基于 RSA 的网关口令认证密钥交换协议的分析与改进

汪 定[1],王 平[1,2],雷 鸣[2]
(1.北京大学信息科学技术学院,北京 100871;2.北京大学软件与微电子学院,北京 102600)

摘 要: 设计安全高效的基于 RSA 的口令认证密钥交换协议是密码学领域的公开难题.2011 年 Wei 等学者首次提出了一个基于 RSA 的可证明安全的网关口令认证密钥交换协议,并声称在随机预言模型下基于大整数的素因子分解困难性证明了协议的安全性.利用该协议中服务器端提供的预言机服务,提出一种分离攻击,攻击者只需发起几十次假冒会话便可恢复出用户的口令.攻击结果表明,该协议无法实现所声称的口令保护这一基本安全目标,突出显示了分离攻击是针对基于 RSA 的口令认证密钥交换协议的一种严重安全威胁.进一步指出了协议形式化安全证明中的失误,给出一个改进方案.分析结果表明,改进方案在提高安全性的同时保持了较高效率,更适于移动通信环境.

关键词: 网关口令认证; RSA; 随机预言机模型; 分离攻击
中图分类号: TP309 **文献标识码:** A **文章编号:** 0372-2112 (2015)01-0176-09
电子学报 URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2015.01.028

Cryptanalysis and Improvement of Gateway-Oriented Password Authenticated Key Exchange Protocol Based on RSA

WANG Ding[1], WANG Ping[1,2], LEI Ming[2]
(1. *School of Electronics Engineering and Computer Science, Peking University, Beijing* 100871, *China*;
2. *School of Software and Microelectronics, Peking University, Beijing* 102600, *China*)

Abstract: It remains an open problem to design a secure and efficient RSA-based password-authenticated key exchange (PAKE) protocol in the areas of cryptography. In 2011, Wei proposed the first provably secure gateway-oriented PAKE protocol using RSA, and claimed that the protocol is provably secure in the random oracle model based on the intractability of the integer factorization problem. However, in this short paper, we point out that an adversary can launch the separation attack on their protocol by exploiting the oracle service unwittingly provided by the server, and a user's password can thus be guessed just after tens of malicious sessions. Our cryptanalysis result invalidates Wei's claim that their protocol can achieve the security goal of password protection, and highlights the damaging threat that separation attack poses to RSA-based PAKE protocols. Furthermore, we uncover the flaws in their formal security proof and put forward an enhancement to overcome the identified defect. The analysis results show that the improved protocol eliminates the vulnerability of Wei's protocol while keeping the merit of high performance, suitable for mobile application scenarios.

Key words: gateway-oriented password authentication; RSA; random oracle model; separation attack

1 引言

认证密钥协商(Authenticated Key Exchange, AKE)协议是两方或多方通过交互来证实对方身份,建立会话密钥,从而在公开信道中实现安全通信的基本手段.其中,基于口令的认证密钥协商(Password-Authenticated Key Exchange, PAKE)协议使得仅共享可记忆的低信息熵口令的用户也可以通过公开信道来协商一个高信息熵的会话密钥.由于 PAKE 协议不依赖于公钥基础设施的支撑,也不需要特殊硬件来存储高信息熵的共享密钥,因而具有广泛的应用前景,已成为近年来安全协议研究的热点之一[1].但是,相较传统的基于公钥或高信息熵主密钥的安全协议,PAKE 协议易遭字典攻击[2].依据攻击者对所猜测口令的验证方式,字典攻击主要分为离线字典攻击和在线字典攻击.

此外,基于 RSA 的 PAKE 协议易遭受一类特殊的字

收稿日期:2013-07-08;修回日期:2014-03-21;责任编辑::李勇锋
基金项目:国家自然科学基金(No.61472016)

典攻击——分割攻击(Partition Attack)[3].这种攻击将离线字典攻击和在线字典攻击结合起来,最典型的两种攻击形式是 e 次剩余攻击(e-Residue Attack)[4]和分离攻击(Separation Attack)[5].当前,针对分割攻击,只能从密码协议设计层面来防御,尚没有其它较好的非密码学手段.

近年来,鉴于 e 次剩余攻击的严重危害性,学者们对该攻击给予了广泛的关注,提出了众多能够抵抗 e 次剩余攻击的基于 RSA 的 PAKE 协议[4~9].另一方面,有关分离攻击的研究成果却较少,协议设计者们尚未充分认识到分离攻击的潜在危害性,常忽视对分离攻击的防御,如文献[6,7]均未充分考虑到分离攻击的潜在危害性.本文指出文献[8,9]中研究对分离攻击都是脆弱的,存在严重安全缺陷.为此,本文以文献[8]为例,突出显示分离攻击的严重危害性,为将来协议设计者提供重要参考和警鉴.

2011 年,Wei 等学者[8]指出现有基于网关的口令认证密钥协商(Gateway-oriented PAKE,GPAKE)协议都存在这样或那样的安全缺陷,首次提出了基于 RSA 的安全高效的 GPAKE 协议(简称 RSA-GPAKE),并宣称在随机预言机模型(Random Oracle Model,ROM)[10]下证明了协议的安全性.但本文分析发现,RSA-GPAKE 协议对分离攻击是脆弱的,主动攻击者可在非常有限的交互次数内恢复出合法用户的口令.因此,RSA-GPAKE 协议存在严重安全缺陷,无法实现所宣称的安全性,在对该缺陷进行修正前不适于实际应用.一个被形式化证明"安全"了的协议为什么结果是不安全的?

为解释这一矛盾现象,本文进一步分析了原协议形式化安全证明中的失误之处.为克服 RSA-GPAKE 协议中的上述严重安全缺陷,本文借鉴文献[6]中 PEKEP 协议防御"有用信息泄露"的思想,提出一个改进方案,并在 ROM 模型下给出了相应的严格归约证明(Rigorous Reduction Proof).本文中"方案"和"协议"表达相同的概念,交替使用.分析结果表明,改进方案在提高安全性的同时,保持了较高的效率,更适用实际移动应用环境.

2　安全模型

Wei 等人[8]的 RSA-GPAKE 协议安全模型基于 Abdalla 等人[11]在 2005 年提出的 GPAKE 安全模型,而 Abdalla 等人的模型又是 BPR2000 ROM 模型[10]在三方认证环境下的一个变形.本节对 GPAKE 安全模型进行简要回顾,并特别强调一些需要注意(容易产生误解)的地方,相关细节参见文献[11].

2.1　通信模型

网关口令认证密钥交换(GPAKE)协议是一个由用户、网关和服务器参与的三方协议,协议的目标是用户和网关在服务器的帮助下建立一个认证的会话密钥.用户和网关之间的通信信道是不安全的公开信道,网关和服务器之间的通信则存在安全信道.假设用户和服务器间事先共享一个低信息熵的口令,并且网关不知道口令的信息.GPAKE 协议的主要安全目标是通过服务器的参与,在用户和网关间建立一个安全的会话密钥,并且实现用户口令对网关的私密性、会话密钥对服务器的私密性.

协议的参与者　GPAKE 协议的参与者包括用户 $C\in\mathfrak{C}$、网关 $G\in\mathfrak{G}$ 和服务器 $S\in\mathfrak{S}$.用集合 $\mathfrak{U}$ 表示所有参与者的集合,即 $\mathfrak{U}=\mathfrak{C}\cup\mathfrak{G}\cup\mathfrak{S}$,用 U 表示 $\mathfrak{U}$ 中的一个任意参与者.

