eCK模型下的密钥协商

柳秀梅[1] 高克宁[1] 薛丽芳[1] 常桂然[1] 周福才[2]
(东北大学计算中心 沈阳 110819)[1] (东北大学软件学院 沈阳 110819)[2]

摘 要 如何构造安全的密钥协商协议是信息安全领域富有挑战性的问题之一。目前安全协议只能达到"启发式"安全,协议的安全假设也不够理想。针对这一问题,提出了基于计算性假设(CDH)的三方认证密钥协商协议,并运用陷门测试定理形式化地证明该协议在eCK模型下是安全的,更好地支持了敌手的询问。

关键词 认证密钥协商,eCK模型,CDH假设,形式化证明
中图法分类号 TP393.08 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2014.08.038

Authenticated Key Exchange in eCK Model

LIU Xiu-mei[1] GAO Ke-ning[1] XUE Li-fang[1] CHANG Gui-ran[1] ZHOU Fu-cai[2]
(Computing Center, Northeastern University, Shenyang 110819, China)[1]
(Software College, Northeastern University, Shenyang 110819, China)[2]

Abstract How to construct the security of key agreement protocol is one of the challenging problems in information security field. Most security protocols can only reach the "heuristic" security, and the assumption of security protocol is not ideal. To solve these problems, this paper presented a new computational assumptions (CDH) based third-party authentication key exchange protocol, and by using the trapdoor test theory, formally proved that the protocol is safe under the eCK model and better supports the adversary inquiries.

Keywords Authenticated key exchange, eCK model, CDH assumption, Formal proof

1 引言

在开放的网络环境下,如何确保会话双方安全通信一直是信息安全领域重要的研究方向。密钥协商协议可以帮助利用不安全信道(如互联网)通信的会话双方协商出一个共同的会话密钥,该密钥可以为后续通信提供保密/认证或完整性等安全服务。近年来,学者们提出了大量的可满足不同安全性要求的密钥协商协议,然而,由于没有一种有效的机制确保这类协议不会在未来被新的攻击方式攻破,早期的密钥协商协议只能达到"启发式"安全性:一旦某协议被新的攻击方式攻击成功,该协议或者需要修改,或者被彻底摒弃。经过一系列的攻击与修改后,那些被认为可以抵抗应用环境下所有已知攻击的方案才可能用于实践,这种方式的弊端是显而易见的:协议使用者很难确认协议的安全性,反反复复的修补更增加了人们对安全性的担心,也增加了实现代价或成本。因此,设计可以准确刻画参与者或敌手特有的形式安全模型,并在该模型下讨论协议的安全性有着重要的意义。

Bellare和Rogaway[1]在1993年为AKE协议提出第一个正式的安全模型,许多正式的安全模型被建议用于证明AKE协议的安全性。这些模型中最著名的一个是CK模型[2](Canetti-Krawczyk),它是在2001年由Canetti和Krawczyk提出的。但是CK模型不能捕获密钥伪装攻击(Key Compromise Impersonation,KCI)或临时私钥的泄漏,所以一个在CK模型下被证明是安全的AKE协议可能仍旧会存在一些问题。为了包含这些攻击,在2007年,LaMacchia、Lauter和Mityagin[3]提出了一个eCK模型(extended Canetti-Krawczyk),其被认为是当前最强的安全模型。在近几年,两方的AKE协议[4]已经在考虑不同敌手能力的各种模型下接受了严格的分析,同时,国内的学者也在AKE协议及安全协议等方面进行了非常深入的研究,并取得了较好的研究成果[5-8]。然而,对三方AKE协议还没有像两方协议那样进行广泛的分析。

本文提出一个三方的基于计算性假设(Computational Diffie-Hellman (CDH))并在eCK模型下安全的密钥交换协议。CDH假设比GDH(Gap Diffie-Hellman)假设更加标准[9],因此所提出的协议也更标准。更近一步地,我们的协议给出了更强的安全性定义,因此更好地支持了敌手的询问,安全性也更强。

2 背景知识

eCK模型是2008年由Yoneyama[10]提出的,它为AKE协议提供了一种标准的安全保证。eCK模型是目前认为较强

到稿日期:2013-10-24 返修日期:2014-01-12 本文受国家自然科学基金资助项目(61272176),中央高校基本科研业务费专项资金资助项目(N100316001,N120416002),教育部高新技术类资助项目(MOE-INTEL-2012-06)资助。

柳秀梅(1976—),女,博士,副教授,主要研究方向为信息安全,E-mail:liuxm@cc.neu.edu.cn;高克宁(1963—),女,博士,教授,主要研究方向为社交网络、Web信息技术等;薛丽芳(1977—),女,博士,讲师,主要研究方向为模式识别、信息安全等;常桂然(1946—),男,博士,教授,博士生导师,主要研究方向为计算机网络、多媒体技术、信息安全等;周福才(1964—),男,博士,教授,博士生导师,主要研究方向为信息安全、计算机网络等。

的安全模型。敌手与协议参与者通过预言询问(Query)交互,这些询问刻画了敌手在真实攻击中的能力。本节首先介绍协议所基于的计算性假设难题[11],然后对证明协议安全性所涉及的 eCK 模型安全属性进行描述。

2.1 数学基础

(1)计算 Diffie-Hellman 难题(CDH 难题)

假设 $g\in F_p^*$ 是有限域 F_p^* 的一个生成元;已知 $g^a, g^b\in F_p^*$,其中整数 $0<a,b<p$。则输出 $g^{ab} \bmod p$ 是计算 Diffie-Hellman 难题。

(2)陷门测试定理(Trapdoor Test 定理)

