一种 LoRaWAN 根密钥更新协议及其安全性分析

李 心[1] 何 磊[1]
LI Xin HE Lei

摘 要 LoRaWAN 是一种低功耗广域网，它使用高级加密标准（AES）来保证消息的完整性和机密性，其根密钥由生产商分发存储于终端与服务器，并且在整个通信过程中不改变。根密钥用于生成后续会话密钥，攻击者一旦得到这个固定的根密钥，就可以开展一系列攻击，所以 LoRaWAN 的根密钥需要更新。基于此，提出了一种基于椭圆曲线迪菲 - 赫尔曼（ECDH）密钥交换算法的非交互式零知识的根密钥更新协议。基于椭圆曲线的 Diffie-Hellman 协议，在其中加入非交互式零知识，用于签名，以防止中间人攻击，同时保证密钥的后向安全性，并加入随机数、时间戳等机制，确保通信的安全性和完整性。通过 BAN 逻辑的形式化安全分析、理论分析和应用自动化验证工具（AVISPA），分析验证协议的安全性。结果表明，所提出的协议可以抵抗重放攻击、中间人攻击和其他常见攻击，提高了 LoRaWAN 的安全性。

关键词 LoRaWAN；ECDH；零知识证明

doi：10.3969/j.issn.1672-9528.2024.07.034

0 引言

随着世界的发展，网络成为人们越来越不可割舍的一部分。物联网（IOT）是指通过信息传感设备，按约定的协议，将任何物体与网络相连接，物体通过信息传播媒介进行信息交换和通信，以实现智能化识别、定位、跟踪、监管等功能[1]。物联网在人们生活中的应用十分广泛，为了提高人们的生活质量，改善工业制造的流程，物联网渗入各行各业，比如智能家居、医疗、农业等。

物联网的无线通信技术主要分为两类：一类是短距离通信技术，即 Zigbee、Wi-Fi、蓝牙等；另一类是广域网通信技术，即低功耗广域网（low-power wide-area network，LPWAN）[2]。LPWAN 又可分为两类：一类是 3GPP 支持的 2/3/4G 蜂窝通信技术，比如 EC-GSM、NB-IoT 等，工作于授权频谱下；另一类是 LoRa、SigFox 等技术，工作于未授权频谱。

LoRaWAN 是在 LoRa 物理层传输技术基础之上的以 MAC 层为主的一套协议标准，对应 OSI 七层模型中的数据链路层（MAC 层）。LoRaWAN 消除了具体硬件的不兼容性，同时还实现了自适应速率、信道管理、节点接入认证与数据加密等特性[3]。LoRaWAN v1.1 的拓扑结构包括许多终端设备、网关、网络服务器、加入服务器和应用服务器[4]，在终端设备与应用服务器的交互中利用 AES 来保护信息的有效载荷，并保证消息的完整性。但是 LoRaWAN 仍存在一些安全问题，在连续通信中，一些密钥保持不变，那么这些不变的密钥有泄露的危险，网络中的通信很容易被攻击，信息会被拦截。例如，LoRaWAN 的根密钥保存在终端设备和加入服务器中，缺乏自己的更新机制，一旦根密钥被攻击者知晓，LoRaWAN 的安全性就可能会崩溃。

为了解决根密钥的更新问题，本文提出了一种基于 ECDH 的非交互式零知识的密钥更新协议，以高效安全地更新 LoRaWAN 的根密钥。该协议基于 ECDH 密钥协商算法，并通过非交互式 Schnorr 进行数字签名，抵御中间人攻击，也通过零知识性隐藏信息，保证根密钥的后向安全性。此外，还加入随机数、时间戳来实现消息的机密性和完整性。该协议能在每次激活前获取更新的根密钥，能够防御多种形式的常见攻击。

1 相关技术概述

1.1 LoRaWAN

本文研究的 LoRaWAN 的版本是 v1.1。LoRaWAN 网络架构中包含了终端（lora device）、网关（gateway）、网络服务器（network server，NS）、加入服务器（join server，JS）和应用服务器（application server，AS），网关与终端之间采用星形拓扑结构，LoRaWAN 网络架构如图 1 所示。

