可证明安全的高效车联网认证密钥协商协议

乔子芮[1),2)] 杨启良[4)] 周彦伟[1),2),3),5)] 杨 波[1)] 顾纯祥[5)] 张明武[2),3)] 夏 喆[6)]

[1)](陕西师范大学计算机科学学院 西安 710062)

[2)](密码科学技术全国重点实验室 北京 100878)

[3)](桂林电子科技大学广西密码学与信息安全重点实验室 广西 桂林 541004)

[4)](上海计算机软件技术开发中心 上海 201112)

[5)](河南省网络密码技术重点实验室 郑州 450052)

[6)](武汉理工大学计算机科学与技术学院 武汉 430070)

摘 要 为进一步解决当前车联网认证密钥协商协议效率低下及车辆公私钥频繁更新的问题,研究者利用无证书密码体制能够解决身份基密码体制中密钥托管不足和传统公钥基础设施中证书复杂管理问题的优势,提出了安全高效的无证书车联网认证密钥协商协议.然而,本文分析发现现有的协议要么不具有轻量级的特征,要么无法满足其所声称的安全性.针对上述问题,本文提出安全性可证明的高效车联网认证密钥协商协议的新型构造,并基于判定性 Diffie-Hellman 和离散对数等复杂性假设对协商密钥的安全性和通信消息的不可伪造性进行了形式化证明.与现有的相关协议相比,本文协议不仅效率更高且安全性更优,同时具备证书和密钥的集中管理、双向认证等属性,上述优势使得该协议在车联网中具有更好的性能和适应性,因此我们的协议更适合在该网络中使用.

关键词 无证书公钥密码机制;认证密钥协商;车辆自组织网;分叉引理

中图法分类号 TP393 **DOI 号** 10.11897/SP.J.1016.2023.00929

An Efficient Authentication Key Agreement Protocol with Provable Security for VANET

QIAO Zi-Rui[1),2)] YANG Qi-Liang[4)] ZHOU Yan-Wei[1),2),3),5)] YANG Bo[1)] GU Chun-Xiang[5)]

ZHANG Ming-Wu[2),3)] XIA Zhe[6)]

[1)](*School of Computer Science, Shaanxi Normal University, Xi'an* 710062)

[2)](*State Key Laboratory of Cryptology, Beijing* 100878)

[3)](*Guangxi Key Laboratory of Cryptography and Information Security, Guilin University of Electronic Technology, Guilin, Guangxi* 541004)

[4)](*Shanghai Computer Software Technology Development Center, Shanghai* 201112)

[5)](*Henan Key Laboratory of Network Cryptography Technology, Zhengzhou* 450052)

[6)](*School of Computer Science and Technology, Wuhan University of Technology, Wuhan* 430070)

Abstract To further solve low efficiency and frequent updates of the public keys and the private keys of vehicles in the authentication key agreement (AKA) protocol for vehicular Ad-Hoc network (VANET), some AKA protocols were proposed by researcher based on certificateless aggregate signature scheme, because the certificateless cryptography primitives can resolve the key escrow shorting of the identity-based cryptography and the certificate management problem of the public key infrastructure. However, we find that the previous AKA protocols either do not have high

收稿日期:2021-10-25;在线发布日期:2022-10-08.本课题得到国家重点研发计划(2017YFB0802000)、国家自然科学基金(62272287,61802242,U2001205)、广西密码学与信息安全重点实验室研究课题(GCIS202108)、河南省网络密码技术重点实验室研究课题(LNCT2021-A04)与中央高校基本科研业务费专项资金资助.**乔子芮**,博士生,主要研究方向为信息安全等.E-mail:qzr_snnu@163.com.**杨启良**,博士,主要研究方向为信息安全、密码学等.**周彦伟**(通信作者),博士,副教授,硕士生导师,主要研究兴趣为抗泄露密码学、匿名通信技术等.E-mail:zyw_snnu@foxmail.com.**杨 波**(通信作者),博士,教授,博士生导师,主要研究领域为信息安全、密码学等.E-mail:byang@snnu.edu.cn.**顾纯祥**,博士,教授,博士生导师,主要研究领域为信息安全等.**张明武**,博士,教授,博士生导师,主要研究领域为信息安全等.**夏 喆**,博士,副教授,硕士生导师,主要研究方向为信息安全等.

computational efficiency or cannot keep their claimed security. Hence, to further solve the above problems, an efficient AKA protocol with provable security is designed in this paper, in which, the security of the session key and the unforgeability of the communication messages are proved based on the hardness of the classic complexity assumptions. Such as decisional Diffie-Hellman, discrete logarithm, etc. Compared with the previous protocols, our proposal has higher computational efficiency and better security, In VANET, these advantages, such as centralized management of certificates and mutual authentication, make our protocol has better performance and adaptability.

Keywords certificateless public-key cryptography; authentication key agreement; VANET; forking lemma

1 引 言

车辆自组织网(Vehicular Ad-Hoc Network, VANET)是为实现智能化交通而部署的物联网,可将其简称车联网,它将为车辆提供实时的交通信息、车辆导航等功能. 如图 1 所示,实际环境中车联网与云计算环境结合部署,构建云车协同的车联网,主要包含下述实体:(1)第三方机构(Third Party, TP)建立相应的系统,完成相应的系统初始化,并负责车辆的注册等功能;(2)路边单元(Road Side Unit, RSU)是部署在道路周边,负责提供基础通信业务的设备;同时与车载单元之间进行无线通信;(3)车载单元(On Board Unit, OBU)是装备在车上与 RSU 进行通讯的装置,具有计算能力较弱、存储空间较少等特点;(4)云服务器(Cloud Server, CS)为用户提供相应的网络服务功能,完成相关服务的大量计算任务.

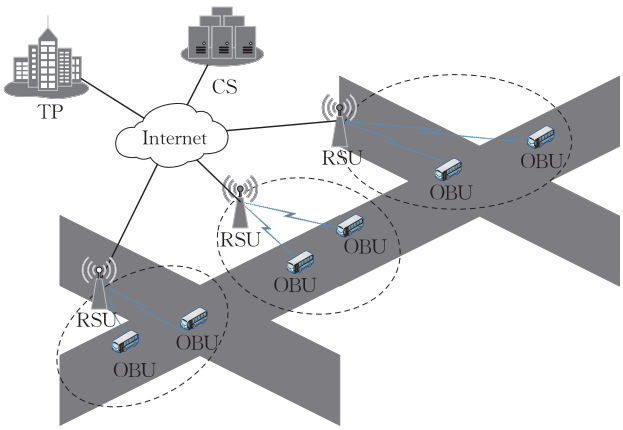


图 1 车联网系统模型

如图 1 所示,为了实现车联网的安全通信目标,身份认证和密钥协商是车联网中不可或缺的安全技术,其中身份认证能够确保车联网用户身份的合法性,使得合法车辆可以访问网络并获得相应的服务计算,而其他任何人都无法伪造身份认证信息,有效避免了恶意用户的加入;密钥协商为车联网中各实体间建立共享的会话密钥,完成安全通信信道的建立,并且会话密钥的使用在一定程度上提升了车联网的数据通信效率(通常会话密钥配合计算效率更高的对称加密解密机制实现数据通信). 由于认证密钥协商协议同时具备身份认证和密钥协商两种安全属性,是车联网推广部署的关键技术之一. 特别地,为实现对恶意车辆的控制,一般情况下车联网的第三方机构中包含一个注册追责中心(Tracing Registration Authority, TRA)负责对 OBU 的注册,并为其颁发授权信息,同时实现可疑车辆的事后追查,并实施相应的控制策略.

鉴于认证密钥协商协议是车联网安全工作的基础,因此众多研究者分别对此开展了相关研究. Al-Riyami 和 Paterson[1]所提出的无证书公钥密码体制(CL-PKC)无需使用完全可信的第三方构建系统环境(在现实环境中很难构建完全可信的第三方),因此众多学者基于该密码机制对认证密钥协商协议展开了相应的研究. 文献[2]设计了适用于无线体域网的认证密钥协商机制,该机制基于环签名技术在实现节点认证的同时,降低了协议的计算负载;此外,环签名技术的使用确保了签名节点的匿名性. 为满足 VANET 的隐私保护需求,文献[3]提出了一个高效的无证书环签名机制,并基于该机制实现了 VANET 通信过程的隐私保护认证. 文献[4]所设计的认证密钥协商协议着重考虑了安全性的增强和计算与通信成本的降低. 为了支持人工智能环境中支持批量认证需求,文献[5]基于区块链技术设计了一个高效的认证协议. 文献[6]提出了一种具有强隐私保护的伪名认证方案,该方案通过更新生成的伪身份证书产生一个分布式证书服务系统. 相对于

VANET 中的移动智能终端而言，双线性映射的运算量较大，因此上述方案[2-6]中双线性映射的使用导致相应协议的计算效率较低，并不适合在计算资源受限的车联网设备上使用.

双线性映射的使用在一定程度上增加了协议的计算负载，车联网中车载单元的低功耗特性使得上述构造并不能很好的在车联网中部署，因此，对于计算性能要求更高的车联网而言，需轻量级的认证密钥协商协议完成共享密钥的建立. 针对车联网的上述实际需求，研究者在不使用双线性映射的前提下展开了对认证密钥协商协议的研究. 文献[7]基于计算性 Diffie-Hellman 问题的困难性，提出了一种安全性增强的两方无证书密钥协商协议，并在扩展的 Canetti-Krawczyk (Extended Canetti-Krawczyk, eCK) 安全模型中对协议的安全性和不可伪造性进行了证明. 然而，文献[8]指出对于任意的恶意用户，上述协议[7]无法满足其所声称的安全性. 文献[8]提出一个新的无证书双方认证密钥协商协议，但该协议缺少密钥协商双方的身份认证过程，易遭受中间人攻击. 文献[9]利用 CL-PKC 提出了一个新型的协议，实现了双向的身份合法性认证和安全的密钥协商，由于移动终端与服务器间需进行 3 次的消息交互，导致该协议的通信时延较大. 智能医疗系统已成为为用户提供低成本、长期持续健康监测的最有效、最实用的解决方案，针对该系统的安全性保护需求，文献[10]提出了一种基于雾计算的智能医疗系统认证和密钥协商方案，并声称该方案是安全的，能够抵御各种已知的攻击；然而，文献[11]发现上述协议包含临时信息攻击和缺乏相应的验证等缺陷，并且敌手可以轻松恢复会话密钥和用户身份. 针对上述问题，文献[11]提出了改进的安全认证和密钥协商协议，弥补了文献[10]中协议的不完善之处. 面向智能城市中车联网的安全性保护需求，文献[12]设计了一种安全消息认证协议，并声称他们的方案能够防御潜在的攻击，并确保具有安全的身份验证和匿名性；然而，文献[13]发现该方案[12]存在密钥泄露、不提供相互认证等问题. 为了解决上述方案所面临的安全威胁，文献[13]设计了一种安全、高效的智慧城市环境下的车联网消息认证协议，遗憾的是文献[14]发现文献[13]的方案还是无法满足认证协议该有的安全性. 为提升密钥协商处理的效率，文献[15]提出了一种轻量级批量认证和密钥协商方案，该协议采用轻量级批量认证和一对一的密钥协商，其中通过相互认证实现通信双方的可信性验证；然而该方案中通信方都可以直接基于各自的秘密值计算出系统的主密钥. 为克服上述缺陷，文献[16]对其进行了改进，并对改进协议的安全性进行了简要分析. 针对文献[17]中协议不具备完美匿名性的现状，文献[18]提出了一个匿名、安全的双因素身份验证和密钥协商协议. 虽然上述方案[7-18]期望实现安全的认证密钥协商，但依然存在一定的不足，未能实现完美的安全目标.

为解决当前车联网认证密钥协商协议效率低下及车辆公私钥频繁更新的问题，基于无证书聚合签名机制文献[19]提出一个适用于车联网的认证密钥协商协议，该协议基于临时身份和预签名的方式实现了车辆隐私信息的保护和匿名认证，并且实现了可疑车辆的匿名追踪功能；此外，服务请求消息的聚合操作及不使用双线性映射运算的协议构造技术在一定程度上提升了该协议的执行效率. 然而，本文分析发现上述协议无法抵抗恶意用户的攻击，导致上述协议无法满足其所声称的安全性. 此外，文献[20]和[21]均对 CL-PKC 下的密钥协商进行了研究，分别提出了相应的密钥协商协议.

综上所述，现有的认证密钥协商协议[2-19]要么由于基于双线性映射构造导致协议[2-6,9,11]的计算或通信效率较低；要么协议[7-8,10,12-13,15,17,19]无法满足其所声称的安全性. 针对上述问题，本文基于 CL-PKC 设计了一个安全性增强的认证密钥协商协议，并对其协商密钥的安全性和通信消息的不可伪造性进行形式化证明. 此外，性能分析表明，本文构造在提升安全性的同时，具有较高的计算效率和通信效率；同时具备证书和密钥的集中管理、双向认证等属性，上述特性使得该协议在车联网中具有更好的性能和适应性. 特别地，本文在附录 A 将对文献[19]中协议的安全性进行详细分析，并剖析该协议的设计缺陷，防止后续相关研究中类似问题的再次发生.

2　基础知识

本文所涉及的主要符号及含义如表 1 所示. 特别地，在本文构造中，r_{Oi}，r_T 表示由 OBU_i 和 TRA 分别选取的进行密钥生成运算的随机数，r_{OiT} 表示由 OBU_i 选取的与 TRA 进行密钥协商运算的随机数，r_{TOi} 表示由 TRA 选取的与 OBU_i 进行密钥协商运算的随机数. 其他参数的下标含义与上述参数相类似，此处不再赘述.

表 1 符号介绍

符号	含义
RSU	路边单元
CS	云服务器
(pk_U, sk_U)	用户 U 的公私钥对
$H_i(i=1,\cdots,5)$	安全的单向哈希函数
$(\mathcal{E},\mathcal{D})$	对称加密/解密算法
P_{Pub}	系统主公钥
$Params$	系统公开参数
msk	系统主私钥
OBU	车载单元
KGC	密钥生成中心
id_U	车辆 U 的身份
tid_{Oi}	车载单元 OBU_i 的临时身份
$a \leftarrow_R A$	从集合 A 中均匀随机的选取 a
n	相应区域内 OBU 的最大数量

2.1 困难性假设

定义 1. 离散对数(Discrete Logarithm,DL)问题. 令 G 是一个阶为素数 q 的加法循环群,P 是群 G 的一个生成元,已知 $aP\in G$ 且 $a\in Z_q^*$,DL 问题的目标是求解 a 的值. 在概率多项式时间内,算法 $\mathcal{A}$ 成功解决 DL 问题的优势定义为

$$Adv_{\mathcal{A}}^{\mathrm{DL}}(\kappa)=\Pr(\mathcal{A}(aP,P)=a),$$

其中概率来源于 a 在 Z_q^* 上的随机选取及算法 $\mathcal{A}$ 的随机选择.