$G=\langle g\rangle$是一个循环群,其中 q 为素数阶,$g\in G$ 为群的生成元。假设 X_1, r, s 是独立的变量,其中 X_1 是从 G 中选择的值,r, s 是在 Z_q 中均匀分布的,并且定义随机变量 $X_2=g^s/X_1^r$,进一步假设 $\hat{Y}, \hat{Z}_1, \hat{Z}_2$ 是从 G 中随机选择的变量,它们中的每一个都被定义为 X_1, X_2 的函数。则有:

①X_2 在 G 中均匀分布;

②X_1 与 X_2 互相独立;

③如果 $X_1=g^{r_1}, X_2=g^{r_2}$,那么等式

$$\hat{Z}_1^{\hat{Z}_2}=\hat{Y}^s \tag{1}$$

$$Z_1=\hat{Y}^{x_1} \wedge Z_2=\hat{Y}^{x_2} \tag{2}$$

不同时成立的概率最多为 $1/q$,即可以通过式(1)成立,来判断式(2)成立。

2.2 敌手模型与安全性定义

敌手能力:通过挑战者与敌手[12](外部敌手或内部恶意敌手)之间一系列的游戏定义 AKE 协议的安全性,在协议中敌手必须在测试会话上有挑战。在挑战中,恶意敌手 M 选择所有诚实用户的身份,进行下面一系列的询问:

Execute($\Pi_{U_1}^i, \Pi_{U_2}^j, \Pi_S^k$):敌手通过该询问获得诚实交互中的信息。

SendCliend(Π_U^i, m):敌手通过该询问获得客户实例 Π_U^i 对信息 m 形成的票据。另外,敌手可以通过泄露 *SendCliend*(Π_U^i, m)来初始化协议。

SendServer(Π_S^i, m):敌手通过该询问获取服务器实例 Π_S^i 对信息 m 形成的票据一系列。

Long-termKeyreveal(Π_P^i):敌手通过该询问获得 P 的长期私钥。

EphemeralKeyreveal(Π_P^i):敌手通过该询问获得 P 在实例 i 中的临时会话密钥。

SessionKeyReveal($\Pi_{U_1,U_2,S}^i$):敌手通过该询问获得实例 $\Pi_{U_1,U_2,S}^i$ 的会话密钥。

EstablishParty(U):登记客户 U 的会话密钥 PW_U。通过该询问,敌手可以控制 U 和初始化未知密钥分享(UKS)的供给。

Test($\Pi_{U_1,U_2,S}^i$):通过该询问区分真实的会话密钥。敌手可以在任何时候进行此询问,返回值为 0 或 1。如果判断出真实的会话密钥,则输出 1,否则输出 0。

TestPassword(U, PW_U):该形式的询问专门用于基于口令的协议。通过该询问可以模拟口令的泄露,敌手可以在任何时间进行此询问,返回值为 0 或 1。如果猜测的口令和真正的口令相同,则输出 1,否则输出 0。

安全性定义:

定义 1(匹配会话) $\Pi_{A,B,S}^i$是一个完全会话,它有公开的输出(ID_A, ID_B, ID_S, X, Y),其中,ID_A, ID_B 是客户的身份,ID_S 是服务器的身份,X, Y 分别是 A, B 的输出。如果 $\Pi_{B,A,S}^j$ 有完全的公开输出(ID_B, ID_A, ID_S, Y, X),则 $\Pi_{B,A,S}^j$ 叫做 $\Pi_{A,B,S}^i$的匹配会话。这里,(ID_A, ID_B, ID_S, X, Y)和(ID_B, ID_A, ID_S, Y, X)并不代表 A, B 的会话密钥不同。(ID_A, ID_B, ID_S, X, Y)的第一个元素为会话初始者,第二个元素为会话的回应者,第四个元素为会话初始者的输出,最后一个元素为会话回应者的输出。在具体的协议中,将统一会话密钥形成函数中独立变量的顺序,来保证两个客户生成相同的密钥。

定义 2(新鲜性) $\Pi_{U_1,U_2,S}^i$是一个完成会话,如果在 U_1,U_2 的会话中,匹配会话 $\Pi_{U_2,U_1,S}^j$是诚实的,并且满足:

①敌手没有对会话 $\Pi_{U_1,U_2,S}^i$或其匹配会话 $\Pi_{U_2,U_1,S}^j$进行过 *Long-termKeyreveal*(Π_P^i)和 *EphemeralKeyreveal*(Π_P^i)询问;

②敌手没有对会话进行过 *SessionKeyReveal*($\Pi_{U_1,U_2,S}^i$)询问。

则称会话 $\Pi_{U_1,U_2,S}^i$是新鲜的。

定义 3(eCK 安全) 在 eCK 模型下,敌手 M 攻击 AKE 协议 Π 的优势被定义为:

$$Adv_\Pi^{eCK}(M)=|2\Pr|b'=b|-1| \tag{3}$$

如果匹配会话能够计算出相同的会话密钥,并且没有高效的敌手可以以大于不可忽略的概率完成前面所述的询问,则称该协议是 eCK 安全的。

3 过程描述

本节提出一个在服务器协助下的三方的密钥协商协议,三方假设为 3 个客户端 A、B、C,它们在可信任服务器 S 的协助下,生成会话密钥。基本符号定义如下:ID:客户身份;pw:客户口令;r:S 的私钥;$R=g^r$:S 的公钥。

