过程依照从这些目标中收集来的聚合路径控制字段。

10.10 多播目标通告消息

一种特殊的 DAO 操作情况，有别于单播的 DAO 操作，是被用来传播“1 跳”路由表条目的多播 DAO 操作。

a) 一个节点可能会多播一条 DAO 消息给本地链路范围内的所有 RPL 节点的多播地址；

b) 一条多播的 DAO 消息必须只能用来通告节点本身的信息, 即节点所拥有或者直接连接的前缀, 例如节点所订阅的一个多播组或者节点所拥有的全局地址;

c) 一条多播的 DAO 消息一定不能用来转播从其他节点学习来的连通性信息;

d) 一个节点一定不能执行任何其他 DAO 相关处理在一条接收的多播 DAO 消息中, 尤其是一个节点一定不能一收到一条多播 DAO 消息就执行 DAO 父节点的动作。

多播 DAO 消息可能会被用来使直接 P2P 通讯成为可能, 而不需要 DODAO 转发数据包。

11 安全机制

本章规定了安全的 RPL 消息的生成和处理。RPL 消息编码中高字节确定了一条 RPL 消息是否是安全的。除了基本控制消息 (DIS、DIO、DAO、DAO-ACK) 的安全版本, RPL 拥有几条在网络中专门确保安全的消息。

实施的复杂性和规格是低功耗有损网络核心关注的问题, 例如一条 RPL 的实现中可能是由于经济的或者物理原因的不可能性而包含复杂的安全条款。更多的是, 许多的调度能利用链路层或者其他安全机制来满足它们的安全要求而不需要使用 RPL 中的安全机制。

因此, 本标准中的安全特性是可选择的来实现。一个给定的实现可能支持所描述的安全特性的一个子集 (包含空集), 例如他可能支持完整性和保密性, 但是不支持签名。一个实现必须清楚地指明哪一种安全机制是被支持的, 并且建议实施者要小心的考虑安全需求和网络中安全机制的可用性。

11.1 安全性概述

RPL 支持三种安全模式:

1) 不安全: 这种安全模式下, RPL 只使用基本的 DIS、DIO、DAO 和 DAO-ACK 消息, 而没有安全部分。因为网络可能使用其它的安全机制, 如链路层安全, 所以不安全模式并不意味着所有消息没有经过保护措施而被发送;

2) 预置安装: 这种安全模式下, RPL 使用安全消息。要加入一个 RPL 实例, 节点必须拥有一个密钥。节点使用该密钥确保消息的机密性、完整性和可靠性。一个节点, 使用这中预置安装的密钥, 以主机或者路由器的身份加入到 RPL 网络中去;

3) 已认证: 这种安全模式下, RPL 使用安全消息。要加入一个 RPL 实例, 节点必须拥有一个密钥。节点使用该密钥确保消息的机密性、完整性和可靠性。一个节点, 使用这种预置安装的密钥, 只能以主机身份加入到 RPL 网络中去。要以路由器的身份加入到网络中, 节点必须拥有从已认证实体中获得的第二个密钥。这个已认证实体能够在提供第二个密钥之前鉴别请求者是否被允许作为路由器。已认证模式不被对称算法支持。本标准中 RPL 只支持对称算法。

RPL 实例是否使用不安全模式是由它是否使用 RPL 安全消息来体现的。一个安全网络是否使用预置安装模式或者已认证模式是由 DAG 配置选项中的 'A' 位来决定的。

本标准定义了 CCM—带有 CBC-MAC 的计数器 (链接式分组编码消息认证码) ——作为 RPL 安全的加密基础。本标准中使用 AES-128 作为优先使用的加密算法。在安全区域有预留位用来指定未来会用到的其他算法。

所有的安全 RPL 消息都有一个消息标识代码 (MAC) 或者一个签名。安全 RPL 消息选择性的拥有机密性加密保护。安全 RPL 消息格式既支持整合的加密/认证方案, 也支持分开的加密和认证数据包的方案。

11.2 加入一个安全网络

RPL 安全假设一个节点想要加入一个安全网络，会事先有个预配置的共享密钥用来与邻居节点和 RPL 根节点通信。为了加入一个安全的 RPL 网络，节点或者监听安全 DIO 消息，或者通过发送一条安全 DIS 消息来触发安全 DIO 消息。除了第 9 章所规定的 DIO/DIS 消息规则，安全 DIO/DIS 消息还要遵循下面这些规则：

- a) 如果发送 DIS 消息，初始的安全 DIS 消息必须设置密钥安全标识模式字段为 0 (00)，必须设置安全级别字段为 1 (001)。所使用密钥必须被预先配置成组密钥（密钥索引 0x00）；
- b) 当节点重置了它的 Trickle 定时器以响应一条安全 DIS 消息时，那么所传输的下一条 DIO 消息，一定是带有与之前安全 DIS 消息相同安全配置的安全 DIO 消息；
- c) 当节点发送一条 DIO 消息来响应一条单播的安全 DIS 消息时，DIO 消息一定是一条安全 DIO 消息。

上述规则允许一个节点使用一个预先配置的共享密钥来加入一个安全的 RPL 实例。一旦节点使用预先配置共享密钥加入 DODAG，配置选项中的‘A’位决定了它的能力。如果配置选项的‘A’位被清除，那么节点可以使用预先安装共享密钥来正常的交换消息：发送 DAO、DIO 消息等等。

如果配置选项的‘A’位被设置并且 RPL 实例运行在已认证模式：

- a) 节点不能通告除了 INFINITE_RANK 之外的 RANK 在一条安全 DIO 消息中，该消息以密钥索引 0x00 来确保安全。当处理一条密钥索引为 0x00 的安全 DIO 消息时，一个处理节点必须把通告的 RANK 认为是 INFINITE_RANK。消息中 RANK 的值为其他任何值，消息都应该被丢弃；
- b) 使用安全密钥索引 0x00 的安全 DAO 消息，一定不能拥有带有前缀信息（除了节点地址之外）的 RPL 目标选项。如果节点使用预置安装共享密钥接收一条安装 DAO 消息，RPL 目标选项不匹配 IPv6 源地址，它必须丢弃这条安全 DAO 消息而不进行进一步处理。

上述规则意味着，在一个‘A’位被设置的 RPL 实例中，使用密钥索引 0x00 节点只能以主机身份加入 RPL 实例，但不能作为一个路由器。节点必须使用一个权威的密钥来交流，以便获得能使它能作为一个路由器加入 RPL 实例的密钥。

11.3 安装密钥

已认证模式需要一个想要成为路由器的节点，动态的安装新的密钥一旦它们以主机的身份加入到网络中。以主机身份加入网络中，节点使用标准 IP 消息与已认证服务器通讯，该服务器提供新的密钥。

11.4 一致性校验

RPL 节点发送一致性校验（CC）消息来防御重播攻击和同步计数器。

- a) 如果节点接收到一条 R 位被清除的单播 CC 消息，并且它是所关联的 DODAG 的一个成员或者，处于加入该 DODAG 过程当中，它应该回复一条单播 CC 消息给发送者。该回复中 R 位必须被设置，CC Nonce，RPL 实例标识和 DODAGID 字段必须与所接收到消息中的相同；
- b) 如果节点接收到一条多播的 CC 消息，它必须丢弃该消息而不做进一步处理。

一致性校验消息允许节点发送一条挑战--响应来验证节点当前的计数器的值。因为 CC Nonce 由挑战者生成，作为对手的重播消息是不太可能产生一个正确的响应。一致性校验响应中的计数器允许挑战者来验证它所听到的计数器的值。

11.5 计数器

最简单的一种情况,计数器的值是一个无符号整数,节点在每一条安全 RPL 传输中以增量为 1 或者更多来增加计数器的值。计数器可以表示一个拥有下面属性的时间戳:

- a) 时间戳必须至少长度为 6 个字节;
- b) 时间戳必须以 1024Hz (二进制毫秒) 为间隔尺寸;
- c) 时间戳的开始时间必须是世界协调时间 1970 年 1 月 1 日 上午 12:00:00 点;
- d) 如果计数器表示这样的时间戳,计数器的值必须按照下面计算得出。 T 为时间戳, S 为使用的密钥的开始时间, E 为使用的密钥的结束时间。 S 和 E 都会按照上面描述的时间戳的规则表示。如果 $E > T < S$, 那么计数器是无效的,节点不会生成一个包。否则,计数器的值等于 $T-S$;
- e) 如果计数器表示这样的时间戳,节点必须可以设置安全 RPL 包中的安全部分的 'T' 标志位;
- f) 如果计数器字段不表示这样的时间戳,节点一定不能设置这样的 'T' 标志位;
- g) 如果节点没有一个本地时间戳以满足上面的要求,它必须忽略 'T' 标志位。

如果一个节点支持这样的时间戳并收到一条 'T' 标志位被设置的消息,它可以在接收的消息中应用 11.7.1 节中规定的暂时校验。如果节点收到一条 'T' 标志位没有被设置的消息,它一定不能应用这种暂时校验。一个节点的安全策略可以,因为应用的原因,包括拒绝所有不设置 'T' 标志位的消息。

'T' 标志位被使用因为现如今的许多低功耗网络,已经以紧次于毫秒的间隔尺寸维持全局时间同步,为了安全、应用和其他的原因。允许 RPL 利用现有的一些功能,目前大大简化了一些安全问题的解决方案,例如延时保护。

11.6 外出数据包的传输

给定一个外出 RPL 控制数据包且需要安全保护,本章节描述 RPL 如何生成安全数据包来传输。同时也描述了编码操作的顺序以提供必须的保护。

外出 RPL 数据包安全保护的需求和安全级别,应该是由节点的安全策略数据库决定的。外出数据包的安全策略数据库的配置过程是具体实施的。

在安全 RPL 消息将要被传输的部分,一个 RPL 节点一定要设定外出 RPL 数据包的安全部分 (T SEC KIM LVL),以描述所应用的保护级别和安全设置。RPL 消息代码字段中的安全子字段位一定要设置以表示安全 RPL 消息。

用于构建 AES-128 CCM Nonce 从而保护外出数据包的计数器的值,必须是传输到特定目标地址的最后的计数器的增量。

在安全策略指定延时保护应用的地方,用于构建 CCM nonce 而保护外出数据包的时间戳计时器,必须根据 11.5 节中的规则增加。在时间戳计数器应用的地方 ('T' 标志位被设置用于标明),本地维护时间计数器被包括进来作为传输安全 RPL 消息的一部分。

用于保护外出数据包的加密算法应该由安全策略数据库指定,并且必须在外出消息的 Sec 字段用值标识出。

外出数据包的安全策略应该决定可适用密钥标识模式 (KIM) 和密钥标识,用于说明加密数据包过程中所使用的安全密钥,包括可选择使用的签名密钥。安全策略同样也将会指定算法和安全序列,以认证的形式或者,认证和密钥的形式,可能使用签名应用于外出数据包。

在加密应用的地方,节点必须使用数据包安全部分指定的安全保护,密钥和 CCM Nonce 加密过的负载代替原始的数据包。

所有安全 RPL 消息包括完整性保护。安全算法处理的同时，一个节点或者起源一个消息标识代码 (MAC) 或者签名，一定要被包括进来作为外出安全 RPL 数据包的一部分。

11.7 传入数据包的接收

本节规定了一个安全 RPL 数据包的接收和处理过程。给定一个传入的安全数据包，该数据包的安全子字段位被设置，本节描述 RPL 如何生成数据包的非加密变量并且证实它的完整性。

接收方使用安全控制字段来决定数据包安全处理过程。如果所描述的消息类型和发起者的安全级别是未知的，或者不满足本地维护安全策略，节点必须丢弃该数据包而不进行进一步处理，可以提高一个管理警报，且不发送任何回复消息。这些策略可能包括安全级别，使用密钥，源标识符，或者缺少以时间戳为基础的计数器（以 ‘T’ 标志位被设置表示）。传入数据包安全策略数据库的配置过程超出了本标准的讨论内容（它可以通过例如 DIO 消息配置或者外带管理路由器配置来定义）。

消息安全级别表明了一条加密的 RPL 消息，节点通过在消息负载加密过程中加入 KIM 字段和 CCM Nonce，以使用被标识密钥信息。CCM Nonce 应该来自于消息计数字段和其他接收的本地维护信息。明文消息内容应该被获得，通过反转加密模式的操作，该操作由接收数据包的 Sec 字段指定。

接收方应该使用 CCM Nonce 和被鉴定过的密钥信息来检查传入数据包的完整性。如果完整性检查失败，违反了接收消息标识代码 (MAC)，节点必须丢弃该数据包。

如果接收消息有一个初始计数器值 (0)，接收者有一个传入计数器维护当前消息的起始者，接收者必须通过发送一个一致性检验响应消息给发送方，以便开始一个计数器的再同步。一致性检验响应消息应该被保护，为了特定节点地址当前所维护的全部外出计数器。外出计数器将会被包含在消息的安全部分，然而传入计数器将会被包含在一致性检验消息的负载中。

基于指定的安全策略，节点对于一条接收 RPL 消息可以应用重播保护。重播检验应该在接收数据包的鉴定之前被执行。从传入数据包获得的计数器应该，与给定源原始节点地址的所保持的传入计数器的水印相比较。如果接收计数器的值为非 0 和小于所保持的传入计数器水印，一个潜在数据包重播所标识的，节点必须丢弃该传入数据包。

如果延时保护被指定作为传入数据包安全策略检验的一部分，那么时间戳计数器被用来证实接收 RPL 消息的及时性。如果传入消息时间戳计数器的值表明，一个消息传输时间优先于始发地址的本地保持的传输时间计数器，表明这是一个重播违例，节点必须丢弃该传入数据包。如果接收时间戳计数器的值表明一个消息传输时间，早于当前时间减去可接收数据包延迟，这表明这是一个延时违例并且节点必须丢弃该传入数据包。

一旦一条消息被解密，在使用的情况下，已经成功地通过完整性校验，重播，和可选择延时保护校验，那么节点能更新它的本地安全信息，例如重播比较的源预期计数器值。

一个节点在收到一条带有失败安全策略校验的消息或者，其他应用完整性，重播或者延时检验的失败，一定不能更新它的安全信息。

11.7.1 时间戳密钥校验

如果消息中的 ‘T’ 标志位被设置并且节点具有一个遵循 11.5 节中要求的本地时间戳，那么节点可能要校验消息的临时一致性。节点通过增加关联密钥起始时间的计数器值，来计算消息的传输时间。如果这个传输时间超过了密钥的结束时间，节点会丢弃该消息而不进行进一步的处理。如果传输时间与接

收方的本地时间相比较，时间靠前，它会丢弃该消息而不进行处理。

11.8 完整性和机密性的覆盖范围

对于一条 RPL ICMPv6 消息，全部的数据包都在 RPL 安全范围之内。

消息标识代码（MAC）和签名对于全部的非安全 IPv6 数据包都需要计算。当计算消息标识代码和签名时，可变的 IPv6 字段被看做以 0 作为填充，遵循 IETF RFC4302 中 3.3.3.1 的规则（IPSec 认证头）。消息标识代码和签名的计算要在底层可能应用的任何压缩之前执行。

当一条 RPL ICMPv6 消息是被加密的，加密开始于安全部分之后的第一个字节并且持续到数据包的最后一个字节。IPv6 头、ICMPv6 头和 RPL 消息一直到安全部分的结尾都是不加密的，因为它们都需要被正确地解密数据包。

例如，节点发送一条消息，LVL=1，KIM=0，Algorithm=0，使用 CCM 算法构建一个属性为 ENC-MAC-32 的数据包：加密数据包并且附加一个 32 位的 MAC。分组密码密钥是由密钥索引决定；CCM Nonce 是如 11.9.1 节描述计算出；要被鉴定和加密的消息是 RPL 消息，开始于安全部分之后的第一个字节，结束于数据包的最后一个字节；额外的认证数据开始于 IPv6 头的开始部分，结束于安全部分的最后一个字节。

