计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2021,30(11):298−303 [doi: 10.15888/j.cnki.csa.008164]
©中国科学院软件研究所版权所有.
E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于口令和智能卡的认证与密钥协商协议①

洪　璇, 王鹏飞

(上海师范大学 信息与机电工程学院, 上海 201400)
通讯作者: 王鹏飞, E-mail: pofian@qq.com

摘　要: 因为身份验证的过程有概率会让使用者的秘密信息暴露, 导致恶意敌手可能会追踪到用户的秘密信息, 并对其进行不法利用产生危害和利益损失. 比如在基于 SmartCard 的匿名 PAKA 协议方案中, SmartCard 遗失后便没有办法防御敌手的离线字典攻击. 因此, 将双线性配对操作、D-H 困难和椭圆曲线运算等操作与注册和认证过程相结合, 分别基于口令和智能卡进行改进设计了新方案. 并将智能卡与改进的基于口令 AKA 方案相结合, 提出了基于智能卡和口令结合的 AKA 协议方案, 给了出安全性证明. 进一步提高了基于 SmartCard 和密码的 PAKA 协议的可靠安全性.

关键词: 密钥协商协议; 身份验证和密钥; 密码; 智能卡; 抗临时密钥泄漏

引用格式: 洪璇,王鹏飞.基于口令和智能卡的认证与密钥协商协议.计算机系统应用,2021,30(11):298–303. http://www.c-s-a.org.cn/1003-3254/8164.html

Key Agreement Protocol Based on Password and Smart Card

HONG Xuan, WANG Peng-Fei

(College of Information, Mechanical and Electrical Engineering, Shanghai Normal University, Shanghai 201400, China)

Abstract: Since there is a chance to expose the user's secret information during authentication, malicious adversaries may trace the user's secret information and make illegal use of it, causing harm and loss of interest. For example, in the anonymous PAKA protocol based on SmartCard, there is no way to defend against the offline dictionary attack from adversaries after the SmartCard is lost. Therefore, the bilinear pairing operation, D-H difficulty and elliptic curve operation are combined with the registration and authentication, and then a new scheme is improved and designed utilizing password and smart card respectively. On the basis of the combination of a smart card with the improved password-based AKA scheme, an AKA protocol scheme relying on both the smart card and password is proposed, with the security proof given. It further improves the reliability and security of the PAKA protocol based on SmartCard and password.

Key words: key agreement protocol; authentication and key; password; smart card; anti temporary key leakage

随着社会科技手段的快速进步, 社会步入了通信发展时代, 秘密通话在大家的生活中尤为重要. 身份验证和密钥协商协议作为通信安全的重要领域, 可以确保用户在不安全的信道上进行安全的信息验证和传输. 它不仅是加密通信的重要组成部分, 而且还允许两个或更多使用者通过敌手控制的非加密通信安全通话, 并产生出公共会话密钥以实现开放网络中的通信安全性. 然而对应的, 我们依然无法忽视恶意攻击者的各种手段, 因此众多学者在认证密钥协商协议上进行探讨与改进, 为密钥协商协议的研究发展做出重大贡献[1].

① 收稿时间: 2021-01-19; 修改时间: 2021-02-26; 采用时间: 2021-03-11; csa 在线出版时间: 2021-10-22

© 中国科学院软件研究所 http://www.c-s-a.org.cn

文献 [2] 针对临时秘钥泄露问题, 设计出解决此问题的 eCK 方案模型, 并给出一套针对此攻击的 Naxos 方案. 但是, eCK 模型仅思考了协议将被哪些攻击手段攻击, 而没有思考协议的强前向安全性. 顾兆军等人则是考虑了协议的前向安全性, 并将双线性对方法引入到协议中形成新的方案, 使得协议安全强度更高[3]. 文献 [4] 在基于困难问题的 DDA 协议下设计了新的可信匿名方案协议, 使得通信双反实现可匿名并且安全通信. 曹阳等人在文献 [5] 中为了使得 SIP 协议的安全性进一步提升, 同样是基于椭圆曲线离散对数等困难操作下设计了新的 AKA 协议. 黄朝阳更是于文献 [6] 中更加具体的将困难问题函数结合到公钥密码协议的认证过程里, 提升运算复杂程度来增加可靠性. 曾继强等则是为了更高效地解决群组秘钥的生成, 基于三叉树方案提出了新的协议, 高效可靠地解决了群组会话通信问题[7]. 赵广强等人为了抵抗 DDOS 手段与薛峰等人协议方案中某一阶段存在的问题, 设计出基于双线性对计算的新方法, 使得协议更加安全可靠[8].

