ICS 35.110
L 79

# 中华人民共和国国家标准

GB/T 30269.602—2017

# 信息技术　传感器网络
# 第602部分：信息安全：低速率无线传感器网络网络层和应用支持子层安全规范

**Information technology—Sensor network—Part 602：Information security：Network layer and application support sublayer security specification for low-rate wireless sensor networks**



2017-12-29 发布　　2017-12-29 实施

中华人民共和国国家质量监督检验检疫总局
中国国家标准化管理委员会　发布

# 目　　次

前言 …… Ⅲ
1　范围 …… 1
2　规范性引用文件 …… 1
3　术语和定义 …… 1
4　缩略语 …… 2
5　安全概述 …… 3
5.1　概述 …… 3
5.2　协议栈结构 …… 3
5.3　安全框架 …… 3
5.4　安全密钥 …… 3
5.5　网络层安全 …… 4
5.6　应用支持子层安全 …… 4
6　网络层安全 …… 6
6.1　网络层安全概述 …… 6
6.2　网络层安全服务 …… 6
6.3　帧安全 …… 8
6.4　命令帧 …… 10
6.5　安全相关的 NIB 属性 …… 12
7　应用支持子层安全 …… 14
7.1　应用支持子层安全概述 …… 14
7.2　应用支持子层安全服务 …… 14
7.3　帧安全 …… 38
7.4　命令帧 …… 40
7.5　安全相关的 AIB 属性 …… 50
附录 A（规范性附录）　网络层安全交互过程 …… 52
附录 B（规范性附录）　应用支持子层安全交互过程 …… 55

# 前　言

GB/T 30269《信息技术　传感器网络》拟分为以下部分：

——第 1 部分：参考体系结构和通用技术要求；

——第 2 部分：术语；

——第 301 部分：通信与信息交换：低速无线传感器网络网络层和应用支持子层规范；

——第 302 部分：通信与信息交换：高可靠性无线传感器网络媒体访问控制和物理层规范；

——第 303 部分：通信与信息交换：基于 IP 的无线传感器网络网络层规范；

——第 401 部分：协同信息处理：支撑协同信息处理的服务及接口；

——第 501 部分：标识：传感节点标识符编制规则；

——第 502 部分：标识：传感节点标识符解析规范；

——第 503 部分：标识：传感节点标识符注册规程；

——第 504 部分：标识：传感节点标识符管理规范；

——第 601 部分：信息安全：通用技术规范；

——第 602 部分：信息安全：低速率无线传感器网络网络层和应用支持子层安全规范；

——第 701 部分：传感器接口：信号接口；

——第 702 部分：传感器接口：数据接口；

——第 801 部分：测试：通用要求；

——第 802 部分：测试：低速无线传感器网络媒体访问控制和物理层；

——第 803 部分：测试：低速无线传感器网络网络层和应用支持子层；

——第 804 部分：测试：传感器接口测试规范；

——第 805 部分：测试：传感器网关测试规范；

——第 806 部分：测试：传感节点标识符解析一致性测试技术规范；

——第 807 部分：测试：低速率无线传感器网络网络层和应用支持子层安全测评规范；

——第 901 部分：网关：通用技术要求；

——第 902 部分：网关：远程管理技术要求；

——第 903 部分：网关：逻辑功能接口技术规范；

——第 1001 部分：中间件：传感器网络节点接口。

本部分为 GB/T 30269 的第 602 部分。

本部分按照 GB/T 1.1—2009 给出的规则起草。

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。

本部分由全国信息技术标准化技术委员会(SAC/TC 28)提出并归口。

本部分起草单位：重庆邮电大学、中国电子技术标准化研究院、无锡物联网产业研究院、成都秦川科技发展有限公司、中国信息安全认证中心、山东省计算中心(国家超级计算济南中心)。

本部分主要起草人：王浩、魏旻、陈书义、苏静茹、吴岳飞、甘杰夫、王平、卓兰、汪付强。

# 信息技术　传感器网络
# 第602部分:信息安全:低速率无线传感器网络网络层和应用支持子层安全规范

## 1　范围

GB/T 30269的本部分规定了低速率无线传感器网络网络层和应用支持子层的原语、命令帧格式以及安全交互规程。

本部分适用于低速率传感器网络传输安全的开发设计。

## 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。

GB/T 15629.15—2010　信息技术　系统间远程通信和信息交换　局域网和城域网　特定要求　第15部分:低速无线个域网(WPAN)媒体访问控制和物理层规范

GB/T 25069—2010　信息安全技术　术语

GB/T 30269.2—2013　信息技术　传感器网络　第2部分:术语

GB/T 30269.301—2014　信息技术　传感器网络　第301部分:通信与信息交换:低速无线传感器网络网络层和应用支持子层规范

GB/T 30269.601—2016　信息技术　传感器网络　第601部分:信息安全:通用技术规范

## 3　术语和定义

GB/T 25069—2010、GB/T 30269.2—2013界定的以及下列术语和定义适用于本文件。

3.1

**直接密钥　direct key**

邻居节点之间建立的共享的对密钥。

3.2

**密钥建立　key establishment**

为一个或多个实体产生一个可用的、共享的秘密密钥的过程。

3.3

**密钥管理　key management**

根据安全策略,实施对密钥材料进行产生、登记、认证、注销、分发、安装、存储、归档、撤销、衍生、销毁和恢复的服务。

3.4

**密钥材料　keying material**

确立和维持密码密钥关系所必需的数据(如密钥,初始化值)。

3.5

**密钥更新　key update**

一种机制，通过给同一个组提供另外一个密钥，在两个或多个设备之间实施共享密钥的更换。

3.6

**消息鉴别　message authentication**

消息鉴别是由声明的始发者预期的接收者的验证，并且该消息在转移中未被更改。

3.7

**网络密钥　network key**

全网所有设备共享的密钥。

3.8

**路径密钥　path key**

在没有直接密钥的情况下，在存在多跳安全的节点之间建立的共享的对密钥。

3.9

**安全级别　security level**

有关敏感信息访问的级别划分，以此级别加之安全范畴能更加精细的控制对数据的访问。

3.10

**会话密钥　session key**

为保证一对节点之间的保密通信或消息鉴别而随机产生的密钥。通信完成后，会话密钥失效。

3.11

**共享密钥　shared key**

两个或多个节点之间在初始密钥材料的基础上建立的长期共享的密钥。

3.12

**安全服务　security service**

根据安全策略，为用户提供的某种安全功能及相关的保障。

3.13

**信任中心　trust center**

被网络内设备信任的设备，为了网络和端到端应用配置进行密钥管理或安全数据融合等安全操作。

## 4　缩略语

下列缩略语适用于本文件。

AIB：应用支持层的信息库（Application Support Layer Information base）

APSME：应用支持子层管理实体（Application Support Sublayer Management Entity ）

APSME-SAP：应用支持子层管理实体—服务接入点（Application Support Sublayer Management Entity—Service Access Point）

ASDU：应用支持子层服务数据单元（APS Service Data Unit）

CCM：联合计数模式的 CBC-MAC（Combined CBC-MAC and Counter Mode of Operation）

CCM * ：扩展的 CCM（Extension of CCM）

MAC：媒体访问控制（Medium Access Control）

MIC：消息完整性校验码（Message Integrity Code）

MSDU：MAC 协议子层数据单元（Medium Access Control Sublayer Service Data Unit）

NIB：网络层信息库（Network-Layer Information Base）

NLDE：网络层数据实体（Network-Layer Data Entity）

NLME:网络层管理实体(Network-Layer Management Entity)
NPDU:网络层协议数据单元(Network-Layer Protocol Data Unit)
PAN:个域网(Personal Area Network)
PDU:协议数据单元(Protocol Data Unit)

## 5 安全概述

### 5.1 概述

网络层和应用支持子层提供的安全服务形成了传感网设备内实施安全策略的结构单元。第5章介绍了安全服务规范和如何使用这些服务的功能性描述。

### 5.2 协议栈结构

无线传感器网络协议栈各层及其模块组成如图1所示,虚线框内为无线传感器网络协议栈内安全部分的内容。

协议栈各层的功能应符合GB/T 30269.301—2014的规定。

网络层的安全包括为网络层帧提供安全的处理,以及为邻近高层提供路由安全服务。

应用支持子层的安全包括为应用层帧提供安全的处理,以及为邻近高层提供建立和维护安全关系的服务。

安全服务提供者即为网络层和应用支持子层可执行的安全策略和安全机制。这些安全策略和安全机制为传感器网络的信息传输安全提供支持。



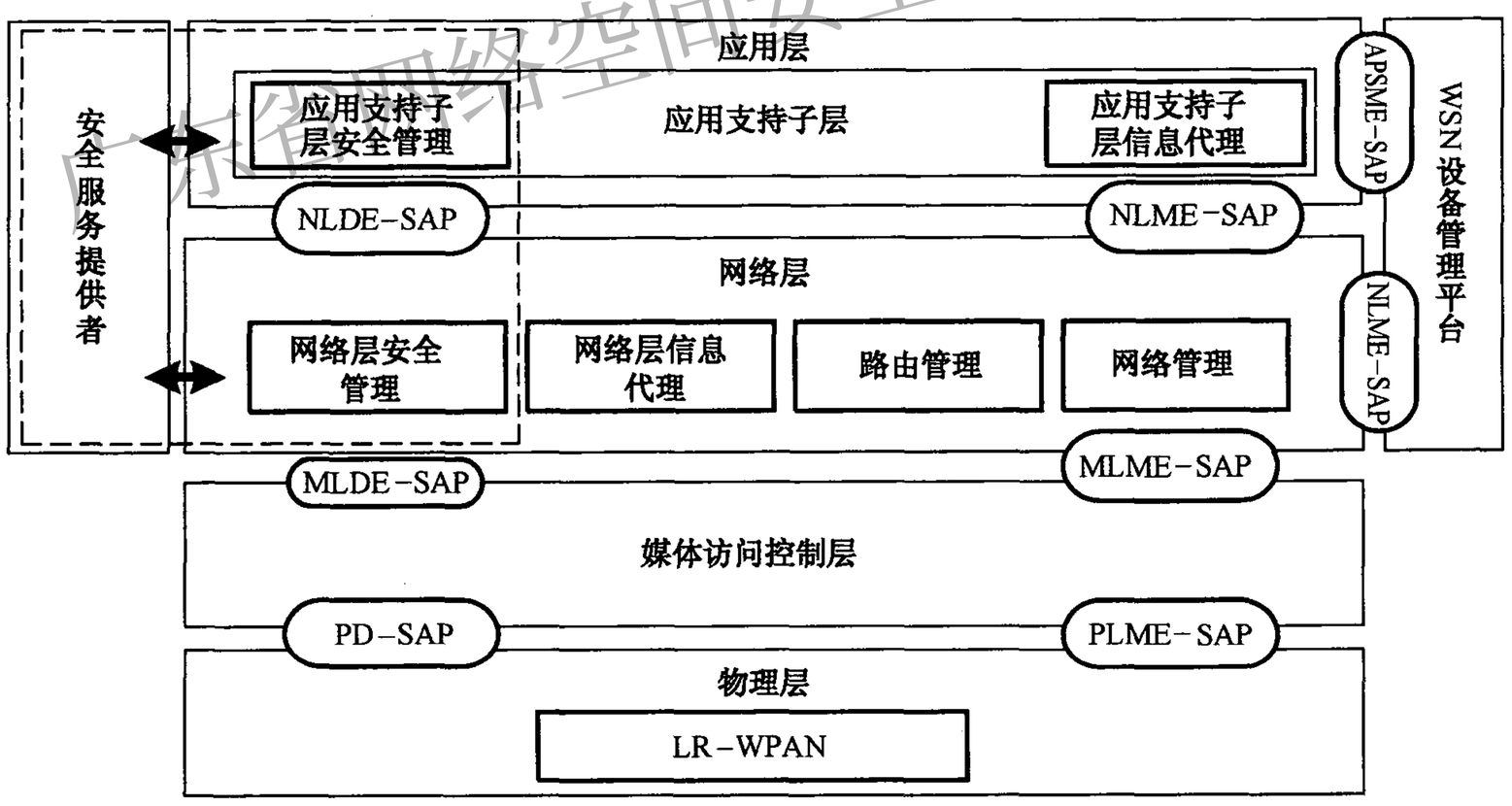


图1 无线传感器网络协议栈结构

### 5.3 安全框架

安全框架包括网络层和应用支持子层两个层次的安全机制。网络层和应用支持子层负责安全传输它们各自的帧。此外,网络层提供了路由安全的服务,应用支持子层提供了建立和维护安全关系的服务。

### 5.4 安全密钥

密钥管理机制涉及到不同类型的密钥,表1列出了本部分所使用的安全密钥。

**表 1　安全密钥**

| 密码机制 | 类型 | | 生成 | 描述 |
|---|---|---|---|---|
| 对称技术 | 密钥材料 | | 由信任中心以安全的方式为节点预分配 | 密钥材料是确立和维持密码密钥关系所必需的数据，密钥材料根据密钥的生成方法不同，可以是节点的 ID 信息等数据或者共享密钥等 |
| | 网络密钥 | | 设备初始化时由信任中心以安全的方式为节点预配置 | 网络密钥由全网设备共享，对输出帧进行安全加密 |
| | 共享密钥 | 直接密钥 | 直接密钥由初始化时写入的初始密钥材料生成 | 直接密钥是由节点与其邻居节点共享的密钥。并通过直接密钥可生成会话密钥 |
| | | 路径密钥 | 在没有直接密钥的情况下，在存在多跳安全路径的节点间建立路径密钥 | 当节点之间没有直接密钥时，通过多跳的安全路径建立的节点之间的共享的密钥 |
| | 会话密钥 | | 两个节点之间通过直接密钥或路径密钥建立会话密钥 | 会话密钥用来对传输的数据进行加密，保证一对节点之间的保密通信或消息鉴别 |

每种密钥从建立到撤销的整个有效期之内，可能会处在多个不同阶段，需要根据具体应用需求对密钥进行维护和更新。不同的密钥类型生存期的长短不同，在同一个密钥材料的有效期内，共享密钥可能撤销和更新多次；在同一个共享密钥的有效期内，会话密钥可能撤销和更新多次。

会话密钥在通信双方节点通信完成后即被销毁。会话密钥的泄露不会影响到共享密钥的安全。若某个共享密钥泄露，则信任中心应将其撤销，并与相关节点交互，分发更新的密钥。共享密钥的泄露并不意味着密钥材料的泄露。但是若某个密钥材料泄露，则相应的共享密钥也必须撤销，并在密钥材料更新后，重新建立共享密钥。

## 5.5　网络层安全

### 5.5.1　假设

当 GB/T 30269.301—2014 中网络层服务原语 NLDE-DATA.request 中的参数 SecurityEnable 置为 FALSE 时，应明确禁止网络层的安全机制。否则，当网络层产生的帧需要加密，或者高层 NIB 中的 nwkSecureAllFrames 属性为 TRUE 时，应启用网络层帧的安全保护机制，网络层的帧保护机制在本部分的附录 A 有说明。此外，网络层还负责为邻近高层提供路由安全的服务。

### 5.5.2　路由安全

网络层的路由安全服务提供了一种在设备之间进行路由选择的安全方式。路由安全机制是以保证网络在受到攻击时仍能进行正确的路由发现、构建和维护为目标的安全机制。

## 5.6　应用支持子层安全

### 5.6.1　概述

当应用层产生的一个帧需要加密，应用支持子层应该负责对它进行安全处理。应用支持子层的帧保护机制在本部分的附录 B 有说明。应用支持子层还负责为邻近高层提供密钥管理、访问控制、鉴别和安全数据融合服务。

### 5.6.2 密钥管理

#### 5.6.2.1 概述

密钥管理机制依赖于基本的密码机制，分为以下两类：

a） 采用对称密码技术的机制；

b） 采用非对称密码技术的机制。

不用的机制适用于不同的应用需求。采用对称密码技术的机制，适用于对安全级别要求较低的传感器网络，例如安全级别在三级以下的传感器网络宜采用基于对称密码技术的机制。而对于安全级别要求较高的传感器网络，则需要采用基于非对称密码技术的机制。

#### 5.6.2.2 密钥分发

应用支持子层的密钥分发服务允许信任中心分发密钥材料到设备。无线传感设备以保护密钥不被泄露的方式写入信任中心分发的密钥材料中。

#### 5.6.2.3 密钥建立

应用支持子层的密钥建立服务允许两个设备之间可以手动建立一个共享密钥，初始信任信息必须在密钥建立之前安装在每个设备上。

#### 5.6.2.4 密钥更新

应用支持子层的密钥更新服务允许一个信任中心去通知其他设备更新密钥。

#### 5.6.2.5 密钥撤销

应用支持子层的密钥撤销服务允许一个信任中心去通知相关设备撤销使用特定密钥信息。密钥的撤销可以是对称密钥的撤销或非对称密钥的撤销。

### 5.6.3 访问控制

应用支持子层的访问控制服务允许控制用户对传感网的节点资源和数据资源的访问。访问控制机制包括：

a） 自主访问控制：

通过访问控制表或访问能力表等策略对用户的访问实施控制。

b） 强制访问控制：

为用户、节点和数据指定敏感标记，通过这些敏感标记对用户的访问实施控制。

### 5.6.4 身份鉴别

应用支持子层的身份鉴别服务允许两个设备之间进行身份鉴别。节点之间的鉴别是基于密码算法的，具有共享密钥的节点之间能够实现相互鉴别。

### 5.6.5 消息鉴别

应用支持子层的消息鉴别服务对设备的广播消息提供了一种安全的鉴别方式。

### 5.6.6 安全数据融合

应用支持子层的安全数据融合服务提供了一个安全的方式进行数据融合。安全数据融合机制是为了保障数据保密性、数据传输安全、数据融合的准确性。

# 6 网络层安全

## 6.1 网络层安全概述

网络层安全规定了输出帧和输入帧的安全传输、路由安全，以及路由安全的服务原语。上层通过建立适当的密钥和帧计数器控制安全处理操作，确定使用哪种安全级别。网络层的 NLDE 和 NLME 见 GB/T 30269.301—2014。

## 6.2 网络层安全服务

### 6.2.1 网络层安全服务原语

网络层安全服务原语如表 2 所示。

**表 2 网络层安全原语**

| NLME 安全原语 | 请求 | 确认 | 指示 | 响应 | 描述 |
|---|---|---|---|---|---|
| NLME-SEC-ROUTE | 6.2.2.1 | 6.2.2.2 | 6.2.2.3 | — | 使用设备鉴别的方法发起路由安全，为节点在设备之间进行路由选择提供一种安全的方式 |

### 6.2.2 路由安全服务

#### 6.2.2.1 NLME-SEC-ROUTE.request

##### 6.2.2.1.1 服务原语的语义

网络层的临近高层可以利用本原语发起一个路由安全的过程。

本原语的语义如下：

```
NLME-SEC-ROUTE.request    {
                          Request Device Address
                          Request Identifier Type
                          }
```

表 3 给出了 NLME-SEC-ROUTE.request 原语的参数。

**表 3 NLME-SEC-ROUTE.request 参数**

| 字段名称 | 字段类型 | 有效范围 | 描述 |
|---|---|---|---|
| Request Device Address | 16 位网络地址 | 任何网络地址 | 标识符请求设备的网络地址 |
| Request Identifier Type | 整型 | 0x00～0x01 | 指示请求标识符类型是“请求设备地址”还是“待鉴别设备地址”的对应的路由器标识符 |

##### 6.2.2.1.2 产生条件

本原语由一个传感器网络设备的邻近高层产生，发送给其 NLME，用于请求指定设备的标识符。

##### 6.2.2.1.3 收后效果

传感器网络设备的 NLME 收到本原语时，NLME 应向邻近高层发送状态值为 INVALID_RE-

QUEST 的 NLME-SEC-ROUTE.request 原语。

NLME 通过使用 MAC 层的 MCPS-DATA.request 原语，尝试发送一个设备标识符请求命令帧来发起设备鉴别。如果 MAC 层因为任何原因发送命令帧失败，NLME 向邻近高层发送 NLME-SEC-ROUTE.confirm 原语，状态参数值等于 MCPS-DATA.confirm 返回的值。

#### 6.2.2.2 NLME-SEC-ROUTE.confirm

##### 6.2.2.2.1 服务原语的语义

本原语允许告知邻居高层路由安全的结果。

本原语的语义如下：

```
NLME-SEC-ROUTE.confirm    {
                              Status
                              }
```

表 4 给出了 NLME-SEC-ROUTE.confirm 原语的参数。

**表 4 NLME-SEC-ROUTE.confirm 参数**

| 名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| Status | 状态 | 从安全套件或 MCPS-DATA.confirm 原语返回的值 | 相应请求的状态 |

##### 6.2.2.2.2 产生条件

本原语由 NLME 产生，传递给邻近高层，作为请求设备标识符的结果。

##### 6.2.2.2.3 收后效果

邻近高层被告知其设备标识符请求命令的结果。

#### 6.2.2.3 NLME-SEC-ROUTE.indication

##### 6.2.2.3.1 服务原语的语义

NLME 通过发送本原语来通知网络层的邻近高层收到一个设备标识符请求命令帧。

本原语的语义如下：

```
NLME-SEC-ROUTE.indication    {
                                 Request Device Address
                                 Request Identifier Type
                                 }
```

表 5 给出了 NLME-SEC-ROUTE.indication 原语的参数。

**表 5 NLME-SEC-ROUTE.indication 参数**

| 字段名称 | 字段类型 | 有效范围 | 描述 |
|---|---|---|---|
| Request Device Address | 16 位网络地址 | 任何网络地址 | 标识符请求设备的网络地址 |
| Request Identifier Type | 整型 | 0x00～0x01 | 指示请求标识符类型是“请求设备地址”还是“待鉴别设备地址”的对应的路由器标识符 |

#### 6.2.2.3.2 产生条件

本原语由一个传感器网络设备的 NLME 产生，告知其邻近高层收到一个标识符请求命令帧。

#### 6.2.2.3.3 收后效果

在接收到 NLME-SEC-ROUTE.indication 原语之后，邻近高层会使用 Request Identifier Type 设置的参数对标识符请求命令帧发起设备进行鉴别。

### 6.2.2.4 安全路由协议

发起者和响应者的 NLME 执行安全路由协议。

## 6.3 帧安全

### 6.3.1 帧安全概述

当 NLDE-DATA.request 原语中的 SecurityEnable 参数为 FALSE 时，应明确禁止网络层的安全机制；当网络层产生的帧需要加密，或者高层 NIB 中的 nwkSecureAllFrames 属性为 TRUE 时，应启用网络层帧的安全保护机制。网络层帧保护机制应使用国家密码行政主管部门指定的算法和密码运算模式。网络层帧所用的安全等级由 NIB 的 nwkSecurityLevel 属性给出。上层通过设置网络密钥、帧计数器以及所用安全等级等方式，来控制网络层的安全处理操作。

