一个具有多个注册中心的双向认证与密钥协商协议

李雪莲[1],李 伟[1],高军涛[2],王海玉[1]
(1. 西安电子科技大学数学与统计学院,陕西西安 710126;2. 西安电子科技大学通信工程学院,陕西西安 710126)

摘 要: 当前许多双向认证与密钥协商(MAKA)协议不具有高效的撤销机制,同时不能抵抗一些新型攻击,如随机数泄露(ESL)攻击、注册中心泄露注册信息(RCDRI)攻击.另一方面,大量的 MAKA 协议都是基于一个注册中心(RC),这无疑对 RC 高效性和稳定性是个挑战.本文基于以上问题,结合自认证公钥(SCPK)提出了一个具有多个注册中心的 MAKA 协议,该协议能够抵抗上述的新型攻击并且具有动态的撤销机制.基于 Diffie-Hellman 困难假设,在随机预言机模型中给出了协议的安全性证明.由于该协议不涉及双线性对运算,比以往同类型方案在执行效率方面也有很大的优势.

关键词: 多服务器环境;多注册中心;随机数泄露攻击;注册中心不可信;自认证公钥;撤销动态
中图分类号: TP393 **文献标识码:** A **文章编号:** 0372-2112 (2018)10-2418-05
电子学报 URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2018.10.015

A Mutual Authentication and Key Agreement Protocol with Multi-Registration Center

LI Xue-lian[1], LI Wei[1], GAO Jun-tao[2], WANG Hai-yu[1]
(1. *School of Mathematics and Statistics, Xidian University, Xi'an, Shaanxi* 710126, *China*;
2. *School of Telecommunications Engineering, Xidian University, Xi'an, Shaanxi* 710126, *China*)

Abstract: Most of the current Mutual Authentication and Key Agreement (MAKA) protocols fail to provide effective revocation mechanism and suffer from some new attacks, such as Ephemeral Secret Leakage (ESL) attack and Registration Center Disclosure Registration Information (RCDRI) attack. On the other hand, a large number of MAKA protocols are based on a Registration Center (RC), which is undoubtedly a challenge to RC efficiency and stability. Based on Self-Certified Public Key (SCPK), this paper proposes a MAKA protocol for multi-server environments with multi-registration centers. It is able to resist the new attacks and has efficient dynamic revocation mechanism. Based on the Diffie-Hellman assumption, the security proof of the proposed protocol is given in the random oracle model. Because the protocol does not involve bilinear pairings of operations, it has a great advantage in the implementation efficiency over the relevant schemes.

Key words: multi-server environment; multi-registration center; ephemeral secret leakage attack; registration center disclosure information attack; self-certified public key; dynamic revocation

1 引言

双向认证与密钥协商(MAKA)协议不但能够使交互的双方认证彼此的合法性,还能产生一个临时的会话密钥来保护数据的机密性.因此当 Lamport[1] 提出第一个认证方案后,大量基于单服务器环境的 MAKA 协议[2,3] 被提出.由于单服务器的 MAKA 协议要求用户记住不同服务器的私钥,因此基于多服务器环境下的 MAKA 协议[4~14] 被广泛研究.在多服务器环境中的协议,用户只需在注册中心(RC)注册,就可以使用不同服务器的资源.但分析发现许多 MAKA[4~8,10~13] 协议在注册中心泄露注册信息(RCDRI)攻击下并不能达到其声称的安全性.另一方面,由于移动设备计算能力有限,它们往往提前计算和储存协议中要使用的随机数为改善效

收稿日期:2017-09-25;修回日期:2018-05-31;责任编辑:李勇锋
基金项目:国家重点研发计划(No. 2016YFB0800601);国家自然科学基金(No. 61303217, No. 61502372)

率,所以要求协议能抵抗随机数泄露(ESL)攻击. 本文基于以上需要,结合自认证公钥(SCPK)密码技术[14]提出一个分级的多注册中心的 MAKA 协议,不但有效的解决以上安全问题还实现了动态的撤销管理.