11.9 加密操作方式

本标准中描述的加密操作方式（Algorithm=0）是以 CCM 和分组密码 AES-128 为基础的。该操作模式被现有的实现广泛的支持。CCM 模式需要一个 nonce（CCM Nonce）。

11.9.1 CCM Nonce

一个 RPL 节点构建一个 CCM Nonce，如图 31 所示。

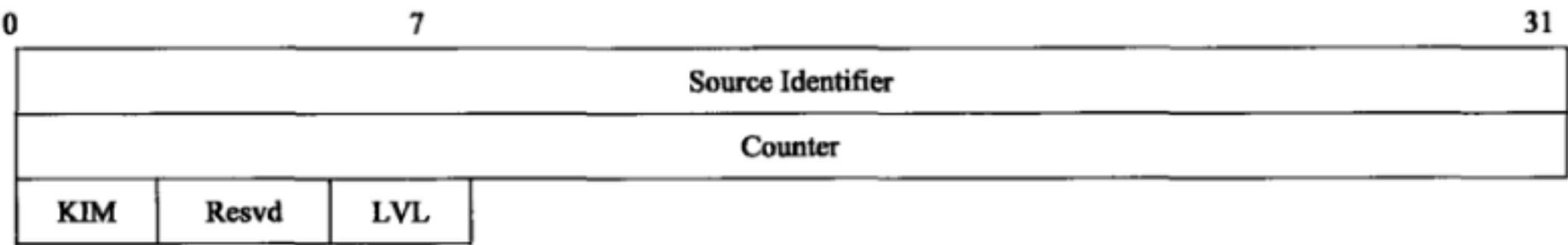


图 31 CCM Nonce

源标识符：8 字节。源标识符被设定为被保护数据包的发起者的逻辑标识符。

计数器：4 字节。计数器被设定为 RPL 控制消息安全选项的相应字段的（未压缩的）值。

密钥标识模式（KIM）：2 位。KIM 被设定为 RPL 控制消息安全选项的相应字段的值。

安全级别（LVL）：3 位。安全级别被设定为 RPL 控制消息安全选项的相应字段的值。

CCM Nonce 中的未分配的位用作保留。它们必须被设为 0 当构建 CCM Nonce。

CCM Nonce 中的所有字段都以最重要字节和最重要位顺序表示。

11.9.2 签名

如果密钥标识模式表示使用了签名（值为 3），那么节点附加一个签名到数据包的数据负载中。安全级别字段描述了签名的长度。

RPL 安全模式中的签名体制是 RSA 算法的一个实例。它像公钥对一样使用 (n,e)，n 是 2048 位或者 3072 位 RSA 系数， $e=2^{16}+1$ 。它使用 CCM 模式作为加密体制且 M=0（最为一个流密码）。注意到虽然在 IETF RFC 3610 不允许 M=0 的 CCM 模式，RPL 明确地允许 M=0 的 CCM 模式当与一个签名连同时使用时，因为签名提供了足够的数据认证。虽然在 IETF RFC 3610 中指定 M=0 的 CCM 模式，但是 M 字段必须设定为 0。它使用 SHA-256 哈希函数。它还使用由 IETF RFC3447 的 8.1 节规定的消息编码规则。

假设‘a’为计数器和消息头的 6 字节的表示的连接。数据包负载是数据包数据‘m’和签名‘s’的正确连接。这个签名体制以消息的 a 和 m 部分的正确的连接而被调用，然而签名确认是以消息的 a 和 m 部分以及签名 s 的正确连接而被调用。

这种形式的 RSA 签名为 RPL 网络提供了充分的保护。如果有必要，可选择的签名体制，该体制能产生更简明的签名并超出了本标准的范围，将会成为将来规范的主题。

一个支持 RSA 认证（该认证或者带有 2048 位签名或者带有 3072 位签名）的实现，应当支持带有 2048 位签名和带有 3072 位 RSA 签名的验证。这是考虑到为了 RPL 的发展而提供一种可升级的途径。

12 转发数据包和环路避免/探测

12.1 转发数据包的建议

这些非规范性的建议用于帮助转发实现的设计，并通过阐述这样的实现怎样与 RPL 协同工作的。

当转发一个数据包到目的地时，应该按照如下选择下一跳继承者：

a) 本规格只阐述了怎样从 DODAG 版本中选择一个继承者，且该继承者是与转发数据包 IPv6 头标记的 RPL 实例标识匹配的。实例之外的路由能够被完成，只要额外的规则被放进例如实例的详细顺序中和路由协议中，以预防环路产生。这样的规则可能会在专门的文档中定义；

b) 如果一个本地管理优先支持一个从不同路由协议学习来的路由，而不是 RPL，那么使用这个继承者；

c) 如果数据包头通过包含一个 RH4 头指定了一个源路由，那么使用该路由。如果节点使用指定的源路由转发数据包失败，那么该数据包会被丢弃。节点可能记录一个错误。节点可能在源路由头消息中发送一条 ICMPv6 错误给数据包的源头；

d) 如果路由表中有一个条目与从多播目的地通告中学习来的目的地是匹配的（即目的地是一跳邻居），那么使用那个继承者；

e) 如果路由表中有一个条目与从单播目的地通告中学习来的目的地是匹配的（即目的地是位于向下的子 DODAG），那么使用那个继承者。如果存在 DAO 路径控制位与多播继承者相关联，那么当选择时，查询路径控制位以优先级整理继承者。如果，对于一个给定的 DAO 路径控制位，多播继承者被记录成已经声明了该位，那么最近声明该位的继承者应该是首选的；

f) 如果存在一个 DODAG 版本提供了一条与前缀目的地匹配的路由，那么根据 OF 和路由权值从这些 DODAG 父节点中选择一个作为继承者；

g) 当没有更好的匹配时，任何其他 as-yet-unattempted 的 DODAG 父节点可能被选择，作为下一个尝试转发单播数据包；

h) 最终数据包被丢弃。ICMP 目的地不可到达可能被呼叫（不一致性被检测到）。

当转发数据包时跳数限制必须递减。

所选择的继承者一定不能是邻居，且该邻居是数据包的祖先（水平分离），除非在这种情况下：它是用于数据包从向上到向下方向的改变，依照进行改变时节点的路由表决定的，例如从 DIO 路由到 DAO 路由转变时，目的地是相近的，以便能继续进行转发到目的地。

12.2 环路避免和检测

RPL 环路避免机制要保持简单并被设计成最小化搅动和状态。环路可能因为许多原因而形成，例如控制数据包的丢失。RPL 包含一个反应的环路检测技术用于避免崩溃和触发损坏路径的修复。

RPL 循环检测使用数据包内传输的 RPL 数据包信息，依靠一个如 IETF RFC6553 规定的外部机制，该机制位于 IPv6 逐跳选项头中的 RPL 数据包信息中。

RPL 数据包信息内容的定义如下：

Down ‘O’：1 位标志位表明数据包是否被期望向上传输还是向下传输。路由器设置 ‘O’ 标志位当数据包期望被向下传输时（使用 DAO 路由），而当向 DODAG 根转发时清除该位（到一个 RANK 较低的节点）。一个主机或者 RPL 叶节点必须要设置 ‘O’ 标志位为 0。

Rank-error ‘R’：1 位标志位表明一个序列错误是否被检测到。当相对的序列和 ‘O’ 位中表明的方向不协调时，序列错误会被检测到。一个主机或者 RPL 叶节点必须要设置 ‘O’ 标志位为 0。

Forwarding-error ‘F’：1 位标志位表明这个节点不能再进一步向目的地转发数据包。‘F’ 位应该被这样一个子节点来设置，该子节点没有一条到目的地的路由且数据包 ‘O’ 位被设置。一个主机或者 RPL 叶节点必须要设置 ‘F’ 位为 0。

