计算机科学 1999Vol. 26№. 1

一个描述密码学安全协议的演算:Spi 演算

The Spi Calculus: A Cryptographic Protocol Calculus

曾小平 孙永强

(上海交通大学计算机科学与工程系 上海 200030)

Abstract Spi calculus is an extension of the pi calculus for the description and analysis of cryptographic protocols. By the increased construct, Spi calculus not only represents the cryptographic protocols explicitly but also authenticate them in a precise semantics.

Keywords Pi calculus, Spi calculus, Shared-key, Cryptographic protocol

Spi 演算是 M. Abadi 和 A. D. Gordon 在 pi 演算[1]基础上加以扩充,从而实现用 pi 演算来描述和分析基于密码学的安全协议的模型[2,3]。pi 演算作为并发计算的基础,其最重要之处是引入了通道(channel)的概念。通过产生并将这种有名字的通道传送给其它进程可以在这些进程之间建立起新的通信。通道具有确定的作用域(scope),作用域之外的进程不能对该通道进行存取,这在一定程度上提供了通道通信的安全性。但它和传统意义上的安全性所不同的是:pi 演算的安全性是通过限制通道作用域来实现对通道中传送的公开数据的保密;而在传统的安全领域中的安全性则主要是通过加密后的数据在公开信道上传输来实现的。pi 演算并没有为数据加密和解密的描述提供相应的结构,但它对并发计算的描述能力和简洁性是其它描述方法所无法比拟的。因此需要在 pi 演算中增加支持密码学的原语以支持用 pi 演算描述基于密码学的安全协议。

Spi 演算和其它安全协议描述方法(见[4],[5])所不同的是:Spi 演算不仅有着简洁而且形式化的语义,而且作为并发语言 pi 演算的扩充,其实现也是非常直观的,而且通过"测试等价"(testing equivalence)可以形式地验证安全协议的安全性。

1. Pi 演算

Pi 演算是 Robin Milner 在 CCS[6]等进程演算的基础上作为并发计算的基础提出的,同时它也是一个很小但具有很强表达能力的并发语言。一个 pi 演算程序就是一个由通过有名字的通道上的消息传递实现同步的相互独立的并行进程所组成的系统。每个进程所知道的通道决定了该进程的通信能力。通过限制通道的作用域可以限制不在作用域内的进程对该通道的存取能力,因此限制通道的作用域可以实现对该通道的保密,从而实现某种程度上的安全性。

1.1 Pi 演算

Pi 演算中最基本的实体(entity)是名(name),名可以用来命名通道。假定有名 m,n,p,q,…所组成的无限集合以及变量 x,y,z,…所组成的无限集。pi 演算中的另一类实体——项(term)是如下定义的集合:

L,M,N::=	项
u	名(如 m,n,p,q,…)
(M,N)	对(pair)
0	零
suc(M)	后继算子
x	变量(如 w,x,y,z,…)

进程(process)则是由项通过如下规则定义的 pi 演算实体:

P,Q,R::=	进程
$\overline{M}\langle N\rangle.P$	输出
$M(x).P$	输入
$P\mid Q$	复合(composition)
$(\nu n)P$	限制(restriction)
$!\ P$	复制(replication)
$[M\ is\ N]P$	匹配(match)
0	空进程
$let(x,y)=Mn\ P$	拆对(pair splitting)
$case\ M\ of\ 0:P\ suc(x):Q$	分支(case)

在(νn)P中，P中的名n称为受限的；在M(x)P中，P中的变量x称为受限的；在case M of 0:P suc(x):Q中变量x在第二个分支Q中是受限的。我们称不受限制的变量(名)的出现是该变量(名)的自由出现。记进程P中所有自由出现的名所组成的集合为fn(P)，所有自由出现的变量组成的集合为fv(P)，如果一个进程中没有任何自由出现的变量(即所有变量都是受限的或fv(P)=φ)则称该进程是闭进程(close process)。另外，记P[M/x]为将P中所有x的自由出现用项M替换后所得到的进程，从直观上看，pi演算进程构造的含义如下：