2 基础知识介绍

定义 1 计算性 Diffie-Hellman 假设:G_1 是椭圆曲线上点构成的群,P 是其生成元. 取 P, xP, yP,不存在概率多项式敌手 A 能以不可忽略概率 ε 计算 xyP.

定义 2 自认证公钥[14](SCPK)是指:通过使用 SCPK,用户的公钥直接通过 RC 关于这名用户身份的签名被计算获得.

3 本文所提方案

一级注册中心初始化阶段:

1. RC_1 选择一个 q 阶的椭圆曲线上的点构成的循环加法群 G_1,其生成元是 P,然后选择主密钥 $s(s \in Z_q^*)$ 并计算系统公钥 $P_{pub} = sP$.

2. RC_1 选择七个安全的哈希函数,其中 $(H_1, H_2, H_3, H_4, H_5, H_7): \{0,1\}^* \to Z_q^*$,另一个 $H_6: \{0,1\}^* \to G_1$,最后公布系统参数 $(q, P, P_{pub}, G_1, H_1, H_2, H_3, H_4, H_5, H_6, H_7)$.

二级注册中心初始化阶段:

1. $RC_{2k}(0 \le k \le t)$ 发送自己的唯一身份辨识符 $RCID_{2k}$ 给 RC_1.

2. RC_1 为每个二级生成的公私密钥对 $d_{RC2k} = s \cdot H_6(RCID_{2k})$, $P_{RC2k} = H_7(d_{RC2k}) \cdot P$,然后将 d_{RC2k} 通过安全信道发送给对应的 RC_{2k} 并把 $(P_{RC2k}, RCID_{2k})$ 添加到公布的系统参数中.

RC_{2k} 收到 d_{RC2k} 后通过验证 $e(d_{RC2k}, P) = e(P_{pub}, H_6(RCID_{2k}))$ 是否成立来防止冒充 RC_1 的攻击. 如果等式成立,RC_{2k} 计算生成最终的私钥 $d_{TRC2k} = H_7(d_{RC2k})$.

用户和服务器注册阶段:

1. 用户 ID_{ui} 选择秘密值 $s_{ui} \in Z_q^*$ 并计算 $S_{ui} = s_{ui} \cdot P$,然后提交 ID_{ui}, S_{ui} 给 RC_{2k}.

2. RC_{2k} 选择随机数 $y_{ui} \in Z_q^*$ 并计算 $Y_{ui} = y_{ui} \cdot P$,然后为用户计算生成私钥 $d_{ui} = (y_{ui} + H_1(ID_{ui}, Y_{ui} + S_{ui}) \cdot d_{TRC2k}) modq$,最后 RC_{2k} 保存用户 ID_{ui} 注册状态并通过安全信道把 $(d_{ui}, Y_{ui}, RCID_{2k})$ 发送给对应用户.

用户收到 $(d_{ui}, Y_{ui}, RCID_{2k})$ 可以用等式 $d_{ui} \cdot P ? = Y_{ui} + H_1(ID_{ui}, Y_{ui} + S_{ui}) \cdot P_{RC2k}$ 验证私钥的正确性.

1. 服务器 ID_{sj} 也选择秘密值 $s_{sj} \in Z_q^*$ 并计算 $S_{sj} = s_{sj} \cdot P$,然后发 ID_{sj}, S_{sj} 给 $RC_{2e}(0 \le e \le t)$.

2. RC_{2e} 选择随机数 $y_{sj} \in Z_q^*$ 并计算 $Y_{sj} = y_{sj} \cdot P$,然后计算生成服务器私钥 $d_{sj} = (y_{sj} + H_1(ID_{sj}, Y_{sj} + S_{sj}) \cdot d_{TRC2e}) modq$,最后 RC_{2e} 保存服务器 ID_{sj} 注册状态并通过安全信道把 $(d_{sj}, Y_{sj}, RCID_{2e})$ 发送给对应服务器.

服务器收到消息后执行类似用户的验证.