RPL 实例标识：8 位字段表明伴随数据包发送的 DODAG 实例。

SenderRank：16 位字段，源节点设为 0，前向 RPL 网络中的路由器设为 DAGRank (rank)。

12.2.1 源节点操作

如果源节点知道数据包首选的 RPL 实例标识，那么它必须相应地设置与数据包相关联的 RPL 实例标识字段，否则它必须设置它为 RPL_DEFAULT_INSTANCE。

12.2.2 路由器操作

12.2.2.1 转发实例

RPL 实例标识是与数据包源相关联的。RPL 实例标识必须与任何节点、主机或者路由器上的数据包之上的 RPL 实例相匹配。RPL 实例标识是 RPL 数据包信息中的一部分。

RPL 网络中一个 RPL 路由器转发一个数据包一定要检查这个数据包是否包含 RPL 数据包信息。如果没有，那么 RPL 路由器必须要插入一个 RPL 数据包信息。如果路由器是一个进口路由器，其注射数据包到 RPL 网络中，路由器必须要设置 RPL 数据包信息中的 RPL 实例标识字段。路由器如何决定 RPL 实例标识的映射的细节超出了本标准的范围。

一个路由器向 RPL 网络外部转发数据包一定要移除 RPL 数据包信息。

当一个路由器接收了一个指定了 RPL 实例标识的数据包，且节点能够沿着与实例关联的 DODAG 转发该数据包，那么路由器必须如此做并且使 RPL 实例标识值不变。

如果任何节点不能沿着与 RPL 实例标识关联的 DODAG 转发数据包，那么节点应该丢弃该数据包并发送一个 ICMP 错误消息。

12.2.2.2 DAG 不一致性环路检测

如果数据包的方向与序列关系不匹配，DODAG 是不一致的。接受者如果收到了如下的数据包就被认为是检测到不一致性：

‘O’ 位被设置（向下）并来自一个高序列的节点；

‘O’ 位被清除（向上）并来自一个低序列的节点。

当 DODAG 根增加 DODAGVersionNumber 时，一个临时的序列中断可能会在下一个 DODAG 版本和前一个 DODAG 版本之间形成，尤其是如果节点在下一个 DODAG 版本中正在调整它们的序列并推迟它们的迁移到下一个 DODAG 版本中去。一个路由器如果仍然是前一个 DODAG 版本中的成员可能会转

发数据包到下一个 DODAG 版本中的父节点。某种情况下这会导致父节点由于前一个 DODAG 版本的序列顺序未必与下一个 DODAG 版本一样，检测出不一致性，并且数据包可能被判定成不会被转发处理。如果发送方路由器意识到所选择的继承者已经加入了下一个 DODAG 版本，那么发送方路由器必须更新 SenderRank 的值为 INFINITE_RANK，因为它会跨越不连续性转发数据包到下一个 DODAG 版本，以避免一个错误的序列不一致性检测。

路径上的一个不一致性错误不会被认为是严重错误，数据包会继续传输。但是路径上的同一个数据包的检测不应该发生，否则数据包一定要被丢弃。

这一过程是由关联数据包的 Rank-Error 位控制的。当数据包中检测到不一致性时，如果 Rank-Error 位没有被设置那么设置它。如果已经设置了那么必须丢弃该数据包，Trickle 定时器必须要重启。

12.2.2.3 DAO 不一致性检测和恢复

DAO 不一致性环路恢复机制只应用在存储模式中。

在非存储模式中，数据包是源路由到目的地的且 DAO 不一致性不是本地校对的。反而，一条带有新代码“源路由头错误”的 ICMP 错误消息会被发还给根节点。“源路由头错误”消息与“目的地不可到达消息”格式一样。调用 ICMP 消息中被送回的数据包的那部分，至少应该记录到路由头，路由报头应该由这个节点消耗，使 IPv6 报头中的目的地是下一跳节点，该节点为不可达到。

当一个路由器有一条子节点从之前 DAO 消息中学习来的向下路由，但是该路由在子节点中不再有效（例如子节点中相关状态已经被清除），这时会发生 DAO 不一致性。使用 DAO 不一致性环路恢复，一个数据包能习惯于递归地探测和清除 DODAG 子集中废弃的 DODAG 状态。

通常的习惯，一个数据包向下传输绝不会再向上传输。如果使用了 DAO 不一致性环路恢复，那么路由器应该把 Forwarding-error ‘F’ 位被设置且 ‘O’ 位保持不变的数据包返还给父节点，否则路由器必须丢掉该包。

一旦接收到一个 Forwarding-error 位被设置的数据包，节点必须移除导致转发到那个邻居的路由状态，清除 Forwarding-error 位并尝试重传。数据包可能会被发送到其他邻居，在一个用户可配置的具体执行定时器过期之后。如果这个邻居仍然有一个不一致的 DAO 状态经过这个节点，那么过程会递归，这个节点会设置 Forwarding-error ‘F’ 位同时该邻居的路由状态也会被清除。

13 多播操作

本章规定了 IPv6 RPL 网络中多播路由操作，和单播 DAO 消息如何被用来转播组的注册。DODAG 构建方式可以用来转发单播和多播流量。多播群组注册使用和单播相同的 DAO 消息，只是地址的类型不一样。最主要的区别是，多播流会复制给所有注册到多播群组的子节点，而单播流只传给一个子节点。支持 RPL 存储模式的节点应当支持上述 DAO 多播操作，RPL 非存储模式节点不必考虑本章的内容。

多播操作由 DIO 消息中 MOP 字段控制：

- a) 如果 MOP 字段需求多播支持，那么节点必须以路由器的身份加入 RPL 网络，以满足在 RPL 网络中多播和转发的操作。节点不被要求支持多播操作时，以叶节点的身份加入即可；
- b) 如果 MOP 字段没有需求多播支持，那么由其他方式解决。

路由器节点可能选择传递一条监听者注册 DAO 消息给它的预置父节点；或者，路由器节点可能选择复制额外的父节点作为 DAO 消息通告的单播目的地。

因此，从监听者到 DODAG 根节点之间路径上的每个路由器节点路由状态，允许根节点复制一个多

播数据包给它的所有路由器子节点，这些节点已经转发了包含该多播群组目标选项的 DAO 消息。

14 路由邻接的维护

继承者的选择，沿着预设路径向上沿着 DODAG，或者沿着从目的地通告学习来路径向下沿着 DODAG，导致路由邻接格式需要维护。

在 IGP 例如 OSPF 或者 IS-IS 中，一个路由邻接的维护包含保活机制（hello 封包）或者其他协议的使用，例如双向转发检测和 MANET 的邻居发现协议（NHDP）。不幸的是，这样一种主动的方法在受限的环境中经常是令人不满意的，这种环境下，根据数据通信流量会导致过多的控制流量并对链路负载和节点资源都会造成消极的影响。

与这些路由协议相比，RPL 没有定义任何保活机制来检测路由邻接的失败：这是因为在很多的情况下，这样一种机制在带宽，甚至重要的能量的消耗都是昂贵的（电池驱动设备支付不起定期发送保活机制）。RPL 仍然需要一种额外的机制来检测一个邻居是不可到达的。这样一种机制应该较好的反应流量，以减小维持路由邻接的开销并集中研究实际使用的链接。

可以被使用的反应性机制的例子有：邻居不可到达检测机制。

15 目标函数的指导方针

目标函数，连同路由度量和约束共同决定选择哪个 DODAG 来加入，并且该 DODAG 中大量的同辈作为父节点。目标函数用来计算父节点的排序列表。目标函数也用来计算 DODAG 版本中设备的序列。

目标函数在 DIO 消息中使用一个目标代码点（OCP）来声明，并且声明了构建 DODAG 所必须使用的方法。目标代码点在 IETF RFC6552 以及其他相关规范中规定。

