

# 基于刚性与相似性概念的密码协议分析方法

田 园<sup>1)</sup> 王 颖<sup>2)</sup> 金 锋<sup>1)</sup> 金 月<sup>1)</sup>

<sup>1)</sup>(大连理工大学软件学院 辽宁 大连 116620)

<sup>2)</sup>(大连理工大学数学科学学院 辽宁 大连 116620)

**摘 要** 如何融合计算密码学与形式演算模型两条途径以有效分析和证明复杂密码协议,是信息安全领域富有挑战性的问题之一.文中提出 Dolev-Yao 刚性和 Dolev-Yao 相似性概念,运用密码协议的语法骨架提取与语义赋值技术,建立起一个能涵盖除具有适应性入侵能力之外的任何主动攻击者和大部分有实际意义的非自由消息代数的理论分析框架.该框架内所证明的协议安全性质具有复合-稳定性,即所证明的安全性质在协议与环境复合时仍然保持成立.文中针对 strand-图模型这一具体情形证明了 Canetti 的 UC-相似性概念与这里所建立的 Dolev-Yao 相似性概念之间接近充分必要程度的对偶关系,从而对融合 UC-理论/strand-图模型这一情形具体证明了该分析框架具有相容性和完备性.最后,根据以上理论结果讨论了如何建立一种对应的新的协议分析方法.

**关键词** 计算密码学;形式模型;UC-相似;Dolev-Yao 刚性;Dolev-Yao 相似性

中图法分类号 TP309

DOI 号: 10.3724/SP.J.1016.2009.00618

## Non-Malleability and Emulation Based Approach to Cryptographic Protocol Analysis

TIAN Yuan<sup>1)</sup> WANG Ying<sup>2)</sup> JIN Feng<sup>1)</sup> JIN Yue<sup>1)</sup>

<sup>1)</sup>(Software School of Dalian University of Technology, Dalian, Liaoning 116620)

<sup>2)</sup>(School of Mathematical Science, Dalian University of Technology, Dalian, Liaoning 116620)

**Abstract** How to integrate computational and symbolic approaches to analyzing complicated cryptographic protocols is one of the most challenging problems in information security area. In this paper the authors propose a novel theoretical framework to analyze cryptographic protocols covering almost all real-world non-free message algebras and against non-adaptive malicious adversaries, based-upon two novel concepts of “Dolev-Yao nonmalleability” and “Dolev-Yao emulation”, techniques of symbolic extraction and semantic assignment. Security properties proved in this framework are universally composable, i. e., all security properties are provably-preserved when combined with any running environment. The authors prove that Canetti’s concept of UC emulation and the concept of Dolev-Yao emulation are almost sufficient-and-necessarily dual each other with respect to integrating UC theory and strand symbolic model(Dolev-Yao style), and this analysis framework is proven both sound and complete in this case. In addition, a new method for cryptographic protocol analysis is established based-on the above theoretical consequences.

**Keywords** computational cryptography; symbolic model; UC emulation; Dolev-Yao non-malleability; Dolev-Yao emulation

## 1 引言

密码协议(或称安全协议)是保证网络信息安全的重要机制。由于当代高度开放的分布式运行环境蕴涵复杂的不可信任因素,因此对密码协议本身的分析与验证是正确实施任何安全策略的基本前提,而且随着当代信息系统规模越来越大、协议越来越复杂,对密码协议的有效分析与证明也就自然成为信息安全领域重要、复杂但又必须解决的课题之一。

目前有两条途径为密码协议的分析与证明提供解决方案:计算密码学(computational cryptography)方法和形式模型(symbolic model)方法,前者基于算法和计算复杂性的概念刻画攻击者的行为、能力和攻击目标,并以攻击成功的概率定量表达安全程度,而后者以形式符号模型表达高度抽象(理想)的密码方案特性、所容许的攻击者行为及安全目标;在具体论证方式上,前者是典型的归结式论证,通过将协议方案的攻击(在概率多项式算法的意义上)有效归结为对基本组成方案的攻击进行论证,而后者将形式模型看做这种或那种意义上的符号演绎体系,通过特定规则的形式演算来判定形式安全目标是否可以被达到(或被破坏);在结论的涵义上,前者的结论表达为对攻击者拥有特定类型和数量的资源时破坏安全目标的一个定量估计(concrete security),而且其肯定性结论具有现实意义,而后者对安全目标和攻击者能力的表达比较理想化,使得其某些类型的肯定性结论往往并不直接具备现实意义(至少不容易做到);在理论证明的复杂程度上,前者是高度创造性的数学式证明,但对目前越来越复杂的密码协议,不仅证明本身越来越复杂,而且验证证明的正确性也相应地越来越不容易,而后者借助于符号演绎体系的演算机制,有可能部分甚至完全自动化。因此,如何融合计算密码学和形式模型两条途径、各取所长,以有效分析和证明复杂密码协议,成为信息安全测评领域在理论和技术方面都富有挑战性的问题之一。

到目前为止,两条途径在分析与证明密码协议安全性质方面都各自取得了重要进展并且包涵一定意义上的普遍性结果,例如文献[1-5]是关于计算密码学方法的典型工作;文献[6-11]是形式模型方法的典型工作(大量的其它工作可以参考这些论文后面的文献及综述<sup>[12-13]</sup>)。直观地看,只要对形式模型及其分析/演算过程赋予计算密码学解释,似乎不难

得出形式模型方法与计算密码学方法的联系,然而更深入的问题在于:(1)对特定类型的形式模型及其分析技术,这种解释是否总能把一个正确的形式命题解释为目前已经普遍接受的计算密码学意义下的正确命题?这就是所谓相容性问题(soundness);(2)假设我们可以找到这种相容性解释,这意味着对形式模型及其相关的演绎推理过程赋予计算密码学语义,从而使我们可以将正确的形式安全命题“翻译”为正确的计算密码学定理;那么反过来,为得到所有在计算密码学意义上正确的定理,我们是否只需要考虑所有正确的形式安全命题就足够了?这是所谓完备性问题(completeness)。显然,我们希望一个能融合两种方法的“好”的理论分析框架既相容又完备,为此需要深入的理论研究。

除了相容性和完备性之外,建立这种综合分析框架还需要考虑的第3类重要问题是其适用范围,更具体的说,就是所容许的攻击者类型和消息代数的类型。攻击者从大的方面可以划分为被动攻击和主动攻击两种类型,显然后者表达更现实的攻击行为,而且是目前绝大多数计算密码学分析方法所涵盖的攻击类型,而形式模型能否完全表达出各种主动攻击则往往不是显而易见的。另外,按照协议过程中消息的实际构造方式,现实的分析模型当然应该考虑攻击者利用算术运算的可能,典型如 $GF(2)$ 上的运算和群运算,甚至共存的多种算术运算。计算密码学天然地涵盖这一方面,然而形式符号模型对此处理起来要困难一些,这导致不少形式分析工具限于处理所谓自由消息代数,而当建立计算密码学语义时对这一问题的处理会变得复杂起来。显然,最有现实意义的分析框架应该涵盖任意的(概率多项式复杂度的)主动攻击者、并且至少能涵盖代表大多数算术运算特点的非自由消息代数,从而比较完整地表达出攻击者的能力。

为简洁起见,今后也将形式模型方法的有关特性简称为语法特性,而将计算密码学模型或方法的有关特性简称为语义特性。综上所述,融合两种理论分析的目标可以看做是寻求特定形式模型的语义构造,使之相容、完备并且最现实地反映出协议的实际运行环境的行为特点。在这方面,文献[14-16]是开创性的工作,其中 Abadi 与 Rogaway 在文献[14]中针对被动攻击者情形和仅包含一个理想密码算子(对称加密算子)的自由消息代数,证明了其特定计算语义(所谓0-型保密的对称加密方案)的相容性;文献[16]则在一个比较自然的条件下证明了以上语

义是完备的;文献[15]将文献[14]的工作推广到进程代数模型,针对被动攻击者和一个十分简单的进程代数语法证明了类似于文献[14]的相容性质.