时间密钥更新和撤销阶段:

1. RC_{2k} 为每个合法的用户和服务器分别选择时间密钥的有效时间段 t_1, t_2.

2. 然后 RC_{2k} 给合法用户和服务器分别选择随机数 $v_{ui}, v_{sj} \in Z_q^*$ 并计算 $V_{ui} = v_{ui} \cdot P, V_{sj} = v_{sj} \cdot P$,最后利用以上参数分别为二者生成更新时间密钥

$$d_{tui} = (v_{ui} + H_2(ID_{ui}, V_{ui}, t_1) \cdot d_{TRC2k}) modq$$

$$d_{tsj} = (v_{sj} + H_2(ID_{sj}, V_{sj}, t_2) \cdot d_{TRC2k}) modq.$$

3. 通过公开信道发送 $(d_{tui}, V_{ui}, t_1, RCID_{2k})$, $(d_{tsj}, V_{sj}, t_2, RCID_{2k})$ 给用户和服务器.

当用户和服务器收到 $(d_{tui}, V_{ui}, t_1, RCID_{2k})$, $(d_{tsj}, V_{sj}, t_2, RCID_{2k})$,分别验证以下等式是否成立 $d_{tui} \cdot P = V_{ui} + H_2(ID_{ui}, V_{ui}, t_1) \cdot P_{RC2k}, d_{tsj} \cdot P = V_{sj} + H_2(ID_{sj}, V_{sj}, t_2) \cdot P_{RC2k}$

如果等式成立,说明更新时间密钥是 RC_{2k} 发布的,不然可能存在攻击者冒充 RC_{2k}.

认证和密钥协商阶段:

为了不失一般性,假设在 ID_{ui} 和 ID_{sj} 两者之间认证与密钥协商.

1. 用户 ID_{ui} 选择一个随机数 $r_{ui} \in Z_q^*$ 并计算 $M_{ui} = (r_{ui} + s_{ui}) \cdot P$,然后发送消息 $(M_{ui}, d_{tui}, V_{ui}, t_1, ID_{ui}, S_{ui}, Y_{ui}, RCID_{2k})$ 给服务器 ID_{sj}.

2. 当 $(M_{ui}, d_{tui}, V_{ui}, t_1, ID_{ui}, S_{ui}, Y_{ui}, RCID_{2k})$ 被 ID_{sj} 收到后. 首先在 RC_1 维护的系统参数中查询唯一标识 $RCID_{2k}$ 对应的公钥 P_{RC2k},然后利用 SCPK 验证等式 $d_{tui} \cdot P = V_{ui} + H_2(ID_{ui}, V_{ui}, t_1) \cdot P_{RC2k}$ 是否成立,从而判断时间密钥是否合法. 如果没有 $RCID_{2k}$ 对应的公钥 P_{RC2k} 或者等式不成立说明对应二级注册中心或者用户已经被撤销,服务器结束会话;不然转向一下步.

3. 服务器也选择一个随机数 r_{sj} 并计算 $M_{sj} = (r_{sj} + s_{sj}) \cdot P, R_{sj} = (d_{sj} + s_{sj}) \cdot M_{ui}$ 和 $Auth_{sj} = H_3(P_{pub}, P_{RC2e}, P_{RC2k}, ID_{ui}, ID_{sj}, M_{ui}, M_{sj}, R_{sj})$ 然后服务器发送 $(Auth_{sj}, ID_{sj}, M_{sj}, d_{tsj}, V_{sj}, t_2, Y_{sj}, S_{sj}, RCID_{2e})$ 给对应用户.

4. 当 $(Auth_{sj}, ID_{sj}, M_{sj}, d_{tsj}, V_{sj}, t_2, Y_{sj}, S_{sj}, RCID_{2e})$ 被用户 ID_{ui} 收到,首先同样在 RC_1 维护的系统参数中查询是否存在 $RCID_{2e}$ 对应的公钥 P_{RC2e},然后查看 ID_{sj} 是否为请求的服务器和验证过等式 $d_{tsj} \cdot P ? = V_{sj} + H_2(ID_{sj}, V_{sj}, t_2) \cdot P_{RC2e}$ 是否成立,进而判断服务器是否被撤销. 如果以上任何一个条件不成立,用户结束会话;不然转向下一步.