该协议经过两轮协商过程,协议过程描述如图 1 所示,两轮协商过程描述如下。

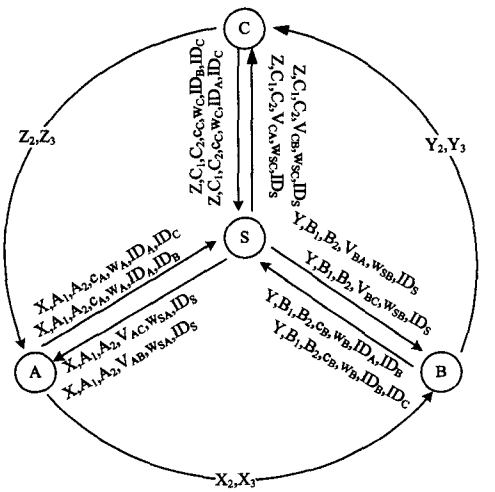

图 1 三方密钥协商

第一轮协商：

(1)客户端 A 选择短暂密钥 $a_1, a_2 \in Z_q$，计算出 $A_1 = g^{a_1}$，$A_2 = g^{a_2}$，A 判断 R 是否包含在 G 中，如果是，则 A 计算：$x = H_1(pw_A, a_1, a_2)$、$X = g^x$、$k_A = H_2(R^{a_1}, R^{a_2})$，及 $c_A = Enc_{k_A}(pw_A, w_A)$。同时，$A$ 选取随机值 $w_A \in Z_q$，然后将 $(X, A_1, A_2, c_A, w_A, ID_A, ID_B)$ 和 $(X, A_1, A_2, c_A, w_A, ID_A, ID_C)$ 发送给服务器 S。

(2)B,C 与 A 做相同的操作。

(3)服务器 S 收到 A,B,C 发来的消息后，通过 S 私钥 r 计算：$k_A' = H_2(A_1^r, A_2^r)$、$k_B' = H_2(B_1^r, B_2^r)$、$k_C' = H_2(C_1^r, C_2^r)$，接着 S 可解密：$(pw_A', w_A) = Dec_{k_A'}(c_A)$、$(pw_B', w_B) = Dec_{k_B'}(c_B)$、$(pw_C', w_C) = Dec_{k_C'}(c_C)$，判断 $pw_A' \neq pw_A$、$pw_B' \neq pw_B$、$pw_C' \neq pw_C$。如果不相等，协议中断。

(4)对于 A 发出的信息，S 计算 $V_{AB} = H_2(k_B', w_{SA}, ID_A, ID_B, ID_s)$ 和 $V_{AC} = H_2(k_C', w_{SA}, ID_A, ID_C, ID_s)$，并且将 $(X, A_1, A_2, V_{AB}, w_{SA}, ID_S)$ 发送给 B，$(X, A_1, A_2, V_{AC}, w_{SA}, ID_S)$ 发送给 C。其中，w_{SA} 是 S 选择的随机数。

类似地，对于 B 和 C 发出的消息 S 做相同的计算和发送操作。

S 删除此次会话的具体信息 $(k_A', k_B', k_C', pw_A', pw_B', pw_C')$。

(5)收到 S 发来的消息后，A 验证等式是否成立：$V_{BA} = H_2(k_A, w_{SB}, ID_B, ID_A, ID_s)$、$V_{CA} = H_2(k_A, w_{SC}, ID_C, ID_A, ID_s)$。如果成立，$A$ 计算：$Z_1 = (YB_1)^x$，$Z_2 = (YB_2)^x$，$Z_3 = Y^{x+a_1}$，$Z_4 = Y^{x+a_2}$，$sid_{AB} = (X, Y, ID_A, ID_B, ID_S)$，并得出与 B 之间的会话密钥 $SK_{AB} = H_2(Z_1, Z_2, Z_3, Z_4, sid_{AB})$；$A$ 还可以计算：$Z_1 = (ZC_1)^x$，$Z_2 = (ZC_2)^x$，$Z_3 = Z^{x+a_1}$，$Z_4 = Z^{x+a_2}$，并得到与 C 之间的会话密钥：$SK_{AC} = H_2(Z_1, Z_2, Z_3, Z_4, sid_{AC})$；$B$ 和 C 的操作类似。

(6)此时，A 与 B，B 与 C，C 与 A 分别生成 3 个会话密钥：

$$K_{AB} = H_2(Z_1, Z_2, Z_3, Z_4, sid) \tag{4}$$

$$K_{BC} = H_2(Z_1, Z_2, Z_3, Z_4, sid) \tag{5}$$

$$K_{CA} = H_2(Z_1, Z_2, Z_3, Z_4, sid) \tag{6}$$

第二轮协商：

(1)A,B,C 分别计算并传送：$A \rightarrow B: X_2 = \{g^{x_2}\}_{K_{AB}}$，$B \rightarrow C: Y_2 = \{g^{y_2}\}_{K_{BC}}$，$C \rightarrow A: Z_2 = \{g^{z_2}\}_{K_{CA}}$；

其中：$x_2 = pw_A + a_1 + a_2 + x$，$y_2 = pw_B + b_1 + b_2 + y$，$z_2 = pw_C + c_1 + c_2 + z$。

(2)收到消息后，A,B,C 分别计算：$X_2' = g^{x_2 z_2}$，$Y_2' = g^{x_2 y_2}$，$Z_2' = g^{y_2 z_2}$，分别再用 K_{AB}，K_{BC}，K_{CA} 加密 X_2'，Y_2'，Z_2' 并传送：

$$A \rightarrow B: X_3 = \{X_2'\}_{K_{AB}} \tag{7}$$

$$B \rightarrow C: Y_3 = \{Y_2'\}_{K_{BC}} \tag{8}$$

$$C \rightarrow A: Z_3 = \{Z_2'\}_{K_{AC}} \tag{9}$$

收到上述信息后，A,B,C 计算共同的会话密钥 K_{ABC}：

$$K_{ABC} = H_3(g^{x_2 y_2 z_2}, A, B, C, X_1, X_2, X_3, Y_1, Y_2, Y_3, Z_1, Z_2, Z_3) \tag{10}$$