文献[14-16]的显著特点是仅包含被动攻击,同时所涵盖的形式模型过于简单,即仅含一种理想加密算子的自由消息代数.文献[18]对包含公钥加密算子的消息代数研究了相应的 Abadi-Rogaway 理论,但仍限于自由消息代数和被动攻击情形.文献[19-20]研究了非自由消息代数模型(都仅限于被动攻击者),其中文献[19]刻画了一类很接近现实的所谓子项-收敛类型的非自由消息代数,并证明了关于这类消息代数形式演绎的两个有效可解性的重要结果(我们的工作将间接地用到这一结果),文献[20]则研究了这类非自由消息代数的计算语义,将 Abadi-Rogaway 理论推广到更广泛的情形,其结论也对我们的工作有支持性涵义(后文具体解释),而文献[21]则明确证明了包含交换群运算和  $GF(2)$  算术的非自由消息代数模型的形式安全分析问题是可解的,具体求解算法是文献[9]的约束求解算法的推广,但文献[21]仅限于证明可解性而未能证明是否有效可解(如果不附加进一步的条件一般地说这是不可能的).

目前涵盖主动攻击者的分析技术也正在深入,文献[22-25,31-32]是这方面的典型工作,特别是文献[22],将新近的 UC-理论<sup>[27-30]</sup>与 Dolev-Yao 形式模型融合起来,针对身份认证和密钥交换协议证明了一类重要结果:密码协议满足计算密码学意义上的安全性质当且仅当其某种对应的形式模型满足某种形式安全性质.从我们的观点看,文献[22]实际上证明了在特定语义赋值之下能够将协议的语法特性与其语义特性精确地联系起来,从而为协议的自动分析/证明(就其所处理的特殊情形而言)建立起令人信服的理论基础.文献[22]仅限于处理含单一公钥加密算子的自由消息代数,文献[23]将其推广到含数字签名算子的情形.文献[24]和文献[25]分别得到与文献[22]和文献[23]类似的结果,但我们认为文献[22]的方法更系统、概念更清楚,同时与强有力的 UC-理论相联系使得有可能进一步作出实质性的推广.另外文献[26]也针对含多项密码算子的非自由消息代数及主动攻击者建立了一个系统的分析框架,并运用这一框架实际分析和证明了一些典型协议的安全性质(这方面的工作请进一步参考文献[26]的参考文献).文献[26]的具体处理技术与文献[22]有许多不同,例如其攻击者模型是 I/O 自动

机而文献[22]是交互式图灵机、文献[22]所依赖的 UC-理论与计算密码学的联系更紧密、更容易处理,而且基于 UC-理论似乎更有希望实现一个概念性更强、更系统的理论,而毋须就每个密码算子的情形单独验证.读者在后文将看到,我们沿循这一思路确实能做到这一点.

关于协议分析与证明方法应该具备的第 4 类重要特性是其所得出的安全性结论是否具有复合-稳定性,即当该协议与任何环境集成(例如被其它协议调用、与其它(恶意)协议并发运行)时其安全性质能否仍然保持有效.作为计算密码学范畴的 UC 理论由于其复合-稳定性定理而具有这一特点(见 3.1 节和文献[27,29-30]),而纯粹的形式模型方法能否具有这一性质就并非显而易见了.

综上所述,最具有现实意义的分析框架应该涵盖典型的非自由消息代数、任意的主动攻击者和具有复合-稳定性质,同时其语义构造应具备相容性和完备性.本文将沿这一方向建立有意义的新概念和新结果.

## 2 本文工作概述

在详细展开具体论述之前,我们概要阐述本文的主要概念与结果,同时概要说明各节内容.首先指出,我们在本文所处理的形式模型属于 Dolev-Yao 模型<sup>[6]</sup>,关于这类形式模型的一个精确的演绎系统就是所谓 strand-图理论<sup>[7-8]</sup>,因此我们对形式模型这部分的处理沿用许多 strand-图的概念和记号.尽管如此,这种处理在概念上并不局限于 strand-图而有可能进一步推广到其它形式模型,例如进程代数.我们对计算密码学这部分的处理则实质性地利用了 Canetti 所创立的 UC-理论的概念和主要结论.

本文所建立的两个关键性概念是密码协议的 Dolev-Yao 刚性和两个形式密码协议之间的 Dolev-Yao 相似性.首先,我们以一种自然的方式使每个以计算形式(computational setting)表达的密码协议  $\pi$  对应一个纯粹的符号形式(symbolic setting)的密码协议  $\tilde{\pi}$ ,反之亦然.在此基础上,直观地说,一个(计算形式的)密码协议  $\pi$  具有 Dolev-Yao 刚性,是指任何(主动或被动)概率多项式复杂度的攻击者  $A$  对  $\pi$  的攻击,在某种意义上讲总是等价于某个 Dolev-Yao 攻击者<sup>[6-7]</sup>  $\tilde{A}$  对  $\tilde{\pi}$  的攻击,或者说, $\pi$  的任何攻击者  $A$  对  $\pi$  所能做到的事情并不超出  $\tilde{\pi}$  的某个 Dolev-Yao 攻击者  $\tilde{A}$  能够对  $\tilde{\pi}$  所做事物的计算语

义. 因此, Dolev-Yao 刚性意味着计算形式的攻击者的能力刚好属于形式模型所能够表达出来的那些类型, 对一个具有 Dolev-Yao 刚性的密码协议  $\pi$  的安全分析也就有可能有效转化为对形式模型  $\bar{\pi}$  的形式安全分析, 而后者是纯粹的符号演算, 并且将形式分析的结论经过语义赋值后有可能重新解释为计算密码学意义上的分析结论. 这就是 Dolev-Yao 刚性概念的意义.

实际上 Dolev-Yao 刚性概念在特殊情况下被某些研究者隐含地使用过, 但我们这里给出的是非常一般的表述. 关于 Dolev-Yao 刚性我们所证明的最重要的结果是第 4 节的刚性复合-稳定性定理, 简单地说就是刚性协议的复合仍然保持刚性以及刚性的相似-遗传定理, 即 UC-相似性<sup>[27,30]</sup>保持 Dolev-Yao 刚性. 以上这个非常普遍性的结果使 Dolev-Yao 刚性不仅是一个重要概念, 而且成为一个有效的工具, 例如我们能据此推出几乎所有具有自然的语义解释的密码协议实际上都是 Dolev-Yao 刚性的, 从而能将这些协议的分析转化为形式分析而不会遗漏任何可能的攻击类型. 我们还将指出如何运用这些结论以特例的形式导出以往一些工作的重要结果.