5. 用户利用以上参数计算 $R_{vsj} = (r_{ui} + s_{ui})(S_{sj} + Y_{sj} + H_1(ID_{sj}, Y_{sj} + S_{sj}) \cdot P_{RC2e})$,然后用 R_{vsj} 来验证认证等式 $Auth_{sj} = H_3(P_{pub}, P_{RC2e}, P_{RC2k}, ID_{ui}, ID_{sj}, M_{ui}, M_{sj}, R_{vsj})$ 是否成立. 如果不成立拒绝会话,不然计算 $K_{ij} = (r_{ui} + s_{ui}) \cdot$

$M_{sj}, R_{ui} = (d_{ui} + s_{ui}) \cdot M_{sj}$ 和 $SK_{ij} = H_4(ID_{ui}, ID_{sj}, M_{ui}, M_{sj}, K_{ij}, P_{pub}, R_{ui}, R_{vsj}, P_{RC2e}, P_{RC2k})$ 会话密钥. 最后计算认证凭证 $Auth_{ui} = H_5(P_{pub}, P_{RC2e}, P_{RC2k}, ID_{ui}, ID_{sj}, M_{ui}, M_{sj}, R_{ui}, R_{vsj})$ 并发送 $Auth_{ui}$ 给服务器 ID_{sj}.

6. 服务器 ID_{sj} 收到 $Auth_{ui}$ 后,利用之前的参数计算 $R_{vui} = (r_{sj} + s_{sj})(S_{ui} + Y_{ui} + H_1(ID_{ui}, Y_{ui} + S_{ui}) \cdot P_{RC2k})$. 然后验证用户发送来的 $Auth_{ui} = H_5(P_{pub}, P_{RC2e}, P_{RC2k}, ID_{ui}, ID_{sj}, M_{ui}, M_{sj}, R_{vui}, R_{sj})$ 是否成立. 如果等式不成立,服务器拒绝这次会话,不然计算 $K_{ji} = (r_{sj} + s_{sj}) \cdot M_{ui}$ 和双方临时会话密钥 $SK_{ji} = H_4(ID_{ui}, ID_{sj}, M_{ui}, M_{sj}, K_{ji}, P_{pub}, R_{vui}, R_{sj}, P_{RC2e}, P_{RC2k})$.

4 安全性证明

定理 1 本文所提的协议中敌手冒充用户攻击事件($C2S$)和敌手冒充服务器攻击事件($S2C$)的成功概率都是可忽略的.

证明 证明事件 $S2C$ 发生的概率是可忽略的. 假设敌手 A 能够以不可忽略的优势 ε 冒充服务器和用户完成认证,即事件 $S2C$ 发生的概率 $\Pr[S2C] \geq \varepsilon$.

首先实例化一个 CDH 问题 $\{P, Q_1 = x \cdot P, Q_2 = y \cdot P\}$,然后 F 模拟初始阶段生成系统参数. q_i, q_n, q_m, q_s 分别是对应哈希函数的查询次数,用户的数量,服务器的数量,$Send$ 查询的次数,其中 $1 \leq i \leq 7$. 最后 F 选择嵌入困难问题的身份 $I \in \{1,2,\cdots,q_n\}$, $J \in \{1,2,\cdots,q_m\}$,与会话 $l_1, l_2 \in \{1,2,\cdots,q_s\}$. F 如下与 A 交互.

***H_k* 查询**:F 维护关于 (m_k, h_{ksj}) 表 L_k, $1 \leq k \leq 7$. 关于 m_k 的询问,当 (m_k, h_{ksj}) 有记录响应 h_{ksj}. 不然随机选 h_{ksj} 记录并响应.