1. 西藏民族大学信息工程学院 陕西咸阳 712000

[基金项目] 西藏民族大学项目“基于北斗与 LoRa 的西藏边远地区生态环境监测 WSN 系统研究”（22MDY014）

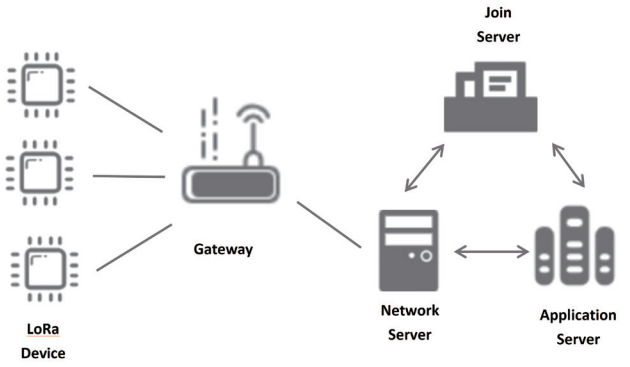

图 1 LoRaWAN 网络架构

网关的任务是在终端设备和网络服务器之间中继消息。在服务器中，NS 是 LoRaWAN 网络拓扑的中心，它将数据包从网络的每个设备发送到相关联的应用服务器。为了保护无线电传输的安全，LoRaWAN 协议依赖由设备根密钥产生的会话密钥的对称加密，设备的根密钥的存储和关联密钥的派生操作都由 JS 来保障。AS 接收端设备发送的所有上行链路消息，并为最终用户提供高级服务。表 1 中提供了用于描述 LoRaWAN 加入过程和所提出的协议的基本符号。

表 1 符号表

符号	描述
Join_Request	将终端设备连接到 LoRa 网络的请求
DevEUI	设备标识符
JoinEUI	加入服务器唯一标识符
DevNonce	设备随机生成的随机数值
Join_Accept	返回给终端设备的加入接受请求
AES128-CMAC(K,M)	基于 128 位 AES 的基于密码的消息认证码（cmac）的认证算法，密钥 K 和消息 M
JoinNonce	加入服务器随机生成的随机数值
Home_NetID	网络标识符
Devaddr	终端设备地址
Dlsetting	提供下行链路设置
RxDelay	发送和接收窗口之间的延迟时间
Cflist	可选网络参数列表（信道频率）
||	串联运算
Key_Update_Req	根密钥更新请求
Key_Update_Ack	根密钥更新接受请求
Key_Update_Ver	根密钥更新验证请求
aes128_encrypt	使用 128 位 AES 加密算法进行加密
pad_{16}	函数添加 0 个 8 位字节以使数据长度为 16 的倍数
ED_{st} 和 JN_{st}	时间戳

LoRaWAN 加密的使用分为两个独立的层：网络层和应用层 [5]。每个端设备都有一对不同的根密钥，一个密钥 NwkKey 用于网络层，另一个 AppKey 用于应用层。两个 AES-128 预共享根密钥在激活之前存储在终端设备和加入服务器的存储器中。

终端设备必须经过身份验证才能加入 LoRaWAN 网络。身份验证可以通过两种不同的方式执行：空中激活（OTAA）或个性化激活（ABP）。

在 OTAA 过程中，Join_Request 和 Join_Accept 消息在加入服务器和终端设备之间交换。新的会话密钥是基于 Join_Request 和 Join_Accept 消息中传输的两个根密钥和 nonce 值派生的。ABP 程序是一种更简单的激活方法，其中终端设备必须预先配置所有必要的加密密钥（终端设备不需要发送加入请求消息）。ABP 程序提供的安全性较低，因为加密密钥在终端设备的整个使用寿命内都将保持不变，所以本文讨论的入网方式都是 OTAA。

对于 OTAA 过程，通过不变的根密钥与随机生成的 nonce 值产生相应的会话密钥，后续的信息交互由会话密钥加密，会话密钥分为网络会话密钥和应用程序会话密钥。网络会话密钥 NwkSEncKey、FNwkSIntKey、SNwkSIntKey。FNwkSIntKey（转发网络会话完整性密钥）用于上行链路数据消息的消息完整性代码（MIC）；SNwkSIntKey（服务网络会话完整性密钥）用于下行链路数据消息的消息完整性码（MIC）；NwkSEncKey 和 AppSKey 密钥（网络和应用程序）用于之后交换的消息的机密性和完整性。应用程序会话密钥 AppSKey，是在终端和应用服务器之间共享的会话密钥，用于加密 / 解密应用层有效载荷。

根密钥既是作为终端和加入服务器之间的端到端加密密钥，又是作为 MIC 计算中的密钥，还是作为导出其他 LoRaWAN 会话密钥的安全系数。因此，LoRaWAN 的根密钥更新很重要。

1.2 非交互式 Schnorr 协议

零知识证明（zero-knowledge proof）是由 S.Goldwasser、S.Micali 及 C.Rackoff 在 20 世纪 80 年代初提出的 [6]。它指的是证明者能够在不向验证者提供任何有用的信息的情况下，使验证者相信某个论断是正确的。