4 安全性证明

证明过程中利用陷门测试定理，构造 CDH 数学难题，把协议的安全性归结于数学难题的难解性。如果敌手能攻破本协议，即敌手可以破解协议所基于的数学难题。然而，由于现实范围内这些数学难题是不可解的，因此可证明协议安全[13]。

定理 1 如果 $H_1(\cdot)$，$H_2(\cdot)$ 是随机预言，G 为支持 GCD 假设的群，没有 PPT 敌手可以在 eCK 模型下计算两方的会话密钥 K_{AB}，K_{AC}，K_{BC}。

该定理的证明与文献[14]中定理一的证明类似，文献[14]中证明了一个两方会话密钥的安全性。借此方法，可以证明三方的协议中 3 个两方会话密钥的安全性。

定理 2 如果 $H_1(\cdot)$，$H_2(\cdot)$，$H(\cdot)$ 是 3 个随机预言，协议中的加密算法能够抵御适应性选择密文攻击，并且 G 为支持 GCD 假设的群，则协议在 eCK 模型下是安全的。

定理 1 可以保证敌手不能计算两方的会话密钥 K_{AB}，K_{AC}，K_{BC}，如果协议中用到的加密算法可以抵御适应性选择密文攻击，那么敌手将不能得到 X_2，Y_2，Z_2，也将不能计算会话密钥 K_{ABC}，意味着协议在 eCK 模型下是安全的。

引理 1 令 E,E' 和 F 是定义在概率空间 Ω 上的随机事件，满足 $\Pr[E \wedge \neg F] = \Pr[E' \wedge \neg F]$，则有 $|\Pr[E] - \Pr[E']| \leqslant \Pr[F]$。

GameG0：代表真实协议的运行，A,B,C 所有的实例和可信任的服务器 S 都和真实的设置一样，因此敌手在 *GameG0* 中成功的概率等于敌手攻击实际协议时的成功概率。

$$Adv_{AKE}^{CK\text{-}ind}(M) = |2 \cdot \Pr[Succ_0] - 1| \tag{11}$$

GameG1：与 *GameG0* 不同，对于所有的输入输出对 $(Inp, Outp)$，都有 3 个初始为空的哈希列表 H_1^{list}，H_2^{list}，H^{list}，通过维护 Hash 表来对 Hash 预言进行模拟。

(1)Hash 询问：

①$H_1(Inp)$($resp.$ $H_2(Inp)$) 如果存在一个输入输出对 $(Inp, Outp)$，那么返回输出 $Outp$，否则随机地选择输出 $Outp \in Z_q^*$ ($resp.$ $Outp \in Z_q^*$)，将其发给敌手，并且存储该新元组 $(Inp, Outp)$ 于 H_1($resp.$ H_2^{list}) 中。

②$H(Inp)$ 如果存在一个输入输出对 $(Inp, Outp)$，那么返回输出 $Outp$，否则随机地选择输出 $Outp \in \{0,1\}^\lambda$，将其发送给敌手，并且存储该新元组 $(Inp, Outp)$ 于 H^{list} 中。

(2)针对协议中的所有加密操作，加密预言维护表：$\wedge E$，如果敌手的询问 (x, k) 在 $\wedge E$ 中存在记录 (x, y, k)，则预言返回给敌手 y，否则预言随机选择密文 y(满足 $|y| = |x|$) 返回给敌手，并将结果 (x, y, k) 写入 $\wedge E$ 中。

(3)针对协议中的所有解密操作，解密预言维护表：$\wedge D$，如果敌手提出的解密询问在 $\wedge D$ 中，那么查询后返回 x 给敌手，否则预言随机选择明文 x(满足 $|x| = |y|$) 返回给敌手，并将结果 (x, y, k) 同时写入表 $\wedge E$ 和 $\wedge D$ 中。

(4)*SendClient*($\prod_U^i$, m) 询问

做 *SendClient*($\prod_A^i$, *Start*) 询问时，假设 $\prod_A^i$ 是诚实的客户。

假设 B 是他的伙伴时，模拟此随机预言：

· 随机选择 $a_1', a_2' \in Z_q^*$，并且计算 A_1'，A_2'，x'，X'，k_A'，c_A'。

· 返回给敌手 $M(X', A_1', A_2', c_A', ID_A, ID_B)$。

假设 C 是他的伙伴时，模拟此随机预言：

· 随机选择 $a_1', a_2' \in Z_q^*$，并且计算 $A_1', A_2', x', X', k_A', c_A'$。

· 返回给敌手 $M(X', A_1', A_2', c_A', ID_A, ID_C)$。

敌手 M 做 $SendClient(\prod_B^i, Start)$ 和 $SendClient(\prod_C^i, Start)$ 询问时，会执行相同的步骤。

(5)做 $SendClient(\prod_A^i, (V_B', w_{SB}, ID_S))$ 询问时，假设 $\prod_A^i$ 是诚实的客户。

①B 是他的伙伴，模拟此随机预言：

· 计算 $sid' = (X', Y', ID_A, ID_B, ID_S), Z_1', Z_2', Z_3', Z_4', SK'$。

· 返回 SK_{AB}' 给敌手 M。

②C 是他的伙伴，模拟此随机预言：

· 计算 $sid' = (X', Y', ID_A, ID_C, ID_S), Z_1', Z_2', Z_3', Z_4', SK'$。

· 返回 SK_{AC}' 给敌手 M。

(6)$SendServer(\prod_S^k, m)$ 询问

①$SendServer(\prod_S^k, (c_A', ID_A, ID_B), (c_B', ID_A, ID_B))$，像实际攻击一样模拟该预言，并且返回给敌手 $M(X', A_1', A_2', V_{AB}', w_{SA}, ID_S)$ 和 $(Y', B_1', B_2', V_{BA}', w_{SB}, ID_S)$。