长期密钥　每个用户 C 拥有口令 $pw_C\in\mathfrak{D}$,其中 $\mathfrak{D}$ 是口令空间;每个服务器 $S\in\mathfrak{S}$ 保存一个口令列表 $pw_S=\{pw_C\}_{C\in\mathfrak{C}}$,每个用户的口令都是 pw_S 中的一条记录.pw_C 和 pw_S 即为用户和服务器的长期密钥.为便于分析,一般假设字典空间 $\mathfrak{D}$ 服从均匀分布.即使用户口令空间不是均匀分布,本文中的相关安全定义也可容易进行相应扩展[1].

2.2　攻击者能力

攻击者 $\mathfrak{A}$ 与协议参与方 $U\in\mathfrak{C}\cup\mathfrak{G}\cup\mathfrak{S}$ 间的交互通过预言查询来模拟,以此来模拟现实中攻击者的能力.用 U^i 表示用户 U 的第 i 个实例.在协议的运行过程中,$\mathfrak{A}$ 可针对某个参与者 U 产生多个并行的会话实例.赋予 $\mathfrak{A}$ 的查询类型主要有:

—Execute(C^i,G^j):这个查询模拟攻击者的被动攻击(窃听)的能力,输出为协议正常运行情况下 C^i 与 G^j 的交互信息;

—Send(U^i,m):这个查询模拟攻击者的主动攻击,$\mathfrak{A}$ 向实例 U^i 发送消息 m,U^i 根据协议规定向 $\mathfrak{A}$ 返回处理 m 所产生的相应的信息.其中,Send(U^i,Start)发起一个会话;

—Test(U^i):这个查询不是模拟攻击者的攻击能力,而是用来定义会话密钥的语义安全性,只对"新鲜"的会话有效.如果实例 U^i 的会话密钥尚没有定义,返回 $\perp$.否则,随机选择一个比特 b,如果 $b=1$,向 $\mathfrak{A}$ 返回会话密钥 sk;如果 $b=0$,向 $\mathfrak{A}$ 返回一个等长的随机串;

—TestPair(C^i,G^j):这个查询不是模拟攻击者的攻击能力,而是用来刻画会话密钥对于服务器的私密性.如果用户实例 C^i 和网关实例 G^j 之间还没有建立共享的会话密钥,返回 $\perp$.否则,随机选择一个比特 b,如果 $b=1$,向 $\mathfrak{A}$ 返回会话密钥 sk;如果 $b=0$,向 $\mathfrak{A}$ 返回一个等长的随机串;

RSA-GPAKE 协议会话密钥的语义安全性通过 Real-Or-Random(ROR)攻击游戏[12]而非传统的 Find-Then-Guess(FTG)来模型化.在 FTG 攻击游戏中,除上述三个

查询外,还使用了 Reveal 查询,$\mathfrak{A}$ 只能进行一次 Test 查询.在 ROR 攻击游戏中,敌手可以进行多次 Test 查询,这些查询的返回结果仍由单个比特 b 来决定.$\mathfrak{A}$ 在两种模型下的目标都是猜测 Test 询问中使用的随机比特 b 的值,如果 $\mathfrak{A}$ 猜测正确则称 $\mathfrak{A}$ 成功破坏了会话密钥 sk 的语义安全性.不难看出,ROR 模型下 $\mathfrak{A}$ 的攻击能力要比 FTG 模型下强,关于二者的更详细比较可参见文献[12].

2.3 安全目标

本小节给出安全模型中相关概念和四个基本安全目标的形式化定义.

会话标识 一般定义会话标识为协议执行结束后所有发送和接收消息的有序级联,记为 sid.

接受/拒绝 如果一个实例完成协议运行并且生成会话密钥,则称该实例接受,否则称为拒绝.

伙伴关系(匹配会话) 我们基于会话标识符 sid 来定义伙伴关系.设 C^i 和 G^j 为两个实例,如果下述条件满足,则为一对伙伴:这两个实例均已接受;这两个实例拥有相同的会话标识符 sid;C^i 的伙伴标识符 $pid_C^i = G^j$,G^j 的伙伴标识符 $pid_G^j = C^i$;不存在其它的实例 U^k,有 $pid_U^k = C^i$ 或 G^j.

新鲜性 新鲜性的概念用来描述这样一个直观事实:会话密钥不能轻易被 $\mathfrak{A}$ 所获知.敌手只能对新鲜的会话密钥进行 Test 询问.如果 U^i 已接受且计算出会话密钥 sk,则认为 U^i 是新鲜的.

Abdalla 等人[11]的 GPAKE 协议安全模型中有四个安全目标,包括语义安全性、认证性、密钥私密性和口令保护.需要特别指出的是,前两项是所有 PAKE 协议所应实现的基本目标[10],后两项是 GPAKE 协议所独有的目标.这四个目标各自独立,都应当予以充分考虑.Wei 等在他们的工作[8,9]中均遗漏了"认证性"这一重要目标,认为 Abdalla 等的安全模型中只有三个安全目标.文献[13]清晰的展示了,一个实现了语义安全性的协议却没有实现认证性,无法抵抗用户仿冒攻击.

会话密钥的语义安全性保障的是,一个外部攻击者不能够多项式时间内将真实的会话密钥和与之等长的随机串区分开来;认证性要求攻击者不能仿冒协议中的真实实体;密钥私密性要求用户和网关之间建立的会话密钥对于诚实而好奇的服务器是不可区分的(专门针对服务器);口令保护是指恶意网关通过协议运行不能得到用户口令的任何信息(专门针对网关).由于篇幅所限,这里仅给出"口令保护"这一安全目标的严格定义.

口令保护 GPAKE 协议要求网关不能得到用户口令的任何信息,但任意攻击者 $\mathfrak{A}$(包括外部攻击者和内部恶意网关)总可以在一次假冒会话中排除掉一个口令,因此我们的目标是使 $\mathfrak{A}$ 的成功优势 $\mathrm{Adv}_{\mathfrak{P},\mathfrak{D}}^{\text{ake-uoda}}(\mathfrak{A})$ 不会明显高于这个值.如果对任意的概率多项式时间攻击者 $\mathfrak{A}$,$\mathfrak{A}$ 的优势

$$\mathrm{Adv}_{\mathfrak{P},\mathfrak{D}}^{\text{ake-uoda}}(\mathfrak{A}) \leqslant \frac{\lambda t}{|\mathfrak{D}|} + \varepsilon(k)$$

其中 t 为 $\mathfrak{A}$ 发起主动攻击的次数,$|\mathfrak{D}|$ 为口令空间的规模,λ 是一个常数,$\varepsilon(k)$ 为关于系统安全参数 k 的一个可忽略的量,则说协议 $\mathfrak{P}$ 实现了口令保护.

3 RSA-GPAKE 协议的安全性分析

3.1 RSA-GPAKE 协议回顾

本节对 RSA-GPAKE 协议[8]进行简要回顾.协议所使用的符号及其含义如表 1 所示.