15.1 目标函数行为

目标函数被期望在一个节点上遵循同样的行为：

a) 每一次当有事件表明一个潜在的下一跳节点信息被更新时，就要触发父节点选择。这可能在接收一条 DIO 消息发生，当一个定时器完毕，所有 DODAG 父节点都不可用，或者有一个事件表明候选邻居的状态已经改变。

b) OF 扫描节点上的所有接口。虽然在大多数应用场景下通常仅有一个接口，有可能是它们中的多个并且一个接口可能被配置成支持 RPL 操作。一个接口同样能够被配置成首选项，或者通过可能依赖于链路层的试探法能动态的被学习到。最后一个接口可能匹配一个目标函数所需的标准，例如保护的程
度。结果，一些接口可能完全被排除到计算之外，例如那些不能满足某些通告约束的接口，那么剩下的接口可能会被优先选择。

c) OF 在可能的接口上扫描所有候选邻居，以检验它们是否可以充当 DODAG 内的一个路由器。有可能是它们中的多个，并且一个候选邻居要在被使用前需要通过一些确认试验。特别的一些链路层需要与路由器的一些行动上的经验，以使路由器成为下一跳节点。

d) 目标函数为了比较节点而计算节点的序列，通过增加到候选者的序列值，该值代表了 DODAG 版本中节点和候选者的相关位置。

——序列增加的值必须至少为 MinHopRankIncrease；

——为了保持环路避免和度量最佳化成一条直线，序列的增量应该反映度量值的任何增量。例如，用一个纯粹地附加的度量例如 ETX，序列的增量能够与度量的增量相匹配；

——导致节点序列增加的候选邻居不会被考虑成为父节点。

e) 通告的 OF 与方针函数中指定的 OF 集不相容, 这样的候选邻居会被忽略。

f) 因为 OF 扫描所有候选邻居, OF 保持当前最好的父节点并与当前的候选邻居比较能力。OF 定义了大量的关键测试以达到目标。路由器之间的测试决定了顺序关系。

——如果路由器之间是平等的关系, 那么下一个测试将会尝试在路由器之间进行;

——否则 2 个路由器中最好的那个成为当前最好的父节点并继续扫描下一个候选邻居;

——有些目标函数可能包含一个测试, 如果节点加入了任何一个路由器, 用来比较产生的 RANK。

g) 当扫描完成时, 首选父节点被确定并且节点的序列被计算作为父节点的序列, 以加强父节点的序列。

h) 可能会需要额外的扫描来选择用于替代的父节点。在下一个扫描中:

——不在同一个 DODAG 中的候选邻居要被忽略掉;

——序列比节点大的候选邻居要被忽略掉;

——序列与节点相等的候选邻居作为父节点选择时要被忽略掉;

——序列小于节点的候选邻居作为首选。

16 邻居发现互操作的建议

本标准直接从 IPv6 ND 消息中借用前缀信息选项 (PIO) 和路由信息选项 (RIO)。可以想象, 以此为基础的将来的规范, 可能有额外的原因影响 IPv6 ND 消息的部分。本章节为未来的规范提供了一些建议。

首先 RPL 是一个路由协议。当 RPL 和 ND 之间进行映射功能时应该竭尽全力维持架构。RPL 只支持路由。那就是说, 可能存在有效的技术方面原因, 在一个特定的实现/部署中以允许 RPL 和 IPv6 ND 之间共享选项。

通常下列的指导方针应用于:

a) RPL 消息代码必须从 RPL 控制消息选项注册表中分配;

b) RPL 长度字段必须以单字节长度单元表示, 与 ND 长度字段以 8 字节长度单元表示不同;

c) RPL 选项通常不需要 8 字节边界对齐;

d) 当映射/传输一个 IPv6 ND 选项重新分配作为一个 RPL 选项时, 任何填充字节如果可能的话应该被移除。例如, PIO 消息中的前缀长度字段足够用来描述前缀字段的长度。当映射/传输一个 RPL 选项重新分配作为一个 IPv6 ND 选项时, 任何填充字节应该被恢复。这一个过程必须要清楚。

17 互操作性的需求摘要

本章总结了 RPL 三种主要实施操作模式的基本的互操作性和引用规范文本。支持的三种模式是没有下行路由, 非存储模式和存储模式。

实施操作中为了适应应用场景需要遵循本标准不同能力需求子集。对于实施者来说, 支持应用场景的相应的互操作性需求级别是非常重要的。

17.1 通用需求

通常情况下, 达成最高级别的互操作性需求级别的条件是, RPL 网络中的所有节点使用相同的 MOP、目标函数、度量和约束条件协同运行, 并能担当 RPL 路由器。如果一个节点不能担当一个 RPL 路由器时, 它会被当成一个叶节点以更受约束的方式运行。

所有的RPL实现需要支持传输RPL数据包。RPL实现需要支持邻居不可到达探测机制或等效机制，确保邻居RPL节点的可到达性。本标准提供了一种获得PIO的方法，从而生成IPv6地址。当使用该方法时，需要执行地址解析和地址复制，例如IPv6邻居发现和6LoWPAN邻居发现。

17.2 作为 RPL 叶节点的操作（唯一）

- a) 充当一个叶节点的实现不再担当一个RPL路由器。具体细节参见9.5节。
- b) 不需要支持特别的MOP编码，尽管当叶节点发送DAO消息建立向下路由时，需要与MOP表示的操作模式一致。
- c) 不需要支持特别的OF。
- d) 总之，叶节点通常不发布DIO消息，可能发布DAO和DIS消息。叶节点接收DIO消息，通常忽略DAO和DIS消息。

17.3 作为 RPL 路由的操作

如果无法使用下面的指导，那么一个RPL路由器实现必须至少支持无度量的目标函数0。一个RPL路由器实现需要支持DODAG中使用的MOP。所有的RPL路由器需要实现Trickle计时器。

17.3.1 仅支持向上路由

仅支持向上路由的RPL路由器需要支持：

- a) 向上路由；
- b) MOP编码为0；
- c) 不发布DAO消息，可能发布DIO和DIS消息。节点接收DIO和DIS消息，通常忽略DAO消息。

17.3.2 非存储模式中支持向上和向下路由

非存储模式中支持向上路由和向下路由的RPL路由器需要支持：

- a) 向上路由；
- b) 向下路由；
- c) MOP编码值为2；
- d) 发布DAO、DIO和DIS消息。节点接收DAO、DIO和DIS消息。此种情况不支持多播。

17.3.2.1 可选的支持基本多播机制

一个存储模式节点可以通过以下操作支持基本多播操作：

- a) 基本多播支持；
- b) MOP编码为3。

17.4 规范将来的考虑的内容

以下内容在将来规范内考虑：

- a) 如何附加一个非RPL节点，例如一个IPv6主机，分配至少一个PIO信息到RPL节点；
- b) 如何获得已认证模式中认证信息；
- c) 细化多重实例的同步操作；
- d) 高级配置机制，例如RPL实例标识，目标函数的参数化以及控制安全参数。

18 RPL 常量和变量

下面是 RPL 常量和变量的摘要：

BASE_RANK：可能用于协调多重根的虚拟根的 rank。值为 0。

ROOT_RANK: DODAG 根的 rank。值为 MinHopRankIncrease (DODAG 根通告), DAGRank (ROOT_RANK) 值为 1。

INFINITE_RANK: rank 的最大常量。值为 0xFFFF。

RPL_DEFAULT_INSTANCE: 本协议中被节点使用不带有任何重置策略的 RPL 实例标识, 值为 0。

DEFAULT_PATH_CONTROL_SIZE: 用来配置 DODAG 配置选项中的 PCS 的默认值, 用来表明传输信息选项中路径控制字段的有效位的数量。值为 0。这样配置最简单的实例限制 fan-out 值为 1, 并限制节点只能发送一条 DAO 消息给一个父节点。

DEFAULT_DIO_INTERVAL_MIN : 用来配置 DIO trickle 定时器中 Imin 的默认值。值为 3。这个配置导致 Imin=8ms。

DEFAULT_DIO_INTERVAL_DOUBLINGS : 用来配置 DIO trickle 定时器中 Imax 的默认值。值为 20。这个配置导致最大时间间隔为 2.3h。