②$SendServer(\prod_S^k, (c_A', ID_A, ID_C), (c_c', ID_A, ID_C))$，像实际攻击一样模拟该预言，并且返回给敌手 $M(X', A_1', A_2', V_{AC}', w_{SA}, ID_S)$ 和 $(Z', C_1', C_2', V_{CA}', w_{SC}, ID_S)$。

(7)$Execute(\prod_{U_1}^i, \prod_{U_2}^i, \prod_S^k)$：$Execute$ 询问描述了被动攻击，过程描述如下：

①做 $SendClient(\prod_A^i, Start)$ 询问，返回给敌手 $(X', A_1', A_2', c_A', ID_A, ID_B)$ 和 $(X', A_1', A_2', c_A', ID_A, ID_C)$；做 $SendClient(\prod_B^i, Start)$ 时，返回给敌手 $(Y', B_1', B_2', c_B', ID_A, ID_B)$ 和 $(Y', B_1', B_2', c_B', ID_B, ID_C)$；做 $SendClient(\prod_C^i, Start)$ 询问时，返回 $(Z', C_1', C_2', c_c', ID_B, ID_C)$ 和 $(Z', C_1', C_2', c_c', ID_A, ID_C)$ 给敌手。

②做询问 $SendClient(\prod_A^i, (V_{AB}', w_{SB}, ID_S))$ 或 $SendClient(\prod_B^i, (V_{BA}', w_{SA}, ID_S))$ 时，返回 SK_{AB}' 给敌手，对 B, C 进行此询问时，返回值与 A 类似。

③做 $SendServer(\prod_S^k, (c_A', ID_A, ID_B)), (c_B', ID_A, ID_B)$ 时，返回给敌手 $(X', A_1', A_2', V_{AB}', w_{SA}, ID_S)$ 和 $(Y', B_1', B_2', V_{BA}', w_{SB}, ID_S)$，对其他的会话进行询问时，返回值与此类似。

(8)$SessionKeyReveal(\prod_{A,B,S}^i)$

①如果 $\prod_{A,B,S}^i$ 是测试会话，那么模拟失败；相似地，做 $SessionKeyReveal(\prod_{A,C,S}^i)$ 或 $SessionKeyReveal(\prod_{B,C,S}^i)$ 询问时，如果是测试会话，模拟失败；

②否则返回相应的会话密钥给敌手。

(9)$Long\text{-}termKeyReveal(P)$

①如果 P 是客户端，返回 pw_p；

②如果 P 是服务器，返回服务器的长期密钥。

(10)$EphmeralKeyReveal(\prod_P^i)$

①如果 $\prod_P^i$ 的临时密钥已经存在，那么返回临时密钥（例如，输出 $(a_1', a_2', x', k_A', w_A)$ $(resp.\ (b_1', b_2', y', k_B', w_B), (c_1', c_2', z', k_C', w_C))$）；

②否则返回一个空符号。

(11)$EstablishParty(U)$

M 可以注册一个长期密钥为 pw_U 的客户 U。通过这种方式，敌手可以完全地控制 U。

(12)$Test(\prod_{A,B,C,S}^i)$

①如果 $SK=\perp$，那么返回 $\perp$；

②否则，随机选择 $b\in(0,1)$，如果 $b=1$，返回 SK；如果 $b=0$，则随机选择一个值 $\zeta\in(0,1)^\lambda$ 返回。

因为 $GameG1$ 与 $GameG0$ 是完全不可区分的，所以有

$$\Pr[Succ1]=\Pr[Succ0] \tag{12}$$

$GameG2$：为了便于下面的分析，将可能产生的冲突情况删除。定义模拟中的 Hash 冲突事件为 $Coll_h$，冲突发生在 H_1, H_2 的输出的概率为 $\Pr[Coll_h]\leqslant(q_{H_1}^2+q_{H_2}^2)/2^{\lambda+1}$。定义加密/解密过程中的手抄本冲突事件为 $Coll_d$，由生日悖论，可得冲突概率 $(q_e+q_d)^2/2(q-1)$。

关于碰撞和生日悖论的关系，以哈希为例，简要证明如下：敌手对哈希预言一共进行 q_h 次询问，哈希函数的输出是空间 $\{0,1\}^\lambda$，即共有 2^λ 种输出形式。令 C_i 表示第 i 次查询的结果和之前的任意一个查询结果相同（即发生碰撞），那么，$\Pr[C_i]=(i-1)/2^\lambda$。所以，

$$\begin{aligned}&\Pr[C_1\vee C_2\vee C_3\vee\cdots\vee C_{q_h}]\\&\leqslant\Pr[C_1]+\Pr[C_2]+\Pr[C_3]+\cdots+\Pr[C_{q_h}]\\&=\frac{0}{2^\lambda}+\frac{1}{2^\lambda}+\frac{2}{2^\lambda}+\cdots+\frac{q_h-1}{2^\lambda}\\&=\frac{q_h(q_h-1)}{2\cdot 2^\lambda}\leqslant\frac{q_h^2}{2^{\lambda+1}}\end{aligned}$$

可得：

$$|\Pr[S_2]-\Pr[S_1]|\leqslant\Pr[Coll]=\frac{q_{H_1}^2+q_{H_2}^2}{2^{\lambda+1}}+\frac{(q_e+q_d)^2}{2(q-1)} \tag{13}$$