在网络传输消息时，接收方希望得到的消息能够被证实在传输的过程中没有被篡改，且希望能够确认发送方的身份，也就是发送者能有一个私钥与这条消息进行关联计算。上述可以由数字签名完成，将非交互式 Schnorr 用于数字签名。首先，Schnorr 协议能够在不泄露私钥任何知识的情况下向对方证明“我”拥有私钥。其次，利用离散对数难题与 Hash 函数满足抗第二原象的假设保证攻击者不能随意伪造签名。图 2 为 Schnorr 签名方案 [7]。

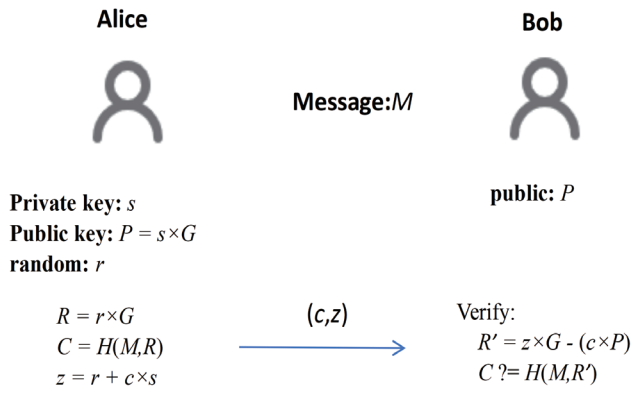

图 2 Schnorr 签名方案

设证明者（Alice）与验证者（Bob）双方拥有相同的生成元 G（G 为椭圆曲线上的一点），Alice 有密钥对 $<s,P>$，其中 $P= s\times G$，且 Bob 从可信第三方处获得 Alice 的公钥 P。Alice 在不透露自己私钥的情况下向 Bob 证明自己拥有私钥进行如下操作。

（1）Alice 随机选择一个标量 r，依次计算 $R=r\times G$，$c=H(M, R)$，$z=r+c\times s$，然后向 Bob 发送 (c,z)，$H(.)$ 为抗碰撞哈希函数。

（2）Bob 收到消息后，计算 $R'=z\times G-(c\times P)$，验证 C 是否等于 $H(M,R')$。若验证成功，则认为 Alice 确实具有匹配 P 的私钥 s，能够证明 Alice 的身份，且消息 M 在传递过程中未被篡改。

通过上述过程，以一种简洁的形式完成对于 Alice 的认证。整个过程中，Alice 并未暴露自己的私钥，且 Bob 无法通过正常或作弊手段获取 Alice 的私钥，因此也是零知识的。

2 密钥管理有关方案

如上所述，LoRaWAN 仍存在一些安全漏洞，许多作者在研究中提出了不同的密钥管理解决方案，这些研究主要集中在但不限于密钥的生成、交换和更新等。

Ismail Butun 等人 [8] 对 LoRaWAN v 1.1 的安全性进行了协议的全面安全风险分析，并针对每种威胁的影响和规模进行了详细的讨论和描述，指出虽然 LoRaWAN v 1.1 的安全性能相较于 LoRaWAN v 1.0 有所提高，但仍然存在一些安全风险，比如终端的物理捕获、重放攻击和中间人攻击等。最后给出了详细的安全建议和未来规划。

Ilsun You 等人 [9] 提出了一种基于增强椭圆曲线（ECDH）的密钥交换协议，与数据传输层安全预共享密钥（DTLS-PSK）和数据传输层安全椭圆曲线密码（DTLS-ECC）相比，在网络延迟、信令开销方面有显著改进和更好的性能。但是该论文侧重于 v 1.0，LoRaWAN v 1.1 无法支持终端与应用服务器之间的端到端安全。

Victor Ribeiro 等人 [10] 将 LoRa 与私有区块链和智能合约相结合，提出了一种密钥管理安全架构，用来提高 LoRaWAN 环境的安全性。将 JS 作为私有区块链基础设施的客户端，将加密密钥由多个对等方存储，由智能合约来管理，实现密钥的存储、更新、获取和销毁。该方案通过结合区块链，采用分布式账本解决 JS 的集中存储问题，使用智能合约管理所有终端设备的加密密钥。这给我们提供了一种与区块链相结合的新思路，但是并未考虑到密钥不更新的安全问题。

Xingda Chen 等人 [11] 提出了一个完整的密钥管理方案，包括密钥更新、密钥生成、密钥备份和密钥向后兼容性问题。通过一种值得信赖的集中式密钥管理服务器（CKMS），处理密钥的生命周期，即密钥派生、更新、备份和吊销，还提出了一种基于 Rabbit PENG 的密钥生成方案来代替 AES 算法，并证明该方案有相当的随机性和唯一性。但是由于该方法涉及其他额外的设备，在安全方面会导致其他的安全问题。