● pi演算中最基本的计算和同步机制是交互(interaction)，在其中输出进程通过名为m的通道向输入进程传送项M。①输出进程$\overline{m}\langle N\rangle.P$准备从通道m上输出项N，然后执行子进程P；②输入进程$m(x).P$准备从通道m上输入项，当输入项N后，执行进程P[N/x]。

● 复合P|Q就是指进程P和进程Q并行运行，每个进程都可以通过通道和其它进程或/和外界独立地进行交互。

● 限制进程(νn)P产生一个新的私有名n(n可能在P中出现)，然后执行P。

● 复制进程！P是指并行地运行无限多个进程P的副本。

● 匹配进程[M is N]P如果项M和项N相同则执行P；否则进程不做任何事情。

● 空进程0不做任何事情。

● 拆对进程let(x,y)=M in P和对的构造相对应。如果项M是对(N,L)，则执行进程P[N/x][L/y]；否则阻塞。

● 分支进程case M of 0:P suc(x):Q当项M为0时执行子进程P；当项M为suc(N)时执行子进程Q[N/x]；否则阻塞。

1.2 利用pi演算的通道限制来描述安全协议

和实际的安全协议不同，pi演算中安全协议的描述只能通过对通道的限制实现，其中通道起着和密码学中的密钥相同的作用。

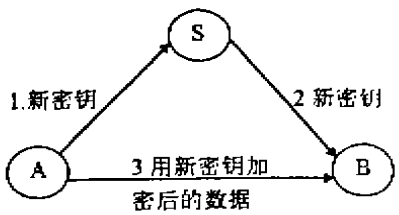

图1 Wide Mouthed Frog协议的结构

图1示出了Wide Mouthed Frog协议[2]的结构，它实现的是A通过S将要传送给B的加密数据的解密密钥传送给B，B将用接收到的密钥对从A接收的数据进行解密。在pi演算描述时，我们用通道名来代替密钥，该协议的非形式描述如下：

消息1 $A\rightarrow S: C_{AS}\ on\ C_{AS}$(A通过通道$C_{AS}$向S发送新通道名$C_{AB}$)

消息2 $S\rightarrow B: C_{AB}\ on\ C_{SB}$(S通过通道$C_{SB}$向B发送新通道名$C_{AB}$)

消息3 $A\rightarrow B: M\ on\ C_{AB}$(A通过新通道$C_{AB}$向B发送消息M)

首先，进程A通过通道C_{AS}将新通道的名字C_{AB}发送给进程S(消息1)；S接收到C_{AB}后将它通过C_{SB}转发给进程B(消息2)；在B接收到新通道名C_{AB}之后，A和B之间就建立了一个新的通道C_{AB}，它们通过该通道可以传送消息M(消息3)。利用pi演算来描述上述安全协议如下：

$$A(M)\equiv(\nu\ C_{AB})\overline{C_{AS}}\langle C_{AB}\rangle.\overline{C_{AB}}\langle M\rangle$$
$$S\equiv C_{AS}(x).\overline{C_{SB}}\langle x\rangle$$
$$B\equiv C_{SB}(x).x(y).F(y)$$
$$Inst(M)\equiv(\nu C_{AS})(\nu C_{SB})(A(M)|S|B)$$

其中A(M)、S和B分别描述进程A、B和S，Inst(M)则描述了整个系统。

从上例中可以看到，协议的安全性是通过对通道C_{AS}、C_{AB}和C_{SB}的限制实现的。由于通道受限，系统之外的进程无法对这些通道进行存取，因而也无法获得有关通道C_{AB}的信息，从而实现协议的安全性，但在实际的安全系统中，信道保护的代价是非常高的。相反，对信道上的数据进行保护不仅代价小，而且实现简单。因此，实际安全协议系统通常是通过先将要传送的数据加密，然后在公开信道上传送这些加密后的数据(密文)，接收者在接收到密文之后要对其解密后才能使用这些数据。这样加密后的数据即使被非法获取，但由于获取方不知道解密所需的密钥，也就不能使用这些密文。因此，要描述安全协议就必须在pi演算中增加描述密码学的机制，提供对基于密码学的安全协议描述的支持。