$GameG3$：在该游戏中，敌手试图获得两方的临时会话密钥来攻破协议的安全性。下面介绍如何让构建一个 CDH 问题的解决者 W，在 $GameG3$ 中以不可忽略的概率攻破协议的数学难题。换句话说，给出一个 CDH 实例，$\Sigma=g^\sigma, \Xi=g^\zeta$，$\Sigma, \Xi\in G$，如果敌手以不可忽略的概率成功获得了测试会话的会话密钥，那么 CDH 问题的解决者 W，可以以不可忽略的概率解决 CDH 问题：$CDH(\Sigma,\Xi)=\Sigma^\zeta=\Xi^\sigma$。

根据 eCK 模型安全性的定义，敌手是被允许获得参与者的长期密钥 (pw_A, pw_B, pw_C, r) 和服务器会话特殊信息（包括临时密钥），但是不允许获得客户端的临时密钥 $(a_1, a_2, x, k_A, w_A), (b_1, b_2, y, k_B, w_B), (c_1, c_2, z, k_c, w_c)$。但是，CDH 问题的解决者 W，可以在不知道 A, B, C 临时密钥的情况下，模拟预言的询问。

首先，W 随机选择被诚实的客户 A, B 执行的匹配会话。接着，W 选择随机值 $\theta, \phi\in Z_q^*$，然后，分配 B 的临时值 $B_1=\Xi, B_2=g^\theta/\Xi^\varphi$。在 $GameG3$ 中，W 因为不知道 B 真正的临时密钥，所以 W 不能回答询问 $(Z_1, Z_2, Z_3, Z_4, X, Y, ID_D, ID_B, ID_S)$，其中 D 为 B 的伙伴，但是可能是不诚实的。对于其他所有的询问，W 的返回值与 $GameG2$ 相同。

$\Pr(M)$ 是事件发生的概率，事件为 M 将会选择两个会话中的一个，这两个会话是在测试会话之前被 W 选择的。因为每一个客户在协议中最多只能有 l 个会话，所以有 $\Pr(M)\geqslant 2/l^2$。下面讨论当敌手 M 对客户 B 做相关询问时，W 的行为。

根据 eCK 模型，当 M 激活一个客户，W 拥有他的长期密钥，遵循协议的过程，当 M 对 $(Z_1,Z_2,Z_3,Z_4,X,Y,ID_D,ID_B,ID_S)$ 做 Hash 询问时，如果 $(Z_1,Z_2,Z_3,Z_4,X,Y,ID_D,ID_B,ID_S)$ 是存在的，则返回给对应的 Hash 值 h，并像 *GameG*1 一样把它存储在 H^{list} 中。否则，W 在整个集合中查找 (X,Y,ID_D,ID_B,ID_S,h)。

(1)如果找到对应的 X,Y,W 计算 $Z_1'=Z_1/X^y$，$Z_2'=Z_2/X^y$。接着 W 计算 $Z_1'^{\phi}Z_2'$，并且判断等式 $Z_1'^{\phi}Z_2'=X^{\theta}$ 是否以最大的概率 $(q-1)/q$ 成立。

如果等式成立，用门限测试的方法，意味着敌手 M 已经计算的 $Z_i(i=1,2,3,4)$ 是正确的，那么 W 从 sid^{list} 中返回存储的 h 值。否则，W 随机选择 $h'\in\{0,1\}^{\lambda}$，并且存储新的元组到 H^{list} 中，返回 h' 给 M。

(2)如果 $(Z_1,Z_2,Z_3,Z_4,X,Y,ID_D,ID_B,ID_S)$ 没有被存储在 H^{list} 中，(X,Y,ID_D,ID_B,ID_S) 没有存储在 sid^{list} 中，W 随机选择 $h'\in\{0,1\}^{\lambda}$，并且存储新的元组到 H^{list} 中，返回 h' 给 M。

如上分析，如果 M 攻破了协议的 eCK 安全性，他必须进行 Hash 询问，并且有正确的 SK（正确的 h 值）在 H^{list} 中。W 以 $1/q_H$ 的概率正确地从 H^{list} 中选择对应的 $(Z_1,Z_2,Z_3,Z_4,X,Y,ID_D,ID_B,ID_S,h)$，那么 W 执行下面的步骤来解决 CDH 问题：

①$Z_2/Z_1=(YB_2)^x/(YB_1)^x=(B_2/B_1)^x=q^{(\theta-b_1\phi-b_1)x}$；

②因为 W 知道 θ 与 ϕ，W 计算 $Z_1'=(g^{(\theta-b_1\phi-b_1)x}/X^{\theta})^{-1/\phi+1}=B_1^x$，定义 $X=\Sigma$，W 用值 Z_1,Z_2 解决 CDH 问题：给出 $X=g^x$，$\Xi=g^{b_1}$，可以在多项式时间内计算 $CDH(\Sigma,\Xi)=B_1^x=\Xi^x$。

通过以上讨论，得出

$$|\Pr[S_3]-\Pr[S_2]|\leqslant\frac{l^2\cdot q\cdot q_H}{2(q-1)}\cdot Adv_{AKE}^{CDH}(W) \tag{14}$$

正如协议中的描述，敌手 M 有 4 种可能的方法可以区分真实协议与 *GameG*3。

Case C_1：密钥复制攻击(Key-replication attack)，这时敌手选择两个不同的非匹配会话，使它们计算相同的密钥，并且选择它们中的一个作为测试会话。这时敌手可以通过对另一个会话做 *SessionKeyReveal* 询问，来获得测试会话的会话密钥。

