文章编号:1674-2974(2022)10-0103-08 DOI:10.16339/j.cnki.hdxbzkb.2022352

基于PUF实现物联网设备的轻量级密钥共享协议

王振宇[1†],李少青[1],郭阳[1],曾健平[2]
(1. 国防科技大学 计算机学院,湖南 长沙 410000;
2. 湖南大学 物理与微电子科学学院,湖南 长沙 410082)

摘 要:物联网承载着大量敏感信息的安全传输与存储.由于物联网设备资源有限,通信开销大、传输速率慢且需存储敏感信息的安全原语(如公钥算法、数字签名等)不适用于轻量级设备的认证.本文利用硬件物理不可克隆函数(PUF)具备的防篡改防克隆特性来生成共享密钥,结合MASK算法、Hash函数等安全原语,为物联网设备提出一种轻量级匿名密钥共享安全认证协议.通过Ban逻辑和形式化工具ProVerif进行安全分析验证,证明该协议能够防御中间人攻击、去同步攻击、假冒攻击、建模攻击等.通过对比其他协议,证明该协议具备计算成本低、通信开销和存储容量小以及安全性能高的优点,适合于资源受限设备的安全通信传输.

关键词:物理不可克隆函数;轻量级;密钥共享;认证协议;物联网
中图分类号:TN918 **文献标志码**:A

A Lightweight Key Sharing Protocol for IoT Devices Based on PUF

WANG Zhenyu[1†], LI Shaoqing[1], GUO Yang[1], ZENG Jianping[2]
(1. College of Computer Science and Technology, National University of Defense Technology, Changsha 410000, China;
2. School of Physics and Microelectronics Science, Hunan University, Changsha 410082, China)

Abstract: The Internet of Things(IoT) carries the safe transmission and storage of a large amount of sensitive information. Since IoT devices are resource-constrained, which have expensive communication, slow mission velocity and need to store sensitive information security primitives (such as public key algorithm and digital signature), they are not suitable for the authentication of lightweight IoT devices. This paper proposes a lightweight anonymous key sharing security authentication protocol for IoT devices, which generates a shared key by the Physical Unclonable Function(PUF) and uses security primitives such as the MASK algorithm and the Hash function. The security analysis and verification are accomplished by Ban logic and ProVerif to prove that the protocol ensures security attributes such as anonymity, non-repudiation, and forward/backward confidentiality. Compared with other protocols, this protocol has the characteristics of low computing cost, small communication overhead and storage capacity, and high security performance, which is suitable for the secure communication transmission of resource-constrained devices.

Key words: physical unclonable function; lightweight; key sharing; authentication protocol; internet of things

---

* **收稿日期**:2021-11-11
基金项目:国家自然科学基金资助项目(61832018), National Natural Science Foundation of China(61832018)
作者简介:王振宇(1995—),男,湖南岳阳人,国防科技大学博士研究生
† 通信联系人,E-mail: wangzhenyu19a@nudt.edu.cn

随着传感、自动化和通信技术的飞速发展,物联网(Internet of Things, IoT)产生的海量数据给设备之间有效安全传输、存储与保护带来了巨大的威胁[1].传统的网络安全协议会采用加密算法[2]、数字签名、Hash 函数[3]、消息验证码等复杂安全原语来保证信息传输的机密性、完整性以及不可否认性等[4].由于物联网设备通常体积小、资源约束强且硬件处理能力低,因此传输速率慢、通信开销大的安全原语不适用于轻量级设备的认证[5].

目前物联网的安全通信是假设硬件设备与系统是安全的,但是恶意攻击者可以通过芯片克隆、解剖等手段来破坏设备保密信息[6].信息加密是保护信息安全认证的有效途径,但是加密算法的密钥通常存储在非易失性存储器(NVM)中,攻击者通过侧信道和物理攻击,可以成功读取存储器中的私密信息[7].物理不可克隆函数(PUF)作为一种新兴的硬件安全原语,利用芯片在制造过程中无法控制的随机工艺偏差,来产生器件独有的数字签名[8].PUF 采用特定的"激励-响应"(CRP)机制触发可以实时生成安全密钥,对物理篡改非常敏感,无须存储,硬件开销小,可以解决传统密钥面临的安全问题,并且适用于轻量级物联网设备安全认证协议[9-11].

密钥共享认证协议通常在软件层采用公钥算法与数字签名完成,但是这些加密原语运行速度慢,通信开销大,且新型量子计算方法能够有效对公钥算法进行破解.为解决物联网设备在信道传输及密钥存储方面存在的安全性问题,本文采用可重构 CRO PUF 防篡改防克隆的特性生成共享密钥,取代通信开销大的非对称加密算法和数字签名,结合 MASK 算法、Hash 函数等加密方法提出一种轻量级匿名密钥共享安全认证协议.该协议确保匿名性、可用性、完整性和前向/后向保密性等安全属性.

1 数学理论知识与相关工作

1.1 基于 PUF 的密钥共享机制

Zhang 等人[12]采用 CRO PUF 为设备生成相同的共享密钥,适用于一对多的安全认证协议.该机制通过两个阶段来获得设备之间的共享密钥.

阶段 1:生成共享密钥的可靠响应.通过建模获取 CRO PUF 的高精度延迟矩阵 $\boldsymbol{R}$;计算所有路径之间的延迟差并按绝对值降序排序.考虑不同温度影响来确定阈值 T,当两条路径之间延迟差的绝对值大于阈值时,则输出响应是稳定的.

