一种多项式时间复杂度的密码协议秘密性验证方法*)

吴立军　苏开乐
(中山大学计算机科学与技术系　广州510275)

摘　要　密码协议的秘密性验证是网络安全领域的一个难题，本文在提出协议行为结构的基础上，通过对协议行为及其结构的分析，提出了一种新的密码协议的秘密性验证算法，该算法的时间复杂度是多项式时间的，从而简化了秘密性验证过程，文中最后，作为实例，给出了TMN密码协议的秘密性验证。

关键词　密码协议，秘密性，验证，TMN密码协议

An Algorithm with Polynomial Time Complexity for Verifying Secrecy Properties of Cryptographic Protocols

WU Li-Jun　SU Kai-Le
(Department of Computer Science and Technology, Zhongshan University, Guangzhou 510275)

Abstract　The verification of secrecy properties of cryptographic protocols is a hard problem in the area of computer network security. In this paper, based on introduction of framework of actions of protocols and analysis of actions of protocols and its framework, a new algorithm for verification of secrecy properties of cryptographic protocols is presented. The time complexity of the algorithm is polynomial, and so it makes the verification of secrecy properties of cryptographic protocols easier. In the last of paper, as example, the verification of secrecy properties of TMN cryptographic protocol is given.

Keywords　Cryptographic protocols, Secrecy properties, Verification, TMN cryptographic protocol

1　引言

协议的安全性验证是近年来一个重要的研究领域，主要方法有定理证明[1]和模型检测[2~6]，另外还有Petri网、Pi演算、CSP、串空间等方法[7]，而密码协议的秘密性验证是其中重要领域，在这一领域，不少专家作了深入的研究，Paulson利用归纳法，借助证明工具Isabelle成功地验证了多个协议[8]。Millen进一步扩充了魔法书事件[9]。Thayer引入strand模型和理想概念，提出了密码协议的秘密性证明方法[10]。胡成军等在Paulson方法的基础，对方法加以改进，提出了一种改进的密码协议秘密性证明方法[11]。但这些证明方法一般是复杂的，而且验证往往需要几小时甚至几周的时间。

本文通过协议行为结构的定义，及对行为和行为结构的分析，提出了一种新的密码协议秘密性的验证算法，该算法的时间复杂度是多项式时间的。

2　协议模型

2.1　参加者、可信服务器和攻击者

在协议模型中，假设有 n 个协议参与者，设为 A_1，A_2，…，A_n。这 n 个参与者是具有完全记忆的智体(完全记忆的定义见2.3节)。为了提高协议的安全性，模型中设立了一个可信服务器 S，服务器 S 的公钥为 K_S，每个智体(包括攻击者 I)都知道 K_S。智体之间通过服务器进行联系(互发信息)，即每个智体只能向服务器发送信息或服务器将信息发送给智体。

假定协议运行中存在攻击者 I，它有很强的能力：能够利用已知公私钥加密解密信息，能截获网络上任何信息，能篡改信息，发送信息，重发信息，并且能假冒任何一个协议参与者(包括服务器 S)进行通信，但是为了使假冒成功，使协议各方深信不疑，攻击者 I 在每一步假冒时所发信息尽量保持与被假冒者相同的格式，如 $A_1 \rightarrow S: A_2, \{N_{A1}\}_{K_I}$，$I$ 假冒 A_1 时为 $I(A_1) \rightarrow S: A_2, \{N_{I(A1)}\}_{K_I}$。

2.2　消息

协议运行过程实际上是参与者、可信服务器和攻击者之间互换信息的过程，以实现秘密消息的传送。模型中基本消息项包括参与者、可信服务器和攻击者的名字：A_1，…，A_n，I，S，公私钥：K_{A1}，…，K_{An}，K_I，K_S，K_{A1}^{-}，…，K_{An}^{-}，K_I^{-}，K_S^{-}，随机数：N_{A1}，…，N_{An}，N_I 等，协议中的消息由这些基本消息组合而成，包括基本信息、基本信息的连接、消息的加密和消息的解密等。消息分析主要是用私钥对消息解密。