协议的参与者包括用户 C、认证服务器 S 和网关 G.在初始化阶段,用户 C 和服务器 S 共享一个从口令空间 $\mathfrak{D}$ 中随机选取的口令 pw.根据 RSA 体制,用户 C 产生公钥 e 和私钥 d,其中 $ed = 1 \bmod \varphi(n)$,$n > 2^{1023}$,e 与 $\varphi(n)$ 互素.类似,网关 G 也生成 RSA 参数 n',e' 和 d'.此外,为抗 e 次剩余攻击要求用户 C 的公钥 e 为奇素数,而对网关 G 的公钥 e 则无此限制.对于 $m \in \mathbb{Z}_n^*$,定义 $E^m(x) = E^{m-1}(E(x)) = x^{e^m} \bmod n = c$,相应地 $D^m(c) = x$.协议假设在网关 G 和服务器 S 间存在安全信道,协议运行具体如下:

表 1 符号定义

符号	含义
C, S, G	用户,认证服务器,网关
$pw, \mathfrak{D}$	用户 C 的口令,口令空间
e, d	RSA 密码体制公钥,私钥
$E(\cdot), D(\cdot)$	RSA 加密函数,解密函数
k	系统安全参数,如 $k = 160$
$\varphi(\cdot)$	欧拉函数
$\|$	比特连接运算
$H(\cdot)$	安全 Hash 函数 $\{0,1\}^* \to \mathbb{Z}_n$
$H_1(\cdot), H_2(\cdot), H_3(\cdot)$	安全 Hash 函数 $\{0,1\}^* \to \{0,1\}^k$
$A \to B: M$	将消息 M 通过普通信道由 A 传送到 B
$A \Rightarrow B: M$	将消息 M 通过安全信道由 A 传送到 B

协议的参与者包括用户 C、认证服务器 S 和网关 G.在初始化阶段,用户 C 和服务器 S 共享一个从口令空间 $\mathfrak{D}$ 中随机选取的口令 pw.根据 RSA 体制,用户 C 产生公钥 e 和私钥 d,其中 $ed = 1 \bmod \varphi(n)$,$n > 2^{1023}$,e 与 $\varphi(n)$ 互素.类似,网关 G 也生成 RSA 参数 n',e' 和 d'.此外,为抗 e 次剩余攻击要求用户 C 的公钥 e 为奇素数,而对网关 G 的公钥 e 则无此限制.对于 $m \in \mathbb{Z}_n^*$,定义 $E^m(x) = E^{m-1}(E(x)) = x^{e^m} \bmod n = c$,相应地

$D^m(c)=x$.协议假设在网关 G 和服务器 S 间存在安全信道,协议运行具体如下:

1)用户 C 选择 $r_1\in_R\{0,1\}^k$;

2)$C\rightarrow G:\{C,n,e,r_1\}$

3)$G\Rightarrow S:\{C,n,e,r_1,n',e'\}$

4)服务器 S 在收到来自网关的认证请求消息后,检查 e 是否为奇素数,n 是否为奇数以及 n 是否强度足够,如 $n>2^{1023}$.如果检测未通过,则服务器 S 拒绝;否则服务器 S 计算 $m=\lfloor\log_e n\rfloor$,选择 $a_1\in_R\mathbb{Z}_n^*$,$r_2\in_R\{0,1\}^k$,计算 $\alpha=H(pw\|r_1\|r_2\|C\|G\|n\|e\|n'\|e')$.

5)S 检查 $\gcd(\alpha,n)=1$ 是否成立,若不成立,则拒绝认证请求.否则,S 计算 $z=E^m(\alpha E(a_1))$.

6)$S\Rightarrow G:\{r_2,z\}$

7)$G\rightarrow C:\{G,n',e',r_2,z\}$

8)用户 C 在收到来自网关的响应消息后,首先检查 n' 是否是足够大的奇数,如果不是则拒绝;否则用户 C 计算 $\alpha=H(pw\|r_1\|r_2\|C\|G\|n\|e\|n'\|e')$.

9)用户 C 验证 $\gcd(\alpha,n)=1$ 是否成立.如果 $\gcd(\alpha,n)\neq1$,则拒绝;否则,用户 C 解密 z 得到 $a_1=D(\alpha^{-1}D^m(z))$,选择 $b_1\in_R\mathbb{Z}_{n'}^*$,计算 $c_1=b_1^{e'}\bmod n'$ 和 $\mu=H_1(a_1\|C\|G\|n\|e\|n'\|e'\|r_1\|r_2\|z\|c_1)$.

10)$C\rightarrow G:\{C,c_1,\mu\}$

11)网关 G 收到来自用户 C 的消息后,选择 $b_2\in_R\mathbb{Z}_n^*$,计算 $c_2=b_2^e\bmod n$.

12)$G\Rightarrow S:\{C,c_1,c_2,\mu\}$

13)服务器 S 在收到 $\{C,c_1,c_2,\mu\}$ 后,验证 μ 是否有效.如果 μ 无效则拒绝,否则计算

$\eta=H_2(a_1\|C\|G\|n\|e\|n'\|e'\|r_1\|r_2\|z\|c_1\|c_2)$.

14)$S\Rightarrow G:\{\eta\}$

15)网关 G 用自己的私钥 d' 解密 c_1 得到 $b_1=c_1^{d'}\bmod n'$,计算会话密钥 $sk=H_3(b_1\|b_2\|\mathrm{SID})$并接受会话,其中 SID 表示用户 C 和网关 G 间所有会话消息的级联.

16)$G\rightarrow C:\{G,\eta,c_2\}$

17)用户 C 收到$\{\eta,c_2\}$后验证 η 是否有效.如果无效则拒绝,否则用私钥 d 解密 c_2 得到 $b_2=c_2^d\bmod n$,计算会话密钥 $sk=H_3(b_1\|b_2\|\mathrm{SID})$并接受会话.

3.2　分离攻击

本节所给出的分离攻击针对的是 RSA-GPAKE 协议的“口令保护”这一基本安全目标,该攻击只需要攻击者 $\mathfrak{A}$(外部攻击者或内部恶意网关)有如下两种能力:

(1)$\mathfrak{A}$ 可以窃听、阻断、删除和窜改流经公开网络中的任何消息,也可以插入伪造的消息;

(2)$\mathfrak{A}$ 可以多项式时间内穷举搜索口令空间 $\mathfrak{D}$;

不难看出,$\mathfrak{A}$ 的上述两个能力假设可由节 2 中的安全模型直接得出.其中,假设 1 遵循 Dolev-Yao 标准威胁模型[14],可由节 2 中的 Execute 和 Send 两种查询来模拟;假设 2 显示了用户口令“低信息熵”的本质特征[15,16].上述两个攻击者能力假设是分析 PAKE 协议安全性的基本假设[2,3],也是所有 PAKE 协议面临的共同难题.

基于上述两个能力假设,攻击者 $\mathfrak{A}$(任意外部攻击者)可在多项式时间恢复出用户的口令,攻破“口令保护”这一基本目标,具体过程描述如下:

1)攻击者 $\mathfrak{A}$ 选取奇素数 $e_{\mathfrak{A}}$ 和大奇数 $n_{\mathfrak{A}}>2^{1023}$,其中 $n_{\mathfrak{A}}=3p$,p 是 $\mathfrak{A}$ 所选取的大素数;

2)$\mathfrak{A}$ 选取 $r_1\in_R\{0,1\}^k$;

3)$\mathfrak{A}\rightarrow G:\{C,n_{\mathfrak{A}},e_{\mathfrak{A}},r_1\}$

4)$\mathfrak{A}$ 如果未收到来自网关 G 的响应消息,则返回步骤 2).否则,$\mathfrak{A}$ 收到来自网关 G 的响应消息$\{G,n',e',r_2,z\}$,称该次会话为“成功测试会话”,$\mathfrak{A}$ 结束此次会话,然后离线执行下面步骤:

5)$\mathfrak{A}$ 从口令空间 $\mathfrak{D}$ 选取一个口令 pw^*;

6)$\mathfrak{A}$ 计算 $\alpha^*=H(pw^*\|r_1\|r_2\|C\|G\|n_{\mathfrak{A}}\|e_{\mathfrak{A}}\|n'\|e')$,需要注意的是,输入 Hash 函数 $H(\cdot)$的所有相关参数已为 $\mathfrak{A}$ 所知.

