对一种远程用户口令认证方案的改进

胡荣磊 刘建伟 张其善

(北京航空航天大学 电子信息工程学院, 北京 100191)

摘 要: 口令认证是远程身份认证中重要的方法. 分析了一种基于 hash函数强图形口令远程认证方案, 指出该方案不能抵抗校验值丢失攻击: 攻击者利用口令校验值可以假冒服务器, 欺骗合法用户发送认证信息, 生成登录信息假冒用户登录, 并通过认证获得授权, 而不用知道用户口令. 引入了智能卡的应用, 提出一种改进方案. 智能卡用来存储服务器认证信息, 实现了用户和服务器的双向认证, 攻击者不能冒充服务器和用户的任何一方, 并且攻击者从截获的信息中不能获得有用的用户认证信息. 改进方案保留了原方案抵抗重放、拒绝服务、口令猜测、伪造、口令文件丢失以及内部攻击的特点, 并能够抵抗校验值丢失和智能卡丢失攻击, 具有更好的安全性.

关 键 词: 口令; 认证; 校验值丢失攻击; 智能卡

中图分类号: TN 915. 08

文献标识码: A **文 章 编 号**: 1001-5965(2008)09-1037-04

Improvement of remote user authentication schemes using passwords

Hu Ronglei Liu Jianwei Zhang Qishan

(School of Electronics and Information Engineering, Beijing University of Aeronautics and Astronautics, Beijing 100191, China)

Abstract: Password authentication scheme is a promising and practical solution to remote user authentication. The security of an authentication scheme using strong graphical passwords basing on hash function was analyzed. The scheme can not resist to stolen-verifier attack. The adversary can pretend to be a server to cheat a legal user to send him authentication message, and then it masquerades as a legal user by creating a valid login message, passes the authentication phase and gains the authority of a legitimate user without knowing user password. An enhanced scheme was proposed in which smart card is used for storing server authentication message. The server and user can authenticate each other. The adversary can personate neither server nor user and can not get useful login message from eavesdropped communication. The enhanced scheme can withstand replay attack, denial-of-service attack, password-guessing attack, forgery attack, password-file compromise attack and inside attack as the former scheme. It can also withstand stolen-verifier attack and smart card loss attack.

Key words: password; authentication; stolen-verifier attack; smart cards

在不安全信道中识别和证实远程用户身份, 常用方法有 3种类型: ①基于用户私有知识, 比如口令、私钥等; ②基于用户私有物品, 比如智能卡、USB key、加密卡等; ③基于用户生物特征, 比如指纹、眼纹等. 其中口令认证是最简单且有效的方法, 在认证方案中占有非常重要的地位. 从 Lamport在 1981年提出一次性口令认证[1], 解决了传统口令认证中密码暴露的问题后, 许多基于口令的认证方案被提出. 从数学基础上, 这些方案可以分为基于 RSA 算法、基于 ElGamal算法和基于 hash函数的认证方案. 其中很多都采用口令和智能卡相结合的方式, 这样可以提高认证系统的安全性. 基于口令认证方案的综述参见文献[2].

2005年, Wei-Chi Ku等提出一种强图形口令远程用户认证方案[3], 该方案是基于 hash函数认证方案的一种, 可以抗重放、拒绝服务、伪造、口令

收稿日期: 2007-08-27

基金项目: 国家自然科学基金资助项目(60672102)

作者简介: 胡荣磊(1977 -), 男, 河北衡水人, 博士生, huronglei@sohu.com.

猜测、口令文件丢失和内部攻击. 根据文献[2]中提出的对口令认证方案的安全需求来分析,发现该方案不能抵抗口令校验值丢失(stolen-verifier)攻击. 如果校验值不慎丢失,攻击者可以对方案进行伪造和拒绝服务攻击,并且能自由地登录服务器,认证方案完全丧失安全性.

本文中用到的符号定义如下: U:用户 U; Q_{ID}:用户的身份号; G_{PW}:用户的图形口令; S:远程服务器; x: S为每个用户产生存储密钥的密钥; T: U向 S注册的最后时间; N:序列号,在用户注册时该值为1,每次认证后该值递加1; ‖:连接操作; ⊕:异或操作; $h(\cdot)$:单向hash函数; $h^2(\cdot)$:两次hash计算迭代; $U_1 \rightarrow U_2: U_1$通过安全信道将信息传给 U_2; $U_1 \dashrightarrow U_2: U_1$通过普通信道将数据传给 U_2.