网络层帧的安全结构如图 2 所示，其中安全机制增加了辅助帧头和信息完整性校验码，信息完整性校验码 MIC 是对所有的网络层帧都进行了完整性保护。

| 同步 | 物理层头 | MAC 层帧头 | 网络层帧头 | 辅助帧头 | 加密的网络层净荷 | 信息完整性校验码 MIC |
|---|---|---|---|---|---|---|

**图 2 网络层帧的安全结构**



### 6.3.2 安全的网络层帧

网络层的帧格式如图 3 所示。

当 NPDU 帧启用安全保护时，帧控制域中的安全位应置为 1，表示辅助帧头的存在。

| 八位位组数:可变 | 14 | 可变 | |
|---|---|---|---|
| 初始的网络层帧头 | 辅助帧头 | 加密的净荷 | 加密的信息完整性校验码 MIC |
| | | 网络层帧的安全载荷 | |
| 完整的网络层帧头 | | 安全的网络层净荷 | |

**图 3 安全的网络层帧格式**

### 6.3.3 辅助帧头

#### 6.3.3.1 格式

辅助帧头的格式如图 4 所示。

| 八位位组数:1 | 4 | 0/8 | 0/1 |
|---|---|---|---|
| 安全控制 | 帧计数器 | 源地址 | 密钥序列号 |

**图 4 辅助帧头的格式**

#### 6.3.3.2 安全控制域

##### 6.3.3.2.1 格式

安全控制域的格式如图 5 所示。

| 位:0～2 | 3～4 | 5 | 6 |
|---|---|---|---|
| 安全级别 | 密钥标识符 | 扩展临时子域 | 保留 |

图 5 安全控制域的格式

##### 6.3.3.2.2 安全级别子域

安全级别标识符用于指示输出帧和输入帧进行保护的方式和对荷载进行加密的方式,以及帧内数据真实性的程度(可由信息完整性代码 MIC 的长度反映)。MIC 的位长度可以是 0、32、64 或 128,决定了对 MIC 进行随机猜测时的猜中概率。安全级别的安全特性如表 6 所示,其中,安全级别标识符并不表示各安全级的相对强度。

表 6 网络层可用的安全级别

| 安全级别标识符 | 安全级别子域 | 安全属性 | 数据加密 | 帧完整性(MIC 长度) |
|---|---|---|---|---|
| 0x00 | ‘001’ | MIC-32 | OFF | YES(M=4) |
| 0x01 | ‘010’ | ENC-MIC-32 | ON | YES(M=4) |
| 0x02 | ‘011’ | ENC-MIC-64 | ON | YES(M=8) |
| 0x03 | ‘100’ | ENC-MIC-128 | ON | YES(M=16) |
| 0x04 | ‘101’ | ENC-MIC-128 | ON | YES(M=16) |



##### 6.3.3.2.3 密钥标识符子域

密钥标识符子域用于标识保护帧所用的密钥。密钥标识符子域的编码如表 7 所示。

表 7 密钥标识符子域的编码表

| 密钥标识符 | 密钥标识符子域 | 描述 |
|---|---|---|
| 0x00 | ‘00’ | 共享密钥 |
| 0x01 | ‘01’ | 网络密钥 |

##### 6.3.3.2.4 扩展临时子域

如果辅助帧头中存在源地址域,扩展临时子域应设置为 1,否则设置为 0。

#### 6.3.3.3 计数器域

计数器域提供了网络层帧的新鲜性,可防止对重复帧的处理。

#### 6.3.3.4 源地址域

仅当扩展临时子域的值为 1 时,辅助帧头中才包含源地址域。源地址域表示负责对网络层帧提供

安全保护的设备的 64 位 IEEE 地址。

#### 6.3.3.5 密钥序列号域

仅当密钥标识符子域的值为 1(对应网络密钥)时,辅助帧头中才包含密钥序列号域。密钥序列号域表示对网络层帧提供安全保护的网络密钥的序列号。

### 6.4 命令帧

#### 6.4.1 概述

网络层定义的命令帧如表 8 所示。

**表 8 网络层命令帧**

| 命令帧标识符 | 命令名称 | 子章节 |
|---|---|---|
| 0x0f | 标识符请求命令帧 | 6.4.2 |
| 0x10 | 标识符响应命令帧 | 6.4.3 |
| 0x13～0xFF | 保留 | — |

#### 6.4.2 设备标识符请求命令帧

##### 6.4.2.1 格式

设备标识符请求命令帧的格式如图 6 所示。

| 八位位组:1 | 1 | 2/8 | 0/2/8 |
|---|---|---|---|
| 命令标识 | 请求类型 | 请求设备地址 | 特鉴别设备地址 |

**图 6 设备标识符请求命令格式**

##### 6.4.2.2 MAC 数据请求服务

为了使用 GB/T 15629.15—2010 定义的 MAC 数据服务发送设备标识符请求命令,应提供以下信息:

a) 目的 MAC 地址和 PAN 标识符应根据请求类型分别设置为相应设备的网络地址或者相应组的多播地址和相应设备或者组所在的 PAN 标识符;

b) 源 MAC 地址和 PAN 标识符应设置为发送设备标识符请求命令帧的设备的网络地址和 PAN 标识符;

c) 帧控制字段应指示该帧是一个 MAC 数据帧,并且帧控制字段的 MAC 安全子字段应设置为禁用;

d) 寻址模式和 Intra-PAN 标志应设置为支持该寻址字段。

##### 6.4.2.3 网络层帧头字段

设备标识符请求命令帧通过单播或组播的方式发送,网络层头字段应设置如下:

a) 源地址字段应设置为命令帧发起设备的 16 位网络地址;

b) 帧控制字段的源地址字段应设置为 1,网络层头的源地址字段应存在,并包含帧发起设备的 64

位地址；

c) 目的地址字段应根据请求类型设置为相应的命令帧接收设备的 16 位网络地址，或者命令帧接收组的多播地址。

#### 6.4.2.4 网络层负载字段

设备标识符请求命令帧的网络层负载包含一个命令标识符、请求类型、请求设备的地址和待鉴别的设备的地址等字段。其中命令标识符为 0x0f，指示为设备标识符请求命令帧，请求类型字段格式如图 7 所示。

| 位：0～2 | 3 | 4 | 5～6 | 7 |
|---|---|---|---|---|
| 保留 | 请求设备地址指示 | 特鉴别设备地址指示 | 请求响应设备类型 | 请求标识符类型 |

**图 7 请求类型字段格式**

其中：

请求设备地址指示：若为 0，表示请求设备地址”字段使用 16 位短地址；若为 1，表示请求设备地址”字段使用 64 位 IEEE 地址，该字段默认值为 1。

待鉴别设备地址指示：若为 0，表示待鉴别设备地址”字段使用 16 位短地址；若为 1，表示如果存在待鉴别设备地址”字段，使用 64 位 IEEE 地址。

请求响应设备类型指示：若为 00，表示发向直接路由器或待鉴别的路由器；若为 01，表示发向直接路由器或待鉴别路由器的父节点；若为 10，表示发向直接路由器或待鉴别路由器父节点的子节点；若为 11，表示发向用户监视端。

请求标识符类型指示：若为 0，表示“请求设备地址”对应的是路由器的标识符；若为 1，表示“待鉴别设备地址”对应的是路由器的标识符。

请求设备地址为初始发起该标识符请求命令帧的设备地址，可以为 16 位短地址，或者 64 位 IEEE 地址。

待鉴别设备地址为“请求标识符类型”为 0 时，本字段不出现；“请求标识符类型”为 1 时，请求本字段指示的设备的标识符。可以为 16 位短地址，或者 64 位 IEEE 地址。

### 6.4.3 设备标识符响应命令帧

#### 6.4.3.1 格式

接收到设备标识符请求命令帧的设备，可以通过设备标识符响应命令帧反馈指定设备的标识符。设备标识符响应命令帧的格式如图 8 所示。

| 八位位组：1 | 1 | 2/8 | 8 |
|---|---|---|---|
| 命令标识 | 响应类型 | 设备地址 | 设备标识符 |

**图 8 设备标识符响应命令格式**

#### 6.4.3.2 MAC 数据服务请求

为了使用 GB/T 15629.15—2010 定义的 MAC 数据服务发送设备标识符响应命令，应提供以下信息：

a) 目的 MAC 地址和 PAN 标识应设置为发送设备标识符请求命令帧的设备的网络地址和 PAN 标识符。

b) 源 MAC 地址和 PAN 标识符应设置为发送设备响应标识符命令帧的设备的网络地址和 PAN 标识符。

c) 帧控制字段应指示该帧是一个 MAC 数据帧，并且帧控制字段的 MAC 安全子字段应设置为禁用；

d) 寻址模式和 Intra-PAN 标志应设置为支持该寻址字段。

#### 6.4.3.3 网络层帧头字段

为了使设备标识符响应命令能够正确到达目的设备，并完成设备标识符反馈，网络层帧头字段应提供以下信息：

a) 网络层帧头中的源地址应设置为发送设备的 16 位网络地址。

b) 网络层帧头中的目的地址应设置为接收到的设备标识符请求命令帧中“请求设备地址”字段对应设备的地址。

c) 帧控制字段的源地址字段应设置为 1，网络层帧头的源地址字段应包含帧的发送设备的 64 位 IEEE 地址。帧控制字段中的目的地址字段应设置为 1，网络层帧头的目的地址应设置为接收到的设备标识符请求命令帧中“请求设备地址”相应的设备的 64 位 IEEE 地址。

#### 6.4.3.4 网络层负载字段

网络层负载字段的格式如图 9 所示。

| 位：0～5 | 6 | 7 |
|---|---|---|
| 保留 | 设备地址指示 | 响应标识符类型 |

**图 9 网络层负载字段**



其中：

设备地址指示：若为 0，表示“设备地址”字段使用 16 位短地址；若为 1，表示“设备地址”字段使用 64 位 IEEE 地址。

响应标识符类型：根据响应的标识符请求命令帧中的“请求标识符类型”设置。

命令标识符为 0x10 时，表示设备标识符响应命令帧。

设备地址是指反馈的为该地址所对应的设备的标识符。可以为 16 位短地址或 64 位 IEEE 地址。

设备标识符是指“设备地址”字段对应设备的标识符。

### 6.5 安全相关的 NIB 属性

网络层的 NIB 属性包含管理网络层安全所需的各种属性，可分别用 NLME-GET.Request 和 NLME-SET.request 原语进行读写。与安全相关的网络层 NIB 属性如表 9、表 10 和表 11 所示。其中表 9 列出了网络环境安全设定的 NIB 属性，表 10 列出了建立密钥时设定的(包含密钥长度类型等)NIB 属性，表 11 列出了使用的计数器的 NIB 属性。

**表 9 网络层安全设定的 NIB 属性**

| 属性 | 标识符 | 类型 | 范围 | 描述 | 默认值 |
|---|---|---|---|---|---|
| nwkSecurity Level | 0xa0 | 整型 | 0x00～0x04 | 接收和发送帧的安全级别 | 0x04 |
| nwkSecurity-MaterialSet | 0xa1 | | 可变的 | 一组网络安全材料描述符的集合，可用于维护活动网络密钥和备用网络密钥 | |

表 9（续）

| 属性 | 标识符 | 类型 | 范围 | 描述 | 默认值 |
|---|---|---|---|---|---|
| nwkActiveKey SeqNumber | 0xa2 | 整型 | 0x00～0xFF | nwkSecurityMaterialSet 中活动的动网络密钥的序列号 | 0x00 |
| nwkAllFresh | 0xa3 | 布尔 | TRUE 或 FALSE | 当输入帧计数器的内存溢出时，用于指示网络层输入帧是否需进行新鲜性检查 | TRUE |
| nwkSecure AllFrames | 0xa5 | 布尔 | TRUE 或 FALSE | 指示是否需对网络层输入和输出数据帧启用安全保护。如果设置为 0x01，需对所有的输入和输出帧进行安全处理（发往当前设备、安全子域为 0 的数据帧除外）；如果该属性值为 0x01，网络层不能转发安全子域为 0 的帧。NLDE-DATA.request 原语中的 SecurityEnable 参数将覆盖该属性的设置 | TRUE |
| nwkRouterAuthen | 0xab | 布尔 | TURE 或 FALSE | 如果该值为 TRUE，节点设备和路由器之间需要相互鉴别，否则不需要进行鉴别操作 | FALSE |



表 10　建立密钥时设定的 NIB 属性

| 名称 | 类型 | 范围 | 描述 | 默认 |
|---|---|---|---|---|
| KeySeqNumber | 整型 | 0x00～0xFF | 信任中心分配给网络密钥的序列号，可在密钥更新时对网络进行区分，并对输入帧进行安全操作 | 0x00 |
| OutgoingFrame Counter | 4 个八位位组的集合 | 0x00000000～0xFFFFFFFF | 输出帧计数器 | 0x00000000 |
| IncomingFrame CounterSet | 见表 11 中接收帧计数器描述符的值的设置 | 可变的 | 输入帧计数器的值、以及相应的设备地址的集合 | 空集 |
| Key | 16 个八位位组的集合 | | 密钥的实际取值 | |
| KeyType | 整型 | 0x00～0xFF | 密钥的类型。<br>0x01：标准<br>0x05：高安全性<br>其他值保留 | 0x01 |

表 11　计数器的 NIB 属性

| 名称 | 类型 | 范围 | 描述 | 默认 |
|---|---|---|---|---|
| SenderAddress | 地址 | 任何有效的 64 位 IEEE 地址 | 扩展设备地址 | — |
| IncomingFrameCounter | 4 个八位位组的有序集 | 0x00000000～0xFFFFFFFF | 输入帧的计数器 | 0x00000000 |

# 7 应用支持子层安全

## 7.1 应用支持子层安全概述

应用支持子层负责安全地转发输出帧、接收输入帧以及建立并管理密钥。上层通过向应用支持子层发出原语来控制密钥的管理。

## 7.2 应用支持子层安全服务

### 7.2.1 概述

表 12 列出了密钥管理和维护可用的原语。

**表 12 应用支持子层安全原语**

| APSME 安全原语 | 请求 | 确认 | 指示 | 响应 | 描述 |
|---|---|---|---|---|---|
| APSME-DISTRIBUTE-KEY | 7.2.2.2 | — | 7.2.2.3 | — | 信任中心分发密钥材料到设备 |
| APSME-ESTABLISH-KEY | 7.2.3.2 | 7.2.3.3 | 7.2.3.4 | 7.2.3.5 | 两个设备之间手动建立一个共享密钥 |
| APSME-UPDATE-KEY | 7.2.4.2 | — | 7.2.4.3 | — | 信任中心去通知其他设备应该更换到一个新的密钥 |
| APSME-REVOCATION-KEY | 7.2.5.1 | — | 7.2.5.2 | — | 信任中心去通知相关设备撤销使用特定密钥信息 |
| APSME-ACCESS-CONTROL | 7.2.6.2 | 7.2.6.3 | 7.2.6.4 | — | 用于控制用户对传感网的节点资源和数据资源的访问 |
| APSME-IDENTITY-AUTHENTICATE | 7.2.7.2 | 7.2.7.3 | 7.2.7.4 | — | 用于两个设备之间进行身份鉴别 |
| APSME-MESSAGE-AUTHENTICATE | 7.2.8.2 | 7.2.8.3 | 7.2.8.4 | — | 用于设备之间进行广播消息鉴别 |
| APSME-SECURE-DATA-AGGREGATION-START | 7.2.9.1 | 7.2.9.2 | 7.2.9.3 | 7.2.9.4 | 用于网络协调器和设备进行安全数据融合 |
| APSME-SECURE-DATA-AGGREGATION-REVOKE | 7.2.10.5 | — | 7.2.10.6 | — | 用于网络协调器和设备安全数据融合的撤销 |



### 7.2.2 密钥分发服务

#### 7.2.2.1 概述

APSME 提供信任中心向设备分发密钥材料的服务。无线传感设备以保护密钥不被泄露的方式写入信任中心分发的密钥材料中。

#### 7.2.2.2 APSME-DISTRIBUTE-KEY.request

##### 7.2.2.2.1 服务原语的语义

APSME-DISTRIBUTE-KEY.request 原语用来分发密钥到其他设备。

该原语应提供以下的接口：

```
APSME-DISTRIBUTE-KEY.request  {
                                DistributeType
                                KeyType
                                NodeID
                                KeyData
                                }
```

表 13 列出了 APSME-DISTRIBUTE-KEY.request 原语的参数。

**表 13 APSME-DISTRIBUTE-KEY.request 原语的参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| DistributeType | 整型 | 0x00～0x06 | 密钥材料分发的方式 |
| KeyType | 整型 | 0x00～0x01 | 被分发的密钥材料的类型 |
| NodeID | 地址 | 通常为设备有效的 64 位 IEEE 地址 | 设备唯一的标识符 |
| KeyData | Variable | 可变的 | 分发使用的安全参数 |

APSME-DISTRIBUTE-KEY.request 原语中的参数 DistributeType 值见表 14。

**表 14 APSME-DISTRIBUTE-KEY.request 原语的 DistributeType 参数值**

| 枚举 | 值 | 描述 |
|---|---|---|
| 信任中心配置 | 0x00 | 由信任中心直接载入节点设备 |
| 手持设备进行分发 | 0x01 | 通过手持设备将密钥材料分发到设备 |

APSME-DISTRIBUTE-KEY.request 原语的参数 KeyType 的值见表 15，参数 KeyData 的类型取决于参数 KeyType 的值。

**表 15 APSME-DISTRIBUTE-KEY.request 原语的 KeyType 参数值**

| 枚举 | 值 | 描述 |
|---|---|---|
| Initialization key | 0x00 | 初始化密钥材料，无线传感设备在安装于现场之前，用于协商共享密钥，应该根据实际需求向设备写入 |
| Initialization value | 0x01 | 初始化值，确立和维持密码密钥关系所必需的数据 |

Initialization key 的参数见表 16。

**表 16 Initialization key 参数列表**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| Key ID | Variable | 可变的 | 密钥材料对应的标识符，用于唯一的密钥材料 |
| Key | Set of 16 octets | 可变的 | 初始化的密钥材料 |

Initialization value 的参数见表 17。

**表 17　Initialization value 参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| value | Variable | 可变的 | 初始化的值 |

#### 7.2.2.2.2　产生条件

当信任中心需要分发密钥材料到设备时，它的邻近高层产生该原语。

#### 7.2.2.2.3　收后效果

在接收到 APSME-DISTRIBUTE-KEY.request 原语后，信任中心的 APSME 作为密钥分发的发起者，直接分发密钥材料到目标设备。

### 7.2.2.3　APSME-DISTRIBUTE-KEY.indication

#### 7.2.2.3.1　服务原语的语义

APSME-DISTRIBUTE-KEY.indication 原语用来通知邻近高层接收到了密钥材料。

该原语应提供以下的接口：

```
APSME-DISTRIBUTE-KEY.indication  {
                                    KeyType
                                    NodeID
                                    KeyData
                                    }
```

表 18 列出了 APSME-DISTRIBUTE-KEY.indication 原语的参数。

**表 18　APSME-DISTRIBUTE-KEY.indication 参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| KeyType | 整型 | 0x00～0x01 | 被分发的密钥材料的类型 |
| NodeID | 地址 | 通常为设备有效的 64 位 IEEE 地址 | 设备唯一的标识符 |
| KeyData | Variable | 可变的 | 分发使用的安全参数。参数的类型取决于下面 KeyType 参数：<br>KeyType＝0x00 见表 20<br>KeyType＝0x01 见表 21 |

APSME-DISTRIBUTE-KEY.indication 原语的参数 KeyType 的值见表 19，参数 KeyData 的类型取决于参数 KeyType 的值。

**表 19　APSME-DISTRIBUTE-KEY.indication 原语的 KeyType 参数值**

| 枚举 | 值 | 描述 |
|---|---|---|
| Initialization key | 0x00 | 初始化密钥材料，无线传感设备在安装于现场之前，用于协商共享密钥的密钥材料，根据实际需求向设备写入 |
| Initialization value | 0x01 | 初始化值，确立和维持密码密钥关系所必需的数据 |

Initialization key 的参数见表 20。

**表 20　Initialization key 参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| Key ID | Variable | 可变的 | 密钥材料对应的标识符，用于唯一的密钥材料 |
| Key | Set of 16 octets | 可变的 | 初始化的密钥材料 |

Initialization value 的参数见表 21。

**表 21　Initialization value 参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| value | Variable | 可变的 | 初始化的值 |

#### 7.2.2.3.2　产生条件

当目标设备接收到信任中心分发的密钥材料后，其 APSME 产生此原语。

#### 7.2.2.3.3　收后效果

收到该原语后，接收者的邻近高层被通知已收到密钥材料。

### 7.2.3　密钥建立服务

#### 7.2.3.1　概述

APSME 提供允许两个设备之间可以手动建立一个共享密钥的服务。初始信任信息（例如一个主密钥）应在运行密钥建立协议之前安装在每个设备上。

#### 7.2.3.2　APSME-ESTABLISH-KEY.request

##### 7.2.3.2.1　服务原语的语义

APSME-ESTABLISH-KEY.request 原语用来启动一个密钥建立协议，以安全地和另一设备通信。一个设备作为发起设备，另一个设备作为响应设备，发起设备启动密钥建立协议。

该原语应提供以下的接口：

```
APSME-ESTABLISH-KEY.request    {
                               KeyTpye
                               DestAddress
                               ResponderAddress
                               KeyEstablishmentMethod
                               KeyData
                               }
```

表 22 列出了 APSME-ESTABLISH-KEY.request 原语的参数。

**表 22　APSME-ESTABLISH-KEY.request 参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| KeyTpye | 整型 | 0x00～0x01 | 被分发的密钥材料的类型 |
| DestAddress | 地址 | 任何有效的 64 位 IEEE 地址 | 发起设备的 64 位 IEEE 地址 |
| Responder-Address | 地址 | 任何有效的 64 位 IEEE 地址 | 响应设备的 64 位 IEEE 地址 |
| KeyEstablishment-Method | 整型 | 0x00～0x03 | 请求密钥建立的方法 |
| KeyData | Variable | 可变的 | 用于密钥建立的安全参数 |

APSME-ESTABLISH-KEY.request 原语中参数 KeyType 的值见表 23。参数 KeyData 的类型取决于参数 KeyType 的值。

**表 23　APSME-ESTABLISH-KEY.request 原语的 KeyTpye 参数值**

| 枚举 | 值 | 描述 |
|---|---|---|
| Shared key | 0x00 | 发起者请求建立共享密钥 |
| Session key | 0x01 | 发起者请求建立会话密钥 |

APSME-ESTABLISH-KEY.request 原语中参数 KeyEstablishment-Method 的值见表 24。

**表 24　APSME-ESTABLISH-KEY.request 原语的 KeyEstablishment-Method 参数值**

| 枚举 | 值 | 描述 |
|---|---|---|
| 基于随机密钥池的方法 | 0x00 | 采用基于密钥池预分发的方法建立直接密钥 |
| 基于多项式池的方法 | 0x01 | 采用多项式池的密钥预分配的分发方法建立直接密钥 |
| 同一簇内路径密钥建立的方法 | 0x02 | 同一簇内建立路径密钥 |
| 不同簇内路径密钥建立的方法 | 0x03 | 不同簇内建立路径密钥 |