DL 假设:对于任何多项式时间算法 $\mathcal{A}$ 成功解决 DL 问题的优势 $Adv_{\mathcal{A}}^{\mathrm{DL}}(\kappa)$ 都是可忽略的.

定义 2. 判定性 Diffie-Hellman(Decisional Diffie-Hellman,DDH)问题. 对于任意且未知的 $x,y,z\in Z_q$,已知 $P,xP,yP\in G$,DDH 问题的目标是区分 xyP 和 zP. 在概率多项式时间内,算法 $\mathcal{A}$ 成功解决 DDH 问题的优势定义为

$$Adv_{\mathcal{A}}^{\mathrm{DDH}}(\kappa)=$$
$$\left|\Pr(\mathcal{A}(P,xP,yP,xyP))-\Pr(\mathcal{A}(P,xP,yP,zP))\right|,$$

其中概率来源于 x,y,z 在 Z_q 上的随机选取及算法 $\mathcal{A}$ 的随机选择.

DDH 假设:对任意的多项式时间算法 $\mathcal{A}$,其成功解决群 G 上 DDH 问题的优势 $Adv_{\mathcal{A}}^{\mathrm{DDH}}(\kappa)$ 是可忽略的.

2.2 分叉引理

文献[22-23]和[24]对分叉引理进行了详细介绍,指出若敌手 $\mathcal{A}$ 能以优势 $\varepsilon(\kappa)$ 伪造一个有效签名,且对随机谕言机 H 进行 q_H 次询问,那么通过对谕言机 H 的重放操作,$\mathcal{A}$ 基于相同的随机数和不同的谕言机应答输出两个有效签名的概率至少为 $\left(1-\frac{1}{\mathrm{e}}\right)\frac{1}{q_H}\varepsilon(\kappa)$,其中 e 是自然对数底数[22-23]. 关于分叉引理的详细介绍请参阅文献[22]、[23]和文献[24]的相关内容,本文不再赘述.

2.3 安全模型

2.3.1 敌手分类

在 CL-PKC 中,KGC 仅负责生成用户的部分密钥,并在此基础上用户自己生成完整的密钥. 对于 KGC 而言,它并不掌握用户的完整密钥,那么无需假设 KGC 是完全可信的第三方,因此在实际应用中,CL-PKC 将面临恶意的用户和 KGC 两类攻击者,其中恶意的用户利用替换公钥的方式将其伪装成其他合法用户实施攻击,而恶意的 KGC 利用其拥有的主密钥进行攻击. 下面详细介绍 $\mathcal{A}^1$ 和 $\mathcal{A}^2$ 两类敌手的能力描述[1,10].

(1) $\mathcal{A}^1$ 能够用已掌握的信息替换任意合法用户的公钥,使系统内的其他用户均认为该用户的公钥就是 $\mathcal{A}^1$ 所替换的值,但 $\mathcal{A}^1$ 无法获悉该用户原始公钥所对应的秘密值和系统的主私钥;此外,$\mathcal{A}^1$ 不能获得挑战身份的私钥和部分私钥,并且不能在挑战阶段之前替换挑战身份的公钥. 特别地,在公钥替换攻击中,除 $\mathcal{A}^1$ 外任何参与者都无法获知替换公钥所对应的秘密值.

(2) $\mathcal{A}^2$ 已掌握了系统的主密钥,但不具有替换任意用户公钥的能力;此外,$\mathcal{A}^2$ 不能获得挑战身份的私钥. 特别地,$\mathcal{A}^2$ 不能重新生成已有的公开参数,即系统的已有公开参数不能被替换.

车联网的认证密钥协商协议具体包括系统建立,注册与授权颁发和认证与密钥协商三个阶段. 其中,在初始化阶段,TP 主要为 RSU 和 OBU 等设备生成相应的密钥;OBU 在注册与授权颁发阶段完成向 TP 的注册,并获得 TP 为其颁发的注册凭证;OBU 基于 RSU 进行消息转发的基础上,与 CS 在认证与密钥协商完成相互的身份认证,并完成后续通信所需的会话密钥. 各阶段的形式化定义详见文献[20-21,25]和文献[26].

2.3.2 *eCK* 安全模型

本文将基于 *eCK* 安全模型[8-9]对认证密钥协商协议的安全性进行形式化证明. 在 *eCK* 安全模型中,敌手具备执行 $Send$,$Reveal$ 和 $Corrupt$ 等询问的能力;此外,敌手通过 $Test$ 询问获得相应的挑战会话密钥,最后敌手输出对挑战会话密钥的猜测值. 特别地,敌手在下述游戏中,将自适应无序地执行谕言机询问. 令 $\Pi_{i,j}^t$ 和 $\Pi_{j,i}^t$ 表示第 t 次密钥协商协议运行时的参与者,其中下标 i、j 表示参与者的索引号.

该游戏包括模拟者 $\mathcal{S}$ 和敌手 $\mathcal{A}$ 两个参与者,具体

分为询问和测试两个阶段.

(1) 询问阶段. 该阶段，$\mathcal{A}$将自适应地进行下述询问：

$Send(\Pi_{i,j}^{t},x)$：$\mathcal{A}$通过发送 $Send$ 询问开始一个新的会话或者完成已有会话的消息发送. 具体地讲，若存在相应的参与者 $\Pi_{j,i}^{t}$，则 $\Pi_{j,i}^{t}$收到消息 x 后继续执行密钥协商协议并输出相应的应答消息 m，或者输出特殊的符号表示拒绝或接收该会话；若不存在相应的参与者 $\Pi_{j,i}^{t}$，基于消息 x 建立该参与者，特别地，若 $x=\kappa$，那么参与者 $\Pi_{i,j}^{t}$是相应密钥协商会话的发起者；否则，$\Pi_{i,j}^{t}$是相应密钥协商会话的响应者，并且消息 x 是 $\Pi_{i,j}^{t}$的输入. 在实际中任何参与者都不能跟自己进行会话密钥的协商，因此在该询问中要求 $i\neq j$.

$Reveal(\Pi_{i,j}^{t})$：$\mathcal{A}$通过发送 $Reveal$ 询问请求获得指定参与者 $\Pi_{i,j}^{t}$所协商的会话密钥. 若 $\Pi_{i,j}^{t}$接受该询问，则返回相应的会话密钥；否则，$\Pi_{i,j}^{t}$输出特殊的符号 $\perp$，并终止该询问.

$Corrupt(i)$：$\mathcal{A}$通过发送 $Corrupt$ 询问请求获得索引号为 i 的用户私钥.

(2) 测试阶段. 询问阶段的结束时间由 $\mathcal{A}$决定，在该阶段$\mathcal{A}$选取一个新鲜的参与者 $\Pi_{i,j}^{\eta}$并执行 $Test(\Pi_{i,j}^{\eta})$询问获得该询问的应答消息. 需要强调的是 $Test(\Pi_{i,j}^{\eta})$仅被执行一次. 特别地，当 $\Pi_{i,j}^{\eta}$满足下述条件时称 $\Pi_{i,j}^{\eta}$是新鲜的参与者：① $\Pi_{i,j}^{\eta}$接受了相应的会话；② 未执行 $Reveal(\Pi_{i,j}^{\eta})$询问；③ 对于 $\Pi_{i,j}^{\eta}$有 $i\neq j$，并且未进行 $Corrupt(j)$询问；④ 对与 $\Pi_{i,j}^{\eta}$相匹配的参与者 $\Pi_{j,i}^{\eta}$，未执行 $Reveal(\Pi_{j,i}^{\eta})$询问.

$Test(\Pi_{i,j}^{\eta})$：如果 $\Pi_{i,j}^{\eta}$是新鲜的，则$\mathcal{S}$选取随机数 $\gamma\leftarrow_{R}\{0,1\}$，若 $\gamma=0$，$\mathcal{S}$发送会话密钥给敌手$\mathcal{A}$；否则，$\mathcal{S}$发送随机的会话密钥给敌手$\mathcal{A}$.

敌手$\mathcal{A}$收到 $Test(\Pi_{i,j}^{\eta})$询问的应答之后，依然可以继续 $Send$，$Reveal$ 和 $Corrupt$ 等询问，但是不能对 $\Pi_{i,j}^{\eta}$或与 $\Pi_{i,j}^{\eta}$相匹配的参与者 $\Pi_{j,i}^{\eta}$进行 $Reveal$ 和 $Corrupt(j)$询问. 最终，$\mathcal{A}$输出 γ'. $\mathcal{A}$获得胜利的优势是 $Adv_{\mathcal{A}}(\kappa)=\left|\Pr[\gamma=\gamma']-\frac{1}{2}\right|$.

定义 3.　协商密钥的安全性. 一个安全的认证密钥协商协议同时具备下述两个条件.

(1) 对于诚实的敌手$\mathcal{A}$，密钥协商协议完成后参与双方获得了相同的协商密钥，并且该密钥对于任意敌手而言是无法伪造的；

(2) 对于任意的概率多项式时间敌手$\mathcal{A}$，协商密钥是均匀随机的，即$\mathcal{A}$在上述游戏中获胜的优势是可忽略的，则有 $Adv_{\mathcal{A}}(\kappa)\leq negl(\kappa)$成立.

2.3.3　密钥协商消息的不可伪造性

本文基于无证书密码机制构造了车联网下的认证密钥协商协议，其中消息的不可伪造性由相应的无证书签名机制所提供，该机制由初始化算法 Setup，密钥生成算法 KeyGen，签名算法 Sign 和签名验证算法 Verify 等四个算法组成. 对于敌手$\mathcal{A}^{1}$和$\mathcal{A}^{2}$，无证书签名机制在适应性选择消息攻击下签名不可伪造的安全模型分别如下所述：

$\mathrm{Ext}_{\mathcal{A}^{1}}^{\mathrm{EUF\text{-}CMA}}(\kappa)$：

1. $(Params,msk)\leftarrow\mathrm{Setup}(1^{\kappa})$

2. $(id^{*},m^{*},\delta^{*})\leftarrow(\mathcal{A}^{1})^{\mathcal{O}^{\mathrm{KeyGen}}(\cdot),\mathcal{O}^{\mathrm{Sign}}(\cdot),\mathcal{O}^{\mathrm{Replace}}(\cdot)}(Params)$

3. If $\mathrm{Verify}(id^{*},m^{*},\delta^{*})=1\wedge(id^{*},m^{*})\notin L_{\mathcal{O}^{\mathrm{Sign}}}$, then output1; Other wise output 0.

$\mathrm{Ext}_{\mathcal{A}^{2}}^{\mathrm{EUF\text{-}CMA}}(\kappa)$：

1. $(Params,msk)\leftarrow\mathrm{Setup}(1^{\kappa})$

2. $(id^{*},m^{*},\delta^{*})\leftarrow(\mathcal{A}^{2})^{\mathcal{O}^{\mathrm{KeyGen}}(\cdot),\mathcal{O}^{\mathrm{Sign}}(\cdot)}(Params,msk)$

3. If $\mathrm{Verify}(id^{*},m^{*},\delta^{*})=1\wedge(id^{*},m^{*})\notin L_{\mathcal{O}^{\mathrm{Sign}}}$, then output 1; Other wise output 0.

其中$\mathcal{O}^{\mathrm{KeyGen}}(\cdot)$是密钥生成谕言机，敌手能对其进行与密钥有关的查询，但是禁止对挑战身份 id^{*} 进行私钥生成询问和部分密钥生成询问(对于$\mathcal{A}^{2}$而言，无需进行部分密钥生成询问)；$\mathcal{O}^{\mathrm{Sign}}(\cdot)$是签名谕言机，敌手能对任意的身份消息组合$(id,m)$进行签名询问，并将相应的信息$(id,m)$添加到列表 $L_{\mathcal{O}^{\mathrm{Sign}}}$中；$\mathcal{O}^{\mathrm{Replace}}(\cdot)$是公钥替换谕言机.

特别地，本文不再赘述无证书签名机制的形式化定义，请读者参考文献[24]和文献[27]中的相关介绍.

3　本文构造

本节将提出我们的安全性增强的车联网认证密钥协商协议，该方案分主要分为下述三个阶段：(1) 在系统建立阶段，OBU、RSU 和 CS 向 KGC 进行注册，生成 CL-PKC 下的公私钥对；(2) 在用户注册阶段，OBU 完成在车联网系统的注册，获得注册追责中心 TRA 为其颁发的授权信息；(3) 在认证与密钥协商阶段，OBU 在 RSU 的协助下，与 CS 完成双向的身份合法性验证，并协商共享的会话密钥.

3.1　具体构造

3.1.1　系统初始化

(1) KGC 生成安全椭圆曲线的相关参数$\{p,q,$

$E/F_q, P, G\}$(参数选择办法由附录 B 所述,此外可以使用现有的安全椭圆曲线选择方法,如 SM2 的参数选择方法);令 H_1、H_2 和 H_3 是三个值域为 Z_q^* 的密码学哈希函数;选取随机数 $\eta \leftarrow_R Z_q^*$ 作为系统主密钥,即 $msk=\eta$,计算系统主公钥 $P_{Pub}=\eta P$;则系统公开参数为 $Params=\{p, q, E/F_q, P, G, H_1, H_2, H_3, P_{Pub}, (\mathcal{E}, \mathcal{D})\}$,其中$(\mathcal{E}, \mathcal{D})$表示对称的加解密算法.

(2) OBU_i 选取 $r'_{Oi} \leftarrow_R Z_q^*$ 作为秘密值,计算 $R'_{Oi}=r'_{Oi}P$,并发消息 (id_{Oi}, R'_{Oi}) 给 KGC;TP 为 OBU_i 随机选取 $r_{Oi} \leftarrow_R Z_q^*$,计算 $R_{Oi}=r_{Oi}P$ 和 $s_{Oi}=r_{Oi}+\eta h_{Oi}$,其中 $h_{Oi}=H_1(id_{Oi}, R_{Oi}, R'_{Oi})$,将部分密钥 (s_{Oi}, R_{Oi}) 发送给 OBU_i.

OBU_i 通过验证等式 $s_{Oi}P=R_{Oi}+h_{Oi}P_{Pub}$ 完成对 (s_{Oi}, R_{Oi}) 的合法性验证;若成立,则设置 OBU_i 的公钥和私钥分别为

$$pk_{Oi}=(R_{Oi}, R'_{Oi}) \text{ 和 } sk_{Oi}=(s_{Oi}, r'_{Oi}).$$

类似地,车联网中的路边单元 RSU_j,云服务器 CS_l 和注册追责中心 TRA 分别执行与 OBU_i 相同的过程并生成对应的公私钥对,其公钥和私钥可分别表示为表 2 中形式. 特别地,本文协议中 OBU 的下标为 i,RSU 的下标为 j,CS 的下标用 l 表示.

