基于身份的动态口令验证*

戴一奇　　张　立

（清华大学计算机科学与技术系　北京　100084）

摘要　基于RSA的基本思想，本文设计了一种动态口令验证方案，解决了Lamport方案中的问题.由于不要求每个用户都拥有公钥或数字签名，该方案还具有较低的系统复杂性.由基本方案进行扩展还可得到一基于身份的动态口令验证方法.

关键词　口令验证，身份验证，单向函数，加密，公开密钥.

口令验证是保护计算机资源不被非法存取的一种基本的、常用的方法，其一般形式是对应于每个用户 A，A 拥有一秘密量（口令）P_A，对于 A，系统拥有一验证信息 V_A，存放于口令文件中，每次用户登录时，系统利用 V_A 对用户提供的 P'_A 进行验证，若通过验证，则系统就认为该用户为用户 A，并将其接受，否则拒绝该用户.

在Lamport[1]中指出了对于口令验证具有以下的攻击方式：

$A1$：通过读取系统内部的口令文件以获取用户的口令.

$A2$：通过窃听用户与系统之间的通信以获取口令.

$A3$：用户无意泄露自己的口令或关于口令的其它秘密信息.

对于一个好的口令验证方案，除 $A3$ 是无法避免的外，应该能够避免诸如 $A1$、$A2$ 的攻击，文献[1]提出了2种口令验证方案，用以对付 $A1$，$A2$.

方法1：针对 $A1$，系统利用一单向函数 f，令 $V_A=f(P_A)$，由于 f 为单向函数，故系统的验证过程是非常容易的，但任何攻击者都无法直接由 V_A 求得 P_A.但该方案不能抵抗 $A2$ 的攻击，因为攻击者可以通过 $A2$ 窃听到用户的口令，以进行非法登录.

方法2：针对 $A2$，如果用户每次登录使用的是相同的口令，那么 $A2$ 的攻击总是有效的.这就要求用户各次登录使用的是不同的口令，即动态口令.关于动态口令，文献[1]中有一个巧妙的方案，仍然是利用单向函数 f，对于用户 A，他拥有一 X_A 作为秘密种子以生成各次登录所需的口令.初始时：

$V_A^{(1)}=f^n(X_A)$，$P_A^{(1)}=f^{n-1}(X_A)$，（$P_A^{(i)}$，$V_A^{(i)}$ 分别表示第 i 次登录的口令和验证信息，其中 n 是由用户与系统商定的），

之后：$V_A^{(i)}=P_A^{(i-1)}=f^{n-i+1}(X_A)$，$P_A^{(i)}=f^{n-i}(X_A)$，$i=2,\cdots,n$.

---

* 本文研究得到清华大学自然科学基金资助.作者戴一奇，1946年生，副教授，主要研究领域为算法设计与分析，密码学.张立，1971年生，助教，主要研究领域为算法分析，密码学.

本文通讯联系人：戴一奇，北京 100084，清华大学计算机科学与技术系

本文1995－07－14收到修改稿

这样第 i 次登录就验证 $V_A^{(i)}=f(P_A^{(i)})$, $i=1,\cdots,n$.

对于任何一个攻击者来说,即使他通过窃听用户与系统之间的通信获得了 $P_A^{(1)},P_A^{(2)},\cdots,P_A^{(i)}$,即 $f^{n-1}(X_A),f^{n-2}(X_A),\cdots,f^{n-i}(X_A)$,由于 f 为单向函数,他仍然计算不出 $P_A^{(i+1)}=f^{n-i-1}(X_A)$,这样就解决了 $A2$ 的攻击.

该方案虽然实现简单,但具有以下缺点:关于 n 的选择问题,由于系统与用户之间每商定一次 X_A,用户只能进行 n 次登录,故 n 应选择为较大的数,但由于用户不宜存储过多的秘密信息,也就是说每次口令的生成都必须依靠计算来完成,若 n 太大的话,计算复杂性将会很高,这样就陷入一个矛盾的选择.而如何避免该缺点就成为一个很有意义的问题.