我们建立的第 2 个关键概念是 Dolev-Yao 相似性. 直观地说, 一个符号形式的密码协议  $\bar{\pi}_2$  Dolev-Yao 相似于另一个符号形式的密码协议  $\bar{\pi}_1$ , 是指  $\bar{\pi}_2$  的任何 Dolev-Yao 攻击者对  $\bar{\pi}_2$  在形式演绎的意义上所能做的任何事情, 都存在  $\bar{\pi}_1$  的某个攻击者对  $\bar{\pi}_1$  做到同样的事情. 熟悉 UC-理论的读者会立刻看出这一概念与 UC-相似性<sup>[27,30]</sup> (UC-emulation) 的类似. 我们将在第 5 节证明这两个概念具有几乎是充分必要程度的对偶特征, 概要地说就是: 如果在某种自然的(其中之一与刚性有关)条件下,  $\pi_2$  UC-相似于  $\pi_1$ , 则  $\bar{\pi}_2$  Dolev-Yao 相似于  $\bar{\pi}_1$ ; 反之, 若  $\bar{\pi}_2$  Dolev-Yao 相似于  $\bar{\pi}_1$ , 则在某种自然的条件下(其中之一也与刚性有关)  $\pi_2$  UC-相似于  $\pi_1$ . 值得指出的是, 这些定理涵盖很广的一类非自由消息代数和除具有适应性入侵能力之外的任意的主动攻击者. 既然(根据 UC-理论的观点)对计算形式的密码协议  $\pi$  的安全性证明实质上就是要证明  $\pi$  与某个具有安全性质的理想密码协议(例如 UC-理论的所谓 ideal functionality 或含理想协议的复合协议)  $\pi^*$  UC-相似, 因此这两个结论将协议的计算密码学意义上的证明与其符号演算意义上的形式分析精确地联系起来, 而且分别表达了我们所建立的分析框架的完备性和相容性. 如果与经典的 Abadi-Rogaway 定理<sup>[14]</sup> 和

Macciancio-Warinschi 定理<sup>[16]</sup> 比较, 则不难看出这些结论实质性地推广了他们的工作.

以上是对本文工作的概述, 也见图 1, 该图表明我们的主要结果(定理 7 和定理 8)如何将计算密码学意义上的安全证明转化为形式演算意义上的安全证明. 但是与目前许多形式分析/证明技术不同, 这里的形式安全证明意味着验证两个形式密码协议之间是否满足某种关系(即 Dolev-Yao 相似性关系, 这使得其中一个形式协议在实际应用中可用来起规范(specification)的作用, 或起 Hybrid-argument 式论证的中间过渡协议的作用, 但我们所建立的分析方法本身并不依赖于任何这类具体解释), 而非目前许多工作那样意味着检验单独一个密码协议是否满足某种形式安全要求<sup>[7-13]</sup>, 在这一点上这里的形式安全证明风格更类似于双向模拟(bisimulation)证明技术. 最后, 由于 UC 相似概念具有复合-稳定性, 因此这里所建立的分析框架中所得出的安全性结论将具有最普遍的适用性.



图 1 融合 UC-理论和 Dolev-Yao 模型的协议分析与证明框架

从下一节开始我们进入精确的论述. 因为本文所用理论工具较多, 一些基础性概念和辅助性结果汇集于第 3.1~3.4 节, Dolev-Yao 刚性和 Dolev-Yao 相似性的精确概念则在第 3.5 小节建立. 第 4 节和第 5 节建立和证明本文的主要定理、解释其涵义, 同时说明这些非常普遍的结果如何将以往一些典型结果作为特例涵盖其中以及如何根据本文的理论结果发展一种新分析方法. 限于篇幅, 本文重点解决基础理论问题, 第 6 节总结并指出待研究的问题.

## 2.1 基本术语与符号

这里概要回顾某些术语并给出符号约定. 概率多项式算法简称为 P.P.T. 算法. 正整数  $k$  的可忽略函数是指这样一类函数, 当  $k$  充分大时函数值下降得比任何多项式的倒数都要快.  $x \parallel y$  表示字  $x$  和  $y$  的联结, 而且我们约定这种联结是有结构的, 即从  $x \parallel y$  总可以准确恢复出  $x$  和  $y$ .  $|x|$  表示字  $x$  的编码长度. 设  $X$  是一个集合, 记号  $a \leftarrow^{\$} X$  表示从  $X$  随机选取一个元素  $a$  ( $a$  在  $X$  上均匀分布). 符号  $\perp$  用以

表示算法在异常或错误情况下的输出。 $\approx^{\text{PPT}}$ 表示两个变量 P.P.T. 不可分辨 (P.P.T. indistinguishable),  $\approx^{\text{PDF}}$ 表示两个变量有相同的概率分布。

### 3 基础理论、辅助性结果及 Dolev-Yao 刚性与相似性概念

本文需要的理论工具较多,我们将所需要的概念和主要结论汇集于此.为节省篇幅,本节的阐述尽可能概略但同时指出原始文献以供参阅.3.5节建立本文的两个新概念.

#### 3.1 UC-理论

UC-理论用四元组 $(\pi, A, Z, u)$ 表达密码协议的实际运行,其中 $\pi$ 是协议, $A$ 是攻击者, $Z$ 是运行环境, $u$ 是 $Z$ 的输入变量, $\pi, A$ 和 $Z$ 都是交互式 P.P.T. 算法.环境 $Z$ 表达协议 $\pi$ 在实际运行时可能与之发生输入/输出关系的、除攻击者 $A$ 之外的其它一切进程(例如其它的协议进程以及那些协议的攻击者).协议运行期间 $A$ 和各个 $\pi$ -进程的输出序列记作 $Exec(\pi, A, Z; u)$ ,这本质上是一个随机序列.关于 UC 运行模型的详细讨论和应用参阅文献[27-30].

今后区分两类主动攻击者 $A$ .第1类是仅具有非适应性入侵能力(non-adaptive corrupt)的攻击者,其入侵对象在开始实施攻击之前就完全确定下来;第2类是具有适应性入侵能力的攻击者,其入侵对象在实施攻击期间动态决定.第2类主动攻击者具有最强的攻击能力,但一般来说难以用形式模型准确刻画,即使运用计算密码学方法也比较难以分析,而第1类攻击者可以由许多形式模型表达出来.另一方面,第1类攻击者在许多情形下足以代表充分现实的攻击行为,包括目前大多数计算密码学理论分析与证明工作都是针对这类攻击.本文主要考虑第1类攻击者.

UC-理论还建立另一类运行模型,称为 UC-理想模型(ideal functionality),用来表达协议的安全性质和某些附加性质.在这类运行模型中,密码协议被表达为一个由某个算法 $F^{\text{ideal}}$ 集中控制的过程,每个参与方仅向 $F^{\text{ideal}}$ 提交输入, $F^{\text{ideal}}$ 则根据各方提供的输入集中计算出正确的输出并返回给相应的参与方,参与方之间不发生消息交换.攻击者 $S$ 与环境 $Z$ 任意地相互作用,但 $S$ 仅仅与 $F^{\text{ideal}}$ (而非与协议的各个参与方)相互作用, $F^{\text{ideal}}$ 如何与 $S$ 相互作用则是特定理想模型的内容之一.直观地看,理想模型是一种对攻击者 $S$ 的能力加以一定限制的模型,正是这

种限制表达出密码协议应该具有的安全性质.