阶段 2:生成共享密钥的激励.将延迟矩阵 $\boldsymbol{R}$ 的所有路径存入集合 D,列举共享密钥 K 的所有位.从集合 D 中随机选择两条不同的路径,对于共享密钥 K 的每一位 K_i,如果 K_i 等于 1 且延迟差大于 T,则表示找到一个可以产生稳定响应 1 的配置激励 C_i;如果 K_i 等于 0 且延迟差小于 $-T$,则表示找到一个可以生成稳定响应 0 的配置激励 C_i;否则,重新选择共享密钥 K.

1.2 MASK 与 UNMASK 算法

Qureshi 等人[13]提出的 MASK 算法包含三个参数:长度 l 位的输入向量 $\boldsymbol{r}=[r_1, r_2, r_3, \cdots, r_l]$、$l$ 个正整数集 $K=\{k_1, k_2, k_3, \cdots, k_l | k_i \in \mathbf{Z}^+\}$ 以及产生 l 位的输出向量 $\boldsymbol{r}_m=[r_{m1}, r_{m2}, r_{m3}, \cdots, r_{ml}]$. MASK 算法是采用正整数集 K 作为辅助数据,将长度 l 位输入向量 $\boldsymbol{r}$ 来生成一个等长度的输出函数 $\boldsymbol{r}_m$. 正整数集 K 通过伪随机数发生器生成 $K \leftarrow \mathrm{PRNG}_i(\boldsymbol{y})$,其中 $\boldsymbol{y}$ 是长度 n 位的输入向量,且 $\boldsymbol{y}=[y_1, y_2, y_3, \cdots, y_n]$. 类似地,MASK 函数的可逆变换 UNMASK 函数,使用正整数集 K 将输出函数 $\boldsymbol{r}_m$ 为复原输出函数 $\boldsymbol{r}$.其转换表达式如下:

$\mathrm{MASK}(\boldsymbol{r}, K): \boldsymbol{r} \Rightarrow \boldsymbol{r}_m$

$\mathrm{UNMASK}(\boldsymbol{r}_m, K): \boldsymbol{r}_m \Rightarrow \boldsymbol{r}$

算法 1(Alogrithm 1)总结了 MASK 函数的运算过程,包括整数集生成、函数范围变换和位混淆等三个步骤.算法 1 具体过程如图 1 所示.

Algorithm 1:Mask 算法过程

输入:

l 位输入向量 $\boldsymbol{x}$

n 位输入向量 $\boldsymbol{y}$

输出:

l 位输入向量 $\boldsymbol{x}_{\mathrm{m}}$

1:produce MASK($\boldsymbol{x}$,$\boldsymbol{y}$)

2: interger Set K: $\{k_1, k_2, k_3, \cdots, k_m\} \rightarrow \mathrm{PRNG}_1(y)$

3: for $i \leftarrow 1$ to m do

4: $N_{\mathrm{new}} \leftarrow \mathrm{RANGE}(k_i, m+1-i)$

5: $x \leftarrow \mathrm{SWAP}(x_{\mathrm{new}}, x_{m-i+1})$

6: $x_m \leftarrow x$

7: end for

8: end procedure

图 1 算法 1 步骤图

Fig.1 Step diagram of algorithm 1

1)整数集生成:向量 $\boldsymbol{y}$ 作为伪随机数发生器 PRNG 电路的种子生成一组正整数 $K=\{k_1, k_2, k_3, \cdots, k_l | k_i \in \mathbf{Z}^+\}$. 整数集 K 包含 l 个 n 位正整数,任何一位正整数的最大值为 2^n-1.

2)函数范围变换:定义一个范围函数Range()为线性映射变换,给定一个l位整数$\{k|k \in K\}$,其取值范围在$[0, 2^l - 1]$,生成一个m位的新集合$Q = \{q|q \in \mathbf{Z}^+\}$,其整数范围在$[0, 2^m - 1]$,其中$m \leqslant l$.线性范围映射由以下等式控制:

$$N_{new} = [\frac{(N_{old} - N_{oldmin}) \times (N_{newmax} - N_{newmin})}{(N_{oldmax} - N_{oldmin})}]$$

$N_{old} \in K$是Range()函数的输入,N_{oldmin}与N_{oldmax}是在范围$[0, 2^l - 1]$内的最小值与最大值,N_{newmin}与N_{newmax}是在新范围$[0, 2^m - 1]$内的最小值与最大值.

3)位混淆:MASK函数最后是基于Fisher-Yates Shuffler洗牌算法完成序列的位混淆,n个不同元素的有限序列生成一个$n!$个随机排列的算法.

MASK算法有两个优点:1)它有效地隐藏了设备PUF激励与响应之间的关系;2)它为设备的输入提供了验证.在不验证输入流的情况下,PUF不会被激活,因此设备不会产生任何响应,这有效地防止了设备受到任何蛮力攻击.

2　协议设计与分析

本文基于嵌入PUF的物联网设备提出了一种轻量级密钥共享认证协议,包括协议注册阶段和双向认证与固件更新阶段.协议相关符号如表1所示.