2.3　协议的运行

为了说明协议的运行，作如下定义：

定义1(协议行为)　协议中消息的传递。它包括三要素：消息的发送者、消息的接受者和所发送的消息，形式如下：$A \rightarrow B: \{M\}$，一般用字母 a_1，a_2，…等表示，为了表述方便，用 $a(A)$ 表示行为中的发送者，$a(B)$ 表示接受者，$a(M)$ 表示此次发送的消息，其中 a 为协议行为。

定义2(完全记忆)　就是指智体在并发协议运行中能记住当前及以前收到的消息，例如：设协议运行到第 k 步，智体 A 的初始消息为 A_{m0}，在第1到第 k 步所收到的新消息为 A_{m1}，…，A_{mk}，那么智体当前记忆到的知识是 $A_{m0} \cup A_{m1} \cup \cdots \cup A_{mk}$。

在我们的协议模型中，所有智体(包括 A_1，…，A_n 和 I)都具有完全记忆，为了实现各智体的完全记忆，我们在算法中通过完全记忆器来实现。

定义3(完全记忆器)　用来帮助各智体记忆当前及以前协议行为产生的消息的一个向量，用 $(m_1, m_2, \cdots, m_n, m_I)$ 表示，其中分量 $m_i (i=1,2,\cdots,n)$ 表示参与者 A_i 目前为止所获

*)国家自然科学基金(No. 60473004)资助。吴立军　博士生，主要研究领域为智能体与网络安全；苏开乐　博士生导师，主要研究领域为智能体与网络安全。

消息的集合，记忆的消息都是源消息，即未通过分析的消息，且只供相应智体消息分析时使用，m_I 表示 I 目前为止所获消息的集合，且只供 I 消息分析使用。完全记忆器的工作过程如下：设协议行为 $a:A_i\rightarrow A_j:\{M\}$ 发生之前，完全记忆器的值为 $(m_1,m_2,\cdots,m_n,m_I)$，那么行为 a 发生后，完全记忆器的值变为 $(m_1,\cdots,m_{i-1},m_i\cup\{M\},m_{i+1},\cdots,m_{j-1},m_j\cup\{M\},m_{j+1},\cdots m_n,m_I\cup\{M\})$。同一协议的行为个数一般是有限的，设为 N，假设标准协议共有 L 步，一般协议行为个数 N 是 L 的倍数，设为 $q\times L$，因此我们可以如下描述协议行为结构（或运行结构）：

$a=\{K_1,K_2,\cdots,K_q\}$；

$K_i=\{a_{i1},a_{i2},\cdots,a_{iL}\}(i=1,2,\cdots,q)$；

$a_{ij}:A_{i1}\rightarrow A_{i2}:\{M_{ij}\}(i=1,2,\cdots,q;j=1,2,\cdots,q;A_{i1},A_{i2}\in\{A_1,\cdots,A_n,I\})$。

其中，$K_i(i=1,2,\cdots,q)$ 为并发运行中的一次运行，a_{ij} 为协议第 i 次运行 K_i 中的第 j 步发生的协议行为。

我们约定在同一协议并发运行中，同一行为只会发生一次，实际上这是符合客观实事的，例如：在同一次协议并发执行中 A 只会向 B 发送一次消息 $\{M\}$（即 $A\rightarrow B:\{M\}$），因为 A 和 B 都是具有完全记忆的智体，发送两次是多余的。