**定义 1**(UC-相似).  $\psi, \varphi$  是密码协议,  $Z^{\text{exec}}(\psi, A, u)$  表示环境  $Z$  通过收集  $\pi, A$  和  $Z$  在外部输入  $u$  之下的相互作用结果而生成的输出(不失一般性,总可以约定该输出为 0 或 1),  $Z^{\text{exec}}(\varphi, S, u)$  的涵义类似. 如果对任何  $A \in \text{P.P.T.}$  都存在  $S \in \text{P.P.T.}$ , 使得对任何  $Z \in \text{P.P.T.}$  和任何  $u, \delta_{A,S}^{\psi, \varphi}(k) \equiv |P[Z^{\text{exec}}(\psi, A, u) = 1] - P[Z^{\text{exec}}(\varphi, S, u) = 1]|$  都是复杂性参数  $k$  的可忽略函数(今后记作  $Z^{\text{exec}}(\psi, A, u) \approx^{\text{PPT}} Z^{\text{exec}}(\varphi, S, u)$ ), 则  $\psi$  定义为与  $\varphi$  UC-相似, 记作  $\psi \rightarrow^{\text{UC}} \varphi$ . 算法  $S$  称为  $A$  的仿真算法.

UC-相似性概念最有价值的特点是复合-稳定性,直观地说就是:如果一个协议 $\pi$ 调用协议 $\varphi$ ,则在 $\pi$ 调用 $\varphi$ 的每个地方用与 $\varphi$  UC-相似的协议 $\psi$ 来替换,所得到的新协议将仍然是原来的协议 $\pi$ 的 UC-相似协议. UC-相似的复合-稳定性是一个非常宝贵的协议分析、证明与设计工具.为对此建立精确的表述,首先建立 UC-细化和协议独立性两个重要概念,然后在此基础上表述 UC-复合稳定性定理.

**定义 2**(子协议、复合协议及细化).  $\pi, \varphi$  和  $\psi$  都是协议. 如果  $\pi$  调用  $\varphi$  且  $\pi$  与  $\varphi$  的每个实例之间仅有(调用时的)输入和(调用结束时的)输出关系,则  $\varphi$  定义为  $\pi$  的子协议,  $\pi$  定义为基于  $\varphi$  的复合协议( $\varphi$ -Hybrid protocol), 记作  $\pi(\varphi)$ . 在  $\pi$  的程序中将其对子协议  $\varphi$  的调用都替换为对  $\psi$  的调用,同时保持所有对  $\varphi$  的输入变量不变且用  $\psi$  的输出变量替换  $\varphi$  的输出变量,这一变换定义为 UC-细化,由此得到的新协议记作  $\pi(\psi/\varphi)$ , 或简记为  $\pi(\psi)$ . 后面表达 Dolev-Yao 刚性的复合-稳定性定理时也要用到 UC-细化的概念.

一个协议  $\pi$  可能是基于多个子协议  $\varphi_1, \dots, \varphi_N$  的复合协议,这时记为  $\pi(\varphi_1, \dots, \varphi_N)$ , 相应的 UC-细化记为  $\pi(\psi_1/\varphi_1, \dots, \psi_N/\varphi_N)$  或简记为  $\pi(\psi_1, \dots, \psi_N)$ .

**定义 3**(协议独立性). 若协议  $\psi_1$  的任何运行实例既不显式也不隐式地(例如通过某个子程序或某个高层控制程序)访问协议  $\psi_2$  的任何运行实例,同时  $\psi_2$  对  $\psi_1$  也有同样性质,则协议  $\psi_1$  和协议  $\psi_2$  定义为相互独立. 若  $\psi$  自身的任何运行实例之间从来既不发生显式访问,也不发生隐式访问,则协议  $\psi$  定义为是自独立的.

**定理 1**(UC 复合-稳定性定理). 设  $\pi(\varphi)$  是基于子协议  $\varphi$  的复合协议, 协议  $\varphi$  和协议  $\psi$  都是自独立的,  $\varphi$  和  $\psi$  之间也是相互独立的. 若  $\psi \rightarrow^{\text{UC}} \varphi$ , 则必有  $\pi(\psi/\varphi) \rightarrow^{\text{UC}} \pi(\varphi)$ .

关于 UC-理论及复合-稳定性定理的证明及详细讨论见文献[27, 29-30].

### 3.2 Dolev-Yao 形式模型与 strand-图模型

strand-图模型是一种具体的 Dolev-Yao 风格的形式模型<sup>[6-7, 18]</sup>. strand-图模型的基本元素是一个常元符号集合  $\mathcal{A}$ 、一个密钥符号集合  $\mathcal{K}$ 、一个明文集合  $\mathcal{T}$  和一个变元集合  $\mathcal{V}$ , 并且这 4 个基本集合互不相交. 在此之上进一步构造出消息、事件和 strand.

消息是一个形式表达式, 其形式定义详见文献[7]. 协议的所有消息的集合称为一个消息代数. 文献[7]定义的实际上是仅含加密算子的自由消息代数, 是下一小节将要阐述的含任意多项密码算子的、非自由消息代数的子类, 但为论述清晰, 先讨论这一特殊情形.

对消息  $M$ , 带符号的消息式  $+M$  或  $-M$  分别表示发送消息  $M$  和接收消息  $M$ . strand 是一个带符号的消息序列及一个指定的消息作用点, 作用点的位置用下划线表示:

$$s = E_1, E_2, \dots, E_{i-1}, \underline{E_i}, E_{i+1}, \dots, E_s.$$

称  $s = \dots, +t, \dots$  是正号的,  $s = \dots, -t, \dots$  是负号的.  $E_i = \pm M_i$ . 不标记作用点位置的事件序列称做一个迹 (trace). 设  $s' = E'_1, E'_2, \dots, E'_{k-1}, \underline{E'_k}, E'_{k+1}, \dots, E'_s$  是另一个 strand, 若  $E'_1 = E_1, \dots, E'_i = E_i$  且  $k = i+1$ , 则定义  $s$  和  $s'$  有关系  $s < s'$ ; 若  $i = k$  且  $E_i = +M, E'_i = -M$ , 则定义  $s$  和  $s'$  有关系  $s \rightarrow s'$ . 两个 strand 之间不同时具有关系“ $<$ ”和“ $\rightarrow$ ”.

strand-图是一个无圈有向图  $(S, D)$ ,  $S$  是 strand 的一个集合, 有向边  $\langle s, s' \rangle \in D$ , 若  $s < s'$  或  $s \rightarrow s'$ , 并且有性质:

(1) 若  $s' \in S$  且  $s < s'$ , 则  $s \in S$ ;

(2) 若  $s' \in S$ ,  $s' = \dots, -M, \dots$ , 则必有且仅有一个  $s \in S$ ,  $s = \dots, +M, \dots$ , 使  $s \rightarrow s'$ .

文献[7]详细刻画了以上自由消息代数情形下的 Dolev-Yao 攻击者 strand 及其与协议的正常 strand 相互作用规则. 正常 strand 表达协议的运行实例而一个 strand-图表达 Dolev-Yao 攻击者对协议的一个攻击实例. 在实际应用中, 常对某些 strand 的消息作用点指定特定的可观测事件, 这些事件之间的序关系就定义为这些作用点的相对时间次序. 将 strand-图的所有事件按照 Lamport 序排列出来 (若两个事件没有序关系则次序任意) 得到一个表达式, 定义为该 strand-图的输出.