***Extract*(*ID*) 查询**:F 维护一个表 L_e 关于 $(ID_{sj}, h_{1sj}, Y_{sj}, S_{sj}, d_{sj}, RCID_{2e})$. 当 A 询问 (ID_{sj}, S_{sj}) 时,若 $(ID_{sj}, h_{1sj}, Y_{sj}, S_{sj}, d_{sj}, RCID_{2e})$ 在表 L_e 上,F 响应 $(Y_{sj}, d_{sj}, RCID_{2e})$. 不然 F 随机选择 $a, b \in Z_q^*$ 计算 $Y_{sj} = a \cdot P + b \cdot P_{RC2e}$,令 $d_{sj} \leftarrow amodq$, $h_{1sj} \leftarrow -bmodq$,然后把 $(ID_{sj}, h_{1sj}, Y_{sj}, S_{sj}, d_{sj}, RCID_{2e})$, $(ID_{sj}, h_{1sj}, Y_{sj}, S_{sj})$ 插入表 L_e 和 L_1 中. 最后发 $(Y_{sj}, d_{sj}, RCID_{2e})$ 给 A.

Update(**ID**,**t**)查询:响应过程与 $Extract(ID)$ 类似.

Corrupt(**ID**)查询:F 返回对应用户或者服务器的注册私钥和时间密钥.

Secret(**ID**)查询:当 $ID_{ui} \neq ID_{uI}, ID_{sj} \neq ID_{sJ}$,$F$ 返回对应用户或服务器的秘密值,不然取消游戏.

Send(**M**)查询:F 根据协议计算参数进行响应.

1. A 进行 $Send('Start')$ 查询,F 响应如下. 如果 $ID_{ui} \neq ID_{uI}$,F 根据协议响应. 不然,F 随机选取 $a \in Z_q^*$ 并计算 $M_{uI} = Q_1 + a \cdot P, S_{uI} = Q_1$,发送 $(M_{uI}, d_{tuI}, V_{uI}, t_1, ID_{uI}, S_{uI}, Y_{uI}, a, RCID_{2k})$ 给 A.

2. 当敌手 A 进行 $Send(F, (M_{ui}, d_{tui}, V_{ui}, t_1, ID_{ui}, S_{ui}, Y_{ui}, r_{ui}, RCID_{2k}))$,当 $ID_{sj} \neq ID_{sJ}$,F 根据协议验证响应. 当 $ID_{ui} \neq ID_{uI}$ 而 $ID_{sj} = ID_{sJ}$,F 首先检查请求的正确性,对正确的请求随机选取 $c \in Z_q^*$,计算 $M_{sJ} = Q_2 + c \cdot P, S_{sJ} = Q_2, R_{sJ} = (r_{ui} + s_{ui})(Q_2 + Y_{sJ} + H_1(ID_{sJ}, Y_{sJ} + Q_2) \cdot P_{RC2e})$ 和 $Auth_{sJ} = H_3(P_{pub}, P_{RC2e}, P_{RC2k}, ID_{ui}, ID_{sJ}, M_{ui}, M_{sJ}, R_{sJ})$,然后发 A 给以下消息 $(Auth_{sJ}, ID_{sJ}, M_{sJ}, d_{tsJ}, V_{sJ}, t_2, Y_{sJ}, S_{sJ}, c, RCID_{2e})$. 最后如果 $ID_{ui} = ID_{uI}$ 并 $ID_{sj} = ID_{sJ}$,F 随机选取 $c, d \in Z_q^*$,计算响应参数 $M_{sJ} = Q_2 + c \cdot P, S_{sJ} = Q_2, R_{sJ} = d \cdot P$ 和 $Auth_{sJ}$,然后发 $(Auth_{sJ}, ID_{sJ}, M_{sJ}, d_{tsJ}, V_{sJ}, t_2, Y_{sJ}, S_{sJ}, c, RCID_{2e})$ 给 A.