本文研究了匿名两方 AKA 协议, 发现其不能防御离线字典攻击, 于是将抗离线字典攻击安全属性引入设计的 AKA 协议同时, 还将椭圆曲线运算、双线性配对操作、D-H 困难操作等与注册和认证方案过程结合, 再将智能卡与口令的协议方案相结合, 可以有效使得协议运行的安全性显著提升, 并将加密的密码存在服务器的身份验证表中, 同时在认证过程进行调整, 使得方案可以实现双方互相认证. 最后进行了安全性证明, 在安全与轻量化计算过程方面更有优势.

1 背景知识

1.1 椭圆曲线间隙假设 (ECCDH 假设)

令Q是在有限域F_q上的椭圆曲线, q是基于秘密参数k的大素数, 在Q中一个阶为大素数x中选取一个点p, 并且通过p生成一个循环加法群G. 其中$G \subset F_q$, 由此可以:

定义 1. ECCDH 假设 (Elliptic Curve Computational Diffie-Hellman, ECCDH). 假如有秘密参数k并且k足够大, 给定任何概率多项式时间算法 A 解决问题的优势函数为:

$$Adv_Z^{\mathrm{ECCDH}}(A) = Pr[A(G,P,aP,bP) = abP] \le \varepsilon(k) \quad (1)$$

则称循环加法群 G 满足 ECCDH 假设, 其中, 等式左边表示算法 A 解决椭圆曲线问题的优势, $\varepsilon(\cdot)$是可以忽略的函数, $(a,b) \in Z_q^*$.

1.2 安全模型

协议参与者和初始化. 我们在系统中设置了 3 种类型的实体: 使用者 U, 敌手 A 和服务器 S, Set Authentication 和密钥协议 P, P 和操作协议称为协议 P. 服务器跟使用者之间可以有很多实例. 我们用$\prod_U^i$表示用 U 的第i次协议实例, S 的第j次实例用$\prod_s^j$来标识. 假如敌手 A 能够操控所有通信环境. 他想要使 U 的认证阶段失败, 或是窃得 S 跟用户 U 的协商密钥, 这里使用敌手查询 P 的例子来反映敌手的能力. 具体来说, 敌手可以如下查询协议实例:

Execute($\prod_U^i, \prod_s^j$)里对 A 的被动攻击进行模拟, 实例$\prod_U^i$和$\prod_s^j$的详细会话进度将被 A 全部监听到. 并且返回信息为 P 的一次实例的所有会话信息.

Send($\prod_U^i, \prod_s^j, m$): 这里对 A 的主动攻击进行模拟, A 会给实例$\prod_U^i$或者$\prod_s^j$传输一个已经修改过的信息 m, 然后把 P 的运行结果输出给 A.

Corrupt (U, password): 这里对密码遗失攻击进行模拟, A 可以使用此查询来得到 U 的密码.

Corrupt (U, smartcard): 这里对智能卡遗失攻击进行模拟, A 可以使用此查询来得到智能卡里存放的秘密消息.

Reveal($\prod_U^i, \prod_s^j$): 这种查询模拟实例$\prod_U^i$与$\prod_s^j$会话密钥泄漏, 会输出询问实例的秘钥 sk. 如果实例返回结果不是($accept$), 那么终止. 执行了本查询的实例状态是打开的($opened$).