DEFAULT_DIO_REDUNDANCY_CONSTANT: 用来配置 DIO trickle 定时器中 k 的默认值。值为 10。这个配置是 trickle 抑制机制的一个保守值。

DEFAULT_MIN_HOP_RANK_INCREASE: MinHopRankIncrease 的默认值。值为 256。这个配置导致 rank 为 8 位宽度整数部分。

DEFAULT_DAO_DELAY: DELAYDAO 定时器的默认值。值为 1s。

DIO TIMER: 节点所属的每个 DODAG 的一个实例。到期触发 DIO 消息传输。Trickle 定时器的可变时间间隔 $[0, \text{DIOIntervalMin} \dots 2^{\text{DIOIntervalDoublings}]$ 。

DAG Version Increment Timer: 取决于作为 DODAG 根节点所属每个 DODAG 的一个实例。可能不是所有的实例都支持。到期触发 DODAGVersionNumber 的增加, 导致一系列新的 DIO 消息的发送。时间间隔的选择应该适应于 DODAG 的传播时间并且要视应用的需求而定 (例如, 响应时间 对比开销)。

DelayDAO Timer: 取决于每个 DODAG 中每个 DAO 父节点 (选择用来接收目的地通告的 DODAG 父节点的子集) 的一个定时器。到期触发发送一条 DAO 消息给 DAO 父节点。

RemoveTimer: 取决于每个邻居的每个 DAO 条目的一个定时器。(即给定的 DAO 消息到这个节点的邻居们被看做是一个 DODAG 父节点)。过期可能触发无路径通告, 或者如果没有 DAO 父节点的迅速解除分配 DAO 条目。

19 可管理性的注意事项

本章的目标是考虑 RPL 的可管理性, 以及在低功耗有损网络中 RPL 将如何操作。包括: 配置、监视、错误管理、统计和根据 IETF RFC 5706 中提出的推荐协议的性能。

19.1 简介

大多数现有的 IETF 管理标准是由以管理信息为基础的数据模型构成的, 用来监视和管理网络设备。

对于许多协议, IETF 组织已经使用了 IETF 标准管理框架, 包括简单网络管理协议、管理信息的结构和 MIB 数据模型来管理新的协议。

在经营和管理方面的通用策略已经扩展到一个更加开放的地步, 是一组工具和管理协议, 而不是单一的依赖一种协议, 例如 SNMP。

2003 年, IAB (互联网架构理事会) 召开了一次网络管理方面的研讨会, 来讨论一些 IETF 网络管理协议的优势和劣势, 并与操作需求相对照, 尤其是配置方面。

讨论的一个问题是 SNMP 的二进制格式的用户不友好。对于低功耗网络情况，必须注意的是，CORE 工作组在撰写时，正积极地致力于低功耗网络设备的资源管理。不过，本章节对 RPL 如何被部署，操作和管理的做出指导。

如 IETF RFC 5706 中阐述，“一个管理信息模型的讨论应该包括什么是易管理的，协议的哪一方面需要被配置，什么类型的操作是被允许的，什么特定协议事件应该发生，哪一个事件能被计数，和哪一个事件应该被操作者通知到”。这些方面都会在下文的章节详细讨论。

RPL 将会被用于各种设备，这些设备拥有从几个 kB 到几百 kB 甚至 MB 的内存资源。当内存被高度地限制时，满足本章节列出的所有需求是不可能的。不过值得以详尽的方式列出所有这些，并且实施者将根据设备的可用资源决定那些需求可以被满足。

19.2 配置管理

本章节讨论配置管理，列出有关配置管理的协议参数。

一些 RPL 参数是可选的。配置的需要只应用于所使用的选项。

19.2.1 初始化模式

IETF RFC 1958 “(互联网架构原则)]” 的 3.8 节中声明：“尽可能的避免选项和参数。任何选项和参数应该被动态地配置或者协商而不是手动地”。尤其是在低功耗网络中，设备的数量非常多的情况下，手工配置是不能实现的。这一点已经在 RPL 的设计中考虑到，据此，DODAG 根节点为加入 DODAG 的设备提供了许多的参数，以避免路由器上笨重的配置和潜在源的错误配置（例如 trickle 定时器的值）。但是还是有额外的 RPL 参数应该被一个 RPL 实现允许配置。

19.2.1.1 启动上的 DIS 操作方式

当一个节点首次启动时：

a) 节点可能决定保持沉默，等待接收所感兴趣的 DODAG 中的 DIO 消息（通告一个支持的 OF 和度量/约束），并且直到加入一个 DODAG 前它都不发送任何多播 DIO 消息。

b) 节点可能决定发送一条或多条 DIS 消息（对于特定的 DODAG 的可选择的请求 DIO 消息）作为对附近 DODAG 的初始探测，并在经过可配置的一段时间后缺少 DIO 消息作为答复，节点可能决定根源于一个浮动的 DODAG 并开始发送多播 DIO 消息。

一个 RPL 实现应该允许配置上面列出的首选操作模式，并连同所必要的参数（第二种模式：DIS 消息的数量和相关的定时器）

19.2.2 DIO 和 DAO 基本消息和可选项配置

RPL 规定了大量的协议参数来考虑将来会使用到 RPL 的广泛应用。那就是说，对于那些必须要配置到每个路由器的参数数量的限制，已经给予了特殊的关注。相反，可以使用许多默认值，当所需求的这些参数能够被 DODAG 根提供时，那么允许进行动态参数设置。

一个 RPL 实现应该允许配置下面的路由协议参数。如上面所指出的，注意到许多组参数是在 DODAG 根中配置的。

19.2.3 配置到低功耗网络每个路由器的协议参数

一个 RPL 实例必须允许配置下面的 RPL 参数：

a) RPL 实例标识 [DIO 消息中] 虽然 RPL 实例标识必须在 DODAG 根上进行配置，但是同样要在每一个节点上作为一个策略进行配置，以便决定该节点是否应该加入一个特定 DODAG。注意第二个 RPL

实例标识能配置到节点上，它应该成为浮动 DODAG 的根；

b) 所支持的目标代码点列表；

c) 所支持的度量表：IETF RFC6551 规定了一些用于 DODAG 信息的度量和约束。因此一个 RPL 实现应该允许配置一个节点能接受和理解的度量列表。如果接收到一条带有不被理解或支持的度量和/或约束的 DIO 消息，节点将会作为一个叶节点加入；

d) 前缀信息，连同有效和首选的生存时间以及 L 和 A 标志位。[DIO 消息，前缀信息选项]。一个 RPL 实现应该允许配置，如果 DIO 消息中一定带有前缀信息，以便为自动配置分配前缀信息。那种情况下，一个 RPL 实现必须允许在前缀信息选项中连同相应标志位通告一个前缀列表；

e) 请求信息[DIS 消息，请求信息选项中]。注意当这种消息应当被发送和在某种特定情况下，一个 RPL 实现应该允许配置连同 RPL 实例标识，V/I/D 标志位的值；

f) ‘K’ 标志位：当一个节点应该在一条 DAO 消息中设置 ‘K’ 标志时；

g) 操作模式[DIO 基本消息中]；

h) 路由信息（和优先权）[DIO 消息路由信息选项]。

19.2.4 低功耗网络中每个非 DODAG 根路由器将要配置的协议参数

一个 RPL 实现必须允许配置目标前缀[DAO 消息，RPL 目标选项中]

此外，有些情况下一个节点可能要指定一个目标以便允许对目标进行特定的处理（优先次序等）。当使用时，一个 RPL 实现应该允许在每一个目标基础上配置目标描述符号（例如使用访问列表）。

