计算可靠的密码协议形式化分析综述

雷新锋[1),2)] 宋书民[2)] 刘伟兵[2)] 薛 锐[1)]

[1)](中国科学院信息工程研究所信息安全国家重点实验室 北京 100093)

[2)](北京信息技术研究所 北京 100097)

摘 要 密码协议的描述和分析有两类截然不同的方法:一类以形式化方法为主要手段,另一类以计算复杂性理论为基础. Abadi 和 Rogaway 首次试图将这两类不同的方法关联起来,证明一个协议在形式化模型下具有某种安全属性,那么在计算模型下也保持相应的安全属性. 在这一工作的带动下,形式化方法的计算可靠性研究越来越受到关注,成为密码协议分析研究的一个重要内容. 围绕这一热点问题,人们做了大量的工作. 该文首先对两类分析方法做概要介绍;其次对形式化分析的计算可靠性研究成果进行分类和总结,并对各种方法的主要思想进行了介绍;最后对该领域未来的研究方向进行了展望.

关键词 密码协议;形式化方法;计算可靠性;信息安全;网络安全

中图法分类号 TP309 **DOI 号** 10.3724/SP.J.1016.2014.00993

A Survey on Computationally Sound Formal Analysis of Cryptographic Protocols

LEI Xin-Feng[1),2)] SONG Shu-Min[2)] LIU Wei-Bing[2)] XUE Rui[1)]

[1)](*State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing* 100093)

[2)](*Beijing Institute of Information Technology, Beijing* 100097)

Abstract There are two different approaches in analysis of cryptographic protocols. One is based on formal methods, and the other is based on computational complexity as modern cryptography does. Abadi and Rogaway tried to reconcile these two approaches in their seminal work. They set up a relation for some formal results that if a security property is proved in formal model, then the corresponding property is also valid in computational model. Motivated by this work, many works appeared in this area. In this survey, we will summarize various approaches on computational soundness of formal methods in analysis cryptographic protocols, present their main ideas, and point out the future research directions in this area.

Keywords cryptographic protocol; formal method; computational soundness; information security; network security

1 引 言

密码协议是保障通信系统安全的重要手段,其安全性直接关系到网络与通信系统的安全. 因此,对密码协议的安全性分析具有重要的意义. 在过去 30 多年里,密码协议分析方法[1]得到了迅速发展,特别是 20 世纪 90 年代以来,人们发展了各种密码协议的形式化分析方法. 近年来,密码协议形式化分析方法的计算可靠性越来越受到关注,成为密码协议分析的一大热点. 本文对密码协议形式化分析在计算可靠性方面的研究成果进行分析总结,并对未来的

收稿日期:2012-07-15;最终修改稿收到日期:2014-01-15. 本课题得到国家自然科学基金(61170280)、中国科学院先导项目(XDA06010701)和中国科学院信息工程研究所密码基金资助.**雷新锋**,男,1973 年生,博士,高级工程师,主要研究方向为密码协议、形式化方法、现代密码学、访问控制等. E-mail: leixinfeng@163.com. **宋书民**,男,1964 年生,博士,高级工程师,主要研究方向为信息安全. **刘伟兵**,男,1966 年生,高级工程师,主要研究方向为信息安全. **薛 锐**,1963 年生,男,博士,研究员,博士生导师. 主要研究领域为现代密码学、密码协议、计算复杂性等.

发展趋势进行展望.

1.1 密码协议

密码协议也称安全协议,它是通过一系列步骤定义的分布式算法,这些步骤确切地规范了两方或多方主体为达到某个安全目标要采取的动作[2].密码协议的安全目标通常包括保密性、认证性、不可否认性、公平性、匿名性等.根据密码协议安全目标的不同,通常可将密码协议分为密钥建立协议、认证协议、公平交换协议、电子投票协议、电子支付协议等.为了保证安全目标的实现,需要对密码协议作安全性的分析,以确信协议能够达到其安全性目标.长期以来,对密码协议的分析基本上是通过经验或观察等方式来完成的.实践证明,这种方法只能发现非常明显的漏洞,而较为微妙的漏洞难以被发现.为了保证协议的安全性,需要更为严谨规范的方法.

1.2 两类分析方法

自 20 世纪 70 年代末、80 年代初开始,形式化方法和现代密码学的发展促使了两类密码协议分析方法的出现,即形式化方法和计算方法,其分析思想截然不同.大体来说,形式化方法(也称符号化方法)基于形式化理论,它将密码协议抽象为符号化的公式,并通过一定的手段来验证或证明协议的安全性;计算方法基于计算复杂性理论,它通常将一些计算困难问题的解决归约为对密码协议的攻击.也就是说,如果敌手能够有效攻击协议,则可利用其攻击策略构造出另一个可有效解决某一困难问题的算法.反过来,由于困难问题目前被认为是缺乏有效解决办法的,说明敌手也不能有效攻击协议,从而表明协议是安全的.

1.2.1 形式化方法

1978 年,Needham 和 Schroeder 在文献[3]中提出了著名的 Needham-Schroeder 协议,并对其安全性进行了简要分析,其中对敌手的能力进行了一定的抽象与假设.一般认为,该文献蕴含了一定的形式化分析思想.1981 年,Dolev 和 Yao[4] 首次明确提出了用形式化方法分析密码协议的思想,并给出了一般协议形式化分析中要遵循的原则和敌手能力刻画模型,被称为 Dolev-Yao 模型(下文简称 DY 模型).DY 模型开启了密码协议形式化分析的先河,并为随后大量出现的协议形式化分析奠定了基础.DY 模型的主要内容包括对密码系统的假设和对敌手能力的假设:(1)假设密码系统是完善的.单向函数的单向性是不可破解的;公共目录是安全的,不会被破坏;公钥是公开的,人人都可以得到;私钥是不公开的,只有其拥有者才知道;(2)假设敌手可发起主动攻击.敌手可获取通过网络传播的任何消息;敌手也可作为合法用户发起和接收会话;敌手可以截断通讯、篡改或转发网络传播的任何消息.

由以上描述可见,在 DY 模型中,任何发送到网络的消息均可被认为发到了敌手那里,敌手可以对该消息作其能力范围内的任何计算.同时,任何从网络收到的消息均可被认为是从敌手那儿收到的经过敌手处理的消息.也就是说,敌手控制了整个网络.另外,由于密码系统是完善的,所以,给定一个密文,只有拥有解密密钥的主体才能得到相应的明文,除此之外,任何主体(包括敌手)都不能从该密文中得到任何关于明文的信息.

1989 年,Burrows、Abadi 和 Needham[5] 提出一种认证逻辑,用于分析认证协议,被称为 BAN 逻辑.BAN 逻辑用模态逻辑的方式将协议的分析过程公理化,并基于该逻辑对众多密码协议进行了分析,发现了一些协议的缺陷,成为密码协议形式化分析的一个里程碑.它有力地促进了该领域的研究,在 20 世纪 90 年代形成一个密码协议形式化分析的高潮.大体说来,这些方法可分为基于逻辑的方法[6-12],基于进程演算的方法[13-14] 以及基于定理证明的方法[15-16] 等.对这些方法的介绍超出本文范围,读者可参考文献[2,17].

1.2.2 计算方法

几乎在密码协议形式化分析方法出现的同时,20 世纪 80 年代初出现了另一类分析方法,即基于计算复杂性的密码学方法,简称计算方法.计算方法源于 Blum 和 Micali[18],Yao[19] 以及 Goldwasser 和 Micali 等人[20] 的工作.该方法用严格的计算复杂性领域的定义和定理奠定了密码学的科学基础,在密码协议的开发与研究中发挥了重要作用.例如,在该方法中,一个加密方案被定义为一个算法的三元组 $\Pi=(\mathcal{K},\mathcal{E},\mathcal{D})$,其中 $\mathcal{K}$ 为密钥生成算法,$\mathcal{E}$ 为加密算法,$\mathcal{D}$ 为解密算法.在对称密钥加密中,给定安全参数 η,$\mathcal{K}$ 可随机生成一个密钥 k;给定消息 m 及密钥 k,并选择随机因子 $\mathcal{E}$ 可输出一个随机加密值 $\mathcal{E}_k(m)$;给定一个密文 c 和密钥 k,$\mathcal{D}$ 可输出一个解密值 $\mathcal{D}_k(c)$.对于合适的 k,m,要求 $\mathcal{D}_k(\mathcal{E}_k(m))=m$.敌手 $\mathcal{A}$ 被定义为一个可访问应答器(Oracle)的图灵机,然后通过一个实验(Game)来定义加密方案的计算安全性.如对称密钥加密方案在选择明文攻击下的不可区分性(或称 IND-CPA 安全)是通过实验 $PrivK^{\mathrm{cpa}}_{\mathcal{A},\Pi}(\eta)$ 定义的:

(1) 运行$\mathcal{K}(1^\eta)$生成密钥.

(2) 敌手$\mathcal{A}$在给定输入1^η和应答器$\mathcal{E}_k(\cdot)$的情况下输出一对相同长度的消息m_0和m_1.

(3) 从$\{0,1\}$中随机选择一个比特b,计算挑战密文$c\leftarrow\mathcal{E}_k(m_b)$并将其交给敌手$\mathcal{A}$.

(4) $\mathcal{A}$继续访问应答器$\mathcal{E}_k(\cdot)$,然后输出比特b'.

(5) 如果$b'=b$,则返回 1,否则返回 0.

根据该实验,IND-CPA 安全可描述为

$$\Pr[PrivK_{\mathcal{A},\Pi}^{\mathrm{cpa}}=1]\leq\frac{1}{2}+negl(\eta),$$

其中$negl$为η的可忽略函数.在证明加密方案的安全性时,主要采用归约的方法来完成.即将某个困难问题的解决归约为对加密方案的攻击,进而可证明敌手对方案的有效攻击是可忽略的.

严格来说,计算方法最初主要用于对密码原语安全性的证明,要采用类似的方法分析协议的安全性,还必须建立适当的模型对协议及其安全性进行建模.1993 年,Bellare 和 Rogaway[21] 提出了一种安全模型,被称为 BR 模型.在 BR 模型中协议被定义为一组应答器,协议的交互过程被定义为敌手和应答器之间的对话,认证性是通过对话匹配来定义的,而保密性是通过敌手"猜测"秘密的优势可忽略来定义的.另外,Canetti 和 Krawczyk[22] 提出了另一种模型以分析密码协议,被称为 CK 模型.在 CK 模型中,首先在理想世界中定义协议,然后将理想协议编译为现实协议,最后证明现实敌手能够模拟理想敌手,从而确保协议在现实世界中也是安全的.随后,文献[23]对 CK 模型进行了扩展,进一步增强了其建模与分析能力.除此之外,还有 BCK 模型[24],模拟模型[25]等.在这些模型的支持下,采用计算方法对密码协议进行分析成为一个活跃的研究课题.

1.3　计算可靠的形式化分析

上述两种方法各有其优点和不足.具体来说:在形式化观点下,消息采用形式化表达式来表达,密码操作是符号化(形式化)表达式空间上的抽象函数,协议的安全性通过形式化表达式来建模,对协议安全性的证明建立在符号化推理的基础上,易于自动化.而在计算观点下,消息是一个比特串,密码操作是比特串上的具体算法,协议的安全性基于敌手成功攻击的概率及计算复杂性,对协议安全性的证明必须采用人工的方法完成,容易出错.总之,形式化模型下的安全性分析相对简洁,且易于利用机器自动验证.缺点在于其抽象化的方法潜在地牺牲了计算方面的可靠性.密码学的方法基于严格的计算复杂性理论,但是其缺点在于只能够证明一个协议是正确的,而对于错误的协议往往不像形式化方法那样能够指出漏洞之所在.同时,往往在证明正确性时,会产生人为的错误.

为了调和这两种方法,2000年,Abadi和Rogway[26]首先对形式化方法的计算可靠性(也称密码学可靠性)进行了研究,从而使得该问题成为密码协议分析领域的一个研究热点.

非形式地说,密码协议形式化分析的计算可靠性是指,如果一个密码协议在形式化模型下被证明是安全的,那么它在计算模型下也是安全的.其意义在于,当对密码协议的形式化分析能够保证计算可靠性时,就可以避开用复杂的计算方法进行协议分析,而是用形式化方法进行分析,但同样可保证其在计算模型下的安全性.为了保证密码协议形式化分析的计算可靠性,近十年来,人们进行了各种探索,提出了各种不同的方法.

本文将从基于映射的方法、模拟的方法、已有形式化方法的计算可靠性以及对计算方法的直接形式化等方面对密码协议形式化分析的计算可靠性进行综述.大体来说,前三种情况是由形式化模型或理想系统下的安全性入手,然后通过映射、模拟、解释等方式,证明计算模型下或实际系统中的安全性,属于一种间接的方法.其中基于映射的方法的主要特征是在形式化模型和计算模型之间建立映射关系;基于模拟的方法通常在理想系统和实际系统之间建立模拟关系;对已有形式化方法的计算可靠性研究包括对协议逻辑、进程演算、串空间以及安全信息流等形式化方法在分析密码协议时的计算可靠性研究.与前三种情况不同,对计算方法的直接形式化从计算模型下的安全性入手,然后通过抽象的方法对其进行形式化,并在形式化系统中对协议的安全性进行分析,属于直接的方法.这种方法目前也包括两类,即基于逻辑的方法和基于进程演算的方法.基于逻辑的方法直接对不可区分性作符号化抽象,采用逻辑公理的形式进行推理,以证明协议的安全性.基于进程演算的方法将概率引入进程演算,用概率意义下的观察等价刻画不可区分性,以证明协议的安全性.

1.4　组织结构

本文第 2 节将对基于映射的方法进行阐述;第 3 节介绍基于模拟的方法;对已有方法的计算可靠性研究将在第 4 节给出;接着,在第 5 节介绍目前在对计算方法直接进行形式化方面的一些结果;最后,

对全文进行总结,并对当前和今后一段时间的发展趋势进行展望.

2 基于映射的方法

在基于映射的方法中,通常定义两种模型,即符号化模型和计算模型,同时给出一个映射函数,将符号化模型中的符号映射到计算模型中,从而建立符号化方法的计算可靠性.本文根据映射主体的不同,将映射分为消息映射和迹映射,它们分别将符号模型下的符号消息和符号迹映射到计算模型下.通常来说,消息映射针对协议中的静态消息,只能处理被动敌手的情况,而迹映射可反映协议中的动态行为,使得处理主动敌手的情况成为可能.

2.1 消息映射:AR 逻辑

Abadi 和 Rogaway 首先在文献[26-27]中给出了一种研究计算可靠性的方法(通常被称为 AR 逻辑).AR 逻辑的主要思路是:通过定义消息模型,在符号化消息之间建立等价关系,然后定义一种映射关系,将符号化消息映射为计算意义下的位串消息,最后证明形式化意义下的等价关系蕴涵计算意义下的不可区分性.

2.1.1 形式化加密与模式等价

在 AR 逻辑中,消息由形式化表达式来表示.设 Keys 表示密码的集合,Bool 表示布尔值的集合,即 $\{0,1\}$,则表达式可形式化定义如下:

$M,N::=$	表达式
K	密钥($K\in$Keys)
i	位($i\in$Bool)
(M,N)	级联
$\{\!\{M\}\!\}_K$	加密

需要说明的是,AR 逻辑中对以上表达式进行了一定的限制,即要求表达式是非循环的.直观地说,一个表达式是非循环的,是指表达式中不存在一个密钥直接或间接地对自身进行加密的情况.严格地说,如果存在 N,使得 $\{\!\{N\}\!\}_K$ 为 M 的一个子表达式,且 K' 在 N 中出现,则称 K 在 M 中加密了 K'.对每一 M,以上定义了一个密钥上的二元关系,即加密关系.如果与 M 相关的加密关系是循环的(非循环的),则称 M 是循环的(非循环的).如 $\{\!\{K\}\!\}_K$ 和 $(\{\!\{K\}\!\}_{K'},\{\!\{K'\}\!\}_K)$ 都是循环的,而 $(\{\!\{K\}\!\}_{K'},\{\!\{0\}\!\}_K)$ 是非循环的.

为了定义表达式之间的等价关系,首先定义一种获取关系 $M\vdash N$,其中,M 和 N 为表达式.直观地说,$M\vdash N$ 表示 N 可以从 M 中计算得出.形式化地定义该关系如下:

$M\vdash 0$,$M\vdash 1$,且 $M\vdash M$;

如果 $M\vdash N_1$ 且 $M\vdash N_2$,则 $M\vdash(N_1,N_2)$;

如果 $M\vdash(N_1,N_2)$,则 $M\vdash N_1$ 且 $M\vdash N_2$;

如果 $M\vdash\{\!\{N\}\!\}_K$ 且 $M\vdash K$,则 $M\vdash N$;

如果 $M\vdash N$ 且 $M\vdash K$,则 $M\vdash\{\!\{N\}\!\}_K$.