表1　协议相关符号说明

Tab.1　Symbol description

符号	含义
CRO PUF	可重构环形振荡器PUF
Delay Matrix $\boldsymbol{M}_A/\boldsymbol{M}_B$	延时矩阵$\boldsymbol{M}_A$或$\boldsymbol{M}_B$
(C, R)	PUF产生的激励C与响应R
timestape()	时间戳函数
(n_{i1}, n_{i2})	伪随机数
Hash(·)	单向哈希函数
Fisher - Shuffler()	洗牌混淆算法
PRNG()/TRNG()	伪/真随机数发生器

2.1　协议注册阶段

协议注册阶段采用Zhang等人[12]所提出的基于可重构CRO PUF的密钥共享机制,解决传统网络协议中公钥算法的开销大、处理速度低以及密钥存储的安全问题.注册阶段是在安全信道中传输,执行算法1,具体步骤如图2所示.设备与服务器注册阶段如图3所示.

算法1:协议注册阶段

1: Device. PUF_A, Device. $PUF_B \leftarrow$ Select. Device()
2: $ID_A, ID_B \leftarrow$ Decice.random()
3: Delay. $Matrix_A$, Delay. $Matrix_B \leftarrow$ Device. PUF_A, Device. PUF_B
4: $T_A, T_B \leftarrow$ Algorithm1(Delay. $Matrix_A$, Delay. $Matrix_B$)
5: Shared. key $R \leftarrow$ Server. random()
6: Challenge $C_A, C_B \leftarrow$ Algorthm2{$(T_A, C_A), (T_A, C_B)$}
7: Store$(C_A, ID_A), (C_B, ID_B)$

图2　协议注册阶段算法步骤图

Fig.2　Algorithm steps in registration phase of the proposed protocol

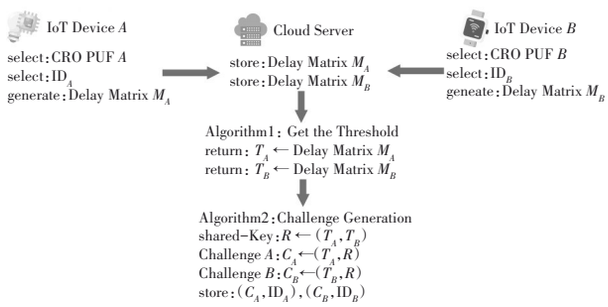

图3　协议注册阶段

Fig.3　Registration phase of the proposed protocol

1)设备A与设备B通过建模CRO PUF分别获得延时矩阵$\boldsymbol{M}_A$与延时矩阵$\boldsymbol{M}_B$,并且将该延时矩阵发送到可信服务器S.

2)服务器S采用1.1节中阶段1来获得稳定的响应值.通过PUF的延时矩阵计算出所有路径之间的延迟差,并将差值按降序排列得到延时序列(T_A, T_B).通过阶段2与共享密钥R来获得PUF_A与PUF_B的激励(C_A, C_B),即$\{PUF(C_A) = PUF(C_B) = R, C_A \neq C_B\}$并存储于服务器中.

2.2　双向认证与固件更新阶段

双向认证与固件更新阶段是PUF产生的共享密钥在不安全的信道中完成设备之间的安全认证,执行算法2,具体步骤如图4所示.设备与服务器认证阶段如图5所示.

1)设备A与B记录当时时刻的时间戳$(T1, T2)$,分别作为随机数发生器的种子产生两对随机值(n_{A1}, n_{A2})与(n_{B1}, n_{B2}),并且发送到服务器S.

2)当服务器S接收到设备的随机值,通过MASK算法保护PUF所产生的激励值C_A与C_B,即$D_A = MASK(C_A, PRNG(n_{A1}))$与$D_B = MASK(C_B, PRNG(n_{B1}))$.然后将$D_A, n_{B2}$和$D_B, n_{A2}$分别发送到设备$A$与设备$B$.

3)设备A通过随机数n_{A1}解析激励C_A,即$C_A \leftarrow UNMASK(D_A, PRNG(n_{A1}))$,再通过PUF计算出

响应值R_A.同样,设备B通过随机数n_{B1}解析激励C_B并计算出响应值R_B.

算法2:双向认证与固件更新阶段

1:(T_1,T_2)←Device. timestape()
2:(n_{A1},n_{A2}),(n_{B1},n_{B2})←Device. $TRNG_A(T_1)$,Device. $TRNG_B(T_2)$
3:Server←Device. (n_{A1},n_{A2}),Device. (n_{B1},n_{B2})
4:(K_A,K_B)←Server. $PRNG(n_{A1})$,Server. $PRNG(n_{B1})$
5:(D_A,D_B)←Server. $MASK(C_A,K_A)$,Server. $MASK(C_B,K_B)$
6:Update$(R^{new},C_A^{new},C_B^{new})$
7:Device. A,Device. B←Server. (D_A,n_{B2}),Server. (D_B,n_{A2})
8:(C_A,C_B)←Device. $UNMASK_A(D_A,PRNG(n_{B2}))$, Device. $UNMASK_B(D_B,PRNG(n_{A2}))$
9:(R_A,R_B)←Device. $PUF_A(C_A)$,Device. $PUF_B(C_B)$
10:(r_{A1},r_{A2}),(r_{B1},r_{B2})←Device. Fisher-shuffler(R_A,R_B)
11:(d_A,d_B)←Device. $Hash_A(r_{A1},PRNG(n_{B2}))$,
Device. $Hash_B(r_{B1},PRNG(n_{A2}))$
12:Device. A←d_B,Device. B←d_A
13:r_{B2}←Device. $Hash_A(r_{A2},PRNG(n_{A2}))$,
r_{A1}←Device. $Hash_B(r_{B1},PRNG(n_{B2}))$
14:if (d_B,r_{B2})<=Device. ε_A,if (d_A,r_{A1})<=Device. ε_B
15:else abort
16:Update(T_1^{new},T_2^{new})←Device.timestape()
17:$(n_{A1}^{new},n_{A2}^{new})$,$(n_{B1}^{new},n_{B2}^{new})$←Device. $TRNG_A(T_1^{new})$,Device. $TRNG_B(T_2^{new})$
18:Store$(n_{A1}^{old},n_{A2}^{old})$,$(n_{B1}^{old},n_{B2}^{old})$