Case C_2：M 在 (pw_A,a_1,a_2) 上做 H_1 询问。

Case C_3：M 在 (pw_B,b_1,b_2) 上做 H_1 询问。

Case C_4：M 在 (pw_C,c_1,c_2) 上做 H_1 询问。

在第一种情况下，敌手尝试初始化密钥复制攻击来获得测试会话的会话密钥。但是相同的会话密钥需要相同的参与者和相同的临时公钥，如果两个会话是不匹配的，构建相同会话密钥的唯一方法是找到两个会话的冲突。然而，C_1 发生的概率是 q_H/q。因为 eCK 模型不允许敌手揭露客户端的临时密钥，那么 C_2 发生的概率为 $(1/q^2)+(q_{H_1}/q)(1/q^2$ 猜测 a_1，a_2，q_{H_1}/q 猜测冲突)。相似地，C_3 和 C_4 也有相同的概率。因此：

$$\Pr[Succ_3]=\frac{1}{2}+\max\{\frac{q_H}{q},\frac{1}{q^2}+\frac{q_{H_1}}{q}\} \tag{15}$$

*GameG*4：从 *GameG*3 中保证敌手不能计算两方的会话密钥 K_{AB}，K_{AC}，K_{BC}，并且协议中所使用的加密/解密算法能够抵御适应性选择密文攻击，那么敌手将不能得到 X_2,Y_2,Z_2，不能计算出会话密钥 K_{ABC}(定理 2)。

$$|\Pr[S_4]-\Pr[S_3]|\leqslant 2Adv_{AKE}^{CCA}(M) \tag{16}$$

由式(11)－式(16)，利用 Hybrid 技巧得到：

$$Adv_{AKE}^{CK}(A)\leqslant\frac{3\cdot l^2\cdot q\cdot q_H}{2(q-1)}\cdot Adv_{AKE}^{CDH}(W)+2Adv_{AKE}^{CCA}(A)+\max\{\frac{q_H}{q},\frac{1}{q^2}+\frac{q_{H_1}}{q}\}+\frac{q_{H_1}^2+q_{H_2}^2}{2^{\lambda+1}}+\frac{(q_e+q_d)^2}{2(q-1)} \tag{17}$$

基于安全性假设，敌手的优势 $Adv_{AKE}^{CDH}(W)$ 和 $Adv_{AKE}^{CCA}(M)$ 都是可忽略的，因此敌手 M 攻破协议的优势是可忽略的，即协议在 eCK 模型下是安全的。

5 性能分析和安全假设对比

表 1 在效率、安全性及所基于的假设等方面对本文的协议与应用较多的几个传统协议进行了对比。为了方便，不考虑可能会应用到的分组验证和加速技巧，只是考虑高代价的计算和由“E”定义的在群中的幂运算。

表 1 协议的性能比较

协议	计算量	安全模型	假设
KEA+[15]	3E	CK,KCI	GDH,ROM
HMQV[16]	2.5E	CK,KCI,wPFS	GDH,KEA1,ROM
NAXOS[17]	4E	eCK	GDH,ROM
CMQV[18]	3E	eCK	GDH,ROM
Kudla-Paterson[19]	3E	BR,KCI	GDH,ROM
Jeong-Katz-Lee[20]	3E	BR,wPFS	DDH,secureMACs,Standard
Okamato[21]	8E	eCK	DDH,πPRF,Standard
Boyd etal[22]	1Enc+1Dec+2E	CK,KCI,wPFS	*1,Standard
本文方案	5E	eCK	CDH,ROM

与 KEA+和 Kudla-Paterson 相比，本文协议在幂指数运算 5E 的计算量上略显劣势，但所基于的数学假设和安全性方面具有一定优势。与 HMQV 相比，本文协议用到的 CDH 假设比 GDH 和 KEA1 假设更加标准。虽然 NAXOS 和 CMQV 被证明是在 eCK 模型下安全的，但是这两个协议都是基于 GDH 假设。

虽然 Jeong-Katz-Lee 的协议在标准模型下是高效的，但是该协议不能抵御 KCI 攻击，安全模型比较弱。另一方面，虽然 Okamoto 的方案在 eCK 模型下是安全的，并且在标准模型下是可证明安全性的，但是 Okamoto 的方案是基于 DDH 难题。

Boyd 等的协议是通用的，只要是 CCA 安全的，它就可以被任意的 KEM 组合实例化。对于 Boyd 等的协议，当所基于的 KEM 是由任意标准模型下的 KEM 实例化时，本文协议因在随机预言模型下而更加高效；而当所基于的 KEM 是由任意随机预言模型下的 KEM 实例化时(比如，它基于 CDH 假设并且是 CCA 安全的)，它比本文协议需要更多的计算量。

总之，本文协议更好地平衡了效率、安全性以及安全性假设 3 个标准。

结束语 本文提出了一个三方基于计算性假设的认证密钥交换协议，该协议很好地平衡了效率、安全性以及安全性假设 3 个标准，并且该协议基于 eCK 模型，与以往模型相比，更

好地支持了敌手的询问。新协议在计算量方面的优势并不明显，如何在安全假设和安全目标不变的情况下提出计算量更小的认证密钥交换协议，是下一步值得研究的问题。

参 考 文 献

[1] Bellare M, Rogaway P. Entity authentication and key distribution [C]// Stinson D R. Crypto'93, Lencture Notes in Computer Science 773. Berlin: Springer, 1993: 232-249

[2] Canetti R, Krawczyk H. Analysis of key-exchange protocols and their use for building secure channels [C] // EUROCRYPT 2001, LNCS. vol. 2045, Springer-Verlag, 2001: 453-474

[3] LaMacchia B, Lauter K, Mityagin A. Stronger security of authenticated key exchange[C]// ProvSec 2007, LNCS, vol. 4784. Springer-Verlag, 2007: 1-16

[4] Lee J, Park J H. Authenticated Key Exchange Secure under the Computational Diffie-Hellman Assumption [R]. Report 2008/344. Cryptology ePrint Archive, 2008