Test($\prod_U^i, \prod_s^j$): 这里的目的是测试出 A 可以掌握多少从实例中得到秘钥信息, 也就是来表达出我们会话密钥过程的安全性. 不管 A 执行多少次上面的查询, 最后都要执行这个查询过程. 并且最终结果如下: 假如测试过程并没有得到会话密钥, 那么结束本次查询; 否则, 开始掷硬币, 如果是数字面, 设置 B=1, 并输出实际的会话密钥; 否则为国徽面, 设置 B=0, 并输出从键空间里一个随机拿出的随机数. 此时 A 可以继续上诉来确认本次查询得到的数是真还是假. 需要补充的是此查询只可以在新对话中执行而且仅可以执行一次.

定义 2. 假如 A 执行 Test 查询后得到的$b' = b$, 那么敌手取胜, 并设定 A 取胜的优势如下:

$$Adv_{p,d}^{ake}(A) = 2 \cdot pr[b = b'] - 1 \quad (2)$$

© 中国科学院软件研究所 http://www.c-s-a.org.cn

如果:

$$Adv_{p,d}^{ake}(A) = o(q_{\text{send}})/|D_{pw}| \quad (3)$$

则称协议 P 为语义安全. 其中, q_{send}表示敌手进行 send 询问的次数, $|D_{pw}|$表示口令空间的大小.

2 匿名 PAKA 协议安全性分析

Sun 等人提出了基于 SmartCard 和密码的匿名 PAKA 协议, 该方案由于使智能卡的计算与存储成本大大降低, 因此在基于智能卡的 AKA 协议中被广泛应用, 该方案还指出其在智能卡遗失时攻击是没有威胁的[9]. 然而, 实验验证表明该方法在 SmartCard 遗失后没有办法防御离线字典的攻击, 也容易发生密码泄露和伪装. 本文模拟敌手使用口令猜测攻击进行攻击验证:

如果敌手 A 已经得到 U 的 SmartCard, 此时 A 便能够掌握卡中全部秘密消息$\{IM, G_B\}$, 接着 A 便能够使用一种能够避开服务器检测的攻击手段来展开离线的密码猜测进攻.

$A: r^B \in [1,n], G_B = r_B \cdot G$

$A \to S: \{IM, G_B\}$

S 验证 U 的身份后选择$G_S = r_s \times G$

$S: K_{SU} = h_1(h(ID_U \| K_S) \| r_s \times G_B)$

$S: M_S = h_2(K_{SU} \| G_B \| G_S)$

$S \to U: \{G_S, M_S\}$

A: 敌手 A 截获信息$\{G_S, M_S\}$, 在密码空间里随机选取一个 U 的潜在密码PW'.

$A: V' = V \oplus h(PW)$, $K_{su} = h_1(V' \| r_B \times G_S)$

$K'_{su} = h_1(V' \| r_B \times G_S)$, $M'_s = h_2(K_{su} \| G_B \| G_S)$

$A: M'_s ? = M_s$

如果敌手 A 验证正确, 说明他选的PW'即为用户的PW, 否则将其在原空间中删去并再次选取潜在密码开始验证, 这个过程持续到验证$M'_s ? = M_s$为止. A 完全可以脱机运行此操作, 所以可以避免服务器的检测与查验, 这样 A 就能得到 U 的密码.

3 改进工作

3.1 针对基于口令的 AKA 协议改进

为了提升协议的安全属性, 本文进行设计, 将椭圆曲线运算、单向散列函数、D-H 困难操作等结合到协议的注册与认证过程, 并添加密码加密后存放到服务器的身份验证表中. 协议分为两个部分: 注册阶段和认证阶段.

在注册阶段做出如下改进设计: 先由 S 选择一个大素数p ($p > 3$)和常数 a 与 b 使得$4a^3 + 27b^2 \neq 0 \bmod p$, 定义在$Z_P^*$上的椭圆曲线 $E: y^2 = x^3 + ax + b$ 由一个无穷远点 O 和一个基于同余式$y^2 = x^3 + ax + b \bmod p$的解集构成, 设 p 是 E 上阶为素数 q (q>2k) 的基点, 使得在群 $G = \langle P \rangle$上的离散对数问题是难解的. 服务器选择私钥 $d_s \in Z_p^*$, 公钥$U_S = d_s \cdot p$, 选择适当的单向散列函数$H(\cdot): \{0,1\}^* \to Z_p^*$. 使协议的运算过程是难解的, 增加了协议的可靠性, 执行过程如图 1.