图4 双向认证和固件更新阶段算法步骤图

Fig.4 Algorithm steps in mutual authentication and firmware update phase

4)通过Fisher-Shuffler洗牌混淆算法分别将响应R_A与R_B分解为(r_{A1},r_{A2})与(r_{B1},r_{B2}),该算法高效且等概率生成随机序列.设备A通过随机数n_{B2}计算PUF响应值r_{A1}的Hash值,即d_A←Hash$(r_{A1},PRNG(n_{B2}))$;同样,设备B计算出Hash值.

5)当设备A接收到数据d_B之后,计算$\tilde{r}_{B2}$←Hash$(r_{A2},PRNG(n_{A2}))$;如果r_{A2}与$\tilde{r}_{B2}$在阈值之内,即if$(r_{A2},\tilde{r}_{B2})\leqslant\varepsilon$,则设备$A$认证设备$B$成功.同样当设备$A$接收到数据$d_A$之后,if$(\tilde{r}_{A1},r_{B1})\leqslant\varepsilon$,则设备$B$认证设备$A$成功.

6)当服务器S完成对设备激励C_A与C_B的发送,则更新密钥R^{new}和激励对(C_A^{new},C_B^{new}).

7)设备记录时间戳来更新随机数,从而设备A与设备B分别产生新的随机对$(n_{A1}^{new},n_{A2}^{new})$与$(n_{B1}^{new},n_{B2}^{new})$,并且存储旧的随机值$(n_{A1}^{old},n_{A2}^{old})$与$(n_{B1}^{old},n_{B2}^{old})$.

3 形式化安全证明

3.1 形式化安全分析

该协议保证物联网设备的信道传输安全,也可以防御攻击者对PUF进行物理攻击,具体安全分析如下.

1)建模攻击.机器学习建模攻击是针对具有可公开访问的CRP接口的强PUF,攻击者通过收集大量CRP,训练、学习和优化一个精确的模型,从而在给定的激励中预测出响应.但是,本文采用的可重构

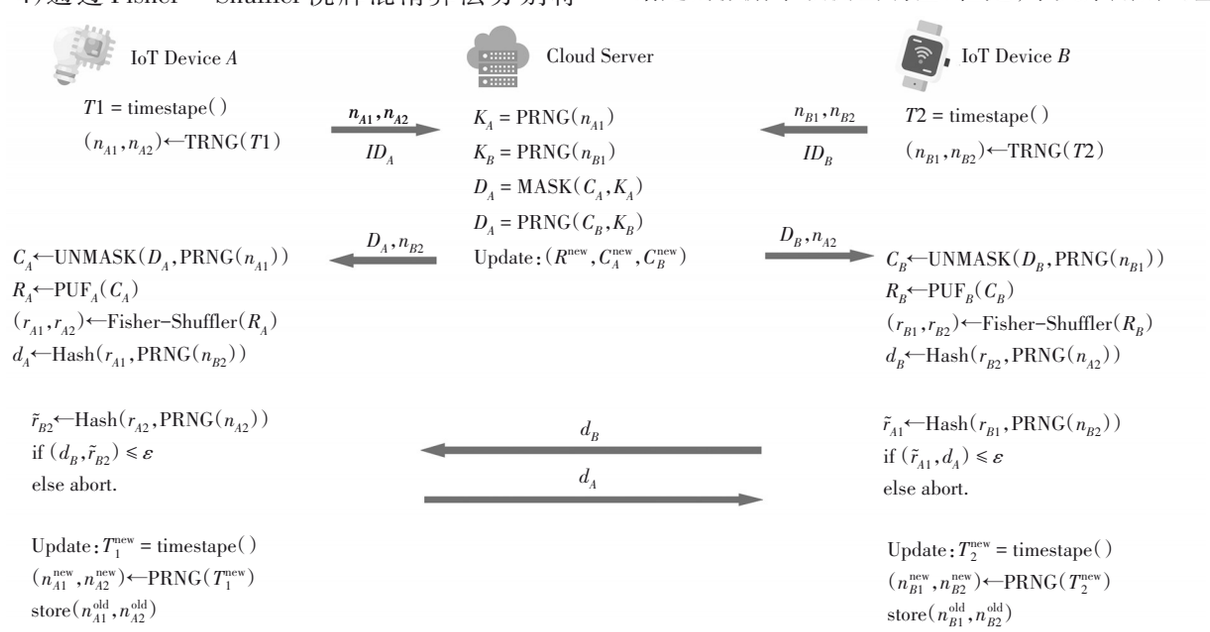

图5 双向认证与固件更新阶段

Fig.5 Mutual authentication and firmware update phase

CRO PUF是用作密钥生成的弱PUF,没有访问接口来读取芯片内部生成的密钥,密钥不会暴露给攻击者.同时,由于协议机制保护,通过MASK算法保护(C_A, C_B)值,以及Fisher－Shuffler混淆算法将响应值分为两部分(r_{A1}, r_{A2})与(r_{B1}, r_{B2}),并采用Hash算法与随机数发生器保护设备中部分的响应值r_{A1}.由于Hash函数单向性的特点,攻击者通过窃听内容d_A获取不到真实的CRP值,因此,攻击者很难对PUF进行机器学习建模攻击.