因此，我们得到下面的命题：

命题1 协议通过有限步运行，即可终止。

证明：因为每个行为只发生一次，而最多有 N 个（$=q\times L$）协议行为发生，所以协议通过有限步运行，即可终止。

由上面命题知在我们模型中，协议最多经过 N 步运行即可终止。

命题2 智体（包括攻击者）某时刻为止的完全记忆的知识与协议行为有关，而与协议行为的排列顺序无关。

证明：设记忆器的初始状态为 $(m_1^0,m_2^0,\cdots,m_n^0,m_I^0)$，到某时刻止，协议发生了 m 个行为：$a_1,a_2,\cdots,a_m$，且 a_i 为：$A_j\rightarrow A_k:\{M_i\}(i=1,2,\cdots,m),A_j,A_k\in\{A_1,\cdots,A_n,I\}$，令 $m_i=\{a_j(M)\mid a_j\in\{a_1,a_2,\cdots,a_m\},a_j(A)=A_i$ 或 $a_j(B)=A_i\}(i=1,2,\cdots,n)$，令 $m_I=\{a_j(M)\mid a_j\in\{a_1,a_2,\cdots,a_m\}\}$，那么协议发生了 m 个行为后完全记忆器的值为 $(m_1^0\cup m_1,m_2^0\cup m_2,\cdots m_n^0\cup m_n,m_I^0\cup m_I)$，因此该时刻 A_i 的完全知识为 $m_i^0\cup m_i(i=1,2,\cdots,n)$，$I$ 的完全知识为 $m_I^0\cup m_I$，但是，$m_i^0\cup m_i(i=1,2,\cdots,n)$ 和 $m_I^0\cup m_I$ 与行为 $a_1,a_2,\cdots,a_m$ 的顺序无关。因此 $A_i(i\in\{1,2,\cdots,n\})$ 及 I 的完全知识与 $a_1,a_2,\cdots,a_m$ 的顺序无关。命题得证。

不同协议的行为及其行为个数是不一样的，在分析协议行为时，一般从以下几个方面考虑（参见5.2节）：

（1）标准协议的行为分析，即在不考虑有攻击者参加时的协议行为分析。行为个数一般为标准协议长度 L。

（2）攻击者分别冒充协议的其他参与者以及可信服务器所产生的协议行为分析。行为个数一般为 $(n+1)L$，其中 n 是其他参与者 $(A_1,\cdots,A_n)$ 的个数。

（3）攻击者与服务器之间以真实身份通信的行为分析，攻击者的目的是在截获信息后，通过服务器，获取秘密性参数或者解密用的私钥或对称密钥。行为个数一般最多为 $n*L$。

3 协议的秘密性验证

3.1 协议秘密性验证的目标

协议运行要达一定目标（用 Goal 表示），从秘密性角度考虑，一般用会话参数的集合代表协议的秘密性目标，如协议双方交换的会话密钥或随机数等[8]，协议满足秘密性就是要使协议的运行或并发运行达到所有这些目标，如果有运行不能达到上述目标，那么协议就不能保证这些秘密性。假设 *Goal* $=\{key[0],key[1],\cdots,key[v-1]\}$ $key[l]$ 为会话参数（$l=0,1,\cdots,v-1$），即在 *Goal* 中有 v 个会话参数。

3.2 秘密性验证

秘密性验证的目的实际上就是要验证对于 *Goal* 中每个会话参数 $key[l](l=0,1,\cdots,v-1)$，n 个参与者 $A_1,A_2,\cdots,A_n$ 中，有若干个参与者（设为 p 个）知道该会话参数，而攻击者 I 不知道该会话参数，在这里，我们所考虑的协议的运行包括有攻击者参加的并发运行。

密码协议的秘密性验证算法的主要部分是对于会话参数 $key[l](l=0,1,\cdots,v-1)$，检查 $K_i(i=1,2,\cdots,q)$ 的每一个行为 a_{ij}，检查过程如下：

（1）检查发送者、接受者和攻击者是否知道 $key[l]$：如果行为 a_{ij} 的发送者 $a_{ij}(A)$ 的分析消息 $analyze(a_{ij}(M),a_{ij}(A))$（到目前为止）包含 $key[l]$，那么将 *Acount* 的值增加 1（*Acount* 用来记录知识 $key[l]$ 的不同参与者个数），并将 $M[hash(a_{ij}(A))]$ 赋值为1（$M[]$ 是标记，值为1说明 $a_{ij}(A)$ 是被计算过的智体），对该行为的接收者 $a_{ij}(B)$ 作同样的处理。