1 Wei-Chi Ku方案

Wei-Chi Ku的方案[3]包括注册协议和登录协议两部分.

1.1 注册协议 (Registration Protocol)

Wei-Chi Ku方案注册协议如图1所示.

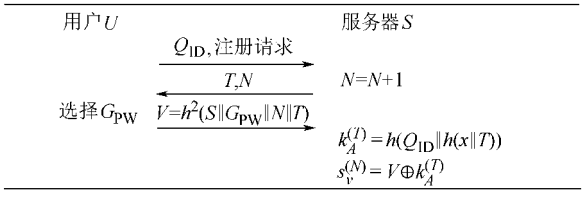


图1 Wei-Chi Ku方案注册协议

Step R1. $U \dashrightarrow S: Q_{ID}$,提出注册请求.

Step R2. S将当前的时间赋给 T,如果 U是首次注册, S将 N值赋为1,否则 $N = N + 1$.

Step R3. $S \Rightarrow U: T, N$.

Step R4. U选择其口令 G_{PW},并计算口令校验值 $V = h^2(S \| G_{PW} \| N \| T)$,这里 S为服务器身份标识.

Step R5. $U \Rightarrow S: V$.

Step R6. S计算存储密钥 $k_A^{(T)} = h(Q_{ID} \| h(x \| T))$,其中密钥 x被严格保护,用存储密钥对校验值加密 $s_v^{(N)} = V \oplus k_A^{(T)}$, S将 $s_v^{(N)}$, T, N存储在一个口令文件(password-file)中.

1.2 登录协议 (Login Protocol)

假设当 U登录 S时, $T = t$, $N = n$,协议见图2.

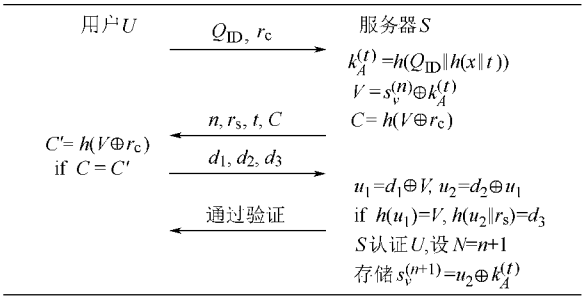


图2 Wei-Chi Ku方案登录协议

Step L1. $U \dashrightarrow S: Q_{ID}$, r_c提出登录请求,其中 r_c是一个随机数.

Step L2. S计算 $k_A^{(t)} = h(Q_{ID} \| h(x \| t))$,并计算出 $V = s_v^{(n)} \oplus k_A^{(t)}$ 和 $C = h(V \oplus r_c)$.

Step L3. $S \dashrightarrow U: n, r_s, t, C$,其中 r_s为随机数.

Step L4. U计算 V和 $C' = h(V \oplus r_c)$,同 S发送过来的值 C比较,如果 $C \neq C'$,则 U中断对话,否则 U认证 S,并计算:

$$d_1 = h^2(S \| G_{PW} \| n \| t) \oplus h(S \| G_{PW} \| n \| t)$$

$$d_2 = h^2(S \| G_{PW} \| n + 1 \| t) \oplus h(S \| G_{PW} \| n \| t)$$

$$d_3 = h(h^2(S \| G_{PW} \| n + 1 \| t) \| r_s).$$

Step L5. $U \dashrightarrow S: d_1, d_2, d_3$.

Step L6. S计算 $u_1 = d_1 \oplus V$, $u_2 = d_2 \oplus u_1$,如果 $h(u_1) = V$, $h(u_2 \| r_s) = d_3$,则 S认证 U,否则 S拒绝 U的登录要求. S计算 $s_v^{(n+1)} = u_2 \oplus k_A^{(t)} = h^2(S \| G_{PW} \| n + 1 \| t) \oplus k_A^{(t)}$,用 $s_v^{(n+1)}$代替 $s_v^{(n)}$,并设 $N = n + 1$.

2 对Wei-Chi Ku方案的攻击

在2004年,Wei-Chi Ku曾经提出过一个类似的方案[4],与文献[3]方案不同的是: ①Step L1中 $U \dashrightarrow S$没有值 r_c; ②Step L3中 $S \dashrightarrow U$没有校验值 $h(V \oplus r_c)$. Minho Kim提出该方案不能抵抗校验值丢失攻击[5],并给出了攻击方案,但是该方案并不是一个有效的方案,因为根据Minho Kim的攻击方法攻击者 A完全没有必要知道校验值是什么. 文献[3]对文献[4]方案的做了改进,但仍然不能抵抗校验值丢失攻击. 本文对Minho Kim的攻击方法进行修正,提出对文献[3]方案校验值丢失有效的攻击方法.