$M\vdash N$ 的定义刻画了在攻击者没有关于 M 中所用到密钥 K 的先验知识的情况下,可从 M 中获取的消息.从以上定义可派生出一种更一般的定义:在有关于 K 的先验知识的情况下从 M 中获取 N,等价于在没有先验知识的情况下从 (M,K) 中获取 N,这一思想可由 $p()$ 函数刻画.以下给出 $p()$ 函数的形式化定义,其中,引入一个符号"□",它表示一个攻击者不能解密的密文.

$$p(K,T)=K(K\in\text{Keys}),$$
$$p(i,T)=i(i\in\text{Bool}),$$
$$p((M,N),T)=(p(M,T),p(N,T)),$$
$$p(\{\!\{M\}\!\}_K,T)=\begin{cases}\{\!\{p(M,T)\}\!\}_K, & \text{如果 } K\in T\\ \square, & \text{否则}\end{cases}.$$

直观地说,假设 M 为一表达式,T 为一密钥集合,则 $p(M,T)$ 表示在具备先验知识 T 的情况下,攻击者可从 M 中获取的消息.如果攻击者在其先验知识的帮助下不能解密消息 M 中的某密文部分,则该部分将被替换为"□".

在 $p()$ 函数的辅助下,可定义表达式的模式作为对表达式的扩展.直观上,表达式的模式就是该表达式所呈现给攻击者的消息形式.给定一个表达式 M,用 $pattern(M)$ 表示 M 的模式,则 $pattern(M)$ 可由 $p()$ 函数定义如下:

$$pattern(M)=p(M,\{K\in\text{Keys}\mid M\vdash K\}).$$

模式的集合通常用 Pat 表示.

接下来就可定义模式等价了.称两个表达式是基于模式等价的,记为 $M\cong N$,当且仅当存在一个一一对应的密钥的重命名(即对密钥的替换)σ,使得 $pattern(M)=pattern(N)\sigma$.将这种基于模式的等价简称为模式等价.

以下给出一个关于模式等价的例子.设 $M=(\{\!\{\{\!\{K_1\}\!\}_{K_2}\}\!\}_{K_3},K_3)$,$N=(\{\!\{\{\!\{0\}\!\}_{K_4}\}\!\}_{K_5},K_5)$,则

$$\begin{aligned}pattern(M)&=p(M,\{K\in\text{Keys}\mid M\vdash K\})\\&=p(M,\{K_3\})\\&=(\{\!\{\square\}\!\}_{K_3},K_3),\end{aligned}$$

$$\begin{aligned}pattern(N)&=p(N,\{K\in\text{Keys}\mid N\vdash K\})\\&=p(N,\{K_5\})\\&=(\{\!\{\square\}\!\}_{K_5},K_5).\end{aligned}$$

这时，存在密钥重命名 $\sigma=(K_3/K_5)$（表示将密钥 K_5 重命名为 K_3），使得

$$pattern(M)=(\{\!|\square|\!\}_{K_3},K_3)=pattern(N)\sigma,$$

从而有 $M\cong N$.

2.1.2　不可区分性及加密方案的安全性

以上给出了在形式化模型下模式等价的概念. 类似地，在计算模型下有不可区分的概念.

函数 $\varepsilon:\mathbb{N}\to\mathbb{R}$ 为一可忽略函数，是指对所有 $c>0$ 存在 N_c 使得对所有 $\eta\geq N_c$，有 $\varepsilon(\eta)\leq\eta^{-c}$. 总体(ensemble)是串的分布簇 $D=\{D_\eta\}$，每一个 η 对应一个分布. 用 $x\leftarrow D_\eta$ 表示按照分布 D_η 随机抽样一个元素 x. 设 D 和 D' 均为总体，如果对所有概率多项式时间的(Probabilistic Polynomial-Time，PPT)敌手 $\mathcal{A}$，函数

$$\varepsilon(\eta)=\Pr[x\leftarrow D_\eta:\mathcal{A}(\eta,x)=1]-\Pr[x\leftarrow D'_\eta:\mathcal{A}(\eta,x)=1]$$

为可忽略函数，则 D 和 D' 是不可区分的(或计算不可区分的)，记为 $D\approx D'$.

加密方案的安全性可用类似于不可区分性的概念来定义. 设 $\Pi=(\mathcal{K},\mathcal{E},\mathcal{D})$ 为一加密方案，η 为安全参数，$\mathcal{A}$ 为敌手，定义 $\mathrm{Adv}_{\Pi[\eta]}(\mathcal{A})$ 为

$$\Pr[k,k'\leftarrow\mathcal{K}:\mathcal{A}^{\mathcal{E}_k(\cdot),\mathcal{E}_{k'}(\cdot)}(\eta)=1]-\Pr[k\leftarrow\mathcal{K}:\mathcal{A}^{\mathcal{E}_k(0),\mathcal{E}_k(0)}(\eta)=1],$$

其中，$\mathcal{A}^{\mathcal{O}}$ 表示一个可访问应答器 $\mathcal{O}$ 的算法. 如果 $\mathrm{Adv}_{\Pi[\eta]}(\mathcal{A})$ 是 η 的一个可忽略函数，则称加密方案 Π 是安全的. 文献[26]将以上安全定义称为类型-0 安全.

该安全定义可以刻画 3 层意思：(1) 给定两个密文，敌手判断它们是否是对同一明文加密的结果；(2) 给定两个密文，敌手判断它们是否是用同一密钥加密的结果；(3) 敌手由密文的长度推测出明文的长度. 对以上 3 方面作不同的选择还可得到不同的安全定义.

需要注意的是，在加密方案安全性的定义中实际上排除了循环加密的情况. 直观地说，当敌手可访问一个关于密钥 k 的应答器时，敌手自己是不拥有该密钥的，从而敌手也不可能将 k 提交给应答器要求加密. 这一点 Goldwasser 和 Micali 在文献[20]已经注意到了，Abadi 和 Rogaway 在文献[26]中明确地提出这一问题. 这也是为什么前面要对形式化表达式附加非循环要求的原因.

2.1.3　通过消息映射建立计算可靠性

为了调和形式化观点和计算观点，可采用以下 2 个步骤：(1) 给定一个加密方案，可给表达式关联一个总体；(2) 在适当的假设下证明模式等价蕴含总体的不可区分性.

首先，为表达式关联总体. 设 $\Pi=(\mathcal{K},\mathcal{E},\mathcal{D})$ 为一个加密方案，η 为安全参数，对每一个表达式 $M\in\mathrm{Exp}$ 关联一个总体 $[\![M]\!]_\Pi=\{[\![M]\!]_{\Pi[\eta]}\}_{\eta\in\mathbb{N}}$. 该关联可由下面给出的初始化过程 Initialize 和转换过程 Convert 来完成. 其中 $Keys(M)$ 表示所有在 M 中出现的密钥符号，用 $\langle x_1,\cdots,x_k\rangle$ 表示由 $x_1,\cdots,x_k$ 所得到的串的编码：

过程 1.　$\mathrm{Initialize}_\eta(M)$.

FOR $K\in Keys(M)$ DO $\tau(K)\leftarrow\mathcal{K}(\eta)$

过程 2.　$\mathrm{Convert}(M)$.

IF $M=K$ WHERE $K\in\mathrm{Keys}$ THEN
  RETURN $\langle\tau(K),\text{"key"}\rangle$
IF $M=b$ WHERE $b\in\mathrm{Bool}$ THEN
  RETURN $\langle b,\text{"bool"}\rangle$
IF $M=(M_1,M_2)$ THEN
  RETURN $\langle\mathrm{Convert}(M_1),\mathrm{Convert}(M_2),\text{"pair"}\rangle$
IF $M=\{\!|M_1|\!\}_K$ THEN
  $x\leftarrow\mathrm{Convert}(M_1)$
  $y\leftarrow\mathcal{E}_{\tau(K)}(x)$
  RETURN $\langle y,\text{"ciphertext"}\rangle$

辅助的初始化过程将每个 $Keys(M)$ 中的密钥映射到一个由 $\mathcal{K}(\eta)$ 生成的密钥 $\tau(K)$. 为避免混淆，对每个串形式的表示加上一个类型标签(如“key”，“bool”，“pair”，“ciphertext”等).

文献[26]采用混合论证(hybrid argument)的方法证明了基于模式等价的计算可靠性定理：

设 M 和 N 为非循环表达式，Π 为类型-0 安全加密方案. 如果 $M\cong N$，则 $[\![M]\!]_\Pi\approx[\![N]\!]_\Pi$.

2.2　AR 逻辑的扩展

AR 逻辑[26-27]在计算可靠性的研究上具有开创性的意义. 它开创了符号安全性分析计算可靠性的先河，但 AR 逻辑本身过于简单，例如，AR 逻辑排除了循环加密，而现实中循环加密的情况是有可能出现的；AR 逻辑中的消息仅包含了对称加密的情况而没有涉及其它密码学原语；AR 逻辑是可靠的，但并不是完备的；另外，它仅讨论了协议传输中的静态消息，而未考虑协议的动态行为，因此，AR 逻辑很难直接用于对密码协议的安全性分析；更重要的是，它只考虑了被动敌手的攻击而未考虑主动敌手的攻击. 针对这些问题，人们进行了大量的研究，并从不同角度对 AR 逻辑进行了改进. 以下对一些直接基于 AR 逻辑的改进进行介绍.

2.2.1 AR 逻辑的完备性

文献[26]证明了 AR 逻辑的计算可靠性,但 AR 逻辑并不具备完备性.简单地说,在 AR 逻辑中,虽然等价性蕴涵不可区分性(可靠性),但不可区分性并不蕴涵等价性(完备性)[26].原因很简单,如果应用表达式转换过程时,所给的表达式引起加密方案对其明文空间之外的串进行加密,那么,即使表达式不等价,与其所关联的总体也将是相同的.这说明,在输入反常的情况下 AR 逻辑的完备性是不成立的.那么是不是将加密的输入限定在明文空间之内,完备性就成立了呢?答案也是否定的.文献[28]在类型-0 加密方案下给出一种反例来避免以上提到的反常情形,但同样可得到否定的结论,即 AR 逻辑是不完备的.这表明了类型-0 的安全性不足以支持 AR 逻辑的完备性,原因在于对于给定的密文和密钥,不能确定该密文是否是用该密钥加密的.

进一步地,文献[28]给出 AR 逻辑完备性的充分条件,即加密方案满足无混淆(Confusion-free)的条件.无混淆属性可非形式化地描述如下:随机地独立地生成两个密钥,当用其中一个密钥加密时,用另一个密钥将只能以可忽略的概率解密.形式化地,设 $D=\{D_1(\eta),D_2(\eta),\cdots,D_l(\eta)\}$ 为分布总体的有限集,$\Pi=(\mathcal{K},\mathcal{E},\mathcal{D})$ 为一个对称加密方案,其安全参数为 η,当且仅当存在一个可忽略函数 ν_D 使得对任意 $1\leq i\leq l$ 有 $\Pr[k_1,k_2\leftarrow\mathcal{K}(\eta),x\leftarrow D_i(\eta):\mathcal{D}_{k_1}(\mathcal{E}_{k_2}(x))\neq\perp]\leq\nu_D(\eta)$,则称 Π 对于 D 来说是无混淆的.其中,$\perp$ 表示无效消息,即不在明文空间内.可以证明,当一个加密方案为安全的签密方案时,该方案对任意多项式时间抽样分布的有限集来说是无混淆的.

Horvitz 在文献[29]中进一步给出了完备性的充要条件,即表达式的弱密钥认证测试(Weak Key-Authenticity Tests for Expressions,WKA-EXP).对于一个给定的加密密钥,WKA-EXP 测试主要用于对"一个密文及其加密密钥"和"一个密文及一随机密钥"进行区分.如果区分的概率是可忽略的,则称该加密方案通过 WKA-EXP 测试.形式化地,设 $\Pi=(\mathcal{K},\mathcal{E},\mathcal{D})$ 为加密方案,其安全参数为 η,M_1,M_2 为非循环表达式,$\mathcal{A}$ 为一算法,定义 $\mathrm{Adv}^{\mathrm{wka-exp}}_{\Pi[\eta],M_1,M_2}(\mathcal{A})$ 为 $\Pr[e\leftarrow[\![M_1]\!]_{\Pi[\eta]},k\leftarrow\mathcal{K}(\eta),c\leftarrow\mathcal{E}_k(e):\mathcal{A}(\eta,c,k)=1]-\Pr[e\leftarrow[\![M_2]\!]_{\Pi[\eta]},(k,k')\leftarrow\mathcal{K}(\eta),c\leftarrow\mathcal{E}_k(e):\mathcal{A}(\eta,c,k')=1]$,当存在 PPT 算法 $\mathcal{A}$,使得 $\mathrm{Adv}^{\mathrm{wka-exp}}_{\Pi[\eta],M_1,M_2}(\mathcal{A})$ 可忽略时,称 Π 对 M_1,M_2 来说是通过 WKA-EXP 测试的.

除此之外,Bana[30] 及 Adão 等人[31] 表明,在更一般的(弱)加密系统下完备性同样成立.

2.2.2 密钥循环

在对 AR 逻辑的改进中,密钥循环问题一直是研究的焦点之一.粗略地看,密钥循环问题似乎没有必要受到如此重视,因为设计良好的协议一般不会出现加密循环.然而,越来越多的研究表明,这一问题有着重要的应用场合.例如,一个备份系统可能将加密密钥保存在将备份的盘上,然后又用该密钥对整个盘进行加密.另外,可能协议本身的设计中就包含密钥循环.如文献[32]给出的一种非传递匿名证书系统中就包含有密钥循环.更重要的是,循环加密在形式化加密的计算可靠性方面有着很关键的作用.从 AR 逻辑对消息(表达式)的形式化定义可知,如果不额外附加条件,消息中的密钥循环是允许的.而在计算模型下,标准的安全定义[20]中是不含密钥循环的.为了保持计算可靠性,AR 逻辑的做法是人为限制形式化模型下消息中不包含密钥循环.那么一个自然的问题便是,如果形式化模型中包含了循环加密,相应的形式化加密是否具备计算可靠性.近年来,人们围绕这一问题展开了一系列研究[26,33-36].归纳起来,对这一问题主要有 2 种处理方法.

一种方法是通过弱化形式化加密,强化形式化敌手以达到其计算可靠性,即认为形式化表达式中可以存在循环加密,但并不保证这种循环加密的安全性.这一方法的代表是 Laud[33].在文献[33]中,Laud 通过增加特定的规则,赋予了敌手破解循环加密的能力,并证明了这一形式化方案的计算可靠性.

另一种方法是通过强化加密方案以达到计算可靠性,即认为形式化表达式中可以存在循环加密,且是安全的,但是,在计算模型下的加密方案被强化为包含密钥循环的加密方案.这一方法的代表是 Adão 等人[34-35],在文献[34-35]中,Adão 等人并没有对形式化模型做任何限制,而是在计算模型中采用了一种由 Black 等人[37] 提出的可确保循环加密安全的方案,即密钥相关消息(Key Dependent Message,KDM)安全加密方案.越来越多的工作致力于构造 KDM 安全的加密方案[37-39],但其大多数或者是在随机应答器模型(Random Oracle Model,ROM)下给出的[37],或者弱化了 KDM 安全概念[38],或者对敌手进行了一定限制[39].文献[40]表明,如果在加密方案安全性的归约证明中,采用黑盒的方式对待敌手和查询函数,那么要证明 KDM 安全是不可能的.因此,构造这样一种方案并不是一件容易的事.

在文献[26,33]中,对敌手获取密钥的能力都采

用了最小不动点的方式进行定义，即给定一个消息，假定敌手开始时不拥有相关密钥，但通过逐步的消息分拆或解密等方式（由反映敌手能力的规则定义）可获得越来越多的密钥，直到其所能获得的密钥不再增加，这时所得到的密钥就是敌手可获得的密钥. 在这种方式下，对密钥循环的处理必须人为说明，如声明表达式是非循环的，或者给出额外的规则，如人为规定循环加密是不安全的. 2010 年，Micciancio[36] 采用了最大不动点的方式对敌手获取密钥的能力进行定义，即给定一个消息，及该消息中包含的所有密钥，然后用这些密钥对消息进行分拆或解密，得到一些密钥，这些密钥必然比消息中的所有密钥要少（消息中仅用于加密的密钥是无法通过分拆或解密而得到的）. 接着用所得到的密钥再次用于消息的解密，由于可用密钥数量减少，所得到的密钥只会更少，一直进行下去，直到所得到的密钥不再减少. 将最后所得到的密钥作为敌手可获取的密钥. 通俗地说，对于一个形式化表达式，不管它是否存在密钥循环，如果表达式无法保证对某密钥的隐藏，则认为该密钥可被敌手获得. 其实质也是从另一个角度增加敌手的能力，但它对循环密钥的处理方式更为自然、通用，可同时适用于非循环表达式和循环表达式，减少了对形式化加密的人为干预. 随后，Micciancio[41] 进一步将该方法用于消息中包含伪随机密钥的情况，对密钥的部分信息进行了讨论，并证明了这种情况下形式化加密的计算可靠性.