3. 当敌手 A 进行 $Send(F, (Auth_{sj}, ID_{sj}, M_{sj}, d_{tsj}, V_{sj}, t_2, Y_{sj}, S_{sj}, r_{sj}, RCID_{2e}))$,如果当 $ID_{ui} \neq ID_{uI}$,F 根据协议响应. 当 $ID_{sj} \neq ID_{sJ}$ 而 $ID_{ui} = ID_{uI}$,F 对合法的请求,计算 $R_{vsJ} = (d_{sj} + s_{sj}) \cdot M_{uI}$,验证 $Auth_{sj} = H_3(P_{pub}, P_{RC2e}, P_{RC2k}, ID_{uI}, ID_{sj}, M_{uI}, M_{sj}, R_{vsj})$ 是否成立,对于等式不成立的查询,F 拒绝这次会话. 不然计算 $K_{Ij} = (r_{sj} + s_{sj}) \cdot M_{uI}, R_{uI} = (r_{sj} + s_{sj})(Q_1 + Y_{uI} + H_1(ID_{uI}, Y_{uI} + Q_1) \cdot P_{RC2k})$, $Auth_{uI}$ 和 SK_{Ij},发 $Auth_{uI}$ 给 A. 如果 $ID_{ui} = ID_{uI}$ 并 $ID_{sj} = ID_{sJ}$,F 令 $R_{vsJ} = R_{sJ} = d \cdot P$ 验证 $Auth_{sJ} = H_3(P_{pub}, P_{RC2e}, P_{RC2k}, ID_{uI}, ID_{sJ}, M_{uI}, M_{sJ}, R_{vsJ})$ 是否成立,对于等式不成立的查询,F 拒绝这次会话. 不然其随机选取 $b, e \in Z_q^*$,计算参数 $R_{uI} = b \cdot P, K_{IJ} = e \cdot M_{sJ}$, $Auth_{uI}$ 和 SK_{IJ},发 $Auth_{uI}$ 给 A.

***Reveal*(*F*)**:如果 $ID_{ui} \neq ID_{uI} \vee ID_{sj} \neq ID_{sJ}$,若会话密钥已经协商完成,$F$ 返回对应的协商密钥. 不然返回 NULL.

事件 $ID_{ui} = ID_{uI}$ 和 $ID_{sj} = ID_{sJ}$ 都发生的概率为 $1/q_n q_m$,在这个事件下刚好对应的是 ID_{sJ} 的第 l_2 次实例与 ID_{uI} 的第 l_1 次实例会话的概率是 $1/q_s^2$. H_3 是一个随机预言机,根据对敌手 A 的假设,F 能够以概率 $1/q_3$ 在表 L_3 找到 R_{sJ} 作为解决 CDH 问题的方法,即

$$xyP = R_{sJ} - aQ_2 - d_{sJ}Q_1 - ad_{sJ}P$$

其中 d_{sJ} 可以通过 $Corrupt(F)$ 查询获得. 因此 F 能够以不可忽略的概率 $\varepsilon/q_n q_m q_s^2 q_3$ 解决 CDH 问题的一个实例. 从而根据定义 1,协议能够有效的抵抗冒充服务器认证攻击. 类似可以证明事件 $C2S$ 发生的概率仍是可忽略的. 对于撤销用户的冒充攻击,基于 *Schnorr* 签名的不可伪造性[15]能够抵抗撤销用户的冒充认证攻击.

定理 2 在随机预言机模型下,本文所提协议能达到协商密钥安全.

证明 假设和模拟过程与定理 1 类似,除了增加 $Test(F)$ 查询. 当 A 进行 $Test(F)$ 查询时,如果 $Test$ 查询对象是 ID_{sJ} 的 l_2 次实例与 ID_{uI} 的 l_1 次实例对应的会话(记为事件 B),F 随机在 $\{0,1\}$ 中选择一个 b. 如果 $b = 1$,F 返回会话密钥 SK,否则返回一个任意的比特串. 其他 $Test(F)$ 查询,取消游戏. 因此根据假设有

$$1/2 + \varepsilon = \Pr[\text{Asucceeds}]$$

$$= \Pr[\text{Asucceeds} \mid C2S \vee B]\Pr[C2S \vee B] + \Pr[\text{Asucceeds} \mid \overline{C2S \vee B}]\Pr[\overline{C2S \vee B}]$$

若 A 没冒充用户认证成功,再加之 H_4 是一个随机预言机,故 $\Pr[\text{Asucceeds} \mid \overline{C2S \vee B}] = 1/2$. 即

$$1/2 + \varepsilon \leq \Pr[\text{Asucceeds} \mid C2S \vee B]\Pr[C2S \vee B] + 1/2 \leq \Pr[C2S \vee B] + 1/2$$