基于随机密钥池的 KeyData 参数见表 25。

**表 25　基于随机密钥池的 KeyData 参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| Key ID | Variable | 可变的 | 密钥材料对应的标识符，用于唯一的密钥材料 |
| Key | Set of 16 octets | 可变的 | 初始化的密钥材料 |

基于多项式池的 KeyData 参数见表 26。

**表 26　基于多项式池的 KeyData 参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| Polynomial ID | Variable | 可变的 | 用于唯一的标识多项式 |
| Nonce | Variable | 可变的 | 随机产生的 Nonce 值 |
| Node ID | Variable | 可变的 | 密钥建立发起设备的标识符 |

基于同一簇内路径密钥建立的 KeyData 参数见表 27。

**表 27　基于同一簇内路径密钥建立的 KeyData 参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| Key ID | Variable | 可变的 | 密钥标识符,用于唯一标准密钥 |
| KeyData | Variable | 可变的 | 信任中心发送到节点的密钥消息 |

基于不同簇内路径密钥建立的 KeyData 参数见表 28。

**表 28　基于不同簇内路径密钥建立的 KeyData 参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| Key ID | Variable | 可变的 | 密钥标识符,用于唯一标准密钥 |
| KeyData | Variable | 可变的 | 可信节点发送到密钥请求节点的密钥消息 |

#### 7.2.3.2.2　产生条件

当发起设备请求和一个响应设备建立一个会话密钥时,它的上层产生本原语。

#### 7.2.3.2.3　收后效果

接收到 APSME-ESTABLISH_KEY.request 原语后,如果 KeyEstablishmentMethod 参数等于 0x00,APSME 执行基于密钥池的预分配协议。本地的 APSME 作为本协议的发起设备,DestAddress 参数指明的 APSME 作为本协议的发起设备,Responder-Address 参数设置为广播地址。

如果 KeyEstablishmentMethod 参数等于 0x01,执行基于多项式池的预分配协议。本地的 APSME 作为本协议的发起设备,DestAddress 参数指明的 APSME 作为本协议的发起设备,Responder-Address 参数设置为广播地址。

如果 KeyEstablishmentMethod 参数等于 0x02,执行同一簇内建立路径密钥协议,即发起节点与目的节点有共同的邻居节点,且存在一条的安全连接。本地的 APSME 作为本协议的发起设备,DestAddress 参数指明的 APSME 作为本协议的发起设备,Responder-Address 参数设置为目的设备的 64 位 IEEE 地址。

如果 KeyEstablishmentMethod 参数等于 0x03,执行不同簇内建立路径密钥协议。本地的 APSME 作为本协议的发起设备,DestAddress 参数指明的 APSME 作为本协议的发起设备,Responder-Addres 参数设置为目的设备的 64 位 IEEE 地址。

### 7.2.3.3　APSME-ESTABLISH-KEY.confirm

#### 7.2.3.3.1　服务原语的语义

本原语在密钥建立协议成功或者失败后发给邻近高层。

该原语应提供以下接口:

```
APSME-ESTABLISH-KEY.confirm    {
                               Address
                               Status
                               }
```

表 29 列出了 APSME-ESTABLISH-KEY.confirm 参数的原语。

表 29　APSME-ESTABLISH-KEY.confirm 参数

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| Address | 地址 | 任何有效的 64 位 IEEE 地址 | 执行密钥建立协议设备的 64 位 IEEE 地址 |
| Status | 状态 | 通过给出的值或者 NLDE-DATA.confirm 原语返回的任何状态值 | 密钥建立协议的最终状态 |

#### 7.2.3.3.2　产生条件

在密钥建立协议完成后，响应设备和发起设备的 APSME 都应向邻近高层发出该原语。

#### 7.2.3.3.3　收后效果

如果密钥建立成功，发起设备和响应设备的 AIB 应用最新的会话密钥更新，发起者应能和响应者加密通信。如果密钥建立不成功，AIB 不能改变。

### 7.2.3.4　APSME-ESTABLISH-KEY.indication

#### 7.2.3.4.1　服务原语的语义

当响应者从发起者接收到一个原始的密钥建立信息或者信任中心从响应者接收到一个密钥建立信息时，它的 APSME 向邻近高层发出该原语。

该原语应提供以下接口：

```
APSME-ESTABLISH-KEY.indication  {
                                    InitiatorAddress
                                    KeyEstablishmentMethod
                                    KeyType
                                    KeyData
                                    }
```

表 30 列出了 APSME-ESTABLISH-KEY.indication 原语的参数。

表 30　APSME-ESTABLISH-KEY.indication 参数

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| InitiatorAddress | 地址 | 任何有效的 64 位 IEEE 地址 | 发起设备的 64 位 IEEE 地址 |
| KeyEstablishment-Method | 整型 | 0x00～0x03 | 请求密钥建立的方法应是以下之一：<br>0x00：基于密钥池的密钥预分配建立直接密钥<br>0x01：基于多项式池的密钥预分配建立直接密钥<br>0x03：同一簇内建立路径密钥<br>0x04：不同簇内建立路径密钥 |
| KeyType | 整型 | 0x00～0x01 | 标识应该被分发的密钥材料的类型 |
| KeyData | Variable | 可变的 | 用于密钥建立的安全参数 |

APSME-ESTABLISH-KEY.indication 原语中参数 KeyEstablishment-Method 的值见表 31。

**表 31　APSME-ESTABLISH-KEY.indication 原语的 KeyEstablishment-Method 参数值**

| 枚举 | 值 | 描述 |
|---|---|---|
| 基于随机密钥池的方法 | 0x00 | 代表采用基于密钥池预分发的方法建立直接密钥 |
| 基于多项式池的的方法 | 0x01 | 代表采用多项式池的密钥预分配的发方法建立直接密钥 |
| 同一簇内路径密钥建立的方法 | 0x02 | 同一簇内建立路径密钥 |
| 不同簇内路径密钥建立的方法 | 0x03 | 不同簇内建立路径密钥 |

基于随机密钥池的 KeyData 参数见表 32。

**表 32　基于随机密钥池的的 KeyData 参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| Key ID | Variable | 可变的 | 密钥材料对应的标识符,用于唯一的密钥材料 |
| Key | Set of 16 octets | 可变的 | 初始化的密钥材料 |

基于多项式池的 KeyData 参数见表 33。

**表 33　基于多项式池的 KeyData 参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| Polynomial ID | Variable | 可变的 | 用于唯一的标识多项式 |
| Nonce | Variable | 可变的 | 随机产生的 Nonce 值 |
| Node ID | Variable | 可变的 | 密钥建立发起设备的标识符 |

基于同一簇内路径密钥建立的 KeyData 参数见表 34。

**表 34　基于同一簇内路径密钥建立的 KeyData 参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| Key ID | Variable | 可变的 | 密钥标识符,用于唯一标准密钥 |
| KeyData | Variable | 可变的 | 来自信任中心的密钥消息 |

基于不同簇内路径密钥建立的 KeyData 参数见表 35。

**表 35　基于不同簇内路径密钥建立的 KeyData 参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| Key ID | Variable | 可变的 | 密钥标识符,用于唯一标准密钥 |
| KeyData | Variable | 可变的 | 来自可信节点的密钥消息 |

#### 7.2.3.4.2 产生条件

当一个发起设备收到开始密钥建立协议的请求时，响应设备的 APSME 向邻近高层发出该原语。

#### 7.2.3.4.3 收后效果

收到 APSME-ESTABLISH-KEY.indication 原语后，邻近高层根据 KeyEstablishmentMethod 和 InitiatorAddress 参数确定是否和发起者建立密钥连接，并使用 APSME-ESTABLISH-KEY.response 原语响应。

### 7.2.3.5 APSME-ESTABLISH-KEY.response

#### 7.2.3.5.1 服务原语的语义

响应设备或者信任中心的邻近高层使用 APSME-ESTABLISH-KEY.Response 原语响应接收到的 APSME-ESTABLISH-KEY.indication 原语。邻近高层决定是否继续密钥建立或者终止，并在 APSME-ESTABLISH-KEY.indication 原语 Accept 参数中指明。

该原语应提供以下接口：

```
APSME-ESTABLISH-KEY.response    {
                                InitiatorAddress
                                Accept
                                }
```

表 36 列出了 APSME-ESTABLISH-KEY.response 原语的参数。

**表 36 APSME-ESTABLISH-KEY.response 参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| InitiatorAddress | 地址 | 有效的 64 位 IEEE 地址 | 初始化密钥建立设备的 64 位 IEEE 地址 |
| Accept | 布尔 | TRUE 或 FALSE | 该参数代表响应到初始化者的请求执行一个密钥建立协议。响应为：<br>TRUE：接收<br>FALSE：拒绝 |

#### 7.2.3.5.2 产生条件

发起者启动一个密钥建立协议（即收到 APSME-ESTABLISH-KEY.indication）后，邻近高层应产生 APSME-ESTABLISH-KEY.response 原语并且提供给 APSME。响应者的邻近高层可利用该原语决定接受或拒绝和给定的发起者建立密钥的请求。

#### 7.2.3.5.3 收后效果

如果参数 Accept 的值为 TRUE，响应者的 APSME 按照 KeyEstablishmentMethod 参数的指示执行密钥建立协议。

## 7.2.4 密钥更新服务

### 7.2.4.1 概述

APSME 提供允许一个信任中心通知其他设备更换到一个新密钥的服务。

#### 7.2.4.2 APSME-UPDATE-KEY.request

##### 7.2.4.2.1 服务原语的语义

该原语应提供以下接口：

```
APSME-UPDATE-KEY.request    {
                                DestAddress
                                KeyType
                                KeyData
                            }
```

表 37 列出了 APSME-UPDATE-KEY.request 原语的参数。

**表 37 APSME-UPDATE-KEY.request 参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| DestAddress | 地址 | 任何有效的 64 位 IEEE 地址 | 密钥更新命令发送到设备的 64 位 IEEE 地址 |
| KeyType | 整型 | 0x00～0x03 | 更新的密钥材料的类型 |
| KeyData | Variable | 可变的 | 分发使用的安全参数 |

APSME-UPDATE-KEY.request 原语中参数 KeyType 的值见表 38。参数 KeyData 的类型取决于参数 KeyType 的值。

**表 38 APSME-UPDATE-KEY.request 原语的 KeyType 参数值**

| 枚举 | 值 | 描述 |
|---|---|---|
| Initialization key | 0x00 | 初始化密钥材料，无线传感设备在安装于现场之前，用于协商共享密钥的密钥材料，应该根据实际需求向设备写入 |
| Initialization value | 0x01 | 初始化值，确立和维持密码密钥关系所必需的数据 |
| Shared key | 0x02 | 信任中心更新的密钥为共享密钥 |
| Session key | 0x03 | 信任中心更新的密钥为会话密钥 |

初始化密钥材料、共享密钥或会话密钥的参数见表 39。

**表 39 初始化密钥材料、共享密钥或会话密钥的参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| Key ID | Variable | 可变的 | 密钥对应的标识符，用于唯一的密钥 |
| Key | Set of 16 octets | 可变的 | 新更新的密钥消息 |

初始化值的 Initialization value 参数的值见表 40。

**表 40 初始化值的 Initialization value 参数值**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| value | Variable | 可变的 | 初始化的值 |

#### 7.2.4.2.2 产生条件

当信任中心通知一个设备应该更换一个新密钥时，它的邻近高层产生 APSME-UPDATE-KEY.request 原语。

#### 7.2.4.2.3 收后效果

接收到 APSME-UPDATE-KEY.request 原语后，设备首先创建一个密钥更新命令帧，该命令帧的目的地址设置为和 DestAddress 相同的地址。

该命令帧应按照附录 B 的帧安全部分执行安全保护，如果安全处理成功，向 DestAddress 参数指定的设备发送 NLDE-DATA.request 原语。

### 7.2.4.3 APSME-UPDATE-KEY.indication

#### 7.2.4.3.1 服务原语的语义

APSME 通过发出 APSME-UPDATE-KEY.indication 原语通知邻近高层其已经收到了密钥更新命令帧。

该原语应提供以下接口：

```
APSME-UPDATE-KEY.indication   {
                                  DestAddress
                                  KeyType
                                  KeyData
                              }
```

表 41 列出了 APSME-UPDATE-KEY.indication 原语的参数。

**表 41 APSME-UPDATE-KEY.indication 参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| DestAddress | 地址 | 任何有效的 64 位 IEEE 地址 | 发送更换密钥命令的设备的扩展 64 位 IEEE 地址 |
| KeyType | 整型 | 0x00～0x03 | 更新的密钥材料的类型 |
| KeyData | Variable | 可变的 | 更新使用的安全参数 |

APSME-UPDATE-KEY.indication 原语中参数 KeyType 的值见表 42。参数 KeyData 的类型取决于参数 KeyType 的值。

**表 42 APSME-UPDATE-KEY.indication 原语的 KeyType 参数**

| 枚举 | 值 | 描述 |
|---|---|---|
| Initialization key | 0x00 | 初始化密钥材料，无线传感设备在安装于现场之前，用于协商共享密钥的密钥材料，应该根据实际需求向设备写入 |
| Initialization value | 0x01 | 初始化值，确立和维持密码密钥关系所必需的数据 |
| Shared key | 0x02 | 信任中心更新的密钥为共享密钥 |
| Session key | 0x03 | 信任中心更新的密钥为会话密钥 |

初始化密钥材料、共享密钥或会话密钥的参数见表 43。

表 43　初始化密钥材料、共享密钥或会话密钥的参数

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| Key ID | Variable | 可变的 | 密钥对应的标识符，用于唯一的密钥 |
| Key | Set of 16 octets | 可变的 | 新更新的密钥消息 |

初始化值的 Initialization value 参数的值见表 44。

表 44　初始化值的 Initialization value 参数

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| value | Variable | 可变的 | 初始化的值 |

#### 7.2.4.3.2　产生条件

当 APSME 收到一个成功加密并认证过的密钥更新命令帧后，产生该原语。

#### 7.2.4.3.3　收后效果

在接收到 APSME-UPDATE-KEY.indication 原语后，邻近高层被告知 SrcAddress 参数指明的信任中心正在请求更新 KeyType 类型的密钥，更新的密钥为 KeyData。

### 7.2.5　密钥撤销服务

#### 7.2.5.1　APSME-REVOCATION-KEY.request

##### 7.2.5.1.1　服务原语的语义

一个信任中心通过发送本原语，通知相关设备撤销使用特定密钥信息。

该原语应提供以下接口：

```
APSME-REVOCATION-KEY.request    {
                                DestAddress
                                Key ID
                                Revocation Time
                                Revocation Reason
                                }
```

表 45 列出了 APSME-REVOCATION-KEY.request 原语的参数。

表 45　APSME-REVOCATION-KEY.request 参数

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| DestAddress | 地址 | 任何有效的 64 位 IEEE 地址 | 密钥撤销命令发送到设备的扩展 64 位 IEEE 地址 |
| Key ID | Variable | 可变的 | 撤销密钥的标识符 |
| Revocation Time | Variable | 可变的 | 撤销密钥的日期时间 |
| Revocation Reason | Variable | 可变的 | 撤销密钥的原因 |

#### 7.2.5.1.2 产生条件

当信任中心需要通知相关设备撤销一个特定密钥时，它的邻近高层产生 APSME-REVOCATION-KEY.request 原语。

#### 7.2.5.1.3 收后效果

在接收到 APSME-REVOCATION-KEY.request 原语后，设备首先创建一个密钥撤销命令帧，该命令帧的目的地址应设置为和 DestAddress 相同的地址，如果需要通知网内所有设备，则目的地址为广播地址。

### 7.2.5.2 APSME-REVOCATION-KEY.indication

#### 7.2.5.2.1 服务原语的语义

APSME 发出 APSME-REVOCATION-KEY.indication 原语通知邻近高层已经收到了密钥撤销命令帧。

该原语应提供以下接口：

```
APSME-REVOCATION-KEY.indication   {
                                      SrcAddress
                                      Key ID
                                      Revocation Time
                                      Revocation Reason
                                  }
```

表 46 列出了 APSME-REVOCATION-KEY.indication 原语的参数。

**表 46 APSME-REVOCATION-KEY.indication 参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| SrcAddress | 地址 | 任何有效的 64 位 IEEE 地址 | 发送密钥撤销命令的设备的扩展 64 位 IEEE 地址 |
| Key ID | Variable | 可变的 | 撤销密钥的标识符 |
| Revocation Time | Variable | 可变的 | 撤销密钥的日期时间 |
| Revocation Reason | Variable | 可变的 | 撤销密钥的原因 |

#### 7.2.5.2.2 产生条件

当接收者收到一个成功认证过的密钥撤销命令帧时，它的 APSME 产生该原语。

#### 7.2.5.2.3 收后效果

在接收到 APSME-REVOCATION-KEY.indication 原语后，邻近高层被告知 SrcAddress 参数指明的信任中心要求撤销的密钥标识符 Key ID，撤销日期 Revocation Time 及撤销的原因 Revocation Reason，设备根据密钥撤销命令删除存储的相关密钥信息。

### 7.2.6 访问控制服务

#### 7.2.6.1 概述

APSME 提供允许控制用户对传感器网络的节点资源和数据资源进行访问的服务。

#### 7.2.6.2 APSME-ACCESS-CONTROL.request

##### 7.2.6.2.1 服务原语的语义

APSME-ACCESS-CONTROL.request 原语用来启动访问控制过程或者响应其他设备发起的访问。

该原语应该提供以下接口：

```
APSME-ACCESS-CONTROL.request  {
                              ResponderAddress
                              Action
                              AccessObject
                              AccessControlMethod
                              }
```

表 47 列出了 APSME-ACCESS-CONTROL.request 原语的参数。

**表 47 APSME-ACCESS-CONTROL.request 原语参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| ResponderAddress | 地址 | 任何有效的 64 位 IEEE 地址 | 响应设备的 64 位 IEEE 地址 |
| Action | 枚举 | INITIATE<br>RESPOND_ACCEPT<br>RESPOND_REJECT | 采取的行动 |
| AccessObject | 整型 | 0x00～0x01 | 请求访问的对象应是以下之一：<br>0x00：数据资源<br>0x01：节点资源 |
| AccessControlMethod | 整型 | 0x00～0x01 | 请求访问控制的方法应是以下之一：<br>0x00：自主访问方法<br>0x01：强制访问方法 |

Action 参数值列举见表 48。

**表 48 Action 参数值列举**

| 列举 | 值 | 描述 |
|---|---|---|
| INITIATE | 0x00 | 启动访问过程 |
| RESPOND_ACCEPT | 0x01 | 接受响应访问请求 |
| RESPOND_REJECT | 0x02 | 拒绝响应访问请求 |

##### 7.2.6.2.2 产生条件

当用户发起请求或响应设备需要启动访问控制时，它的上层产生该原语。

#### 7.2.6.2.3 收后效果

在接收到 APSME-ACCESS-CONTROL.request 原语后，如果 Action 参数设置为 INITIATE，APSME 启动访问控制过程。本地的 APSME 作为这个过程的发起者，ResponderAddress 参数指示的 APSME 作为这个过程的响应者。如果 Action 参数设置为 RESPOND_ACCEPT，APSME 参加访问控制。如果 Action 参数设置为 RESPOND_REJECT，不会发生访问控制过程。

### 7.2.6.3 APSME-ACCESS-CONTROL.confirm

#### 7.2.6.3.1 服务原语的定义

APSME-ACCESS-CONTROL.confirm 用于响应设备的 APSME 向其邻近高层通知访问完成或失败。

该原语应提供以下接口：

```
APSME-ACCESS-CONTROL.confirm    {
                                    Address
                                    Status
                                    }
```

表 49 列出了 APSME-ACCESS-CONTROL.confirm 原语的参数。

**表 49 APSME-ACCESS-CONTROL.confirm 原语的参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| Address | 地址 | 任何有效的 64 位 IEEE 地址 | 访问控制发生设备的 64 位 IEEE 地址 |
| Status | 状态 | 从 NLDE-DATA.confirm 原语返回的值 | 访问控制的最终状态 |



#### 7.2.6.3.2 产生条件

完成访问控制后，访问控制响应者的 APSME 发出该原语到它的邻近高层。

#### 7.2.6.3.3 收后效果

接收到该原语后，响应者的邻近高层获知发起者的访问结果。如果传输成功，Status 参数会设置为 SUCCESS，否则，Status 参数会显示错误。

### 7.2.6.4 APSME-ACCESS-CONTROL.indication

#### 7.2.6.4.1 服务原语的定义

APSME-ACCESS-CONTROL.indication 原语用于响应者的 APSME 向其邻近高层通知接收到发起者的访问信息。

该原语应提供以下接口：

```
APSME-ACCESS-CONTROL.indication    {
                                       InitiatorAddress
                                       AccessObject
                                       AccessControlMethod
                                       }
```

表 50 列出了 APSME-ACCESS-CONTROL.indication 原语的参数。

**表 50 APSME-ACCESS-CONTROL.indication 原语的参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| InitiatorAddress | 地址 | 列举任何有效的 64 位 IEEE 地址 | 发起设备的 64 位 IEEE 地址 |
| AccessObject | 整型 | 0x00～0x01 | 请求访问的对象应是以下之一：<br>0x00:数据资源<br>0x01:节点资源 |
| AccessControlMethod | 整型 | 0x00～0x01 | 请求访问控制的方法应是以下之一：<br>0x00:自主访问控制方法<br>0x01:强制访问控制方法 |

#### 7.2.6.4.2 产生条件

当响应设备收到了来自于发起者的访问请求时，它的 APSME 向其邻近高层发送该原语。

#### 7.2.6.4.3 收后效果

接收到 APSME-ACCESS-CONTROL.indication 原语后，邻近高层会决定它是否参加 InitiatorAddress 参数指定的发起者的访问控制，并用 APSME-ACCESS-CONTROL.request 原语响应。如果参加发起者的访问过程，Action 参数设置为 RESPOND_ACCEPT，相应的 APSME-ACCESS-CONTROL.request 原语的 AccessControlMethod 参数设置为 0x00 或 0x01。如果它不希望参加发起者的访问过程，Action 参数设置为 RESPOND_REJECT。

### 7.2.7 身份鉴别服务

#### 7.2.7.1 概述

身份鉴别服务是指 APSME 允许两个设备之间进行身份鉴别。

#### 7.2.7.2 APSME-IDENTITY-AUTHENTICATE.request

##### 7.2.7.2.1 服务原语的语义

当需要和另一个设备进行身份鉴别时使用 APSME-IDENTITY-AUTHENTICATE.request 原语。

一个设备作为发起设备，另外一个设备作为响应设备。

该原语应提供以下接口：

```
APSME-IDENTITY-AUTHENTICATE.request    {
                                       DestAddress
                                       AuthenticateMethod
                                       RandomChallenge
                                       }
```