Abadi 和 Warinschi[42] 基于文献[43]对 AR 逻辑在秘密共享方面进行了扩展，使得在采用密码学方法实现访问控制策略时可保证其计算可靠性. 但该扩展同样未考虑密钥循环问题. Lei 等人[44] 通过对文献[36,42]的扩展，证明了同时存在密钥循环与秘密共享时形式化加密的计算可靠性.

2.2.3　消息序列与适应性攻击

在前述 AR 逻辑的扩展中，均只考虑了敌手可窃听到的消息本身，而不关心这些消息出现在环境中的先后顺序或是否重复. 事实上，很多时候同一消息可能会多次出现在协议中. 而且，系统所使用密码的体系也可能会影响协议的控制流. 例如，一种程序生成了两个密钥，并对其进行比较，由比较的结果来决定下一步执行程序哪个分支. Abadi 等人在文献[45]中对文献[26]做了扩展，其中采用一种简单的编程语言对系统进行描述，该编程语言对消息的描述提供了更强的支持，并可反映程序执行中的控制流. 在讨论计算可靠性时，首先用该语言对协议进行描述；然后，将相应的程序映射为一个消息序列，并用该消息序列表示程序的运行迹；最后通过将消息序列映射到计算模型下，证明了相应的计算可靠性定理. 在文献[45]中，虽然只考虑了被动敌手，但它用消息序列表示协议运行迹的思路，向处理主动敌手情况迈进了一步.

进一步地，文献[46]对适应性攻击下符号加密的计算可靠性进行了讨论. 适应性攻击是指敌手在攻击中使用了适应性策略，即当前获得消息的能力依赖于以前所得到的消息. 为解决由适应性攻击可能引发的问题，文献[46]在文献[26]中对形式化表达式附加非循环条件的基础上进一步附加了语法限制，即将消息中密钥的使用分为两个阶段：密钥分发阶段和密钥应用阶段. 在密钥分发阶段，密钥可以作为一般消息使用，而在密钥应用阶段，密钥仅被用来加密其它消息. 在这种执行模式下，敌手可通过与执行环境的交互适应性改变协议的执行流程. 但要注意，敌手并不能修改或删除合法主体发送或收到的消息. 从这个意义上说，适应性攻击强于纯被动攻击，这种改进更接近现实的协议执行，但同样将协议环境限制在被动攻击下.

2.2.4　其它扩展

除了在完备性及密钥循环方面的扩展外，还有许多研究计算可靠性的文献对 AR 逻辑进行了扩展. 文献[47]表明 DY 敌手可被看成所有现实敌手的有效抽象，与 AR 逻辑不同的是它采用了公钥密码系统. 文献[48]将 AR 逻辑中的密钥由原子密钥扩展为组合密钥，即任何表达式都可作为密钥. 文献[49-50]对 AR 逻辑在 Hash 函数方面进行了扩展，并分别证明了扩展后的计算可靠性和完备性.

2.3　迹映射：MW 方法

消息映射仅涉及到静态的消息，而未涉及到动态的行为，因此，仅适用于被动攻击. 本节对迹映射[51] 的方法进行介绍，此处的迹中不仅包含了静态的消息而且包含了动态的行为，因而适用于存在主动敌手时形式化加密的计算可靠性. 迹映射的方法源于 Micciancio 和 Warinschi 的工作[51]，下文将文献[51]中的方法简称为 MW 方法.

2.3.1　协议的执行

在 MW 方法中，协议中传送的消息可用类似 AR 逻辑中的语法来描述，协议可由类似于上节提到的消息序列来描述. 将协议的执行看成是攻击者与运行诚实主体程序的应答器之间的交互. 敌手可向应答器发出如下 3 种指令：

(1) $new(A,B)$:开始执行协议的一个新的实例,其中 A 为发起者,B 为响应者.应答器收到该指令后选择一个新的会话标识 s,并开始执行一个由 A 和 B 运行的新的协议实例,然后将协议标识符 s 和由 A 发出的第一个消息返回给敌手.

(2) $send(s:I,m)$:向会话 s 的发起者发送消息 m.应答器收到该指令后更新发起者的状态,并将发起者对消息 m 的响应返回给敌手.

(3) $send(s:R,m)$:向会话 s 的响应者发送消息 m.应答器收到该指令后更新响应者的状态,并将响应者对消息 m 的响应返回给敌手.

对于协议的执行来说,有 2 种不同的敌手模型:即抽象模型和具体模型.抽象敌手(也称 DY 敌手)使用符号化表达式与主体通信,而具体敌手(也称实际敌手)使用的是由运行特定加密算法而得到的位串.设集合 M 表示抽象敌手在协议执行中某点所拥有的消息,具体来说,M 包括主体标识集 $Id=\{A_1,A_2,\cdots\}$、公钥集 $Keys=\{K_1,K_2,\cdots\}$、由敌手生成的新鲜值集合 $Nonce$ 以及表示与敌手合谋的腐化主体标识集 C.用 $closure(C,M)$ 表示敌手可从 M 中计算出的消息集合,$closure(C,M)$ 可形式化定义如下:

(1) $M\subseteq closure(C,M)$.

(2) 如果 $T_1,T_2\in closure(C,M)$,则 $(T_1,T_2)\in closure(C,M)$.

(3) 如果 $(T_1,T_2)\in closure(C,M)$,则 $T_1,T_2\in closure(C,M)$.

(4) 如果 $T\in closure(C,M)$,$K\in Keys$,则 $\{\!|T|\!\}_K\in closure(C,M)$.

(5) 如果 $\{\!|T|\!\}_{K_i}\in closure(C,M)$,其中,$K_i$ 为 A_i 的公钥,且 $A_i\in C$,则 $T\in closure(C,M)$.

实际敌手被限制为 PPT 敌手,但可执行任意操作.与抽象敌手类似,实际敌手也可向应答器发送 3 类指令:$new(i,j)$,$send(s:I,m)$ 及 $send(s:R,m)$,但这时的消息为位串而不是形式化表达式.类似地,应答器返回的消息也是由主体使用其密钥及加密函数计算出的位串.

2.3.2 两种状态迹

设 $\mathcal{F}$ 为抽象模型下符号表达式的集合,它是由基本项 $\mathcal{F}^{\text{const}}$(包括标识符、密钥和新鲜数)根据项的语法构造而来.$\mathcal{C}_\eta$ 为具体模型下位串的集合(η 为安全参数),它是由基本位串 $\mathcal{C}_\eta^{\text{const}}$ 通过并置或加密而构造的.$\mathcal{O}^{\mathcal{F}}$ 和 $\mathcal{O}^{\mathcal{C}}$ 分别为抽象模型和具体模型下的应答器.$Identifiers$ 为协议抽象描述中的主体标识集合,SId 为所有可能的会话集.则协议执行中由 $\mathcal{O}^{\mathcal{F}}$ 和 $\mathcal{O}^{\mathcal{C}}$ 维护的全局状态可分别由二元组 (F,k) 和 (f,l) 给出:

$$F:SId\times\{I,R\}\to(Identifiers\to\mathcal{F}^{\text{const}}),$$
$$k:SId\times\{I,R\}\to(\mathbb{N}\cup\{\surd\}),$$
$$f:SId\times\{I,R\}\to(Identifiers\to\mathcal{C}_\eta^{\text{const}}),$$
$$l:SId\times\{I,R\}\to(\mathbb{N}\cup\{\surd\}),$$

$F(s,I)$ 和 $f(s,I)$ 分别表示形式化执行与具体执行下会话 s 中发起者 I 的本地状态,$F(s,R)$ 和 $f(s,R)$ 与之类似,但针对响应者 R.函数 k 用以反映协议在形式化执行中主体(I 或 R)是否期望接收消息以及期望接收哪个消息.具体来说,如果 k 的返回值属于自然数集 $\mathbb{N}$,则它表示主体所期望的下一消息编号;如果 k 的返回值为 $\surd$,则它表示主体完成了协议的执行,不再期望接收消息.函数 l 的功能与 k 类似,但针对协议的具体执行.

形式化敌手 $\mathcal{A}_f$ 由 $send$ 类型的查询列表表示(简单起见,假设所有可能的会话已经发起).如果敌手 $\mathcal{A}_f$ 发送的每个查询都在相应的 $closure(C,M)$(参见 closure 的定义)中,则称该敌手是 DY 有效的.敌手和应答器之间的交互过程由 $\mathcal{O}^{\mathcal{F}}$ 所经历的状态序列 $((F_0,k_0),(F_1,k_1),\cdots)$ 表示,并称之为 $\mathcal{A}_f$ 执行的形式化状态迹,记为 $STr(\mathcal{A}_f,\mathcal{O}^{\mathcal{F}})$.所有形式化迹的集合用 $\mathcal{F}Strace$ 表示.

类似地,可定义具体模型下的协议执行,这时的执行过程被随机化了,因此,对实际敌手 $\mathcal{A}_c$ 和环境应答器 $\mathcal{O}^{\mathcal{C}}$ 分别加入了随机因子 $R_{\mathcal{A}}$ 和 $R_{\mathcal{O}}$ 作为参数.从而,具体状态迹由 $((f_0,l_0),(f_1,l_1),\cdots)$ 描述,记为 $STr(\mathcal{A}_c(R_{\mathcal{A}}),\mathcal{O}^{\mathcal{C}}(R_{\mathcal{O}}))$.所有具体迹的集合用 $STrace$ 表示.

2.3.3 基于迹映射的计算可靠性

为建立形式化证明的计算可靠性,在 MW 方法中定义了一个函数 $\mathcal{R}:\mathcal{F}^{\text{const}}\to\mathcal{C}_\eta^{\text{const}}$,当 $\mathcal{R}$ 为单射,且 $\mathcal{R}(\mathcal{F}^k)\subseteq\mathcal{C}_\eta^k$,$\mathcal{R}(\mathcal{F}^n)\subseteq\mathcal{C}_\eta^n$,$\mathcal{R}(\mathcal{F}^i)\subseteq\mathcal{C}_\eta^i$ 时,称 $\mathcal{R}$ 为描述函数.此处的 k,n,i 分别为密钥(keys)、新鲜数(nonces)和标识符(identities)的缩写,$\mathcal{F}^k$,$\mathcal{F}^n$ 和 $\mathcal{F}^i$ 分别表示 $\mathcal{F}^{\text{const}}$ 中的密钥、新鲜数和标识符部分,而 $\mathcal{C}_\eta^k$,$\mathcal{C}_\eta^n$ 和 $\mathcal{C}_\eta^i$ 分别表示 $\mathcal{C}_\eta^{\text{const}}$ 中的密钥、新鲜数和标识符部分.通过描述函数可在形式化迹和具体迹之间建立一种映射关系.设 $fstr=((F_0,k_0),(F_1,k_1),\cdots,(F_n,k_n))$ 为形式化迹,$cstr=((f_0,l_0),(f_1,l_1),\cdots,(f_n,l_n))$ 为具体迹,$\mathcal{R}:\mathcal{F}\to\mathcal{C}$ 为一描述函数,如果对所有的 $1\leq i\leq n$ 有 $\mathcal{R}(F_i)=f_i$,且 $k_i=l_i$,则称 $cstr$ 是 $fstr$ 通过 $\mathcal{R}$ 的实现,记为 $fstr\leq_{\mathcal{R}}cstr$.如果存在这样的描述函数使得 $fstr\leq_{\mathcal{R}}cstr$ 成立,则称 $cstr$ 是

$fstr$ 的实现，记为 $fstr \leq cstr$. 可以证明，通过固定敌手和环境应答器的随机因子所得到的具体迹，可以以极大概率实现一个 DY 有效敌手的形式化迹. 形式化地，设一个协议实现中所使用的加密方案是 IND-CCA 安全的，则对任意具体敌手 $\mathcal{A}_c$ 满足

$$\Pr_{R_{\mathcal{A}},R_{\mathcal{O}}}[\text{存在有效的}\mathcal{A}_f\text{使得}\varphi\text{成立}] \geq 1-\nu(\eta).$$

其中

$$\varphi := STr(\mathcal{A}_f,\mathcal{O}^{\mathcal{F}}) \leq STr(\mathcal{A}_c(R_{\mathcal{A}}),\mathcal{O}^{\mathcal{C}}(R_{\mathcal{O}})).$$

这一结论通常被称为映射引理.

给定协议 $\mathcal{P}$，在抽象模型下其安全属性可用形式化状态迹上的断言 P_f 来表示，从集合论的角度来看，P_f 可表示 $\mathcal{F}Strace$ 的一个子集. 对每个安全属性 $P_f \subseteq \mathcal{F}Strace$，如果对所有有效的形式化敌手 $\mathcal{A}_f$ 有 $STr(\mathcal{A}_f,\mathcal{O}^{\mathcal{F}}) \in P_f$，则称协议 $\mathcal{P}$ 满足属性 P_f，记为 $\mathcal{P} \models_f P_f$. 相应地，具体安全属性是具体状态迹上的断言 P_c. 对每个具体安全属性 $P_c \subseteq \mathcal{C}Strace$，如果对所有 PPT 敌手 $\mathcal{A}_c$ 有

$$\Pr_{R_{\mathcal{A}},R_{\mathcal{O}}}[STr(\mathcal{A}_c(R_{\mathcal{A}}),\mathcal{O}^{\mathcal{C}}(R_{\mathcal{O}})) \in P_c] \geq 1-\nu(\eta),$$

则称协议 $\mathcal{P}$ 满足属性 P_c，记为 $\mathcal{P} \models_c P_c$. 其中，$R_{\mathcal{A}}$ 和 $R_{\mathcal{O}}$ 为适当长度（即安全参数 η 的多项式长度）的随机串，$\nu(\cdot)$ 为可忽略函数.

文献[51]在映射引理的基础上证明了存在主动敌手时形式化安全分析的如下计算可靠性定理：设 P_f 和 P_c 分别为形式化安全属性和具体安全属性，且有

$$(\forall fstr \in \mathcal{F}Strace, \forall cstr \in \mathcal{C}Strace)$$
$$((fstr \in P_f \wedge ftr \leq cstr) \Rightarrow cstr \in P_c).$$

如果加密方案是 IND-CCA 安全的，则有

$$(\mathcal{P} \models_f P_f) \Rightarrow (\mathcal{P} \models_c P_c).$$

2.4　MW 方法的扩展

MW 方法在研究计算可靠性时引入了主动敌手，使得对密码协议的分析更接近现实环境. 文献[52-56]等对其作了进一步扩展，使其更为通用.

Cortier 和 Warinschi[52] 对 MW 方法在数字签名方面进行了扩展，同时讨论了协议保密性的建模及其计算可靠性. 在文献[52]中，形式化消息不仅包含了加密操作，而且包含了解密、签名、验证等操作，从而对形式化敌手的消息处理能力也做了进一步的刻画. 在协议的执行中，敌手的行为有 3 种：在协议执行之前腐化协议主体（corrupt）、发起新会话（new）以及发送消息（send）. 这些行为表明敌手是主动敌手. 在安全属性方面，更为具体地给出了保密性的表达. 在计算可靠性方面，通过给出相应的映射引理，基于实现中所使用签名方案和加密方案的安全性给出了相应的可靠性定理. 文献[53]对 MW 方法的扩展中同样允许消息签名的使用. 另外，由于文献[51]中的计算可靠性基于 IND-CCA 安全，只考虑了一般消息的加密，而不便于考虑对密钥的加密，从而限制了协议中密钥的发送，在文献[53]中使用了不同形式的加密方案及签名方案，消除了以上限制.

文献[54]将 MW 方法应用于通用可复合框架（简称 UC 框架，下文将对 UC 框架进行介绍），建立了通用可复合的符号化安全性分析方法. 其核心工具就是在具体协议和符号协议之间建立对应关系的映射引理. 该映射引理是文献[51]中的映射引理在 UC 框架下的应用. 其主要思路是：首先，定义简单协议（由复合协议分解而来）在 UC 框架下的执行迹；其次，定义符号模型下协议的符号迹；然后定义迹映射，将具体迹映射到符号迹；最后证明基于符号迹的安全属性具备计算可靠性.

文献[55-56]通过将映射引理应用于 Hash 函数对 MW 方法进行了扩展.

3　基于模拟的方法

基于模拟的方法的基本思路是：为了实现某个安全的属性，首先定义一个理想协议. 在其中假设有一个可信方完成所有的通信和计算，从而理想化地保证该属性的安全实现. 当证明某个实用协议安全实现该属性时，只需对于任何一个攻击实用协议的行为，模拟出一个攻击理想协议的行为. 这种利用模拟形式定义安全性的方法在密码学中很常用. 典型的代表有 Canetti 提出的通用可复合（Universally Composable，UC）框架和 Backes、Pfitzmann、Waidner 等人[57]提出的互动式模拟（Reactive Simulatability，RSIM）方案.