容易将所有这些概念推广到任意的非自由消息代数. 最后指出一点, strand 和 strand-图都可以用

图论形式等价而直观地表达出来<sup>[7-8]</sup>, 今后在阐述时也常借助于这种图论语言, 例如将一个 strand 的消息作用点称为其所在 strand-图的节点, 并且依该消息作用点是正号还是负号分别称为正节点和负节点. 这两种表述形式之间很容易相互转换.

### 3.3 含任意多项密码算子的非自由消息代数及其计算语义的一般性构造

这一小节更一般性地描述密码协议的形式模型及其计算语义, 并引述一些有用的普遍结论, 详细论述参考文献[19-20].

一个消息代数  $\mathcal{A}$  是一个 5 元组  $(S, \mathcal{F}, \mathcal{N}, \mathcal{V}, \mathcal{E})$ , 其中  $S$  是域符号的集合;  $\mathcal{F}$  是函数符号的集合, 并且对每个函数符号  $f \in \mathcal{F}$  关联有一个非负整数和  $S$  中的域符号的一个有序组, 前者表示函数变元的个数, 记作  $ar(f)$ , 0-元函数称为常数, 后者表示函数的定义域和值域, 记作  $S_1 \times \dots \times S_{ar(f)} \rightarrow S$ ;  $\mathcal{N}$  是随机符号的集合, 并且对其中每个符号  $a \in \mathcal{N}$  关联一个域符号, 记作  $S(a)$ ;  $\mathcal{V}$  是变元符号的集合;  $\mathcal{E}$  是形如  $M = N$  的形式表达式的有限集合, 其中  $M$  和  $N$  是消息项 (今后简称为项), 项的递归定义如下:

变元符号  $x \in \mathcal{V}$  是项;

随机符号  $a \in \mathcal{N}$  是项;

若  $t_1, \dots, t_m$  是项,  $f \in \mathcal{F}$  且  $ar(f) = m$ , 则  $f(t_1, \dots, t_m)$  是项.

今后用  $fn(t)$  表示项  $t$  所含随机符号的集合,  $fv(t)$  表示项  $t$  所含变元符号的集合.

将消息代数  $\mathcal{A}$  的项的集合记为  $\mathcal{T}$ , 定义  $\mathcal{T}$  上的关系  $=_E$  如下:

若  $(M = N) \in \mathcal{E}$ , 则  $M =_E N$ ;

若  $M =_E N$ ,  $\sigma$  是变元替换, 则  $\sigma(M) =_E \sigma(N)$ ;

若  $M_1 =_E N_1, \dots, M_s =_E N_s, f \in \mathcal{F}$  且  $ar(f) = s$ , 则  $f(M_1, \dots, M_s) =_E f(N_1, \dots, N_s)$ .

与上一小节关于消息代数的定义不同, 这里的定义并不针对任何特定的密码运算, 例如对称加密/解密算子等, 而是通过  $\mathcal{E}$  中的形式等式来一般性地刻画这些运算. 这样可以将自由消息代数和自由消息代数统一处理, 而不是分开进行. 这时起关键作用的是  $\mathcal{E}$  的形式特征, 一些典型的实例见文献[20-21].

$\mathcal{A}$  是消息代数, 引进一个新符号  $\nu$ , 称为约束量词.  $\mathcal{A}$  上的观测式 (frame) 是一个形式公式  $\varphi = \nu \bar{a}. \sigma$ , 其中  $\bar{a}$  是一组随机符号  $a_1, \dots, a_m$ , 称为  $\varphi$  的约束元或非自由元,  $\sigma$  是一个变元替换  $\{t_1/x_1, \dots, t_n/x_n\}$ ,  $x_1, \dots, x_n$  是变元,  $t_1, \dots, t_n$  是项且规定各  $t_i/x_i$  的位置是有顺序的并把位置记作  $t_1/x_1 \leq \dots \leq t_n/x_n$ , 最



后还有  $fn(t_1) \cup \dots \cup fn(t_n) \subseteq \{a_1, \dots, a_m\}$ . 记  $dom(\varphi) = \{x_1, \dots, x_n\}$ ,  $fv(\varphi) = fv(t_1) \cup \dots \cup fv(t_n)$ , 若  $fv(\varphi)$  为空, 则称  $\varphi$  是闭的.

与文献[19-20]的约定类似, 今后总是使用闭观测式. 注意  $\bar{a}$  可以为空, 这时  $\varphi$  退化为纯粹的变元替换.  $\mathcal{A}$  上的一个观测式  $\varphi$  表示网络上的任何观察者在协议执行期间所记录的属于消息代数  $\mathcal{A}$  上的一组消息项,  $\varphi$  的约束元表示协议在被观察期间生成的对象, 例如随机数、会话密钥等, 而各个替换项位置的顺序关系  $\leq$  用以表达这些对象被观测到的时间顺序. 观测式在文献[19-20]中用来表达被动攻击, 但如果包含由攻击者生成的项, 也可以用来表达协议在任意主动攻击下的结果, 后面正是这样来使用观测式的.

设  $T$  是一个项,  $\varphi = \nu \bar{a}. \sigma$  是一个观测式. 若存在项  $M$  使  $fv(M) \subseteq dom(\varphi)$ ,  $fn(M) \cap (fn(\varphi) \cup fn(T))$  是空集且  $\sigma(M) =_E T$ , 则定义  $\varphi \models_E T$ . 为记号简捷, 今后记  $\sigma(M)$  为  $\varphi(M)$ .

设  $\varphi_1, \varphi_2$  是两个观测式,  $dom(\varphi_1) = dom(\varphi_2)$ . 若对任何项  $M, N$ ,  $fv(M) \subseteq dom(\varphi_1)$ ,  $fv(N) \subseteq dom(\varphi_2)$ , 有  $\varphi_1(M) =_E \varphi_1(N)$  当且仅当  $\varphi_2(M) =_E \varphi_2(N)$ , 则定义  $\varphi_1 \cong_E \varphi_2$ .

关于 Dolev-Yao 形式模型的以上刻画方法, 文献[19]证明了一个重要结论, 对我们的工作有支持性涵义, 概要地说就是: 如果模型中表达非自由约束的等式集合  $\mathcal{E}$  属于所谓收敛的子项理论 (convergent subterm equational theory) 范畴, 则这类消息代数上  $\models_E$  和  $\cong_E$  判定问题均有效可解. 文献[19-20]有关于这类消息代数的精确定义和许多实例, 它们表明这类非自由代数能涵盖非常广泛的、具有现实意义的情况.

**定理 2**<sup>[19]</sup>. 设  $\mathcal{E}$  是收敛的子项理论, 则必存在多项式复杂度算法  $A_E^*$  和  $A_{E^*}^*$  分别判定任何消息表达式  $M$  和观测式  $\varphi, \varphi_1$  是否有  $\varphi \models_E M$  和  $\varphi \cong_E \varphi_1$ .