2)不可追溯性.在物联网设备身份认证过程中,如果攻击者无法有效关联两次认证的请求和答复信息,即输入与输出结果不能相映射,则认为设备是无法追踪的.攻击者通过窃听获取消息D_A与D_B时,因为激励C_A与C_B通过MASK函数进行加密防护,攻击者不能推断出激励C_A与C_B的值.攻击者窃听消息d_A与d_B之后,即$d_A \leftarrow \mathrm{Hash}_A(r_{A1}, \mathrm{PRNG}(n_{B2}))$,$d_B \leftarrow \mathrm{Hash}_B(r_{B2}, \mathrm{PRNG}(n_{A2}))$.由于哈希函数单一性,其不能获取共享密钥$R_A$与$R_B$的值.因此该协议可以防止位置跟踪.

3)去同步攻击.在协议的更新阶段,设备端产生新的随机数$(n_{A1}^{\mathrm{new}}, n_{A2}^{\mathrm{new}})$与$(n_{B1}^{\mathrm{new}}, n_{B2}^{\mathrm{new}})$,同时也存储上一轮认证的随机数$(n_{A1}^{\mathrm{old}}, n_{A2}^{\mathrm{old}})$与$(n_{B1}^{\mathrm{old}}, n_{B2}^{\mathrm{old}})$.当设备$B$被去同步攻击时,设备$A$的随机数会正常更新,但是设备$B$的随机数不会进行更新.当进行下一轮认证,服务器会返回值$D_A, n_{B2}^{\mathrm{old}}$给设备$A$,同时返回值$D_B, n_{A2}^{\mathrm{new}}$给设备$B$.由于设备$A$所返回的验证值$d_A^{\mathrm{old}}$与上一轮认证一致,而设备$B$返回的$d_B^{\mathrm{new}}$与上一轮不同.因此可以检测出设备之间未同步.

4)重放攻击.该协议机制采用时间戳和更新的随机数(n_{A1}, n_{A2})与(n_{B1}, n_{B2})来防御重放攻击.以设备A认证为例,假定第i次和i+1次的随机数分别为(n_{A1}^i, n_{A2}^i)与$(n_{B1}^{i+1}, n_{B2}^{i+1})$,当攻击者获得第$i$次的会话消息$D_A^i, d_B^i$之后,进行第$i$+1认证时,设备$A$收到的会话消息已经更新为$D_A^{i+1}, d_B^{i+1}$.因此,攻击者认证将会失败,重放攻击将被检测.

5)假冒攻击.当攻击者伪装成合法设备时,需要发送有效消息d_A与d_B.以设备A为例,因为信息d_A的生成需要有效的r_{A1},通过MASK函数来保护PUF的输入激励C_A,同时通过混淆算法将响应R_A进行防护.由于Hash函数具有单一性,获得信息d_A仍然不能得到有效信息r_{A1}.因此在该协议机制中,攻击者不能假冒成合法设备与服务器认证.同理,当攻击者伪装成服务器时,也不能获取激励C_A与C_B的值,从而不能与设备相互认证.

6)中间人攻击.该协议可以对中间人攻击进行防御,攻击者窃听消息$D_A, n_{B2}, D_B, n_{A2}, d_A$与$d_B$是不能获取有效信息.因为所窃听的消息都是加密后的信息,攻击者若替换新的消息之后,设备之间将不能识别导致认证失败.若攻击者想解析加密后的信息,由上面假冒攻击可知,攻击者不能获取PUF的CRP对(C_A, R_A),因此设备之间认证会失败.

3.2　协议证明

本节使用BAN逻辑来证明PUF产生共享密钥的安全性.BAN逻辑是认证协议形式化分析方法,可以提示一些非形式化方法很难发现的缺陷.BAN逻辑中常用的逻辑符号如表2所示,与本节相关的BAN逻辑常用的逻辑规则如表3所示.

表2　BAN逻辑中常用的逻辑符号说明

Tab.2　Common logic symbols description in BAN logic

符号	含义	符号	含义
$P\mid\equiv X$	P相信X是真实的	$P\mid\Rightarrow X$	实体P对X有管辖权
$P\mid\sim X$	P曾经发送消息X	$\#(X)$	X是新鲜的
$P\lhd X$	实体P收到消息X	(X,Y)	X或Y是(X,Y)一部分
$\{X\}_K$	使用密钥K对消息X执行加密操作	$P\overset{K}{\longleftrightarrow}Q$	K是P和Q之间的共享密钥