值得注意的是，实用协议是基于计算模型来刻画的. 用户在证明模拟关系时通常需要基于计算复杂性，采用归约的方法进行. 理想协议只是用于对安全属性的一个直观上安全的实现，这个实现的安全性要得到公认. 一般称为理想功能（ideal functionality）. 这种理想的协议或功能与形式化方法中把某些功能抽象为理想状态有着天壤之别. 在某些文献[58-59]中利用理想协议或系统进行模拟，对分析方法的计算可靠性进行探讨，对此需要仔细甄别.

3.1　通用可复合框架（UC）

3.1.1　基本框架

通用可复合（UC）框架[60-65]提供了一种密码协

议安全性分析的通用方法，即一个协议在独立运行情况下能实现其规范可以被推广为不管其周围网络中有什么样的活动，其规范照样可以被实现. 以下给出概略的 UC 框架.

UC 框架用以描述分布式系统下的程序，其基本计算单元为交互式图灵机(ITM). 一个分布式系统由多个 ITM 组成，顾名思义，这些 ITM 之间可通过一定的方式(如输入或发送消息，调用其它 ITM 等)进行交互. 交互的功能缘于 ITM 在图灵机的基础上所进行的扩展. 具体来说，ITM 定义了多个特定的带子，其中，有 3 个专用带子分别用于描述来自外部的 3 种不同消息：输入带(input)描述来自调用协议的信息、通信带(communication)描述通信链接中来自于其它主体的信息、子例程输出带(subroutine output)描述来自子例程协议的信息. 一个 ITM 可通过写其它 ITM 的相应带实现相互之间的交互. 一个 ITM 的运行称为 ITM 实例(简记为 ITI).

ITM 系统是一个二元组 (I,C)，其中，I 表示初始 ITM，即系统的执行是从该 ITM 开始的. $C:\{0,1\}^*\rightarrow\{0,1\}^*$ 为一个控制函数. ITM 系统的输出为初始 ITI 在系统执行结束后的输出，用 $\mathrm{OUT}_{I,C}$ 表示总体(ensemble)

$$\{\mathrm{OUT}_{I,C}(\eta,x)\}_{\eta\in\mathbb{N},x\in\{0,1\}^*}.$$

3.1.2 协议执行与协议模拟

由于密码协议本身是一种分布式系统，因此，可很容易地在 UC 框架中得到刻画. 在 UC 框架下，协议 p 的执行模型包括多个 ITM：环境 Z，敌手 $\mathcal{A}$ 以及多个协议主体. 环境 Z 生成主体的输入并读取其输出；敌手 $\mathcal{A}$ 接收协议主体发出的消息并向主体发送消息；主体执行协议中与自己相关的代码，完成相应的计算、发送或接收行为. 协议 p 的执行由系统 (Z,C) 来刻画，其中，环境 Z 被设置为初始 ITI，控制函数 C 刻画了 p 的单个实例与 Z 和 $\mathcal{A}$ 交互的模型. Z 控制着主体的输入并读取其输出，所有通过通信带进行的通信必须经过敌手 $\mathcal{A}$. 另外，p 的主体可以创建子例程，可以写子例程的输入带，还可通过其子例程输出带接收子例程 ITI 的输出. 协议执行的结果用以下总体表示：

$$\mathrm{EXEC}_{\pi,\mathcal{A},Z}=\{\mathrm{EXEC}_{\mathcal{A},Z}(\eta,x)\}_{\eta\in\mathbb{N},x\in\{0,1\}^*}.$$

设 p 和 p' 为协议，如果对所有的敌手 $\mathcal{A}$，存在一个敌手 $\mathcal{A}'$，使得对所有输出值在 $\{0,1\}$ 中的环境 Z，有 $\mathrm{EXEC}_{p',\mathcal{A}',Z}\approx\mathrm{EXEC}_{p,\mathcal{A},Z}$，则称协议 p 可 UC 模拟协议 p'.

密码协议的安全属性，即协议规范，通常由理想功能来刻画. 理想功能体现了协议的期望行为，被建模为一个与主体和敌手交互的 ITM. 用以实现理想功能 F 的程序被称为理想协议，记为 I_F. F 的理想进程被描述为一个运行理想协议 I_F 的进程.

设 p 为一个协议，F 为一个理想功能，如果 p 能够 UC 模拟 F 的理想协议 I_F，则称 p 可 UC 实现 F. UC 实现建立在 UC 模拟的基础上，并刻画了协议对其期望属性的满足.

3.1.3 计算可靠性与通用可复合性

由于 UC 框架本身是一个计算模型，因此，当一个协议在 DY 模型中被证明安全，而且它可被在 UC 框架中实现时，说明该协议在计算模型下也是安全的，因此具备计算可靠性. 文献[54]基于 UC 框架给出了符号化分析的计算可靠性.

与基于映射方法不同，由于 UC 框架中充分考虑了任意环境因素，因此，它还具备通用可复合性. 设协议 p UC 模拟协议 f，协议 r 将 f 当成其子例程，并可访问 f 的多个会话，则混合协议 $r^{p/f}$ 表示在 r 中将对 f 的调用替换为对 p 的调用，同时将来自 p 的子例程输出被当成来自 f 的子例程输出. 此处混合的意义在于，协议中部分为理想协议，部分为具体协议. 以下给出通用可复合定理：

设 p,f,r 为协议，如果 p 可 UC 模拟 f，则 $r^{p/f}$ 可 UC 模拟 r. 特别地，如果 r 可 UC 实现一个理想功能 F，那么 $r^{p/f}$ 也可 UC 实现理想功能 F.

3.2 互动式模拟(RSIM)

互动式模拟方案由 Backes，Pfitzmann 及 Waidner 等人[57,66]提出. 该方案采用互动式系统描述密码协议，系统的状态用数据库来记录. 针对 DY 模型可构造一种理想系统，针对计算模型可构造实际系统，然后证明存在一个模拟子，使得理想系统可模拟实际系统，从而保证计算可靠性. 以下对其作简要介绍.

在 RSIM 中，系统由多个可能的结构组成. 结构由一组相互连接的机器和一组自由端口的子集(称为指定端口)组成. 机器的模型是一个概率状态变迁机，类似于概率化的 I/O 自动机，机器之间可通过端口实现异步交互. 系统包括理想系统和实际系统. 在理想系统中，结构只包含一个可信机，而实际系统中的结构包含多个相互连接的机器. 理想系统的结构由一个函数 f 指定. 对一个标准的密码系统，函数 f 将实际结构映射为一个具有相同指定端口的可信机. 系统的运行状态由密码库来记录，这里的密码库类似于数据库. 根据密码库所记录的对象可将

密码库分为理想密码库和实际密码库.

在理想系统中，理想密码库对其用户提供抽象的密码操作，如对消息的加解密、签名及验证、生成新鲜值(Nonce). 所有这些命令有一个简单的确定的语义. 在互动式场景下，这种语义基于特定用户的状态. 系统的运行状态被保存在一个数据库中. 数据库的每个记录有一个类型以及指向其参数的指针. 它对应于 DY 模型中项的高级抽象，由该指针可找到整个项. 另外，每个记录还包含一些句柄，这些句柄指向知道该项的主体. 从而，通过数据库索引以及这些句柄可定位相应的密码对象. 大部分库具有导出操作，以将消息传给其用户. 理想密码库不允许欺骗. 例如，如果它收到一个加密的命令，那么它将只为密文构造一个抽象的数据库记录. 其它用户只有在拥有密文和密钥的句柄时才能请求解密. 类似地，如果一个用户发布一个消息签名的命令，理想系统将查询该用户是否拥有私钥，如果是，它就将该消息已经被签名的情况保存下来. 以后的验证只需要做数据库查询即可. 发送操作将使得其它主体知道一个记录，即将主体的句柄加入到该记录中. 对于安全的概率加密方案和概率签名方案而言，如果同一个消息被多次加密或签名，那么需要区分不同的版本(由记录的不同索引体现).

实际密码库所提供的命令与理想密码库相同，即诚实用户通过句柄对密码对象进行操作. 但在实际密码库中包含对密码的存储，各种命令非常类似于计算机编程中的标准 API 函数. 实际系统的数据库包含有实际的密钥、密文等. 当在不安全通道上发送消息时将会把实际的位串交给敌手. 敌手同样可以在非认证通道上插入任意消息.

模拟实质上意味着，无论在实际系统中对一定用户可发生什么，在理想系统中对该用户也一定有同样的情况发生. 对实际系统的每个结构 $struct_1$，每个用户 H，每个敌手 $\mathcal{A}_1$，存在一个针对相应实际结构 $struct_2$ 的敌手 $\mathcal{A}_2$，使得从 H 的视角看来，两个系统的运行格局是不可区分的. 为了证明理想系统可模拟实际系统，需要构造一个模拟子. 该模拟子工作在可信机与实际敌手之间. 当它收到来自可信机的输入时，需要将输入中的句柄转换为实际的数据，然后通过网络端口输出给实际敌手；当它收到来自实际敌手的输入时，需要将输入中的实际数据转换为相应的句柄，然后输出给可信机. 对模拟的证明将表明，任意实际敌手可完成的事情同样也可由理想系统中的敌手完成，否则底层的密码系统将被攻破.

互动式模拟也可实现通用可复合性，其复合原理与上节所述的 UC 框架类似，在此不再赘言.

4　已有分析方法的计算可靠性

以上基于映射和基于模拟的方法大多在提出的同时就考虑了其计算可靠性. 为了保持计算可靠性，它们大多注重了理论上的正确性，但或者距离实用还有一定的距离，或者分析过程较为复杂. 诚然，这些不足可以通过进一步的扩展来克服，但如果能够在已有的密码协议形式化分析方法的基础上保证其计算可靠性也不失为一种可行的方法，因为这些方法已经在实践中得到了广泛应用. 事实上，许多方法(如本文第 1.2.1 节中的形式化方法)在提出之初没有考虑计算可靠性，近年来，许多文献对这些方法进行了扩展，并证明了其计算可靠性. 这些方法包括协议逻辑、进程演算、串空间、安全信息流等. 以下对这些方法的计算可靠性研究做简要阐述. 需要说明的是，以下所说的各种方法的计算可靠性，并不是说这类方法都是计算可靠的，而是指对这类方法进行了某种改进，使得改进后的特定方法满足了计算可靠性.

4.1　密码协议逻辑分析的计算可靠性

基于逻辑的方法是密码协议形式化分析的一种重要方法[6-8,67-68]，但许多逻辑的可靠只是建立在逻辑语义的基础上，从而只能保证逻辑可靠性，要保证计算可靠性，必须建立其计算语义，并做出计算可靠性证明. 例如，CPCL(Computational PCL)[69] 就是一种建立在协议复合逻辑(Protocol Composition Logic，PCL)[70] 上的计算可靠的密码协议逻辑，以下对其作简要阐述.

与 PCL 相比，CPCL 给出了一种新的语义. 即用概率多项式时间的计算语义代替了 PCL 中的符号化语义. 正是这种新的语义为逻辑的计算可靠性提供了保障. 具体来说，该语义的定义借鉴了文献[51]的方法，即将协议的符号迹映射到其计算迹，从而将符号系统的语义建立在计算迹上，用迹的概率分布上的运算来解释公式. 由于逻辑本身的语法中并未显示涉及概率，因此，一个公式为真是指它能够以极大的概率成立. 这种语义是保证计算可靠性的基础.

在 CPCL 的语义模型中，协议的执行与 BR 模型中的执行方式类似，即将协议的执行看成是敌手与协议应答器之间的交互. 给定一个协议，一个敌手以及安全参数的值，可定义协议迹的集合. 每个迹关

联一个生成该行为序列的随机因子及其它随机因子(公式语义中相关算法的随机性).

协议的符号迹为执行串 $e\in ExecStrand$,它根据执行顺序记录了诚实主体以及非诚实主体的发送与接收行为.该串包含两个部分,$InitialState(I)$保存了初始化数据,其余部分为所有交换数据以及诚实主体内部行为的列表.

给定协议 Q,敌手$\mathcal{A}$,安全参数 η 以及由诚实主体和敌手使用的随机比特序列 $R\in\{0,1\}^{p(\eta)}$,则协议的运行可表示为五元组$\langle e,\lambda,O,K,R\rangle$,其中,$e$ 为符号执行串,$\lambda:Term(e)\rightarrow\{0,1\}^{p(\eta)}$将 e 中的符号项映射到比特串,O 为写在输出带上的整数对,K 为写在知识带上的消息的顺序列表,$p(x)$为 x 的多项式.

协议的计算迹被定义为一个七元组,即在协议运行的基础上加上两个元素 $R_T\in\{0,1\}^{p(\eta)}$ 和 σ:$\mathrm{FVar}(\varphi)\rightarrow\{0,1\}^{p(\eta)}$,其中,$R_T\in\{0,1\}^{p(\eta)}$用于测试不可区分性的随机比特序列,$\sigma$:$\mathrm{FVar}(\varphi)\rightarrow\{0,1\}^{p(\eta)}$为将公式 φ 中的自由变元映射到位串的一个替换.计算迹的集合 $T_Q(\mathcal{A},\eta)$表示如下集合:

$\{\langle e,\lambda,O,K,R,R_T,\sigma\mid R,R_T$ 的选择是均匀的$\rangle\}$.

公式的语义是由迹的集合来体现的.设 φ 为一个公式,该公式在迹 T 上的语义由 T 的一个子集 T'来刻画,即 T 中满足公式 φ 的迹的集合.

CPCL 的计算可靠性证明思路类似于其它逻辑可靠性的证明,即只要证明每个公理及推理规则的有效性,就可以通过对证明序列长度的归纳完成该定理的证明.在公理的有效性证明中,有些公理的有效性需要通过基于复杂性理论的归约方法完成.也就是说,需要表明,如果存在一个敌手可攻击该属性,那么一定存在一个敌手可攻破一定安全级别的密码方案.

无论是 PCL,还是 CPCL,事实上还都处于不断改进中,目前所能见到的相关资料并不一定是完善的.例如 CPCL 的计算可靠性事实上并没有给出一个全面的,严密的证明.对 PCL 和 CPCL 感兴趣的读者还可进一步参考文献[71-74]等.

4.2 基于进程演算分析方法的计算可靠性

进程演算[75-77]通常用以对并发系统进行建模,其中系统的属性常用等价关系来刻画.由于密码协议是一种典型的并发系统,因此,进程演算也被广泛地用于密码协议的安全性分析中[13-14].文献[78]在 Pi 演算[77]的基础上提出一种专门用于分析密码协议的 SPi 演算,用进程对密码协议进行建模.进而,文献[79]在 SPi 演算的基础上进行了扩充,提出一种应用 Pi 演算(以下简称 APi).与 SPi 演算相比,APi 中加入了等式理论.利用等式理论可非常灵活地对各种密码原语进行建模,这使得 APi 比 SPi 更为通用.

在基于 APi 的方法中,通常会用静态等价或观察等价的方式来描述协议的安全属性.静态等价与观察等价的主要区别在于前者不允许系统和观察者之间的持续交互,而后者允许.

4.2.1 静态等价

前面关于 AR 逻辑的扩展基本上是针对某种特定的密码原语,但实际上在应用中可能会使用各种各样的密码原语.另外,人们通常将密码原语看成一个黑盒子,但实际上具体的密码原语通常具备一定的代数属性[80],如交换律、结合律、同态等.因此,建立一种通用的、灵活的、可处理各种不同密码原语及其属性的计算可靠性分析方法是一件非常有意义的工作.在密码协议的形式化分析方法中,APi[79]便可满足这一要求.如果能够以 APi 为基础,证明形式化方法的计算可靠性,那么,将使得计算可靠性结论更为一般化.

在 APi 中,定义了进程的框架(frame).它是由 0 进程与主动替换通过并行与限制组合而成的进程.进程的框架实际上体现了该进程暴露给其环境的静态知识,即消息.静态等价就是建立在框架上的、由等式理论确定的等价关系.从这个意义上说,静态等价所反映的实际上还是消息之间的等价.需要注意的是,虽然基于静态等价来研究计算可靠性克服了 AR 逻辑中密码原语过于单一的缺点,但由于它所处理的仅限于暴露给敌手的静态消息,因此,它仍然针对的是被动攻击.

文献[81]首次对 APi 中静态等价的计算可靠性进行了研究.在利用静态等价的方法讨论形式化方法的计算可靠性时,首先要确定一组等式,以形式化地体现密码原语的特征或属性;然后给出各种符号的计算解释;最后证明在一定的安全性假设下,框架间的静态等价蕴涵其计算解释间的不可区分性.

文献[82]认为,由等式理论所定义的静态等价对于某些密码原语来说不够细化,从而在静态等价的基础上提出了形式化不可区分性的概念,进而对形式化不可区分性的计算可靠性进行了讨论.文献[83]认为,静态等价依赖于等式理论,所以,对不同的原语,选择合适的等式很重要.如果选择了不合适的等式,则可能对于不可区分性来说条件不够充分,或者是充分但不必要.该文中给出一组等式,然后证

明了由该等式理论所确定的静态等价关系蕴涵所给计算解释下的不可区分性，并将相应的形式化模型用于猜测攻击下的安全建模中.