Kun-Lin Tsai 等人 [12] 采用了带有动态替换框的改进 AES 算法，提出了一种两阶段的加密密钥更新方案，即根密钥更新和会话密钥更新。改进后的 AES 的 D-box 加密周期简化，此方案具有相互认证和消息完整性的特点，能够抵抗重放和窃听攻击。其更新密钥需要由应用服务器产生随机数并与加入服务器和终端设备通信，加入其余设备参与通信，改变了原来的通信结构，开销变大。

3 提出的根密钥更新协议

对于更新根密钥，本研究在 OTAA 前添加三条新消息用于交换数据，分别是 Key_Update_Req、Key_Update_Ack、Key_Update_Ver，并用基于 HMAC 的密钥推导函数（HKDF）派生新密钥。接下来，使用更新后的根密钥完成激活生成新的会话密钥。具体协议描述可见图 3。

步骤 1：

预先在 LoRa Device 和 Join Server 存储相同的密钥对 $<r, R>$，且设定相同的椭圆曲线参数（固定），公钥 $R=r\times G$（G 为选取的椭圆曲线的基点）。

对于 LoRa Device，随机选取范围在 $[1,n-1]$ 的一个数作为私钥 r_d，则公钥 $R_d=r_d\times G$。消息 $M=H(\text{AES128NwkKey}, \text{Nonce}_1||R_d)$，该消息在传输过程中需要不被篡改，保证信息的完整性。计算消息 M 时，通过 AES 算法加密 $\text{Nonce}_1||R_d$，这里的密钥使用了 NwkKey。生成随机数 c 进行挑战，$c=H(M,R_d)$，根据挑战 c 构造响应：$z=r_d+c\times r$。

LoRa Device 构造会话密钥更改请求并向 Join Server 发送：

Key_Update_Req{DevEUI,JoinEUI,c,z,Nonce_1,ED_{st}}。

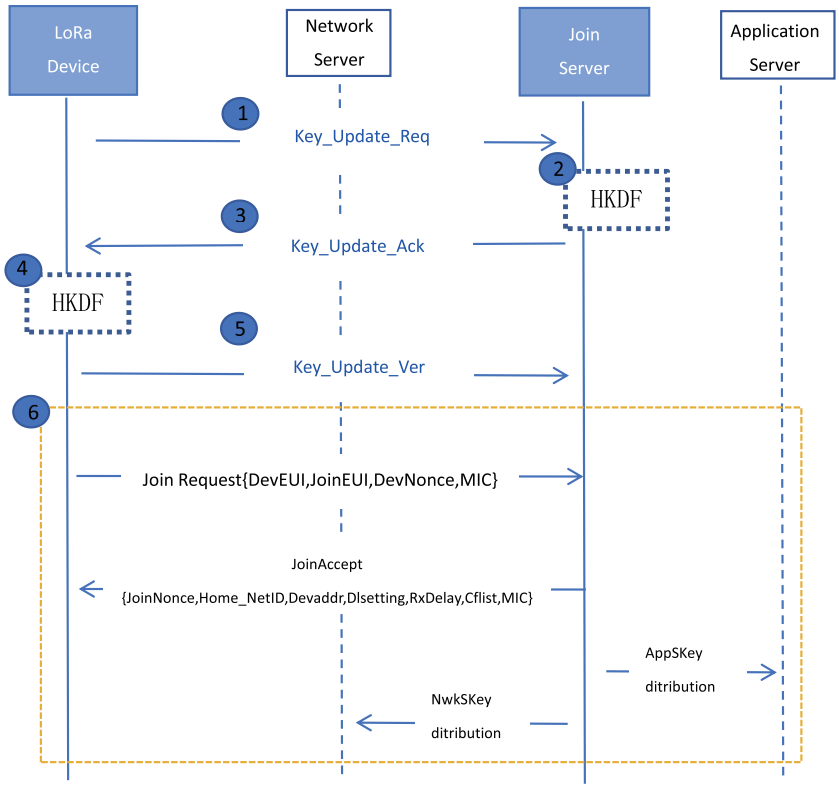

图 3 根密钥更新过程

步骤 2：

对于 Join Server，随机选取范围在 [1,n-1] 的一个数作为私钥 r_j，$R_j=r_j\times G$。接收到密钥更新请求后，解码消息根据 $z=r_d+c\times r_1$，两边同时乘以 G，可算出 $R_d'=z\times G-(c\times R)$。接着，验证 $c=H(M,R_d')$ 是否成立。附加 MIC 保证消息整体传输的完整性，MIC_1=AES128-CMAC(NwkKey,key_Update_Ack)。

通过 ECDH 协议交换获得双方相同的共享机密 NK=$r_j\times R_d=r_j\times r_d\times G$，将共享机密 NK 与原始密钥 AppKey 和 NwkKey 相结合，形成新的字符串 S，S_1=AppKey||NK 和 S_2=NwkKey||NK。