假设: ①攻击者 A知道hash函数 $h(\cdot)$和口令校验值 $V = h^2(S \| G_{PW} \| N \| T)$; ②$A$可以监听用户 U和服务器 S之间的信息; ③A可以冒充用户 U或者服务器 S,分别用 $A(U)$和 $A(S)$表示. 攻击分为两个阶段完成. 攻击过程如图3所示.

阶段1 攻击者 A冒充服务器 S获取认证信息

Step L1. $U \dashrightarrow A(S): Q_{ID}$, r_c提出登录请求.

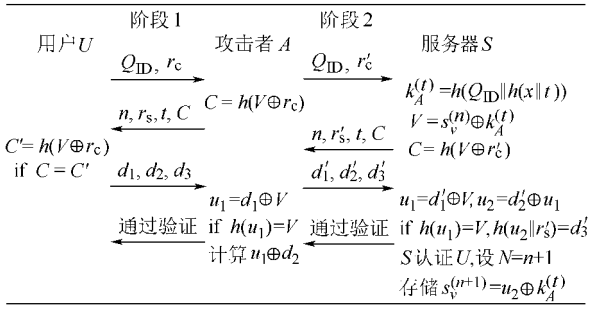


图 3　对 Wei-Chi Ku方案的校验值丢失攻击

Step L2. A计算 $C=h(V\oplus r_c)$.

Step L3. $A(S)$ - >U: n, r_s, t, C.

Step L4. U计算 V和 $C'=h(V\oplus r_c)$,同$A(S)$发送过来的值 C比较,可以看出 $C=C'$,U认证 $A(S)$,并计算 d_1, d_2, d_3.

Step L5. U - >$A(S)$: d_1, d_3, d_3.

Step L6. A计算 $u_1=d_1\oplus V$,验证 $h(u_1)=V$,如果验证成功计算 $h^2(S\|G_{PW}\|n+1\|t)=u_1\oplus d_2$,$A(S)$通知 U通过认证.

阶段 2　攻击者 A冒充用户 U登录服务器 S

Step L1.′ $A(U)$ - >S: Q_D, r_c提出登录请求.

Step L2.′ S计算 $k_A^{(t)}=h(Q_D\|h(x\|t))$,并计算出 $V=s_v^{(n)}\oplus k_A^{(t)}$ 和 $C=h(V\oplus r_c')$.

Step L3.′ S - >$A(U)$: n, r_s', t, C.

Step L4.′ A如果不想替换 $h^2(S\|G_{PW}\|n+1\|t)$,则计算 $d_1'=d_1$, $d_2'=d_2$, $d_3'=h(h^2(S\|G_{PW}\|n+1\|t)\|r_s')$,如果想替换 $h^2(S\|G_{PW}\|n+1\|t)$,则选择随机数 k,计算 $h(k)$: $d_1'=d_1$, $d_2'=h(k)\oplus h(S\|G_{PW}\|n\|t)$, $d_3'=h(h(k)\|r_s')$.

Step L5.′ $A(U)$ - >S: d_1', d_2', d_3'.

Step L6.′ S计算 $u_1=d_1'\oplus V$, $u_2=d_2'\oplus u_1$,如果 $h(u_1)=V$, $h(u_2\|r_s')=d_3'$,S认证 $A(U)$,可以看出两式都成立. S计算 $s_v^{(n+1)}=u_2\oplus k_A^{(t)}$,用 $s_v^{(n+1)}$代替 $s_v^{(n)}$,并设 $N=n+1$.

至此攻击者 A成功登录服务器 S. 如果 A没有替换 $h^2(S\|G_{PW}\|n+1\|t)$,则攻击不容易被发现;如果 A替换了 $h^2(S\|G_{PW}\|n+1\|t)$,则下次 U将不能正确登录服务器,对协议造成拒绝服务攻击,此时攻击容易被发现.

3　改进方案

借鉴文献[6]认证方案思想,对 Wei-Chi Ku的方案进行了改进. 在改进方案中加入了智能卡(SC, Smart Card)的应用,用来保存服务器的认证信息,以增强用户对服务器的认证. 改进方案对原方案的安全性进行了加强,能够抵抗校验值丢失攻击. 方案仍分为注册协议和认证协议两部分.

3.1　注册协议

协议如图 4所示.