表 2 车联网中各实体的公私钥对信息

实体	公私钥对
RSU_j	$pk_{Rj}=(R_{Rj}, R'_{Rj})$ 和 $sk_{Rj}=(s_{Rj}, r'_{Rj})$
CS_l	$pk_{Cl}=(R_{Cl}, R'_{Cl})$ 和 $sk_{Cl}=(s_{Cl}, r'_{Cl})$
TRA	$pk_T=(R_T, R'_T)$ 和 $sk_T=(s_T, r'_T)$

3.1.2 用户注册

TRA 收到 RSU_j 转发的注册消息后,为身份合法的 OBU_i 生成相应的授权信息.

(1) $OBU_i \rightarrow TRA$: $\{id_T, tid'_{Oi}, e_{OiT}, R_{OiT}, m_{OiT}\}$

① OBU_i 计算 $h_{Oi}=H_1(id_{Oi}, R_{Oi}, R'_{Oi})$,随机选取 $r_{OiT} \leftarrow_R Z_q^*$,计算 $tid'_{Oi}=r_{OiT}(R_{Oi}+R'_{Oi}+h_{Oi}P_{Pub})$. 生成消息 $m_{OiT}=\{id_{Oi}, id_T, r_{OiT}, T_{OiT}\}$,其中 id_{Oi} 是 OBU_i 的身份信息,id_T 为 TRA 的身份,T_{OiT} 为时间戳.

② 选取随机数 $r'_{OiT} \leftarrow_R Z_q^*$,计算 $R_{OiT}=r'_{OiT}P$ 和 $\delta_{OiT}=h_{OiT}r'_{Oi}+s_{Oi}+r'_{OiT}$,其中 $h_{OiT}=H_2(id_{Oi}, m_{OiT}, R_{OiT}, R_{Oi}, R'_{Oi})$. 特别地,此处 δ_{OiT} 是 OBU_i 对 m_{OiT} 的签名.

③ OBU_i 首先计算与 TRA 间的共享会话密钥 $k_{OiT}=H_3(r_{OiT}(r'_{Oi}+s_{Oi})(R_T+R'_T+h_TP_{Pub}))$,其中 $h_T=H_1(id_T, R_T, R'_T)$;然后用 k_{OiT} 对 m_{OiT} 加密生成 $e_{OiT}=\mathcal{E}_{k_{OiT}}(m_{OiT})$,并将 $\{id_T, tid'_{Oi}, e_{OiT}, R_{OiT}, m_{OiT}\}$ 发给 TRA.

(2) $TRA \rightarrow OBU_i$: $\{id_T, e_{TOi}, R_{TOi}, \delta_{TOi}\}$

① TRA 计算与 OBU_i 间的共享会话密钥 $k_{TOi}=H_3(tid_{Oi}(r'_T+s_T))$,其中 $h_T=H_1(id_T, R_T, R'_T)$,利用 k_{TOi} 解密 e_{OiT} 获得 $m_{OiT}=\{id_{Oi}, id_T, r_{OiT}, T_{OiT}\}$(即 $m_{OiT}=\mathcal{D}_{k_{TOi}}(e_{OiT})$),若 T'_{OiT} 已失效,则协议被终止执行. 否则当 $\delta_{OiT}P=h_{OiT}R'_{Oi}+R_{Oi}+h_{Oi}P_{Pub}+R_{OiT}$ 成立时,TRA 随机选取 $r_{TlO} \leftarrow_R Z_q^*$,计算 $tid_{Oi}=r'_{TOi}tid'_{Oi}$ 和 $R'_{TOi}=r'_{TOi}P$,把相应的元组 $\{tid_{Oi}, id_{Oi}\}$ 记录到本地数据库 L_{OBU} 中,然后计算 $m_{TOi}=\{id_T, tid_{Oi}, T_{endi}\}$,其中 T_{endi} 是标识 tid_{Oi} 的有效期限. 特别地,会话密钥 k_{TOi} 和签名 δ_{OiT} 合法性验证的正确性由下述等式获得.

$$\begin{aligned} tid_{Oi}(r'_T+s_T) &= r_{OiT}(R_{Oi}+R'_{Oi}+h_{Oi}P_{Pub})(r'_T+s_T) \\ &= r_{OiT}(r'_{Oi}+s_{Oi})(r'_T+s_T)P \\ &= r_{OiT}(r'_{Oi}+s_{Oi})(R_T+R'_T+h_TP_{Pub}), \end{aligned}$$

$$\begin{aligned} \delta_{OiT}P &= (h_{OiT}r'_{Oi}+s_{Oi}+r'_{OiT})P \\ &= h_{OiT}R'_{Oi}+R_{Oi}+h_{Oi}P_{Pub}+R'_{OiT}. \end{aligned}$$

② TRA 并产生消息 $m'_{TOi}=\{m_{TOi}, R'_{TOi}, T_{TOi}\}$,其中 T_{TOi} 为时间戳;然后随机选取 $r_{TOi} \leftarrow_R Z_q^*$,计算 $R_{TOi}=r_{TOi}P$ 和 $\delta_{TOi}=h_{TOi}r'_T+s_T+r_{TOi}$,其中 $h_{TOi}=H_2(id_T, m_{TOi}, R_{TOi}, R_T, R'_T)$. 特别地,$\delta_{TOi}$ 是 TRA 对 m_{TOi} 的签名. 最后,TRA 对 m'_{TOi} 用 k_{TOi} 加密得到 $e_{TOi}=\mathcal{E}_{k_{TOi}}(m'_{TOi})$,将 $\{id_T, e_{TOi}, R_{TOi}, \delta_{TOi}\}$ 发送给 OBU_i.

(3) OBU_i 利用 K_{OiT} 解密 e_{TOi} 获得消息 $m'_{TOi}=\{m_{TOi}, T_{TOi}\}$(即 $m'_{TOi}=\mathcal{D}_{k_{OiT}}(e_{TOi})$),验证时间戳 T_{TOi} 的新鲜性,若有效,则验证等式 $\delta_{TOi}P=h_{TOi}R'_T+R_T+h_TP_{Pub}+R_{TOi}$ 是否成立,如果验证不通过,则终止协议;否则,OBU_i 通过判断等式 $tid_{Oi}=r_{OiT}(r'_{Oi}+s_{Oi})R'_{TOi}$ 是否成立,验证临时身份 tid_{Oi} 的合法性,若临时身份信息合法,则 OBU_i 可利用 TRA 的签名 δ_{TOi} 向 CS_l 请求认证并申请服务;否则终止协议. 我们的构造中,OBU_i 的临时身份 tid_{Oi} 由 OBU_i 与 TRA 双方共同协商生成,避免了单方面生成易导致出现的伪造现象. 特别地,tid_{Oi} 和 δ_{TOi} 合法性验证的正确性由下述等式获得.

$$\begin{aligned} tid_{Oi} &= r'_{TOi}tid'_{Oi}=r'_{TOi}r_{OiT}(R_{Oi}+R'_{Oi}+h_{Oi}P_{Pub}) \\ &= r'_{TOi}r_{OiT}(r'_{Oi}+s_{Oi})P=r_{OiT}(r'_{Oi}+s_{Oi})R'_{TOi}; \end{aligned}$$

$$\begin{aligned} \delta_{TOi}P &= (h_{TOi}r'_T+s_T+r_{TOi})P \\ &= h_{TOi}R'_T+R_T+h_TP_{Pub}+R_{TOi}. \end{aligned}$$

3.1.3 认证与密钥协商

该阶段 OBU 与 CS 间的消息交互过程如图 2

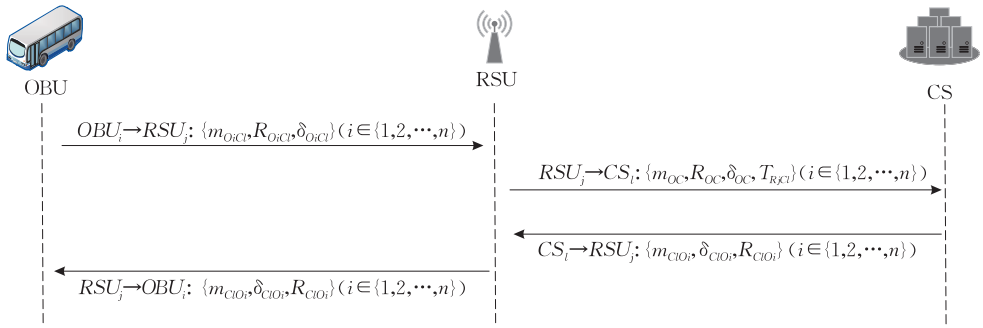


图 2　认证与密钥协商阶段的消息交互流程

所示. 当 OBU_i 向云服务器 CS_l 发起服务请求时，RSU_j 对 $OBU_i(i \in \{1,2,\cdots,n\})$ 的认证信息进行聚合并转发，实现 OBU_i 与 CS_l 间高效的认证与会话密钥协商. 为提升认证的效率，RSU_j 将多个 OBU_i 的认证请求消息聚合后发送给 CS_l. 详细步骤叙述如下：

(1) $OBU_i \rightarrow RSU_j$: $\{m_{OiCl}, R_{OiCl}, \delta_{OiCl}\}$

① OBU_i 选取 $r_{OiCl} \leftarrow_R Z_q^*$，计算 $R_{OiCl} = r_{OiCl}P$.

② 产生消息 $m_{OiCl} = \{id_{Cl}, m_{TOi}, R_{TOi}, T_{OiRj}\}$，其中 T_{OiRj} 是时间戳，id_{Cl} 为 CS_l 的身份.

③ OBU_i 计算 $\delta_{OiCl} = \delta_{TOi} + h_{OiCl}r'_{Oi} + s_{Oi} + r'_{OiCl}$，其中 $h_{OiCl} = H_2(id_{Oi}, m_{OiCl}, R_{OiCl}, R_{Oi}, R'_{Oi})$，然后将 $\{m_{OiCl}, R_{OiCl}, \delta_{OiCl}\}$ 发至 RSU_j. 特别地，上述运算中的 R_{TOi} 和 δ_{TOi} 是 OBU_i 从 TRA 处获得的授权信息.

特别地，在本文协议中，OBU_i 基于 TRA 签发的注册授权信息 R_{TOi} 和 δ_{TOi} 生成服务请求消息 $\{m_{OiCl}, R_{OiCl}, \delta_{OiCl}\}$，对于 CS 而言，确保只有通过 TRA 注册的 OBU_i 才能向其提出服务请求.

(2) $RSU_j \rightarrow CS_l$: $\{m_{OC}, R_{OC}, \delta_{OC}, T_{RjCl}\}$

当 RSU_j 收到请求 $\{m_{OiCl}, R_{OiCl}, \delta_{OiCl}\}_{(i=1,2,\cdots,n)}$ 后，当所有 T_{OiRj} 都新鲜时生成聚合签名 $\{m_{OC}, R_{OC}, \delta_{OC}\}$ 其中 $\delta_{OC} = \sum_{i=1}^{n}\delta_{OiCl}$、$m_{OC} = \{m_{O1Cl}, m_{O2Cl}, \cdots, m_{OnCl}\}$ 和 $R_{OC} = \{R_{O1Cl}, R_{O2Cl}, \cdots, R_{OnCl}\}$，最后转发聚合后的消息 $\{m_{OC}, R_{OC}, \delta_{OC}, T_{RjCl}\}$ 给对应的 CS_l. 需要说明的是，该阶段 OBU_i 的签名 δ_{OiCl} 经 RSU_j 聚合后发送给 CS_l，CS_l 通过一次聚合签名合法性验证完成多个 OBU_i 的身份合法性验证.

(3) $CS_l \rightarrow RSU_j$: $\{m_{ClOi}, \delta_{ClOi}, R_{ClOi}\}$

① CS_l 首先验证时戳 T_{RjCl} 和临时身份 tid_{Oi} 有效期 T_{endi} 是否有效，若有效，然后通过判断等式

$$\delta_{OC}P = nR_T + nh_TP_{Pub} + R'\sum_{i=1}^{n}h_{TOiT} + \sum_{i=1}^{n}R_{TOi} + \sum_{i=1}^{n}h_{OiCl}R'_{Oi} + \sum_{i=1}^{n}R_{Oi} + P_{Pub}\sum_{i=1}^{n}h_{Oi} + \sum_{i=1}^{n}R_{OiCl}$$

是否成立实现对聚合签名 δ_{OC} 的有效性验证.

② 若聚合签名 δ_{OC} 的合法性验证通过，CS_l 为每个 OBU_i 随机选取 $r_{ClOi} \leftarrow_R Z_q^*$，计算 $R_{ClOi} = r_{ClOi}P$；令与 OBU_i 间的共享会话密钥为 $k_{ClOi} = H_3(r_{ClOi}R_{OiCl} + (r'_{Cl} + s_{Cl})(R_{Oi} + R'_{Oi} + h_{Oi}P_{Pub}))$.

③ 生成消息 $m_{ClOi} = \{id_{Cl}, tid_{Oi}, T_{ClOi}\}$，并计算 $\delta_{ClOi} = h_{ClOi}r'_{Cl} + s_{Cl} + r'_{ClOi}$，其中 $h_{ClOi} = H_2(id_{Cl}, m_{ClOi}, R_{ClOi}, R_{Cl}, R'_{Cl})$；最后将消息 $\{m_{ClOi}, \delta_{ClOi}, R_{ClOi}\}(i=1,2,\cdots,n)$ 发送至 RSU_j.

(4) $RSU_j \rightarrow OBU_i$: $\{m_{ClOi}, \delta_{ClOi}, R_{ClOi}\}$

OBU_i 验证 T_{ClOi} 是否有效，若有效，则验证等式 $\delta_{ClOi}P = h_{ClOi}R'_{Cl} + R_{Cl} + h_{Cl}P_{Pub} + R_{ClOi}$ 是否成立，若成立，则认证成功，然后计算与 CS_l 间的共享密钥 $k_{OiCl} = H_3(r_{OiCl}R_{ClOi} + (r'_{Oi} + s_{Oi})(R_{CL} + R'_{Cl} + h_{Cl}P_{Pub}))$；否则认证失败.

此时，云服务器 CS_l 即可使用该会话密钥 k_{OiCl} 对 OBU_i 申请的服务数据进行加密并发送至对应的车辆，保证该服务只能被授权的车辆解密获得.