7)$\mathfrak{A}$ 验证 $\gcd(\alpha^*,n_{\mathfrak{A}})=1$ 是否成立,若不成立,则将 pw^* 从 $\mathfrak{D}$ 中删除;否则,将 pw^* 保留在 $\mathfrak{D}$ 中.转步骤(5).

在上述攻击的步骤 4 中,$\mathfrak{A}$ 收到来自网关 G 的响应消息意味着 $\gcd(\alpha^*,n_{\mathfrak{A}})=1$,因此 $\mathfrak{A}$ 可以在步骤(7)中通过检验 $\gcd(\alpha^*,n_{\mathfrak{A}})=1$ 是否成立来验证所猜测的口令的正确性.需要指出的是,在 RSA-GPAKE 协议中,对于用户传送过来的 n,服务器 S 只检测其奇偶性和强度,故上述攻击中只要 $n_{\mathfrak{A}}$ 为奇数且足够大即可通过服务器的检测,如令 $n_{\mathfrak{A}}=3p,15p,65537p$.不失一般性,本文以 $n_{\mathfrak{A}}=3p$ 为例.此外,还需要注意的是,计算 $\gcd(\cdot)$的扩展欧几里德算法是多项式时间确定性算法,无论 $\mathfrak{A}$ 是否知道 $n_{\mathfrak{A}}$ 的分解方法,$\mathfrak{A}$ 计算 $\gcd(\alpha^*,n_{\mathfrak{A}})$ 都是容易的.

设“口令 pw^* 使 $\gcd(\alpha^*,n_{\mathfrak{A}})\neq1$ 成立”的事件为 E,其中 $\alpha^*=H(pw^*\|r_1\|r_2\|C\|G\|n_{\mathfrak{A}}\|e_{\mathfrak{A}}\|n'\|e')$.

由于 $H(\cdot)$是随机预言机,则口令空间 $\mathfrak{D}$ 中任一口令 pw 使 $\gcd(\alpha^*,n_{\mathfrak{A}})\neq1$ 成立的概率均为 $\Pr[\mathrm{E}]$,其中 $\alpha=H(pw\|r_1\|r_2\|C\|G\|n_{\mathfrak{A}}\|e_{\mathfrak{A}}\|n'\|e')$.由 $n_{\mathfrak{A}}=3p$ 可得

$$\Pr[\mathrm{E}]=\Pr[(\alpha^*\bmod3=0)\vee(\alpha^*\bmod p=0)]$$
$$=\frac{1}{3}+\frac{1}{p}-\frac{1}{3p}=\frac{1}{3}+\frac{2}{3p}\approx\frac{1}{3}$$

这意味着,运行一次上述攻击过程,将有不少于三分之一的口令从 $\mathfrak{D}$ 中被删除,约三分之二的口令仍保留在 $\mathfrak{D}$ 中.为使 $\mathfrak{D}$ 中剩余唯一一个口令(该口令即为用户 C 的正确口令),则需平均进行 l 次“成功测试会话”,其中$(\frac{2}{3})^l\cdot|\mathfrak{D}|=1$.现实中,为了方便记忆,用户自主选择的口令往往是弱口令[15].因此,用户口令空间 $\mathfrak{D}$ 通常十分有限,如 $|\mathfrak{D}|=10^6$[16],由 $(\frac{2}{3})^l\cdot|\mathfrak{D}|=$

$(\frac{2}{3})^l \cdot 10^6 = 1$ 可计算

$$l = \log_{\frac{3}{2}} |\mathfrak{D}| = \log_{\frac{3}{2}} 10^6 \approx 34.07$$

又由于攻击者 $\mathfrak{A}$ 每次发起的攻击会话有三分之二的概率成为“成功测试会话”,故平均需要攻击者 $\mathfrak{A}$ 发起 $\frac{3}{2}l$ 次假冒会话便可恢复出用户 C 的口令.由此可知,$\mathfrak{A}$ 平均需要发起 $\frac{3}{2}l = \frac{3}{2} \times 34.07 < 52$ 次假冒会话便可成功恢复出用户的口令.

同理,可得 $\mathfrak{A}$ 恢复出用户 C 的口令的优势为

$$\mathrm{Adv}^{\text{ake-uoda}}_{\mathfrak{A},\mathfrak{D}}(\mathfrak{A}) = 1 - (\frac{2}{3})^t > \frac{\lambda t}{|\mathfrak{D}|} + \varepsilon(k)$$

其中 $t < 52$ 为 $\mathfrak{A}$ 主动发起的假冒会话次数,λ 是一个小常数[11],$\varepsilon(k)$是关于 k 的一个可忽略的量.这显然违反了节 2 中定义的 GPAKE 协议的“口令保护”这一安全目标.需要注意的是,上述给出的攻击者 $\mathfrak{A}$ 是任意外部攻击者.实际上,$\mathfrak{A}$ 也可以是恶意网关 G 本身,此时恶意网关采用的攻击方法与算法 1 类似,只需将算法 1 的 Step 5 中 Send(G^j, $< C^i$, $n_{\mathfrak{A}}$, $e_{\mathfrak{A}}$, $r_1 >$)替换为 Send(S^k, $< C^i, n, e, r_1, n', e' >$)即可.

3.3 防御方法

不难发现,在节 3.2 的分离攻击中,若令 $n_{\mathfrak{A}} = 65537p$,则攻击者可在一次“成功测试会话”后排除约 $\frac{1}{65537}$的口令.相应地,一种防御上述攻击的办法就是禁止 n 含小因子:服务器 S 端,如果检测到 n 含有小因子,则拒绝.但这种办法并不理想.一方面如果对“小因子”界定得较小,$\mathfrak{A}$ 仍可在一次“成功测试会话”后排除较多的口令.比如,假设系统检测 n 不含小于 1000 的因子,此时攻击者可使 $n_{\mathfrak{A}} = 1009p$,通过一次假冒会话仍可排除$\frac{1}{1009}$的口令.另一方面,如果对“小因子”界定得较大,在服务器 S 端需要进行大量的因子测试运算,会大大增加服务器的负载.此外,还需要特别指出的是,通过限制 $\mathfrak{A}$ 发起假冒会话次数(总次数或连续假冒会话次数)的方式都不能有效防御上述攻击.

3.4 协议形式化安全证明中的失误

文献[8]采用了随机预言机模型(ROM)[10]对 RSA-GPAKE 协议进行用户实例以及攻击者能力的形式化,提出了会话密钥语义安全性、会话密钥私密性和口令保护这三个安全目标,并给出了相应的形式化证明.现在,矛盾现象产生了:一个被形式化证明“安全”了的协议为什么会是不安全的(“Provably secure, but actually insecure”)?