4.2.2　观察等价

研究观察等价的计算可靠性基于以下原因：(1)静态等价仅针对被动攻击，而观察等价可体现进程中的动态行为，有利于处理主动攻击；(2)基于迹映射的方法可以处理主动攻击，但它对协议的安全性描述通常是用迹属性来描述的，有些属性难以用迹属性来描述，却很容易用等价性来描述. 例如，基于协议的一个运行迹可以方便地描述保密性，但难以描述协议的匿名性，而等价性在描述匿名性方面有其优势.

文献[84-85]对基于进程演算观察等价的计算可靠性进行了研究. 其主要思路是：首先给出用于分析密码协议的形式化进程演算，以及进程间的观察等价定义；其次，对进程做计算解释，由于进程不仅体现了协议中所出现的消息，而且体现了协议主体间的交互，因此对进程的计算解释需要考虑进程交互时的一些状态信息. 最后证明协议的观察等价蕴涵其计算解释的等价. 文献[85]与文献[84]的不同之处在于，在文献[84]所给出的形式化模型中没有显式的密码构造，而文献[85]中包含了显式的密码构造，并充分应用了等式理论的灵活性. 文献[85]认为，如果不显式地使用密码构造，则对密码协议的描述很可能使得一些安全隐患被隐藏. 例如，由于没有显式的密码构造，密钥自然是不会被发送或泄露的，而实际协议并不一定能保证这一点. 从而，文献[84]更适于设计协议而不适于对已有协议的验证.

由文献[79]知，观察等价实际上是一种标记互模拟(labeled bisimilarity)关系，与静态等价不同，它可以反映出进程的动态行为. 就密码协议的描述而言，它给了敌手做适应性选择的自由，同时也使得敌手进行主动攻击成为可能. 所以，观察等价的计算可靠性并不是对静态等价计算可靠性的平凡扩展.

4.3　串空间的计算可靠性

串空间[16]是又一种被广泛应用的密码协议形式化分析模型. 在串空间中，串是事件的序列，它可表示协议主体的执行过程. 串空间是由各种合法主体的串以及敌手的串组成的. 在串空间中，用丛来描述协议的交互过程，而用丛所满足的特性来刻画安全属性. 近年来，串空间的方法也在不断地发展[86-88]. 所以，对串空间的计算可靠性进行研究具有一定的价值.

在文献[89]中，Guttman 等人对串空间的两种处理方法进行了关联，即当一个协议在抽象模型(串空间模型)下满足其安全目标，那么，在其所定义的随机模型(Stochastic Model)下时，敌手成功攻击它的概率将低于一个合适的概率 ε(如 2^{-32}). 文献[33]认为，文献[89]中的模型不能称为计算模型，而只能称为概率模型，因为其安全定义只是采用了概率的描述，而并没有建立在计算复杂性的基础上. 但不可否认，文献[89]表明了串空间下的协议正确性蕴涵了其在该概率模型下的正确性. 另外，Herzog 在文献[90]中用串空间方法分析了 Diffie-Hellman 密钥交换协议，同时给出了如何在计算模型下定义并证明形式化模型下的安全性. 其主要思路是：首先以串空间的形式给出 Diffie-Hellman 假设下的安全性，然后将用于描述协议的丛转换为计算模型下的算法，最后基于底层的密码学假设证明仅当 CDH(Computation Diffie-Hellman)假设错误的情况下才会违反协议的安全性.

4.4　安全信息流的计算可靠性

信息流的概念是由 Denning[91]提出的，它主要用在程序安全性分析中. 在程序中，信息流主要是指信息从一个变量向另一个变量的流动. 如果一个变量的安全级高于另一个变量的安全级①，那么就禁止由前一个变量到后一个变量的信息流. 满足这一规定的信息流被称为安全信息流. 在形式化方法中，安全信息流通常用无干扰性(noninterference)刻画. 即公开输出一定不能包含任何关于秘密输入的信息. 如果将密码协议看作一段程序，将协议主体的输入看作程序的输入，将协议暴露给敌手的消息看作程序的公共输出，那么，如果能够证明不存在从秘密输入到公开输出的信息流，则可以证明该程序所代表的协议能够满足秘密消息的保密性.

Laud[92]首先对计算可靠的信息流方法进行了研究. 在文献[92]中，Laud 对计算安全信息流进行了定义，与无干扰性不同，此处的安全信息流是建立在计算复杂性基础上的. 它与无干扰性的不同在于，后者是以全能敌手的方式定义安全信息流的，很难对密码原语进行处理，因为密码原语通常只是计算安全的，而不是信息论安全的. 进一步地，文献[92]给出了对安全信息流的分析，并证明了如果能够从可观察到的输出中得到关于秘密输入的消息，那么

① 安全级是多级安全系统中的一个重要概念，例如安全级可以是绝密级、机密级、秘密级或公开级，其中绝密级高于机密级，机密级高于秘密级，秘密级高于公开级.

对安全信息流的分析也一定能够报告这一泄露.文献[92]的不足之处在于对程序的限制过多,如不能将密钥看作一般数据等.文献[93]重点讨论了安全信息流中包含加密操作的情况.其中,减少了对程序结构的限制,这样,密码也可以当作一般数据来对待,甚至可以包含密钥循环.文献[94]采用信息流的方法对对称加密下存在主动敌手时协议的保密性进行了分析.在以上文献中,计算无干扰性的定义在结构上以及所使用的域上非常复杂,在这一概念下,程序的语义也被概率化,从而与其相应的安全性描述也很复杂,而且分析的正确性不是很明显.2006 年,Askarov 等人[95]提出了定义在抽象模型上的密码掩流(Cryptographically Masked Flows)的概念.Laud[96]在密码掩流的基础上,给出编程语言的抽象语义与计算语义,并证明了抽象模型下的程序安全蕴涵计算模型下的程序安全.然后对该抽象模型进行了进一步简化(类似于 AR 逻辑中的形式化模型),并表明该模型下的安全性也具备计算可靠性.

5 计算方法的直接形式化

在上述方法中,通常是先给出一种形式化模型,然后再给出一种计算模型,最后表明形式化模型下的安全性蕴涵计算模型下的安全性,从而达到计算可靠的目的.近年来出现一种新的思路,即直接对计算方法进行形式化.换言之,它是直接从计算模型入手所构造的形式化模型.以这种方式构造的形式化方法与计算方法有着严格的对应性,因此其计算可靠性更为显然.由于概率是计算方法中的一个重要工具,因此,对计算方法的直接形式化中通常包含了对概率的描述.目前,这类方法主要包括基于逻辑的方法和基于进程演算的方法.

5.1 基于逻辑的方法

5.1.1 IK 逻辑

IK 逻辑是由 Impagliazzo 和 Kapron[97]于 2006 年提出的.与其它密码协议逻辑不同的是,IK 逻辑是以现代密码学的证明方法为出发点,并以逻辑的方法通过抽象而构造的逻辑系统,而在此之前的协议逻辑是以一些现有逻辑为出发点,并赋予一定的密码学特性而得到的逻辑系统,从而 IK 逻辑的计算可靠性更为直接,这也是本文将其归入直接方法的原因.建立这样一个系统的难点在于:安全定义的公式化,安全定义中概率的使用,涉及随机选择及概率分布方面的论证,以及对敌手计算能力的量化等.

IK 逻辑事实上包含了 2 个逻辑系统,第 1 个逻辑被称为 T 系统,更为通用,能力更强,但较复杂.第 2 个逻辑(下文提到 IK 逻辑时更多地是指第 2 个逻辑)相对专门化,提供了一个更简单的逻辑,即一个用于论证计算不可区分性的、简单的、可靠的逻辑.非严格地说,T 系统之于 IK 逻辑类似于图灵机之于现代计算机.

T 系统是一个支持概率及渐近的多项式函数推论的一阶逻辑.其语言基于算术语言[98],所不同的是,T 系统的语言是多类型的,即其变量建立于不同的域上.另外,T 系统中引入了 $\sharp(|x|=t)\varphi$ 形式的计数项,其中 $|x|$ 表示串 x 的长度.它表示在给定长度约束 $|x|=t$ 下满足公式 φ 的串 x 的个数.计数项可用以形式化地处理涉及概率的推导.例如,概率 $\Pr_{|x|=t}\varphi$ 可在 T 系统中表示为 $\dfrac{\sharp(|x|=t)\varphi(x)}{\sharp(|x|=t)(x=x)}$.T 系统的公理包括了含等词的多类型一阶逻辑公理,一组包含 36 条公理的公理集 BASIC,安全参数公理,多项式函数公理,计数公理以及归纳公理.

对于一个算术系统来说,通常以包含自然数集 $\mathbb{N}$ 的集合作为其模型的域①.假设 $\varphi(\boldsymbol{f},\boldsymbol{z},x)$ 为 T 系统的一个公式,其中 $\boldsymbol{f},\boldsymbol{z},x$ 为 φ 中出现的所有自由变元($\boldsymbol{f},\boldsymbol{z}$ 表示自由变元序列).在 T 系统的模型中,假设 $\boldsymbol{\alpha}$ 为任意定义在 $\mathbb{N}$ 上的多项式时间函数序列,$\boldsymbol{p}$ 为任意多项式序列,$\boldsymbol{s}$ 及 $t\in\mathbb{N}$.将 $\boldsymbol{f}$ 中的 f_i 对应地解释为 $\boldsymbol{\alpha}$ 中的 α_i,$\boldsymbol{z}$ 中的 z_i 对应地解释为 $\boldsymbol{s}$ 中的 s_i,x 解释为 t.假设对任意的 $\boldsymbol{\alpha},\boldsymbol{p}$,存在 $n_0\in\mathbb{N}$,使得当 $|t|>|n_0|$,$|s_i|=p_i(|x|)$ 时,φ 在 $\mathbb{N}$ 中成立,则称 φ 在 $\mathbb{N}$ 中渐近地成立.大致来说,T 系统的计算可靠性是指,如果在 T 系统内证明了某协议或密码构造的安全性是成立的,那么该协议或密码构造的安全性在计算模型下也是渐近地成立的.

根据 T 系统的计算可靠性,可用 T 系统对一些密码构造或密码协议进行形式化分析,并保证形式化分析的计算可靠性.特别地,一些计算模型下的概念,如计算不可区分性,可在 T 系统中得到形式化的刻画.以下以一个实例来说明.

设 p 为任意正的多项式,$g:\{0,1\}^*\to\{0,1\}^*$ 为一个多项式时间函数,如果对所有 $x\in\{0,1\}^*$ 有 $|g(x)|=|x|+p(|x|)$,且对所有多项式时间函数 $A:\{0,1\}^*\times\{0,1\}^*\to\{0,1\}$ 以及多项式 r 和 q,存

① 通常将以自然数为域的模型称为算术系统的标准模型.T 系统由于安全参数公理的存在,事实上使用了与标准模型不同构的非标准模型.

在 n_0 使得对所有 $n\geq n_0$ 有

$$\left(\Pr_{\substack{R\leftarrow\{0,1\}^{r(n)}\\X\leftarrow\{0,1\}^{n}}}[A(R,g(X))=1]-\Pr_{\substack{R\leftarrow\{0,1\}^{r(n)}\\Y\leftarrow\{0,1\}^{n+p(n)}}}[A(R,Y)=1]\right)\leq\frac{1}{q(n)}.$$

则称 g 是扩展因子，为 $p(n)$ 的伪随机生成器. 显然，以上定义中的伪随机性是以计算不可区分性描述的. 该不可区分性在 T 系统中可描述为

$$\forall A\,\forall z_1\forall z_2\left(\frac{\#\left(\begin{matrix}|r|=|z_2|\\|x|=n\end{matrix}\right)(A(r,g(x))=1)}{2^{|z_2|+n}}-\frac{\#\left(\begin{matrix}|r|=|z_2|\\|y|=n+p(n)\end{matrix}\right)(A(r,y)=1)}{2^{|z_2|+n+p(n)}}\right)\leq\frac{1}{|z_1|}.$$

可以验证，该式中的符号均是 T 系统语言可描述的. 该式在计算模型中渐近地成立正好对应于以上用以描述伪随机性的不可区分性.

由该例不难体会到，T 系统虽然可以形式化地刻画不可区分性，但这种刻画比较繁琐. 事实上，T 系统中的证明也将是一个异常繁琐的过程. 因此，有必要在其基础上构造一种新的、使用方便的、但同样可保持计算可靠性的逻辑，这就是 IK 逻辑. IK 逻辑主要针对计算不可区分性，其证明实质上是处理相等性，从而可消除对概率的显式处理，极大地简化了对不可区分性的推理. 以下对 IK 逻辑进行介绍.

IK 逻辑中的项表示 PPT 函数，更严格地说，项表示 PPT 函数总体. 为此，在 IK 逻辑中使用一组基本函数和典型的闭方案来定义多项式函数. 对于任意多项式 p，IK 逻辑还包含了如下形式的项：

$$\text{let } i\leftarrow\text{rand}(p(n))\text{ in } t,$$

$$\text{let } x\leftarrow\text{rs}(p(n))\text{ in } t.$$

直观地说，前者表示项 t，且在 t 中，变量 i 是从集合 $\{0,\cdots,p(n)-1\}$ 中随机均匀选择的. 后者同样表示项 t，但其中 x 是从长度为 $p(n)$ 的串中随机均匀选择的. “rand”的直观意思是随机索引（random index），“rs”的直观意思是随机串（random string）. 在不引起混淆的情况下，通常会采用缩写形式，如将“let $x\leftarrow$rs($p(n)$) in t”缩写为 $t[\text{rs}(p(n))/x]$. 另外，在以上两种形式的项中，通常将 $i\leftarrow$rand($p(n)$) 和 $x\leftarrow$rs($p(n)$) 称为随机绑定. 设 $b_1,\cdots,b_k$ 为一个绑定序列，有时会将“let b_1 in $\cdots$ let b_k in t”缩写为 $\text{let}\begin{pmatrix}b_1\\\cdots\\b_k\end{pmatrix}\text{ in } t$.

如果一个在“let b_1 in $\cdots$ let b_k in t”形式的项中，t 只是一个 T 系统中的基本项，则称该项是正规的.

如果 $v=$let b_1 in $\cdots$ let b_k in t 为正规项，x 为 t 中的一个自由变元，$u=$let c_1 in $\cdots$ let c_k in s 为任意自由项，则 $\{u/x\}$ 表示

let b_1 in$\cdots$let b_k in let c_1 in $\cdots$ let c_k in $[s/x]$.

如果正规项 t 中的所有变元都是受某一绑定中 $(b_1,\cdots,b_k)$ 约束的，则称该正规项是闭的.

设 s,t 为闭正规项，则引入公式 $s\approx t$ 以表示由 s 和 t 所代表的分布总体是计算不可区分的.

为了直接对计算不可区分进行推导，IK 逻辑采用以下规则模式刻画 $\approx$ 的属性：

REFL　$\dfrac{}{t\approx t}$

SYMM　$\dfrac{s\approx t}{t\approx s}$

TRAN　$\dfrac{t_1\approx t_2,\quad t_2\approx t_3}{t_1\approx t_3}$

UNIV　$\dfrac{T\vdash Q_1Q_2\cdots Q_k(s=t)}{\text{let } c_1 \text{ in } \cdots \text{ let } c_k \text{ in } s\approx \text{let } b_1 \text{ in } \cdots \text{ let } b_k \text{ in } t}$

SUB　$\dfrac{u\approx u'}{v\{u/x\}\approx v\{u'/x\}}$

H-IND　$\dfrac{\begin{matrix}\text{let } i\leftarrow\text{rand}(p(n)) \text{ in } u\approx\\ \text{let } i\leftarrow\text{rand}(p(n)) \text{ in } u[i+1/i]\end{matrix}}{u[0/i]\approx u[p(n)/i]}$

EDIT　$\dfrac{}{\text{let}\begin{pmatrix}\boldsymbol{i}\leftarrow\text{rand}(\boldsymbol{p}(n))\\x\leftarrow\text{rs}\left(\sum_{j=1}^{k}(p_j(n)-i_j)\right)\end{pmatrix}\text{ in } x\approx \text{let}\begin{pmatrix}\boldsymbol{i}\leftarrow\text{rand}(\boldsymbol{p}(n))\\x\leftarrow\text{rs}(p(n))\end{pmatrix}\text{ in }\bigcirc_{j=1}^{k}x_{j_{\{1\cdots p_j(n)-i_j\}}}}$

REFL 规则、SYMM 规则以及 TRAN 规则分别表明了 $\approx$ 的自反性、对称性和传递性.