3.2　正确性

由下述等式可知，OBU_i 与 CS_l 间协商了相同的共享会话密钥，即 $k_{OiCl} = k_{ClOi}$.

$$\begin{aligned} &r_{ClOi}R_{OiCl} + (r'_{Cl} + s_{Cl})(R'_{Oi} + R_{Oi} + h_{Oi}P_{Pub}) \\ &= r_{ClOi}r_{OiCl}P + (r'_{Cl} + s_{Cl})(r'_{Oi} + r_{Oi} + h_{Oi}x)P \\ &= r_{ClOi}r_{OiCl}P + (r'_{Cl} + r_{Cl} + h_{Cl}x)(r'_{Oi} + s_{Oi})P \\ &= r_{OiCl}R_{ClOi} + (r'_{Oi} + s_{Oi})(R_{Cl} + R'_{Cl} + h_{Cl}P_{Pub}). \end{aligned}$$

3.3　安全性证明

3.3.1　认证密钥的安全性

定理 1.　若 DDH 假设成立，那么本文的认证密钥协商协议在 eCk 安全模型下是可证明安全的.

证明.　(1) 证明本文认证密钥协商协议满足协商密钥安全性定义的条件 1.

由正确性分析可知 $k_{OiCl} = k_{ClOi}$，则 OBU_i 与 CS_l 协商了相等的会话密钥. 此外，由于随机数 r_{ClOi} 和 r_{OiCl} 分别由 OBU_i 与 CS_l 从 Z_q^* 上均匀随机选取的，确保了协商密钥 k_{OiCl} 和 k_{ClOi} 对于任意敌手而言都是无法伪造的.

(2)证明本文认证密钥协商协议满足协商密钥安全性定义的条件2,通过下述两个声称进行证明.

声称1. 任意的第一类敌手$\mathcal{A}^1$在密钥协商游戏中获胜的优势$Adv_{\mathcal{A}^1}(\kappa)$是可忽略的,即对于任意的$\mathcal{A}^1$而言,协商密钥$k_{OiCl}$和$k_{ClOi}$是均匀随机的.

具体证明过程详见附录C.

声称2. 任意的第二类敌手$\mathcal{A}^2$在密钥协商游戏中获胜的优势$Adv_{\mathcal{A}^2}(\kappa)$是可忽略的,即对于任意的$\mathcal{A}^2$而言,协商密钥$k_{OiCl}$和$k_{ClOi}$是均匀随机的.

声称1的证明中模拟器$\mathcal{S}$掌握了完整的主私钥,具备向敌手$\mathcal{A}^2$提供主私钥的能力,因此由声称1的证明可知$\mathcal{A}^2$获胜的优势$Adv_{\mathcal{A}^2}(\kappa)$是可忽略的.也就是说,若$\mathcal{A}^2$能以不可忽略的优势$\varepsilon_2(\kappa)$攻击本文密钥协商协议成功,则$\mathcal{S}$至少能以不可忽略的优势$\left(\frac{1}{q_k}+\frac{1}{q_C}\right)\frac{\varepsilon_2(\kappa)}{\mathrm{e}q^2}$解决DDH困难问题.

综上所述,认证密钥满足相关的安全性要求,因此本文协议是安全的.

3.3.2 通信消息的不可伪造性

本节将在随机谕言机模型下证明本文认证密钥协商协议中通信消息的不可伪造性.

定理2. 在随机谕言机模型下,若存在敌手$\mathcal{A}^1$能够在多项式时间内以不可忽略的优势$\varepsilon_3(\kappa)$成功伪造上述协议的协商消息,那么存在一个模拟器$\mathcal{S}$能以优势$\left(1-\frac{1}{\mathrm{e}}\right)\frac{\varepsilon_3(\kappa)}{\mathrm{e}(q_d+q_s+1)q_{H_1}}$成功解决DL困难性问题,$\mathcal{A}^1$最多进行$q_d$次部分密钥提取询问、$q_s$次私钥提取询问和$q_{H_1}$次谕言机$H_1$询问.

具体证明过程详见附录D.

定理3. 在随机谕言机模型下,若存在敌手$\mathcal{A}^2$能够在多项式时间内以不可忽略的优势$\varepsilon_4(\kappa)$成功伪造上述协议的协商消息,那么存在一个模拟器$\mathcal{S}$能以优势$\left(1-\frac{1}{\mathrm{e}}\right)\frac{\varepsilon_4(\kappa)}{\mathrm{e}(q_s+q_v+1)q_{H_2}}$成功解决DL困难性问题,$\mathcal{A}^2$最多进行$q_s$次私钥协商询问、$q_v$次秘密值提取询问和$q_{H_2}$次谕言机$H_2$询问.

具体证明过程详见附录E.

3.4 性能分析与比较

对于n个*OBU*的密钥协商过程,本节将本文协议与相关工作[8-9,16,18-19,25-26]在效率和性能两方面进行对比,其中效率主要从计算、通信和存储三个方面进行考量,同时还考虑方案中证书的管理或使用到的其他底层工具对方案效率的影响;性能主要分析匿名性、安全性、临时私钥泄露攻击等特性.特别地,计算效率主要统计密钥协商过程中*OBU*和*CS*端相关运算的执行次数;通信效率由*OBU*与*CS*间的通信次数和公钥的长度来衡量;存储效率由公开参数和私钥的长度所决定.

表3中T_M表示加法循环群上的点乘运算,T_P表示双线性映射运算.此外,用$|G|$表示群G中元素的长度,$|Z_q^*|$表示Z_q^*中元素的长度,H表示密码学哈希函数,相应系数表示元素的个数.特别地,公开参数长度将忽略对基础群结构的统计.文献[18]基于智能卡的对称密钥所设计,因此表3中未统计相应公私钥的长度.此外,在表4中,符号"√"和"×"分别表示相关协议是否满足相应的要求.

表3 效率比较

机制	证书管理(或使用的其他辅助工具)	计算效率		通信效率		存储效率	
		OBU	*CS*	*OBU*与*CS*间通信次数	公钥长度	公开参数长度	私钥长度
文献[8]	无需管理证书	$5T_M$	$5nT_M$	3	$2\|G\|$	$1\|G\|+2H$	$2\|Z_q^*\|$
文献[9]	无需管理证书	$4T_M$	$4nT_M$	3	$2\|G\|$	$1\|G\|+4H$	$2\|Z_q^*\|$
文献[16]	无需管理证书	$9T_M$	$(4n+6)T_M$	4	$1\|G\|$	$1\|G\|+4H$	$1\|Z_q^*\|$
文献[18]	需智能卡协助	$3T_M$	$3nT_M$	3	$\perp$	$2\|G\|+1H$	$\perp$
文献[19]	无需管理证书	$5T_M$	$(n+4)T_M$	2	$2\|G\|$	$1\|G\|+5H$	$2\|Z_q^*\|$
文献[25]	无需管理证书	$6T_M+2T_P$	$6nT_M+2nT_P$	2	$2\|G\|$	$1\|G\|+2H$	$2\|Z_q^*\|$
文献[26]	需模糊提取器的协助	$5T_M$	$4nT_M+2nT_P$	3	$2\|G\|$	$1\|G\|+6H$	$1\|G\|$
本文协议	无需管理证书	$7T_M$	$(2n+5)T_M$	2	$2\|G\|$	$1\|G\|+2H$	$2\|Z_q^*\|$

表4 性能比较

机制	匿名性	抗临时私钥泄露	批量认证	安全性
文献[8]	无	√	×	安全
文献[9]	无	√	×	安全
文献[16]	无	×	√	安全
文献[18]	无	√	×	安全
文献[19]	强	√	√	不安全
文献[25]	无	×	×	安全
文献[26]	弱	√	×	安全
本文协议	强	√	√	安全

由表3和表4可知,文献[8-9,16,18]和文献[25]未提供匿名性保护,文献[16]仅具有弱匿名性,因此敌手能够通过*OBU*的身份信息实现对车辆的定位.文献[25]基于参与者的私钥完成协商密钥的计算,导致其无法抵抗临时私钥泄露攻击.文献[8-9,18,25]和文献[26]均无法提供*OBU*与*CS*间的批量认证及密钥协商,使得上述机制在完成多个密钥协商时的计算效率较低.本文分析发现文献[19]中

的协议无法满足其所声称的安全性，任意第一类敌手均可通过替换合法 *OBU* 公钥的形式完成与 *CS* 的密钥协商，建立共享的会话密钥. 此外，文献[8-9，16，18]和文献[26]的构造中，*OBU* 与 *CS* 间的消息交互次数较多，导致相应构造的通信效率较低. 文献[18]和文献[25]的方案分别基于智能卡和模糊提取器等基础工具实现，在一定程度上降低了方案的普适性.

本文对基础密码操作的运行时间在个人电脑(CPU：Intel(R) Core i5-4200H；主频：2.8 GHz；内存：2 GB；硬盘：128 GB 固态；操作系统：Ubuntu 18.04 64 位，PBC 算法库的版本号：PBC-0.5.14)上进行了测算，通过对点乘和双线性映射操作计算运行 20 次的平均值得到相应的运行时间，通过测算可知相应的 $T_M=1.112$ ms 和 $T_P=1.365$ ms. 特别地，本文在计算效率仿真时，PBC 算法库中相应参数的具体设置详见附录 B. 图 3 所示为 *OBU* 数量为 100 ($n=100$)时，相关构造在 *OBU* 和 *CS* 端运行时间总和的对比结果. 图 4 所示为完成不同数量的密钥协商时，相关构造总运行时间的对比结果.

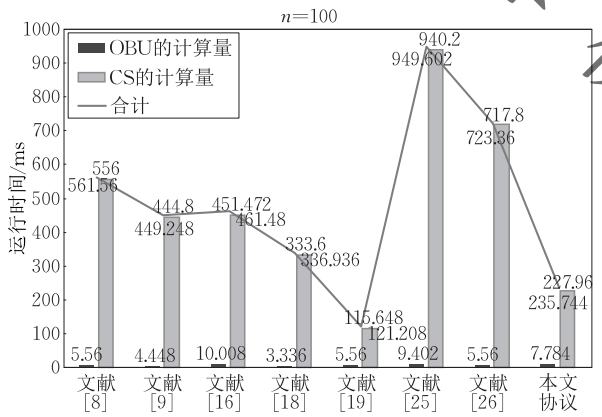


图 3　本文构造与相关机制的运行时间对比

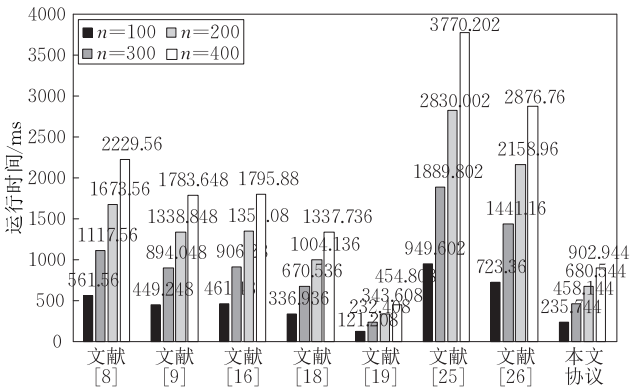


图 4　本文构造与相关机制的总运行时间对比

由图 3 和图 4 可知，相对于现有的认证与密钥协商协议[8-9,16,18,25-26]，本文构造在计算效率方面具有明显的优势，满足车联网对计算性能的高效性要求. 虽然文献[19]的计算效率较高，但是该协议不具备其应有的安全性；由于缺少必要的验证计算，导致任意的第一类敌手能够伪造密钥协商消息.

本文方案在计算效率方面具备明显优势，在安全性能够更好地满足车联网认证需求的同时，通过高效的批处理认证技术提升了方案的工作效率. 此外，为提高云服务器 *CS* 对车载单元 *OBU* 的认证效率，路边单元 *RSU* 将本区域内多个 *OBU* 的服务请求聚合后统一发给相应的 *CS*，*CS* 可在一次验证操作内完成对多个 *OBU* 的合法性验证.

综上所述，本文构造中消息的聚合在一定程度上提升了消息的传输效率. 此外，*OBU* 向 *CS* 申请服务时，基于 *TRA* 签发的授权签名 δ_{TOi} 生成相应的请求消息，*CS* 基于 *OBU* 和 *TRA* 的公开信息对请求消息的合法性进行了验证，避免了非授权的 *OBU* 向 *CS* 提出服务申请. 特别地，本文所提到的协议的轻量级特征是一个相对的概念，由于双线性映射与群上的乘法运算相比，它的运算量较大，因此不使用双线性映射构造的协议具有较高的计算效率，则相应的协议更接近拥有轻量级特征.

4　结束语

对现有认证密钥协商协议的分析基础上，本文提出了安全性可证明的高效车联网认证密钥协商协议的新型构造，并基于 DDH 和 DL 困难性假设对协商密钥的安全性和通信消息的不可伪造性进行了形式化证明. 与现有相关协议相比，本文构造具有更高的计算和通信效率，能满足双方认证、抵抗临时私钥泄露攻击等；此外，上述特性使得该协议在车联网中具有更好的性能和适应性. 此外，本文在附录 F 中对分叉引理进行签名安全性证明的方法进行了总结.