关于上述问题,文献[2]提出了几种解决方法.其基本思想是利用公开密钥和数字签名的概念,利用把时间常数(即每次登录和次数,这在动态口令验证方案中是必须的)的数字签名作为口令,显见,此时系统的验证是容易进行的,同时又避免了上述缺陷,但此时又要求每个用户拥有一个公钥或数字签名方案,势必会增加系统复杂性.

自从 Shamir[3]提出基于身份的加密和验证后,在口令验证中,基于身份这一概念也得到了应用,文献[4]提出了几种基于身份的静态口令验证方案,其基本思想是利用公钥和数字签名,但由于其是静态的,故仍抵抗不了 $A2$ 的攻击.另外,它的系统复杂性也较高.文献[5]利用方法 2 的一种特殊情形,设计了一种基于身份的动态口令验证方案,其作法是将方法 2 中的单向函数 f 换为一陷门单向函数,X_A 的生成由系统进行,满足 $X_A=f^{-s}(ID'_A)$ (ID'_A 是用户 A 可用于验证的身份),这样,原方案就成为一基于身份的方案了,但由于它是方法 2 的变形,故仍避免不了上面所说的问题.

另外,随着集成电路工艺的发展以及安全性要求的不断提高,带有 CPU,能进行计算的 IC 卡已有逐渐取代磁卡(类似于静态口令验证)的趋势,而动态口令验证若能做到存储及计算复杂性相对较低,则将很适合于用 IC 卡进行身份验证.

本文将给出一种动态口令验证方案,它解决了方法 2 中的问题,同时由于没有使用公钥和数字签名的方法,因而具有较低的系统复杂性,对该方案略加修改,就可以得到一基于身份的动态口令验证方法.

1　口令验证方案

该方案的基本思想是利用一仔细构造的序列 $V_1,V_2,\cdots$,由于用户掌握了关于该序列的某些陷门信息,可以很容易地计算出 $P_1,P_2,\cdots$,满足 $f(P_i)=V_i$,其中 f 为一陷门单向函数,这里 f 选用的是 RSA 系统中的加(解)密函数 $f(X)\equiv X^e \bmod n$,$n=pq$,p,q 均为大素数,但在本方案中并不要求每个用户都具有一公钥系统[6,7],即大家使用的是同一个 n,只有系统掌握 n 的分解,所以这里用户掌握的陷门信息不同于 RSA 中的陷门信息,在这里,要求用户掌握的陷门信息,能够也只能够计算出某一特定序列的 f^{-1} 即可,而无需也不能对所有的 Y,求得 $X=f^{-1}(Y)$.

基于以下考虑,关键就在于寻求满足上述条件的序列 $V_1,V_2,\cdots$,考察下列序列:

$$V_1\equiv a^{X_n X_1} \bmod n, V_2\equiv a^{X_n X_1 a^{X_n X_1}} \bmod n, \ldots, V_k\equiv a^{X_n X_1 a^{X_n X_1 \cdots a^{X_n X_1}}} \bmod n$$

只要知道了 a, X_u, X_s，就不难计算出 $\sqrt[X_s]{V_i} mod n$，因为 $\sqrt[X_s]{V_i}(mod n)=a^{X_u a^{X_u X_s} \cdots a^{X_u X_s}} mod n$，若只知道 X_s 是无法直接计算出 $mod n$ 的，但看下面的分析：

$\sqrt[X_s]{v_2} \equiv a^{X_u a^{X_u X_s} X_s} mod n$，这样只要知道了 $\sqrt[X_s]{V_1} mod n$ 及 V_1，是不难计算出 $\sqrt[X_s]{V_2} mod n$ 来的，所以上述序列不能直接用来进行口令验证，但如果加入某些因素打乱原序列的规整结构，我们就得到下述的口令验证方案.