文献[8]完整的给出了对“语义安全性”这一目标的证明,并在此基础上论证了“密钥私密性”这一目标.但在证明其“口令保护”这一目标时(见文献[8]附录 2,Wei 等人在给出“口令保护”相应的形式化结果,即文献[8]中定理 3 时,误将 $\mathrm{Adv}^{\text{ake-uoda}}_{\mathrm{p},\mathfrak{D}}(\mathfrak{A})$写为 $\mathrm{Adv}^{\text{ake-ror}}_{\mathrm{p},\mathfrak{D}}(\mathfrak{A})$),并没有采用严格的推理将攻击者对口令的猜测攻击归约到求解数学难题上(如 RSA-GPAKE 所基于的 RSA 困难性假设),而是在关键点下了轻率的结论——“恶意网关想要成功进行不可检测在线字典攻击,必须返回一个有效的认证值 μ.如果恶意网关不知道 α(即本文中的 a_1),那么成功的概率至多为 2^k.”然而,节 3.2 中我们所给出的分离攻击表明,外部攻击者 $\mathfrak{A}$(或恶意网关)向服务器 S 发送认证请求消息(即$\{C, n_{\mathfrak{A}}, e_{\mathfrak{A}}, r_1\}$)后,根本不需要返回认证值 μ,只需根据服务器 S 响应与否就可以排除约 1/3 的口令.因此,上述非严格的证明导致了整个结论的失效.

不难看出,节 3 中的分离攻击能够成功的根本原因在于,攻击者 $\mathfrak{A}$ 利用了服务器 S 所提供的互素判别预言机服务——服务器 S 对 $\gcd(\alpha, n) = 1$ 是否成立作出反应,并且该反应能够被 $\mathfrak{A}$ 所区分.相应地,防御该攻击的最直接有效办法就是去除服务器 S 所提供的互素判别预言机服务——虽然服务器 S 对 $\gcd(\alpha, n) = 1$ 是否成立作出反应,但该反应无法被攻击者 $\mathfrak{A}$ 所区分.

4 对 RSA-GPAKE 协议的改进与分析

4.1 改进方案 RSA-GPAKE +

本节根据节 3.4 中对 RSA-GPAKE 协议形式化证明失误原因的分析,借鉴 PEKEP 协议中防止“有用信息泄露”的思想,遵循有效、简单的原则,对 RSA-GPAKE 协议进行如下修改(修正后的协议如图 1 所示,改进部分用虚框标注,其余与 RSA-GPAKE 协议相同).

在改进协议 RSA-GPAKE + 中,如果攻击者 $\mathfrak{A}$ 发起如节 3 中所给出的假冒会话,无论 $\gcd(\alpha, n) \neq 1$ 是否成立,$\mathfrak{A}$ 都将会收到来自服务器 S 的响应消息$\{r_2, z\}$,$\mathfrak{A}$ 无法得到有关 $\gcd(\alpha, n) \neq 1$ 是否成立的任何有用信息.另一方面,在收到响应消息$\{r_2, z\}$后,$\mathfrak{A}$ 如果不能正确猜测出用户 C 的口令 pw,将无法计算出 S 所产生的 α 的值,进而无法得到 S 所产生的随机数 a_1 的值,也就无法计算出合法的 μ,进而 S 必然会对 $\mathfrak{A}$ 的此次假冒会话进行拒绝.因此,改进协议不会泄露口令 pw 的相关信息,可有效抵抗节 3 中的分离攻击,具体的形式化证明过程见下节.

根据节 4.1 中协议 RSA-GPAKE + 的描述,不难看出,当 $\gcd(\alpha, n) = 1$ 时,RSA-GPAKE + 与原协议完全相同,没有增加任何计算量和通信量;当 $\gcd(\alpha, n) \neq 1$ 时,相较原协议的直接拒绝,RSA-GPAKE + 采取“警惕性”继续执行策略,这样仅在用户端和服务器端各引入一次 Hash 运算和两次随机数生成等轻量级运算,在网

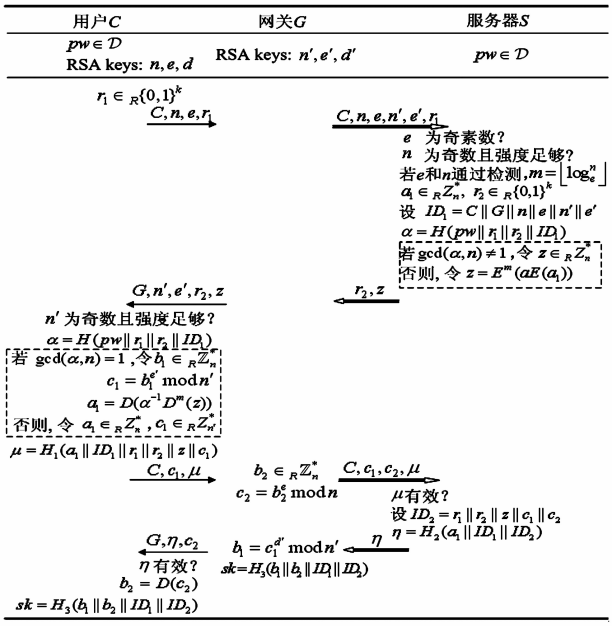

图1　本文改进协议RSA-GPAKE+

关 G 没有引入新的计算量.并且,无论 $\gcd(\alpha,n)=1$ 与否,改进协议均未引入新的通信量.因此,改进协议保持了较高效率.

4.2　安全性证明

由于改进方案 RSA-GPAKE+ 基于 Wei 等人的 RSA-GPAKE 协议[8]并继承了该协议所具有的安全性,例如实现会话密钥语义安全性和私密性等,本小节我们仅对增强的安全特性——"口令保护"进行分析,其它安全目标的证明与文献[8]类似,这里不再赘述.我们利用节 2 中介绍的安全模型,基于 RSA 困难性假设,给出"口令保护"这一目标的形式化证明.下面首先介绍 RSA.假设:

RSA 假设　设 k 是系统安全参数,$\mathfrak{IG}$ 为 RSA 生成器,即运行 $\mathfrak{IG}$ 有 $(p,q,e,d,n)\leftarrow\mathfrak{IG}(1^k)$,其中 $n=pq$,大素数 p,q 的规模相同,$\gcd(e,\varphi(n))=1$ 并且有 $ed=1\bmod\varphi(n)$.当 k 充分大时,对于任意概率多项式时间的攻击者 $\mathfrak{A}$,其优势

$$\mathrm{Adv}^{\mathrm{rsa}}(\mathfrak{A})=\Pr[m\leftarrow\mathfrak{A}(n,e,c=m^e(\bmod n))]\leqslant\varepsilon(k)$$

其中,$\varepsilon(k)$是一个可忽略的量.