表3　BAN逻辑常用的逻辑规则

Tab.3　Common logic rules of BAN logic

规则说明	逻辑表达式
规则1:消息含义规则	$\dfrac{P\mid\equiv P\overset{K}{\longleftrightarrow}Q, P\lhd\{X\}_K}{P\mid\equiv Q\mid\sim X}$
规则2:新鲜性规则	$\dfrac{P\mid\equiv\#(X)}{P\mid\equiv\#(X,Y)}$
规则3:临时值校验规则	$\dfrac{P\mid\equiv\#(X), P\mid\equiv Q\mid\sim X}{P\mid\equiv Q\mid\equiv X}$
规则4:信任规则	$\dfrac{P\mid\equiv Q\mid\equiv(X,Y)}{P\mid\equiv Q\mid\equiv X}$

为便于分析,将设备A与B表示为$D1$与$D2$,服务器表示为S,对协议进行理想化后,消息发送规则调整如下:

M1: $D1\rightarrow S: n_{A1}, n_{A2}$;　M2: $D2\rightarrow S: n_{B1}, n_{B2}$

M3: $S\rightarrow D1: \{c_A, n_{A1}\}_{n_{A1}}, n_{B2}$;　M4: $S\rightarrow D2: \{c_B, n_{B1}\}_{n_{B1}}, n_{A2}$

M5: $D1\rightarrow D2: \{r_{A1}, n_{B2}\}_{n_{B2}}$;　M6: $D2\rightarrow D1: \{r_{B1}, n_{A2}\}_{n_{A2}}$

协议初始化假设如下:

A1: $D1\mid\equiv\#\{n_{A1}, n_{A2}\}$　A2: $D2\mid\equiv\#\{n_{B1}, n_{B2}\}$

A3: $D1\mid\equiv D1\overset{n_{A1}}{\longleftrightarrow}S$;　$D1\mid\equiv D1\overset{n_{A2}}{\longleftrightarrow}S$

A4: $D2\mid\equiv D2\overset{n_{B1}}{\longleftrightarrow}S$;　$D2\mid\equiv D2\overset{n_{B2}}{\longleftrightarrow}S$

A5: $D1\mid\equiv D1\overset{n_{A2}}{\longleftrightarrow}D2$;　$D2\mid\equiv D1\overset{n_{B2}}{\longleftrightarrow}D2$

目标公式如下：

G1: $D1| \equiv S| \equiv \{c_A\}$;G3: $D2| \equiv D1| \equiv \{r_{A1}\}$;

G2: $D2| \equiv S| \equiv \{c_B\}$;G4: $D1| \equiv D2| \equiv \{r_{B1}\}$;

如果目标公式成立，则说明设备与服务器S都与对方协商并确认秘密密钥，并且秘密密钥与平台完整性报告及通信信道绑定. 可根据表3中的各规则进行如下逻辑推理.

由规则1与A3、M3可知：

$$\frac{D1| \equiv D1 \overset{n_{A1}}{\longleftrightarrow} S, D1 \lhd \{c_A, n_{A1}\}_{n_{A1}}}{D1| \equiv S| \sim \{c_A, n_{A1}\}} \tag{1}$$

由规则2与A1可知：

$$\frac{D1| \equiv \#(n_{A1})}{D1| \equiv \#(c_A, n_{A1})} \tag{2}$$

由规则3及式(1)与式(2)可知：

$$\frac{D1| \equiv \#(c_A, n_{A1}), D1| \equiv S| \sim \{c_A, n_{A1}\}}{D1| \equiv S| \equiv \{c_A, n_{A1}\}} \tag{3}$$

由规则4与式(3)可知：

$$\frac{D1| \equiv S| \equiv \{c_A, n_{A1}\}}{D1| \equiv S| \equiv \{c_A\}} \tag{4}$$

因此，可证得目标公式G1: $D1| \equiv S| \equiv \{c_A\}$，设备$A$与服务器$S$共享密钥$c_A$.

由规则1与A4、M4可知：

$$\frac{D2| \equiv D2 \overset{n_{B1}}{\longleftrightarrow} S, D2 \lhd \{c_B, n_{B1}\}_{n_{B1}}}{D2| \equiv S| \sim \{c_B, n_{B1}\}} \tag{5}$$

由规则2与A2可知：

$$\frac{D2| \equiv \#(n_{B1})}{D2| \equiv \#(c_B, n_{B1})} \tag{6}$$

同理，由规则3和4，及式(5)与式(6)可证明目标公式：G2: $D2| \equiv S| \equiv \{c_B\}$.

由规则1与A5、M5可知：

$$\frac{D2| \equiv D1 \overset{n_{B2}}{\longleftrightarrow} D2, D2 \lhd \{r_{A1}, n_{B2}\}_{n_{B2}}}{D2| \equiv D1| \sim \{r_{A1}, n_{B2}\}} \tag{7}$$

由规则2与消息A2可知：

$$\frac{D2| \equiv \#\{n_{B2}\}}{D2| \equiv D1| \equiv \{r_{A1}, n_{B2}\}} \tag{8}$$

由规则3与式(7)、式(8)可知：

$$\frac{D2| \equiv \#\{r_{A1}, n_{B2}\}, D2| \equiv D1| \sim \{r_{A1}, n_{B2}\}}{D2| \equiv D1| \equiv \{r_{A1}, n_{B2}\}} \tag{9}$$

由规则4与式(9)可知：

$$\frac{D2| \equiv D1| \equiv \{r_{A1}, n_{B2}\}}{D2| \equiv D1| \equiv \{r_{A1}\}} \tag{10}$$

因此，可证得目标公式G3: $D2| \equiv D1| \equiv \{r_{A1}\}$，设备$B$与设备$A$共享密钥$r_{A1}$.