该方案具体描述为：

初始：系统选择大素数 $p, q, n=pq$，并选择 $X_s \in Z_n^* = \{X \mid gcd(X, n)=1\}$，每个用户选择 $a \in Z_n^*$，$X_u, X_V \in Z_{\phi(n)}^*$，并计算 $Y_1 \equiv a^{X_u X_s} mod n$，$Z_1 \equiv a^{X_V X_s} mod n$，其中 (n, X_s) 公布，(Y_1, Z_1) 作为验证信息先存于系统中，用户拥有秘密信息 (a, X_u, X_V).

验证过程：第 j 次验证

用户计算：

$$P1_j \equiv a^{X_u (P1_{j-1}^{X_s} mod n)} mod n \quad P1_0=1, j \geqslant 1$$

$$P2_j \equiv a^{jX_V} mod n$$

$$P_j \equiv P1_j \cdot P2_j mod n$$

其中 P_j 作为第 j 次登录使用的口令送给系统.

系统验证

计算：

$$Y_j \equiv Y_1^{Y_{j-1}} mod n \quad Y_0=1, \ j \geqslant 1$$

$$Z_j \equiv Z^{j_1} mod n$$

$$V_j \equiv Y_j \cdot Z_j mod n$$

验证 $P_j^{X_s} \equiv V_j mod n$ 是否成立. 若成立，则通过验证，否则拒绝该用户.

下面，先证该方案的正确性，即

定理 1. 若按以上步骤进行，每次登录，合法的用户均可通过验证.

证明：先证 $Y_j = (P1_j^{X_s} mod n)$

(归纳法) $j=1$ 时，$Y_1 = a^{X_u X_s} mod n$

$$(P1_j^{X_s} mod n) = (a^{X_u X_s} mod n) = Y_1$$

设 $j=k$ 时成立，$j=k+1$ 时

$$\begin{aligned}(P1_{k+1} mod n) &= ((a^{X_u (P1_k^{X_s} mod n)})^{X_s} mod n) \\ &= a^{X_u X_s Y_k} mod n \\ &= (a^{X_u X_s})^{Y_k} mod n \\ &= Y_1^{Y_k} mod n \\ &= Y_{k+1}\end{aligned}$$

故 $j=k+1$ 时亦成立，即 $Y_j \equiv (P1_j^{X_s} mod n)$，当 $j \geqslant 1$ 时

下证 $P_j^{X_s} \equiv V_j mod n$

$$\begin{aligned}P_j^{X_s} &\equiv (P1_j \cdot P2_j)^{X_s} \equiv P1_j^{X_s} \cdot P2_j^{X_s} \equiv Y_j \cdot a^{jX_V X_s} \\ &\equiv Y_j (a^{X_V X_s})^j \equiv Y_j Z_1^j \equiv Y_j \cdot Z_j \equiv V_j (mod n)\end{aligned}$$

故若按上述步骤进行登录，每个合法的用户总可以通过验证. □

2 方案分析

下面对上述方案进行更详尽的分析.

由上述方案的实现可以看到,该方案为一动态口令验证方案,但它不同于 Lamport 的方案,Lamport 方案中的问题在于口令及验证信息的逆向生成,具体来说,是它的口令产生和验证顺序恰好相反,即最后生成的 $f^n(X_A)$要作为第 1 次登录的口令,而最先生成的 X_A 却作为最后 1 次登录的口令,这样的逆向生成势必会引起引言中提出的问题.但本文提出的方案中并不存在这样的问题,它的口令生成和使用是同步进行的,这样就解决了文章开始时提出的问题.

该方案的另一特点是其系统的复杂性较低,在本文的方案中,初始时 a, X_u, X_V, X_i 的选择都是随机进行的,并没有多大的特殊要求.这样,与采用公钥或数字签名的方案相比较,其系统复杂性较低.

下面,我们主要对该方法的安全性及复杂性进行分析.

首先,考虑该系统的安全性,仍然考虑 $A1$ 和 $A2$ 的攻击.

对于 $A1$ 的攻击,此时攻击者要直接从 $Y_j \cdot Z_j$ 计算出 $\sqrt[X_u]{Y_j \cdot Z_j} mod n$ 来,只要 RSA 体制是安全的,这种攻击就是无效的.