表 51 列出了 APSME-IDENTITY-AUTHENTICATE.request 原语的参数。

**表 51　APSME-IDENTITY-AUTHENTICATE.request 的参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| DestAddress | 地址 | 任何有效的 64 位 IEEE 地址 | 目的设备的 64 位 IEEE 地址 |
| AuthenticateMethod | 整型 | 0x00～0x03 | 表明请求鉴别的方法：<br>0x00：基于异或的鉴别方法<br>0x01：基于杂凑运算的鉴别方法<br>0x02：基于分组密码算法的鉴别方法<br>0x03：基于非对称密码算法的鉴别方法 |
| RandomChallenge | 16 个八位位组的集合 | — | 从发起者收到的 16 个八位位组的随机质疑 |

#### 7.2.7.2.2　产生条件

当发起者或者响应者启动或响应身份鉴别时，它的邻近高层发出或响应该原语。

#### 7.2.7.2.3　收后效果

接收到 APSME-IDENTITY-AUTHENTICATE. request 原语后，APSME 根据 Authenticate Method 参数值启动相应的鉴别机制。本地的 APSME 为该身份鉴别的发起者，DestAddress 参数指示的 APSME 为该身份鉴别的响应者。

### 7.2.7.3　APSME-IDENTITY-AUTHENTICATE.confirm

#### 7.2.7.3.1　服务原语的语义

设备身份鉴别成功或失败后会发送 APSME-IDENTITY-AUTHENTICATE. confirm 给邻近高层。

该原语应提供以下接口：

```
APSME-IDENTITY-AUTHENTICATE.confirm    {
                                       Address
                                       Status
                                       }
```

表 52 列出了 APSME-IDENTITY-AUTHENTICATE.confirm 原语的参数。

**表 52　APSME-IDENTITY-AUTHENTICATE.confirm 原语的参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| Address | 地址 | 任何有效的 64 位 IEEE 地址 | 身份鉴别发起设备的 64 位 IEEE 地址 |
| Status | 状态 | NLDEDATA.confirm 原语返回的值 | 发起身份鉴别的结果 |

#### 7.2.7.3.2　产生条件

完成身份鉴别后，发起者或者响应者的 APSME 会向它的邻近高层发出该原语。

#### 7.2.7.3.3 收后效果

发起者和响应者的邻近高层接收到 APSME-IDENTITY-AUTHENTICATE.confirm 原语后，如果 Status 参数为 SUCCESS，则传输成功，否则，传输失败。

### 7.2.7.4 APSME-IDENTITY-AUTHENTICATE.indication

#### 7.2.7.4.1 服务原语的语义

当响应者收到来自于发起者的鉴别信息时，它的 APSME 向它的邻近高层发出 APSME-IDENTITY-AUTHENTICATE.indication 原语。

该原语应提供以下接口：

```
APSME-IDENTITY-AUTHENTICATE.indication   {
                                         InitiatorAddress
                                         AuthenticateMethod
                                         RandomChallenge
                                         }
```

表 53 列出了 APSME-IDENTITY-AUTHENTICATE.indication 原语的参数。

**表 53 APSME-IDENTITY-AUTHENTICATE.indication 原语的参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| InitiatorAddress | 地址 | 列举任何有效的 64 位 IEEE 地址 | 发起设备的 64 位 IEEE 地址 |
| AuthenticateMethod | 整型 | 0x00～0x03 | 表明请求鉴别的方法：<br>0x00：基于异或的鉴别方法<br>0x01：基于杂凑运算的鉴别方法<br>0x02：基于分组密码算法的鉴别方法<br>0x03：基于非对称密码算法的鉴别方法 |
| RandomChallenge | 16 个八位位组的集合 | — | 从发起者上收到的 16 个八位位组的随机集合 |



#### 7.2.7.4.2 产生条件

当响应设备收到发起者身份鉴别的请求时，它的 APSME 向邻近高层发出该原语。

#### 7.2.7.4.3 收后效果

接收到 APSME-IDENTITY-AUTHENTICATE. indication 原语后，邻近高层根据 Initiator Address 参数指定的发起者、AuthenticateMethod 参数指定的鉴别方法对设备进行鉴别，并且发出 APSME-IDENTITY-AUTHENTICATE.request 原语响应。

## 7.2.8 消息鉴别服务

### 7.2.8.1 概述

APSME 允许设备进行消息鉴别。

#### 7.2.8.2 APSME-MESSAGE-AUTHENTICATE.request

##### 7.2.8.2.1 服务原语的语义

APSME-MESSAGE-AUTHENTICATE.request 用来启动一个广播消息鉴别，当需要和其他设备进行广播消息鉴别时使用该原语。

一个设备作为发起设备，广播消息的接收设备作为广播消息鉴别的接收设备。发起设备将会启动广播消息鉴别。

该原语应提供以下接口：

```
APSME-MESSAGE-AUTHENTICATE.request   {
                                     PartnerAddress
                                     RandomChallenge
                                     }
```

表 54 列出了 APSME-MESSAGE-AUTHENTICATE.request 原语的参数。

**表 54 APSME-MESSAGE-AUTHENTICATE.request 的参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| PartnerAddress | 地址 | 任何有效的 64 位 IEEE 地址 | 发起设备的 64 位 IEEE 地址 |
| RandomChallenge | 16 个八位位组的集合 | — | 从发起者收到的 16 个八位位组随机质疑 |

##### 7.2.8.2.2 产生条件

当发起者或者响应者需要启动或响消息鉴别时，它的邻近高层产生该原语。

##### 7.2.8.2.3 收后效果

接收到 APSME-MESSAGE-AUTHENTICATE.request 原语后。本地的 APSME 作为该消息鉴别的响应者，PartnerAddress 参数指示的 APSME 作为发起者。

#### 7.2.8.3 APSME-MESSAGE-AUTHENTICATE.confirm

##### 7.2.8.3.1 服务原语的语义

在设备广播消息鉴别成功或失败后发送 APSME-MESSAGE-AUTHENTICATE.confirm 原语给邻居高层。

该原语应提供以下接口：

```
APSME-MESSAGE-AUTHENTICATE.confirm   {
                                     Address
                                     Status
                                     }
```

表 55 列出了 APSME-MESSAGE-AUTHENTICATE.confirm 原语的参数。

**表 55 APSME-IDENTITY-AUTHENTICATE.confirm 原语的参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| Address | 地址 | 任何有效的 64 位 IEEE 地址 | 消息鉴别发起设备的 64 位 IEEE 地址 |
| Status | 状态 | NLDE-DATA.confirm 原语返回的值 | 发起消息鉴别的结果 |

#### 7.2.8.3.2 产生条件

在完成消息鉴别之后，发起者或者响应者的 APSME 向它的邻近高层发出该原语到它的邻近高层。

#### 7.2.8.3.3 收后效果

发起者和响应者的邻近高层接收到 APSME-MESSAGE-AUTHENTICATE.confirm 原语后，如果 Status 参数为 SUCCESS，则传输成功，否则，传输失败。

### 7.2.8.4 APSME-MESSAGE-AUTHENTICATE.indication

#### 7.2.8.4.1 服务原语的语义

当响应者收到发起者的鉴别信息时，它的 APSME 向邻近高层发出 APSME-MESSAGE-AUTHENTICATE.indication 原语。

该原语应提供以下接口：

```
APSME-MESSAGE-AUTHENTICATE.indication {
                                        InitiatorAddress
                                        RandomChallenge
                                        }
```

表 56 列出了 APSME-MESSAGE-AUTHENTICATE.indication 原语的参数。

**表 56 APSME-MESSAGE-AUTHENTICATE.indication 原语的参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| InitiatorAddress | 地址 | 列举任何有效的 64 位 IEEE 地址 | 发起设备的 64 位 IEEE 地址 |
| RandomChallenge | 16 个八位位组的集合 | — | 从发起者上收到的 16 个八位位组的随机集合 |

#### 7.2.8.4.2 产生条件

当响应设备收到开始鉴别的请求时，它的 APSME 向它的邻近高层发出该原语。

#### 7.2.8.4.3 收后效果

响应设备的邻近高层接收到 APSME-MESSAGE-AUTHENTICATE.indication 原语后，根据 InitiatorAddress 参数指定的发起者对广播消息进行鉴别，并且发出 APSME-MESSAGE-AUTHENTICATE.request 原语，其中 RandomChallenge 参数根据表 53 进行设置。

## 7.2.9 安全数据融合服务

### 7.2.9.1 APSME-SECURE-DATA-AGGREGATION-START.request

#### 7.2.9.1.1 服务原语的语义

当需要使用安全数据融合机制时使用 APSME-SECURE-DATA-AGGREGATION-START.request 原语。

网络协调器将会作为发起设备，另外两个设备将会作为响应设备(响应设备包括融合设备和监督设

备，且融合设备与监督设备为不同设备）。

该原语应提供以下接口：

```
APSME-SECURE-DATA-AGGREGATION-START.request    {
                                                AggregationNodeAddress
                                                SupervisorNodeAddress
                                                AggregationCycleTime
                                                }
```

表 57 列出了 APSME-SECURE-DATA-AGGREGATION-START.request 原语的参数。

**表 57　APSME-SECURE-DATA-AGGREGATION-START.request 参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| AggregationNodeAddress | 地址 | 任何有效的 64 位 IEEE 地址 | 融合设备的扩展 64 位 IEEE 地址 |
| SupervisorNodeAddress | 地址 | 任何有效的 64 位 IEEE 地址 | 监督设备的扩展 64 位 IEEE 地址 |
| AggregationCycleTime | 整型 | 有效并合理的时间长度 | 融合周期与监督周期一致，并且大于或等于簇内普通节点的通信周期，确保融合过程和监督过程能够完成 |

#### 7.2.9.1.2　产生条件

当网络协调器启用安全数据融合机制时，它的上层产生本原语，指定有效的融合设备与监督设备，并且给定有效的融合周期（即监督周期）。

#### 7.2.9.1.3　收后效果

接收到 APSME-SECURE-DATA-AGGREGATION-START.request 原语后，本地的 APSME 作为本服务的发起设备，参数 AggregationNodeAddress 和 SupervisorNodeAddress 指明的 APSME 作为本服务的响应设备，监督周期为 AggregationCycleTime 参数的值。

### 7.2.9.2　APSME-SECURE-DATA-AGGREGATION-START.confirm

#### 7.2.9.2.1　服务原语的语义

在安全融合服务启动成功或失败之后发起设备和响应设备发送 APSME-SECURE-DATA-AGGREGATION-START.confirm 原语给邻近高层。

该原语应提供以下接口：

```
APSME-SECURE-DATA-AGGREGATION-START.confirm    {
                                                Address
                                                Status
                                                }
```

表 58 列出了 APSME-SECURE-DATA-AGGREGATION-START.confirm 原语的参数。

**表 58　APSME-SECURE-DATA-AGGREGATION-START.confirm 参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| Address | 地址 | 任何有效的 64 位 IEEE 地址 | 融合设备或监督设备的 64 位 IEEE 地址 |
| Status | 状态 | NLDE-DATA.confirm 原语返回的值 | 启动安全数据融合的结果 |

#### 7.2.9.2.2 产生条件

安全数据融合服务启动过程完成之后，响应设备和发起设备的 APSME 发出该原语给邻近高层。

#### 7.2.9.2.3 收后效果

如果安全数据融合服务启动成功，发起设备和响应设备开始执行安全数据融合过程。

### 7.2.9.3 APSME-SECURE-DATA-AGGREGATION-START.indication

#### 7.2.9.3.1 服务原语的语义

当响应设备从发起设备接收到安全数据融合服务启动的请求时，它的 APSME 发出该原语给它的邻近高层。

该原语应提供以下接口：

```
APSME-SECURE-DATA-AGGREGATION-START.indication  {
                                                InitiatorAddress
                                                Function
                                                Time
                                                }
```

表 59 列出了 APSME-SECURE-DATA-AGGREGATION-START.indication 原语的参数。

**表 59 APSME-SECURE-DATA-AGGREGATION-START.indication 参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| InitiatorAddress | 地址 | 任何有效的 64 位 IEEE 地址 | 发起设备的扩展 64 位 IEEE 地址 |
| Function | 整型 | 0x00～0x01 | 功能应该是下列之一：<br>0x00：融合功能<br>0x01：监督功能 |
| Time | 整型 | 有效并合理的时间长度 | 接收到的融合/监督周期 |

#### 7.2.9.3.2 产生条件

当从一个发起设备收到启动安全数据融合服务的请求，并且和发起者相关的对密钥存在于 AIB 中时，响应设备的 APSME 应该发出这个原语给邻近高层。

#### 7.2.9.3.3 收后效果

当收到 APSME-SECURE-DATA-AGGREGATION-START.indication 原语后，邻近高层会使用 InitiatorAddress、Function 和 Time 参数确定是否具备完成该功能的条件，并使用 APSME-SECURE-DATA-AGGREGATION-START.response 原语响应。

### 7.2.9.4 APSME-SECURE-DATA-AGGREGATION-START.response

#### 7.2.9.4.1 服务原语的语义

响应设备的邻近高层使用 APSME-SECURE-DATA-AGGREGATION-START.response 原语去响应 APSME-SECURE-DATA-AGGREGATION-START.indication 原语。邻近高层决定是否启动发起者所要求的在安全数据融合服务过程中所承担的融合或监督功能，这个决定在 APSME-SECURE-

DATA-AGGREGATION-START.response 原语 Accept 参数中指明。

该原语应提供以下的接口：

APSME-SECURE-DATA-AGGREGATION-START.response {
InitiatorAddress
Function
Accept
}

表 60 列出了 APSME-SECURE-DATA-AGGREGATION-START.response 原语的参数。

**表 60 APSME-SECURE-DATA-AGGREGATION-START.response 参数**

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| InitiatorAddress | 地址 | 任何有效的 64 位 IEEE 地址 | 发起设备的 64 位 IEEE 地址 |
| Function | 整型 | 0x00～0x01 | 功能应该是下列之一：<br>0x00：融合功能<br>0x01：监督功能 |
| Accept | 布尔 | TRUE 或 FALSE | 响应应该为：<br>TRUE：接受<br>FALSE：拒绝 |

#### 7.2.9.4.2 产生条件

在发起者启动一个安全数据融合服务请求(即收到 APSME-SECURE-DATA-AGGREGATION-START.indication)之后，发起者的邻近高层向 APSME 发出 APSME-SECURE-DATA-AGGREGATION-START.response 原语。

#### 7.2.9.4.3 收后效果

如果 Accept 参数是 TRUE，那么响应设备的 APSME 会尝试按照 AggregationCycleTime 参数所指定的融合/监督周期开始执行融合/监督功能。如果 Function 参数为 0，执行融合功能；如果 Function 参数为 1，则执行监督功能。本地的 APSME 会作为该服务的响应者，InitiatorAddress 指明的 APSME 会作为该服务的发起者。如果 Accept 参数为 FALSE，本地的 APSME 会终止并清除有关未定的安全数据融合启动过程的中间数据。

### 7.2.9.5 APSME-SECURE-DATA-AGGREGATION-REVOKE.request

#### 7.2.9.5.1 服务原语的语义

当发起设备判定融合设备融合信息不可信时，它的邻近高层应该发出这个原语给本地 APSME，下发报文撤销融合节点以及对应的监督节点。

该原语应提供以下的接口：

APSME-SECURE-DATA-AGGREGATION-REVOKE.request {
AggregationNodeAddress
SupervisorNodeAddress
}

表 61 列出了 APSME-SECURE-DATA-AGGREGATION-REVOKE.request 原语的参数。

表 61　APSME-SECURE-DATA-AGGREGATION-REVOKE.request 参数

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| AggregationNodeAddress | 地址 | 任何有效的 64 位 IEEE 地址 | 融合设备的扩展 64 位 IEEE 地址 |
| SupervisorNodeAddress | 地址 | 任何有效的 64 位 IEEE 地址 | 监督设备的扩展 64 位 IEEE 地址 |

#### 7.2.9.5.2　产生条件

当发起设备判定融合设备融合信息不可信时，它的邻近高层应该发出这个原语给本地 APSME。

#### 7.2.9.5.3　收后效果

本地 APSME 接收到 APSME-SECURE-DATA-AGGREGATION-REVOKE.request 原语后，下发命令报文撤销融合节点与对应的监督节点。

### 7.2.9.6　APSME-SECURE-DATA-AGGREGATION-REVOKE.indication

#### 7.2.9.6.1　服务原语

当响应设备 APSME 接收到撤销融合节点以及对应的监督节点命令后，本地 APSME 应该发出这个原语给邻近高层。

该原语应提供以下的接口：

```
APSME-SECURE-DATA-AGGREGATION-REVOKE.indication {
                                        InitiatorAddress
                                        AggregationNodeAddress
                                        SupervisorNodeAddress
                                        Command
                                        }
```

表 62 列出了 APSME-SECURE-DATA-AGGREGATION-REVOKE.indication 原语的参数。

表 62　APSME-SECURE-DATA-AGGREGATION-REVOKE.indication 参数

| 参数名称 | 类型 | 有效范围 | 描述 |
|---|---|---|---|
| InitiatorAddress | 地址 | 任何有效的 64 位 IEEE 地址 | 发起设备的扩展 64 位 IEEE 地址 |
| AggregationNodeAddress | 地址 | 任何有效的 64 位 IEEE 地址 | 融合设备的扩展 64 位 IEEE 地址 |
| SupervisorNodeAddress | 地址 | 任何有效的 64 位 IEEE 地址 | 监督设备的扩展 64 位 IEEE 地址 |
| Command | 布尔 | 0x00～0x01 | 撤销融合/监督功能<br>0x00：撤销融合功能<br>0x01：撤销监督功能 |

#### 7.2.9.6.2　产生条件

当响应设备 APSME 接收到撤销融合节点以及对应的监督节点命令后，本地 APSME 产生该原语发送给邻近高层。

#### 7.2.9.6.3　收后效果

响应设备邻近高层接收到本原语之后，结束安全数据融合过程，结束融合/监督功能。

## 7.3 帧安全

### 7.3.1 帧安全概述

应用支持子层负责为应用层和邻近高层提供密钥建立、密钥传输和设备管理服务。应用支持子层帧的安全结构如图 10 所示。

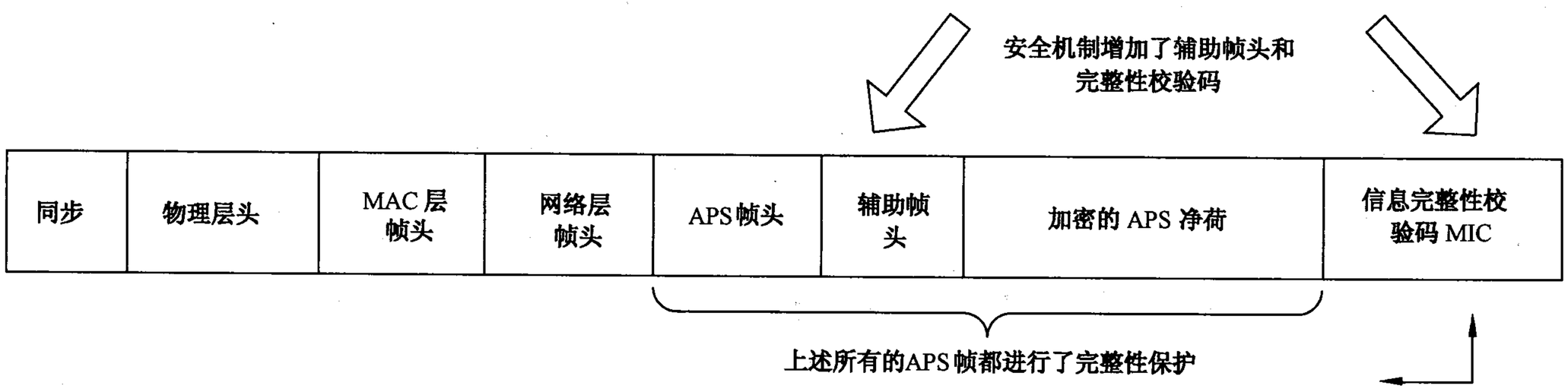


**图 10 APS 层帧的安全结构**

### 7.3.2 应用支持子层帧

应用支持子层的帧格式包括一个应用支持子层帧头和一个应用支持子层净荷域。

当 ASDU 帧启用安全保护时，初始的 APS 帧头中的安全使能位为 1，即表示辅助帧头是存在的。加密的应用支持子层帧的格式如图 11 所示。

| 八位位组数：可变 | 14 | 可变 | |
|---|---|---|---|
| 初始的 APS 帧头 | 辅助帧头 | 加密的净荷 | 加密的信息完整性校验码 MIC |
| | | APS 帧的安全载荷＝CCM＊的输出 | |
| 完整的 APS 帧头 | | 安全的 APS 净荷 | |



**图 11 安全的应用支持子层帧格式**

### 7.3.3 辅助帧头

#### 7.3.3.1 格式

辅助帧头的格式如图 12 所示：

| 八位位组数：1 | 4 | 0/8 | 0/1 |
|---|---|---|---|
| 安全控制 | 帧计数器 | 源地址 | 密钥序列号 |

**图 12 辅助帧头的格式**

#### 7.3.3.2 安全控制域

##### 7.3.3.2.1 格式

安全控制域包含安全级别、密钥标识符和扩展的临时子域，其格式如图 13 所示。

| 位:0～2 | 3～4 | 5 | 6～7 |
|---|---|---|---|
| 安全级别 | 密钥标识符 | 扩展临时子域 | 保留 |

图 13　安全控制域的格式

#### 7.3.3.2.2　安全级别子域

安全级别标识符规定了如何对输出帧和输入帧进行保护(安全等级见 GB/T 30269.601—2016)。安全级别子域的值如表 63 所示。

表 63　应用支持子层可用的安全级别

| 安全级别标识符 | 安全级别子域 | 安全等级 |
|---|---|---|
| 0x00 | ‘000’ | 第一级 |
| 0x01 | ‘001’ | 第二级 |
| 0x02 | ‘010’ | 第三级 |
| 0x03 | ‘011’ | 第四级 |
| 0x04 | ‘100’ | 第五级 |
| 0x05～0x07 | ‘101’～‘111’ | 保留 |

#### 7.3.3.2.3　密钥标识符子域

密钥标识符子域用来标识保护帧所用的密钥。它的编码如表 64 所示。

表 64　密钥标识符子域的编码表

| 密钥标识符 | 密钥标识符子域 | 描述 |
|---|---|---|
| 0x00 | ‘00’ | 共享密钥 |
| 0x01 | ‘01’ | 网络密钥 |