由规则1与A5、M6可知：

$$\frac{D1| \equiv D1 \overset{n_{A2}}{\longleftrightarrow} D2, D1 \lhd \{r_{B1}, n_{A2}\}_{n_{A2}}}{D1| \equiv D2| \sim \{r_{B1}, n_{A2}\}} \tag{11}$$

由规则2与A1可知：

$$\frac{D1| \equiv \#\{n_{A2}\}}{D1| \equiv \#\{r_{B1}, n_{A2}\}} \tag{12}$$

同理，由规则3和4，及式(11)与式(12)可证明目标公式：G4: $D1| \equiv D2| \equiv \{r_{B1}\}$.

3.3 形式化工具分析

ProVerif[14]是应用于验证形式模型中密码协议的工具，支持对称和非对称加密、数字签名、Hash函数、Diffie-Hellman密钥协议等许多密码原语进行自动有效的安全分析. 在本节中，使用ProVerif来证明所提协议的保密性和身份验证属性.

协议定义服务器与设备之间的共享激励skc，以及设备A与设备B的共享密钥skr. 在Dolev-Yao模型[15]下，图6是协议对设备与服务器的认证所建模四个事件：UserStarted和UserAuthed，Server-Started和ServerAuthed，会话密钥skc以及skr的安全查询. ProVerif查询协议的结果如图7所示，通过结果可得该协议保持身份验证属性和长期秘密的保密性，会话密钥对模拟攻击者具有鲁棒性.

```
event UserStarted(bitstring).
event UserAuthed(bitstring).
event ServerStarted(bitstring).
event ServerAuthed(bitstring).
query id:bitstring;inj-event(UserAuthed(id))==>inj-event(UserAuthed(id))
query attacker(skc).
query attacker(skr).
```

图6 设备与服务器的认证事件以及密钥安全查询

Fig.6 Authentication events and key security query

```
Verification summary:
Query inj-event(UserAuthed(id)) ==> inj-event(UserStarted(id)) is true.
Query not attacker(skc[]) is true.
Query not attacker(skr[]) is true.
```

图7 协议查询结果

Fig.7 Query result in the proposed protocol

4 协议性能分析

本节从安全属性、存储容量及通信成本等方面

对认证协议进行分析与评估.通过Python编写设备和服务器之间的认证协议程序.通过抽象TCP客户端/服务器连接的套接字完成网络交互,使得服务器等待与指定IP地址上的设备连接.一旦设备成功与服务器建立连接,该协议就执行一次相互认证会话.服务器和设备运行在Windows 10,采用Intel Core i7-9750 CPU,频率为2.60 GHz,配备8 GB RAM,模拟所提出的身份验证方案.

在安全属性方面与其他协议进行对比分析,其结果如表4所示.协议[2,10-11,16-17]机制中,攻击者可以通过窃听、假冒以及物理攻击获取PUF的CRP对,从而不能防御建模攻击.协议[9]通过d次锁定机制可以防御PUF攻击,但信道传输中的信息未加密,导致设备认证中被窃听、去同步以及重放攻击等安全威胁.协议[2-3,9,16-17]中的设备存储了新旧两个身份,攻击者通过访问内存获取当前身份信息,从而跟踪到前一轮或下一轮的认证信息,因此协议不具备不可追溯性.但是本协议采用PUF生成的共享密钥,采用MASK算法与Hash函数的加密保证设备的隐私性与不可追溯性.

表4　协议安全属性分析与比较

Tab.4　Analysis and comparison of protocol security attributes

安全属性	Hossain[2]	Akgün[16]	Yu[9]	Huang[10]	Aysu[11]	Zhou[3]	Moriyama[17]	Bian[5]	本协议
双向认证	T	T	T	T	T	T	T	T	T
位置跟踪	F	T	F	F	F	F	F	T	T
可扩展性	F	F	T	F	F	T	F	F	T
前向不可追溯性	F	F	F	T	T	F	F	T	T
抗克隆攻击	T	T	T	T	T	F	T	T	T
抗建模攻击	F	F	T	F	F	无	F	T	T
抗假冒攻击	F	T	F	F	T	F	T	T	T
抗中间人攻击	F	T	F	F	T	F	T	T	T
抗去同步攻击	T	F	F	F	T	T	T	T	T

图8列出与其他协议在设备存储与通信开销方面的比较.参考Aysu[11]的论文,伪随机身份PID_D^i的字节长128 bit,CRP对(C_i,R_i)的字长均为128 bit,密钥的字节长度为96 bit.本协议只存储了128 bit随机数(n_1^{old},n_1^{old}),远远低于其他协议的存储容量.协议只传输信息$(n_{A1},n_{A2},D_A,n_{B2},d_A,d_B)$,通信开销为640 bit.相比较其他协议[3,5,9-11,16-17],所提出协议的通信成本均低于其他方案的通信成本,适用于轻量级设备的安全认证场景.