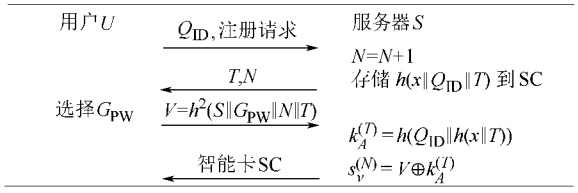


图 4　改进方案注册协议

Step R1. U - >S: Q_D,提出注册请求.

Step R2. S将当前的时间赋给 T,如果 U是首次注册,S将 N值赋为 1,否则 $N=N+1$. S计算 $h(x\|Q_D\|T)$,将其存储在 SC内,并且该值不能被读出卡外.

Step R3. S = >U: T, N.

Step R4. U选择口令 G_{PW},并计算口令校验值 $V=h^2(S\|G_{PW}\|N\|T)$.

Step R5. U = >S: V.

Step R6. S计算存储密钥 $k_A^{(T)}=h(Q_D\|h(x\|T))$,并对校验值加密 $s_v^{(N)}=V\oplus k_A^{(T)}$,$S$将 N, T, $s_v^{(N)}$存储在一个口令文件中. 将 SC发给用户 U.

3.2　登录协议

假设当 U登录 S时, $T=t$, $N=n$,协议见图 5.

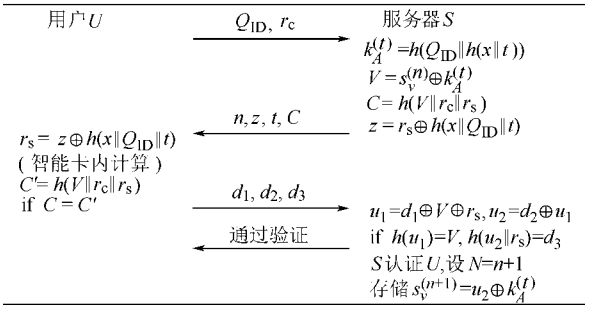


图 5　改进方案登录协议

Step L1. U - >S: Q_D, r_c,提出登录请求,其中 r_c是一个随机数.

Step L2. S计算 $k_A^{(t)}=h(Q_D\|h(x\|t))$,并计算出 $V=s_v^{(n)}\oplus k_A^{(t)}$.

Step L3. S - >U: n, t, $z=r_s\oplus h(x\|Q_D\|t)$, $C=h(V\|r_c\|r_s)$. 其中 r_s为随机数.

Step L4. U计算 V,并计算 $r_s=z\oplus h(x\|Q_D\|t)$ (SC卡内完成),计算 $C'=h(V\|r_c\|r_s)$,同 S发送过来的值 C比较,如果 $C\neq C'$,则 U中断对话,否则 U认证 S,计算:

$$d_1=h^2(S\|G_{PW}\|n\|t)\oplus h(S\|G_{PW}\|n\|t)\oplus r_s$$

$$d_2=h^2(S\|G_{PW}\|n+1\|t)\oplus h(S\|G_{PW}\|n\|t)$$

$$d_3=h(h^2(S\|G_{PW}\|n+1\|t)\|r_s)$$

Step L5. $U->S: d_1, d_2, d_3$.

Step L6. S计算 $u_1 = d_1 \oplus V \oplus r_s$, $u_2 = d_2 \oplus u_1$, 如果 $h(u_1) = V$, $h(u_2 \| r_s) = d_3$,则 S认证 U,否则 S拒绝 U的登录要求. S计算 $s_v^{(n+1)} = h^2(S \| G_{PW} \| n+1 \| t) \oplus k_A^{(t)} = u_2 \oplus k_A^{(t)}$,用 $s_v^{(n+1)}$代替 $s_v^{(n)}$, 并设 $N = n+1$.

4 改进方案安全性分析

4.1 抗校验值丢失攻击

在改进方案中,即使口令校验值 V丢失,攻击者也不能冒充用户或者服务器.

1) 假设攻击者 A想冒充服务器 S欺骗用户 U. $U->A(S): Q_D, r_c$提出登录请求. A不能获得服务器 S的密钥 x,也没有智能卡 (智能卡丢失情况见 4.2节分析),因此 A不知道 $h(x \| Q_D \| T)$, 在改进方案的 Step L3中,如果 A随机选择 z,则无法计算 r_s和 $h(V \| r_c \| r_s)$,如果 A随机选择 r_s, 则无法计算 $z = r_s \oplus h(x \| Q_D \| t)$. 改进方案对随机数 r_s进行了加密,并对其完整性进行了保护. 这样当 A伪造的 z或 r_s连同 $h(V \| r_c \| r_s)$送到用户 U后,在 Step L4中 U计算 $r_s = z \oplus h(x \| Q_D \| t)$和 $C' = h(V \| r_c \| r_s)$,此时 C肯定不等于 C',拒绝向攻击者发送 d_1, d_2, d_3. 可以看出,改进方案加强了用户对服务器的认证,实现双向认证功能.