定理 1(口令保护)　假设 $\mathfrak{A}$ 是一个运行时间为 t,并且进行了 q_{send}次 Send 询问、q_{exe}次 Execute 询问、q_{oh}次 Hash 询问的概率多项式时间攻击者.$\mathfrak{A}$ 对 RSA-GPAKE+协议(简记为 $\mathfrak{P}$)进行不可检测在线字典攻击成功的优势为

$$\mathrm{Adv}^{\text{ake-uoda}}_{\mathrm{p},\mathfrak{D}}(\mathfrak{A})\leqslant\frac{q_{\mathrm{send}}}{|\mathfrak{D}|}+\frac{q_{\mathrm{send}}}{2^k}+\frac{q_{\mathrm{oh}}}{\varphi(n)}+\frac{q_{\mathrm{send}}}{\varphi(n)}+\frac{(q_{\mathrm{send}}+q_{\mathrm{exe}})^2}{2\varphi(n)}+\frac{q_{\mathrm{oh}}^2}{2^{k+1}}+(q_{\mathrm{send}}+q_{\mathrm{exe}})\mathrm{Adv}^{\mathrm{rsa}}(O(t))$$

证明　设 $\mathfrak{A}$ 为试图破坏协议"口令保护"这一安全目标的攻击者,由于服务器 S 是可信实体,$\mathfrak{A}$ 可假冒的角色(实体)仅为用户 C 和网关 G.证明的主要思路是,利用 $\mathfrak{A}$ 来构建破坏密码原语的攻击者,如果 $\mathfrak{A}$ 成功破坏协议的"口令保护"安全性,则至少有一个密码原语(如 RSA 假设)被攻破.我们使用与文献[1,7]类似的技巧,定义一系列的混合仿真游戏($\mathrm{Game}_0,\mathrm{Game}_1,\cdots\cdots,\mathrm{Game}_6$),从一个真实的攻击 Game_0开始,在后续的游戏中逐步修改预言机的回答方式,最后以一个 $\mathfrak{A}$ 的优势为零的游戏 Game_6 结束.在所有的游戏中,预言机按照协议的描述处理查询.在每个游戏 $\mathrm{Game}_n(n=0,1,2,\cdots\cdots,6)$中,我们定义下述事件:

—Succ_n:$\mathfrak{A}$ 成功猜测出用户 C 的口令 pw;

—$\mathrm{AskPara}_n$:$\mathfrak{A}$ 通过对 $pw\parallel r_1\parallel r_2\parallel ID_1$ 查询 H,从而成功计算出核心安全参数 α;

—$\mathrm{AskAuth}_n$:$\mathfrak{A}$ 成功计算出由服务器 S 产生的随机参数 a_1,并且对 $a_1\parallel ID_1\parallel r_1\parallel r_2\parallel z\parallel c_1$ 查询 H_1,或对 $a_1\parallel ID_1\parallel ID_2$ 查询 H_2;

—AskH_n:$\mathfrak{A}$ 正确地查询了随机预言机,即 $\mathfrak{A}$ 对 $a_1\parallel ID_1\parallel r_1\parallel r_2\parallel z\parallel c_1$ 查询 H_1,或者对 $a_1\parallel ID_1\parallel ID_2$ 查询 H_2,或者对 $b_1\parallel b_2\parallel ID_1\parallel ID_2$ 查询 H_3;

Game_0:该游戏对应 ROM 下的真实攻击,根据定义有

$$\mathrm{Adv}^{\text{ake-uoda}}_{\mathrm{p},\mathfrak{D}}(\mathfrak{A})=\Pr[\mathrm{Succ}_0]\tag{1}$$

Game_1:本游戏中,我们正常模拟所有的 Hash 查询 H、H_i($i=1,2,3$)以及将在 Game_5 中出现的 H_i'($i=1,2,3$)、Game_6 中出现的 H',维护一个 Hash 查询结果列表 $\Lambda_{\mathfrak{H}}$(和 $\Lambda_{\mathfrak{A}}$,该列表跟踪由 $\mathfrak{A}$ 所直接实施的 Hash 查询).对于 Hash 查询的每个输入 x,H_i(或 H)首先检查是否曾经被查询过:检查列表 $\Lambda_{\mathfrak{H}}$ 里面是否已有 $x_i=x$.如果存在 x_i,则返回对应的 y_i 作为 Hash 查询的输出;否则,随机预言选择一个随机数 y 作为对 x 的回答,并且把(x,y)这条记录添加到 $\Lambda_{\mathfrak{H}}$ 列表中(如果是由 $\mathfrak{A}$ 询问的,同时还要把这条记录添加到 $\Lambda_{\mathfrak{A}}$ 列表中).对于其它查询,保持与真实协议攻击不变.不难看出,

$$\Pr[\mathrm{Succ}_1]-\Pr[\mathrm{Succ}_0]=0\tag{2}$$

Game_2:为便于分析,本游戏中我们去掉一些不太可能发生的碰撞:

—通信消息$((C,n,e,r_1),(n',e',r_2,z),(C,c_1,\mu),(c_2,\eta))$的碰撞,需要指出的是,产生这些通信消息的参与方中至少有一个是诚实实体,故 r_1 和 r_2 中至少有一个是随机分布的,进而 μ 和 η 中至少有一个是随机分布的;

—Hash 输出的碰撞;

根据生日悖论原理,可得

$$\Pr[\mathrm{Succ}_2]-\Pr[\mathrm{Succ}_1]\leqslant\frac{(q_{\mathrm{send}}+q_{\mathrm{exe}})^2}{2\varphi(n)}+\frac{q_{\mathrm{oh}}^2}{2^{k+1}} \quad (3)$$

为便于后文分析,在本游戏中进一步去掉 $\gcd(\alpha,n)\neq 1$ 的情形.由于当服务器 S(可信实体)发现 $\gcd(\alpha,n)\neq 1$ 时,z 被设置成一个与用户 C 的口令 pw 完全无关$\mathbb{Z}_n^*$ 中一个随机值.此时,即使用正确的 α 去解密 z 仍会得到一个与 pw 完全无关的随机值.下面分三种情况讨论:

—$\mathfrak{A}$ 是被动攻击者.此时协议交互的所有消息中均不含口令的任何有效信息,$\mathfrak{A}$ 无法验证所猜测口令 pw 的正确性.

—$\mathfrak{A}$ 仿冒网关 G.此时协议交互的所有消息中均不含口令的任何有效信息,$\mathfrak{A}$ 与被动攻击者面临的情况相同.

—$\mathfrak{A}$ 仿冒用户 C.假设 $\mathfrak{A}$ 猜测出了正确的口令 pw,进而计算出了正确的 α.此时 $\mathfrak{A}$ 有两种可能途径去验证 α 的正确性:(1)通过检查 $\gcd(\alpha,n)\neq 1$ 是否成立;(2)向服务器发送消息$\{C,c_1,\mu\}$.由于无论 $\gcd(\alpha,n)\neq 1$ 是否成立,$\mathfrak{A}$ 都会收到 S 的响应消息 z,途径 1 是行不通的;同时,当 $\mathfrak{A}$ 尝试用正确的 α 去解密随机值 z 时,仍会得到一个与 pw 完全无关的随机值.因此,$\mathfrak{A}$ 无法验证 α 的正确性,进而也就无法验证 pw 的正确性.

综上可得,本游戏中去掉 $\gcd(\alpha,n)\neq 1$ 的情形不会影响 $\mathfrak{A}$ 成功的概率.这也解释了为什么文献[7]直接略过对 $\gcd(\alpha,n)\neq 1$ 情形的讨论;本文为了保持证明过程的完整性,特给出上述分析.那么,自 Game_2 之后的游戏都将是 $\gcd(\alpha,n)\neq 1$ 的情形,即 $\alpha\in\mathbb{Z}_n^*$,其取值空间大小为 $\varphi(n)$.