一个节点的 DODAG 父集为空那么该节点可能成为一个浮动 DODAG 的 DODAG 根节点。同时它会设置它的 DAGPreference 为次优先级。因此在这种情况下，一个 RPL 实现必须允许配置节点应该初始化的一系列动作：

a) 启动它自己的（浮动的）DODAG：除了它的 DAGPreference 必须还要配置新的 DODAGID；

b) Poison 损坏的路径；

c) 触发一个本地修复。

19.2.5 将要配置到 DODAG 根的参数

此外，其他几个参数仅仅只配置在 DODAG 根和 DIO 消息携带的选项中通告。

如 9.3 节中规定，一个 RPL 实现利用 trickle 定时器来管理 DIO 消息的发送。Trickle 算法的运算是由一组可配置参数决定的，这些参数必须是可配置的，并且由 DODAG 根节点沿着 DODAG 在 DIO 消息中通告。

a) DIOIntervalDoublings [DIO 消息，DODAG 配置选项中]；

b) DIOIntervalMin [DIO 消息，DODAG 配置选项中]；

c) DIORedundancyConstant [DIO 消息，DODAG 配置选项中]。

此外，一个 RPL 实现应该考虑到配置下面几组 RPL 参数：

a) Path Control Size [DIO 消息，DODAG 配置选项中]；

b) MinHopRankIncrease [DIO 消息，DODAG 配置选项中]；

c) The DODAGPreference field [DIO 消息，DIO 基本对象]；

d) DODAGID[DIO 消息，DIO 基本选项]和[DIO 消息，当 DAO 消息中 ‘D’ 标志位被设置时]。

DAG 根的行为：某些情况下，如果一个节点不能加入一个固定的 DODAG，那么该节点可能不希望

永久地作为一个浮动的 DODAG 根节点来活动。例如一个电池供电节点可能不希望长时间地作为一个浮动 DODAG 根节点。因此，一个 RPL 实现可以支持配置一个节点在一段配置的时间内是否作为一个浮动的 DODAG 根的能力。

DAG 版本号增量：一个 RPL 实现可能允许在 DODAG 根配置通过更新 DODAGVersionNumber 来更新 DODAG 的状态。一个 RPL 实现应该允许 DODAG 根是否使用配置周期的或者时间触发机制来控制 DODAGVersionNumber 的变化（如 4.2.2 中，会触发一个全局的修复）。

19.2.6 与以 DAO 消息机制相关 RPL 参数的配置

DAO 消息为可选并用需要向下路由操作的 DODAG 中。本节阐述与 DAO 消息相关的参数并且提供它们配置的建议。

如 10.5 节中建议延时向 DAO 父节点发送 DAO 消息以便最大化完成路由聚合的机会。一旦收到一条 DAO 消息，节点就应该启动一个 DelayDAO 定时器。默认值为 DEFAULT_DAO_DELAY。一个 RPL 实现可以考虑到配置 DelayDAO 定时器。

存储模式中，一个存储节点可以增加 DTSN 以便从它最近的子节点可靠地触发一组 DAO 更新，作为路由表更新和维护的一部分。一个 RPL 实现可以考虑到配置一组规则来说明 DTSN 增加的触发（手册或者基于事件的）。

当一个 DAO 条目超时或者无效时，节点应该做适当的尝试来报告 No-path 到它的每一个 DAO 父节点。尝试的次数可以是可配置的。

一个实现应该支持发送 DAO 消息速率受限。相关参数可以是可配置的。

19.2.7 与安全机制相关的 RPL 参数的配置

如本标准中 11 章描述的安全特性是可通过选择来实现的，并且一个给定的实现支持所描述安全特性的一部分（包括空集）。

为此一个实现支持所描述的安全特性可以概念地实现一个安全策略数据库。为了支持这种安全机制，一个 RPL 实现应该考虑到配置下面参数的一个子集：

- a) Security Modes accepted [Unsecured mode, Pre-Installed mode, Authenticated mode];
- b) KIM values accepted [Secure RPL Control messages, in Security Section];
- c) Level values accepted [Secure RPL Control messages, in Security section];
- d) Algorithm values accepted [Secure RPL Control messages, in Security section];
- e) Key material in support of Authenticated or Pre-Installed key modes.

此外，一个 RPL 实现应该考虑使用下面参数的一个子集来配置一个 DODAG 根：

- a) Level values advertised [Secure DIO message, in Security Section];
- b) KIM value advertised [Secure DIO message, in Security Section];
- c) Algorithm value advertised [Secure DIO message, in Security Section]。

19.2.8 默认值

本标准规定了下面这组 RPL 变量的默认值：

DEFAULT_PATH_CONTROL_SIZE

DEFAULT_DIO_INTERVAL_MIN

DEFAULT_DIO_INTERVAL_DOUBLINGS

DEFAULT_DIO_REDUNDANCY_CONSTANT

DEFAULT_MIN_HOP_RANK_INCREASE

DEFAULT_DAO_DELAY

建议在协议中指定默认值；在 IETF RFC 5706 中，默认值的意义可能越来越小。RPL 为路由协议，被期望用于许多环境，该环境网络特性例如节点数量，链路和节点类型变化显著。因此，这些默认值可能会随着环境和技术的演进而改变。的确，低功耗有损网络相关技术（例如硬件，链路层）在过去的几年内已经戏剧性的演进发展，并且这些技术在未来的时间内都被认为会改变和演进的。

所建议的值不是基于广泛的最佳当前做法，且被认为是保守的。

19.3 RPL 监视操作

一些 RPL 参数应该被监测来核实路由协议和网络本身的正确操作。本章节列出了感兴趣的几组监测参数。

19.3.1 监视 DODAG 参数

一个 RPL 实现提供下面参数的信息：

- a) DODAG Version number [DIO message, in DIO base message];
- b) Status of the G flag [DIO message, in DIO base message]
- c) Status of the MOP field [DIO message, in DIO base message]
- d) Value of the DTSN [DIO message, in DIO base message]
- e) Value of the rank [DIO message, in DIO base message]
- f) DAOSequence: Incremented at each unique DAO message, echoed in the DAO-ACK message [DAO and DAO-ACK messages]
- g) Route Information [DIO message, Route Information option] (list of IPv6 prefixes per parent along with lifetime and preference)
- h) Trickle parameters:
 - 1) DIOIntervalDoublings [DIO message, in DODAG configuration option]
 - 2) DIOIntervalMin [DIO message, in DODAG configuration option]
 - 3) DIORedundancyConstant [DIO message, in DODAG configuration option]
- i) Path Control Size [DIO message, in DODAG configuration option]
- j) MinHopRankIncrease [DIO message, in DODAG configuration option]只有在 DODAG 根上可以监视到的值
- k) Transit Information [DAO, Transit Information option]: 一个 RPL 实现应该允许配置所接收到的传输信息选项是否应该被显示在 DODAG 根上。这种情况下，所接收的传输信息的 RPL 数据库应当同时包含：路径序列，路径控制，路径生命期和父节点地址。

19.3.2 监测 DODAG 不一致性和环路检测

DODAG 不一致性的检测在 RPL 网络中非常关键。因此推荐一个 RPL 实现要提供合适的监测工具。一个 RPL 实现应当提供计数器来报告节点已经检测到相关 DODAG 父节点的不一致性次数，例如，如果 DODAGID 已经改变。

可能的时候更多有关不一致性检测的细微信息都应该被提供。一个 RPL 实现可以提供计数器来报告

下面不一致性的数量:

- a) 从高序列节点接收的‘0’位被设置的数据包;
- b) 从低序列节点接收的‘0’位被清除的数据包;
- c) ‘F’位被设置的数据包的数量;
- d) ‘R’位被设置的数据包的数量。

19.4 RPL 数据结构的监测

19.4.1 候选邻居数据结构

候选邻居列表中的一个节点有资格成为一个父节点(带有足够高的本地信任)。一个 RPL 实现应该提供一个方法来允许用某种度量来监控候选邻居列表,来反映某种度量测量的本地信任。

一个 RPL 实现可以提供一个计数器来报告一个候选邻居已经被忽略的次数,候选邻居的数量超过最大已认证值。

19.4.2 DODAG 表

对于每一个 DODAG,一个 RPL 实现被期待记录下面的 DODAG 表中的值:

- a) RPLInstanceID;
- b) DODAGID;
- c) DODAGVersionNumber;
- d) Rank;
- e) Objective Code Point;
- f) A set of DODAG Parents;
- g) A set of prefixes offered upwards along the DODAG;
- h) Trickle timers used to govern the sending of DIO messages for the DODAG;
- i) List of DAO parents;
- j) DTSN;
- k) Node status (router versus leaf)。