化简有 $\Pr[B] \geq \varepsilon - \Pr[C2S]$,其中 $\Pr[C2S]$ 根据定理 1 是可忽略的,则事件 B 发生的概率不可忽略. 这意味着敌手 A 能够有效的计算 ID_{sJ} 的 l_2 次实例与 ID_{sI} 的 l_1 次实例的会话密钥. 从而 F 能够以概率 $1/q_4$ 在表 L_4 找到 K_{IJ},R_{isJ}或 R_{sI}作为解决 CDH 问题的方法. 因此,如果一个敌手 A 能够以不可忽略的概率 ε 猜对 b,那么 F 能够以不可忽略的概率 $3\Pr[B]/q_n q_m q_s^2 q_4 \geq 3(\varepsilon - \Pr[C2S])/q_n q_m q_s^2 q_4$ 解决 CDH 问题的一个实例. 故 ε 是可忽略的.

5　效率和安全对比

首先就实现的安全性和同类型方案进行对比.

表 1　安全性对比

方案	文献[3]	文献[2]	文献[10]	文献[11]	文献[12]	文献[13]	本文协议
*B*1	是	否	否	否	是	否	是
*B*2	– –	– –	否	否	否	否	是
*B*3	– –	– –	否	否	否	否	是
*B*4	– –	– –	否	否	否	是	是
*B*5	单	单	多	多	多	多	多

B1:是否可以抵抗 ESL 攻击. B2:是否可以抵抗客户端的注册信息泄露攻击. B3:是否可以抵抗服务器端的注册信息泄露攻击. B4:是否具有有效的撤销机制. B5:具体服务器架构. 注意‘ – – ’表示当前环境下这种攻击不存在.

正如表 1 所示,一些 MAKA 协议建立在 RC 绝对可信的基础上,容易造成 RCDRI 攻击. 本文结合注册私钥和秘密值来抵抗 RCDRI 攻击. 另一方面,对于安全性依赖于随机数的协议[2,4,9~11,13]来说易发生 ESL 攻击. 本文利用秘密值和 ECC 来抵抗 ESL 攻击.

基于 Miracl 库,本文分别在个人电脑端(酷睿三代 I7 处理器,主频 2.4GHz)和移动设备端(高通骁龙 801,主频 2.5GHz)进行了协议基本运算的时长测试. 测试结果如表 2 所示,测试的参数达到 AES128 安全级别.

利用表 2 中的数据来计算本文所提协议和最近同类型协议的运行时间. 需要指出由于 Tseng 等人[13]的协议为提高效率缓存认证重要凭证,分析发现这样的缓存有一定风险. 所以在计算 Tseng 等人的协议效率时,不考虑其缓存的情况.

表 2　对应运算操作的时长

	*T*1	*T*2	*T*3	*T*4	*T*5	*T*6
S	63.4ms	20.7ms	13.4ms	114.2ms	5.3ms	0.13ms
U	0.181s	0.055s	0.032s	0.273s	0.012s	0.001s

U:客户端测试结果. S:服务端测试结果. T1:执行一次双线性对运算所需要的时间. T2:执行一次基于双线对的椭圆曲线上的点乘所需的时间. T3:执行一次标准椭圆曲线上点乘所需的时间. T4:执行一次哈希映射到点所需的时间. T5:执行一次模指数运算所需的时间. T6:执行一次普通单向哈希函数所用的时间.

正如表 3 中所示,本文所提协议在服务器端具有很大的运算优势. 另一方面,由于没有双线对运算,本文的协议在客户端也具有一定的优势. 在注册结构方面,本文采取一种分级的多中心结构,将减轻 RC 的工作量,为系统整体增强稳定性.

表 3　协议执行时间对比

	文献[10]	文献[12]	文献[3]	文献[13]	本文协议
UP	0.237s	1.13s	0.496s	0.84s	0.229s
SP	194.59ms	438.26ms	460.52ms	333.59ms	94.45ms
AS	多	多	单	多	多
AR	单	单	– –	RC	RC

UP:客户端主要运算的时长和　SP:服务端主要运算的时长和
AS:服务器架构　AR:注册中心架构　注意’ – – ‘表示不存在.