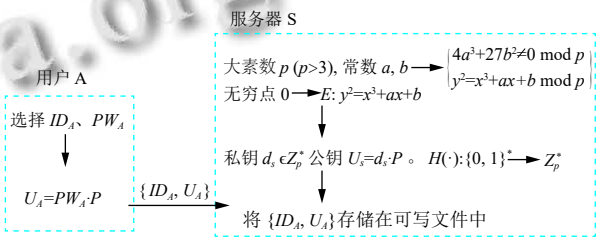

图 1 基于口令的新方案的注册阶段

在认证阶段做出如下改进设计: 用户 A 输入ID_A、PW_A, 选取随机数$r_A \in Z_p^*$, 利用随机数计算$W_A = r_A \cdot PW_A \cdot U_S$、$R_A = r_A \cdot U_A = (k_x, k_y)$、$Y_A = r_A \cdot p$. 用$k_x$作为对称加密算法的密钥, 对$(ID_A, Y_A)$加密, 得到$M_1 = E_{k_x}(ID_A, Y_A)$, 然后将信息$(ID_A, W_A, M_1)$发送给 S. 接收到使秘密信息之后,S 使用私钥 d_s 计算私钥 d_s 计算$R'_{A_x001D} = W_A \cdot d_s^{-1} = (k'_x, k'_y)$, 接着用$k'_x$解密信息 M_1,得到ID'_A、Y'_A, 服务器先检查等式是否相等, 即$ID'_A = ID_A$、$e(Y'_A, U_A) = e(R'_A, P)$, 如果等式不相等, S 将终止会话. 反之服务器 S 选择一个随机数$r_s \in Z_P^*$, 计算$M_2 = R'_A + W_S$、$M_3 = H(W_S)$. 其中$W_S = r_s U_s$, 并将$\{M_2 + M_3\}$发送给用户 A.

得到信息 M_2 后, 用户 A 计算$W'_S = M_2 - R_A$, 并检查$H(W'_S) = M_3$是否相等, 如果不相等则终止通话, 若相等, A 将$M_4 = H'(R_A, W'_S)$发送给 S.

S 接收来自使用者的消息后, 第一步检查$H(W'_S) = M_4$是否相等, 假如不相等, S 否定 A 的会话请求, 假如等式相等, S 将通过使用者的会话请求. 此时得到$SK = r_s \cdot W_A = r_A \cdot PW_A \cdot r_S \cdot d_S \cdot P$, 然后认证完成.

3.2 针对基于智能卡的 AKA 协议改进

同样的, 针对智能卡丢失后不能防御敌手的离线密码猜测这一问题, 我们将身份认证和会话密钥相结合的双向认证方案改进到基于智能卡的协议方案中,

© 中国科学院软件研究所 http://www.c-s-a.org.cn

给协议增加了抗离线字典攻击安全属性. 协议分为两个部分: 注册阶段和认证阶段.

在注册阶段做出如下改进: 用户选择自己的标识ID_i, 口令PW_i并计算$h(PW_i)$, 使用可信通道发给 S.

服务器计算$h(ID_i,x)$, 用$h(PW_i)$, 对$h(ID_i,x)$加密, 用私钥 x 对ID_i加密, 结果分别为$E_{h(PW_i)}[h(ID_i,x)]$、$E_x(ID_2)$计算$A_i=h(h(ID)\oplus h(PW_i))\bmod n$. 然后将信息$\left\{E_x(ID_i),A_i,E_{h(pw_i)}[h(ID_i,x),h(\cdot),n]\right\}$写入智能卡.

最后 S 把 SC 通过可信通道交给用户, 如图 2.