一个 RPL 实现应该允许监视上面列出的几组参数。

19.4.3 路由表和 DAO 路由条目

一个 RPL 实现维护着几个与 DODAG 和 DAO 条目(存储模式)相关的信息单元。非存储节点的情况下,有限的信息量被维护然而在存储节点的情况下,信息量会随着路由条目的而增加。

一个 RPL 实现应该考虑到下面要监测的参数:

- a) Next Hop (DODAG parent);
- b) Next Hop Interface;
- c) Path metrics value for each DODAG parent。

一个 DAO 路由表条目概念地包含下面的元素(仅仅存储模式)

- a) Advertising Neighbor Information;
- b) IPv6 Address;
- c) Interface ID to which DAO Parents has this entry been reported;
- d) Retry Counter;

- e) Logical equivalent of DAO Content:
- f) DAO-Sequence
- g) Path Sequence
- h) DAO Lifetime
- i) DAO Path Control
- j) Destination Prefix (or Address or Mcast Group)。

一个 RPL 实现应该提供每一条 DAO 路由表条目状态的状态信息。

19.5 错误管理

错误管理是用来解决问题、协议正确操作模式验证、网络设计的重要组成，同时也是网络性能监测的关键要素。一个 RPL 实现应该考虑提供下面有关错误管理的信息：

- a) Memory overflow along with the cause (e.g. routing tables overflow, ...)
- b) Number of times a packet could not be sent to a DODAG parent flagged as valid
- c) Number of times a packet has been received for which the router did not have a corresponding RPLInstanceID

- d) Number of times a local repair procedure was triggered
- e) Number of times a global repair was triggered by the DODAG root
- f) Number of received malformed messages
- g) Number of seconds with packets to forward and no next hop (DODAG parent)
- h) Number of seconds without next hop (DODAG parent)
- i) Number of times a node has joined a DODAG as a leaf because it received a DIO with metric/constraint not understood and it was configured to join as a leaf node in this case.

推荐至少要通过错误日志消息来报告错误。可能会用到其他的协议来报告错误。

19.6 策略

策略规则能够被一个 RPL 实现用来决定一个节点是否被允许加入一个邻居通过 DIO 消息所通告的特定的 DODAG。

本标准规定了在一个单独的 DODAG 中的操作。一个 DODAG 以元组 (RPLInstanceID, DODAGID) 为特性。此外，如上面所说，DIO 消息用来通告其他 DODAG 的特性例如用来建设 DODAG 和 OF 的路由度量和约束。

第一条策略规则由下面的条件组成，且加入一个 DODAG 的 RPL 节点必须满足：

- a) RPLInstanceID
- b) List of supported routing metrics and constraints
- c) Objective Function (OCP values)

一个 RPL 实现必须允许配置这些参数并且应该指定是否节点必须简单地忽略 DIO 消息，如果通告的 DODAG 不符合本地策略，或者是否节点应该以叶节点只要支持的路由度量和约束列表，但是 OF 是不支持的。此外，一个 RPL 实现应该允许 DODAGID 的添加作为策略的一部分。

一个 RPL 实现应该允许配置可选的或者首选的目标函数集，并通过它们的客观的代码点作为一个节点加入一个 DODAG 的参考，并且如果节点的候选邻居都没有通告许可的配置目标函数中的一个，或者

通告的度量/约束是不被支持的, 这时该采取什么行动。这种情况下可以采取两种行动:

- a) 节点以叶节点的身份加入到 DODAG 中
- b) 节点不加入 DODAG 中

低功耗有损网络中的一个节点从不同的路由协议包括 RPL 路由协议中学习路由信息。正是在这种情况下, 通过控制管理优先级选择路由是合适的。一个实现应当允许一个规格化的管理优先级, 为路由学习到的路由协议。

内部数据结构: 一些 RPL 实现可以限制候选邻居列表的大小, 以便限制内存的使用, 这种情况下剩下的其他可行的候选邻居可能会不被考虑并简单地从候选邻居列表中丢弃。

19.7 错误隔离

非常推荐对以不可接受的速率开始发出畸形消息的邻居进行隔离。

19.8 对其他协议的影响

RPL 对其他协议的影响非常有限。凡是在一个路由器上需要多个路由协议的地方, 例如 LBR, 非常期待设备支持在路由协议之间的重新分配功能, 以便允许在两个路由域间的可达性。这种重新分配应该被用户可配置策略的使用而管理。

RPL 被设计成限制控制流量, 是因为使用了 Trickle 定时器机制。因此 RPL 对其他协议的影响应该被极其地限制。

19.9 性能管理

性能管理经常是一个协议的重要方面, RPL 也不例外。IP 性能监视工作组 (IPPM) 已经制定了几个感兴趣的指标:

- a) Number of repairs and time to repair in seconds (average, variance)
- b) Number of times and duration during which a devices could not forward a packet because of a lack of reachable neighbor in its routing table
- c) Monitoring of resources consumption by RPL in terms of bandwidth and required memory
- d) Number of RPL control messages sent and received

19.10 诊断

可能会有一个节点被放置到“繁琐”模式的情况来改善诊断。因此一个 RPL 实现应该提供放置或者拿出一个节点到“繁琐”模式的能力, 以便获得额外的诊断信息。

20 安全考虑

20.1 概述

RPL 网络安全与其他网络安全没有太大差异。它很容易被动的监听攻击甚至在没有物理连接通信时被主动的篡改。Ad hoc 网络结构和成本目标增加了额外的安全约束, 使得 RPL 网络安全面临着非常复杂的环境。RPL 网络中设备成本低廉, 有限的计算和存储能力及功率, 不能保证其有可信任的计算和高质量的随即数发生器。通信不能依赖固网的在线能力, 通信设备之间是短期通信关系。这些约束严格地限制了密码算法和协议的选择, 并因为建立和维护设备之间的信任关系问题影响了安全架构的设计。此外电池寿命和开销严格地限制了有限带宽网络的容忍能力。

本标准的安全机制基于对称密钥和公钥密码并使用上层应用提供的密钥。

规定的安全机制提供下述安全服务:

- a) 数据机密性：确保传输的数据仅能被目的地接收；
- b) 数据可靠性：确保传输数据来源的可靠；
- c) 数据转发保护：确保传输数据被复制时能够检测；
- d) 延时保护：确保传输数据能被及时的接收。

提供的实际保护措施可以应用在每个数据包并允许不同级别的数据可靠性和可选的数据机密性。在数据保护过程和存储一些状态信息过程中通过使用一个非重复值（CCM nonce）来提供数据转发保护，并提供检测机制确认该 CCM nonce 值是否被使用过。此外，在同步时钟松散的设备之间提供延时保护。

密码保护可能在两个对等设备或者一组设备之间共享一个密钥。如果一个组密钥用于点对点通信中，那么保护对于组内有潜在威胁的设备不提供保护。通过使用对称密钥或公钥技术提供数据的机密性。使用公钥技术，对于传输信息的发起者提供有力的保证；然而使用对称密钥技术，仅仅只对密钥共享组中的设备提供数据可靠性保证。因此，在需要更严格的身份认证场景下，基于公钥的身份验证比基于对称密钥的认证技术更有用，例如广播和多播场景。

中 华 人 民 共 和 国
通 信 行 业 标 准
IPv6 路由协议
适用于低功耗有损网络的 IPv6 路由协议(RPL)技术要求
YD/T 2710-2014

*

人民邮电出版社出版发行
北京市丰台区成寿寺路1号邮电出版大厦
邮政编码: 100164
北京康利胶印厂印刷
版权所有 不得翻印

*

开本: 880 × 1230 1/16 2015年9月第1版
印张: 4.25 2015年9月北京第1次印刷
字数: 112千字

15115 • 498

定价: 45元

本书如有印装质量问题, 请与本社联系 电话: (010)81055492