生成新密钥的过程需要使用 Extract 函数和 Expand 函数与哈希算法一起导出新密钥。Extract 函数使用输入的原始密钥，派生出一个符合密码学安全伪随机性的伪随机密钥。Expand 函数使用 Extract 派生出的伪随机密钥，扩展出指定长度的密钥（同时仍保证密码学安全伪随机性）。

先使用 Extract 函数生成中间密钥 PRK：

PRK=Extract(salt,S)

Salt 作为输入，是加盐操作的盐，如果不提供则全部初始化为 0 的字符串，长度则为所采用哈希函数的散列值长度。使用 Salt 增加原始密钥材料 IKM 的随机性。

再用中间密钥和文本哈希通过 Expand 函数生成对称密钥 SK：

TH=(Key_Update_Req)

SK=Expand(PRK,TH)

这里将新生成的对称密钥重新命名，由 S1 输入的命名为 NappKey，由 S2 输入的命名为 NnwkKey。

步骤 3：

验证成立，则合法，Join Server 发送回应：

Key_Update_Ack$\{Nonce_2,R_j,MIC_1,JN_{st}\}_{NwkKey}$

步骤 4：

LoRa Device 收到 key_Update_Ack 后用 NwkKey 解码得到 R_j，通过 ECDH 协议交换计算得到双方相同的共享机密 NK=$R_j\times r_d=r_d\times r_j\times G$，将共享机密 NK 与原始密钥 AppKey 和 NwkKey 结合形成新字符串 S，S_1=AppKey||NK 和 S_2=NwkKey||NK。

与前面相同，先使用 Extract 函数生成中间密钥 PRK：

PRK=Extract(salt,S)

再用中间密钥和文本哈希通过 Expand 函数生成对称密钥 SK：

TH=(Key_Update_Ack)

SK=Expand(PRK,TH)

这里将新生成的对称密钥重新命名，由 S_1 输入的命名为 NappKey，由 S_2 输入的命名为 NnwkKey。

步骤 5：

使用新生成的密钥 NnwkKey 发送验证请求：

Key_Update_Ver$\{Nonce_3, MIC_2, ED_{st}\}_{NnwkKey}$

其中，MIC_2=AES128-CMAC (NnwkKey,Key_Update_Ver)。

Join Server 若生成相同密钥 NnwkKey，则可以解密 key_Update_Ver 消息。接下来，LoRa Device 与 Join Server 通信就可以使用新密钥 NnwkKey 进行。

步骤 6：

在接下来的空中激活中，网络会话密钥 NwkSEncKey、FNwkSIntKey、SNwkSIntKey 由新根密钥 NnwkKey 创建：

FNwkSIntKey = aes128_encrypt(NnwkKey,0x01|JoinNonce|JoinEUI|DevNonce|pad_{16})

SNwkSIntKey = aes128_encrypt(NnwkKey,0x03|JoinNonce|JoinE UI|DevNonce|pad_{16})

NwkSEncKey = aes128_encrypt(NnwkKey,0x04|JoinNonce|JoinEUI|DevNonce|pad_{16})

应用程序会话密钥 AppSKey 由新根密钥 NappsKey 创建：

AppSKey = aes128_encrypt(NappKey,0x02|JoinNonce|Join

EUI|DevNonce|pad_{16})

Join Server 将 AppSKey 分发给 Application Server，将三个网络会话密钥分发给 Network Server。

4 对协议的安全性分析

本节采用了 3 种分析方式对协议进行分析，首先是使用 BAN 逻辑对协议进行逻辑证明，再进行理论分析，最后基于 AVISPA 进行形式化分析，三者都可以证明该协议的安全可靠性。

4.1 BAN 逻辑

BAN 逻辑是基于知识与信任的一种形式逻辑分析方法[13]。它通过对认证协议的运行进行形式化分析，从协议执行者最初的一些基本信仰出发，根据每个参与者收发的消息，推理得到最终信仰。

4.1.1 基本术语

BAN 逻辑有如表 2 中的常用符号与表达式。

表 2 BAN 逻辑中的符号与定义

表达式	描述
P、Q	通信主体
X、Y	任意语句
$P\mid\equiv(X)$	P 认为 X 为真
$P\lhd X$	P 曾收到包含 X 的消息
$\#(X)$	X 为新鲜的
$P\mid\sim X$	P 曾经说过 X
$P\overset{k}{\longleftrightarrow}Q$	P、Q 可使用共享密钥 K 通信
$\overset{k}{\longrightarrow}P$	K 是 P 的公钥
$\{X\}_k$	用密钥 K 加密 X 的结果
$P\overset{X}{\longleftrightarrow}Q$	P、Q 共享秘密 X