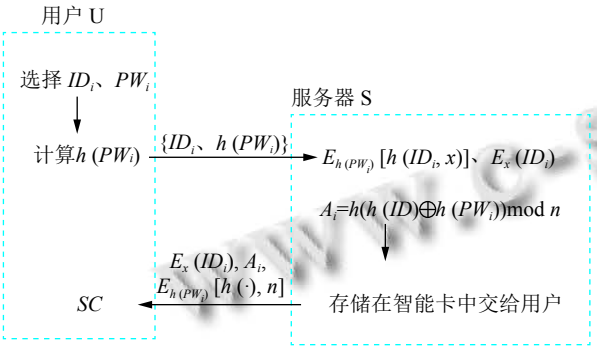

图 2 基于智能卡的新方案的注册阶段

在认证阶段做出如下改进:服务器 S 收到信息$\{C_1,E_x(ID_i)\}$后, 首先用私钥 x 解密$E_x(ID_i)$得到用户身份ID_i并判断它的有效性, 如果有效则继续下面的操作. 选择一个随机数x_B, 计算$C_2=h(ID_i,x)^{x_B}$, $k=h(ID_i,x)^{x_A^{x_B}}=C_1^{X_B}$, $B_1=h(ID_s\|K\|0)$. 将$\{B_1,C_2\}$发送给 SC(智能卡), SC 接收 S 的信息$\{B_1,C_2\}$之后, 计算$K'=C_2^{x_A}$, $B_1^*=h(ID_S\|K'\|0)$. 验证$B_1^*=B_1$, 若不成立, 那么对 S 的验证失败, 并结束通话, 否则计算$B_2=h(ID_s\|K'\|0)$; 并把B_2发给服务器, 服务器收到B_2后, 计算$B_2^*=h(ID_S\|K\|0)$, 并验证$B_2^*=B_2$, 若不成立, 则结束通话, 否则对用户的信息验证便为通过并协商公共通信私钥为$SK=h(ID_s\|ID_i\|k\|2)=h(ID_s\|ID_i\|h'\|2)$. 认证完成.

4 基于智能卡和口令结合的 AKA 协议方案

在上面的协议基础上, 我们将智能卡与上面设计的匿名口令 AKA 协议注册过程相结合并在认证过程进行改进, 使得方案的安全可靠性进一步提升. 协议包括两个阶段: 用户注册阶段和用户认证阶段.

4.1 注册阶段

由 SC 先选择一个大素数$p\ (p>3)$和常数 a 与 b 使得$4a^3+27b^2\neq 0 \bmod p$, 定义在Z_P^*上的椭圆曲线$E:y^2=x^3+ax+b$由一个无穷远点 O 和一个基于同余式$y^2=x^3+ax+b \bmod p$的解集构成, 设 p 是 E 上阶为素数 q $(q>2k)$的基点, 使得在群$G=\langle P\rangle$上的离散对数问题是难解的.

用户 A 选择PW_A 当做密码口令. SC 选取随机数$b\in\{0,1\}^{1024}$使得PW_A的熵值提升. 并将秘密信息发送给 S, S 选择私钥$d_s\in Z_p^*$, 公钥$U_S=d_s\cdot p$, 选择适当的单向散列函数$H(\cdot):\{0,1\}^*\rightarrow Z_p^*$.

$A:h(PW_A,b)$

$A\rightarrow S:\{h(PW_A,b),ID_A,h(p)\}$

S: $U_S=d_s\cdot P$, $ID_A=ID_A\|U_S$

S: 利用私钥d_s计算$W_A=h(d_s,ID_A)\oplus(PW_A,b)$, 将$\{ID_A,W_A\}$存入 SC 内发送给 A.

A 将随机数 b 和大素数 p 发送到 SC. 此时, SC 中的信息为$\{ID_A,b,p,W_A\}$.

4.2 认证阶段

当 A 想要访问 S, A 和 S 之间开始相互认证.