2. Spi演算及其语义

目前常用的加密解密系统有共享密钥系统(Shared-Key)、公共密钥系统(Public-Key)和数字签名(Digital Signature)等。本文对密码学机制的描述主要是针对基于共享密钥系统的密码学安全协议的。共享密钥系统中数据加密和解密所使用的密钥相同，因此实现比较简单但要求的保护对系统的安

全性有着非常重要的影响，其它密码学机制的描述可以通过在pi演算中加入相应的原语来实现。

2.1 基于共享密钥密码学的Spi演算

这种Spi演算是通过在pi演算中增加利用共享密钥对数据进行加密和解密的两个原语来实现的。相应的项和进程的定义如下：

L、M、N::=项

…　　如1.1节

$\{M\}_N$　　基于共享密钥的加密

对应于数据的加密，在进程结构中增加数据解密原语：

P、Q、R::=　　进程

…　　如1.2节

$case\ L\ of\ \{x\}_N\ in P$　　基于共享密钥的解密

其中变量x在P中是受限的，直观地：①项$\{M\}_N$代表在共享密钥系统(如DES)下利用密钥N对项M加密后得到的密文。②进程$case\ L\ of\ \{x\}_N\ in\ P$试图利用密钥N对项L进行解密。如果项L是形如$\{M\}_N$的密文，则执行子进程P[M/x]；否则进程阻塞。

有了Spi演算，我们可以更为自然地描述1.2节图1中所示的Wide Mouthed Frog协议。该协议非形式的基于共享密钥的描述如下：

消息1　A→S：$\{K_{AB}\}_{K_{AS}} on\ C_{AS}$

消息2　S→B：$\{K_{AB}\}_{K_{SB}} on\ C_{SB}$

消息3　A→B：$\{M\}_{K_{AB}} on\ C_{AB}$

首先，进程A通过通道C_{AS}将新密钥K_{AB}用A和S的共享密钥K_{AS}加密之后传送给进程S(消息1)；S在收到密文后先将其解密得到K_{AB}，然后利用S和B的共享密钥K_{SB}对K_{AB}进行加密后将密文传送给进程B(消息2)；B利用K_{SB}对密文进行解密后就到了新密钥K_{AB}，之后B就利用该密钥对从进程A来的数据M进行解密(消息3)。在Spi演算中，上述协议描述如下：

$A(M)\equiv(\nu\ K_{AB})(\overline{C_{AS}}\langle\{K_{AB}\}_{K_{AS}}\rangle.\overline{C_{AB}}\langle\{M\}_{K_{AB}}\rangle)$

$S\equiv C_{AS}(x).case\ x\ of\ \{y\}_{K_{AS}}\ in\ \overline{C_{SB}}\langle\{y\}_{K_{SB}}\rangle$

$B\equiv C_{SB}(x).case\ x\ of\ \{y\}_{K_{SB}}\ in\ C_{AB}(z).case\ z\ of\ \{W\}_y\ in\ F(w)$

$Inst(M)\equiv(\nu\ K_{AS})(\nu K_{SB})(A(M)\mid S\mid B)$

和用pi演算描述所不同的是：Spi演算中以对密钥的保护(限制)代替了对通道的保护；用加密后的密文的传送代替了非加密项的传送，从而使得对协议的描述更加直观自然。

2.2 Spi演算的形式语义

从直观上看，数据保密性是指系统之外的第三方不能观察到系统中传送的数据M[2]，因此为了定义Spi演算的语义需要引入某种关系来描述这种保密性，通过Spi进程之间的观察等价(即从外部环境观察传送数据M的进程P(M)和传送数据M'的进程P(M')之间的等价性)可以验证安全协议的保密性(secrecy)。