4.1.2 推理规则

推理规则包含有消息含义规则、随机数验证规则、裁判规则和新鲜性规则等。本节给出此次推理所需的规则，其余规则可参考文献 [13]。

（1）规则 1：消息含义规则

$$\frac{P\mid\equiv P\overset{k}{\longleftrightarrow}Q, P\lhd\{X\}k}{P\mid\equiv Q\mid\sim X}$$

表示 P 相信 K 是 P、Q 之间的密钥，当 P 看到 K 加密的 X 时，则相信 Q 曾经说过 Q。

（2）规则 2：随机数验证规则

$$\frac{P\mid\equiv\#(X), P\mid\equiv Q\mid\sim X}{P\mid\equiv Q\mid\equiv X}$$

表示 P 相信 X 是新鲜的，且 P 相信 Q 说过 X，则 P 相信 Q 相信 X 是真实的。

（3）规则 3：ECDH 规则

$$\frac{P\mid\equiv Q\mid\sim\overset{yG}{\longrightarrow}Q, P\mid\equiv\overset{xG}{\longrightarrow}P}{P\mid\equiv P\overset{xyG}{\longleftrightarrow}Q}$$

表示 P 相信 Q 说过 yG 是 Q 的公钥，且 P 相信 xG 是 P 的公钥，则 P 相信 P 和 Q 之间共享会话密钥 xyG。

（4）规则 4：新鲜性规则

$$\frac{P\mid\equiv\#(X)}{P\mid\equiv\#(X,Y)}$$

表示 P 相信 X 是新鲜的，则 P 相信 (X,Y) 也是新鲜的。

（5）规则 5：信仰规则（一）

$$\frac{P\mid\equiv X, P\mid\equiv Y}{P\mid\equiv(X,Y)}$$

（6）规则 6：接受规则

$$\frac{P\mid\equiv Q\mid\sim(X,Y)}{P\mid\equiv Q\mid\sim X}$$

（7）规则 7：信仰规则（二）

$$\frac{P\mid\equiv Q\mid\equiv(X,Y)}{P\mid\equiv Q\mid\equiv X}$$

4.1.3 准备工作

使用上述 BAN 逻辑术语与推理规则，证明本文设计的 LoRaWAN 密钥更新方案能够完成安全的密钥更新、共享和身份认证。JS 表示加入服务器，ED 表示终端设备。