Game_3:本游戏中,如果 $\mathfrak{A}$ 幸运地猜测出认证元 μ 或 η,即服务器接受但 $\mathfrak{A}$ 未查询 H_1 或 H_2(通过检查 $\Lambda_{\mathfrak{A}}$ 列表来判别),终止协议的运行.由于合法认证元 μ、η 生碰撞的可能性在 Game_2 中已排除,因此除非由 $\mathfrak{A}$ 自行产生的合法认证元被拒绝,Game_2 和 Game_3 是不可区分的,可得

$$\Pr[\mathrm{Succ}_3]-\Pr[\mathrm{Succ}_2]\leqslant\frac{q_{\mathrm{send}}}{2^k} \quad (4)$$

Game_4:本游戏中,如果 $\mathfrak{A}$ 成功地计算出认证元 μ 或 η,即 $\mathfrak{A}$ 通过查询 H_1 或 H_2(通过检查 $\Lambda_{\mathfrak{A}}$ 列表来判别)得到认证元(Authenticator),终止协议的运行.在事件 $\mathrm{AskAuth}_4$ 不发生的情况下,Game_4 和 Game_3 是不可区分的,可得

$$\Pr[\mathrm{Succ}_4]-\Pr[\mathrm{Succ}_3]\leqslant\Pr[\mathrm{AskAuth}_4] \quad (5)$$

而 $\mathrm{AskAuth}_4$ 发生意味着 $\mathfrak{A}$ 成功计算出由服务器 S 产生的随机参数 a_1,下面分两种情况讨论:

—$\mathfrak{A}$ 在得知 α 的情况下(通过直接猜测或者 $\mathrm{AskPara}_n$ 事件的发生),成功由 z 计算出 a_1,

设此事件为 $\mathrm{AskAuth}_4\mathrm{WithPara}_4$;

—$\mathfrak{A}$ 在未知 α 的情况下,成功由 z 计算出 a_1,设此事件为 $\mathrm{AskAuth}_4\mathrm{WithoutPara}_4$;

我们首先来确定 $\Pr[\mathrm{AskAuth}_4\mathrm{WithPara}_4]$.在事件 $\mathrm{AskAuth}_4\mathrm{WithPara}_4$ 中,如果 $\mathfrak{A}$ 仅是被动攻击者,即仅进行 Execute 查询,我们可以利用 $\mathfrak{A}$ 构造一个求解 RSA 难题的有效(多项式时间 t)算法 π:算法 π 按照 Game_4 的规定运行(唯一不同是实例 S^i 在计算 z 时,令 $z=(\alpha c)e^m \bmod n$,其中 $c\in_R\mathbb{Z}_n^*$),以 $\mathfrak{A}$ 的输出(设为 x)为输出.由于 $\mathfrak{A}$ 不知道实例 C^i 的 RSA 公钥 e 对应的私钥 d,如果 $\mathfrak{A}$ 的输出正确,必有 $(\alpha x^e)e^m=z \bmod n$,即 $x^e=c \bmod n$.可得,

$$\Pr[\mathrm{AskAuth}_4\mathrm{WithPara}_4]\leqslant q_{\mathrm{exe}}\mathrm{Adv}^{\mathrm{rsa}}(O(t)) \quad (6)$$

现在,如果 $\mathfrak{A}$ 是主动攻击者,也可分两种情况讨论:

—$\mathfrak{A}$ 仿冒用户 C.此时由于消息$\{C,n,e,r_1\}$是 $\mathfrak{A}$ 自己产生的,$\mathfrak{A}$ 知道公钥 e 对应的私钥 d,在知道 α 的情况下可以概率 1 计算 a_1.又由于 $\mathfrak{A}$ 只能通过直接猜测或者 $\mathrm{AskPara}_n$事件的发生来得到α 的值,可得

$$\Pr[\mathrm{AskAuth}_4\mathrm{WithPara}_4]\leqslant q_{\mathrm{send}}\left(\frac{1}{\varphi(n)}+\frac{1}{|\mathfrak{D}|}\right) \quad (7)$$

—$\mathfrak{A}$ 仿冒网关 G.此时由于 $\mathfrak{A}$ 不知道实例 C^i 的 RSA 公钥 e 对应的私钥 d,这与 $\mathfrak{A}$ 仅是被动攻击者时类似,可得

$$\Pr[\mathrm{AskAuth}_4\mathrm{WithPara}_4]\leqslant q_{\mathrm{send}}\mathrm{Adv}^{\mathrm{rsa}}(O(t)) \quad (8)$$

综合(6)~(8)的结果,由于$\frac{1}{|\mathfrak{D}|}\gg\mathrm{Adv}^{\mathrm{rsa}}(O(t))$,可得

$$\Pr[\mathrm{AskAuth}_4\mathrm{WithPara}_4]\leqslant q_{\mathrm{send}}\left(\frac{1}{\varphi(n)}+\frac{1}{|\mathfrak{D}|}\right) \quad (9)$$

现在我们再来确定 $\Pr[\mathrm{AskAuth}_4\mathrm{WithoutPara}_4]$.由于 $\mathfrak{A}$ 不知道 α 的值,实例 S^j 产生的 z 对 $\mathfrak{A}$ 来说就相当于一个随机数,此时如果 $\mathfrak{A}$ 未对 $a_1\|ID_1\|r_1\|r_2\|z\|c_1$ 查询 H_1,Game_2 和 Game_3 是不可区分的,可得

$$\Pr[\mathrm{AskAuth}_4\mathrm{WithoutPara}_4]\leqslant\frac{q_{\mathrm{oh}}}{\varphi(n)} \quad (10)$$

综合(9),(10)的结果,可得

$$\Pr[\mathrm{Succ}_4]-\Pr[\mathrm{Succ}_3]\leqslant\frac{q_{\mathrm{oh}}}{\varphi(n)}+q_{\mathrm{send}}\left(\frac{1}{\varphi(n)}+\frac{1}{|\mathfrak{D}|}\right) \quad (11)$$

Game_5:本游戏中,我们用私有的 Hash 函数 H_i' 来代替 $H_i(i=1,2,3)$,即令

$$\mu=H_1'(ID_1\|r_1\|r_2\|z)$$

$$\eta=H_2'(ID_1\|r_1\|r_2\|z)$$

$$sk = H_3'(ID_1 \| r_1 \| r_2 \| z)$$

这样,μ 和 η 完全独立于服务器 S 产生的随机参数 a_1,sk 完全独立于 b_1 和 b_2.在事件 AskH_5 不发生的情况下,Game_5 和 Game_4 是不可区分的,可得

$$\Pr[\mathrm{Succ}_5] - \Pr[\mathrm{Succ}_4] \leqslant \Pr[\mathrm{AskH}_5] \quad (12)$$

而事件 AskH_5 发生意味着 $\mathfrak{A}$ 可以:(1)计算出服务器 S 产生的随机参数 a_1,这种可能性已在 Game_4 中被排除;或者(2)由 c_1 解密出 b_1,且由 c_2 解密出 b_2,这种可能性的存在预示着 $\mathfrak{A}$ 至少要实现一次 RSA 假设的破坏(任意多项式时间 t 内).因此,

$$\Pr[\mathrm{AskH}_5] \leqslant (q_{\mathrm{send}} + q_{\mathrm{exe}})\mathrm{Adv}^{\mathrm{rsa}}(O(t)) \quad (13)$$

至此,我们成功的将 $\mathfrak{A}$ 破坏"口令保护"目标的可能性限制到协议的前两次交互中.

Game_6:本游戏中,我们用私有的 Hash 函数

H' 来代替 H,即令

$$\alpha = H'(r_1 \| r_2 \| C \| G)$$

这样,α 是完全独立于用户 C 的口令 pw 的随机数.