UNIV 规则将在 T 系统中可证明的全称量化等式与计算不可区分关系 $\approx$ 关联起来. 在该规则中 s 和 t 为 T 系统中的项，$Q_1Q_2\cdots Q_k$ 为 $\forall|x|\leq p(n)$ 或 $\forall i<p(n)$ 形式的量词序列，b_i 为对应于 Q_i 的随机绑定. 所有 s 与 t 中的自由变元必须出现在某 Q_i 中.

SUB 规则表明在任何 PPT 环境中，不可区分项可相互替换，其中 v 为 IK 逻辑中任意仅包含自由变元 x 的项.

H-IND 规则有归纳规则的风格，但它还刻画了混合论证中所需的关于计算不可区分性的基本事实.

EDIT 规则可用于合并、划分或者缩短随机串，其结果与相应长度随机选择的串是不可区分的. 其

中$\bigcirc_{j=1}^{k}t_j$表示$t_1\circ\cdots\circ t_k$，即串的级联.

IK 逻辑可用于密码函数构造及密码协议的安全性证明.对 IK 逻辑的可靠性证明是以 T 系统为元逻辑而完成的，将 IK 逻辑中的不可区分性解释为 T 系统中的不可区分性，进而由 T 系统的计算可靠性得到 IK 逻辑的计算可靠性.

5.1.2 CIL 逻辑

IK 逻辑对不可区分性进行形式化，但在该逻辑中没有提供对应答器以及适应性敌手的支持，从而在分析一些标准密码方案时有一定的局限性.文献[99]提出另一种逻辑，即计算不可区分逻辑(Computational Indistinguishability Logic，CIL).CIL 逻辑以应答器为出发点，采用了可证明安全的论证模式，可在计算模型下对密码原语的安全性进行分析.它具备标准模型下的可靠性，同时支持 RO 模型和其它理想化模型.

对 CIL 逻辑的建立是以一个通用框架入手的.该通用框架用于建模敌手与含应答器的密码方案之间的交互.

应答器系统是一个可对敌手提供应答访问的有状态系统，不仅可建模加密系统，而且可建模签名系统.应答器系统$\mathcal{O}$由以下部分组成：

(1)应答器存储器集合$M_{\mathcal{O}}$及应答器集合$N_{\mathcal{O}}$.

(2)每个$o\in N_{\mathcal{O}}$有一个询问域$\mathrm{In}(o)$，一个应答域$\mathrm{Out}(o)$以及一个实现：

$$O_o:\ \mathrm{In}(o)\times M_{\mathcal{O}}\to D(\mathrm{Out}(o)\times M_{\mathcal{O}}).$$

(3)初始存储器$\bar{m}_{\mathcal{O}}\in M_{\mathcal{O}}$以及分别用于初始化和终止化的区分应答器$o_{\mathrm{I}}$和$o_{\mathrm{F}}$，且有$\mathrm{In}(o_{\mathrm{I}})=\mathrm{Out}(o_{\mathrm{F}})=1$，记$\mathrm{Res}=\mathrm{In}(o_{\mathrm{F}})$.

类似地，CIL 还给出了敌手的形式化定义.敌手与应答器系统的交互是由询问和接收应答来完成的.对于应答器系统$\mathcal{O}$来说，一次交换用一个三元组(o,q,a)表示，其中$o\in N_{\mathcal{O}}$，$q\in\mathrm{In}(o)$，$a\in\mathrm{Out}(o)$.交换的集合用Xch表示.当一个交换中的o为初始化应答器时，该交换被称为初始交换，记为Xch_{I}.相应地，当交换中的o为终止化应答器时，该交换被称为终止交换，记为Xch_{F}.Que表示询问集合，定义为$\{(o,q)\mid(o,q,a)\in Xch\}$.$Ans$表示应答集合，定义为$\{(o,a)\mid(o,q,a)\in Xch\}$.

安全属性是由敌手的状态抽象而来的，具体来说，是用迹来建模的.在基于 CIL 的构造与证明中，针对事件与迹使用了多种运算.如事件的合取、析取以及时态逻辑中的 F(Future)、G(Global)、U(Until)等算子.

基于归约的论证要求敌手可部分地模拟相关行为.有些情况下，给定一些值，敌手必须从他自身的角度检查谓词$\varphi\in Xch\times M_{\mathcal{O}}\times M_{\mathcal{O}}$是否成立.具体来说敌手判定该谓词是否成立的依据是到目前为止所进行的询问应答序列.如果这种测试是可能的，就称该谓词是可测试的.

CIL 中有 2 种常用语句：可忽略语句和不可区分语句.设E为一事件，$\varepsilon:((N_{\mathcal{O}}\to\mathbb{N})\times\mathbb{N})\to[0,1]$为一函数，在 CIL 中，用$\mathcal{O}:_{\varepsilon}E$表示事件$E$成立的概率是可忽略的.形式化地，称$\mathcal{O}:_{\varepsilon}E$是有效的，记为$\models\mathcal{O}:_{\varepsilon}E$，当且仅当对任意的$(k,t)$[①]有界敌手$\mathcal{A}$，有$\Pr(\mathcal{A}\mid\mathcal{O}:E)\leq\varepsilon(k,t)$.

设$\mathcal{O},\mathcal{O}'$为可兼容的应答器，则不可区分语句可用如下形式表示：$\mathcal{O}\sim_{\varepsilon}\mathcal{O}'$.形式化地，称$\mathcal{O}\sim_{\varepsilon}\mathcal{O}'$是有效的，记为$\models\mathcal{O}\sim_{\varepsilon}\mathcal{O}'$，当且仅当对任意$(k,t)$敌手$\mathcal{A}$有

$$|\Pr(\mathcal{A}\mid\mathcal{O}:R=\mathrm{true})-\Pr(\mathcal{A}\mid\mathcal{O}':R=\mathrm{true})|\leq\varepsilon(k,t),$$

其中，$\mathcal{A}\mid\mathcal{O}$表示$\mathcal{A}$与$\mathcal{O}$的复合，$R=\mathrm{true}$为$F_{\lambda(o,q,r).o=o_{\mathrm{F}}\wedge r=\mathrm{true}}$的缩写.

CIL 的语句对描述一些标准安全假设或安全定义提供了充分支持.如 DDH、单向置换、IND-CPA、IND-CCA 以及 EF-CMA 等.

CIL 的基本规则如下：

自反性 $\dfrac{}{\mathcal{O}\sim_0\mathcal{O}}$；

对称性 $\dfrac{\mathcal{O}\sim_{\varepsilon}\mathcal{O}'}{\mathcal{O}'\sim_{\varepsilon}\mathcal{O}}$；

传递性 $\dfrac{\mathcal{O}\sim_{\varepsilon}\mathcal{O}',\ \mathcal{O}'\sim_{\varepsilon'}\mathcal{O}''}{\mathcal{O}\sim_{\varepsilon+\varepsilon'}\mathcal{O}''}$；

UR $\dfrac{\mathcal{O}:_{\varepsilon_i}E_i(i\in I)\quad E\to\bigvee_{i\in I}E_i}{\mathcal{O}:_{\sum_{i\in I}\varepsilon_i}E}$，其中，$\sum_{i\in I}\varepsilon_i$被定义为$\lambda(k,t).\sum_{i\in I}\varepsilon_i(k,t)$；

POST-S $\dfrac{\mathcal{O}\{E\}}{\mathcal{O}:_0\neg E}$；

FAIL $\dfrac{}{\mathcal{O}:_{\varepsilon}F_{\varphi}}$，其中，

φ为$Xch\times M_{\mathcal{O}}\times M_{\mathcal{O}}$上的一个谓词，

$\varepsilon=\lambda(k,t).\sum_{o\in N_{\mathcal{O}}}k_o\varepsilon_o$，

ε_o被定义为

$$\max_{\substack{q\in Que,m\in M_{\mathcal{O}}\\ a\in Ans}}\ \sum_{\substack{m'\in M_{\mathcal{O}}\\ \varphi((o,q,a),m,m')}}Pr[O_o(q,m)=(a,m')].$$

前 3 个规则分别表明不可区分性的自反性、对

① 此处k用以约束敌手调用应答器的数目，t用以约束敌手运行时间.

称性和传递性.合并规则 UR 类似于将霍尔逻辑应用于概率程序时得到的规则.在 POST-S 规则中，$\mathcal{O}\{E\}$表示对于所有敌手$\mathcal{A}$，事件 E 在任意执行 $\mathcal{A}|\mathcal{O}$下都是成立的.从而，POST-S 规则是说，如果 $\mathcal{O}\{E\}$成立，那么事件 E 以 0 概率不成立.这其实是对非概率事件的概率表示.FAIL 规则根据应答器调用的数量以及单个应答器触发某事件的概率来计算该事件发生概率的上界.该规则的正确性可直接由合并规则得到.

以上是 CIL 的基本系统，在此基础上，CIL 还定义了上下文的概念，以及互模拟的概念.上下文可看作应答器与敌手间的媒介，它可与应答器复合形成一个新的应答器，与敌手复合形成的一新的敌手.互模拟的提出使得构造不同应答器间的等价关系成为可能.从而增加了 CIL 的建模能力.

CIL 逻辑是一种对不可区分性进行形式化的密码协议逻辑，它能够以较少的规则刻画密码证明中的许多论证模式.其最大特点是在逻辑中引入了应答器系统，这使得其对密码协议的安全性建模更为直接.另外，采用应答器的另一个好处是，它可以方便地反映适应性敌手的攻击策略.即敌手在攻击协议之前会尽可能通过应答器获得一定的信息，并根据所获得的信息不断调整攻击策略.因此，对应答器系统的支持使得 CIL 的表达力更强.这一点是 IK 逻辑所不具备的.CIL 的不足之处在于，虽然其逻辑规则比较简洁，但在应用这些规则时必须要构造合适的应答器系统、上下文及其复合，使得所构造的系统之间满足一定的关系(如互模拟关系等)，同时还需要计算相应的概率.

5.2　基于进程演算的方法

此处的进程演算与前面所提到的进程演算的不同之处在于概率的引入，从而有利于对密码学相关概念的刻画.

5.2.1　概率多项式时间演算

概率多项式时间演算(Probabilistic Polynomial-time Calculus，PPC)[100-102]是 Pi 演算、SPi 演算以及 APi 演算的变体.该方法提出的目的是以进程演算为基础，以更接近于计算方法的方式对密码协议进行分析.其主要特点是将概率引入进程演算，从而可以更自然地描述现代密码学中的相关概念.

在 PPC 中，对每一个包含变量 $x_1,\cdots,x_k$ 的项 T，存在一个有 $k+1$ 个输入的 PPT 图灵机 M_T 及多项式 $q_T(v_1,\cdots,v_{k+1})$，使得对于输入 $a_1,\cdots,a_k$ 及安全参数 η 有，$M_T(a_1,\cdots,a_k,\eta)$最多在 $q_T(|a_1|,\cdots,|a_k|,|\eta|)$时间内停机.而且对每一个 PPT 函数 f，存在一个项 T，使得 M_T 计算函数 f.这样就可以用概率图灵机来定义项 T 的涵义.如果 $M_T(a_1,\cdots,a_k,\eta)=a$ 的概率为 r，则称 T 以概率 r 被赋值为 a，记为 $T\xrightarrow{r}_e a$.

进程的语法如下：

$$
\begin{aligned}
P::=\ & 0 && \text{(终止进程)}\\
& \nu_{c_{p(|\eta|)}}.(P) && \text{(产生私有通道)}\\
& c_{p(|\eta|)}(x).(P) && \text{(输入)}\\
& c_{p(|\eta|)}\langle T\rangle.(P) && \text{(输出)}\\
& [T=T].(P) && \text{(匹配)}\\
& (P|P) && \text{(并行复合)}\\
& !_{\gamma(|\eta|)}.(P) && (\gamma(|\eta|)\text{次重复})
\end{aligned}
$$

其中，$p(|\eta|)$为安全参数长度的多项式，当它被关联到某信道 c 时，表示该信道的带宽参数.

没有自由变元的进程表达式被称为闭表达式.将进程中不在输入算子论域内的项或匹配称为暴露项或暴露匹配.概率函数 $outerEval$ 可对闭进程作外部赋值，使进程的暴露项化归为原子，暴露匹配被分解.如果 $outEval(P,P')=s$，则称进程 P 被以概率 s 外部赋值为 P'，记为 $P\xrightarrow{s}_0 P'$.

进程赋值是指在外部赋值的基础上对进程所进行的进一步约化.其大致过程如下：首先将进程中等待输入或已准备好输出的子进程组成进程的可调度集，如果调度集中对同一信道存在多个输入及多个输出，那么就从其中以一定的概率选出一对完成通信，这个过程被称为调度.进程 P 在调度 S 下以概率 r 在单步内被约化为进程 P' 记为 $P\xrightarrow{r}_S P'$.在调度 S 下进程 P 被赋值为 Q，是指根据调度 S 通过对进程 P 进行多步约化而得到进程 Q.记为 $P\Rightarrow_S P'$.将 P 赋值为 Q 的概率可根据加法原理和乘法原理得到.

观察被定义为在特定公共信道对特定自然数的一个测试，定义 Obs 为所有可能观察的集合，即二元组$\langle i,c_{p(|\eta|)}\rangle$的集合，其中，$i\in[0,2^{p(|\eta|)}-1]$为一个自然数，$c_{p(|\eta|)}$为一个公共信道.如果在对进程表达式 P 的赋值过程中，i 被在公共信道 $c_{p(|\eta|)}$ 上传递，则称$\langle i,c_{p(|\eta|)}\rangle\in Obs$ 可被观察到，记为 $P\rightsquigarrow\langle i,c_{p(|\eta|)}\rangle$.

上下文是指留空的进程，通常用 $C[]$表示.敌手上下文主要用来对敌手进行描述，在敌手上下文中，留空不能出现在 ν 算子的论域内.设 q 为多项式 $q(x)$的集合，且对所有的 y 有 $q(y)>0$，A 表示所有

敌手上下文簇，P，Q 为两个进程簇（以安全参数为索引），如果 P 和 Q 满足如下条件，则称 P 和 Q 观察等价，记为 $P\cong Q$：

$\forall q(x)\in q.\forall C[]\in A.\forall o\in Obs.\exists n_0.\forall \eta>n_0$：

$$|\Pr[C_\eta[P_\eta]\rightsquigarrow o]-\Pr[C_\eta[Q_\eta]\rightsquigarrow o]|\leq\frac{1}{q(\eta)}.$$

观察等价具有如下属性：

（1）$P\cong Q\Leftrightarrow\forall C[]\in A:C[P]\cong C[Q]$.

（2）如果 $P\cong Q$，则 $!_{\gamma(|\eta|)}P\cong !_{\gamma(|\eta|)}Q$.

（3）如果 $P\cong Q$，则

$$c_{1p(|\eta|)}\langle m_1\rangle|\cdots|c_{ip(|\eta|)}\langle m_i\rangle|P$$
$$\cong c_{1p(|\eta|)}\langle m_1\rangle|\cdots|c_{ip(|\eta|)}\langle m_i\rangle|Q.$$

（4）如果 $P\cong Q$，则 $[T_1=T_2]P\cong[T_1=T_2]Q$.

（5）$\forall\mathcal{A}\in A.(\mathcal{A}|P\cong\mathcal{A}|Q)\Leftrightarrow P\cong Q$.

（6）如果 $P_1\cong P_2$，且 $Q_1\cong Q_2$，则 $P_1|Q_1\cong P_2|Q_2$.

（7）设 f,g 为 PPT 可计算函数，且其值域均为 $X\subseteq\mathbb{N}$，二者均可导出 X 上相同的分布. 设 T_f，T_g 分别对应于函数 f,g 的项，即 M_{T_f}，M_{T_g} 分别计算函数 f,g，则存在一个多项式 q，使得 $c_{q(|\eta|)}\langle T_f\rangle\cong c_{q(|\eta|)}\langle T_g\rangle$.

概率多项式演算下的观察等价与密码学中的不可区分性是一致的. 在进行密码协议分析时，可用进程建模协议，用敌手上下文建模敌手，用观察等价建模安全属性，从而完成对密码协议或密码原语的分析.

5.2.2 基于实验序列的方法

实验序列（Sequences of Games）的方法通常用在密码学中进行安全性证明. Shoup 在文献[103]中对其进行了整理与总结. 密码学原语的安全性通常由敌手和挑战者之间的攻击实验来定义，这种定义通常与某特定事件 S 关联起来. 安全意味着，对于一个多项式时间敌手来说，事件 S 发生的概率接近“目标概率”，目标概率通常为 0 或 1/2，或者是该敌手与不同挑战者在不同实验中进行交互时某事件发生的概率. 在实验序列法中，采用了一个实验序列：$G_0,G_1,\cdots,G_n$，其中 G_0 为最初敌手对密码原语的攻击实验. 设 S_i 为 G_i 中的事件，S_0 为 S，则安全性证明需要表明，对于 $i=0,\cdots,n-1$ 来说，$\Pr[S_i]$ 与 $\Pr[S_{i+1}]$ 非常接近，而且 $\Pr[S_n]$ 等于或非常接近于“目标概率”. 严格来说，此处的非常接近是指两个概率之差为安全参数的可忽略函数. 由于此处的 n 为常数，所以，$\Pr[S]$ 非常接近于“目标概率”，从而完成安全性证明.