期目如下：

目标 1：$JS\mid\equiv ED\mid\equiv JoinEUI, DevEUI, N_1, R_d$

目标 2：$ED\mid\equiv JS\mid\equiv N_2, R_j$

目标 3：$JS\mid\equiv JS\overset{Nnwk}{\longleftrightarrow}ED$

目标 4：$ED\mid\equiv ED\overset{Nnwk}{\longleftrightarrow}JS$

理想模型如下：

消息 1：$ED\rightarrow JS$ ： $<JoinEUI, DevEUI, R_d, N_1>_{NwkKey}$

消息 2：$JS\rightarrow ED$ ： $<N_2, R_j, \{N_2, R_j\}_{NwkKey}>_{NwkKey}$

消息 3：$ED\rightarrow JS$ ： $<N_3, \{N_3\}_{Nnwk}>_{Nnwk}$

初始假设如下：

假设 1：$JS\mid\equiv JS\overset{NwkKey}{\Longleftrightarrow}ED$

假设 2：$ED\mid\equiv ED\overset{NwkKey}{\Longleftrightarrow}JS$

假设 3：$JS\mid\equiv JS\overset{NwkKey}{\longleftrightarrow}ED$

假设 4：$ED\mid\equiv ED\overset{NwkKey}{\longleftrightarrow}JS$

假设 5：$JS\mid\equiv\#(N_1)$

假设 6：$ED\mid\equiv\#(N_2)$

假设 7：$JS\mid\equiv\#(N_3)$

假设 8：$ED\mid\equiv\overset{R_d}{\rightarrow}ED$

假设 9：$JS\mid\equiv\overset{R_j}{\rightarrow}JS$

4.1.4 逻辑推导

主要推理过程如下。

对于目标 1：

（1）由消息 1，有：

$JS \lhd <JoinEUI, DevEUI, N_1, R_d>_{NwkKey}$

（2）由逻辑规则 1，假设 1 和上述（1）可得：

$JS|\equiv ED|\sim JoinEUI, DevEUI, N_1, R_d$

（3）由逻辑规则 4，假设 5 得：

$JS|\equiv \#(JoinEUI, DevEUI, N_1, R_d)$

（4）由逻辑规则 2，上述（2）和（3）得：

$JS|\equiv ED|\equiv JoinEUI, DevEUI, N_1, R_d$

加入服务器 JS 相信终端设备 ED 的真实性。结束目标 1 的推导。

对于目标 2：

（5）由消息 2，有：

$ED \lhd <N_2, R_j>_{NwkKey}$

（6）由逻辑规则 1，假设 4 和上述（5）可得：

$ED|\equiv JS|\sim N_2, R_j$

（7）由逻辑规则 4，假设 6 可得：

$ED|\equiv \#(N_2, R_j)$

（8）由逻辑规则 2，上述（6）和（7）得：

$ED|\equiv JS|\equiv N_2, R_j$

终端设备 ED 通过验证相信加入服务器 JS 的身份。完成目标 2 的推导。

对于目标 3：

（9）由逻辑规则 6 和上述（2）有：

$JS|\equiv EDRd$

（10）由逻辑规则 3，假设 9 和上述（9）可得：

$JS|\equiv SK$

这里 $SK=R_j\times r_d\times G$。

（11）由逻辑规则 7，上述（4）和（10）得：

$JS|\equiv JS \overset{Nnwk}{\longleftrightarrow} ED$

目标 3 推导完成，表明协议在 ED 与 JS 两者间安全的生成了共享密钥。

对于目标 4：

（12）由逻辑规则 6 和上述（6）有：

$ED|\equiv JS|\sim R_j$

（13）由逻辑规则 3，假设 8 和上述（12）可得：

$ED|\equiv SK$

这里 $SK=R_d\times r_j\times G$。

（14）由逻辑规则 7，上述（8）和（13）得：

$ED|\equiv ED \overset{NwkKey}{\longleftrightarrow} JS$

目标 4 推导完成，所有目标完成推理，证明了本文提出的协议在逻辑上是安全可靠的。

4.2 理论分析

该协议能够抵御常见攻击，下面给出具体分析。

（1）抗重放攻击。本文提出的协议在消息中引入了随机数和时间戳，每条消息的随机数和时间戳都不同，且保存在服务器端。若服务器检测到收到消息的随机数重复或时间不同步，即可认定受到重放攻击，并丢弃该信息。

（2）中间人攻击。本协议引入了非交互式 Schnorr 用于数字签名。该签名方案不仅可以完成认证，也是零知识的。该方案使用只有终端与加入服务器才知晓的根密钥。

（3）前向安全性。前向安全性是指当前密钥被攻击者获取后，历史的密钥是安全的。当攻击者破解了某次通信中的密钥后，新根密钥是由上一次的根密钥联合共享机密通过提取、扩展的 HKDF 算法产生的，因此新根密钥与之前的密钥具有一定的独立性，攻击者并不能依此破解之前的密钥。

（4）后向安全性。后向安全性是指当前密钥被攻击者获取后，未来的密钥是安全的。当攻击者破解了某次通信中的密钥后，由于在密钥的更新过程中，不仅需要当前的密钥，还需要通过零知识证明验证自己的身份。攻击者在伪装成通信一方时，无法通过零知识证明自己，那么另一方将会终止与之通信。

4.3 AVISPA

AVISPA 是一种用于自动证明网络安全协议与应用的工具集 [14]。使用广泛接受的 AVISPA 工具模拟了本文的方案，使用 Dolev–Yao 攻击模型 [15]，表明此方案对被动和主动攻击（包括重放和中间人攻击）是安全的。

AVISPA 的输出格式使用后端生成（4 个）：OFMC、CL-AtSe、SATMC、TA4SP，其中选择支持异或操作的实施模型检查器（OFMC）、基于约束逻辑的攻击搜索器（CL-AtSe）这两个后端进行测试。

首先，仿真模型设计了两个基本角色，分别是协议中的终端设备 D 和加入服务器 S。其次，对协议环境、会话和保密性目标也进行了定义。在两个后端测试中，均得到“SUMMARY”下结果为“SAFE”，结果表明协议在 AVISPA 认证下是安全的，能够抵御常见的主被动。验证结果如图 4 所示。

```
% OFMC
% Version of 2006/02/13
SUMMARY
 SAFE
DETAILS
 BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
 /home/span/span/testsuite/results/NIZK.if
GOAL
 as_specified
BACKEND
 OFMC
COMMENTS
STATISTICS
 parseTime: 0.00s
 searchTime: 0.14s
 visitedNodes: 2 nodes
 depth: 1 plies
```

```
SUMMARY
 SAFE

DETAILS
 BOUNDED_NUMBER_OF_SESSIONS
 TYPED_MODEL

PROTOCOL
 /home/span/span/testsuite/results/NIZK.if

GOAL
 As Specified

BACKEND
 CL-AtSe

STATISTICS

 Analysed  : 1 states
 Reachable  : 0 states
 Translation: 0.01 seconds
 Computation: 0.00 seconds
```