在事件 $\mathrm{AskPara}_6$ 不发生的情况下,Game_6 和 Game_5 是不可区分的,可得

$$\Pr[\mathrm{Succ}_6] - \Pr[\mathrm{Succ}_5] \leqslant \Pr[\mathrm{AskPara}_6] \leqslant \frac{q_{\mathrm{oh}}}{\varphi(n)} \quad (14)$$

另一方面,由于 α 是完全独立于用户口令 pw 的随机数,而 $z = (\alpha a_1^{\ e})e^m \bmod n$,即 z 被服务器 S(可信实体)设置成一个与 pw 完全无关的 $\mathbb{Z}_n^*$ 中一个随机值,这与 Game_2 中 $\gcd(\alpha, n) \neq 1$ 时的情形完全相同,由 Game_2 中相关分析可知

$$\Pr[\mathrm{Succ}_6] = 0 \quad (15)$$

综合(1)~(15)的结果,可得

$$\mathrm{Adv}_{p,\mathfrak{D}}^{\mathrm{ake\text{-}uoda}}(\mathfrak{A}) = \Pr[\mathrm{Succ}_0] = \Pr[\mathrm{Succ}_0] - \Pr[\mathrm{Succ}_6] + \Pr[\mathrm{Succ}_6]$$

$$\leqslant |\Pr[\mathrm{Succ}_1] - \Pr[\mathrm{Succ}_0]| + |\Pr[\mathrm{Succ}_2] - \Pr[\mathrm{Succ}_1]| + \cdots + |\Pr[\mathrm{Succ}_6] - \Pr[\mathrm{Succ}_5]| + \Pr[\mathrm{Succ}_6]$$

$$= \frac{q_{\mathrm{send}}}{|\mathfrak{D}|} + \frac{q_{\mathrm{send}}}{2^k} + \frac{q_{\mathrm{oh}}}{\varphi(n)} + \frac{q_{\mathrm{send}}}{\varphi(n)} + \frac{(q_{\mathrm{send}} + q_{\mathrm{exe}})^2}{2\varphi(n)} + \frac{q_{\mathrm{oh}}^2}{2^{k+1}} + (q_{\mathrm{send}} + q_{\mathrm{exe}})\mathrm{Adv}^{\mathrm{rsa}}(O(t))$$

定理 1 得证.　　证毕.

5　结束语

确保密码协议的安全性是一个公开难题,对已有协议的安全性分析旨在为新协议的设计和分析提供更好的参考和借鉴.对于基于 RSA 的口令认证协议,已往研究多关注 e 次剩余攻击的危害性,本文的攻击结果突出显示了分离攻击也是针对此类协议的一种严重安全威胁,设计此类协议时必须予以充分考虑.进一步讨论了协议形式化证明中的疏漏之处,并给出一个改进方案 RSA-GPAKE+,克服了原协议安全缺陷,且未引入新的计算量和通信开销.需要指出的是,本文在显示协议的完备性方面没有建树,如何确保此类协议完备性将是我们下一步重点工作.此外,GPAKE 协议适用于移动通信环境,而移动环境中用户的隐私是重要的关注对象,因此设计具有匿名性的安全高效的基于 RSA 体制的 GPAKE 协议,也是值得进一步研究的方向.

参考文献

[1] Katz J, Ostrovsky R, Moti Y. Efficient and secure authenticated key exchange using weak passwords[J]. Journal of the ACM, 2009, 57(1): 1 – 39.

[2] Halevi S, Krawczyk H. Public-key cryptography and password protocols[J]. ACM Transactions on Information and System Security, 1999, 2(3): 230 – 268.

[3] Bellovin S M, Merritt M. Encrypted key exchange: password based protocols secure against dictionary attacks[A]. Proceedings of IEEE S&P 1992[C]. Washington DC, USA: IEEE, 1992. 72 – 84.

[4] Patel S. Number theoretic attacks on secure password schemes [A]. Proceedings of IEEE S&P 1997[C]. Washington DC, USA: IEEE, 1997. 236 – 247.

[5] Youn T Y, Park Y H, Kim C, Lim J. Weakness in a RSA-based password authenticated key exchange protocol[J]. Information Processing Letters, 2008, 108(6): 339 – 342.

[6] Zhang Mu-Xiang. New approaches to password authenticated key exchange based on RSA[A]. Proceedings of Asiacrypt 2004[C]. Berlin: Springer-Verlag, LNCS, Vol 3329, 2004. 230 – 244.

[7] Park S, Nam J, Kim S, Won D. Efficient password authenticated key exchange based on RSA[A]. Proceedings of CT-RSA 2007 [C]. Berlin: Springer-Verlag, LNCS, Vol 4377, 2007. 309 – 323.

[8] 魏福山,马传贵,程庆丰.基于 RSA 的网关口令认证密钥交换协议[J].计算机学报,2011,34(1):38 – 46.
Wei Fu-Shan, Ma Chuan-Gui, Cheng Qing-Feng. Gateway oriented password authenticated key exchange based on RSA[J]. Chinese Journal of Computers, 2011, 34(1): 38 – 46. (in Chinese)

[9] Wei Fu-Shan, Ma Chuan-Gui, Cheng Qing-Feng. Anonymous gateway-oriented password-based authenti-cated key exchange based on RSA[J]. EURASIP Journal on Wireless Communications and Networking, 2011, 2011: 162 – 173.

[10] Bellare M, Pointcheval D, Rogaway P. Authenticated key exchange secure against dictionary attacks[A]. Proceedings of Eurocrypt 2000 [C]. Berlin: Springer-Verlag, LNCS, Vol 1807, 2000. 139 – 155.

[11] Abdalla M, Chevass O, Fouque P, Pointcheval D. A simple

threshold authenticated key exchange from short secrets[A]. Proceedings of Asiacrypt 2005[C]. Berlin: Springer-Verlag, LNCS, Vol 3788, 2005. 566 – 584.

[12] Abdalla M, Fouque P, Pointcheval D. Password-based authenticated key exchange in the three-party setting[A]. Proceedings of PKC 2005[C]. Berlin: Springer-Verlag, LNCS, Vol 3386, 2005. 65 – 84.

[13] Wang Gui-Lin, Yu Jiang-Shan, Xie Qi. Security analysis of a single sign-on mechanism for distributed computer networks [J]. IEEE Transactions on Industrial Informatics, 2013, 9(1): 294 – 302.

[14] Dolev D, Yao A C. On the security of public key protocols [J]. IEEE Transactions on Information Theory, 1983, 29(12): 198 – 208.

[15] Florencio D, Herley C. A large-scale study of web password habits[A]. Proceedings of WWW 2007[C]. Passau: ACM Press, 2007. 657 – 666.

[16] Bonneau J. The science of guessing: analyzing an anonymized corpus of 70 million passwords[A]. Proceedings of IEEE S&P 2012[C]. Washington DC, USA: IEEE, 2012. 538 – 552.

作者简介

汪 定 男. 1985 年 12 月生, 湖北十堰人. 北京大学信息科学技术学院博士研究生. 研究方向为公钥密码学与信息安全.
E-mail: wangdingg@pku.edu.cn

王 平 男. 1961 年 5 月生, 北京人. 北京大学软件工程国家工程研究中心教授, 博士生导师. 研究方向为信息安全、系统软件和物联网.
E-mail: pwang@pku.edu.cn