对攻击者 I 来说，如果 I 对该行为产生的消息 $a_{ij}(M)$ 的分析消息 $analyze(a_{ij}(M),I)$（到目前为止）包含 $key[l]$，那么将 *Icount* 设置为 1（*Icount* 标志 I 是否知道 $key[l]$）。

（2）在完全记忆器中记录行为 a_{ij} 产生的消息 $a_{ij}(M)$。

（3）判定目前为止是否存在不满足秘密性参数 $key[l]$ 的并发运行。

如果知道秘密性参数 $key[l]$ 的参与者个数 *Acount* 大于等于 $p-1$，且攻击者也知道秘密性参数 $key[l]$（即 *Icount* $=1$），那么协议不满足秘密性，且存在攻击 $T=\{K_1,\cdots,K_i\}$，否则继续检查下一个行为，直到所有会话参数的都检查完，如果还不存在攻击，那么协议就满足秘密性。

算法的描述如下：

```
function secrecycheck()
  for all l∈{1,2,…,}do
    Acount:0;
    Icount:0;
    for all i∈}{1,2,…,n}do
      M[hash(Ai)]:=0;
    end for
    for all i∈{1,2,…,q}do
      for all j∈{1,2,…,L}do
        If key[l]∈analyze(aij(M),aij(A)) andM[hash(aij(A))]=
          0 then
          Acount:=Acount+1;
          M[hash(aij(A))]:=1;
        end if
        If key[l]∈analyze(aij(M),aij(B)) and M[hash(aij(B))]=
          0 then
          Acount:=Acount+1;
          M[hash(aij(B))]:=1;
        end if
        If key[l]∈analyze(aij(M),I )then
          Icount:=1;
        end if
        //设智体 A 在完全记忆器 m 中对应的部分用 m(A)表示
        m(aij(A)):=m(aij(A))+aij(M);
        m(aij(B)):=m(aij(B))+aij(M);
        m(I):=m(I)+aij(M);
        if Acount>=p-1 and Icount=1 hen
          T:={K1,…,Ki};
          return(T);
        end if
      end for
    end for
  end for
  return("yes");
end function
```

消息分析函数 *analyze*() 有两个参数，一个是消息 *msg*，一个是智体 A（包括参与者和攻击者），其功能是：智体 A 利用完全记忆器 m 中自己的知识，对消息 *msg* 进行分析，主要是解密，以便知道秘密会话参数。

算法的描述如下：

```
Function analyze(msg,A)
  //设智体 A 在完全记忆器 m 中对应的部分用 m(A)表示
  if A∈{A1,…,An,I}then
    return(∅),//∅表示空集
  end if
  Q:=m(A)∪msg;
  Flag:=1;
  While flag=1 do
    flag:=0
    l:=the number of keys in Q;
    h:the numbr of basic messages in Q;
    //Akey[i]表示 Q 中第 i-1个密钥(i=0,1,…,l-1);
    //messages[j]表示 Q 中第 j-1个消息(j=0,1,…,h-1);
    For all i∈{0,1,…,l-1}do
      For all j∈{0,1,…,h-1}do
        If messagess[j] has form {item}k and k=opposite (Akey
          [i]) then
          //opposite(k)表示与 k 相对的密钥,即如果 k 是公钥,那
            么 opposite(k)
          //是 k 相对的私钥,如果 k 是私钥,那么 opposite(k)是 k
            相对的公//钥,如果 k 是对称密钥,那么 opposite(k)与
            k 相同。
          Q:=Q∪item;//即解密;
          Flag:=1;
        End if
      End for
    End for
  End while
End function
```