$A\rightarrow$Smart Card: $\{PW_A,ID_A\}$

Smart Card: $V_A=W_A\oplus h(PW_A,b,p)=(d_s,ID_A,p)$

Smart Card: 选随机数 $y\in Z_p^*$, $Y=g^y$,

$k=H_1(X^y,V_A),K=g^k$

Smart Card$\rightarrow$S: $\{ID_A,Y,K\}$//临时密钥为用户所选取的秘密值 y

S: 验证ID_A是否正确, 若正确执行下一步

S: 利用私钥d_s计算$V_A'=h(d_s,ID_A,p)$

S 验证$K=g^{H_1(X^y,V'_A)}$是否成立, 若成立, 则继续

S: $z\in Z_p^*$, $Z=g^z$

S: $Auth_s=H_2(K^x,Y^z,ID_A,Z)$//S 认证信息

$S\rightarrow A$: $\{ID_S,Z,Auth_s\}$

A: 验证 A 的秘密信息, 若通过, 则继续

Smart Card: 计算 A 与 S 会话密钥$sk=H_2(X^k,Z^y,ID_A,ID_s,X,Y,K,Z)$

Smart Card: $Auth_A=H_2(X^k,Z^y,ID_s,Y)$

Smart Card$\rightarrow S$: $\{Auth_A\}$

S: 验证$Auth_s=H_2(K^x,Y^z,ID_A,Z)$

S: 验证$Auth_A=H_2(X^k,Z^y,ID_s,Y)$

S: $sk=H_2(k^x,Y^z,ID_A,ID_s,X,Y,K,Z)$//所有认证通过, 计算出会话密钥如上$sk$, 认证完毕.

5 安全证明

定理 1. 在 1.2 节的模型下, 令Q是在有限域 Fq 上

© 中国科学院软件研究所 http://www.c-s-a.org.cn

的椭圆曲线, q是基于秘密参数k的大素数, p为 Q 中大素数x的一个点并且生成循环加法群 G. 敌手使用的Execute查询, Reveal查询, Send查询以及Oracle查询的次数分别用$q_{\text{excute}}, q_{\text{reveal}}, q_{\text{send}}, q_{\text{oracle}}$来标识, 假设$G$在ECCDH 假设条件下满足条件, 那么该协议语义是安全的. 如下:

$$\begin{aligned} Adv_{g,G}^{ake}(A) \le\ & 2(q_{\text{send}}+q_{\text{execute}})Adv_{g,G}^{hddh} \\ & +4\left(\frac{2q_{\text{send}}+q_{\text{execute}}}{q}\right) \\ & +2\left(\frac{q_{\text{send}}}{|D_{pw}|}\right)+\frac{q_{\text{oracle}}^2+q_{\text{send}}^2+q_{\text{execute}}^2}{(q-1)} \\ & +2\min\{(q_{\text{send}}+q_{\text{execute}}), q_{\text{reveal}}\}Adv_{g,G}^{hddh} \end{aligned} \tag{4}$$

证明. 在进行数次循环实例实验后 $(\prod_P^0, \prod_P^1, \prod_P^2, \cdots)$, 实例按照协议运行并且返回敌手的查询, 敌手 A 在多次修改查询方式情况下使用 $pr[S_i]$ 来代表敌手在某次实例中的概率.

$\prod_P^i(i=0)$: 本次实例攻击, 假设敌手成功, 那么S_0代表敌手在 Test 查询中成功使得硬币返回为 B=1, 则可以得到:

$$Adv_{g,G}^{ake}(A) = 2\cdot pr[b=b'] - 1 \tag{5}$$

$\prod_P^i(i=1)$: 本次实例攻击, 敌手使用随机数代替认证中的消息 Y, 则可以在 ECCDH 下得到:

$$|pr[S_1]-pr[S_0]| \le q_{\text{execute}}Adv_{g,G}^{ake}(A) \tag{6}$$

$\prod_P^i(i=2)$: 本次实例攻击, 敌手使用随机数代替服务器向用户认证中的消息 Z, 执行 Execute 查询, 可以得到:

$$|pr[S_2]-pr[S_1]| \le 1/2\left(\frac{q_{\text{execute}}^2+q_{\text{oracle}}^2}{q-1}\right) \tag{7}$$