Blanchet 等人[104-107]采用进程演算描述实验，对实验序列法进行了形式化，并开发了一种计算可靠的密码协议自动验证工具 CryptoVerif. 该方法不是依赖于 DY 模型，而是直接依赖于计算模型. 以下对其主要思想进行简单介绍.

在 Blanchet 等人的方法中，沿用了实验序列法的基本思路，但为了自动化这一过程，对实验的描述采用了一种基于 Pi 演算的进程演算语言. 与上一节中概率多项式时间演算类似，该演算具有概率语义，其中，所有的进程运行在多项式时间内. 密码协议或密码原语的安全属性也由观察等价刻画，即如果敌手只能以可忽略概率区分进程 Q 和 Q'，则称 Q 与 Q' 观察等价，记为 $Q\approx Q'$. 所不同的是，在本方法中，进程执行过程中所有变量的值都被保存在数组中，如，$x[i]$ 为第 i 个进程拷贝中变量 X 的值. 这样做的好处是，消除了那些虽然是平凡可靠的，但很难自动化的进程语法转换.

进程转换在本方法中起到了核心的作用. 其目的是将用于描述协议的最初进程通过一步步的转换，化为另外的进程，直到在所得到的进程中期望安全属性很明显为止. 这种转换是一种重写规则，它将一个进程重写为另一个等价的、或者在一定假设下几乎等价的进程. 进程转换通常有 2 种转换方式，一种为语法转换，一种为运用原语安全定义的转换（以下简称密码学转换）.

语法转换的目的主要是为运用原语安全定义作准备，或者对使用过密码学转换后的进程作进一步简化. 常用的转换包括对赋值语句的应用，例如，某语句对进程 P 中的某变量进行赋值，则转换后可消除该赋值部分，而直接用所赋的值替换 P 中该变量. 另外，还包括对变量的换名，以及一些简化规则，这些简化规则可能来自于进程演算本身，也可能来自于用户定义的规则.

密码学转换是最重要的转换，这类转换可描述如下：用观察等价的方式描述密码原语的安全性定义，得到等价式 $L\approx R$. 其中进程 L 和 R 分别代表一定的函数，其输入为该函数的输入，其输出为该函数的输出. 如果一个进程 Q 调用了 L 的函数，即 Q 中包含和 L 的函数相同的计算，则将其中对 L 的函数调用转换为对 R 的函数调用，从而得到转换后的进程 Q'. 当需要考虑各种原语的安全性时，只需要简单地将描述其安全性的观察等价式 $L\approx R$ 加入到演算系统中即可. 采用这种技术，可方便地描述各种密码原语安全性，如对称与非对称加密、签名、消息认证码以及哈希函数等. 这种转换所带来的灵活性从

某种程度上说，类似于等式理论为 API[79] 所带来的灵活性.

在对密码协议进行证明时，首先用进程描述协议；其次用某种进程等价式描述协议的安全性；接着，如果协议的安全性是以某原语的安全假设为基础的，则将其描述为 $L \approx R$ 的形式以便在密码学转换时使用；最后，对协议进程进行各种转换，直至协议的安全属性可以很显然地判断.

以上方法由一种自动验证工具 CryptoVerif 实现. 该验证器使用了如下证明策略组织各种转换以证明协议的安全性. 在证明之初和每次成功应用密码原语定义转换后，均对进程做简化转换，并检查转换后的进程是否已经证明了期望的安全属性. 如果已经证明，则成功退出. 在执行进程转换时，验证器采用了一定的建议策略. 具体来说，验证器会按某种顺序依次执行可用的密码学转换，如果一个转换失败，则返回某种语法转换，以使得所进行的密码学转换可以继续. 这种返回的语法转换就是一种建议. 然后验证器执行所建议的语法转换，如果失败，同样会返回一定的转换建议. 当语法转换成功时，再接着进行先前的密码学转换. 这时，也可能成功，也可能失败，如果失败了，又会返回一定的转换建议. 这种建议策略使得对协议的证明常常可以全自动完成，万一无法自动完成，还可以使用交互模式，这时，用户可手动指定一定的转换规则.

6　总结与展望

计算可靠的密码协议形式化分析是密码协议分析的一个重要的研究方向. 十年来，已经发展了许多可保证计算可靠性的协议分析方法. 本文对这些方法进行了综述，将其分为基于映射的方法，基于模拟的方法，以及基于现有形式化方法和直接对计算方法进行形式化的方法. 这些方法或者从形式化方法入手，保持其计算可靠性，或者从计算方法入手，对其做形式化抽象，其主要目的就是在形式化方法与计算方法之间建立关联，使得对密码协议的分析既保持了形式化方法的简单性，又能保证计算方法的严密性. 表 1 对这几类方法进行了归纳，分别指出了其特点与适用范围.

表 1　各种方法的特点与适用范围

方法	特点	适用范围
基于映射的方法	从协议自身的角度，通过建立映射函数，将形式化模型下的消息或行为映射到计算模型下，从而保证协议分析的计算可靠性. 其中消息映射的方法仅用协议交互中传输的消息来建模协议，而迹映射方法用协议的执行迹来建模协议.	适用于对简单协议的安全性分析，即不考虑协议的复合性. 其中消息映射的方法适用于进行被动攻击下的安全性分析. 而迹映射的方法适用于主动攻击下的安全性分析.
基于模拟的方法	基于理想功能对安全属性进行建模，然后，从协议环境的角度，通过判断具体协议与理想协议在执行中能否被环境所区分，在抽象协议和具体协议之间建立模拟关系，从而保证协议分析的计算可靠性.	由于这类方法在分析安全性的同时可保证协议的通用可复合性，因此，适用于对复合协议的安全性进行分析. 另外，从环境的角度来看，该方法适用于主动攻击下的安全性分析.
已有分析方法的计算可靠性	类似的方法已经存在，且得到广泛应用，但其计算可靠性未得到保证，本类方法在已有方法的基础上进行扩展，并建立其计算模型，以保证这类方法的计算可靠性.	主要适用于在已有形式化分析下已经进行了分析，但分析结果的计算可靠性尚无保证的情况. 具体到每种方法要视其安全性刻画的方便性及用户喜好而定.
计算方法的直接形式化	该类方法直接从计算模型入手，通过形式化抽象，构造密码协议的形式化模型. 以这种方式构造的形式化方法与计算方法有着严格的对应性，因此其计算可靠性更为显然. 另外，这类方法在其形式化中通常包含了对概率的描述.	除了可对安全协议进行分析外，还适用于对一些更低级的密码学构造进行分析，如加密方案，签名方案等. 另外，在使用这类方法时，用户需要具备一定的密码学基础.

从密码协议分析的发展趋势来看，当前和今后一段时间计算可靠的密码协议分析研究将呈现以下几个特点：

(1) 实现可靠的密码协议的自动分析工具是该领域的一个长期目标. 在人类文明的发展过程中，工具始终是文明发展的标志. 农业革命中手工工具的出现延伸了人的双手，工业革命中机械化工具的发展延伸了人的体力，而信息革命中自动化工具的发展所延伸的将是人类的思维. 从这个意义上说，自动化工具是密码协议分析的重要目标. 本研究领域在这方面已经有了一个很好的开端，但还需要进一步增加验证协议的范围，并尽可能减少人工辅助. 这里要指出的是，很多协议是不可判定的，所以难以实现全部的自动化. 即使某些范围内的自动化工具也是非常有帮助的.

(2) 通用可复合性或者特定性类型协议的可复合性研究，作为本领域研究热点的状况仍将继续. 随着全球信息化程度的不断深化，密码协议分析面临

越来越严峻的形势.例如,密码协议在规模上不断复杂化的趋势为协议的安全性分析提出了很大的挑战.这种协议通常由许多较小的协议复合而成,但其安全性却不能简单地由各个较小协议的安全性自然地得到.原因在于敌手可将不同协议运行或同一协议不同运行中的消息相互交叉使用,从而对协议的安全性构成威胁.这种威胁使得通用可复合性或特定性可复合性成为当前和今后的一个研究热点.当前在可复合性研究方面存在的主要问题是分析模型过于复杂,需要进一步简化.

(3)对实用协议计算可靠的安全性分析将受到进一步重视.当前在计算可靠性研究方面,大部分方法还主要体现在理论形成阶段,对具体协议的分析结果还比较缺乏.主要原因在于,当前的研究主要注重结果的正确性,在实际应用中对具体协议的建模能力有限.例如,AR 逻辑虽然在理论上取得了成功,但实际分析中使用不广泛.UC 框架在理论上可以分析由简单协议复合而成的复杂协议,但在实际应用中,由于其分析过程过于复杂,对实用协议的分析并不多见.

(4)直接对计算方法进行形式化的方法近年刚刚兴起,还需要进一步发展与完善.本文在直接对计算方法进行形式化的方法方面介绍了基于进程演算和基于逻辑的方法,但总体来说,这类方法还处于起步阶段,需要进一步丰富.例如,在 CIL 逻辑中,应答器及上下文的构造等工作需要提前构造,从而增加了利用 CIL 分析密码协议的复杂性及对人工参与的依赖性,也增加了对其进行自动化的难度.

(5)信息业务的多样化呼唤更多的安全属性分析.当前,密码协议计算可靠的形式化分析主要集中在保密性与认证性上.事实上,协议的安全性远远不止这些.例如在电子商务中,公平性是一种非常重要的安全属性,但目前对公平性的分析手段极为有限,形式化的分析方法更为缺乏.在电子投票协议中,匿名性是一种很典型的安全属性,但对匿名性的分析也很少受到关注.

(6)代码级的密码协议分析将成为该领域的又一热门方向.当前,很少有代码级的密码协议分析,但从发展的角度来看,代码级的分析将是一个必然方向.任何理论问题要发挥现实作用,必须落实在实现上,密码协议也不例外.从应用的角度来说,密码协议最终的表现形式是程序代码.一种经过理论分析被证明安全的协议,很可能在实现过程中引入新的安全问题.要防止这类安全问题的出现,必须在代码级研究协议的安全性.在这一点上,软件正确性分析的历史就是一个很好的例证.不仅如此,软件工程以及软件形式化验证的一些方法可以为代码级的密码协议分析提供很好的借鉴.在代码级对密码协议作计算可靠的验证方面,文献[108]是一个值得注意的动向.

参 考 文 献

[1] Xue Rui, Lei Xin-Feng. Present status and trends of researches on analyses of security protocols. Bulletin of Chinese Academy of Sciences, 2011, 26(3): 287-296(in Chinese)
(薛锐,雷新锋. 信息安全保障的灵魂——安全协议分析研究现状与发展趋势. 中国科学院院刊, 2011, 26(3): 287-296)

[2] Xue Rui, Feng Deng-Guo. The approaches and technologies for formal verification of security protocols. Chinese Journal of Computers, 2006, 29(1): 1-20(in Chinese)
(薛锐,冯登国. 安全协议的形式化分析技术与方法. 计算机学报, 2006, 29(1): 1-20)

[3] Needham R M, Schroeder M D. Using encryption for authentication in large networks of computers. Communications of the ACM, 1978, 21(12): 993-999

[4] Dolev D, Yao A C. On the security of public key protocols//Proceedings of the 22nd Symposium on Foundations of Computer Science. Oakland, USA, 1981: 350-357

[5] Burrows M, Abadi M, Needham R. A Logic of Authentication. Palo Alto, USA: Digital Equipment Corp. (DEC), Systems Research Center: 39, 1989

[6] Gong L, Needham R, Yahalom R. Reasoning about belief in cryptographic protocols//Proceedings of the 1990 IEEE Symposium on Research in Security and Privacy. Oakland, USA, 1990: 234-248

[7] Abadi M, Tuttle M. A semantics for a logic of authentication//Proceedings of the 10th Annual ACM Symposium on Principles of Distributed Computing. Montreal, Canada, 1991: 201-216

[8] Syverson P, van Oorschot P C. On unifying some cryptographic protocol logics//Proceedings of the IEEE Symposium on Research in Security and Privacy. Oakland, USA, 1994: 14-28

[9] Kailar R. Accountability in electronic commerce protocols. IEEE Transactions on Software Engineering, 1996, 22(5): 313-328

[10] Syverson P. Adding time to a logic of authentication//Proceedings of the 1st ACM Conference on Computer and Communications Security. Fairfax, USA, 1993: 97-101

[11] Syverson P F. The use of logic in the analysis of cryptographic protocols//Proceedings of the 12th IEEE Symposium on Security and Privacy. Oakland, USA, 1991: 156-170

[12] Lei Xin-Feng, Liu Jin, Xiao Jun-Mo. Time-dependent cryptographic protocols logic and its formal semantics. Journal of

Software, 2011, 22(3): 534-557(in Chinese)
(雷新锋, 刘军, 肖军模. 时间相关密码协议逻辑及其形式化语义. 软件学报, 2011, 22(3): 534-557)

[13] Lowe G. Breaking and fixing the needham schroeder public-key protocol using FDR//Proceedings of the TACAS. Berlin, German, 1996: 147-166

[14] Abadi M, Gordon A D. A calculus for cryptographic protocols: The spi calculus. Information and Computation, 1999, 148(1): 1-70

[15] Paulson L C. The inductive approach to verifying cryptographic protocols. Journal of Computer Security, 1998, 6: 85-128

[16] Fabrega F J T, Herzog J C, Guttman J D. Strand spaces: Why is a security protocol correct//Proceedings of the 1998 Conference on Security and Privacy (S&P-98). New York, USA, 1998: 160-171

[17] Xue Rui. Formal analysis of security protocols and its development status//Chinese Association for Cryptologic Research. Development Report of Chinese Cryptography'2008. Beijing: Publishing House of Electronics Industry, 2009: 103-138(in Chinese)
(薛锐. 安全协议的形式化分析方法及其发展现状//中国密码学会. 中国密码学发展报告 2008. 北京: 电子工业出版社, 2009: 103-138)

[18] Blum M, Micali S. How to generate cryptographically strong sequences of pseudo random bits//Proceedings of the 23rd Annual Symposium on Foundations of Computer Science. Los Alamitos, USA, 1982: 112-117

[19] Yao A C. Theory and application of trapdoor functions//Proceedings of the 23rd IEEE Symposium on Foundations of Computer Science. USA, 1982: 80-91

[20] Goldwasser S, Micali S. Probabilistic encryption. Journal of Computer and System Sciences, 1984, 28(2): 270-299

[21] Bellare M, Rogaway P. Entity authentication and key distribution//Proceedings of the CRYPTO' 93 Advances in Cryptology. Santa Barbara, USA, 1994: 232-249

[22] Canetti R, Krawczyk H. Analysis of key-exchange protocols and their use for building secure channels//Proceedings of the EUROCRYPT' 2001. Innsbruck (Tyrol), Austria, 2001: 453-474

[23] LaMacchia B A, Lauter K, Mityagin A. Stronger security of authenticated key exchange//Proceedings of the 1st International Conference on Provable Security. Wollongong, Australia, 2007: 1-16

[24] Bellare M, Canetti R, Krawczyk H. A modular approach to the design and analysis of authentication and key exchange protocols//Proceedings of the 30th Annual ACM Symposium on Theory of Computing. New York, USA, 1998: 419-428

[25] Shoup V. On formal models for secure key exchange. USA: IBM, IBM Research: RZ 3120, 1999

[26] Abadi M, Rogaway P. Reconciling two views of cryptography //Proceedings of the International Conference IFIP on Theoretical Computer Science. London, UK, 2000: 3-22

[27] Abadi M, Rogaway P. Reconciling two views of cryptography. Journal of Cryptology, 2002, 15(2): 103-127

[28] Micciancio D, Warinschi B. Completeness theorems for the Abadi-Rogaway language of encrypted expressions. Journal of Computer Security, 2004, 12(1): 99-130

[29] Horvitz D O. Expressiveness of definitions and efficiency of constructions in computational cryptography[Ph. D. dissertation]. Maryland: University of Maryland, 2007

[30] Bana G. Soundness and completeness of formal logics of symmetric encryption[Ph. D. dissertation]. Pennsylvania: University of Pennsylvania, 2004

[31] Adão P, Bana G, Scedrov A. Computational and information-theoretic soundness and completeness of formal encryption//Proceedings of the 18th IEEE Computer Security Foundations Workshop. Aix-en-Provence, France, 2005: 170-184

[32] Camenisch J, Lysyanskaya A. An efficient system for non-transferable anonymous credentials with optional anonymity revocation//Advances in Cryptology—EUROCRYPT'2001. Innsbruck (Tyrol), Austria, 2001: 93-117