#### 7.3.3.2.4　扩展临时子域

如果辅助帧头中存在源地址域,那么扩展临时子域设置为 1,否则设置为 0。

#### 7.3.3.3　计数器域

计数器域是为了防止对帧的重复处理。

#### 7.3.3.4　源地址域

仅当扩展临时子域的值为 1 时,辅助帧头中才包含源地址域,源地址域对网络层帧提供安全保护设备的 64 位 IEEE 地址。

#### 7.3.3.5　密钥序列号域

仅当密钥标识符子域的值为 1(对应网络密钥)时,辅助帧头中才包含密钥序列号域,密钥序列号域对网络层帧提供安全保护的网络密钥的序列号。

## 7.4 命令帧

### 7.4.1 概述

所有的应用支持子层命令帧除非明文规定都通过加密发送，应用支持子层命令帧标识符如表 65 所示。

**表 65 应用支持子层命令帧**

| 命令帧标识符 | 命令名称 | 章节号 |
|---|---|---|
| 0x01 | 密钥建立命令帧 | 7.4.2 |
| 0x02 | 密钥更新命令帧 | 7.4.3 |
| 0x03 | 密钥撤销命令帧 | 7.4.4 |
| 0x04 | 访问控制请求命令帧 | 7.4.5 |
| 0x05 | 访问控制响应命令帧 | 7.4.6 |
| 0x06 | 身份鉴别请求命令帧 | 7.4.7 |
| 0x07 | 身份鉴别响应命令帧 | 7.4.8 |
| 0x08 | 身份鉴别响应确认命令帧 | 7.4.9 |
| 0x09 | 广播消息鉴别命令帧 | 7.4.10 |
| 0x10 | 安全数据融合启动请求命令帧 | 7.4.11 |
| 0x11 | 融合设备响应命令帧 | 7.4.12 |
| 0x12 | 监督设备响应命令帧 | 7.4.13 |
| 0x13 | 安全数据启动请求确认命令帧 | 7.4.14 |
| 0x14 | 安全数据融合撤销命令帧 | 7.4.15 |
| 0x15～0xFF | 保留 | — |



应用支持子层命令帧帧类型如表 66 所示。

**表 66 帧类型列表**

| 帧类型 | 描述 |
|---|---|
| APS_CMD_DIRECT_1 | 基于密钥池分发方法广播的命令帧类型 |
| APS_CMD_DIRECT_2 | 基于多项式分发方法广播的命令帧类型 |
| APS_CMD_PATHKEY_1 | 同一簇内建立路径密钥方法的命令帧，表示密钥建立中的第一种报文类型 |
| APS_CMD_PATHKEY_2 | 同一簇内建立路径密钥方法的命令帧，表示密钥建立中的第二种报文类型 |
| APS_CMD_PATHKEY_3 | 同一簇内建立路径密钥方法的命令帧，表示密钥建立中的第三种报文类型 |
| APS_CMD_PATHKEY_4 | 同一簇内建立路径密钥方法的命令帧，表示密钥建立中的第四种报文类型 |

表 66（续）

| 帧类型 | 描述 |
| --- | --- |
| APS_CMD_PATHKEY_5 | 不同簇实体建立路径密钥方法的命令帧，表示密钥建立中的第一种报文类型 |
| APS_CMD_PATHKEY_6 | 不同簇实体建立路径密钥方法的命令帧，表示密钥建立中的第二种报文类型 |
| APS_CMD_PATHKEY_7 | 不同簇实体建立路径密钥方法的命令帧，表示密钥建立中的第三种报文类型 |
| APS_CMD_PATHKEY_8 | 不同簇实体建立路径密钥方法的命令帧，表示密钥建立中的第四种报文类型 |
| APS_CMD_CERTIFICATE | 基于证书密钥建立方法的命令帧类型 |
| APS_CMD_UPDATE_KEY | 帧类型为密钥更新命令帧 |
| APS_CMD_REVOCATION_KEY | 帧类型为密钥撤销命令帧 |
| APS_CMD_REQUEST_ACCESS_COMTROL | 帧类型为请求访问控制帧 |
| APS_CMD_RESPOND_ACCESS_COMTROL | 帧类型为响应访问控制帧 |



### 7.4.2 密钥建立命令

#### 7.4.2.1 格式

所有的密钥建立命令帧不进行加密发送，密钥建立命令帧格式如图 14。

| 八位位组数:1 | 1 | 1 | 8 | 8 | 1 | 16 |
| --- | --- | --- | --- | --- | --- | --- |
| 帧控制 | APS 计数器 | APS 命令标识符 | 发起者地址 | 响应者地址 | 密钥类型 | 数据 |
| APS 头 | | 负载 | | | | |

图 14 密钥建立帧命令格式

#### 7.4.2.2 命令标识符域

命令标识符域表示应用支持子层命令类型。对于基于密钥池分发方法的命令帧，命令标识符表示广播的密钥标识消息，帧类型为 APS_CMD_DIRECT_1，该命令帧不进行加密处理；对于基于多项式分发方法的命令帧，命令标识符表示广播的设备标识消息，帧类型为 APS_CMD_DIRECT_2，该命令帧不进行加密处理；对于同一簇内建立路径密钥方法的命令帧，命令标识符表示其密钥建立过程中发送的 4 种报文类型，帧类型分别为 APS_CMD_PATHKEY_1、APS_CMD_PATHKEY_2、APS_CMD_PATHKEY_3 和 APS_CMD_PATHKEY_4；对于不同簇实体建立路径密钥方法的命令帧，命令标识符表示其密钥建立过程中发送的 4 种报文类型，帧类型分别为 APS_CMD_PATHKEY_5、APS_CMD_PATHKEY_6、APS_CMD_PATHKEY_7 和 APS_CMD_PATHKEY_8；对于基于证书密钥建立方法的命令帧，命令标识符表示发送的密钥协商材料，帧类型为 APS_CMD_CERTIFICATE，该命令帧不进行加密处理。

#### 7.4.2.3 发起者地址域

发起者地址域是密钥建立协议发起设备的 64 位 IEEE 地址。

#### 7.4.2.4 响应者地址域

响应者地址域是密钥建立协议响应设备的 64 位 IEEE 地址。

#### 7.4.2.5 数据域

数据域的内容取决于命令标识符域。

### 7.4.3 密钥更新命令

#### 7.4.3.1 格式

密钥更新命令帧格式如图 15 所示。

| 八位位组数:1 | 1 | 1 | 8 | 8 | 1 | 可变的 |
|---|---|---|---|---|---|---|
| 帧控制 | APS 计数器 | APS 命令标识符 | 发起者地址 | 响应者地址 | 密钥类型 | 数据 |
| APS 头 | | 负载 | | | | |

**图 15 Update-Key 命令帧**

#### 7.4.3.2 命令标识符域

命令标识符域表示密钥更新命令帧类型(APS_CMD_UPDATE_KEY)。

#### 7.4.3.3 发起者地址域

发起者地址域是密钥建立协议发起设备的 64 位 IEEE 地址。

#### 7.4.3.4 响应者地址域

响应者地址域是密钥建立协议响应设备的 64 位 IEEE 地址。

#### 7.4.3.5 密钥类型域

##### 7.4.3.5.1 概述

密钥类型域长度为 8 位,描述了正在传输中的密钥类型。

##### 7.4.3.5.2 初始化密钥材料、共享密钥或会话密钥描述符域

当密钥类型域设置为 0x00、0x02 或 0x03 时,密钥描述符域应该使用如图 16 的格式。其中密钥标识符应包含新密钥的标识符,密钥子域应包含新的密钥消息。

| 八位位组数:可变的 | 可变的 |
|---|---|
| 密钥标识符 | 密钥 |

**图 16 在 Update-Key 命令中的初始化密钥材料、共享密钥或会话密钥描述符**

#### 7.4.3.5.3 初始化值描述符域

当密钥类型子域为 0x01 时，格式如图 17 所示。

| 八位位组数:可变的 |
| --- |
| 数据 |

**图 17 在 Update-Key 命令中的初始化值描述符域**

### 7.4.4 密钥撤销命令

#### 7.4.4.1 格式

密钥撤销命令帧格式如图 18 所示。

| 八位位组数:1 | 1 | 1 | 8 | 8 | 可变的 | 可变的 | 可变的 |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 帧控制 | APS 计数器 | 命令标识符 | 发起者地址 | 响应者地址 | 密钥标识符 | 时间 | 原因 |
| APS 头 | | 负载 | | | | | |

**图 18 REVOCATION-Key 命令帧的格式**

#### 7.4.4.2 命令标识符域

命令标识符域应指明密钥撤销应用支持子层命令类型(APS_CMD_REVOCATION_KEY)。

#### 7.4.4.3 发起者地址域

发起者地址域是密钥建立协议发起设备的 64 位 IEEE 地址。

#### 7.4.4.4 响应者地址域

响应者地址域是密钥建立协议响应设备的 64 位 IEEE 地址，如果向全网发送密钥撤销命令，则响应者地址为广播地址。

#### 7.4.4.5 密钥标识符域

密钥标识符域应包含被撤销密钥的标识符。

#### 7.4.4.6 密钥撤销时间域

密钥撤销时间域应包含被撤销密钥的日期时间。

#### 7.4.4.7 密钥撤销原因域

密钥标识符域应包含撤销密钥的原因。

### 7.4.5 访问控制请求命令

#### 7.4.5.1 格式

访问控制过程中发起者发给响应者的命令帧，格式如图 19 所示。

| 八位位组数:1 | 1 | 1 | 8 | 8 | 1 |
|---|---|---|---|---|---|
| 帧控制 | APS 计数器 | APS 命令标识符 | 发起者 | 响应者 | 访问对象 |
| APS 头 | | 负载 | | | |

图 19　访问控制请求命令帧格式

#### 7.4.5.2　命令标识符域

命令标识符域应指示访问控制请求应用支持子层帧类型(APS_CMD_REQUEST_ACCESS_COMTROL)。命令标识符域的值如表 67 所示。

表 67　命令标识符的值

| 值 | 描述 |
|---|---|
| 0x00 | 自主访问控制方法 |
| 0x01 | 强制访问控制方法 |

#### 7.4.5.3　发起者域

发起者域设置为发起设备的 64 位 IEEE 地址或用户的身份标识。

#### 7.4.5.4　响应者域

响应者域设置为响应设备的 64 位 IEEE 地址。

#### 7.4.5.5　访问对象标识符域

访问对象域指示访问控制的对象,如表 68 所示。

表 68　访问对象的值

| 值 | 访问对象 |
|---|---|
| 0x00 | 数据 |
| 0x01 | 节点 |

### 7.4.6　访问控制响应命令

#### 7.4.6.1　格式

访问控制响应命令帧格式如图 20 所示。

| 八位位组数:1 | 1 | 1 | 8 | 8 |
|---|---|---|---|---|
| 帧控制 | APS 计数器 | APS 命令标识符 | 发起者 | 响应者 |
| APS 头 | | 负载 | | |

图 20　访问控制响应命令帧格式

#### 7.4.6.2 命令标识符域

命令标识符域应指示访问控制响应应用支持子层帧的类型(APS_CMD_ RESPOND_ACCESS_COMTROL)。

#### 7.4.6.3 发起者域

发起者域的值设置为发起设备的 64 位 IEEE 地址。

#### 7.4.6.4 响应者域

响应者域的值设置为响应设备的 64 位 IEEE 地址。

### 7.4.7 身份鉴别请求命令

#### 7.4.7.1 格式

身份鉴别请求命令帧格式如图 21 所示。

| 八位位组数:1 | 1 | 1 | 8 | 8 | 16 | 可变的 |
|---|---|---|---|---|---|---|
| 帧控制 | APS 计数器 | APS 命令标识符 | 源地址 | 目的地址 | 质疑 | 鉴别选项 |
| APS 头 | | 负载 | | | | |

**图 21 身份鉴别请求命令帧格式**

#### 7.4.7.2 命令标识符域

命令标识符域的值表示应用支持子层命令类型。应用支持子层命令标识符域的值如表 69 所示。

**表 69 应用支持子层命令标识符域的值**

| 值 | 描述 |
|---|---|
| 0x00 | 标识使用的身份鉴别方法是基于异或运算的鉴别方法 |
| 0x01 | 标识使用的身份鉴别方法是基于杂凑运算的鉴别方法 |
| 0x02 | 标识使用的身份鉴别方法是基于分组密码算法的鉴别方法 |
| 0x03 | 标识使用的身份鉴别方法是基于非对称密码算法的鉴别方法 |

#### 7.4.7.3 源地址域

发起者地址域的值是身份鉴别发起设备的 64 位扩展地址。

#### 7.4.7.4 目的地址域

响应者地址域的值是身份鉴别响应设备的 64 位扩展地址。

#### 7.4.7.5 质疑域

质疑域的值为发起者产生的质疑八位位组。

#### 7.4.7.6 鉴别选项域

鉴别材料域的长度可变,应包含鉴别发起者针对不同身份鉴别机制加入的鉴别材料信息。鉴别材

料域根据命令标识符域设置不同的参数值并一一对应。

### 7.4.8 身份鉴别响应命令

#### 7.4.8.1 格式

身份鉴别响应命令帧格式如图 22 所示。

| | | | | | | |
|---|---|---|---|---|---|---|
| 八位位组数:1 | 1 | 1 | 8 | 8 | 16 | 可变的 |
| 帧控制 | APS 计数器 | APS 命令标识符 | 源地址 | 目的地址 | 质疑 | 鉴别选项 |
| APS 头 | | 负载 | | | | |

**图 22 身份鉴别响应命令帧格式**

#### 7.4.8.2 命令标识符域

命令标识符域指示应用支持子层命令类型。命令标识符的值如表 70 所示。

**表 70 应用支持子层命令标识符域的值**

| 值 | 描述 |
|---|---|
| 0x00 | 标识使用的身份鉴别方法是基于异或运算的鉴别方法 |
| 0x01 | 标识使用的身份鉴别方法是基于杂凑运算的鉴别方法 |
| 0x02 | 标识使用的身份鉴别方法是基于分组密码算法的鉴别方法 |
| 0x03 | 标识使用的身份鉴别方法是基于非对称密码算法的鉴别方法 |



#### 7.4.8.3 源地址域

源地址域的值是身份鉴别响应设备的 64 位扩展地址。

#### 7.4.8.4 目的地址域

目的地址域的值是身份鉴别发起设备的 64 位扩展地址。

#### 7.4.8.5 质疑域

质疑域的值为响应者产生的质疑八位位组。

#### 7.4.8.6 鉴别选项域

鉴别材料域的长度可变,应包含鉴别响应者针对发起者不同身份鉴别机制的鉴别材料响应的鉴别材料信息。

### 7.4.9 身份鉴别响应确认命令

#### 7.4.9.1 格式

身份鉴别响应确认命令帧格式如图 23 所示。

| 八位位组数:1 | 1 | 1 | 8 | 8 | 16 | 可变的 |
|---|---|---|---|---|---|---|
| 帧控制 | APS 计数器 | APS 命令标识符 | 源地址 | 目的地址 | 质疑 | 鉴别选项 |
| APS 头 | | 负载 | | | | |

**图 23　身份鉴别响应确认命令帧格式**

#### 7.4.9.2　命令标识符域

命令标识符域指示应用支持子层命令类型。命令标识符的值类型如表 71 所示。

**表 71　应用支持子层命令标识符域的值**

| 值 | 描述 |
|---|---|
| 0x00 | 标识使用的身份鉴别方法是基于异或运算的鉴别方法 |
| 0x01 | 标识使用的身份鉴别方法是基于杂凑运算的鉴别方法 |
| 0x02 | 标识使用的身份鉴别方法是基于分组密码算法的鉴别方法 |
| 0x03 | 标识使用的身份鉴别方法是基于非对称密码算法的鉴别方法 |

#### 7.4.9.3　源地址域

源地址域的值是身份鉴别发起设备的 64 位扩展地址。

#### 7.4.9.4　响应者地址域

目的地址域的值是身份鉴别响应设备的 64 位扩展地址。

#### 7.4.9.5　质疑域

质疑域应为响应者产生的质疑八位位组表示。

#### 7.4.9.6　鉴别选项域

鉴别材料域的长度可变,应包含鉴别响应者针对发起者不同身份鉴别机制的鉴别材料响应的鉴别材料信息。

### 7.4.10　广播消息鉴别命令

#### 7.4.10.1　格式

广播消息鉴别命令允许设备在需要对广播消息进行鉴别时,向接收节点广播消息鉴别命令。广播消息鉴别命令帧格式如图 24 所示。

| 八位位组数:1 | 1 | 1 | 8 | 8 | 16 |
|---|---|---|---|---|---|
| 帧控制 | APS 计数器 | APS 命令标识符 | 源地址 | 目的地址 | 密钥信息 |
| APS 头 | | 负载 | | | |

**图 24　广播消息鉴别命令帧格式**

#### 7.4.10.2 命令标识符域

命令标识符域应指明设备更新命令类型。

#### 7.4.10.3 源地址域

源地址域的值设置为发起设备的 64 位扩展地址。

#### 7.4.10.4 目的地址域

目的地址域的值设置为广播地址。

#### 7.4.10.5 密钥信息域

密钥信息域应包换发起者针对广播消息延迟公布的密钥信息。

### 7.4.11 安全数据融合启动请求命令

#### 7.4.11.1 格式

安全数据融合启动请求命令帧格式如图 25 所示。

| 八位位组数:1 | 1 | 1 | 8 | 8 | 8 |
|---|---|---|---|---|---|
| 帧控制 | APS 计数器 | APS 命令标识符 | 发起者地址 | 融合设备地址 | 监频设备地址 |
| APS 头 | | 负载 | | | |

**图 25 安全数据融合启动请求命令帧格式**



#### 7.4.11.2 应用支持子层命令标识符域

应用支持子层命令标识符域应指示应用支持子层命令帧类型。

#### 7.4.11.3 发起者地址域

发起者地址域的值设置为安全数据融合服务发起设备(网络协调器)的 64 位 IEEE 地址。

#### 7.4.11.4 融合设备地址域

融合设备地址域的值设置为安全数据融合服务响应设备中作为融合设备的 64 位 IEEE 地址。

#### 7.4.11.5 监督设备地址域

监督设备地址域设置为安全数据融合服务响应设备中作为监督设备的 64 位 IEEE 地址。

### 7.4.12 融合设备响应命令

#### 7.4.12.1 格式

融合设备响应命令帧格式如图 26 所示。

| 八位位组数:1 | 1 | 1 | 8 | 1 | 1 |
|---|---|---|---|---|---|
| 帧控制 | APS 计数器 | APS 命令标识符 | 响应者地址 | 响应者功能 | 响应结果 |
| APS 头 | | 负载 | | | |

**图 26 融合设备响应命令帧格式**

#### 7.4.12.2 命令标识符域

应用支持子层命令标识符域应指示应用支持子层命令帧类型。

#### 7.4.12.3 响应者地址域

响应者地址域的值设置为融合设备的 64 位 IEEE 地址。

#### 7.4.12.4 响应者功能域

响应者功能域的值设置为 0x00,表示融合功能。

#### 7.4.12.5 响应结果域

响应结果域的值应为 TRUE 或者 FALSE。

### 7.4.13 监督设备响应命令

#### 7.4.13.1 格式

监督设备响应命令帧格式如图 27 所示。

| 八位位组数:1 | 1 | 1 | 8 | 1 | 1 |
|---|---|---|---|---|---|
| 帧控制 | APS 计数器 | APS 命令标识符 | 响应者地址 | 响应者功能 | 响应结果 |
| APS 头 | | 负载 | | | |

**图 27 监督设备响应命令帧格式**



#### 7.4.13.2 命令标识符域

应用支持子层命令标识符域应指示应用支持子层命令帧类型。

#### 7.4.13.3 响应者地址域

响应者地址域的值设置为监督设备的 64 位 IEEE 地址。

#### 7.4.13.4 响应者功能域

响应者功能域的值设置为 0x01,表示监督功能。

#### 7.4.13.5 响应结果域

响应结果域的值为 TRUE 或者 FALSE。

### 7.4.14 安全数据启动确认命令

#### 7.4.14.1 格式

安全数据启动确认命令帧格式如图 28 所示。

| 八位位组数:1 | 1 | 1 | 8 | 8 | 8 | 1 |
|---|---|---|---|---|---|---|
| 帧控制 | APS 计数器 | APS 命令标识符 | 发起者地址 | 融合设备地址 | 监频设备地址 | 启动结果 |
| APS 头 | | 负载 | | | | |

**图 28 安全数据启动命令帧格式**

#### 7.4.14.2 命令标识符域

应用支持子层命令标识符域应指示应用支持子层命令帧类型。

#### 7.4.14.3 发起者地址域

发起者地址域的值设置为安全数据融合服务发起设备(网络协调器)的 64 位 IEEE 地址。

#### 7.4.14.4 融合设备地址域

融合设备地址域的值设置为安全数据融合服务响应设备中作为融合设备的 64 位 IEEE 地址。

#### 7.4.14.5 监督设备地址域

监督设备地址域的值设置为安全数据融合服务响应设备中作为监督设备的 64 位 IEEE 地址。

#### 7.4.14.6 启动结果域

启动结果域的值为 TRUE 或者 FALSE。

### 7.4.15 安全数据融合撤销命令

#### 7.4.15.1 格式

安全数据融合撤销命令帧格式如图 29 所示。

| 八位位组数:1 | 1 | 1 | 8 | 8 | 8 |
|---|---|---|---|---|---|
| 帧控制 | APS计数器 | APS命令标识符 | 发起者地址 | 融合设备地址 | 监频设备地址 |
| APS头 | | 负载 | | | |



**图 29 安全数据融合撤销命令帧格式**

#### 7.4.15.2 命令标识符域

应用支持子层命令标识符域应指示应用支持子层命令帧类型。

#### 7.4.15.3 发起者地址域

发起者地址域的值设置为安全数据融合服务发起设备(网络协调器)的 64 位 IEEE 地址。

#### 7.4.15.4 融合设备地址域

融合设备地址域的值设置为安全数据融合服务响应设备中作为融合设备的 64 位 IEEE 地址。

#### 7.4.15.5 监督设备地址域

监督设备地址域的值设置为安全数据融合服务响应设备中作为监督设备的 64 位 IEEE 地址。

## 7.5 安全相关的 AIB 属性

AIB 包括应用支持子层安全管理所需的属性。每个属性都可以分别用 APSME-GET.request 原语读和 APSME-SET.request 原语写。

在应用支持子层 AIB 中包含的与安全相关的属性见表 72 和表 73。其中表 72 列出了应用支持子

层管理所使用的 AIB 总体安全属性，主要针对网络管理者，表 73 为建立密钥时的 AIB 属性，针对通信双方。

**表 72　AIB 总体安全属性**

| 属性 | 标识符 | 类型 | 范围 | 描述 | 默认值 |
|---|---|---|---|---|---|
| apsDeviceKeyPairSet | 0xaa | 密钥对描述符目录的设置 | 可变的 | 一组密钥对描述符，包括自身的密钥信息和与其他设备共享的密钥信息 | — |
| apsTrustCenterAddress | 0xab | 地址 | 任何有效的 64 位地址 | 设备的信任中心的地址 | |
| apsSecurityTimeOutPeriod | 0xac | 整型 | 0x0000<br>～0xFFFF | 一个设备等待一个期待的安全协议帧的时间周期（以毫秒为单位） | 1 000 |