参　考　文　献

[1] Al-Riyami S S, Paterson K G. Certificateless public key cryptography//Proceedings of the 9th International Conference on the Theory and Application of Cryptology and Information Security. Taipei, China, 2003: 452-473

[2] Mwitende G, Ye Y, Ali I, Li F. Certificateless authenticated key agreement for blockchain-based WBANs. Journal of Systems Architecture, 2020, 110(11): 101777

[3] Bouakkaz S, Semchedine F. A certificateless ring signature scheme with batch verification for applications in VANET. Journal of Information Security and Applications, 2020, 55: 102669

[4] Lopes A P G, Gondim P R L. Group authehntication protocol based on aggregated signatures for D2D communication. Computer Networks, 2020, 178: 107192

[5] Bagga P, Sutrala A K, Das A K, Vijayakumar P. Blockchain-based batch authentication protocol for Internet of Vehicles. Journal of Systems Architecture, 2021, 113: 101877

[6] Sun Yipin, Lu Rongxing, Lin Xiaodong, et al. An efficient pseudonymous authentication scheme with strong privacy preservation for vehicular communications. IEEE Transactions on Vehicular Technology, 2010, 59(7): 3589-3603

[7] Wu Tao, Jing Xiaojun. Two-party certificateless authenticated key agreement protocol with enhanced security. The Journal of China Universities of Posts and Telecommunications, 2019, 26(1): 12-20+39

[8] Xu Sheng-Wei, Ren Xiong-Peng, Chen Cheng, et al. Provably secure certificateless two-party authenticated key agreement protocol. Journal of Cryptologic Research, 2020, 7(6): 886-898(in Chinese)
(许盛伟，任雄鹏，陈诚等. 可证安全的无证书两方认证密钥协商协议. 密码学报，2020，7(6)：886-898)

[9] Liu Bo, Zhou Yu-Yang, Hu Fei, Li Fa-Gen. User authentication and key agreement protocol for mobile client-multi-server environment. Journal of Cryptologic Research, 2018, 5(2): 111-125(in Chinese)
(刘波，周雨阳，胡飞，李发根. 适用于移动客户端-多服务器环境的用户认证与密钥协商协议. 密码学报，2018，5(2)：111-125)

[10] Jia Xiaoying, He Debiao, Kumar N, Choo K-K R. Authenticated key agreement scheme for fog-driven IoT healthcare system. Wireless Networks, 2019, 25(8): 4737-4750

[11] Wu Tsu-Yang, Wang Tao, Lee Yu-Qi, et al. Improved authenticated key agreement scheme for fog-driven IoT healthcare system. Security and Communication Networks, 2021, 6658041:1-6658041:16

[12] Vasudev H, Das D, Vasilakos A V. Secure message propagation protocols for IoVs communication components. Computers & Electrical Engineering, 2020, 82: 106555

[13] Yu Sungjin, Lee JoonYoung, Kisung P, et al. IoV-SMAP: Secure and efficient message authentication protocol for IoV in smart city environment. IEEE Access, 2020, 8: 167875-167886

[14] Rahmani A M, Mohammadi M, Rashidi S, et al. Questioning the security of three recent authentication and key agreement protocols. IEEE Access, 2021, 9: 98204-98217

[15] Yao Ying-Ying, Chang Xiao-Lin, Misic J V, Misic V B. Lightweight batch AKA scheme for user-centric ultra-dense networks. IEEE Transactions on Cognitive Communications and Networking, 2020, 6(2): 597-606

[16] Cheng Qing-Feng, Chen Ting, Ma Si-Qi, Li Xing-Hua. Improvement on a batch authenticated key agreement scheme. Frontiers of Computer Science, 2022, 16(2): 162803

[17] Huang Bao-Jun, Khan M K, Wu Li-Bing, et al. An efficient remote user authentication with key agreement scheme using elliptic curve cryptography. Wireless Personal Communications, 2015, 85: 225-240

[18] Bouchaala M, Ghazel C, Saïdane L A. Enhancing security and efficiency in cloud computing authentication and key agreement scheme based on smart card. The Journal of Supercomputing, 2022, 78(1): 497-522

[19] Zhang Wen-Fang, Lei Li-Ting, Wang Xiao-Min, Wang Yu. Secure and efficient authentication and key agreement protocol using certificateless aggregate signature for cloud service oriented VANET. Acta Electronica Sinica, 2020, 48(9): 1814-1823(in Chinese)
(张文芳，雷丽婷，王小敏，王宇. 面向云服务的安全高效无证书聚合签名车联网认证密钥协商协议. 电子学报，2020，48(9)：1814-1823)

[20] Lippold G, Boyd C, Nieto J M G. Strongly secure certificateless key agreement//Proceedings of the 3rd International Conference on Pairing-Based Cryptography (Pairing 2009). Palo Alto, USA, 2009: 206-230

[21] Su Hang, Liu Jian-Wei, Tao Rui. Hierarchical certificateless authenticated key agreement protocol. Journal on Communications, 2016, 37(7): 161-171(in Chinese)
(苏航，刘建伟，陶芮. 无证书的层次认证密钥协商协议. 通信学报，2016，37(7)：161-171)

[22] Pointcheval D, Stern J. Security proofs for signature schemes//Advances in Cryptology—Proceedings of the International Conference on the Theory and Application of Cryptographic Techniques (EUROCRYPT'96). Saragossa, Spain, 1996: 387-398

[23] Yang Bo. Modern Cryptography. Beijing: Tsinghua University Press, 2015: 65-69(in Chinese)
(杨波. 现代密码学. 北京：清华大学出版社，2015：65-69)

[24] Hu Bing-Jie, Zhou Yan-Wei, Yang Bo, Zhang Jing. Secure and efficient certificateless signature scheme without bilinear parings. Journal of Yunnan University: Natural Sciences Edition, 2021, 43(3): 462-469(in Chinese)
(胡冰洁，周彦伟，杨波，张晶. 安全高效的无双线性对的无证书签名方案. 云南大学学报(自然科学版)，2021，43(3)：462-469)

[25] Li Yan-Ping, Chen Wei-Feng, Cai Zhi-Ping, Fang Yu-Guang. CAKA: A novel certificateless-based cross-domain authenticated key agreement protocol for wireless mesh networks. Wireless Networks, 2016, 22(8): 2523-2535

[26] Odelu V, Das A K, Kumari S, et al. Provably secure authenticated key agreement scheme for distributed mobile cloud computing services. Future Generation Computer Systems, 2017, 68: 74-88

[27] Zhang Zhen-Chao, Liu Ya-Li, Yin Xin-Chun, Huang Ke-Ke. Analysis and improvement of certificateless signature schemes. Journal of Cryptologic Research, 2020, 7(3): 389-403(in Chinese)
(张振超，刘亚丽，殷新春，黄可可. 无证书签名方案的分析及改进. 密码学报，2020，7(3)：389-403)

附录 A.　文献[19]中协议的安全性分析.

为解决当前车联网认证密钥协商协议效率低下及车辆公私钥频繁更新的问题,文献[19]提出一个认证密钥协商协议.本文分析发现,该协议无法满足其所声称的安全性.下面将对文献[19]中协议的安全性进行分析.此外,该协议的具体过程详见原文,此处不再赘述.

第一类敌手$\mathcal{A}^1$在进行伪造攻击之前将对已掌握身份的OBU_i(对应的身份信息为id_{Oi})进行公钥替换操作,即$\mathcal{A}^1$将该OBU_i的合法公钥替换为自己已掌握的信息$pk_{Oi}=(P_{Oi},R'_{Oi})$:

$\mathcal{A}^1$随机选取$\alpha,\beta\leftarrow_R Z_q^*$,计算$\tilde{P}_{Oi}=\alpha P$和$\tilde{R}_{Oi}=\beta P$,然后用已知的$pk'_{Oi}=(\tilde{P}_{Oi},\tilde{R}_{Oi})$替换$OBU_i$的原始公钥$pk_{Oi}=(P_{Oi},R'_{Oi})$,则系统中的其他参与者认为$OBU_i$的公钥是$pk'_{Oi}=(\tilde{P}_{Oi},\tilde{R}_{Oi})$.$\mathcal{A}^1$在公钥替换完成之后,将伪装成$OBU_i$与$RSU_j$、$TRA$和$CS_l$分别执行文献[19]的认证密钥协商协议.

A.1. $\mathcal{A}^1$对注册授权阶段的伪造攻击

(1) $\mathcal{A}^1\rightarrow TRA:\{id_T,TID_i,e_{OiT}\}$

①$\mathcal{A}^1$计算$\tilde{h}_{Oi}=H_1(id_{Oi},\tilde{R}_{Oi})$,随机选取$r_{OiT}\leftarrow_R Z_q^*$,得到临时身份$tid_{Oi}=r_{OiT}(\tilde{R}_{Oi}+\tilde{h}_{Oi}\tilde{P}_{Oi})$,计算$h'_T=H_1(id_T,R'_T)$及与$TRA$的共享会话密钥$k_{OiT}=H_2(r_{OiT}(\alpha\tilde{h}_{Oi}+\beta)(R'_T+h'_TP_T))$,产生消息$m_{OiT}=\{id_{Oi},id_T,r_{OiT},T_{OiT}\}$.

②OBU_i用会话密钥k_{OiT}对m_{OiT}加密生成$e_{OiT}=Enc_{k_{OiT}}(m_{OiT})$,将$\{id_T,tid_i,e_{OiT}\}$发给$TRA$.

(2) $TRA\rightarrow\mathcal{A}^1:\{id_T,e_{OiT}\}$

①TRA计算$h'_T=H_1(id_T,R'_T)$及共享的会话密钥$k_{TOi}=H_2(tid_{Oi}(r'_T+s_Th'_T))$,利用$k_{TOi}$解密$e_{OiT}$,由于$T_{OiT}$是$\mathcal{A}^1$选取的新鲜时戳,能通过$TRA$的有效性检测.

②计算$h'_{Oi}=H_1(id_{Oi},\tilde{R}_{Oi})$,由于临时身份$tid_{Oi}$的验证等式$r_{OiT}(\tilde{R}_{Oi}+h'_{Oi}\tilde{P}_{Oi})=tid_{Oi}$成立(由于$h'_{Oi}=\tilde{h}_{Oi}$,则该等式成立),将元组$\{tid_{Oi},id_{Oi}\}$保存到相应的数据库$L_{OBU}$中,并生成消息$m_{TOi}=\{id_T,tid_{Oi},T_{endi}\}$,其中$T_{endi}$为临时身份$tid_{Oi}$的有效期.

③TRA选取$r_{TOi}\leftarrow_R Z_q^*$,计算$R_{TOi}=r_{TOi}P$,$h_{TOi}=H_3(m_{TOi},R_{TOi})$和$v_{TOi}=(r'_T+h'_Ts_T)h_{TOi}+r_{TOi}$,并产生消息$m'_{TOi}=\{m_{TOi},v_{TOi},R_{TOi},T_{TOi}\}$.

④TRA用共享会话密钥k_{TOi}对m'_{TOi}加密得到$e_{TOi}=Enc_{k_{TOi}}(m'_{TOi})$,最后将$(id_T,e_{TOi})$发送给$\mathcal{A}^1$.

(3) $\mathcal{A}^1$利用k_{OiT}解密e_{TOi},验证时间戳T_{TOi}的有效性,然后计算$h_{TOi}=H_3(m_{TOi},R_{TOi})$,由于等式$v_{TOi}P=(R'_T+h'_TP_T)h_{TOi}+R_{TOi}$成立,则敌手$\mathcal{A}^1$完成对TRA的身份合法性验证,实现了向$TRA$注册的目的,并且能够利用$TRA$的签名$v_{TOi}$向$CS_l$请求认证并申请服务.签名$v_{TOi}$的合法性验证由下述等式保障.

$$\begin{aligned}v_{TOi}P&=((r'_T+h'_Ts_T)h_{TOi}+r_{TOi})P\\&=(r'_T+h'_Ts_T)h_{TOi}P+r_{TOi}P\\&=(R'_T+h'_TP_T)h_{TOi}+R_{TOi}.\end{aligned}$$

由下述等式可知,敌手$\mathcal{A}^1$通过了TRA的身份合法性验证,并与其协商了相等的共享密钥k_{OiT}和k_{TOi}.换句话讲,$\mathcal{A}^1$伪装成合法OBU_i通过了TRA的身份合法性验证,并获得了TRA颁发的授权信息(v_{TOi},R_{TOi}).

$$\begin{aligned}tid_i(r'_T+s_Th'_T)&=r_{OiT}(\tilde{R}_{Oi}+h'_{Oi}\tilde{P}_{Oi})(r'_T+s_Th'_T)\\&=r_{OiT}(\beta P+h'_{Oi}\alpha P)(r'_T+s_Th'_T)\\&=r_{OiT}(\beta+h'_{Oi}\alpha)(r'_TP+s_Th'_TP)\\&=r_{OiT}(h'_{Oi}\alpha+\beta)(R'_T+h'_TP_T).\end{aligned}$$

A.2. $\mathcal{A}^1$对认证密钥协商阶段的伪造攻击

$\mathcal{A}^1$向云服务器CS_l发起服务请求时,RSU_j对OBU_i($i\in\{1,2,\cdots,n\}$)(包括敌手$\mathcal{A}^1$)的认证信息进行聚合并转发,实现OBU_i与CS_l间高效的认证与会话密钥协商,详细步骤如下:

(1) $\mathcal{A}^1\rightarrow RSU_j:\{m_{OiCl},v_{OiCl}\}(i\in\{1,2,\cdots,n\})$

①$\mathcal{A}^1$选取$r_{OiCl}\leftarrow_R Z_q^*$,计算$\tilde{h}_{Oi}=H_1(id_{Oi},\tilde{R}_{Oi})$和$R_{OiCl}=r_{OiCl}(\alpha\tilde{h}_{Oi}+\beta)P$,生成相应的消息$m_{OiCl}=\{id_{Cl},m_{TOi},R_{TOi},R_{OiCl},T_{OiRj}\}$.

②$\mathcal{A}^1$计算$v_{OiCl}=v_{TOi}+r_{OiCl}(\alpha\tilde{h}_{Oi}+\beta)h_{OiCl}$,其中$h_{OiCl}=H_4(m_{OiCl})$.将消息$\{m_{OiCl},v_{OiCl}\}$发至本区域对应的$RSU_j$.

由下述等式可知,敌手$\mathcal{A}^1$的伪造签名v_{OiCl}能够通过相应签名验证算法的验证.

$$\begin{aligned}v_{OiCl}P&=(v_{TOi}+r_{OiCl}(\beta+\alpha\tilde{h}_{Oi})h_{OiCl})P\\&=v_{TOi}P+r_{OiCl}(\beta+\alpha\tilde{h}_{Oi})h_{OiCl}P\\&=(R'_T+h'_TP_T)h_{TOi}+R_{TOi}+h_{OiCl}R_{OiCl}.\end{aligned}$$

由于敌手$\mathcal{A}^1$伪造的签名v_{OiCl}能够通过CS_l的合法性验证,因此RSU_j将敌手$\mathcal{A}^1$的伪造签名v_{OiCl}连同其他车辆的签名一起聚合以后,相应的聚合签名v_{OC}同样能够通过CS_l的签名合法性验证操作.

(2) $RSU_j\rightarrow CS_l:\{m_{OiCl},v_{OC},T_{RiCl}\}(i=1,2,\cdots,n)$

RSU_j收到相应的服务请求后,若所有的时间戳T_{OiRj}都是新鲜的,则生成聚合消息$v_{OC}=\sum_{i=1}^{n}v_{OiCl}$,然后将处理后的信息$\{m_{OiCl},v_{OC},T_{RjCl}\}(i\in\{1,2,\cdots,n\})$转发给对应的云服务器器$CS_l$.

(3) $CS_l\rightarrow RSU_j:\{m_{ClOi},v_{ClOi}\}(i=1,2,\cdots,n)$

①CS_l首先验证T_{RjCl}和T_{endi}的有效性,计算$h'_T=H_1(id_T,R'_T)$,$h_{TOi}=H_3(m_{TOi},R_{TOi})$,$h_{OiCl}=H_4(m_{OiCl})$和$R_{TO}=\sum_{i=1}^{n}R_{TOi}$,

②由于聚合签名v_{OC}的合法性验证等式$v_{OC}P=(R'_T+h'_TP_T)\sum_{i=1}^{n}h_{TOi}+R_{TO}+\sum_{i=1}^{n}R_{OiCl}h_{OiCl}$成立,则$CS_l$为每个$OBU_i$(包含敌手$\mathcal{A}^1$)随机选取$r_{ClOi}\leftarrow_R Z_q^*$,计算$R_{ClOi}=r_{ClOi}(r'_{Cl}+s_{Cl}h'_{Cl})P$和共享会话密钥$sk_{ClOi}=H_2(r_{ClOi}(r'_{Cl}+s_{Cl}h'_{Cl})R_{OiCl})$;产生消息$m_{ClOi}=\{id_{Cl},tid_{Oi},R_{ClOi},T_{ClOi}\}$.