对以上形式模型可以非常一般性地建立其计算语义.  $\mathcal{A} = (\mathcal{S}, \mathcal{F}, \mathcal{N}, \mathcal{V}, \mathcal{E})$  是一个消息代数,  $\mathcal{A}$  的计算语义  $\mathcal{A}^*$  是对其符号和形式表达式按以下方式赋予的解释<sup>[20]</sup>:

对  $\mathcal{S}$  中的每个域符号  $S$ , 指派一个长度不定的二进制数的集合  $[S]_A$ .

对  $\mathcal{F}$  中的每个函数符号  $f: S_1 \times \dots \times S_{arf(f)} \rightarrow S$ , 指派一个  $arf(f)$ -元函数  $f_A: [S_1]_A \times \dots \times [S_{arf(f)}]_A \rightarrow [S]_A$ , 并且  $f_A$  是作用于其自变量上的 P.P.T. 算法.

对  $\mathcal{S}$  中的每个域符号  $S$ , 指派  $[S]_A$  上的一个可 P.P.T-检验的等价关系  $=_A$ , 即一个 P.P.T. 算法  $X_{S,A}$ , 对任何元素  $u, v \in [S]_A$ , 记  $u =_A v$  表示  $X_{S,A}(u, v) = 1$ , 则  $=_A$  满足自反、对称和传递性质.

对  $\mathcal{S}$  中的每个域符号  $S$ , 指派  $[S]_A$  上的一个 P.P.T-采样算法  $R_{S,A}$  (不要求采样遵循特定的分布).

用  $R_{S,A}$  生成一个样本  $x$  的过程记作  $x \xleftarrow{R} [S]_A$ .

对观测式  $\varphi = \nu \bar{a}. \sigma$ , 其中  $\bar{a} = (a_1, \dots, a_m)$ ,  $\sigma = \{t_1/x_1, \dots, t_n/x_n\}$ , 指派一个分布  $[\varphi]_A$  (称为  $\varphi$  的语义赋值),  $[\varphi]_A$  的样本 (称为  $\varphi$  的语义值) 按以下过程递归计算:

对每个  $a_i \in \bar{a}$ , 若  $a_i$  关联的域符号是  $S_i$ , 则  $[a_i]_A \xleftarrow{R} [S_i]_A$ ;

若项  $M = M_1 M_2$ , 则  $[M]_A = [M_1]_A \parallel [M_2]_A$ ;

若项  $M = f(M_1, \dots, M_s)$ , 则  $[M]_A = f_A([M_1]_A, \dots, [M_s]_A)$ ;

最后,  $[\varphi]_A = \{[t_1]_A/x_1, \dots, [t_n]_A/x_n\}$ . 这一定义也连带建立了项的语义值.

对非闭观测式  $\varphi$ , 设  $fv(\varphi) = \{y_1, \dots, y_n\}$ , 固定对自由变元  $y_1, \dots, y_n$  的任何一组赋值  $Y_1, \dots, Y_n$ , 称为一个自由赋值点, 定义  $\varphi$  在该赋值点上的语义值  $[\varphi]_{A, y_1=Y_1, \dots, y_n=Y_n}$  为按以上过程计算的结果, 其中自由变元  $y_1, \dots, y_n$  的值固定在  $Y_1, \dots, Y_n$  上. 对项  $M$  也可以完全类似地定义  $[M]_{A, y_1=Y_1, \dots, y_n=Y_n}$ .

今后用  $[\Theta]_A^k$  表示  $\mathcal{A}$  的形式表达式  $\Theta$  在复杂度参数的固定值  $k$  之下的分布, 因此  $[\Theta]_A = \{[\Theta]_A^k\}_{k>0}$ .

**定义 4**( $\cong_{E^-}$ -相容<sup>[20]</sup>).  $\mathcal{A}$  的计算语义  $\mathcal{A}^*$  定义为  $\cong_{E^-}$ -相容的, 若对任何观测式  $\varphi_1, \varphi_2$ ,  $\varphi_1 \cong_E \varphi_2$  蕴涵  $[\varphi_1]_A^k \approx^{\text{PPT}} [\varphi_2]_A^k$  ( $\approx^{\text{PPT}}$  表示 P.P.T. 不可分辨).

$\cong_{E^-}$ -相容是我们需要的一个重要概念, 对此文献[20]有详细讨论, 并且建立了一个关于  $\cong_{E^-}$ -相容性的接近充分必要程度的判定条件, 涵盖很广泛的一类形式模型. 这一条件正是经典的 Abadi-Rogaway 模式函数<sup>[14]</sup> (pattern) 的一个很一般性的推广. 我们的工作并不直接需要这些结果, 但它表明我们所建立的分析框架和对此所证明的结论的普遍性程度.

### 3.4 协议的形式模型的计算语义与算法模型的语法骨架

这里将 3.2~3.3 小节的理论综合起来, 对 strand-图这一具体的 Dolev-Yao 风格的密码协议形式模型建立一种自然的语义赋值方法, 同时对计算

形式的密码协议给出一种自然的语法骨架提取方法. 前者使一个符号形式的密码协议被解释为一个计算形式的密码协议, 后者将一个计算形式的密码协议与一个符号形式的密码协议对应起来. 这种技术曾在这种或那种特殊形式下为某些研究者使用过, 例如文献[18, 22], 本节的处理与其本质相同, 但以更具有普遍性同时也最适合于本文需要的形式表述这些内容.

从现在起处理的都是含任意密码算子的、非自由消息代数. 根据第 3.2~3.3 小节的刻画, 一个密码协议形式模型的核心是其消息代数  $A = (\mathcal{S}, \mathcal{F}, \mathcal{N}, \nu, \mathcal{E})$ . 我们对每个函数符号  $f \in \mathcal{F}$ ,  $\text{arf}(f) = m$  引进一种一般形式的  $f$ -strand, 形式为  $-t_1 \cdots -t_m + f$ , 用以取代以往 strand-图模型中针对特定密码算子所定义的 strand, 从而将 strand-图概念推广到适合任意非自由消息代数的情况. 每个 strand-图表达攻击者对符号形式密码协议的某种攻击, 可以用一种系统而自然的程序对这一攻击赋予算法解释, 即构造给定 strand-图的计算语义, 详见文献[18].

现在阐述从计算形式到符号形式方向的两个过程, 分别记为  $\text{symb}_P$  和  $\text{symb}_T$ .  $\text{symb}_P$  从给定的计算形式的密码协议  $\pi$  构造出一个纯粹符号形式的密码协议  $\bar{\pi}$ , 而  $\text{symb}_T$  从计算形式的密码协议 (包括与攻击者相互作用) 的输出  $\tau$  构造出一个 strand-图的输出  $\bar{\tau}$ .

先描述  $\text{symb}_P(\pi)$ , 其中  $\pi$  是协议程序. 为处理方便, 假设  $\pi$  不含任何循环, 实际上很多实际的协议程序都符合这一要求, 或可以容易地用一个等价的无循环程序来表达.

$\text{symb}_P(\pi)$  定义如下:

建立一个表  $\Delta$  和 strand  $\bar{\pi}$ , 两者初始均为空;

对  $\pi$  的每个变量指派一个变元符号, 相同的变量被指派同一个变元符号, 不同的变量则被指派不同的变元符号; 对常数、随机数和算法调用也类似处理, 分别指派常元符号、随机符号和函数符号; 将这些对应关系记录在  $\Delta$  中.

根据  $\pi$  中出现的各种运算的实际性质建立相应的等式约束集合  $E$ .