2) 假设攻击者 A想冒充用户 U登录 S. A如果冒充用户 U登录 S,必须在 Step L4中能够成功伪造 d_1, d_2和 d_3,因为 A不知道 $h(x \| Q_D \| T)$, 无法计算 r_s. 因此 A无法伪造 d_1和 d_3. 并且攻击者从侦听到的 d_1, d_2, d_3无法计算出 $h(S \| G_{PW} \| n \| t)$和 $h^2(S \| G_{PW} \| n+1 \| t)$.

4.2 抗智能卡丢失攻击

假设智能卡丢失. 这种情况下,用户 U要向服务器 S申请智能卡挂失,重新执行注册协议,原智能卡作废. 假设在丢失智能卡仍有效的情况下, 即 $h(x \| Q_D \| T)$泄漏给攻击者 A,有两种情况:

1) A不知道口令校验值 V. A无法计算 Step L3中的 $h(V \| r_c \| r_s)$,因此无法冒充服务器 S欺骗用户 U. 同样 A无法计算 d_1, d_2和 d_3,因此不能冒充用户 U.

2) 如果 A获得了校验值 V,想冒充 S骗取用户得到认证信息 d_1, d_2和 d_3,这时因为用户 U智能卡得丢失,用户自己也无法计算出正确的 d_1和 d_3. 如果 A想冒充 U登录 S, A不能通过 G_{PW}和 V计算 $h(S \| G_{PW} \| n \| t)$,也不能欺骗用户得到该信息,在 Step L4中无法计算 d_1和 d_2,因此攻击者不能产生正确的登录信息.

综上所述,改进方案可以有效地防止校验值丢失攻击和智能卡丢失攻击.

在改进方案中仍然对口令校验值进行保密存储,这样可以防止内部窃取. 该方案可以自由更换口令,只需要在执行登录协议时,在 Step L4将 $U->S$的 d_2和 d_3中 G_{PW}更换为新的 G'_{PW},在下次登录时即可用新的口令 G'_{PW}. 关于该方案可以抵抗重放、拒绝服务、口令猜测、伪造、口令文件丢失以及内部攻击的分析,参见文献 [3].

5 结束语

本文对 Wei-Chi Ku方案的安全性进行了分析,指出其方案存在不能抵抗口令校验值丢失攻击的缺陷. 导致这个缺陷的原因是对口令校验值的过度依赖. 本文对方案做了改进,引入了服务器 S校验值 $h(x \| Q_D \| t)$,并将此值保存在 SC中. 新方案的安全性不单纯地依赖于口令校验值 V, 同时口令 G_{PW}、口令校验值和智能卡相互牵制,即使口令校验值和智能卡都丢失,攻击者仍然不能登录服务器. 改进方案对原方案的安全性进行了加强,解决了原方案中存在安全缺陷,具有更高的安全性.

参考文献 (References)

[1] Lamport L. Password authentication with insecure communication[J]. Communication of ACM, 1981, 24: 770 - 772

[2] Tsai Chwei-Shyong, Lee Cheng-Chi, Hwang Min-Shiang. Password authentication schemes-current status and key issues[J]. International Journal of Network Security, 2006, 3(2): 101 - 115

[3] Ku Wei-Chi, Tsaur Maw-Jinn. A remote user authentication scheme using strong graphical passwords[C]// 30th Annual IEEE Conference on Local Computer Networks (LCN 2005). Sydney, Australia: IEEE Computer Society, 2005: 351 - 357

[4] Ku W C. A hash-based strong-password authentication scheme without using smart cards[J]. ACM Operating System Review, 2004, 38(1): 29 - 34

[5] Kim Minho. Cryptanalysis and enhancement of authentication protocols[D]. Corvallis: Philosophy in Electrical and Computer Engineering of Oregon State University, 2006

[6] Chen T H, Lee W B, Horng G. Secure SAS-like password authentication schemes[J]. Computer Standards & Interfaces, 2004, 27: 25 - 31

(责任编辑:娄 嘉)