**表 73　建立密钥时的 AIB 属性**

| 名称 | 类型 | 范围 | 描述 |
|---|---|---|---|
| DeviceAddress | 地址 | 任何有效的 64 位 IEEE 地址 | 代表该密钥对共享的实体的地址 |
| Initialization key ID | 可变 | — | 初始化密钥材料标识符 |
| Initialization key | 16 个八位位组集 | — | 初始化密钥材料 |
| Polynomial ID | 可变 | — | 多项式的标识符 |
| Polynomial | 可变 | — | 用于密钥协商的多项式 |
| Initialization value | 可变 | — | 初始化值 |
| Shared key ID | 可变 | — | 共享密钥标识符 |
| Shared key | 16 个八位位组集 | — | 共享密钥值 |
| Session key ID | 可变 | — | 会话密钥标识符 |
| Session key | 16 个八位位组集 | — | 会话密钥值 |

# 附 录 A
# （规范性附录）
# 网络层安全交互过程

## A.1 概述

网络层的安全交互过程包括网络层的帧安全以及路由安全，网络层的帧安全以及路由安全应遵循本附录。

## A.2 帧安全

### A.2.1 概述

安全处理输出和输入的 NWK 帧涉及的步骤分别在 A.2.2 和 A.2.3 中有说明。

### A.2.2 输出帧的安全处理

如果 NWK 层有一个帧需要安全保护并且 NIB 属性中 nwkSecurityLevel>0 时应进行输出帧的安全处理，如果是一个 NWK 数据帧，且 GB/T 30269.301—2014 的网络层服务 NLDE-DATA.request 原语的 SecurityEnabled 参数值为 TRUE，应按照如下步骤实施安全：

a) 从 NIB 属性 nwkSecurityMaterialSet 中获得 nwkActiveKeySeqNumber，使用它去检索网络密钥、输出帧计数器（OutgoingFrameCounter）和密钥序列号（KeySeqNumber）。从 nwkSecurityLevel 获得安全等级。如果输出帧计数器值为 4 个八位位组表示的整数 $2^{32}-1$，或如果密钥不能获得，则安全处理失败，不能对本帧进行进一步安全处理。

b) 建立辅助头 AuxiliaryHeader：

1) 安全控制域应该如下设置：

——安全等级子域应该设置为步骤 a)中获得的安全等级。

——密钥标识符子域应该设置为‘01’。

——扩展的临时子域应该设置为 1。

2) 设置源地址域为本地设备的 64 位 IEEE 地址。

3) 帧计数器域设置为步骤 a)中的输出帧计数器读数。

4) 密钥序列号域应该设置为步骤 a)获得的序列号。

c) 执行 CCM * 模式（见 GB/T 15629.15—2010）加密和认证操作：

1) 参数 M 从表 2 对应于第 a)步的安全级别中获得。

2) 位字符串 Key 应从第 a)步获得。

3) 该临时 N 应是 13 个八位位组构成的字符串，使用第 1)步安全控制域、第 4)步获得的帧计数器域、第 3)步获得的源地址域构成。

4) 如果安全等级需要加密，八位位组构成的字符串 a 应该是字符串 NwkHeader||AuxiliaryHeader，八位位组构成的字符串 m 应该是字符串 Payload。否则，长度为 8 个八位位组的字符串 a 是字符串 NwkHeader||AuxiliaryHeader||Payload，m 则是长度为零的字符串。

d) 如果步骤 c)调用的 CCM * 模式输出“无效”，安全处理失败，本帧不能进一步进行安全处理。

e) 设步骤 c)的输出为 C(out)。如果安全等级需要加密，加密的输出帧为 NwkHeader||AuxiliaryHeader||C(out)，否则，加密的输出帧为 NwkHeader||AuxiliaryHeader||Payload||C(out)。

f) 如果加密的输出帧大于 aMaxMACFrameSize，安全处理失败，本帧不能进一步进行安全处理。

g) 第一步获得的输出帧计数器应该增加一，并且储存在网络安全材料描述符的 OutgoingFrameCounter 元素中，可由 NIB 属性 nwkActiveKeySeqNumber 引用；即这个和保护帧使用的密钥相联系的的输出帧计数器值被更新。

h) 安全控制域的安全等级子域重置为 3 位的全零字符串‘000’。

### A.2.3 输入帧的安全处理

当 NWK 层收到一个加密帧（由网络层帧头、辅助头和负载构成），且网络层帧头帧控制域的安全子域指明，应该进行如下安全处理：

a) 从 NIB 属性 nwkSecurityLevel 中决定安全等级。重写 3 位的 AuxillaryHeader 的安全控制域的安全等级子域字符串为这个值。从辅助头 AuxiliaryHeader 中决定序列号 SequenceNumber，发送地址 SenderAddress 和接收帧计数器 ReceivedFrameCount。如果 ReceivedFrameCount 值为 4 个八位位组表示的整数 $2^{32}-1$，安全处理会给上层一个状态为“错误帧计数器”的失败，本帧不能进一步进行安全处理。

b) 通过匹配 nwkSecurityMaterialSet 和 NIB 属性 nwkSecurityMaterialSet 中的任何密钥的序列号，获取合适的安全材料（包括密钥和其他属性）。如果对应于 SenderAddress 的邻居表节的关系域值为 0x01（子节点），该域必须设置为 0x05（未经验证的子节点）。如果不能获得安全材料，安全处理会指示上层一个状态为“帧安全失败”的失败，本帧不能进一步进行安全处理。

c) 如果有一个输入帧计数 FrameCount 对应于步骤 b）获得的安全材料的 SenderAddress，且如果 ReceivedFrameCount 小于 FrameCount，安全处理会指示上层一个状态为“错误帧计数器”的失败，本帧不能进一步进行安全处理。

d) 执行 CCM * 模式加密和认证检查操作，有如下实例：

1) 参数 M 应从表 2 对应于第 a）步的安全级别获得。

2) 位字符串 Key 应从第 b）步获得。

3) 临时 N 应是 13 个八位位组构成的字符串，使用从 AuxiliaryHeader 中获得的安全控制域、帧计数器域和源地址域构成的。注意安全控制域的安全等级子域已经在步骤 a）中被重写，现在包含了从 NIBnwkSecurityLevel 属性中确定的值。

4) 解析八位位组字符串 SecuredPayload 为 $Payload_1||Payload_2$，最右边的字符串 $Payload_2$ 是一个由 M 个八位位组构成的字符串。如果此操作失败，安全处理会指示上层一个状态为“帧安全失败”的失败，本帧不能进一步进行安全处理。

5) 如果安全级别需要加密，由 8 个八位位组构成的字符串 a 应是字符串 NwkHeader||AuxiliaryHeader，由 8 个八位位组构成的字符串 c 应是字符串 SecuredPayload。否则，a 是字符串 NwkHeader||AuxiliaryHeader||$Payload_1$，c 是 $Payload_2$。

e) 返回 CCM * 操作的结果：

1) 如果步骤 d）调用的 CCM * 模式输出“无效”，安全处理会指示上层一个状态为“帧安全失败”的失败，本帧不能进一步进行安全处理。

2) 设步骤 d）的输出结果为 m。如果安全等级需要加密，设置八位位组字符串 UnsecuredNwkFrame 为字符串 NwkHeader||m，否则，设置八位位组字符串 UnsecuredNwkFrame 为 NwkHeader||$Payload_1$。

f) 设置 FrameCount 为（ReceivedFrameCount+1），在 NIB 中保存 FrameCount 和 Sender Address。UnsecuredNwkFrame 现在表示未加密的收到的网络帧，安全处理成功。为了不会导致存储帧计数和地址信息超过可用内存，为 NWK 层安全所需的输入帧计数器分配的内存用 M * N 来限制，其中 M 和 N 分别表示 nwkSecurityMaterialSet 和 nwkNeighborTable。

g) 如果收到的帧的序列号属于 nwkSecurityMaterialSet 中的一个新条目，那么 nwkActiveKeySeqNumber 应该设置为收到的序列号。

h) 如果在 NIB 中 nwkNeighborTable 有一个条目，它的扩展地址和 SenderAddress 匹配，关系域值为 0x05(未经认证的子节点)，那么设置该条目的关系域值为 0x01(子节点)。

## A.3 路由安全

### A.3.1 发起者操作

当发起设备接收到路由响应命令后，可以对响应的路由器进行鉴别，以提高路由安全性。当 NIB 属性中 nwkRouterAuthen 属性如果为 TRUE，发起设备的邻近高层会发出一个 NLME-SEC-ROUTE.request 原语来启动路由安全的过程。Request Device Address 应设置为发起设备的 16 位网络地址，Request Identifier Type 应根据路由响应命令设置为 0x00 或 0x01。当 Request Identifier Type 设置为 0x00 时指示请求标识符类型是‘请求设备地址’对应的路由器标识符。当 Request Identifier Type 设置为 0x01 时指示请求标识符类型是‘待鉴别设备地址’对应的路由器标识符。

发起设备的 APSME 接收到 NLME-SEC-ROUTE.request 原语之后，发起设备应使用 MCPS-DATA.request 原语发送标识符请求命令，请求对路由响应设备进行鉴别。

如果 MAC 子层因为任何原因发送命令帧失败，APSME 将给邻近高层发出 NLME-SEC-ROUTE.confirm 原语，状态参数值等于 MCPS-DATA.confirm 返回的值。

如果设备成功鉴别，APSME 应通知发起者邻居高层鉴别的结果以及过程的成功。这通过发出 NLME-SEC-ROUTE.confirm 原语完成，并带有鉴别结果，且 Status 参数设置为 SUCCESS。

### A.3.2 响应者操作

在接收到 NLME-SEC-ROUTE.indication 原语之后，响应者应该根据标识符请求命令对标识符请求命令的发起设备进行鉴别。Request Device Address 应设置为发起设备的 16 位网络地址，Request Identifier Type 根据请求标识符类型是‘请求设备地址’还是‘待鉴别设备地址’设置为 0x00 或 0x01。

当响应设备的邻近高层接收到一个 NLME-SEC-ROUTE.indication 原语之后，邻近高层被告知接收到一个标识符请求命令。响应设备的邻近高层应发起一个 NLME-SEC-ROUTE.request 原语到 NLME，并发送一个标识符响应命令到发起设备。

如果 MAC 子层因为任何原因发送命令帧失败，NLME 将给邻近高层发出 NLME-SEC-ROUTE.confirm 原语，状态参数值等于 MCPS-DATA.confirm 返回的值。

如果设备成功鉴别，NLME 应通知响应者邻居高层鉴别的结果以及过程的成功。这通过发出 NLME-SEC-ROUTE.confirm 原语完成，并带有鉴别结果，且 Status 参数设置为 SUCCESS。安全路由过程的消息序列图如图 A.1 所示。

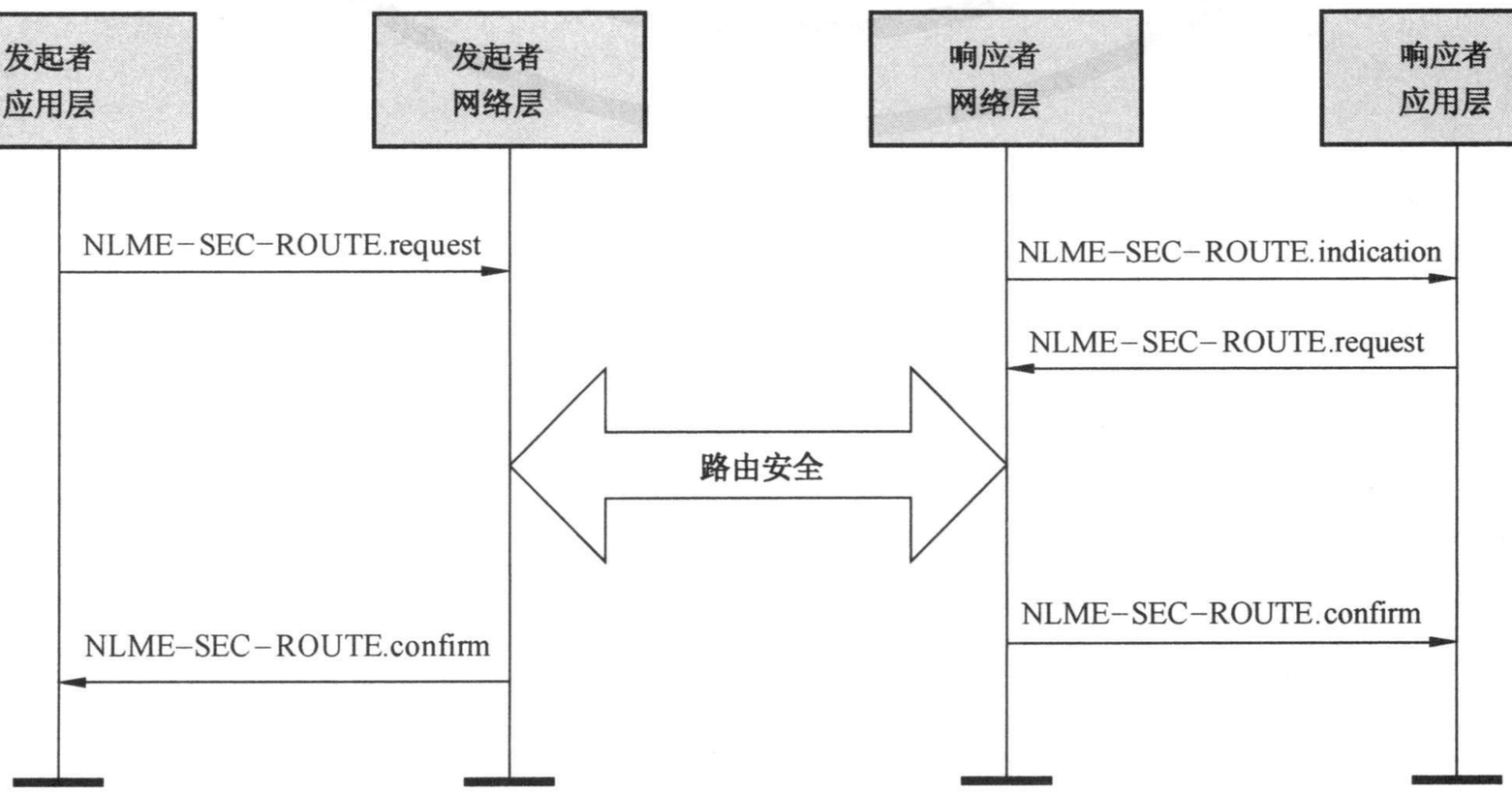


图 A.1 路由安全信息序列图

# 附 录 B
（规范性附录）
应用支持子层安全交互过程

## B.1 概述

应用支持子层的安全交互过程包括应用支持子层的帧安全、密钥管理、访问控制、鉴别以及数据融合，应用支持子层的帧安全、密钥管理、访问控制、鉴别以及数据融合应遵循本附录。

## B.2 帧安全

### B.2.1 概述

本条规定了输出帧和输入帧的安全处理步骤。

### B.2.2 输出帧的安全处理

如果应用支持子层帧需要安全保护，且 NIB 属性中 nwkSecurityLevel＞0，应该使用如下安全步骤：

a) 使用如下步骤获得安全材料和密钥标识符 KeyIdentifier。如果安全材料和密钥标识符不能决定，那么安全处理失败，该帧不能进一步进行安全处理。

1) 如果该帧是一个需要加密的应用支持子层数据帧：
如果 GB/T 30269.301—2014 的应用支持子层服务中 APSDE-DATA.request 原语的 TxOptions 参数规定网络密钥需要用于加密数据帧，那么应该从 NIB 用 nwkActiveKeySeqNumber 获得安全材料，以从 NIB 属性 nwkSecurityMaterialSet 中检索网络密钥、输出帧计数器和序列号。KeyIdentifier 应该设置为‘01’(即网络密钥)。否则，和输出帧目标地址相关的安全材料应该从 AIBapsDeviceKeyPairSet 属性中获得。KeyIdentifier 应该设置为‘00’(即一个共享密钥)。

2) 如果该帧是一个需要加密的应用支持子层命令帧：
尝试从 AIBapsDeviceKeyPairSet 属性中检索和输出帧目标地址相关的安全材料，KeyIdentifier 应该设置为‘00’(即一个共享密钥)。如果尝试失败，那么安全材料应该使用 nwkActiveKeySeqNumber 属性从 NIB 中获得，来检索网络密钥、输出帧计数器，且序列号从 nwkSecurityMaterialSet 属性中获得。KeyIdentifier 应该设置为‘01’(即网络密钥)。

b) 如果 KeyIdentifier 等于 01(即网络密钥)，应用支持子层应该先检查 NWK 层是否也需进行安全操作。应用支持子层可以通过检查 NIBnwkSecureAllFrames 属性值是否为 TRUE，以及 nwkSecurityLevel 属性值是否为非零，来确定 NWK 层是否也需进行安全操作。如果 NWK 层也需进行安全操作，那么应用支持子层就不能进行安全操作。

c) 提取从步骤 a)获得的安全材料的输出的帧计数器(且如果 KeyIdentifier 等于 01，密钥序列号)。如果输出帧计数器值为 4 个八位位组代表的整数 $2^{32}-1$，或者如果密钥无法获得，安全处理失败，本帧不能进一步进行安全处理。

d) 从 NIB 属性的 nwkSecurityLevel 中获得安全等级。

e) 建立辅助头 AuxiliaryHeader。安全控制域应该设置如下：
   1) 安全等级子域应该设置为步骤 d)中获得的安全等级。密钥标识符子域应该设置为 KeyIdentifier。扩展的临时子域应该设置为 0。
   2) 帧计数器域应该设置为步骤 3 中的帧计数器。
   3) 如果 KeyIdentifier 是 01，密钥序列号域应该存在，且设置为步骤 c)获得的密钥序列号。否则密钥序列号域不存在。
f) 执行 CCM * 模式(见 GB/T 15629.15—2010)加密和认证检查操作：
   1) 参数 M 应从对应于第 c)步的安全级别的表 36 获得。
   2) 位字符串 Key 从第 a)步获得。
   3) 该临时 N 应是 13 个八位位组构成的字符串，使用第 e)步获得的安全控制域和帧计数器域，以及本地设备中获得的 64 位 IEEE 地址构成。
   4) 如果安全等级需要加密，八位位组构成的字符串 a 应该是字符串 ApsHeader||AuxiliaryHeader，八位位组构成的字符串 M 应该是 Payload。否则，八位位组构成的字符串 a 是字符串 ApsHeader||AuxiliaryHeader||Payload，M 则是长度为零的字符串。
g) 如果步骤 f)引用的 CCM * 模式输出“无效”，安全处理失败，本帧不能进一步进行安全处理。
h) 设步骤 f)的输出结果为 F(out)。如果安全等级需要加密，则加密的输出帧为 ApsHeader||AuxiliaryHeader||F(out)，否则，加密的输出帧为 ApsHeader||AuxiliaryHeader||Payload||F(out)。
i) 如果 MSDU 中加密的输出帧比 aMaxMACFrameSize 大，安全处理会失败，本帧不能进一步进行安全处理。
j) 第 c)步的输出帧计数器应该增加，并且储存在 NIB、AIB 和 MACPIB 的合适位置，和用来保护输出帧的密钥对应。
k) 重置安全控制域的安全等级子域为 3 位的全零字符串‘000’。

### B.2.3 输入帧的安全处理

如果应用支持子层收到一个加密的帧，如应用支持子层帧头控制域的安全子域所指示，它应按照如下步骤执行安全处理：

a) 设置序列号 SequenceNumber，密钥标识符 KeyIdentifier 和来自辅助头 AuxiliaryHeader 的帧计数器值 ReceivedFrameCounter。如果 ReceivedFrameCounter 是 4 个八位位组表示的整数 $2^{32}-1$，安全处理会失败，本帧不能进一步进行安全处理。
b) 从 NIB 地址映射表中确定源地址 SourceAddress，使用应用支持子层帧的源地址作为索引。如果源地址是不完整的或者不可用的，安全处理会失败，本帧不能进一步进行安全处理。
c) 通过以下方式获得合适的安全材料。如果安全材料不能获得，安全处理会失败，本帧不能进一步进行安全处理。
   1) 如果 KeyIdentifier 是‘00’(即一个共享密钥)，和输入帧 SourceAddress 相关的安全材料应该从 AIBapsDeviceKeyPairSet 属性中获得。
   2) 如果 KeyIdentifier 是‘01’(即一个网络密钥)，安全材料应该通过匹配 SequenceNumber 和 NIB 中 nwkSecurityMaterialSet 属性的任一密钥的序列号获得。
d) 如果有一个输入的帧计数 FrameCount 和从第 c)步中获得的安全材料 SourceAddress 相关，并且如果 ReceivedFrameCount 小于 FrameCount，安全处理会失败，本帧不能进一步进行安全处理。
e) 设置安全等级 SecLevel 如下。如果 ApsHeader 帧控制域的帧类型子域指示了一个应用支持子层数据帧，那么 SecLevel 应该设置为 NIB 的 nwkSecurityLevel 属性。否则，SecLevel 应该

设置为 7。AuxillaryHeader 中的安全控制域的安全等级子域值为 SecLevel。

f) 执行 CCM * 模式加密和认证检查操作，有如下实例：

1) 参数 M 应从对应于第 5)步的安全级别的表 36 中获得。

2) 位字符串 Key 从第 c)步获得。

3) 该临时 N 应是 13 个八位位组构成的字符串，使用 AuxiliaryHeader 获得的安全控制域，帧计数器域，和第 2)步获得的 SourceAddress 构成。

4) 解析字符串 SecuredPayload 为 $Payload_1||Payload_2$，最右边字符串 $Payload_2$ 是一个 M 个八位位组构成的字符串。如果此操作失败，安全处理会失败，本帧不能进一步进行安全处理。

5) 如果安全等级需要加密，八位位组构成的字符串 a 应该是 ApsHeader||AuxiliaryHeader，八位位组构成的字符串 c 应该是 SecuredPayload。否则，a 是字符串 ApsHeader||AuxiliaryHeader||$Payload_1$，c 是字符串 $Payload_2$。

g) 返回 CCM * 操作的结果：

1) 如果步骤 f)调用的 CCM * 模式输出“无效”，安全处理会失败，本帧不能进一步进行安全处理。

2) 设步骤 f)的输出结果为 m。如果安全等级需要加密，则设置八位位组构成的字符串 UnsecuredApsFrame 为 ApsHeader||m。否则，设置 UnsecuredApsFrame 为 ApsHeader||Payload。

h) 设置 FrameCount 为(ReceivedFrameCount+1)，在第 c)步获得的合适安全材料中保存 FrameCount 和 SourceAddress。如果存储帧计数和地址信息超过这类信息分配的内存存储量，且 NIB 中 nwkAllFresh 属性设置为 TRUE，那么安全处理会失败，本帧不能进一步进行安全处理。否则安全处理成功。

i) 如果收到的帧的序列号是 nwkSecurityMaterialSet 中的一个新条目，那么 nwkActiveKeySeqNumber 应该设置为收到的序列号。