对  $\pi$  的语句顺序扫描, 每当出现  $x^* \leftarrow t_1^* \parallel t_2^*$ , 则在  $\Delta$  中记录  $(x, t_1^* \parallel t_2^*)$ ; 每当出现算法调用  $x^* \leftarrow f^*(t_1^*, \dots, t_m^*)$ , 便在  $\Delta$  中记录  $(x, f(t_1^*, \dots, t_m^*))$ , 其中  $x$  是为  $x^*$  指派的变元符号、 $f$  是为  $f^*$  指派的函数符号、 $t_i$  是  $t_i^*$  的形式表达式 (由程序  $\pi$  的因果性质, 这时  $t_i^*$  对应的形式表达式  $t_i$  一定已经被计算

出来).

对  $\pi$  的语句顺序扫描, 每当出现发送消息的语句  $\text{send}(t^*)$ , 便在 strand  $\bar{\pi}$  中插入  $+t$ , ; 每当出现接收消息的语句  $\text{recv}(t^*)$ , 便在 strand  $\bar{\pi}$  中插入  $-t$ , 其中  $t$  是  $t^*$  的形式表达式 (由程序  $\pi$  的因果性质, 这时  $t^*$  对应的形式表达式  $t$  一定已经被计算出来).

如此最后得到的 strand  $\bar{\pi}$  定义做  $\text{symb}_P(\pi)$ . 不难看出  $\text{symb}_P$  是一个多项式复杂度算法, 而且按照前面的方法对  $\bar{\pi}$  赋予算法语义就恰好得到  $\pi$ .

现在描述  $\text{symb}_T(\tau)$ , 其中  $\tau$  是协议程序  $\pi$ 、攻击者  $A$  和环境  $Z$  在外部输入  $u$  之下相互作用的输出样本, 记作  $\tau \in \text{Exec}(\pi, A, Z; u)$  (注意  $\text{Exec}(\pi, A, Z; u)$  是随机变量而  $\tau$  是其样本或实例). 我们还假设消息代数上的  $\models_E$ -问题或等价的  $\not\models_E$ -问题是多项式可判定的, 其中等式约束集合  $E$  源自上面对  $\text{symb}_P(\pi)$  的计算.

设  $\tau = e_1 e_2 \cdots e_N$ , 其中各  $e_i$  是输出事件. 算法  $\text{symb}_T(\tau)$  置  $\bar{\tau}$  初始为空, 然后分两个阶段完成计算. 第一阶段顺序扫描  $\tau$  的每个事件, 目的是生成相关的形式符号:  $\text{symb}_T(\tau)$  生成一个表  $\Sigma$ , 其初始内容包括环境  $Z$  的所有输入变量  $u_1, \dots, u_m$  对应的形式符号  $c_1, \dots, c_m$  且将所有这些符号作为常元符号; 设当前  $\bar{\tau} = \nu \bar{a}. \sigma$ , 对当前事件  $e_i$ , 设  $y^*$  是其输出变量, 若  $y^*$  是随机数且  $\Sigma$  中不存在关于  $y^*$  的表项, 则生成一个新符号  $a$ , 将  $(y^*, a)$  插入  $\Sigma$ 、 $\bar{a} \leftarrow \{a\} \cup \bar{a}$ 、 $\bar{\tau} \leftarrow \nu \bar{a}. \sigma \cup \{a/y\}$ , 其中  $\cup \leq$  表示将  $a/y$  作为  $\leq$  关系 (参见 3.3 小节) 的最后一项插入  $\sigma$ ; 若  $y^*$  是随机数且  $\Sigma$  中存在表项  $(y^*, a)$ , 则  $\bar{\tau} \leftarrow \nu \bar{a}. \sigma \cup \{a/y\}$ ,  $\cup \leq$  涵义如前; 若  $y^*$  并非随机数而是来源于计算  $y^* \leftarrow f^*(x_1^*, \dots, x_m^*)$ ,  $f^*$  对应的函数符号是  $f$ , 则首先 (递归地) 计算出各  $x_i^*$  的形式表达式  $\Sigma(x_i^*)$  (若该表达式尚不存在于  $\Sigma$  中)、将  $(x_i^*, \Sigma(x_i^*))$  插入  $\Sigma$ 、 $\bar{a} \leftarrow \{a_1, \dots, a_i\} \cup \bar{a}$ 、 $\bar{\tau} \leftarrow \nu \bar{a}. \sigma \cup \{ \Sigma(x_1^*)/x_1 \} \cup \dots \cup \{ \Sigma(x_m^*)/x_m \} \cup \{ f(\Sigma(x_1^*)/x_1, \dots, \Sigma(x_m^*)/x_m)/y \}$ , 其中  $a_1, \dots, a_i$  是出现于  $\Sigma(x_1^*), \dots, \Sigma(x_m^*)$  之一但不属于  $\bar{a}$  的所有自由变元.

设第一阶段结束后有  $\bar{\tau} = \nu \bar{a}. \sigma_0 \cup \{ M_1/x_1, \dots, M_N/x_N \}$ , 其中  $\sigma_0$  只含形如  $c/u$  的替换、 $c$  是常元,  $M_i$  都是消息代数表达式. 记  $\bar{\tau}_0 = \sigma_0$ ,  $\bar{\tau}_i = \nu \bar{a}. \sigma_0 \cup \{ M_1/x_1, \dots, M_i/x_i \}$ ,  $i=1, \dots, N$ , 第二阶段顺序扫描每个  $M_i/x_i$ , 若  $\bar{\tau}_{i-1} \not\models_E M_i$ , 则终止并输出  $\perp$ .

如此最后得到的  $\bar{\tau}$  就作为  $\text{symb}_T(\tau)$  的结果. 当  $\models_E$  或等价地  $\not\models_E$  多项式可判定, 则  $\text{symb}_T(\tau)$  是多项



式复杂度算法. 不难直接验证, 若  $\bar{\tau} = \text{symp}_T(\tau) \neq \perp$ , 对  $\bar{\tau}$  按照文献[18]的方法赋予计算语义, 那么  $\tau$  恰是  $\bar{\tau}$  的一个语义值.

### 3.5 密码协议的 Dolev-Yao 刚性和 Dolev-Yao 相似性

现在建立两个重要的新概念. 首先注意到, 一个 strand-图总可以看做是形式密码协议  $\pi$  的合法 strand 和 Dolev-Yao 攻击者 strand  $\tilde{A}$  这两个子图在特定的消息作用点上粘合而成, 为明确攻击者  $\tilde{A}$ , 我们今后也将 strand-图记作  $(\pi, \tilde{A})$ . 将 strand-图  $(\pi, \tilde{A})$  的输出 (参见 3.2 小节) 中所出现的随机符号都作为束缚变元用  $\nu$ -量词约束起来, 如此就得到一个观测式, 记作  $\text{Trace}(\pi, \tilde{A})$ .

**定义 5** (Dolev-Yao 刚性).  $\pi$  是计算形式的密码协议,  $\bar{\pi} = \text{symp}_P(\pi)$ . 若对任何 P.P.T. 攻击算法  $A$ 、任意的环境  $Z$  和输入  $u$  都存在复杂性参数  $k$  的可忽略函数  $\delta(k)$  使

$P[\tau \in \text{Exec}(\pi, A, Z; u): \text{存在 Dolev-Yao 攻击者 } \tilde{A} \text{ 使 } \text{symp}_T(\tau) = \text{Trace}(\pi, \tilde{A})] > 1 - \delta(k),$   
(上式的概率在  $\text{Exec}(\pi, A, Z; u)$  的样本空间上计算) 则  $\pi$  定义做具有 Dolev-Yao 刚性.