4 算法分析

假设协议中参与者个数为 n,协议行为个数为 N,标准协议的步数为 L,目标 $Goal$ 中会话参数个数为 v,协议中各次发送的消息中最多包含 R 个基本消息,消息最多加密曾数为 U。

命题3 协议满足秘密性的充要条件是算法不返回任何攻击(即返回"YES")。

证明:先证必要性:假设算法返回一个攻击,那么显然协议不满足秘密性,也就是说如果协议满足秘密性,那么算法不返回任何攻击。下面证明充分性,假设协议不满足秘密性,那么一定存在一个并发运行,使得 n 个参与者 $A_1,A_2,\cdots,A_n$ 中,至少有 $p-1$个参与者知道会话参数,而攻击者 I 也知道会话参数,也就是说该并发运行是一个攻击。而该并发运行的每次运行一定属于$\{K_1,K_2,\cdots,K_q\}$,所以该并发运行必定是$\{K_1,K_2,,K_q\}$中某几个元素的一个排列,由命题2知,攻击的存在性与排列的顺序无关,所以一定存在 j,使得$\{K_1,K_2,\cdots,K_j\}$包含该并发运行,且也是一个攻击运行,所以由算法的过程知道算法对 $K_1,K_2,\cdots,K_j$ 进行检查一定能得出一个攻击。也就是说,如果算法不返回任何攻击,那么协议一定满足秘密性。命题得证。

命题4 密码协议秘密性的验证算法的时间复杂度为 $O(N^2*v^2*R*U*n)$,是多项式时间的。

证明:先计算函数 *analyze*()的时间复杂度,因为最内层循环次数 h 最多为 $N*R$,而第二次循环次数 l 最多为 v,所以 *analyze*()的时间复杂度为 $O(N*R*v*U)$,所以 *secrecy-check*()的最内层循环(即第三层循环)的时间复杂度为 $O(N*R*v*U*L)$,因而第二层循环的时间复杂度为 $O(N*R*v*U*L*q)$,第一层循环的时间复杂度为 $O(N*R*v*U*L*q*v*n)$,而 $N=L*q$,所以算法的时间复杂度为 $O(N^2*v^2*R*U*n)$,是多项时间的。

5 实例

我们以 TMN 密码协议作为实例来进行分析[12]。

5.1 TMN 密码协议

TMN 密码协议是应用于移动通信系统的密码分配协议,原始协议如下:

$A\rightarrow S:B,\{N_a\}Ks$
$S\rightarrow B:A$
$B\rightarrow S:A,\{N_b\}Ks$
$S\rightarrow A:B,\{N_b\}Na$

其中 A 为初始者,B 为响应者,S 为可信第三方,K_s 为 S 的公钥,N_a、N_b 是 A 和 B 发布的具有新鲜性的随机数,N_a 也是 A 的公钥,N_b 是 B 的公钥也作为 A,B 间秘密通信的会话密钥,协议运行的目的是在 A 和 B 之间建立一个会话密钥 N_b(即 K_{ab}),这个密钥在他们今后秘密通信时使用。

5.2 协议行为分析

我们假设有一个攻击者 I,它具有很强的能力,具体如2.1节,协议可以并发多次运行,协议的行为如下:

K_1:
$a_{11}: A\rightarrow S:B,\{Na\}_{K_s}$
$a_{12}: S\rightarrow B:A$
$a_{13}: B\rightarrow S:A,\{N_b\}_{K_s}$
$a_{14}: S\rightarrow A:B,\{N_b\}_{N_a}$

K_2:
$a_{21}: I\rightarrow S:I,\{N_1\}_{K_s}$
$a_{22}: S\rightarrow I:I$
$a_{23}: I\rightarrow S:I,\{N_2\}_{K_s}$
$a_{24}: S\rightarrow I:I,\{N_2\}_{N_1}$
(以上 N_1、N_2可以为 N_a、N_b 和 N_I)