图 4 OFMC 与 CL-AtSe 的验证结果

（下转第 169 页）

[7] 李阳 , 何文峰 , 黄伦春 . 一种设施普查中多源异构数据的处理方法 [J]. 城市勘测 ,2023(S1):181-184+204.
[8] 王彩霞 , 陶健 . 数据库中多源异构异常数据清洗方法 [J]. 通化师范学院学报 ,2023,44(12):54-60.
[9] 程雪婷 , 王玮茹 , 暴悦爽 , 等 . 基于联邦学习的多源异构数据安全融合方法 [J]. 通信技术 ,2023,56(10):1173-1183.
[10] 李坚 , 杨峰 , 吴佳 , 等 . 基于改进 FCM 的多源异构能源数据预处理与去噪 [J]. 微型电脑应用 ,2023,39(10):80-82+87.
[11] 杨桥桥 , 洪东彬 , 周扬 , 等 . 互联网背景下基于个性化推送的供电服务学习机制 [J]. 互联网周刊 ,2022(1):40-43.
[12] 李学威 , 孙滨 . 基于深度学习的自适应移动学习服务智能推荐研究 [J]. 信息与电脑 (理论版),2023,35(3):254-256.

【作者简介】

谢梦怡（1986—），女，福建泉州人，硕士，讲师，研究方向：云计算、大数据。

（收稿日期：2024-03-29）

（上接第 164 页）

5 结语

本文设计了一种基于椭圆曲线的 Diffie-Hellman 的零知识根密钥更新协议。在设备激活前，通过带有零知识性的密钥交换算法生成共享机密，再由密钥推导函数将共享机密与上一次根密钥作为原材料派生出新的根密钥，解决 LoRaWAN 的 OTAA 中根密钥不更新的问题。通过对协议的验证分析，证明该协议是安全的，能够抵御常见攻击。

参考文献：

[1] 黄长清 . 智慧武汉 [M]. 武汉 : 长江出版社 ,2012.
[2]XU Z, JIE N.A study on key LPWAN technologies[EB/OL].(2021-04-11)[2024-03-21].https://iopscience.iop.org/article/10.1088/1742-6596/1871/1/012011.
[3]LORA ALLIANCE TECHNICAL COMMITTEE.LoRaWAN-backend-interfaces-v1.0[EB/OL].(2017-10-11)[2024-03-21]. https://lora-alliance.org/wp-content/uploads/2020/11/lorawantm-backend-interfaces-v1.0.pdf.
[4]LORA ALLIANCE TECHNICAL COMMITTEE.LoRaWAN® specification v1.1[EB/OL].(2017-10-11)[2024-03-21].https://resources.lora-alliance.org/technical-specifications/lorawan-specification-v1-1.
[5]HAN J, WANG J.An enhanced key management scheme for LoRaWAN[C]//Security,Privacy,and Anonymity in Computation,Communication,and Storage.Cham: Springer, 2018: 407-416.
[6]ANATOLY K, SERGEY Z.Zero knowledge proof and ZK-SNARK for private blockchains[J].Journal of computer virology and hacking techniques,2023,19(3):443-449.
[7]SCHNORR C P.Efficient signature generation by smart cards[J]. Journal of cryptology,1991,4:161-174.
[8]ISMAIL B, NUNO P, MIKAEL G.Security risk analysis of LoRaWAN and future directions[J].Future internet, 2019, 11(1): 1-22.
[9]ILSUN Y, SOONHYUN K, GAURAV C, et al.An enhanced LoRaWAN security protocol for privacy preservation in IoT with a case study on a smart factory-enabled parking system[J]. Sensors,2018,18(6):1888.
[10] VICTOR R, RAIMIR H F, ALEX R.A secure and fault-tolerant architecture for LoRaWAN based on blockchain[C]//2019 3rd Cyber Security in Networking Conference. Piscataway: IEEE, 2019:35-41.
[11]CHEN X, LECH M, WANG L.Complete key management scheme for LoRaWAN v1.1[J].Sensors,2021,21(9):2962.
[12]TSAI K, CHEN L, LEU F, et al.Two-Stage high-efficiency encryption key update scheme for LoRaWAN based IoT environment[J]. Computers,materials& continua, 2022, 73(1): 547-562.
[13]BURROWS M, ABADI M, NEEDHAM R.A logic of authentication[J].ACM transactions on computer systems, 1990, 8(1):18-36.
[14]ARMANDO A, BASIN D, BOICHUT Y, et al.The AVISPA tool for the automated validation of internet security protocols and applications[C]//Computer Aided Verification.Berlin: Springer-Verlag, 2005:281-285.
[15]DOLEV D, YAO A.On the security of public key protocols[J]. IEEE transactions on information theory,1983,29(2):198-208.

【作者简介】

李心（1999—），女，四川眉山人，硕士，研究方向：无线传感器网络。

（收稿日期：2024-04-23）