计算$v_{ClOi}=r'_{ClOi}(r'_{Cl}+s_{Cl}h'_{Cl})+(r'_{Cl}+s_{Cl}h'_{Cl})h_{ClOi}$,其中$h_{ClOi}=H_5(m_{ClOi})$,最后将相应的通信消息$\{m_{ClOi},v_{ClOi}\}(i\in\{1,2,\cdots,n\})$发送至$RSU_j$.

(4) $RSU_j\rightarrow OBU_i:\{m_{ClOi},v_{ClOi}\}(i=1,2,\cdots,n)$

$\mathcal{A}^1$收到RSU_j转发的CS_l的应答消息后,首先验证时间

戳 T_{ClOi} 的有效性，然后计算 $h_{ClOi}=H_5(m_{ClOi})$ 和 $h'_{Cl}=H_1(id_{Cl},R'_{Cl})$，由于等式 $v_{ClOi}P=R_{ClOi}+(R'_{Cl}+P_{Cl}h'_{Cl})h_{ClOi}$ 成立，则计算会话密钥 $sk_{OiCl}=H_2(r_{OiCl}(\beta+\alpha\tilde{h}_{Oi})R_{ClOi})$. 签名 v_{ClOi} 的合法性验证由下述等式保障.

$$\begin{aligned}v_{ClOi}P&=(r'_{ClOi}(r'_{Cl}+s_{Cl}h'_{Cl})+(r'_{Cl}+s_{Cl}h'_{Cl})h_{ClOi})P\\&=r'_{ClOi}(r'_{Cl}+s_{Cl}h'_{Cl})P+(r'_{Cl}+s_{Cl}h'_{Cl})h_{ClOi}P\\&=R_{ClOi}+(R'_{Cl}+P_{Cl}h'_{Cl})h_{ClOi}.\end{aligned}$$

由下述等式可知，敌手 $\mathcal{A}^1$ 与云服务器 CS_l 间协商了相等的会话密钥 sk_{OiCl} 和 sk_{ClOi}，CS_l 将使用该会话密钥 sk_{OiCl} 对 $\mathcal{A}^1$ 申请的服务进行加密并发送至对应的车辆. 换句话讲，$\mathcal{A}^1$ 伪装成 OBU_i 通过了 CS_l 的验证，并将获得 CS_l 提供的相关服务.

$$\begin{aligned}r_{ClOi}(r'_{Cl}+s_{Cl}h'_{Cl})R_{OiCl}&=r_{ClOi}(r'_{Cl}+s_{Cl}h'_{Cl})r_{OiCl}(\beta+\alpha\tilde{h}_{Oi})P\\&=r_{OiCl}(\beta+\alpha\tilde{h}_{Oi})R_{ClOi}.\end{aligned}$$

综上所述，第一类敌手 $\mathcal{A}^1$ 通过公钥替换的方式伪装成 OBU_i，在认证授权阶段通过了 TRA 的合法性验证，并获得了 TRA 签发的授权信息；在认证密钥协商阶段通过了 CS_l 的合法性验证，并与其协商了相等的共享会话密钥. 此外，上述过程中，OBU_i 的私钥 $sk_{Oi}=(s_{Oi},r'_{Oi})$ 和系统主私钥 $msk=x$ 对 $\mathcal{A}^1$ 是保密的，仅进行了 OBU_i 的公钥替换，上述操作完全符合对敌手 $\mathcal{A}^1$ 的能力限制.

A.3. 设计缺陷分析

我们认为该协议出现上述安全性缺陷的主要原因有以下几个方面：

(1) 在该协议中，将 OBU_i 的公私钥分别设置为 $pk_{Oi}=(P_{Oi},R'_{Oi})$ 和 $sk_{Oi}=(s_{Oi},r'_{Oi})$，其中 $P_{Oi}=s_{Oi}P$ 和 $R'_{Oi}=r'_{Oi}P$，这导致协议中签名合法性的验证等式中不会出现主公钥 P_{Pub}，直接使用 OBU_i 的公钥元素 P_{Oi} 和 R'_{Oi} 即可完成相应签名的合法性验证. 由于验证算法缺少主公钥 P_{Pub} 的参与，即使 $\mathcal{A}^1$ 不知道主私钥，很容易通过替换公钥的方式生成合法的伪造签名.

(2) 注册授权阶段，OBU_i 发给 TRA 的消息中缺少相应的签名信息，TRA 对 OBU_i 的验证是通过临时身份 $tid_{Oi}=r_{OiT}(R'_{Oi}+h'_{Oi}P_{Oi})$ 的计算完成的，其中随机数 r_{OiT} 是由 OBU_i 加密后发送给 TRA 的，并且 TRA 解密用的对称密钥 $k_{TOi}=H_2(tid_{Oi}(r'_T+s_Th'_T))$ 是由 OBU_i 提供的临时身份 tid_{Oi} 和 TRA 的私钥 $sk_T=(s_T,r'_T)$ 计算完成. 那么 $\mathcal{A}^1$ 可以根据 TRA 的计算方式提前进行相应的准备，使其与 TRA 能够协商出相等的会话密钥，并且确保 TRA 能够计算出与 $\mathcal{A}^1$ 相等的临时身份 tid_{Oi}.

(3) 认证密钥协商中，OBU_i 发给 CS_l 的消息中虽包含签名 $v_{OiCl}=v_{TOi}+r_{OiCl}(r'_{Oi}+s_{Oi}h'_{Oi})h_{OiCl}$，但验证由 $v_{OiCl}P=(R'_T+h'_TP_T)h_{TOi}+R_{TOi}+R_{OiCl}h_{OiCl}$ 完成，其中 $(R'_T+h'_TP_T)h_{TOi}+R_{TOi}$ 是对 TRA 授权签名 v_{TOi} 的合法性验证，而对 OBU_i 的验证部分 $R_{OiCl}h_{OiCl}$ 完全是由 OBU_i 提供的，并未使用 OBU_i 的公钥信息. 那么 $\mathcal{A}^1$ 可以根据 CS_l 的验证等式提前进行准备，使得伪造签名能通过 CS_l 的合法性验证，并与其协商出相等的会话密钥.

综上所述，协议设计缺陷的核心问题是公私钥的设置，导致主公钥 P_{Pub} 不再参与相应签名的合法性验证；此外签名合法性验证等式中未涉及用户公钥信息的不足同样助力了 $\mathcal{A}^1$ 的公钥替换攻击. 针对该协议的不足，为有效抵抗 $\mathcal{A}^1$ 的公钥替换攻击，在 CL-PKC 的设计中应考虑以下两点：

(1) 用户私钥的生成算法中应包含公钥的哈希值信息，这样用户私钥就包含了公钥的承诺，一旦敌手 $\mathcal{A}^1$ 进行了公钥替换攻击，势必导致关于私钥的相关验证等式中的公钥哈希值发生变化，使得该等式不再成立.

(2) 主公钥必须参与签名的合法性验证，该操作保证了签名的生成私钥必须包含主私钥，由于敌手 $\mathcal{A}^1$ 无法掌握系统主私钥，增加了敌手 $\mathcal{A}^1$ 伪造合法签名的难度.

附录 B. 相关参数的选择方法

B.1. 有限域 $GF(q)$ 上选取椭圆曲线的建立

(1) 选择大素数 $q=p$ 或 $q=2^m$，且有 $q>2^{160}$.

(2) 生成随机数 a 和 b，$a,b\in GF(q)$ 且 $(4a^3+27b^2)q\neq 0$.

(3) 求解有限域 $GF(q)$ 上椭圆曲线的阶 $\sharp E(F_q)$.

(4) 对 $\sharp E(F_q)$ 进行素数检测（通过重复选择相应的随机参数 a 和 b，使得 $\sharp E(F_q)$ 含有大素数因子）；

(5) 如果曲线上点的个数不是素数，则分解，看有没有大素数因子，若有且大素数因子的规模也大于 2^{160}，则选用该曲线；否则返回到第(2)步，重新选取椭圆曲线参数 a 和 b，直至满足条件.

(6) 得到有限域 $GF(q)$ 上的椭圆曲线 E/F_q，相应的方程式为 $y^2=x^3+ax+b$.

B.2. PBC 算法库的参数设置

本文在仿真时，PBC 算法库中相关参数的设置如下所示：椭圆曲线是 $GF(q)$ 上的曲线 $y^2=x^3+x$，其中 q 是满足 $q=3 \bmod 4$ 的素数，G_1 和 G_2 都是群 $GF(q)$ 上的点集合，G_T 为 $F(q^2)$ 的一个子群，它的阶 r 是 $q+1$ 的素因子，令 $q+1=r\cdot h$（h 是 12 的倍数），且素数 r 具有 $2^a\pm 2^b\pm 1$ 的形式，其中整数 a 和 b 满足 $0<b<a$.

附录 C. 声称 1 的形式化证明.

证明. 模拟器 $\mathcal{S}$ 收到 DDH 困难问题的挑战元组 (P,aP,bP,T_μ)，其中 $T_\mu=abP$ 或者 $T_\mu=cP$，$a,b,c\in Z_q^*$，其目标是计算. 令敌手 $\mathcal{A}^1$ 在游戏中进行了 q_k 次私钥生成询问，并且对 q_C（其中 $q_k>q_C$）个参与者执行了 $Corrupt$ 询问.

$\mathcal{S}$ 随机猜测 $J\leftarrow_R(0,q_k]$ 是 $\mathcal{A}^1$ 在 $Test$ 询问中要挑战的会

话，则$\mathcal{S}$猜测正确的概率为 $\rho=\frac{1}{q_k}$. $\Pi_{i,j}^t$和$\Pi_{j,i}^t$为第 t 次认证密钥协商协议运行中的两个参与者，其中 i、j 分别是身份标识为 ID_i和 ID_j的两个参与者的索引标号. $\mathcal{S}$的构造如下：

(1) 系统建立

$\mathcal{S}$选取随机数 $x\leftarrow_R Z_q^*$ 作为系统主密钥，即 $msk=x$，计算系统主公钥 $P_{Pub}=xP$，公开参数 $Params=\{p,q,E/F_q,P,G,H_1,H_2,H_3,P_{Pub},(\mathcal{E},\mathcal{D})\}$给敌手$\mathcal{A}^1$，其中$(\mathcal{E},\mathcal{D})$是对称密码机制的加解密算法；同时，$\mathcal{S}$维持四个初始为空的列表$\mathcal{L}_p$、$\mathcal{L}_{sk}$、$\mathcal{L}_{pk}$和$\mathcal{L}_s$，分别用于跟踪$\mathcal{A}^1$所执行的部分密钥生成、私钥生成、公钥生成和 *Send* 询问.

部分密钥生成询问. 收到$\mathcal{A}^1$关于(id_i,R_i')的部分密钥生成询问，若$(id_i,s_i,R_i)\in\mathcal{L}_p$，则$\mathcal{S}$返回$(s_i,R_i)$给$\mathcal{A}^1$；否则，$\mathcal{S}$选取随机值 $r_i\leftarrow_R Z_q^*$，并计算 $R_i=r_iP$ 和 $s_i=r_i+xh_i$，其中 $h_i=H_1(id_i,R_i,R_i')$；返回(s_i,R_i)给$\mathcal{A}^1$的同时在列表$\mathcal{L}_p$中添加相应的元组(id_i,s_i,R_i). 特别地，当 $i=J$ 时，$\mathcal{S}$输出特殊的符号$\perp$并终止.

私钥生成询问. 收到$\mathcal{A}^1$关于 id_i的私钥生成询问，若$(id_i,s_i,r_i')\in\mathcal{L}_{sk}$，则$\mathcal{S}$返回 $sk_i=(s_i,r_i')$给$\mathcal{A}^1$；否则，$\mathcal{S}$选取随机值 $r_i'\leftarrow_R Z_q^*$，并计算 $R_i'=r_i'P$；然后执行关于(id_i,R_i')的部分密钥生成询问，并获得相应的应答(s_i,R_i)，返回$sk_i=(s_i,r_i')$给$\mathcal{A}^1$的同时在列表$\mathcal{L}_{sk}$中添加相应的元组(id_i,s_i,r_i')；并且在$\mathcal{L}_{pk}$中添加相应的元组(id_i,R_i,R_i'). 特别地，当 $i=J$ 时，$\mathcal{S}$输出特殊的符号$\perp$并终止. 此外，私钥生成询问中会涉及部分密钥生成询问，因此该游戏中$\mathcal{A}^1$对部分密钥生成的询问次数依然是 q_k次.

公钥生成询问. 收到$\mathcal{A}^1$关于 id_i的公钥生成询问，若$(id_i,R_i,R_i')\in\mathcal{L}_{pk}$，则$\mathcal{S}$返回 $pk_i=(R_i,R_i')$给$\mathcal{A}^1$；否则，$\mathcal{S}$执行关于id_i的私钥生成询问(在该询问中id_i所对应的公钥信息$pk_i=(R_i,R_i')$将被添加到列表$\mathcal{L}_{pk}$中)，返回 $pk_i=(R_i,R_i')$给$\mathcal{A}^1$.

Send$(\Pi_{i,j}^t,M)$：列表$\mathcal{L}_s$的格式为$(\Pi_{i,j}^t,M_i,M_i',a,k)$，其中 M_i和M_i'分别表示参与者$\Pi_{i,j}^t$收到和产生的消息，$a\in Z_q^*$是$\mathcal{S}$为$\Pi_{i,j}^t$选取的随机数，k 是相应的协商会话密钥，且初始时 k 被设置为$\perp$，表示该值为空；$\mathcal{L}_s$可能会在 *Reveal* 询问中被更新. 当$\mathcal{S}$收到消息 M 时，进行如下操作：

① 如果 $\Pi_{i,j}^t$已经存在，且为会话的发起者，则设 M 为$\Pi_{i,j}^t$接收到的消息并接受会话；否则，$\Pi_{i,j}^t$不存在，则进行公钥生成询问和私钥生成询问为索引为 i 的参与者生成对应的公私钥对.

② 如果 $M=\kappa$，则 $\Pi_{i,j}^t$是相应会话的发起者；否则，$\Pi_{i,j}^t$是相应会话的响应者；此外将消息 M 作为 $\Pi_{i,j}^t$的输入，同时设置 $\Pi_{i,j}^t$接受该会话.

③ 若 $i=J$；$\mathcal{S}$随机选取 $\delta_i\leftarrow Z_q^*$，并令 $R_1=aP$(隐含设置 $r_1=a$)，然后生成初始化消息 $M=\{id_i,m_i,R_1,\delta_i\}$，更新$\mathcal{L}_s$并返回 M.