定义1　闭进程上的推导关系(reduction relation)是由如下规则定义的关系>：

$!\ P>P\mid!\ P$

$[M\ is\ M]P>P$

$let(x,y)=(M,N)in\ P>P[M/x][N/y]$

$case\ 0\ of\ 0:P\ suc(x):Q>P$

$case\ suc(M)of\ 0:P\ suc(x):Q>Q[M/x]$

$case\{M\}_N of\ \{x\}_N\ in\ P>P[M/x]$　□

定义2　结构等价(structural equivalence)≡是满足下列等式和规则的闭进程上的最小关系：

$P\mid 0\equiv P$

$P\mid Q\equiv Q\mid P$

$P\mid(Q\mid R)\equiv(P\mid Q)\mid R$

$(\nu m)(\nu n)P\equiv(\nu n)(\nu m)P$

$(\nu n)0\equiv 0$

$(\nu n)(P\mid Q)\equiv P\mid(\nu n)Q\ if\ n\notin fn(P)$

$\dfrac{P>Q}{P\equiv Q}\quad\dfrac{P\equiv Q}{P\equiv P}\quad\dfrac{P\equiv Q}{Q\equiv P}$

$\dfrac{P\equiv Q\ Q\equiv R}{P\equiv R}\quad\dfrac{P\equiv P'}{P\mid Q\equiv P'\mid Q}\quad\dfrac{P\equiv P'}{(\nu m)P\equiv(\nu m)P'}$　□

定义3　反应关系(reaction relation)→是满足$\overline{m}\langle N\rangle.P\mid m(x).Q\to P\mid Q[N/x]$和下列规则的闭进程上的最小关系[7]：

$\dfrac{P\equiv P'\ P'\to Q'\ Q'\equiv Q}{P\to Q}\qquad\dfrac{P\to P'}{P\mid Q\to P'\mid Q}$

$\dfrac{P\to P'}{(\nu n)P\to(\nu n)P'}$　□

定义4　一个barb β是指一个输入或输出通道(或者是输入通道名m或者是输出通道名$\overline{m}$)。对闭进程P，通过如下两个公理和三条规则定义谓词P展示barb β(写作$P\downarrow\beta$)：

公理：$m(x).P\downarrow m\qquad\overline{m}\langle M\rangle.P\downarrow\overline{m}$

规则：$\dfrac{P\downarrow\beta}{P\mid Q\downarrow\beta}\quad\dfrac{P\downarrow\beta\ \beta\notin\{m,\overline{m}\}}{(\nu m)P\downarrow\beta}$

$\dfrac{P\equiv Q\qquad Q\downarrow\beta}{P\downarrow\beta}$　□

从直观上看，$P\downarrow\beta$是指闭进程P可以马上在barb β(通道)上输入或输出数据(即P进程的第一个动作是通道操作)。通过谓词↓我们再定义谓词⇓：

定义5　P是闭进程。如果P在进行一系列反应后展示barb β，则称$P\Downarrow\beta$：

$$\frac{P\downarrow\beta}{P\downarrow\beta}\qquad\frac{P\rightarrow Q\quad Q\Downarrow\beta}{P\Downarrow\beta}$$ □

定义 6 一个测试(test)由任意闭进程 R 和 barb β组成，记为(R,β)。当且仅当(P|R)⇓β时称闭进程 P 通过测试(R,β)。对任意两个闭进程 P 和 Q，有测试等价关系≅：

$P\cong Q\equiv$对任意测试(R,β)，$(P\ R)\downarrow\beta$当且仅当$(Q|R)\Downarrow\beta$ □

由测试等价关系我们可以定义保密性(secrecy)：

定义 7 安全协议系统 Sys(M)(M 是系统中传送的数据)中。对任意 M、M′，如果 F(M)≅F(M′)则有 Sys(M)≅Sys(M′)。称安全协议系统 Sys(M)具有保密性。 □