[33] Laud P. Encryption cycles and two views of cryptography//Proceedings of the 7th Nordic Workshop on Secure IT Systems. Karlstad, Sweden, 2002: 85-100

[34] Adão P, Bana G, Herzog J, Scedrov A. Soundness of formal encryption in the presence of key-cycles//Proceedings of the 10th European Symposium on Research in Computer Security. Milan, Italy, 2005: 374-396

[35] Adão P, Bana G, Herzog J, Scedrov A. Soundness and completeness of formal encryption: The cases of key cycles and partial information leakage. Journal of Computer Security, 2009, 17(5): 737-797

[36] Micciancio D. Computational soundness, co-induction, and encryption cycles//Advances in Cryptology—EUROCRYPT 2010. Riviera, French, 2010: 362-380

[37] Black J, Rogaway P, Shrimpton T. Encryption-Scheme security in the presence of key-dependent messages//Nyberg K, Heys H M eds. Selected Areas in Cryptography. Lecture Notes in Computer Science 2595. Berlin: Springer, 2002: 62-75

[38] Hofheinz D, Unruh D. Towards key-dependent message security in the standard model//Advances in Cryptology—EUROCRYPT 2008. Istanbul, Turkey, 2008: 108-126

[39] Boneh D, Halevi S, Hamburg M, Ostrovsky R. Circular-secure encryption from decision diffie-hellman//Crypto W D ed. Lecture Notes in Computer Science 5157. Berlin: Springer, 2008: 108-125

[40] Haitner I, Holenstein T. On the (Im) possibility of key dependent encryption//Tcc R O ed. Lecture Notes in Computer Science 5444. Berlin: Springer, 2009: 202-219

[41] Micciancio D. Pseudo-randomness and partial information in symbolic security analysis. IACR: Cryptology ePrint Archive, Report 249, 2009

[42] Abadi M, Warinschi B. Security analysis of cryptographically controlled access to XML documents. Journal of the ACM, 2008, 55(2): 1-29

[43] Miklau G, Suciu D. Controlling Access to published data using cryptography//Proceedings of the 29th International Conference on Very Large Data Bases. Berlin, Germany. Los Altos: Morgan Kaufmann Publishers, 2003: 898-909

[44] Lei X, Xue R, Yu T. Computational soundness about formal encryption in the presence of secret shares and key cycles//Proceedings of the 13th International Conference on Information and Communication Security (ICICS 2011). Beijing, China, 2011: 29-41

[45] Abadi M, Jürjens J. Formal eavesdropping and its computational interpretation//Kobayashi N, Pierce B eds. Theoretical Aspects of Computer Software. Lecture Notes in Computer Science 2215. Berlin: Springer, 2001: 82-94

[46] Micciancio D, Panjwani S. Adaptive security of symbolic encryption//Kilian J ed. Lecture Notes in Computer Science 3378. Berlin: Springer, 2005: 169-187

[47] Herzog J. A computational interpretation of Dolev-Yao adversaries. Theoretical Computer Science, 2005, 340(1): 57-81

[48] Laud P, Corin R. Sound computational interpretation of formal encryption with composed keys//Lim J I, Lee D H eds. Lecture Notes in Computer Science 2971. Berlin: Springer, 2003: 55-66

[49] Garcia F D, van Rossum P. Sound computational interpretation of symbolic hashes in the standard model//Yoshiura H, Sakurai K, et al, eds. Lecture Notes in Computer Science 4266. Berlin: Springer, 2006: 33-47

[50] Garcia F D, van Rossum P. Sound and complete computational interpretation of symbolic hashes in the standard model. Theoretical Computer Science, 2008, 394(1-2): 112-133

[51] Micciancio D, Warinschi B. Soundness of formal encryption in the presence of active adversaries//Naor M ed. Lecture Notes in Computer Science 2951. Berlin: Springer, 2004: 133-151

[52] Cortier V, Warinschi B. Computationally sound, automated proofs for security protocols//Proceedings of the European Symposium on Programming. Edinburgh, UK, 2005: 157-171

[53] Janvier R, Lakhnech Y, Mazaré L. Completing the picture: Soundness of formal encryption in the presence of active adversaries//Sagiv S ed. Lecture Notes in Computer Science 3444. Berlin: Springer, 2005: 172-185

[54] Canetti R, Herzog J. Universally composable symbolic security analysis. Journal of Cryptology: The Journal of the International Association for Cryptologic Research, 2011, 24(1): 83-147

[55] Cortier V, Kremer S, Küsters R, Warinschi B. Computationally sound symbolic secrecy in the presence of hash functions//Proceedings of the 26th Conference on Foundations of Software Technology and Theoretical Computer Science. Kolkata, India, 2006: 176-187

[56] Janvier R, Lakhnech Y, Mazaré L. Computational soundness of symbolic analysis for protocols using hash functions//Proceedings of the Electronic Notes in Theoretical Computer Science. 2007: 121-139

[57] Backes M, Pfitzmann B, Waidner M. A general composition theorem for secure reactive systems//Lecture Notes in Computer Science 2951. Berlin: Springer, 2004: 336-354

[58] Backes M, Dürmuth M, Küsters R. On simulatability soundness and mapping soundness of symbolic cryptography//Proceedings of the 27th Conference on Foundations of Software Technology and Theoretical Computer Science. New Delhi, India, 2007: 108-120

[59] Mazaré L, Warinschi B. Separating trace mapping and reactive simulatability soundness: The case of adaptive corruption//Degano P, Viganò L eds. Lecture Notes in Computer Science 5511. Berlin: Springer, 2009: 193-210

[60] Canetti R. Universally composable security: A new paradigm for cryptographic protocols//Proceedings of the 42nd IEEE Symposium on Foundations of Computers Science. Oakland, USA, 2001: 136-145

[61] Canetti R, Rabin T. Universal composition with joint state//Boneh D ed. Advances in Cryptology—CRYPTO'2003. Berlin: Springer-Verlag, 2003: 265-281

[62] Canetti R. Universally composable signature, certification, and authentication//Proceedings of the 17th IEEE Computer Security Foundations Workshop. Pacific Grove, USA, 2004: 219-233

[63] Canetti R, Herzog J. Universally composable symbolic analysis of cryptographic protocols: The case of encryption-based mutual authentication and key exchange. IACR: Cryptology ePrint Archive, Report 334, 2004

[64] Canetti R, Herzog J. Universally composable symbolic analysis of mutual authentication and key-exchange protocols//Halevi S, Rabin T eds. TCC 2006. Lecture Notes in Computer Science 3876. Berlin: Springer, 2006: 380-403

[65] Canetti R. Composable formal security analysis: Juggling soundness, simplicity and efficiency//Aceto L, Damgård I, Goldberg LA, et al, eds. Lecture Notes in Computer Science 5126. Berlin: Springer, 2008: 1-13

[66] Backes M, Pfitzmann B, Waidner M. A universally composable cryptographic library. IACE: Cryptology ePrint Archive, Report 015, 2003

[67] Burrows M, Abadi M, Needham R. A logic of authentication. ACM Transactions on Computer Systems, 1990, 8(1): 18-36

[68] Lei X, Xue R, Yu T. A timed logic for modeling and reasoning about security protocols. ICAR: Cryptology ePrint Archive, Report 645, 2010

[69] Datta A, Derek A, Mitchell J C, Shmatikov V, Turuani M. Probabilistic polynomial-time semantics for a protocol security logic//Caires L, Italiano G F, Monteiro L, et al, eds. Lecture Notes in Computer Science 3580. Berlin: Springer, 2005: 16-29

[70] Datta A, Derek A, Mitchell J C, Pavlovic D. A derivation system for security protocols and its logical formalization//Proceedings of the 16th IEEE Computer Security Foundations Workshop. Pacific Grove, USA, 2003: 109-125

[71] Datta A, Derek A, Mitchell J C, Roy A. Protocol composition logic(PCL)//Proceedings of the Electronic Notes in Theoretical Computer Science. Berlin, German, 2007, 172: 311-358

[72] Datta A, Derek A, Mitchell J, Warinschi B. Computationally sound compositional logic for key exchange protocols// Proceedings of the 19th IEEE Computer Security Foundations Workshop. Venice, Italy, 2006: 321-334

[73] Roy A, Datta A, Derek A, Mitchell J C. Inductive proofs of computational secrecy//Proceedings of the 12th European Symposium on Research in Computer Security (ESORICS). Dresden, Germany, 2007: 219-234

[74] Roy A, Datta A, Derek A, Mitchell J C. Inductive trace properties for computational security. Journal of Computer Security, 2010, 18(6): 1035-1073

[75] Milner R. A calculus of communicating systems. Lecture Notes in Computer Science 92. Berlin: Springer-Verlag, 1980

[76] Hoare C A R. Communicating Sequential Processes. Upper Saddle River, New Jersey: Prentice Hall, 1985

[77] Milner R, Parrow J, Walker D. A Calculus of Mobile Processes I&II. Information and Computation, 1992, 100(1): 1-77

[78] Abadi M, Gordon A D. A calculus for cryptographic protocols: The spi calculus. Cambridge, UK: University of Cambridge Computer Laboratory, 414, 1997

[79] Abadi M, Fournet C. Mobile values, new names, and secure communication//Proceedings of the 28th ACM Sigplan-Sigact Symposium on Principles of Programming Languages. London, UK, 2001: 104-115

[80] Cortier V, Delaune S, Lafourcade P. A survey of algebraic properties used in cryptographic protocols. Journal of Computer Security, 2006, 14(1): 1-43

[81] Baudet M, Cortier V, Kremer S. Computationally sound implementations of equational theories against passive adversaries//Caires L, Italiano G F, Monteiro L, et al, eds. Lecture Notes in Computer Science 3580. Berlin: Springer, 2005: 652-663

[82] Bana G, Mohassel P, Stegers T. Computational soundness of formal indistinguishability and static equivalence//Okada M, Satoh I eds. Lecture Notes in Computer Science 4435. Berlin: Springer, 2006: 182-196

[83] Abadi M, Baudet M, Warinschi B. Guessing attacks and the computational soundness of static equivalence//Aceto L, Ingólfsdóttir A eds. Lecture Notes in Computer Science 3921. Berlin: Springer, 2006: 398-412

[84] Adão P, Fournet C. Cryptographically sound implementations for communicating processes//Bugliesi M, Preneel B, Sassone V, et al, eds. Lecture Notes in Computer Science 4052. Berlin: Springer, 2006: 83-94

[85] Comon-Lundh H, Cortier V. Computational soundness of observational equivalence//Proceedings of the ACM Conference on Computer and Communications Security. Alexandria, USA, 2008: 109-118

[86] Guttman J D, Thayer F J. Authentication tests and the structure of bundles. Theoretical Computer Science, 2002, 283(2): 333-380

[87] Doghmi S F, Guttman J D, Thayer F J. Searching for shapes in cryptographic protocols//Grumberg O, Huth M eds. Lecture Notes in Computer Science 4424. Berlin: Springer, 2007: 523-537

[88] Guttman J D. Cryptographic protocol composition via the authentication tests//deAlfaro L ed. Lecture Notes in Computer Science 5504. Berlin: Springer, 2009: 303-317

[89] Guttman J D, Thayer F J, Zuck L D. The faithfulness of abstract protocol analysis: Message authentication//Proceedings of the ACM Conference on Computer and Communications Security. Chicago, USA, 2001: 186-195

[90] Herzog J C. The diffie-hellman key-agreement scheme in the strand-space model//Proceedings of the 16th IEEE Computer Security Foundations Workshop. Pacific Grove, USA, 2003: 234-247

[91] Denning D. A lattice model of secure information flow. Communications of the ACM, 1976, 19(5): 236-243

[92] Laud P. Semantics and program analysis of computationally secure information flow//Sands D ed. Lecture Notes in Computer Science 2028. Berlin: Springer, 2001: 77-91

[93] Laud P. Handling encryption in an analysis for secure information flow//Esop D P ed. Lecture Notes in Computer Science 2618. Berlin: Springer, 2003: 159-173

[94] Laud P. Symmetric encryption in automatic analyses for confidentiality against active adversaries//Proceedings of the IEEE Symposium on Security and Privacy. Oakland, USA, 2004: 71-85

[95] Askarov A, Hedin D, Sabelfeld A. Cryptographically-masked flows//Sas Y K ed. Lecture Notes in Computer Science 4134. Berlin: Springer, 2006: 353-369

[96] Laud P. On the computational soundness of cryptographically masked flows//Proceedings of the 35th ACM Sigplan-Sigact Symposium on Principles of Programming Languages. San Francisco, USA, 2008: 337-348

[97] Impagliazzo R, Kapron B M. Logics for reasoning about cryptographic constructions. Journal of Computer and System Sciences, 2006, 72(2): 286-320

[98] Buss S R. Bounded Arithmetic. Napoli, Italy: Bibliopolis, 1986

[99] Barthe G, Daubignard M, Kapron B, Lakhnech Y. Computational indistinguishability logic//Proceedings of the 17th ACM Conference on Computer and Communications Security. Chicago, USA, 2010: 375-386

[100] Lincoln P, Mitchell J C, Mitchell M, Scedrov A. A probabilistic poly-time framework for protocol analysis//Proceedings of the 5th ACM Conference on Computer and Communications Security. San Francisco, USA, 1998: 112-121

[101] Lincoln P, Mitchell J C, Mitchell M, Scedrov A. Probabilistic polynomial-time equivalence and security analysis//Wing J M,

Woodcock J, Davies J eds. World congress on formal methods. Lecture Notes in Computer Science 1708. Berlin: Springer, 1999: 776-793

[102] Mitchell J C, Ramanathan A, Scedrov A, Teague V. A probabilistic polynomial-time process calculus for the analysis of cryptographic protocols. Theoretical Computer Science, 2006, 353(1-3): 118-164

[103] Shoup V. Sequences of games: A tool for taming complexity in security proofs. IACR: Cryptology ePrint Archive, Report 332, 2004

[104] Blanchet B, Pointcheval D. Automated security proofs with sequences of games//Crypto D C ed. Lecture Notes in Computer Science 4117. Berlin: Springer, 2006: 537-554

[105] Blanchet B. A computationally sound mechanized prover for security protocols//Proceedings of the IEEE Symposium on Security and Privacy. Los Alamitos, USA, 2006: 140-154

[106] Blanchet B, Jaggard A D, Scedrov A, Tsay J K. Computationally sound mechanized proofs for basic and public-key Kerberos//Proceedings of the ACM Symposium on Information, Computer and Communications Security. Tokyo, Japan, 2008: 87-99

[107] Blanchet B. A computationally sound mechanized prover for security protocols. IEEE Transactions on Dependable and Secure Computing, 2008, 5(4): 193-207

[108] Backes M, Maffei M, Unruh D. Computationally sound verification of source code//Proceedings of the 17th ACM Conference on Computer and Communications Security. Chicago, USA, 2010: 387-398

LEI Xin-Feng, born in 1973, Ph. D., senior engineer. His main research interests include cryptographic protocols, formal method, modern cryptography and access control.

SONG Shu-Min, born in 1964, Ph. D., senior engineer. His main research interest is security information.

LIU Wei-Bing, born in 1966, bachelor, senior engineer. His main research interest is security information.

XUE Rui, born in 1963, Ph. D., professor, Ph. D. supervisor. His main research interests include modern cryptography, cryptographic protocols and computational complexity.

Background

Computationally Sound Formal Analysis of Cryptographic Protocols is one of the hottest topics in the field of information security. From 1980s, there are two main approaches to security analysis, which developed along with their own directions independently. One of approaches is based on formal models and the other is based on computational models. Each of the approaches has its advantages and disadvantages. The approach of formal models is succinct and easily automatized in analysis, but its abstract features could lose rigorousness in the sense of modern cryptography. The approach of computational models sets on the computational complexity theory and is rigorous in mathematics sense, but the analyzing processes are usually complicated and can be completed only manually by hands, which is proved error prone. Till the beginning of this century, Abadi and Rogaway developed a method to bridge the gap between these two approaches, and established computational soundness of formal security analysis. Intuitively, in security analysis, computational soundness means that if a security property is proved in formal model, then the corresponding property is also valid in computational model. During the last decade, computational soundness has gained a lot of attention, and works in this area are still in full swing.

To survey the researches in this area, this paper summarizes various approaches to computational soundness including the state-of-the-arts techniques. These approaches are divided into four categories, i. e. the approaches based on mapping, the approaches based on simulation, computational soundness of the existed methods and the direct formalization of computational approaches. The motivations, main ideas and features of these approaches are given, and the future research directions are discussed.

This work was supported jointly by the Fund of the National Natural Science Foundation of China under Grant No. 61170280, the Strategic Priority Research Program of the Chinese Academy of Sciences under Grant No. XDA06010701 and IIE's Cryptography Research Project. These projects aim to research the methods for analyzing cryptographic protocols. The team has been working on the research and has published some results, including some papers and books about formal method and computational soundness. A book named Logical Methods in Analyzing Cryptographic Protocols has be published by Science Press on March 2013.