6　结论

本文结合 SCPK 密码学技术提出了一个高效并且抵抗 ESL、RCDRI 攻击的多服务器环境下的 MAKA 协议. 并且本文的协议是一个拥有多个注册中心的方案. 通过安全性和效率分析,表明所提的协议具有较好的安全性和较高的运行效率.

参考文献

[1] Lamport L. Password authentication with insecure communication[J]. Communications of the ACM, 1981, 24(24): 770 – 772.

[2] He D. An efficient remote user authentication and key agreement protocol for mobile client-server environment from pairings[J]. Ad Hoc Networks, 2012, 10(6): 1009 – 1016.

[3] Tseng Y M, Huang S S, Tsai T T, et al. Anovel ID-based authentication and key exchange protocol resistant to ephemeral-secret-leakage attacks for mobile devices[J]. International Journal of Distributed Sensor Networks, 2015, 2015: 1 – 12.

[4] Choi K Y, Hwang J Y, Dong H L, et al. ID-based Authenticated Key Agreement for Low-Power Mobile Devices [M]. Information Security and Privacy, DBLP, 2005. 494

-505.
[5] Liao Y P, Wang S S. A secure dynamic ID based remote user authentication scheme for multi-server environment [J]. Computer Standards & Interfaces, 2009, 31 (1): 24-29.
[6] Hsiang H C, Shih W K. Improvement of the secure dynamic ID based remote user authentication scheme for multi-server environment[J]. Computer Standards & Interfaces, 2009, 31 (6): 1118-1123.
[7] Lee C C, Lin T H, Chang R X. A secure dynamic ID based remote user authentication scheme for multi-server environment using smart cards[J]. Expert Systems with Applications, 2011, 38(11): 13863-13870.
[8] Li X, Ma J, Wang W, et al. A novel smart card and dynamic ID based remote user authentication scheme for multi-server environments[J]. Mathematical & Computer Modelling, 2013, 58(s 1-2): 85-95.
[9] D Zhao, H Peng, et al. An efficient dynamic ID based remote user authentication scheme using self-certified public keys formulti-server environment, arXiv preprint arXiv: 1305.6350[OL]. http://arxiv.org/abs/1305.6350, 2013-05.
[10] Chuang Y H, Tseng Y M. Towards generalized ID-based user authentication for mobile multi-server environment [J]. International Journal of Communication Systems, 2012, 25(4): 447-460.
[11] Han W, Zhu Z. An ID-based mutual authentication with key agreement protocol for multi-server environment on elliptic curve cryptosystem [J]. International Journal of Communication Systems, 2015, 27(8): 1173-1185.
[12] Islam S H. Aprovably secure ID-based mutual authentication and key agreement scheme for mobile multi-server environment without ESL attack [J]. Wireless Personal Communications, 2014, 79(3): 1975-1991.
[13] Tseng Y M, Huang S S, Tsai T T, et al. List-free ID-based mutual authentication and key agreement protocol for multiserver architectures[J]. IEEE Transactions on Emerging Topics in Computing, 2016, 4(1): 102-112.
[14] Liao Y P, Hsiao C M. A novel multi-server remote user authentication scheme using self-certified public keys for mobile clients[J]. Future Generation Computer Systems, 2013, 29(3): 886-900.
[15] Fleischhacker N, Jager T, Schröder D. On Tight Security Proofs for Schnorr Signatures[A]. Advances in Cryptology-ASIACRYPT 2014[C]. Springer Berlin Heidelberg, 2014. 512-531.

作者简介

李雪莲 女,1979 年生人,副教授. 2010 年获得密码学博士学位. 研究方向为信息安全、密码函数与区块链.
E-mail: xuelian202@163.com

李 伟 男,1992 年生人,硕士研究生. 研究方向为认证协议设计与应用、网络安全.
E-mail: liwei3013@126.com