## B.3 密钥分发

### B.3.1 信任中心操作

信任中心在为目标设备生成密钥参数后，会发出一个 APSME-DISTRIBUTE-KEY.request 原语，开始执行分发密钥的操作，如果采用手持设备进行密钥的分发，则 DistributeType 参数应该设置为 0x01，NodeID 参数设置为被分发密钥的设备的 64 位 IEEE 地址，KeyType 参数设置为分发的密钥类型：

如果 KeyType 为 0x00，表示信任中心分发的密钥为初始化密钥材料：

a) KeyID 应该设置为密钥材料的标识符。

b) Key 应该设置为密钥消息，支持一次分发多一个密钥材料及其相应的密钥标识符。

如果 KeyType 为 0x01，表示信任中心分发的密钥为初始化值：

初始化值并不具备 KeyID 参数，仅应将 value 设置为初始化的值。

信任中心执行进行密钥的分发时，还需将 NodeID，KeyType，KeyID 与 Key 进行绑定。

### B.3.2 设备的操作

如果 APS 层收到信任中心分发的密钥消息，会发出一个 APSME-DISTRIBUTE-KEY.indication 原语，并执行如下操作：

a) 如果设备接收到信任中心分发的 KeyType 为 0x00，且选择接受，则 AIB 表中 Initialization

key,KeyID,Key 等属性应该被更新。

b) 如果设备接收到信任中心分发的 KeyType 为 0x01,且选择接受,则 AIB 表中 Initialization value,Key 等属性应该被更新。

c) 如果设备接收到信任中心分发的 KeyType 为 0x02,且选择接受,则 AIB 表中 KeyType,KeyID,Key 等属性应该被更新。

## B.4 密钥建立

### B.4.1 设备操作

若节点设备中写入的初始密钥材料即为其与邻居节点间共享的直接密钥,且节点知道密钥是与哪个节点所共享,则无需执行此过程,用于建立共享密钥,而是直接建立会话密钥。否则,发起设备应发出 APSME-ESTABLISH-KEY.request 原语开始这个程序。具体的密钥建立方法根据 KeyEstablishmentMethod 参数来确定,执行步骤如下:

a) 如果 KeyEstablishmentMethod 为 0x00,则采用基于密钥池的分发方式,Responder-Address 应该设置为广播地址,KeyType 应该设置为需要建立的密钥类型,KeyID 应该设备为需要广播的密钥材料的标识符,Key 设置为对应的密钥材料,在该密钥建立方法下发送密钥建立命令帧时,命令帧不包含 Key 的信息。如果 KeyEstablishmentMethod 为 0x01,则采用基于多项式的分发方式,Responder-Address 应该设置为广播地址,KeyType 应该设置为需要建立的密钥类型,Polynomial ID 应该设置为信任中心分发的多项式标识符,Nonce 应该设置为设备生成的随机数,NodeID 应该设置被设备的标识符,在该密钥建立方法下发送密钥建立命令帧时,命令帧包含 NodeID 和 Nonce。如果 KeyEstablishmentMethod 为 0x02,则采用路径密钥建立 1 的方法,Responder-Address 应该设置为响应设备的地址,KeyType 应该设置为需要建立的密钥类型,并启动路径密钥建立 1 的协议。如果 KeyEstablishmentMethod 为 0x03,则采用路径密钥建立 2 的方法,Responder-Address 应该设置为响应设备的地址,KeyType 应该设置为需要建立的密钥类型,并启动路径密钥建立 2 的协议。

b) 当设备接收到一个共享密钥建立消息后,向它的邻近高层发送一个 APSME-ESTABLISH-KEY.indication 原语,DeviceAddress 元素应该设置为 InitiatorAddress 密钥建立信息发起设备的地址,如果 KeyEstablishmentMethod 为 0x00,则设备将只选取与自身 Initialization key ID 元素相等的密钥材料建立所需密钥,并将 Shared key 元素设置为建立的密钥,Shared key ID 元素设置为协商的密钥标识符。如果 KeyEstablishmentMethod 为 0x01,则设备将只选取与自身 Polynomial ID 元素相等的密钥材料建立所需密钥,并将 Shared key 元素设置为建立的密钥,Shared key ID 元素设置为协商的密钥标识符。如果 KeyEstablishmentMethod 为 0x02,如果接收到的密钥建立请求来自非信任中心,则 DeviceAddress 元素应该设置为 InitiatorAddress 密钥建立信息发起设备的地址,在接收到来自信任中心的密钥建立消息后,并将 Shared key 元素设置为 Key,Shared key ID 元素设置为 Key ID。如果 KeyEstablishment-Method 为 0x03,如果接收到的密钥建立请求来自非可信的设备,则 DeviceAddress 元素应该设置为 InitiatorAddress 密钥建立信息发起设备的地址;如果接收到的密钥建立消息来自可信的设备,并将 Shared key 元素设置为 Key,Shared key ID 元素设置为 Key ID。

在共享密钥建立完成后,设备可在共享密钥的基础上,采用上述密钥建立协商,建立会话密钥,过程与共享密钥建立过程基本相同。

### B.4.2 信任中心操作

当采用路径密钥建立 1 的方式,即当 KeyEstablishmentMethod 为 0x02 时,在接收到一个来自设

备的密钥建立请求，信任中心会向它的邻近高层发送一个 APSME-ESTABLISH-KEY.indication 原因，信任中心为请求设备生成密钥消息及其相应的标识符，并执行发送密钥操作。

当信任中心需要发送密钥消息到请求节点，它的邻近高层会发送一个 APSME-ESTABLISH-KEY.request 原语，开始为密钥建立的发起者和响应者发送密钥，KeyType 应该设置为发送的密钥类型，KeyID 应该设置为发送到设备的密钥标识符，Key 应该设置为发送的密钥消息。

### B.4.3 可信节点操作

当采用路径密钥建立 2 的方式，即当 KeyEstablishmentMethod 为 0x03 时，在接收到一个来自设备的密钥建立请求，可信节点会向它的邻近高层发送一个 APSME-ESTABLISH-KEY.indication 原语，可信节点开始进行密钥的协商，它的邻近高层将发起一个 APSME-ESTABLISH-KEY.request 原语，Responder-Address 应该设置为响应设备的可信节点的地址。完成密钥的协商后，可信节点的开始协商密钥的发送，它的邻近高层将发起一个 APSME-ESTABLISH-KEY.request 原语，Responder-Address 应该设置为建立密钥请求节点的地址，KeyID 应该设置为协商密钥的标识符，Key 应该设置为协商的密钥消息。密钥建立的消息序列图如图 B.1 所示。

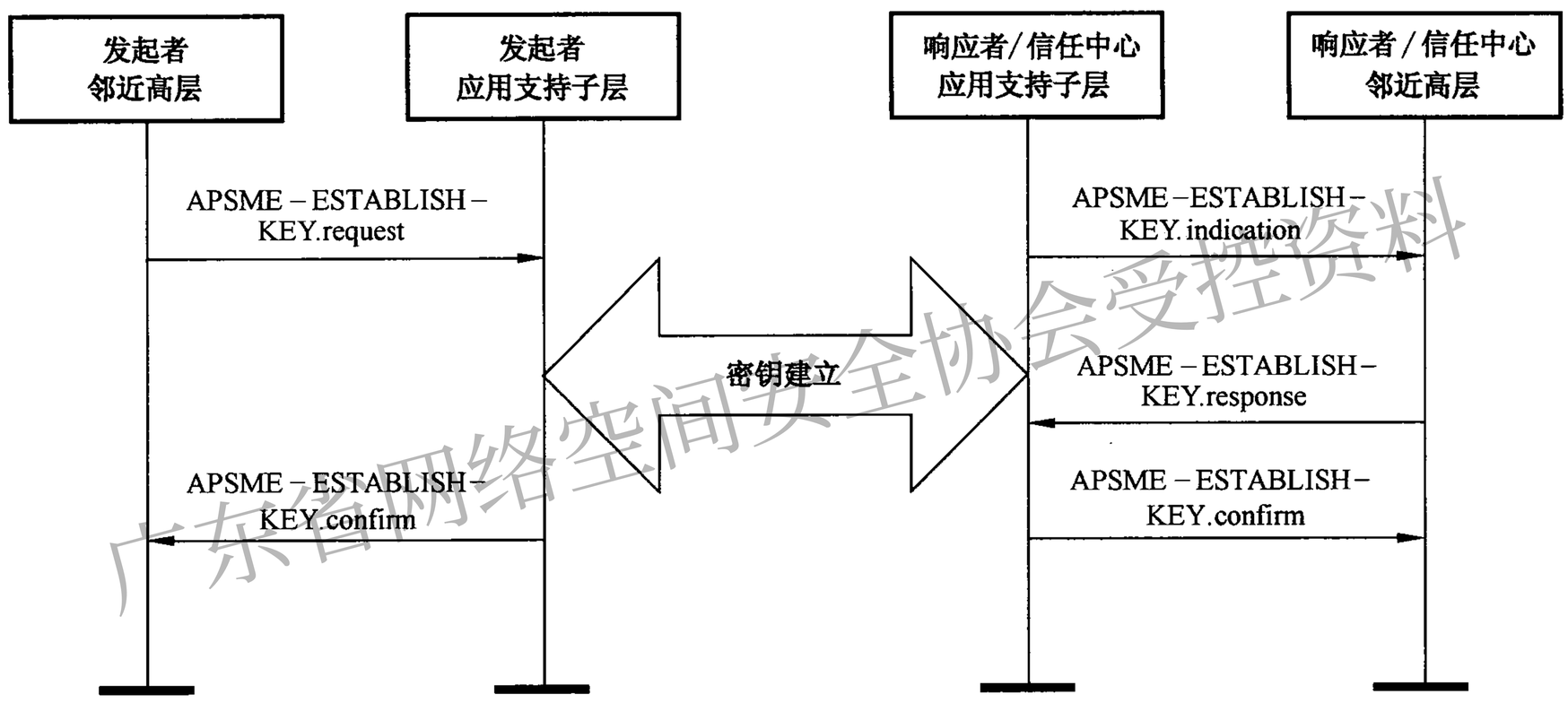


**图 B.1 密钥建立消息序列图**

## B.5 密钥更新

### B.5.1 信任中心操作

当信任中心需要向特定设备进行密钥更新时，它的邻近高层将发送一个 APSME-UPDATE-KEY.request 原语，DestAddress 应该设置为被更新设备的 64 位 IEEE 地址，KeyType 应该设置为更新的密钥类型，当 KeyType 为 0x00，0x03 时，KeyID 应该设置为密钥的标识符，Key 应该设置为更新的密钥消息；当 KeyType 为 0x01 时，value 应该设置为更新的初始化值。

### B.5.2 设备操作

在收到 APSME-UPDATE-KEY.indication 原语之后，查找 DeviceAddress 元素为对应参数 DestAddress 的设备地址，如果 KeyType 为 0x00，DestAddress 为设备自身的地址，Initialization key 和 Initialization key ID 元素应该设置为 KeyData 参数中的初始化密钥材料及其相应的标识符。

如果 KeyType 为 0x01，Initialization value 元素应该设置为 KeyData 参数中的初始化值。如果 KeyType 为 0x02，Shared key 和 Shared key ID 元素应该设置为 KeyData 参数中的初始化密钥材料及其相应的标识符。如果 KeyType 为 0x03，Session key ID 和 Session key 元素应该设置为 KeyData 参

数中的初始化值。密钥更新的信息序列图如图 B.2 所示。

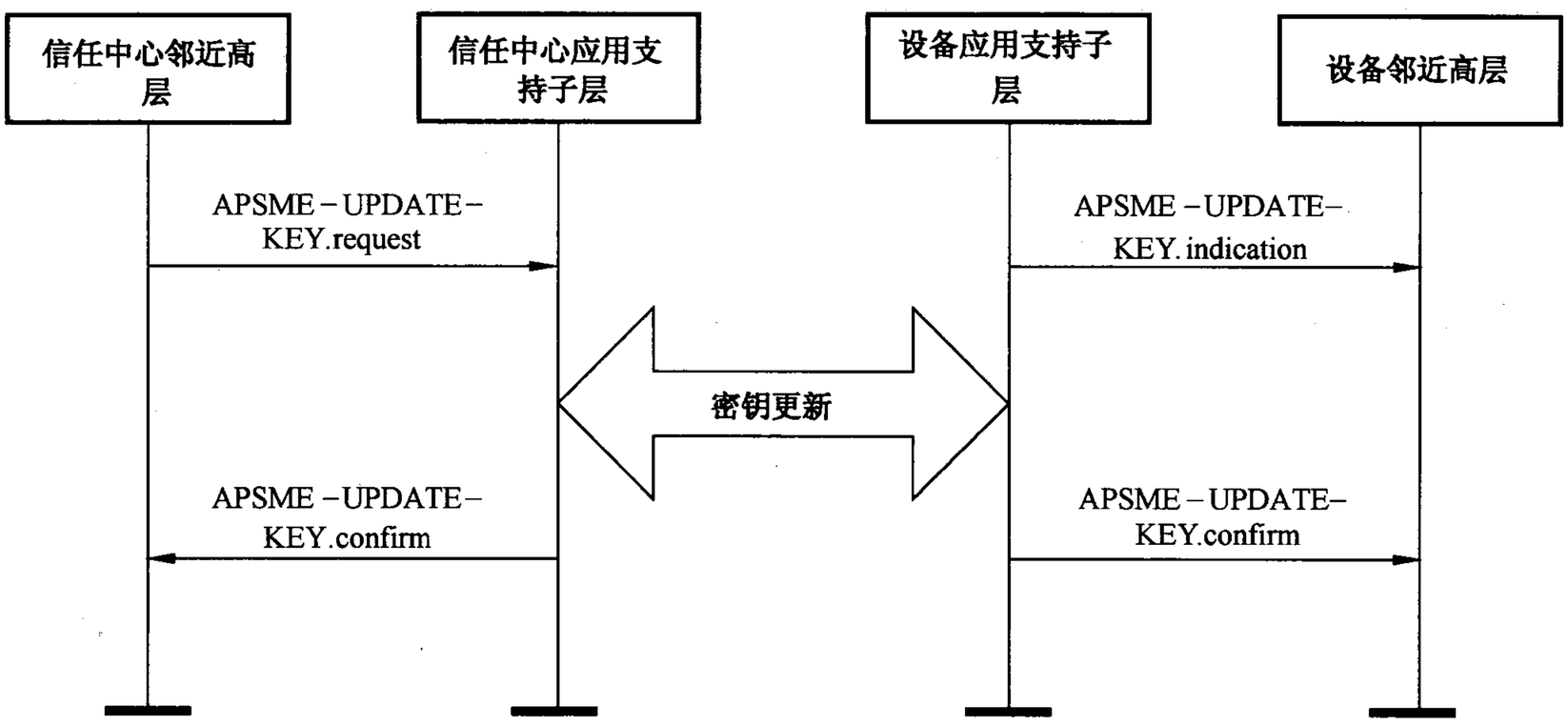


**图 B.2　密钥更新消息序列图**

## B.6　密钥撤销

### B.6.1　信任中心操作

当信任中心需要执行特定设备密钥撤销操作时，它的邻近高层将发出一个 APSME-REVOCATION-KEY.request 原语，DestAddress 应该设置为被撤销密钥设备的地址，KeyID 应该设置为被撤销密钥的标识符，Revocation Time 应该设置为密钥撤销的时间，Revocation Reason 应该设置为密钥撤销的原因。

### B.6.2　设备操作

设备在收到信任中心发送的密钥撤销命令后，向邻居高层发送一个 APSME-REVOCATION-KEY.indication 原语之后，设备查找 DeviceAddress 中对应参数 DeviceAddress 的选项，将设备计时器值设置为 Revocation Time 参数的值，在设备计时器到达时，根据原语中的密钥标识符参数删除 AIB 中相应的密钥元素。密钥撤销的消息序列图如图 B.3 所示。

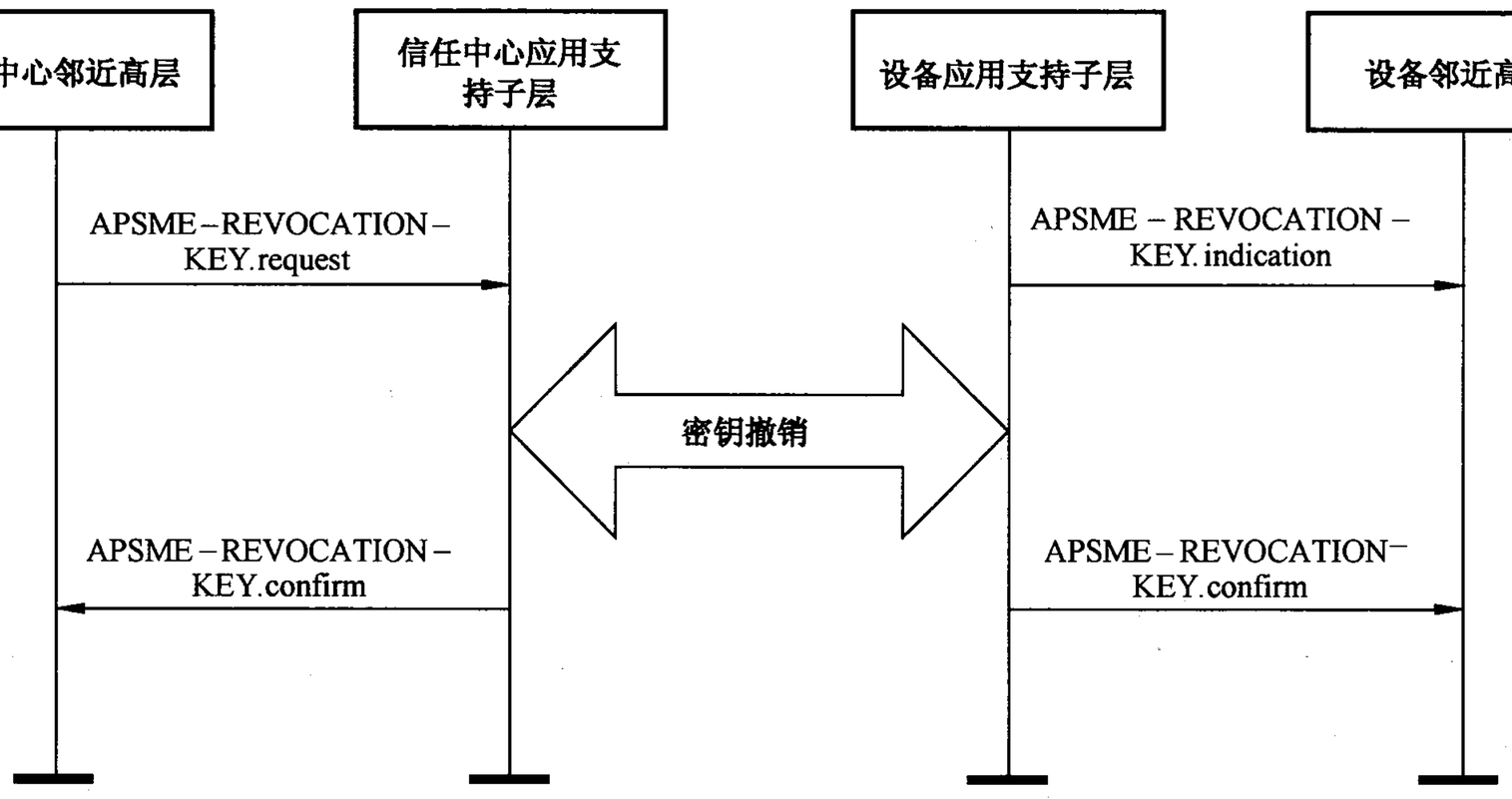


**图 B.3　密钥撤销消息序列图**

## B.7 访问控制

### B.7.1 自主访问控制

访问者启动访问过程，网关收到访问请求。网关使用 APSME-ACCESS-CONTROL.request 原语发起访问控制的过程，ResponderAddress 参数设置为网关地址，Action 参数设置为发起，AccessObject 参数设置为对应的访问对象，AccessControl-Method 参数设置为 0x00 自主访问控制。

在收到访问控制请求命令后，应用支持子层应发出一个 APSME-ACCESS-CONTROL.indication 原语给邻近高层，InitiatorAddress 参数设置为发起设备的地址，AccessControl-Method 参数设置为 0x00 自主访问控制。应用支持子层应发出一个 APSME-ACCESS-CONTROL.confirm 原语来通知邻近高层访问者的访问结果，Address 参数设置为访问控制发生的地址，如果传输成功，Status 参数会设置为 SUCCESS，否则，Status 参数会显示错误。网关利用访问控制表与访问能力表，决定用户能访问网内的哪些设备以及进行何种类型的操作，通过访问控制响应命令发送给访问发起者。

### B.7.2 强制访问控制

访问者启动访问过程，响应者收到访问请求后，使用 APSME-ACCESS-CONTROL.request 原语发起访问控制的过程，ResponderAddress 参数设置为网关地址，Action 参数设置为发起，AccessObject 参数设置为对应的访问资源，AccessControl-Method 参数设置为 0x01 强制访问控制。

在收到访问控制请求命令后，应用支持子层应发出一个 APSME-ACCESS-CONTROL.indication 原语，InitiatorAddress 参数设置为发起设备的地址，AccessControl-Method 参数设置为 0x01 强制访问控制。应用支持子层应发出一个 APSME-ACCESS-CONTROL.confirm 原语来通知邻近高层访问者的访问结果，Address 参数设置为访问控制发生的地址，如果在网关处实施访问控制，Address 参数设置为网关地址，如果在节点处实施访问控制，Address 参数设置为节点地址。如果传输成功，Status 参数会设置为 SUCCESS，否则，Status 参数会显示错误。网关对比用户与访问对象的安全级别，决定是否接受用户的访问，通过访问控制响应命令发送给访问发起者。