[5] 秦波，伍前红，王育民，等. 密钥协商协议进展[J]. 计算机科学，2008, 35(9): 9-12

[6] 任勇军，王建东，王箭，等. 标准模型下基于身份的认证密钥协商协议[J]. 计算机研究与发展，2010, 47(9): 1604-1610

[7] 赵建杰，谷大武. eCK 模型下可证明安全的双方认证密钥协商协议[J]. 计算机学报，2011, 34(1): 47-54

[8] Zhou Qing-lei, Yang Zeng-fu. TUP: A New eCK-Secure AKE Protocol under the CDH Assumption [J]. Int. J. Communications, Network and System Sciences, 2012, 5: 332-336

[9] Bian Shi-zhu, Wang Jian-dong, Ren Yong-jun. Strongly-secure and Efficient Authenticated Key Exchange Protocol [J]. Computer Engineering, 2010, 36(7)

[10] Nishide T, Yoneyama K, Ohta K. Attribute-based encryption with partially hidden encryptor-specified access structures[C]// Proceedings of the 6th international conference on Applied cryptography and network security, 2008. New York, USA, 2008: 111-129

[11] Kim M B, Fujioka A, Ustaoglu B. Strongly Secure Authenticated Key Exchange without NAXOS' Approach[C]// Proc. of the fourth International Workshop on Security, IWSEC09 Lecture Notes in Computer Science. vol. 5824, Springer-Verlag, 2009: 174-191

[12] Cremers C J F. Formally and practically relating the ck, ck-hmqv, and eCK security models for authenticated key exchange [OL]. Cryptology ePrint Archive, August 2009

[13] Cheng Qing-feng, Ma Chuan-gui, Wei Fu-shan. A modified eCK model with stronger security for tripartite authenticated key exchange[C]// Zhengzhou information Science and Technology Institute. Zhengzhou, China, 2010

[14] McCullagh N, Barreto S L. A new two-party identity-based authenticated key agreement [C]// Proceedings of CT-RSA 2005, Lecture Notes in Computer Science 3376. Springer-Verlag, 2005: 262-274

[15] Lauter K, Mityagin A. Security analysis of KEA authenticated key exchange[C]// Public Key Cryptography PKC2006, LNCS 3958. 2006: 378-394

[16] Krawczyk H. HMQV: A high-performance secure Diffie-Hellman protocol, Advances in Cryptology CRYPTO 2005 [C]// LNCS 3621. 2005: 546-566

[17] LaMacchia B, Lauter K, Mityagin A. Stronger security of authenticated key exchange[C]// ProvSec2007, Lecture Notes in Computer Science 4784. 2007: 1-16

[18] Ustaoglu B. Obtaining a secure and efficient key agreement protocol from (H)MQV and NAXOS[J]. Designs, Codes and Cryptography, 2008, 46(3): 329-342

[19] Kudla C, Paterson K. Modular security proofs for key agreement protocols[C]// Advances in Cryptology-Asia crypt 2005, LNCS 3788. Springer-Verlag, 2005: 549-565

[20] Jeong I R, Katz J, Lee D H. One-round protocols for two-party authenticated key exchange [C] // Applied Cryptography and Network Security, Second International Conference, ACNS 2004, volume 3089 of Lecture Notes in Computer Science. Springer, 2004: 220-232

[21] Okamoto T. Authenticated key exchange and key encapsulation in the standard model [C] // Advances in Cryptology-ASIACRYPT 2007, volume 4833 of Lecture Notes in Computer Science. Springer, 2007: 474-484

[22] Boyd C, Cliff Y, Nieto J G, et al. Efficient one-round key exchange in the standard model [C] // Information Security and Privacy 2008, Volume 5107 of Lecture Notes in Computer Science. Springer, 2008: 69-83

（上接第 168 页）

[7] 欧阳一鸣，董少周，梁华国. 基于 2DMesh 的 NoC 路由算法设计与仿真[J]. 计算机工程，2009, 35(22): 227-229

[8] 朱红雷，彭元喜，尹亚明，等. 一种动态分配虚拟通道输出队列结构的片上路由器[J]. 计算机研究与发展，2012, 49(1): 183-192

[9] 周芳，吴宁，周磊，等. 面向低功耗的片上网络虚拟通道分配算法[J]. 东南大学学报：自然科学版，2013, 43(2): 263-267

[10] 胡哲琨，陈杰. 完全自适应路由算法的虚拟通道分配优化策略[J]. 微电子学与计算机，2013, 30(8): 1-7

[11] Yoon Y J, Concer N, Petracca M, et al. Virtual channels vs. multiple physical networks: A comparative analysis [C] // Proceedings of the 47th ACM/IEEE Design Automation Conference. 2010: 162-165

[12] Lin J, Lin X, Tang L. Making-a-stop: A new bufferless routing algorithm for on-chip network[J]. Journal of Parallel and Distributed Computing, 2012, 72(4): 515-524

[13] Chawade S D. Review of XY routing Algorithm for Network-on-chip Architecture[J]. International Journal of Computer Applications, 2012, 43(21): 48-52

[14] Chiu Ge-ming. The Odd-Even Turn Model for Adaptive Routing [J]. IEEE Transactions on Parallel and Distributed Systems, 2000, 11(7): 729-736

[15] Hu S, Lin X. A Symmetric Odd-Even Routing Model in Network-on-Chip[C]// Proceedings of the 11th International Conference on Computer and Information Science. Shanghai, China, 2012: 457-462

[16] Moosavi S R, Chang C-Y, Rahmani A, et al. An efficient history-based routing algorithm for interconnection networks[C]// Proceedings of the International SoC Design Conference. 2012: 277-280

[17] Al-Nayeem A, Zerin T. gpNoCsim 1.0 User's Guide 2006 [OL]. http://www.buet.ac.bd/cse/research/noc/