攻击者 I 冒充 A

K_3:
$a_{31}: I(A)\rightarrow S:B,\{N_I\}_{K_s}$
$a_{32}: S\rightarrow B:A$
$a_{33}: B\rightarrow S:A,\{N_b\}_{K_s}$
$a_{34}: S\rightarrow I(A):B,\{N_b\}_{NI}$

攻击者 I 冒充 B

K_4:
$a_{41}: A\rightarrow S:B,\{Na\}_{K_s}$
$a_{42}: S\rightarrow I(B):A$
$a_{43}: I(B)\rightarrow S:A,\{N_I\}_{K_s}$
$a_{44}: S\rightarrow A:B,\{N_I\}_{N_a}$

攻击者 I 冒充服务器 S

K_5:
$a_{51}: A\rightarrow I(S):B,\{Na\}_{K_s}$
$a_{52}: I(S)\rightarrow B:A$
$a_{53}: B\rightarrow I(S):A,\{N_b\}_{K_s}$
$a_{54}: I(S)\rightarrow A:B,\{N_b\}_{N_a}$

5.3 协议的秘密性验证

协议中会话参数只有 N_a, N_b, N_I。故协议目标 $Goal=\{N_a, N_b, N_I\}$。利用第4节的方法我们容易得到下列攻击:

K_1:
$a_{11}: A\rightarrow S:B,\{N_a\}_{K_s}$
$a_{12}: S\rightarrow B:A$
$a_{13}: B\rightarrow S:A,\{N_b\}_{K_s}$
$a_{14}: S\rightarrow A:B,\{N_b\}_{N_a}$

K_2:
$a_{21}: I\rightarrow S:I,\{N_I\}_{K_s}$
$a_{22}: S\rightarrow I:I$
$a_{23}: I\rightarrow S:I,\{N_b\}_{K_s}$
(攻击者 I 在 a_{13}截取消息$\{N_b\}_{K_s}$后重发)
$a_{24}: S\rightarrow I:I,\{N_b\}_{NI}$

因此协议是不能保证会话参数的密秘性的,事实上在5.2节的基本行为分析中也能看出来。

结论 本文提出了一种多项式时间复杂度的密码协议秘密性验证方法,并给出了一个实例,进一步说明了该方法的应用价值,文中给出了算法的描述,我们也正在进一步完成算法的代码编写,以便使用这些方法验证更多的密码协议的秘密性。

参 考 文 献

1 Burrows M, Abadi M, Needham R. A Logic of Authentication. DIGITAL, Systems Research Center, N. 39 Feb. 1989. http://www.research.digital.com/SRC/Publications/

数据流管理系统中适应性查询机制的研究*)

宋宝燕[1] 武珊珊[2] 于 戈[2]

(辽宁大学信息科学与技术学院 沈阳110036)[1] (东北大学信息科学与工程学院 沈阳110004)[2]

摘 要 介绍了数据流技术的发展现状,然后讨论了适应性查询在数据管理中的发展演变,特别是在数据流管理中的特殊性。最后,在此基础上,提出了一个支持适应性查询的数据流管理系统RealStream,并详细介绍了其适应性查询处理机制。

关键词 数据流,适应性查询,查询处理

The Study on Adaptive Query Processing Mechanism in DSMS

SONG Bao-Yan[1] WU Shan-Shan[2] YU Ge[2]

(Department of Information Science and Technology, Liaoning University, Shenyang 110036)[1]

(Department of Information Science and Engineering, Northeastern University, Shenyang 110004)[2]

Abstract This paper presents the state of art of data stream technology, and discusses the evolvement of adaptive query processing mechanism in data management, with an emphasis on the charateristics of data stream management. Based on this, RealStream, a prototype system of dada stream management system for adative query, is proposed and implemented focusing on adaptive query processing mechanism.