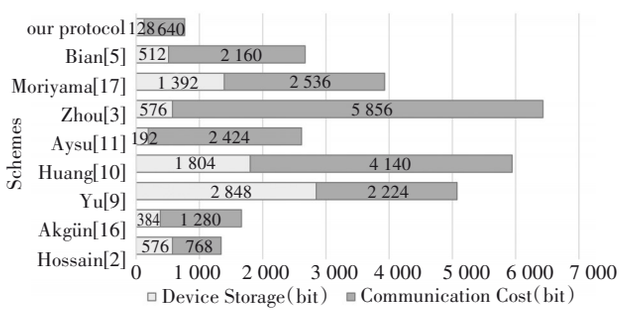

图8　协议中设备存储与通信开销对比图

Fig.8　Performance comparison of device storage and communication cost in the proposed protocol

5　结　论

由于物联网产生海量数据给资源受限的终端设备带来信息传输安全威胁,同时硬件设备通常面临着芯片克隆、设备伪造以及密钥存储的安全问题.因此,运行速率慢、通信开销大的传统网络协议不适合轻量级物联网设备的安全认证.本文面向物联网设备提出一种轻量级匿名密钥共享认证协议.该机制采用PUF防篡改防克隆的特性在硬件端生成共享密钥,结合混淆算法、MASK算法、Hash函数等安全原语来保证信息传输的匿名性、不可跟踪、不可否认和前向/后向保密等安全属性.通过形式化验证工具ProVerif、BAN逻辑以及非形式化的安全性分析验证,证明协议运行的安全性、可靠性以及抗信道攻击能力.相比较于其他现有协议,所提出的协议具备计算成本低、通信开销和存储容量小以及高安全性的特点,适合轻量级物联网设备的安全通信传输.

参考文献

[1] HASSIJA V, CHAMOLA V, SAXENA V, *et al*. A survey on IoT security: application areas, security threats, and solution architectures[J]. IEEE Access, 2019, 7: 82721-82743.

[2] HOSSAIN M, NOOR S, HASAN R. HSC-IoT: a hardware and software co-verification based authentication scheme for Internet of Things[C]//2017 5th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (Mobile Cloud). San Francisco, CA, USA: IEEE, 2017: 109-116.

[3] ZHOU L, LI X, YEH K H, *et al*. Lightweight IoT-based authentication scheme in cloud computing circumstance[J]. Future Generation Computer Systems, 2019, 91: 244-251.

[4] 王鑫,贾庆轩,高欣,等. 可证明安全的轻量级无服务型RFID安全搜索协议[J]. 湖南大学学报(自然科学版),2014,41(8):117-124.
WANG X, JIA Q X, GAO X, *et al*. Provable security lightweight service-less RFID security search protocol[J]. Journal of Hunan University (Natural Sciences), 2014, 41(8): 117-124. (In Chinese)

[5] BIAN W X, GOPE P, CHENG Y Q, *et al*. Bio-AKA: an efficient fingerprint based two factor user authentication and key agreement scheme[J]. Future Generation Computer Systems, 2020, 109: 45-55.

[6] 胡锦,谢立红,邹望辉,等. 基于低功耗SoC的微型图像采集系统设计[J]. 湖南大学学报(自然科学版),2019,46(2):86-91.
HU J, XIE L H, ZOU W H, *et al*. Design of miniature image acquisition system based on low power system on chip[J]. Journal of Hunan University (Natural Sciences), 2019, 46(2): 86-91. (In Chinese)

[7] ZEITOUNI S, OREN Y, WACHSMANN C, *et al*. Remanence decay side-channel: the PUF case[J]. IEEE Transactions on Information Forensics and Security, 2016, 11(6): 1106-1116.

[8] CHANG C H, ZHENG Y, ZHANG L. A retrospective and a look forward: fifteen years of physical unclonable function advancement[J]. IEEE Circuits and Systems Magazine, 2017, 17(3): 32-62.

[9] YU M D, HILLER M, DELVAUX J, *et al*. A lockdown technique to prevent machine learning on PUFs for lightweight authentication[J]. IEEE Transactions on Multi-Scale Computing Systems, 2016, 2(3): 146-159.

[10] HUANG Z, WANG Q. A PUF-based unified identity verification framework for secure IoT hardware via device authentication[J]. World Wide Web, 2020, 23(2): 1057-1088.

[11] AYSU A, GULCAN E, MORIYAMA D, *et al*. End-to-end design of a PUF-based privacy preserving authentication protocol[C]// Cryptographic Hardware and Embedded Systems, CHES 2015, 2015: 556-576.

[12] ZHANG J L, QU G. Physical unclonable function-based key sharing via machine learning for IoT security[J]. IEEE Transactions on Industrial Electronics, 2020, 67(8): 7025-7033.

[13] QURESHI M A, MUNIR A. PUF-RAKE: a PUF-based robust and lightweight authentication and key establishment protocol[J]. IEEE Transactions on Dependable and Secure Computing, 2021, 19(4): 2457-2475.

[14] XIE Q, HU B, TAN X, *et al*. Robust anonymous two-factor authentication scheme for roaming service in global mobility network[J]. Wireless Personal Communications, 2014, 74(2): 601-614.

[15] DOLEV D, YAO A. On the security of public key protocols[J]. IEEE Transactions on Information Theory, 1983, 29(2): 198-208.

[16] AKGÜN M, ÇA LAYAN M U. Providing destructive privacy and scalability in RFID systems using PUFs[J]. Ad Hoc Networks, 2015, 32: 32-42.

[17] MORIYAMA D, MATSUO S, OHKUBO M. Relation among the security models for RFID authentication protocol[C]//ECRYPT Workshop on Lightweight Cryptography. Berlin: Computer Science, 2013: 333-349.