访问控制的消息序列图如图 B.4 所示。

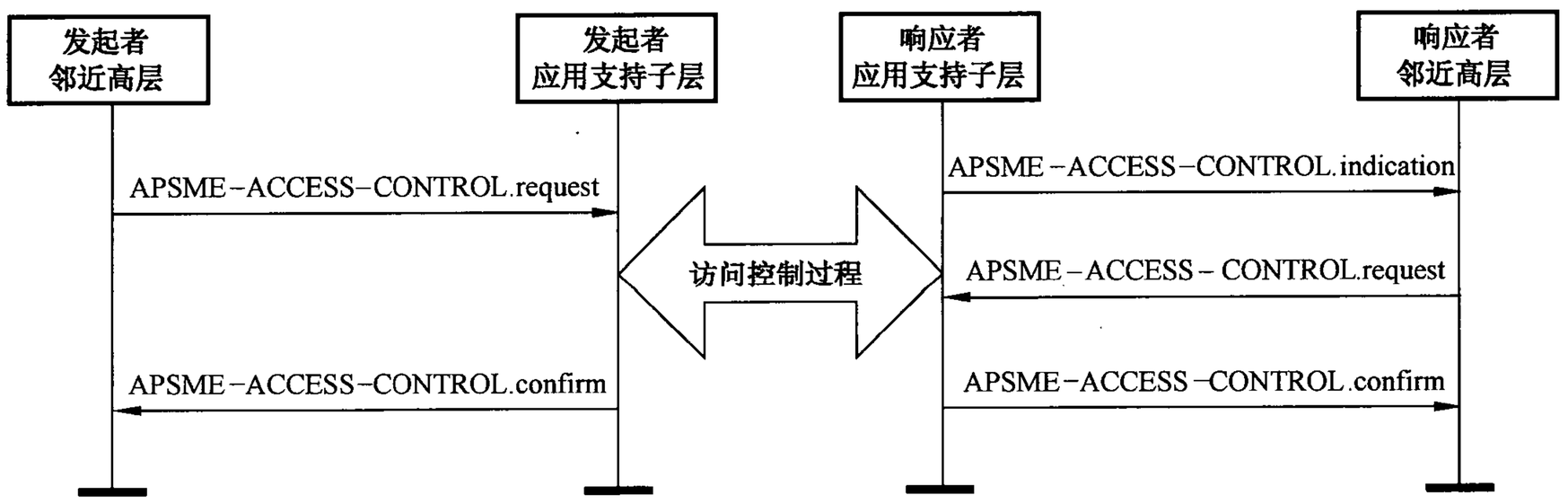


**图 B.4 访问控制消息序列图**

## B.8 身份鉴别

### B.8.1 概述

发起设备应发出 APSME-IDENTITY-AUTHENTICATE.request 原语和响应设备开始身份鉴别的程序。包括基于异或运算的鉴别、基于杂凑运算的鉴别、基于分组密码算法的鉴别、基于非对称密码算法的鉴别应该遵守本小节的操作。

### B.8.2 发起者操作

当发起者需要使用身份鉴别机制和响应者之间进行身份鉴别时，发起者的邻近高层会发出APSME-IDENTITY-AUTHENTICATE.request 原语，PartnerAddress 应设置为发起设备的 64 位IEEE 地址，RandomChallenge 参数设置为一个新的随机值，AuthenticateMethod 应设置为相应的身份鉴别方法。

其中 AuthenticateMethod 设置为 0x00 表示发起者和响应者之间是采用基于异或运算的鉴别方法，AuthenticateMethod 设置为 0x01 表示发起者和响应者之间是采用基于杂凑运算的鉴别方法，AuthenticateMethod 设置为 0x02 表示发起者和响应者之间是采用基于分组密码算法的鉴别方法，AuthenticateMethod 设置为 0x03 表示发起者和响应者之间是采用基于非对称密码算法的鉴别方法，PartnerAddress 设置为发起设备的 64 位 IEEE 地址。且 RandomChallenge 参数设置为一个新的随机值。

发起设备的 APSME 接收到 APSME-IDENTITY-AUTHENTICATE.request 原语之后，发起设备应使用 MCPS-DATA.request 原语发送身份鉴别请求命令，请求与响应设备进行身份鉴别的过程。

当发起设备的邻近高层接收到 APSME-IDENTITY-AUTHENTICATE.indication 原语之后，邻近高层应发送 APSME-IDENTITY-AUTHENTICATE.request 原语到 APSME，并使发起设备使用 MCPS-DATA.request 原语发送一个身份鉴别响应确认命令以响应响应设备的身份鉴别响应命令。

如果 MAC 子层因为任何原因发送命令帧失败，APSME 将给邻近高层发出 APSME-IDENTITY-AUTHENTICATE.confirm 原语，状态参数值等于 MCPS-DATA.confirm 返回的值。

如果设备成功鉴别，APSME 应通知发起者邻居高层鉴别的结果以及过程的成功。这通过发出 APSME-IDENTITY-AUTHENTICATE.confirm 原语完成，并带有鉴别结果，且 Status 参数设置为 SUCCESS。

### B.8.3 响应者操作

在接收到 APSME-IDENTITY-AUTHENTICATE.indication 原语之后，响应者应该根据身份鉴别请求命令决定使用哪种鉴别方法。InitiatorAddress 应设置为发起设备的 64 位 IEEE 地址，RandomChallenge 参数设置为一个新的随机值，AuthenticateMethod 应设置为相应的身份鉴别方法。

如果发起鉴别请求的设备请求使用的鉴别方法是基于杂凑运算的鉴别。APSME-IDENTITY-AUTHENTICATE.indication 中的 AuthenticateMethod 参数应设置为 0x01。此时设备应启动基于杂凑运算的鉴别机制。

如果发起鉴别请求的设备请求使用的鉴别方法是基于异或运算的鉴别。APSME-IDENTITY-AUTHENTICATE.indication 中的 AuthenticateMethod 参数应设置为 0x00，此时设备应启动基于异或运算的鉴别机制。

如果发起鉴别请求的设备请求使用的鉴别方法是基于分组密码算法的鉴别。PSME-ENTITY-AUTHENTICATE.indication 中的 AuthenticateMethod 参数应设置为 0x02。此时设备应启动基于分组密码算法的鉴别机制。

如果发起鉴别请求的设备请求使用的鉴别方法是基于非对称密码算法的鉴别。PSME-ENTITY-AUTHENTICATE.indication 中的 AuthenticateMethod 参数应设置为 0x03。此时设备应启动基于非对称密码算法的鉴别机制。

当响应设备的邻近高层接收到一个 APSME-IDENTITY-AUTHENTICATE.indication 原语之后，邻近高层被告知接收到一个身份鉴别请求命令。响应设备的邻近高层应发起一个 APSME-IDENTITY-AUTHENTICATE.request 原语到 APSME，并根据对发起设备的鉴别信息发送一个身份鉴别响应确认命令。

如果 MAC 子层因为任何原因发送命令帧失败，APSME 将给邻近高层发出 APSME-IDENTITY-

AUTHENTICATE.confirm 原语，状态参数值等于 MCPS-DATA.confirm 返回的值。

如果设备成功鉴别，APSME 应通知响应者邻居高层鉴别的结果以及过程的成功。这通过发出 APSME-IDENTITY-AUTHENTICATE.confirm 原语完成，并带有鉴别结果，且 Status 参数设置为 SUCCESS。身份鉴别的消息序列图如图 B.5 所示。

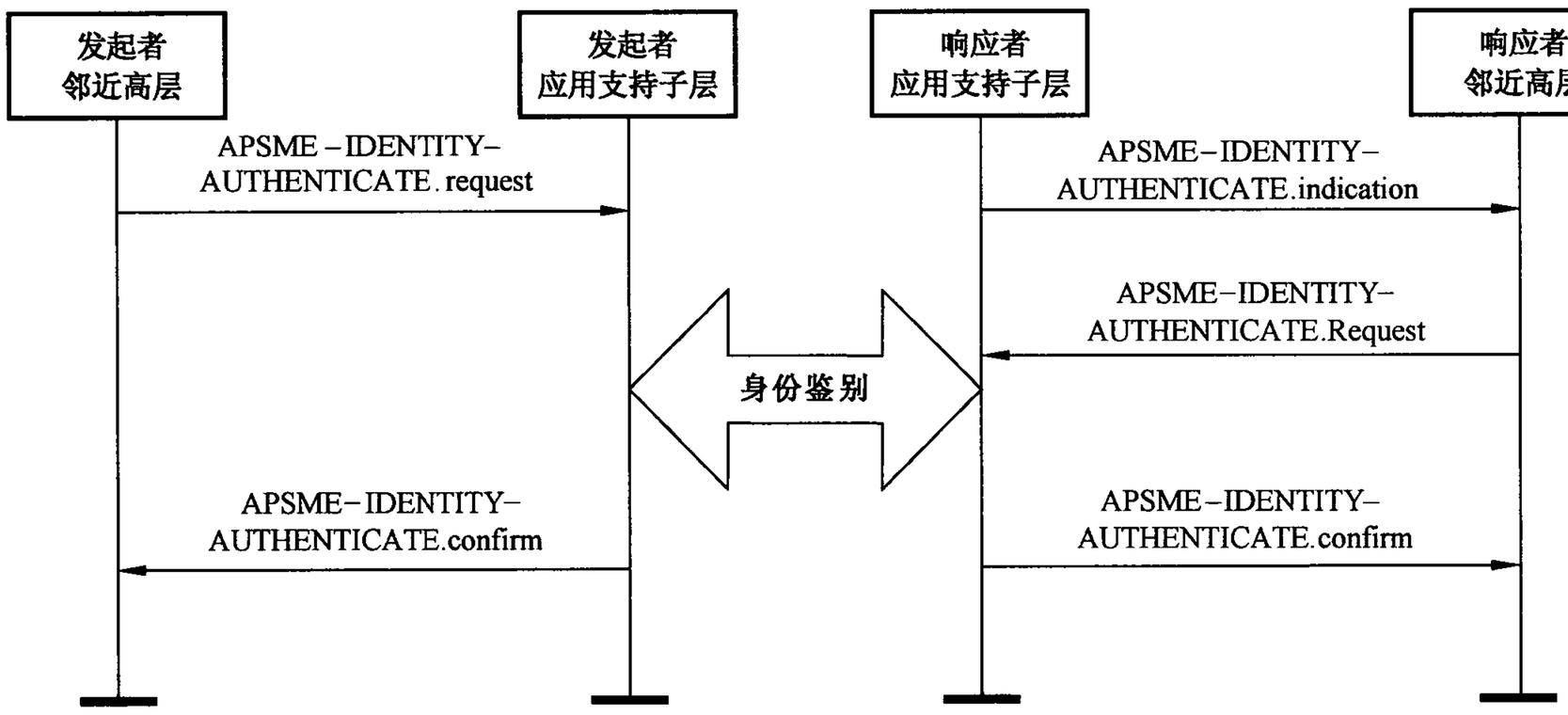


**图 B.5　身份鉴别消息序列图**

## B.9　广播消息鉴别

### B.9.1　概述

发起设备应发出 APSME-MESSAGE-AUTHENTICATE.request 原语和响应设备开始广播消息鉴别的程序。广播消息鉴别应该遵守本小节的操作。

### B.9.2　发起者操作

广播消息鉴别的过程通过使用 APSME-MESSAGE-AUTHENTICATE.request 原语发起。PartnerAddress 应设置为发起设备的 64 位 IEEE 地址，RandomChallenge 参数设置为一个新的随机值。发起设备通过 APSME-MESSAGE-AUTHENTICATE.request 原语生成广播消息鉴别命令帧，并以广播的方式发送。

如果 MAC 子层因为任何原因发送命令帧失败，APSME 将给邻近高层发出 APSME-MESSAGE-AUTHENTICATE.confirm 原语，状态参数值等于 MCPS-DATA.confirm 返回的值。

### B.9.3　响应者操作

响应设备在收到发起设备发送的广播消息鉴别命令帧后，APSME 向邻居高层发送一个 APSME-MESSAGE-AUTHENTICATE.indication 原语，InitiatorAddress 应设置为发起设备的 64 位 IEEE 地址，RandomChallenge 参数设置为一个新的随机值。在接收到广播消息命令帧后响应设备开始对广播消息的鉴别。

如果设备成功鉴别，APSME 应通知邻居高层鉴别的结果以及过程的成功。这通过发出 APSME-IDENTITY-AUTHENTICATE.confirm 原语完成，并带有鉴别结果，且 Status 参数设置为 SUCCESS。广播消息鉴别的消息序列图如图 B.6 所示。

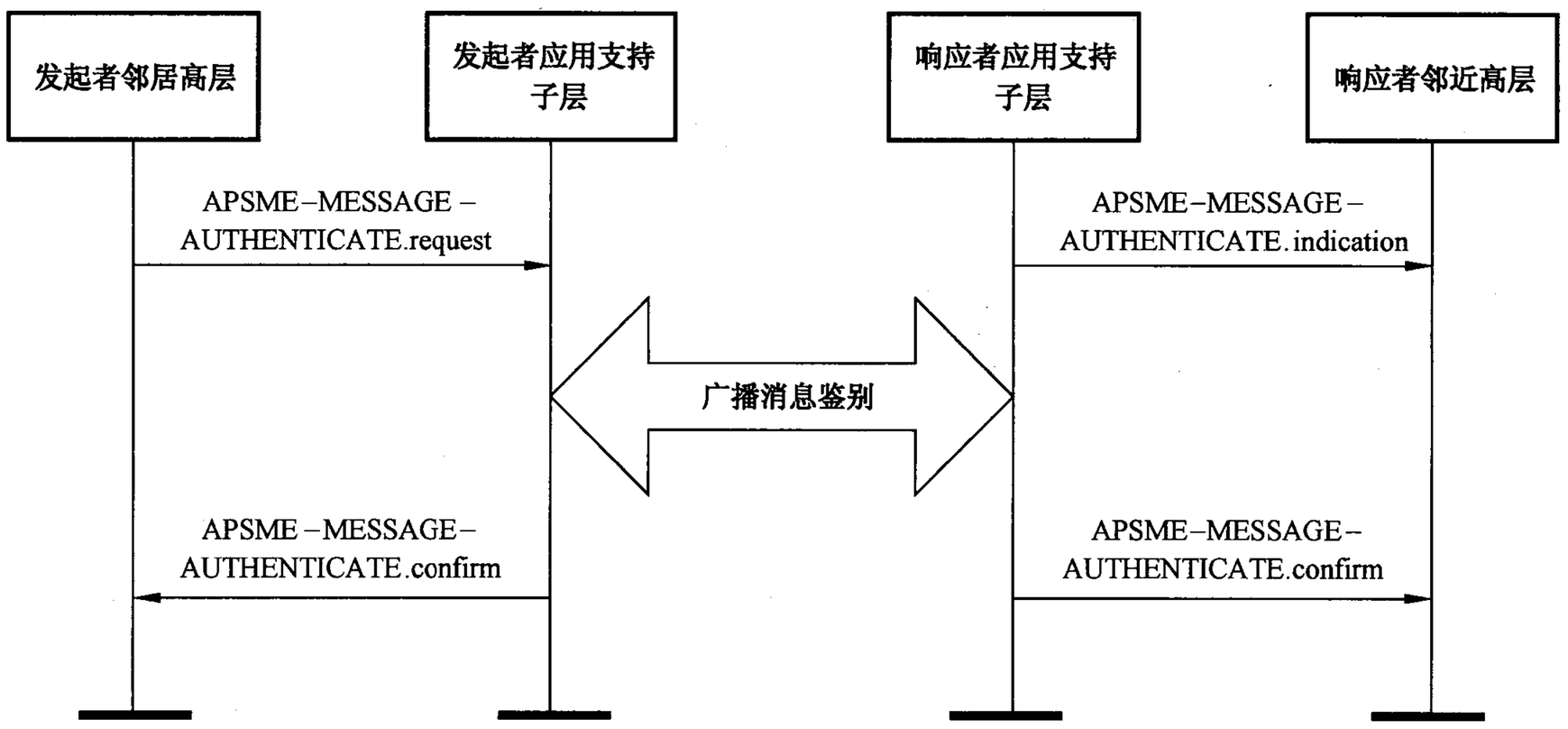


**图 B.6　广播消息鉴别序列图**

## B.10　安全数据融合

### B.10.1　信任中心操作

监督节点的选定可由信任中心指定。

### B.10.2　发起者操作

选定融合设备和监督设备之后，发起设备邻近高层发送 APSME-SECURE-DATA-AGGREGATION-START.request 原语给本地的 APSME，本地的 APSME 应该形成安全数据融合命令发送到指定响应设备。

发起设备 APSME 根据响应设备响应结果，确定安全数据融合服务启动结果（融合设备和监督设备响应结果同时为“接受”时，启动成功；否则，启动失败），通过 APSME-SECURE-DATA-AGGREGATION-START.confirm 原语响应给邻近高层，并形成启动确认命令发送给响应设备。

当发起设备判定融合设备融合信息不可信时，它的邻近高层发出 APSME-SECURE-DATA-AGGREGATION-REVOKE.request 这个原语给本地 APSME，并下发报文撤销融合节点以及对应的监督节点。

### B.10.3　响应者操作

#### B.10.3.1　概述

响应设备主要包括融合设备和监督设备两部分。

#### B.10.3.2　融合设备操作

融合设备 APSME 接收到安全数据融合命令后，给邻近高层发送 APSME-SECURE-DATA-AGGREGATION-START.indication 原语。

邻近高层会根据 InitiatorAddress、Function 和 Time 参数确定是否具备完成该功能的条件并决定是否启动发起者所要求的在安全数据融合服务过程中所承担的融合功能，然后使用 APSME-SECURE-DATA-AGGREGATION-START.response 原语去响应 APSME-SECURE-DATA-AGGREGATION-START.indication 原语。邻近高层的决定结果在 APSME-SECURE-DATA-AGGREGATION-START.response 原语 Accept 参数中指明。

融合设备 APSME 将 APSME-SECURE-DATA-AGGREGATION-START.response 原语响应结果形成响应命令，发送给发起设备 APSME，并等待启动结果。

融合设备 APSME 使用 APSME-SECURE-DATA-AGGREGATION-START.confirm 原语提示邻近高层安全数据融合服务启动结果。

邻近高层根据启动结果，决定是否启动融合过程。若启动结果为“成功”，则启动融合功能；否则，不启动数据融合功能。

融合设备邻近高层接收到 APSME-SECURE-DATA-AGGREGATION-REVOKE.indication 原语之后，终止融合功能。

### B.10.3.3 监督设备操作

监督设备 APSME 接收到安全数据融合命令后，给邻近高层发送 APSME-SECURE-DATA-AGGREGATION-START.indication 原语。

邻近高层会根据 InitiatorAddress、Function 和 Time 参数确定是否具备完成该功能的条件并决定是否启动发起者所要求的在安全数据融合服务过程中所承担的监督功能，然后使用 APSME-SECURE-DATA-AGGREGATION-START.response 原语去响应 APSME-SECURE-DATA-AGGREGATION-START.indication 原语。邻近高层的决定结果在 APSME-SECURE-DATA-AGGREGATION-START.response 原语 Accept 参数中指明。

监督设备 APSME 将 APSME-SECURE-DATA-AGGREGATION-START.response 原语响应结果形成响应命令，发送给发起设备 APSME，并等待启动结果。

监督设备 APSME 使用 APSME-SECURE-DATA-AGGREGATION-START.confirm 原语提示邻近高层安全数据融合服务启动结果。

邻近高层根据启动结果，决定是否启动监督过程。若启动结果为“成功”，则启动监督功能；否则，不启动监督功能。

监督设备邻近高层接收到 APSME-SECURE-DATA-AGGREGATION-REVOKE.indication 原语之后，终止监督功能。

安全数据融合的消息序列图如图 B.7 所示。

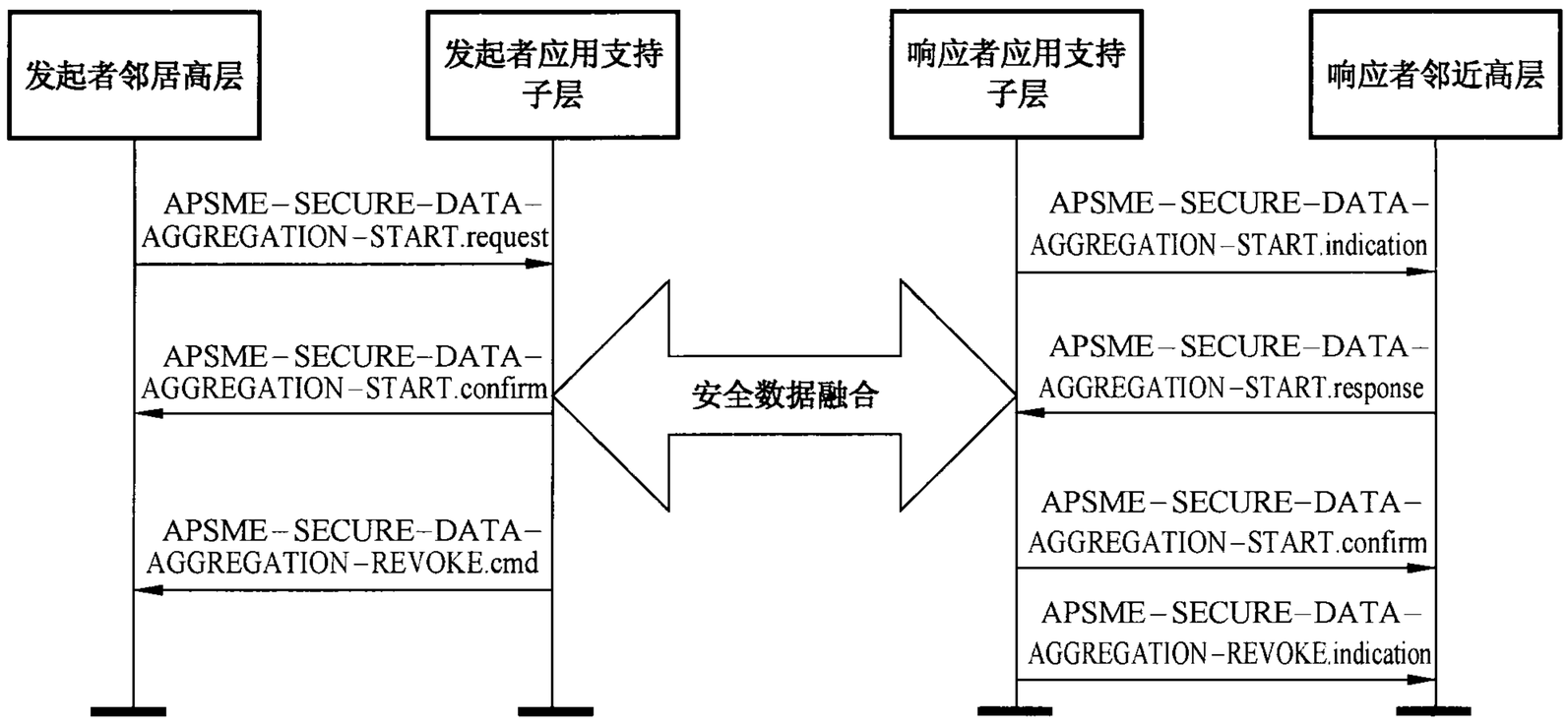


图 B.7 安全数据融合消息序列图

中华人民共和国
国家标准
信息技术 传感器网络
第602部分：信息安全：低速率无线传感器网络网络层和应用支持子层安全规范
GB/T 30269.602—2017
*
中国标准出版社出版发行
北京市朝阳区和平里西街甲2号(100029)
北京市西城区三里河北街16号(100045)
网址 www.spc.net.cn
总编室：(010)68533533 发行中心：(010)51780238
读者服务部：(010)68523946
中国标准出版社秦皇岛印刷厂印刷
各地新华书店经销
*
开本 880×1230 1/16 印张 4.5 字数 130千字
2018年1月第一版 2018年1月第一次印刷
*
书号：155066·1-58710 定价 60.00 元





GB/T 30269.602-2017