Keywords Data stream, Adaptive query, Query processing

1 引言

随着信息处理在通信、工业生产、经济信息处理等领域的广泛应用,数据已不仅仅拘泥于文件、数据库等传统的静态形式,一种连续、无界、不定速度的流式数据(即数据流)已经出现在越来越多的应用领域,如:网络监控、网络流量管理、入侵检测、传感器的数据处理、生产线管理、股市信息分析等等。如何对数据流实施有效管理是这些应用领域中的核心问题。对数据流管理技术的研究目前已经成为数据库领域又一新的研究热点[1~6]。在传统的数据库管理系统中,所有的数据都是以持久的数据集合形式出现的,而近年来在越来越多的应用中,数据是以流的形式在线到达的。在这些应用中,不仅原始数据是以流的形式到达,对这些数据流进行各种处理(主要是查询操作)之后得到的结果也同样是以流的形式输出的。针对数据流的特征,传统的数据处理技术(如数据库技术)已不能满足其处理要求,所以研究合适的数据流处理技术是非常必要的。

数据流的查询处理有两个特性:实时性和适应性。简单地说,实时性是指系统需在规定的时间内对查询处理任务有所响应;适应性是指系统能随着数据特性和系统属性的变化动态地调整查询执行的行为,以在满足实时性要求的前提下完成查询任务。适应性查询一直是传统数据管理中的一个热点和难点,随着数据库系统的演变、发展以及查询需求的不断提高,适应性查询的内涵也不断丰富,特别是在数据流这一新型的数据形式下,适应性查询具有更重要的涵义和作用。

本文首先介绍了数据流技术研究与开发的发展状况,然后讨论了适应性查询在数据管理中的发展演变,特别是在数据流管理中的特殊性。最后,在此基础上,给出了一个支持适应性查询的数据流管理系统RealStream,并详细介绍了其适应性查询处理机制。

2 相关工作

近年来,有关流式数据的研究已经成为当前数据库领域新的研究热点。这种流式数据一般被称作为数据流(data stream),其定义如下[2]:

*)本课题得到辽宁省自然科学基金(编号:20022027),国家"863"高技术计划GIMS主题(编号:2002AA1Z2308,2002AA118030)资助。宋宝燕 博士,教授,研究方向为数据流技术、数据库技术;武珊珊 博士研究生,研究方向为数据流技术;于 戈 教授,博士生导师,研究方向为数据库技术、数据仓库技术以及网格计算技术。

2 Clarke M, Orna Jr, Peled A. Model Checking. The Mit Press Cambridge Massachusetts London, England
3 McMillan M L. Symbolic Model Checking: An Approach to the state Explosion Problem. Kluwer Academic, 1993
4 van der Hoek W, Wooldridge M. Model Checking Knowledge and Time. In: Model Checking Software—— Proceedings of SPIN, 2002
5 Su Kaile. Model Checking Temporal Logics of Knowledge in Distributed Systems. In: The Nineteenth National Conf. on Artificial Intelligence, 2004
6 吴立军,苏开乐.多智体系统时态认知规范的模型检测算法.软件学报,2004,15(7):1012~1020
7 吴立军,苏开乐.安全协议认证的形式化方法研究.计算机工程与应用,2004,40(17):152~155
8 Paulson I. The inductive approach to verifying cryptographic protocol. Journal of Computer Security, 1998, 6(1): 85~128
9 Millen J, Rue β H. Protocol-independent secrecy. In: Proc. of the 2000 IEEE Symposium on Security and Privacy, Berkeley California, USA, 2000. 100~119
10 Thayer J, Herzog J, Guttman J. Honest ideals on strand space. In: Proc. of the 11th IEEE Computer Security Foundations Workshop, Rockport massachusets, USA, 1998. 66~78
11 胡成军,郑援,沈昌祥.密码协议的秘密性证明.计算机学报,2003,26(3): 367~372
12 Tatebayashi M, Matsuzaki, Newman D B. Key distribution protocol for digital mobile communication systems. In: Proc of Crypto'89, JNCS vol 435. Berlin: Springer Verlag, 1990. 324~333