对于 $A2$ 的攻击,首先系统验证过程中,并没有涉及到用户的任何秘密信息,故攻击者从系统的计算中是得不到什么秘密信息的.下面再看若攻击者通过窃听用户与系统之间的通信获取了 $P_1, \cdots, P_i$,他怎样才能得到 P_{i+1}.

$P_1, \cdots, P_i$ 的形式为:

$$a^{X_u} a^{X_V} mod n, a^{X_u (a^{X_u X_i} mod n)} \cdot a^{2X_V} mod n, \ldots, a^{X_u (a^{X_u X_i} \cdots^{a^{X_u X_i} mod n})} \cdot a^{iX_V} mod n.$$

要计算 $P_{i+1} = a^{X_u (a^{X_u X_i} \cdots^{a^{X_u X_i} mod n})} \cdot a^{(i+1)X_V} mod n$,由于 X_u, X_V 的选取是随机的,所以要产生 P_{i+1},必须包含因子 $a^{(i+1)X_V}$,下面看几种获得 $a^{(i+1)X_V}$ 为因子的办法:

$$P_i \cdot P_1 = a^{X_u (a^{X_u X_i} \cdots^{a^{X_u X_i} mod n}) + X_u} \cdot a^{(i+1)X_V} \ mod n$$

$$P_{i-4} \cdot P_3 \cdot P_2 = (a^{X_u (a^{X_u X_i} \cdots^{a^{X_u X_i} mod n}) + X_u (a^{X_u X_i (a^{X_u X_i} mod n)} mod n) + X_u (a^{X_u X_i} mod n)} \cdot a^{(i+1)X_V}) mod n$$

此时,虽然获得了以 $a^{(i+1)X_V}$ 为因子的数,但与 X_u 相关的部分变得很混乱,这样 P^{i+1} 也就无法获得了.

从以上非形式化的描述中,可以认为该方案是安全的.

关于本方法的计算复杂性,我们有:

定理 2. 该方案中用户及系统验证过程的计算复杂性为 $O(|n|^3)$,其中 $|n|$ 指 n 的长度.

证明:

(1)用户计算过程

用户第 j 次产生的 $P1_j, P2_j$ 都可保留起来,用以产生 P_{j+1},此时保存的信息很少,可减少计算复杂性.

计算 $P1_j$ 需用 2 次幂乘运算,因为计算 $P1_{j-1}^{X_u} mod n$ 及 $(a^{X_u})^{P1_{j-1}^{X_u} mod n} mod n$ 各需 1 次幂乘

运算. 计算 $P2_j$ 需用 1 次模乘运算，因为 $P2_j \equiv P2_j \cdot a^{X_V} mod n$ 计算 P_j 需用 1 次模乘运算.

故用户计算过程需要 2 次模 n 的幂乘和 2 次模 n 的模乘运算，复杂性为 $O(|n|^3)$.

(2)系统验证过程

在计算及验证 Y_j 时各需一次幂乘运算，计算 Z_j 及 V_j 时各需一次模乘运算，故其需 2 次模 n 的幂乘和 2 次模 n 的模乘运算，复杂性也为 $O(|n|^3)$. □

根据(1)、(2)的分析，该方案的计算复杂性是不高的. 若考虑到 RSA 的变形，取 $X_s=2$，此时并不会降低系统的安全性，但计算复杂性可降为用户与系统各作 1 次幂乘运算和 3 次模乘运算.

由以上分析可以看到，本文提出的方法由于具有复杂性较低、需存储的秘密量少以及不需要交互等优点，是一种很适合于 IC 卡进行身份验证的方案.

3 基于身份的动态口令验证方法

所谓基于身份的动态口令验证方法是指利用用户的身份进行验证，这是防止非法登录的一种有效手段(关于基于身份的概念，详见文献[3]).