$\prod_P^i(i=3)$: 本次实例攻击, 敌手使用 $Send(\prod_{\text{P}}^i, null)$ 查询, 随机选择 Y 并在 ECCDH 下混合证明得到:

$$|pr[S_3]-pr[S_2]| \le q_{\text{send}}Adv_{g,G}^{hddh}(A) \tag{8}$$

$\prod_P^i(i=4)$: 本次实例攻击, 假设敌手验证失败, 那么执行 (i=3) 时的实例, 否则便假设敌手已经验证成功, 使用 $Send(S^j, ID_i\|Y\|K)$ 查询验证成功. 至此, 如果敌手一直重复攻击查询, 那么他获得的消息数目最多为 $q_{\text{execute}}+q_{\text{send}}$, 那么他成功的概率则是$\frac{q_{\text{execute}}+q_{\text{send}}}{q}$. 如果敌手口令验证正确, 并且使用了$Corrupt(U,2)$查询, 并且所有消息都与正确口令无关, 则他的成功率即为$\frac{q_{\text{send}}}{|D_{pw}|}$.

如果他验证密码一直不成功, 但是却执行了一个满足条件需求的新查询, 此时他没有一条和正确密码有关的消息, 他选定的秘密信息 Y,K 也被唯一的确定下来, 则此时的成功率为$\frac{q_{\text{send}}}{q}$. 综上可得:

$$|pr[S_4]-pr[S_3]| \le \frac{q_{\text{send}}}{|D_{pw}|}+\frac{2q_{\text{send}}+q_{\text{execute}}}{q} \tag{9}$$

$\prod_P^i(i=5)$:本次实例攻击, 敌手使用生日攻击, 随机选择秘密信息 Z, 执行 $Send(S^j, ID_i\|Y\|K)$ 得到:

$$|pr[S_5]-pr[S_4]| \le \frac{1}{2}\cdot\frac{q_{\text{oracle}}^2+q_{\text{send}}^2}{q-1} \tag{10}$$

$\prod_P^i(i=6)$: 本次实例攻击, 假设在上次实例中敌手没有猜中口令, 即排除概率$\frac{q_{\text{send}}}{|D_{pw}|}$事件, 本次使用随机数代替会话秘钥并使用 Reveal 操作, 假如执行 Execute 和 Send 后不使用 Reveal, 那么就与 i=5 实例中获得信息相同, 则综合可以得到:

$$|pr[S_6]-pr[S_5]| \le \min\{(q_{\text{send}}+q_{\text{execute}}), q_{\text{reveal}}\}Adv_{g,G}^{hddh} \tag{11}$$

本次实例里, 敌手无法获得任何和密码相关的信息, 那么在执行 Test 时, 他可以正确预测模型硬币的概率是$\frac{1}{2}$, B (b=0, b=1). 因此包含所有实例事件结果的综合概率即为:

$$\begin{aligned} Adv_{g,G}^{ake}(A) \le\ & 2(q_{\text{send}}+q_{\text{execute}})Adv_{\text{g},G}^{hddh} \\ & +4\left(\frac{2q_{\text{send}}+q_{\text{execute}}}{q}\right) \\ & +2\left(\frac{q_{\text{send}}}{|D_{pw}|}\right)+\frac{q_{\text{oracle}}^2+q_{\text{send}}^2+q_{\text{execute}}^2}{(q-1)} \\ & +2\min\{(q_{\text{send}}+q_{\text{execute}}), q_{\text{reveal}}\}Adv_{g,G}^{hddh} \end{aligned} \tag{12}$$

由此, 可以得出定理 1 正确, 协议可证明安全.

6 性能分析

由于安全性能的提升往往伴随着算法复杂度的提升, 这就意味着可能会牺牲小部分效率, 因此, 我们将本文所提方案与传统协议和最新的协议进行比较和性能分析.