否则，$\mathcal{S}$随机选择 $r_1\in Z_q^*$ 作为索引为 i 的参与者的随机秘密数，并生成初始化消息 $M=\{id_i,m_i,R_1=r_1P,\delta_i=h_ir_i'+s_i+r_1\}$，其中 $h_i=H_2(id_i,m_T,R,R_i,R_i')$，更新$\mathcal{L}_s$并返回 M.

Corrupt(i)：若$(id_i,s_i,r_i')\in\mathcal{L}_{sk}$，则$\mathcal{S}$返回 $sk_i=(s_i,r_i')$给$\mathcal{A}^1$；否则，$\mathcal{S}$执行关于 id_i的私钥生成询问，返回 $sk_i=(s_i,r_i')$给$\mathcal{A}^1$.

Reveal$(\Pi_{i,j}^t)$：$\mathcal{S}$从$\mathcal{L}_s$中查找相应的参与者 $\Pi_{i,j}^t$，若$(\Pi_{i,j}^t,M_i,M_i',a,k)\notin\mathcal{L}_s$，或者 $\Pi_{i,j}^t$未接受会话，则返回$\perp$；否则有$(\Pi_{i,j}^t,M_i,M_i',a,k)\in\mathcal{L}_s$，$\mathcal{S}$进行下述操作：

① 若 $k\neq\perp$，则返回对应的 k.

② 否则，若 $i=J$，$\mathcal{S}$随机选取 $\delta_j\leftarrow_R Z_q^*$，并令 $R_2=bP$(隐含设置 $r_2=b$)，然后生成相应的消息 $M=\{id_j,m_j,R_2,\delta_j\}$，计算 $k=H_2(T_\mu+(r_i'+s_i)(R_j+R_j'+h_jP_{Pub}))$，并更新列表$\mathcal{L}_s$并返回 k.

若 $i\neq J$，$\mathcal{S}$随机选取 $r_2\leftarrow_R Z_q^*$，并令 $R_2=bP$，生成相应的消息 $M=\{id_j,m_j,R_2,\delta_j=h_jr_j'+s_j+r_2\}$，其中 $h_j=H_2(id_j,m_j,R_2,R_j,R_j')$. 然后计算 $k=H_2(r_2R_1+(r_i'+s_i)(R_j+R_j'+h_jP_{Pub}))$，并更新列表$\mathcal{L}_s$并返回 k.

Test$(\Pi_{i,j}^t)$：当敌手$\mathcal{A}^1$决定结束询问阶段时，假设$\mathcal{A}^1$完全遵守 *eCK* 安全模型，则$\mathcal{A}^1$选择一个新鲜的参与者 $\Pi_{i,j}^t$发起 *Test* 询问. $\mathcal{S}$收到询问请求后，执行下述操作：

① 如果 $t\neq j$，或者 $t=j$ 但存在会话标识与$\Pi_{i,j}^t$相同的会话，并且该会话已经被打开，则$\mathcal{S}$终止；

② 否则，$\mathcal{S}$查找列表$\mathcal{L}_s$获取 $\Pi_{i,j}^t$对应的 k_β，并输出给$\mathcal{A}^1$.

猜　测. 敌手$\mathcal{A}^1$询问完成后，对 β的猜测值 β'. 若有 $\beta'=\beta$，则$\mathcal{S}$返回 $\mu=1$，表示 $T_\mu=abP$；否则返回 $\mu=0$，表示 $T_\mu=cP$.

若敌手$\mathcal{A}^1$攻击本文认证密钥协商协议成功，则$\mathcal{S}$能够解决输出 DDH 困难问题的有效解；否则，$\mathcal{S}$无法输出它的有效解.

(a) 上述模拟过程中，若$\mathcal{S}$未终止，那么模拟游戏与真实游戏对于敌手$\mathcal{A}^1$而言是不可区分的.

对于敌手$\mathcal{A}^1$而言，在模拟游戏中所接收到的参数均与真实游戏中的相应参数具有相同的分布. 因此，$\mathcal{A}^1$无法区分是真实攻击还是模拟攻击.

(b) 若$\mathcal{A}^1$能以不可忽略的优势 $\varepsilon_1(\kappa)$攻击本文密钥协商协议成功，则$\mathcal{S}$至少能以不可忽略的优势 $\left(\frac{1}{q_k}+\frac{1}{q_C}\right)\frac{\varepsilon_1(\kappa)}{\mathrm{e}q^2}$解决 DDH 困难问题，其中 e 是自然对数底数.

令$\mathcal{E}_1$表示$\mathcal{A}^1$进行私钥生成询问时$\mathcal{S}$未终止；$\mathcal{E}_2$表示 *Send* 询问时$\mathcal{S}$生成了正确的消息 M；$\mathcal{E}_3$表示 *Corrupt* 询问时$\mathcal{S}$生成了正确的消息 M；$\mathcal{E}_4$表示$\mathcal{A}^1$进行 Test 询问时$\mathcal{S}$未终止. 则有 $\Pr[\mathcal{E}_1]=(1-\rho)^{q_k}$；$\Pr[\mathcal{E}_2]=\frac{1}{q}$；$\Pr[\mathcal{E}_3]=\frac{1}{q}$；$\Pr[\mathcal{E}_4]=\rho+\frac{1}{q_C}$.

当$\mathcal{A}^1$以不可忽略的优势 $\varepsilon(\kappa)$成功攻击本文密钥协商协议时，$\mathcal{S}$输出解决 DDH 困难问题的优势为

$$\begin{aligned}\Pr[\mathcal{S}\text{ wins}]&=\Pr[\mathcal{E}_1]\Pr[\mathcal{E}_2]\Pr[\mathcal{E}_3]\Pr[\mathcal{E}_4]\Pr[\mathcal{A}^1\text{wins}]\\&=(1-\rho)^{q_k}\frac{1}{q^2}\left(\rho+\frac{1}{q_C}\right)\varepsilon_1(\kappa)\end{aligned}$$

$$=\left(1-\frac{1}{q_k}\right)^{q_k}\frac{1}{q^2}\left(\frac{1}{q_k}+\frac{1}{q_C}\right)\varepsilon_1(\kappa)$$
$$\geq\left(\frac{1}{q_k}+\frac{1}{q_C}\right)\frac{\varepsilon_1(\kappa)}{\mathrm{e}q^2}.$$

由于$\rho=\frac{1}{q_k}$，则q_k足够大时$\left(1-\frac{1}{q_k}\right)^{q_k}$趋向于$\mathrm{e}^{-1}$（e是自然对数底数），因此，当$\mathcal{A}^1$以不可忽略的优势$\varepsilon(\kappa)$成功攻击本文协议时，$\mathcal{S}$赢得上述游戏的优势至少为$\left(\frac{1}{q_k}+\frac{1}{q_C}\right)\frac{\varepsilon_1(\kappa)}{\mathrm{e}q^2}$. 综上所述，若$\mathcal{S}$在模拟过程中未终止，且$\mathcal{A}^1$攻破本文协议的优势是$\varepsilon_1(\kappa)$，那么$\mathcal{S}$将以优势$\left(\frac{1}{q_k}+\frac{1}{q_C}\right)\frac{\varepsilon_1(\kappa)}{\mathrm{e}q^2}$输出DDH困难问题的有效解.

附录 D. 定理2的形式化证明.

证明. 假设$\mathcal{S}$一个DL困难问题的解决者，其困难问题的输入为(P,aP)，其中$a\in Z_q^*$且未知，DL困难问题的目标是计算出a.

初始化. 模拟器$\mathcal{S}$令$P_{Pub}=aP$（隐含地设置主私钥为a，但对于$\mathcal{S}$而言a是未知的参数），随机选取安全的单向哈希函数$H_2:\{0,1\}^*\rightarrow Z_q^*$，并输出$Params=\{p,q,E/F_q,P,G,H_1,H_2,P_{Pub},(Enc,Dec)\}$给$\mathcal{A}^1$，其中$(Enc,Dec)$是对称密码机制的加解密算法. 此外，维持初始为空的列表$\mathcal{L}_c$和$\mathcal{L}_1$分别用于跟踪$\mathcal{A}^1$对用户公钥生成和谕言机H_1的询问信息，其中$\mathcal{L}_1$的元组格式为(id_i,R_i,R_i',h_i^1)，$\mathcal{L}_c$的元组格式为(id_i,R_i,R_i',s_i,r_i'). 此外，$\mathcal{C}$基于列表$\mathcal{L}_{\mathrm{Sign}}$完成对敌手$\mathcal{A}^1$所提交的密钥协商询问的跟踪.

询问阶段. 该阶段敌手$\mathcal{A}^1$能够适应性的对下述询问进行多项式时间次的询问. 在伪造阶段之前，挑战者$\mathcal{C}$无法确定敌手$\mathcal{A}^1$的挑战身份，$\mathcal{S}$在$\mathcal{A}^1$的询问过程中适应性地猜测一个挑战身份ID'，由于$\mathcal{A}^1$共提交了q_d+q_s+1个不同的身份，则$\mathcal{S}$能以概率$\frac{1}{q_d+q_s+1}$猜中$\mathcal{A}^1$的挑战身份.

(1) 公钥生成询问. 当$\mathcal{S}$收到$\mathcal{A}^1$对用户id_i的公生成钥询问时，若$(id_i,R_i,R_i',s_i,r_i')\in\mathcal{L}_c$，则返回$pk_i=(R_i,R_i')$给$\mathcal{A}^1$；否则，随机选取$r_i'\in Z_q^*$，$h_i^1\in Z_q^*$和$s_i\in Z_q^*$，并计算$R_i'=r'P$和$R_i=s_iP-P_{pub}h_i^1$，分别在列表$\mathcal{L}_c$和$\mathcal{L}_1$中添加元组$(id_i,R_i,R'_i,s_i,r_i')$和$(id_i,R_i,R_i',h_i^1)$，并返回相应的$pk_i=(R_i,R_i')$给$\mathcal{A}^1$.

(2) 谕言机H_1询问. 当$\mathcal{S}$收到$\mathcal{A}^1$对id_i的谕言机H_1询问时，若$(id_i,R_i,R_i',h_i^1)\in\mathcal{L}_1$，返回相应的$h_i^1$给$\mathcal{A}^1$；否则，则$\mathcal{S}$对$id_i$执行公钥生成询问算法（在该询问中相应的元组$(id_i,R_i,R_i',s_i,r_i')$被添加到列表$\mathcal{L}_1$中），然后搜索$\mathcal{L}_1$并返回相应的$h_i^1$给$\mathcal{A}^1$.

(3) 秘密值询问. 当$\mathcal{S}$收到$\mathcal{A}^1$对id_i的秘密值询问时，若$(id_i,R_i,R_i',s_i,r_i')\in\mathcal{L}_c$，则返回$r_i'$给$\mathcal{A}^1$；否则对$id_i$执行公钥生成询问后，搜索$\mathcal{L}_c$并返回相应的$r_i'$给$\mathcal{A}^1$.

(4) 部分密钥生成询问. 当$\mathcal{S}$收到$\mathcal{A}^1$对id_i的部分密钥生成询问时，若$id_i=id'$，则$\mathcal{S}$终止. 若$(id_i,R_i,R_i',s_i,r_i')\in\mathcal{L}_c$，返回$(R_i,s_i)$给$\mathcal{A}^1$；否则对$id_i$执行公钥生成询问后，返回$(R_i,s_i)$给$\mathcal{A}^1$.

(5) 公钥替换询问. 当$\mathcal{S}$收到$\mathcal{A}^1$对id_i的公钥替换询问时，若$id_i=id'$，则$\mathcal{C}$忽略本次询问；否则，$\mathcal{C}$将id_i的公钥$pk_i=(R_i,R_i')$替换为$\mathcal{A}^1$所提交的$pk_i'=(\bar{R}_i,\bar{R}_i')$.

(6) 私钥生成询问. 当$\mathcal{S}$收到$\mathcal{A}^1$对id_i的私钥生成询问时，若$id_i=id'$，则$\mathcal{C}$终止；否则，$\mathcal{C}$执行下述操作：

若$(id_i,R_i,R_i',s_i,r_i')\in\mathcal{L}_c$，则返回$sk_i=(s_i,r_i')$给$\mathcal{A}^1$；否则，对$id_i$执行公钥生成询问后，返回相应的$sk_i=(s_i,r_i')$给$\mathcal{A}^1$.

(7) 密钥协商消息询问. 当$\mathcal{S}$收到$\mathcal{A}^1$关于身份id_i和消息m_i的密钥协商消息询问时，$\mathcal{S}$随机选取$r_1\leftarrow_R Z_q^*$，计算$R_1=r_1P$和$\delta_i=h_ir_i'+s_i+r_1$，其中$h_i=H_2(id_i,m_i,R_1,R_i,R_i')$，将相应的结果$\{m_i,R_1,\delta_i\}$返回给$\mathcal{A}^1$，同时将$(id_i,m_i)$记录到列表$\mathcal{L}_{\mathrm{Sign}}$中. 特别地，若$id_i$所对应的公钥$pk_i$被替换，由于$\mathcal{S}$拥有私钥$sk_i$，相应的密钥协商消息依然能够生成.

伪 造. $\mathcal{A}^1$输出关于身份id^*和消息m^*的伪造密钥协商消息$\{m^*,R^*,\delta^*\}$，若$id^*\neq id'$或$(id^*,m^*)\in\mathcal{L}_{\mathrm{Sign}}$，则$\mathcal{S}$终止；否则，由于$\mathcal{S}$无法利用$\mathcal{A}^1$的一次成功伪造解决困难问题，根据分叉引理[13-14]可知，$\mathcal{S}$通过更改谕言机H_1的输出$\tilde{h}_{id^*}^{1*}$，使得敌手$\mathcal{A}^1$基于相同的随机数r_1^*（即$R_1^*=r_1^*P$）生成了一个新的伪造密钥协商消息$\{m^*,R^*,\delta'\}$，由于上述密钥协商消息都是有效的，因此有下述等式成立.

$$\begin{cases}\delta^*=r_i'^*h_{id^*}^{2*}+r_i^*+ah_{id^*}^{1*}+r_1^*\\ \delta'^*=r_i'^*h_{id^*}^{2*}+r_i^*+a\tilde{h}_{id^*}^{1*}+r_1^*\end{cases}.$$

基于上述等式，$\mathcal{S}$能求得$a=\frac{\delta^*-\delta'}{h_{id^*}^{1*}-\tilde{h}_{id^*}^{1*}}$，那么$\mathcal{S}$利用$\mathcal{A}^1$作为子程序成功地解决了离散对数问题的困难性.