2.1 节用 Spi 演算描述的 Wide Mouthed Frog 协议系统 Inst(M)中，对任意 M 有且仅有 barb 集合 $B=\{C_{AB},\overline{C_{AB}},C_{AS},\overline{C_{AS}},C_{SB},\overline{C_{SB}}\}$，因此对任意测试(R,β)(β∈B∪$B_R$，$B_P$ 是 R 所展示的 barb 的集合)，通过对 Inst(M)的每个反应及 barb 的展示的推导可以证明：对任意 M、M′，如果 F(M)≅F(M′)则有 Inst(M)≅Inst(M′)。也就是说，Wide Mouthed Frog 协议系统具有保密性。

结论 通过对 pi 演算和 Spi 演算的介绍可以看出，标准的 pi 演算不能很好地描述现有的基于密码学的安全协议。通过在 pi 演算中加入支持基于共享密钥的密码系统描述的原语，Spi 演算不仅实现了对安全系统的描述，而且通过测试等价的定义提供了验证安全系统安全性的形式而简洁的方法。并且通过对 Wide Mouthed Frog 安全协议系统的描述和验证，充分说明了 Spi 演算的描述能力和简洁的语义，虽然这个例子很小，但对更为复杂的安全系统的 Spi 演算描述也是很直观的。

本文所述的 Spi 演算是针对基于共享密钥的密码学系统的。目前的密码学系统还有哈希函数法、基于公共密钥的密码系统、数字签名等[8]。我们对 Spi 演算进一步研究将主要集中在两个方面：一是对 Spi 进行进一步扩展使能描述更多的数据加密标准，实现对复杂安全协议系统的描述；二是对 Spi 的语义进行更深入的研究，实现更为直观和精确的语义描述，并且提供密码学协议系统安全性验证的更简单、更形式化的证明方法

参 考 文 献

1 Milner R, Parrow J, Walker D. A Calculus of Mobile Processes, parts I and II. Information and Computation, Sept 1992: 1～40 and 41～77

2 Abadi M, Gordon A D. A Calculus for Cryptographic Protocols: The Spi Calculus. In the Proc. of the Fourth ACM Conference on Computer and Communications Security, April 1997

3 Abadi M, Gordon A D. Reasoning about Cryptographic Protocols in the Spi Calculus. In the Proc. of CONCUR'97, Aug. 1997

4 Liebl A. Authentication in Distributed Systems: A Bibliography. ACM Operating Systems Reviews, 1993, 27(4): 31～41

5 Burrows M, Abadi M, Needham R M. A Logic of Authentication. In: the Proc. of the Royal Society of London A, 426, 1989: 233～271

6 Milner R. Communication and Concurrency. Prentice-Hall International, 1989

7 Milner R. Functions as Processes. Mathematical Structures in Computer Science, 1992. 119～141

8 Schneider B. Applied Cryptography Protocols, Algorithm and Source Code in C. John Wiley & Sons, Inc., 1994

(上接第 45 页)

networks by probabilistic logic sampling, Uncertainty in Artificial Intelligence. North Holland, Amsterdam, 1988

[4] Lauritzen S L, et al. Local computation with probabilities in graphical structures and their applications to expert system. J. Royal Statistical Society B, 1988, 50(2)

[5] Pearl J. Fusion, propagation, and structuring in belief networks. Artificial Intelligence, 1986, 29

[6] Ribeiro Berthier A N. A Belief Network Model for IR. Proc. of 19th Ann. Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval, Zuich, Switzerland, August, 1996

[7] Sarkar Sumit. Constructing Efficient Belief Network Structures With Expert Provided Information, IEEE Trans. on Knowledge and Data Engineering, 1996, 8(1)