显然第 1 节中的方案不是基于身份的，但将它略作修改就可得到一基于身份的口令验证方案. 修改的方案如下：

初始时，系统选择大系数 $p,q,n=p\cdot q$，系统选定 X_s，并为每个用户随机产生 X_u, X_V，满足 $(X_u X_s, \varphi(n))=1$，$(X_V X_s, \varphi(n))=1$. 此时系统能计算出 a,b，满足：

$$a^{X_u X_s} \equiv f(ID) mod n$$

$$b^{X_V X_s} \equiv g(ID) mod n$$

其中 f,g 为可公开的单向函数，由系统选定，以便于计算 a,b，同时有助于系统的安全性；ID 为用户的身份号，是一不可伪造由用户唯一确定的公开量(如用户的身份证号码等). 将 (a, b, X_u, X_V) 给用户作为秘密信息. 对应于每个用户系统在口令文件中存放着 (ID, j, Y)，j 为登录次数，Y 为辅助验证信息，初始时 $j=1, Y=1$.

第 j 次登录时，用户计算：

$$P1_j \equiv a^{X_u (P1_{j-1}^{X_s} mod n)} mod n \qquad j \geq 1, P1_0 = 1$$

$$P2_j \equiv b^{j X_V} mod n$$

$$P_j \equiv P1_j \cdot P2_j mod n$$

将 (ID, P_j) 给系统. 系统验证：

系统根据用户提供的 ID 在文件中找到 (ID, j, Y) 并计算：

$$Y_j \equiv f(ID)^Y mod n$$

$$Z_j \equiv g(ID)^j mod n$$

$$V_j \equiv Y_j \cdot Z_j mod n$$

并验证 $P_j^{X_s} \equiv V_j mod n$ 是否成立.

若成立，则通过验证，并用 $j+1 \to j, Y_j \to Y$，若不成立，拒绝该用户.

定理 3. 若按以上步骤进行，每个合法的用户均可通过验证.

证明：略(同定理 1).

可以看出，上述方法是基于身份的口令验证方案，因为系统只需利用身份号和(n, X_r)进行验证，而 Y 也是只与身份号有关的量. 此时的安全性与复杂性讨论与第 2 节类似.

4 结　语

从以上的分析可以看到，这里给出的动态口令验证方法，解决了 Lamport 方案中的登录次数是否受限的问题，且该方法在只存储少量秘密信息的前提下，仍具有较低的计算复杂性. 特别对于系统而言，除 n 的产生(在基于身份的方案中，还包括 a，b 的产生)外，其所有操作都可以是公开进行的，并不涉及任何秘密信息. 另外，该方法具有的各种优点还使得它是一种适合于在 IC 卡内实现的身份验证的方案.

参考文献

1 Lamport L. Password authentication with insecure communication. Comm. of ACM, 1981, 24(11): 770～774.

2 何敬民. 口令验证的几种新方法. 计算机研究与发展, 1988, 25(5): 47～51.

3 Shamir A. Identity－based cryptosystem and signature schemes. Proceedings of CRYPTO'84 (LNCS 196), Springer Verlag, 1985, 47～53.

4 Harn L, Hung P, Laih C S. Password authentication using public－key cryptosystems. Computer Math. App., 1989, 18(12): 1001～1017.

5 Harn L. A public－key based dynamic password scheme. personnel communication.

6 Diffie W, Hellman M E. New direction in cryptography. IEEE Trans. Inform. Theory, 1976, IT－22: 644～654.

7 Rivest R L, Shamir A, Adleman L. A method for obtaining digital signatures and public－key cryptosystems. Comm. of ACM, 1978, 21(22): 120～126.

IDENTITY－BASED DYNAMIC PASSWORD AUTHENTICATION

Dai Yiqi　Zhang Li

(*Department of Computer Science　Tsinghua University　Beijing*　100084)

Abstract　Based on the fundamental idea of RSA, the authors designed a dynamic password authentication scheme, solving the problem in the Lamport scheme. In this scheme, there is no requirement that every user should hold a public key cryptosystem or a signature system, thus its system complexity is rather low. A slight modification on the fundamental scheme can also lead to an identity－based dynamic password authentication scheme.

Key words　Password authentication, identity authentication, one_way function, encryption, public_key.