令$\mathcal{E}_1$表示$\mathcal{A}^1$在询问阶段不终止，$\mathcal{E}_2$表示$\mathcal{A}^1$在伪造阶段不终止，则$\mathcal{A}^1$在询问训练阶段和伪造阶段不终止的概率分别为

$$\Pr(\mathcal{E}_1)\geq\left(1-\frac{1}{q_d+q_s+1}\right)^{q_d+q_s}\text{和}\Pr(\mathcal{E}_2)=\frac{1}{q_d+q_s+1}.$$

令$\mathcal{E}_3$表示$\mathcal{A}^1$成功输出两个有效的伪造密钥协商$\{m^*,R^*,\delta^*\}$和$\{m^*,R^*,\delta'\}$. 由于$\mathcal{A}^1$输出一个有效签名的概率为$\varepsilon_3(\kappa)$，则根据分叉引理可知，$\mathcal{A}^1$成功输出上述两个有效签名的概率为$\Pr(\mathcal{E}_3)\geq\left(1-\frac{1}{\mathrm{e}}\right)\frac{\varepsilon_1(\kappa)}{q_{H_1}}$. 则有

$$\Pr(\mathcal{E}_1\wedge\mathcal{E}_2\wedge\mathcal{E}_3)\geq\left(1-\frac{1}{q_d+q_s+1}\right)^{q_d+q_s}\frac{1}{q_d+q_s+1}\left(1-\frac{1}{\mathrm{e}}\right)\frac{\varepsilon_1(\kappa)}{q_{H_1}}$$
$$\geq\left(1-\frac{1}{\mathrm{e}}\right)\frac{\varepsilon_3(\kappa)}{\mathrm{e}(q_d+q_s+1)q_{H_1}}.$$

因此，若$\varepsilon_1(\kappa)$是不可忽略的，那么$\mathcal{S}$至少能以显而易见的优势$\left(1-\frac{1}{\mathrm{e}}\right)\frac{\varepsilon_3(\kappa)}{\mathrm{e}(q_d+q_s+1)q_{H_1}}$成功解决DL问题.

附录 E.　定理 3 的形式化证明.

证明.　假设$\mathcal{S}$是解决 DL 问题的一个敌手，其输入为(P,aP)，目标是求解未知的随机数 a. $\mathcal{S}$与敌手$\mathcal{A}^2$间的消息交互过程如下所述：

初始化. 模拟器$\mathcal{S}$选取随机数$x\leftarrow_R Z_q^*$作为系统主密钥，即$msk=x$，计算系统主公钥$P_{Pub}=xP$，随机选取$H_1:\{0,1\}^*\rightarrow Z_q^*$，并输出$Params=\{p,q,E/F_q,P,G,H_1,H_2,P_{Pub},(\mathcal{E},\mathcal{D})\}$给$\mathcal{A}^2$，其中$(\mathcal{E},\mathcal{D})$是对称密码机制的加解密算法. 此外，维持初始为空的$\mathcal{L}_c$和$\mathcal{L}_2$分别用于跟踪$\mathcal{A}^2$对用户公钥生成和谕言机 H_2的询问信息，其中$\mathcal{L}_2$的元组格式为$(id_i,m_i,R_i,R_i',h_i^2)$，$\mathcal{L}_c$的元组格式为$(id_i,R_i,R_i',s_i,r_i')$. 此外，$\mathcal{C}$基于列表$\mathcal{L}_{\text{Sign}}$完成对敌手$\mathcal{A}^2$签名询问的跟踪.

询问阶段. 该阶段敌手$\mathcal{A}^2$能够适应性的对下述询问进行多项式时间次的询问. 在伪造阶段之前，挑战者$\mathcal{S}$无法确定敌手$\mathcal{A}^2$的挑战身份，$\mathcal{S}$在$\mathcal{A}^2$的询问过程中适应性地猜测一个挑战身份 id'，由于$\mathcal{A}^2$共提交了q_s+q_v+1个不同的身份，则$\mathcal{S}$能以概率$\frac{1}{q_s+q_v+1}$猜中敌手$\mathcal{A}^2$的挑战身份.

(1) 公钥生成询问. 当$\mathcal{S}$收到$\mathcal{A}^2$对id_i的公钥生成询问时，若$(id_i,R_i,R_i',s_i,r_i')\in\mathcal{L}_c$，则返回 $pk_i=(R_i,R_i')$给$\mathcal{A}^2$，否则进行下述计算：

① 若 $id_i\neq id'$，随机选择 $r_i'\in Z_q^*$ 和 $r_i\in Z_q^*$，并计算 $R_i=r_iP$，$R_i'=r_i'P$ 和 $s_i=r_i+xH_1(id_i,R_i,R_i')$，并在$\mathcal{L}_c$中添加$(id_i,R_i,R_i',s_i,r_i')$，然后返回 $pk_i=(R_i,R_i')$给$\mathcal{A}^2$；

② 若 $id_i=id'$，令 $R_i'=aP$(隐含设定 $r_i'=a$)，随机选择$r_i\in Z_q^*$，并计算 $R_i'=r_i'P$ 和 $s_i=r_i+xH_1(id_i,R_i,R_i')$，然后在$\mathcal{L}_c$中添加$(id_i,R_i,R_i',s_i,\perp)$，并返回 $pk_i=(R_i,R_i')$给$\mathcal{A}^2$，其中，$\perp$表示$\mathcal{A}^2$不知道相应的值.

(2) 秘密值询问. 当$\mathcal{S}$收到$\mathcal{A}^2$对 id_i的秘密值询问时，若$id_i=id'$，则$\mathcal{S}$终止. 若$(id_i,R_i,R_i',s_i,r_i')\in\mathcal{L}_c$，返回相应的 r_i'给$\mathcal{A}^2$，若不存在，则执行公钥生成询问后，返回$\mathcal{L}_c$中相应的 r_i'给$\mathcal{A}^2$.

(3) H_2询问. 当$\mathcal{S}$收到$\mathcal{A}^2$关于 id_i的谕言机 H_2询问时，若$(id_i,m_i,R_i,R_i,R_i',h_i^2)\in\mathcal{L}_2$，返回相应的 h_i^2给$\mathcal{A}^2$；若不存在，则$\mathcal{S}$随机选择 $h_i^2\in Z_q^*$，在列表$\mathcal{L}_2$中添加相应的元组$(id_i,m_i,R_i,R_i,R_i',h_i^2)$，并返回相应的 h_i^2给$\mathcal{A}^2$.

私钥生成询问和签名生成询问与定理 2 相类似，此处不再进行赘述.

密钥协商消息伪造. $\mathcal{A}^2$输出一个关于身份 id^* 和消息 m^* 伪造协商消息$\{m^*,R^*,\delta^*\}$，若 $id^*\neq id'$或$(id^*,m^*)\in\mathcal{L}_{\text{Sign}}$，则$\mathcal{S}$终止；否则，由于$\mathcal{S}$无法利用$\mathcal{A}^2$的一次成功伪造解决困难问题，根据分叉引理[13-14]可知，$\mathcal{S}$通过更改谕言机 H_2的输出$\tilde{h}_{id^*}^2$，使得$\mathcal{A}^2$基于相同的随机数 r_1^*(即 $R_1^*=r_1^*P$)生成了一个新的伪造密钥协商消息$\{m^*,R^*,\delta'\}$ 由于上述两个密钥协商消息都是有效的，因此有下述等式成立.

$$\begin{cases}\delta^*=ah_{id^*}^2+r_i^*+xh_{id^*}^1+r_1^*\\ \delta'=a\tilde{h}_{id^*}^2+r_i^*+xh_{id^*}^1+r_1^*\end{cases}.$$

基于上述等式$\mathcal{C}$能求得 $a=\frac{\delta^*-\delta'}{h_{id^*}^2-\tilde{h}_{id^*}^2}$，那么$\mathcal{S}$利用$\mathcal{A}^2$作为子程序成功地解决了 DL 问题的困难性.

令$\mathcal{F}_1$表示$\mathcal{A}^2$在询问训练阶段不终止，$\mathcal{F}_2$表示$\mathcal{A}^2$在消息伪造阶段不终止，则$\mathcal{A}^2$在询问训练阶段和伪造阶段不终止的概率分别为

$$\Pr(\mathcal{F}_1)\geq\left(1-\frac{1}{q_s+q_v+1}\right)^{q_s+q_v}\text{和}$$

$$\Pr(\mathcal{F}_2)=\frac{1}{q_s+q_v+1}.$$

令$\mathcal{F}_3$表示$\mathcal{A}^2$成功输出两个有效签名$\{m^*,R^*,\delta^*\}$和$\{m^*,R^*,\delta'\}$. 由于$\mathcal{A}^2$输出一个有效签名的概率为 $\varepsilon_4(\kappa)$，则根据分叉引理可知，$\mathcal{A}^2$成功输出上述两个有效签名$\{m^*,R^*,\delta^*\}$和$\{m^*,R^*,\delta'\}$的概率为 $\Pr(\mathcal{F}_3)\geq\left(1-\frac{1}{\mathrm{e}}\right)\frac{\varepsilon_4(\kappa)}{q_{H_2}}$. 则有

$$\begin{aligned}&\Pr(\mathcal{F}_1\wedge\mathcal{F}_2\wedge\mathcal{F}_3)\\&\geq\left(1-\frac{1}{q_s+q_v+1}\right)^{q_s+q_v}\frac{1}{q_s+q_v+1}\left(1-\frac{1}{\mathrm{e}}\right)\frac{\varepsilon_4(\kappa)}{q_{H_2}}\\&\geq\left(1-\frac{1}{\mathrm{e}}\right)\frac{\varepsilon_4(\kappa)}{\mathrm{e}(q_s+q_v+1)q_{H_2}}.\end{aligned}$$

因此，若 $\varepsilon_4(\kappa)$是不可忽略的，那么$\mathcal{S}$至少能以显而易见的优势 $\left(1-\frac{1}{\mathrm{e}}\right)\frac{\varepsilon_4(\kappa)}{\mathrm{e}(q_s+q_v+1)q_{H_2}}$ 成功解决 DL 问题的困难性.

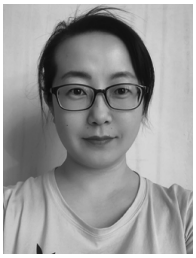
附录 F.　分叉引理使用方法总结.

分叉引理推进了签名机制的安全性证明进程，本文在这里对使用分叉引理证明签名的不可伪造性方法进行简要总结. 通常情况下签名机制均为随机化的算法，因此相应的签名均基于选取的随机数生成，因此在挑战阶段敌手所提交的伪造签名中包含了挑战者未知的随机数；此外，由密码学安全性证明方法可知，挑战者会将他/她所面临的困难性问题嵌入到跟伪造签名相关的询问中，在这种情况下，敌手的伪造签名中肯定包含困难问题的解. 然而，签名机制通常基于计算的困难性假设进行证明(如 DL 问题、计算性 Diffie-Hellman 问题等)，那么对于挑战者而言，签名机制中会涉及至少两个未知量，一个是敌手选择用来生成伪造签名的随机数，一个是挑战者所面临的困难问题，因此挑战者无法通过敌手的一次成功伪造解决相应的困难性问题，那么挑战者将利用控制随机谕言机输出的方法使得敌手基于相同的随机数和不同的谕言机应答生成一个新的伪造签名(敌手输出两个伪造签名的概率值能由分叉引理获知)，基于这两个

伪造签名，挑战者就能将敌手选择的随机数消除，未知量就只有相应的困难问题的解，那么挑战者就能够通过计算求解出相应的困难问题的解，达到解决困难问题的目的.在随机谕言机可重放的前提下，分叉引理给出了敌手基于相同随机数和不同谕言机的应答生成两个有效伪造签名的概率关系.

文献[22-24]中详细介绍了分叉引理的具体内容，指出若存在敌手能以一定的优势伪造一个签名，那么在随机谕言机的协助下分叉引理推导出敌手成功伪造满足相应条件的两个合法签名的概率，具体可表示如下：

对于一个形式为(m,δ,r,h)的签名，其中m是待签名消息，δ是相应的签名值，r和h分别为生成签名所使用的随机数和哈希值.若存在概率多项式时间敌手$\mathcal{A}$能以优势$\varepsilon(\kappa)$伪造一个有效签名，且对随机谕言机H进行了q_H次询问，那么基于谕言机H的重放操作，$\mathcal{A}$使用相同的随机数r^*和不同的谕言机应答$h^*,h'(h^*\neq h')$输出两个有效签名组(m^*,δ^*,r^*,h^*)和(m^*,δ',r^*,h')的概率至少为$\left(1-\frac{1}{\mathrm{e}}\right)\frac{\varepsilon(\kappa)}{q_H}$，其中e是自然对数底数.特别地，上述交互中挑战者获得了两个合法签名δ^*和δ'.



QIAO Zi-Rui, Ph. D. candidate. Her research interests focus on information security.

YANG Qi-Liang, Ph. D. His research interests include information security and cryptography.

ZHOU Yan-Wei, Ph. D., associate professor, M. S. supervisor. His research interests include leakage-resilient cryptography and anonymous communication.

YANG Bo, Ph. D., professor, Ph. D. supervisor. His current research interests include information security and cryptography.

GU Chun-Xiang, Ph. D., professor, Ph. D. supervisor. His current research interest is information security.

ZHANG Ming-Wu, Ph. D., professor, Ph. D. supervisor. His current research interest is information security.

XIA Zhe, Ph. D., associate professor. His research interest is information security.

Background

Certificateless public-key cryptography (CL-PKC) does not require management of certificate and has no key escrow problem. Due to these advantages, certificateless signature scheme is widely used in the design of authentication key agreement (AKA) protocol. For CL-PKC, which will face two kinds of attacks, one is from the outside attacks (it is a malicious user), and one is from the inside attacks (it is a malicious key generation center(KGC)). Similarly, the AKA protocol created with CL-PKC faces two kinds of adversaries, one is Type I adversary, which is a malicious user, and can replace the public key of any user but not know the master secret key; the other is Type II adversary, which is a malicious KGC, and has obtained the master secret key but cannot replace the public key of any user. Especially, the Type I adversary cannot replace the corresponding public key of the challenge identity before the challenge stage, and Type II adversary does not need to perform the partial private key generation query because he/she can calculate it himself.

In this paper, to obtain an efficient AKA protocol with provable security for VANET, a concrete construction is created with CL-PKC, and the security of our proposal is proved based on the hardness of discrete logarithm problem by using Forking lemma under the random oracle. Compared with the previous AKA protocols, we have that our protocol has stronger security while the high computational efficiency is provided. Furthermore, for the previous construction, some attacks are proposed, which shows that the above constructions are insecure for any Type I adversary. Moreover, the main reason for the above security flaws are analyzed in detail, and our analyses can help the researchers to avoid the above security flaws in future studies.

This work was supported by the National Key R&D Program of China (No. 2017YFB0802000), the National Natural Science Foundation of China(Nos. 62272287, 61802242, U2001205), the Research Funds of Guangxi Key Laboratory of Cryptography and Information Security (No. GCIS202108), the Research Funds of Henan Key Laboratory of Network Cryptography Technology (No. LNCT2021-A04), and the Fundamental Research Funds for the Central Universities.