我们以文献 [9] 作为传统协议的代表, 再以文献 [3] 作为最近的改进协议作为代表与本文方案进行对比, 完成一次哈希运算的效时为E_h, 完成一次乘法运算与

© 中国科学院软件研究所 http://www.c-s-a.org.cn

次方运算为E_p, 假设在相同等级系统中运行交换, 那么数据交换次数越多, 用时越长, 我们使用E_c来表示. 并且已知椭圆曲线运算的效率约比双线性运算节约近二十倍. 我们将运行效率结果通过表格展示.

由表 1 和表 2 可以看出, 本文方案在运行效率上明显略优于文献 [3] 和文献 [9], 相比较而言文献 [3] 的数据交换次数略多, 并且从安全性上, 本文方案强度显然是高于文献 [3] 和文献 [9]. 实际中, 我们将口令与智能卡相结合, 增加安全强度的同时, 用户的使用体验也会更加方便, 因此, 总体上我们的方案具有一定优势.

表 1 效率对比

方法	哈希运算	运行效率	通信轮数
传统协议	8	$8E_h+21E_p+4E_c$	4
文献[3]	7	$7E_h+17E_p+5E_c+20$	5
本文方案	7	$7E_h+18E_p+3E_c+21$	3

表 2 性能对比

方法	传统协议	文献[3]	本文方案
认证目标	智能卡、用户	用户、服务器	智能卡、服务器、用户
口令猜测攻击	×	√	√
口令泄露伪装攻击	×	√	√
认证方法	智能卡	口令	智能卡、口令

7 结束语

本文设计了一种基于口令的 AKA 协议方案. 并利用单向 hash 函数和椭圆曲线上的 D-H 问题来保证系统的安全与可靠. 针对基于 SmartCard 的 AKA 方案容易因为 SmartCard 被盗并无法防御离线密码猜测手段. 给出了一个将通信密钥和身份验证相结合的双向认证方案, 最后对协议进行了安全性证明. 该方案能防范多种攻击, 并且效率运行高、实际计算量较小、实际实用性强. 下一步将致力于在保证安全可用的基础下提出效率更高的认证方法.

参考文献

1 胡志言, 杜学绘, 曹利峰. 会话密钥协商协议研究进展. 计算机应用与软件, 2018, 35(5): 1–9, 72. [doi: 10.3969/j.issn.1000-386x.2018.05.001]

2 LaMacchia B, Lauter K, Mityagin A. Stronger security of authenticated key exchange. Proceedings of the 1st International Conference on Provable Security. Wollongong: Springer-Verlag, 2007. 1–16.

3 顾兆军, 刘东楠. 基于身份的无证书双线性对密钥协商方案. 中国民航大学学报, 2019, 37(1): 55–59.

4 关晨至, 石永革. 基于 DAA 的可信双向匿名认证密钥协商协议. 计算机系统应用, 2009, 18(12): 59–61. [doi: 10.3969/j.issn.1003-3254.2009.12.014]

5 曹阳. 基于 ECDLP 的 SIP 认证密钥协商协议. 计算机系统应用, 2016, 25(3): 225–228.

6 黄朝阳. 无双线性对双向认证密钥协商协议. 计算机系统应用, 2016, 25(7): 192–195. [doi: 10.15888/j.cnki.csa.005157]

7 曾继强, 史国振. 基于 ECC 的三叉树群组密钥协商方案. 计算机应用与软件, 2018, 35(9): 311–316. [doi: 10.3969/j.issn.1000-386x.2018.09.055]

8 赵广强, 凌捷. 两个基于智能卡口令认证方案的改进. 计算机应用与软件, 2015, 32(5): 278–282. [doi: 10.3969/j.issn.1000-386x.2015.05.068]

9 Sun DZ, Huai JP, Sun JZ, *et al*. Improvements of Juang's password-authenticated key agreement scheme using smart cards. IEEE Transactions on Industrial Electronics, 2009, 56(6): 2284–2291. [doi: 10.1109/TIE.2009.2016508]

© 中国科学院软件研究所 http://www.c-s-a.org.cn