直观地说, 一个计算形式的密码协议  $\pi$  具有 Dolev-Yao 刚性, 是指任何 (无论主动或被动) 概率多项式复杂度的攻击者  $A$  对  $\pi$  的攻击, 其形式化几乎总是等价于某个 Dolev-Yao 攻击者  $\tilde{A}$  对  $\pi$  的攻击, 或者说,  $\pi$  的任何攻击者  $A$  对  $\pi$  所能做到的事情并不超出  $\tilde{\pi}$  的某个攻击者  $\tilde{A}$  能够对  $\pi$  所做事情的计算语义. 因此, Dolev-Yao 刚性意味着计算形式的攻击者的能力刚好属于形式模型所能够表达出来的那些类型, 对一个具有 Dolev-Yao 刚性的密码协议  $\pi$  的安全分析也就有可能有效转化为对形式模型  $\bar{\pi}$  的形式安全分析, 后者是纯粹的符号演算, 并且将形式分析的结论经过语义赋值后有可能重新解释为计算密码学意义上的分析结论. 这就是 Dolev-Yao 刚性概念的意义.

**定义 6** (Dolev-Yao 相似性).  $\pi_1, \pi_2$  是符号形式的密码协议, 若对任何 Dolev-Yao 攻击者  $\tilde{A}_2$  必存在 Dolev-Yao 攻击者  $\tilde{A}_1$  使  $\text{Trace}(\pi_2, \tilde{A}_2) \sqsubseteq_E \text{Trace}(\pi_1, \tilde{A}_1)$ , 则  $\pi_2$  定义为与  $\pi_1$  Dolev-Yao 相似, 记作  $\pi_2 \rightarrow^{\text{DY}} \pi_1$ .

Dolev-Yao 相似性纯粹是一个语法范畴的概念. 直观地说, 一个符号形式的密码协议  $\pi_2$  Dolev-Yao 相似于另一个符号形式的密码协议  $\pi_1$ , 是指  $\pi_2$  的任何 Dolev-Yao 攻击者对  $\pi_2$  在形式演绎的意义上所能做到的任何事情, 都存在  $\pi_1$  的某个攻击者对

$\pi_1$  做到同样的事情. 熟悉 UC-理论的读者会看出这一概念与 UC-相似性类似.

最后建立一个形式上稍强的概念, 称为 \*Dolev-Yao 相似性, 我们在第 5 节需要用到它.

**定义 7** (\*Dolev-Yao 相似性).  $\pi_1, \pi_2$  是符号形式的密码协议, 若存在多项式复杂度算法  $J^*$ , 对任何 Dolev-Yao 攻击者  $\tilde{A}_2$  必有 Dolev-Yao 攻击者  $\tilde{A}_1 = J^*(\pi_1, \text{Trace}(\pi_2, \tilde{A}_2))$  使  $\text{Trace}(\pi_2, \tilde{A}_2) \sqsubseteq_E \text{Trace}(\pi_1, \tilde{A}_1)$ , 则  $\pi_2$  定义为与  $\pi_1$  \*Dolev-Yao 相似, 记作  $\pi_2 \rightarrow^{*\text{DY}} \pi_1$ .

## 4 关于 Dolev-Yao 刚性的主要结论及证明

在这一节和下一节都仅考虑第 1 类主动攻击者 (参见 3.1 节).

### 4.1 Dolev-Yao 刚性的复合-稳定性定理

回顾 3.1 小节关于复合协议和 UC-细化的概念,  $\pi(\varphi)$  是基于子协议  $\varphi$  的复合协议意味着  $\pi$  调用  $\varphi$  且  $\pi$  仅在调用  $\varphi$  时对  $\varphi$  有输入关系及在  $\varphi$  运行完成时对  $\pi$  有输出关系, 而 UC-细化  $\pi(\psi/\varphi)$  意味着  $\pi$  对  $\psi$  的调用完全沿袭  $\pi$  对  $\varphi$  的调用关系, 只是调用的对象以  $\psi$  取代了  $\varphi$ . 在实际应用中往往  $\psi$  是一个较之  $\varphi$  更具有操作性的子协议, 而  $\varphi$  相对  $\psi$  而言是一个更具有说明性的抽象协议, 同时  $\pi$  对无论哪种子协议都扮演高层调度程序, 因此 UC-细化实际上是一个非常符合当代软件分层架构和自顶向下设计原理的概念. 一个自然的问题是: 如果较抽象或较理想的“高级”协议  $\pi(\varphi)$  具有 Dolev-Yao 刚性, 细化之后的“低级”协议  $\pi(\psi/\varphi)$  是否还保持 Dolev-Yao 刚性? 我们称之为 Dolev-Yao 刚性的复合-稳定性问题.

为精确表述和证明关于 Dolev-Yao 刚性的复合-稳定性性质, 需要适当限定所讨论密码协议的类型, 以作为进行论证的一个技术性条件.

**定义 8** (密码协议在子协议上的原子性).  $\pi$  是密码协议, 如果  $\pi$  在每次开始调用某个子协议 (例如  $\varphi$ ) 后其程序便完全进入该子协议的会话, 直到该子协议完成运行后才返回自己的会话; 不仅如此,  $\pi$  在子协议尚未完成之前拒绝参与任何其它会话, 并且总能准确识别出当前从外部到达的任何消息是否属于正确的会话, 如果属于不正确的会话则终止运行. 这样的协议  $\pi$  定义为在子协议  $\varphi$  上具有原子性.

设协议  $\pi$  在子协议  $\varphi$  上具有原子性, 如果  $\varphi$  还

是自独立的(参见定义 3, 实际上这是很普遍的情形<sup>[27]</sup>), 则不难看出子协议原子性的直接后果是复合协议  $\pi(\varphi)$  的攻击算法  $A$  的运行实例与  $\pi(\varphi)$  的相互作用过程总能依次划分为一系列连续的阶段  $\pi^{(1)}, \varphi^{(1)}, \pi^{(2)}, \varphi^{(2)}, \dots, \varphi^{(n-1)}, \pi^{(n)}$ , 在阶段  $\pi^{(i)}$  上  $A$  实际上仅与  $\pi$  作用(符号  $\pi^{(i)}$  既是阶段的名称也代表  $\pi(\varphi)$  在这一阶段上的运行过程), 而在阶段  $\varphi^{(i)}$  上  $A$  只与子协议  $\varphi$  的一个完整实例(第  $i$  个实例)相互作用(由于子协议原子性,  $A$  不可能成功地诱骗  $\pi(\varphi)$  使一个处于  $\pi^{(i)}$  或  $\varphi^{(i)}$  阶段的进程与另一个处于非  $\pi^{(i)}$  或非  $\varphi^{(i)}$  阶段的进程相互作用). 于是  $A$  的运行实例等价于一系列攻击算法的实例  $A^{(1)}, A^{(2)}, A^{(3)}, A^{(4)}, \dots$  的串行复合, 其中  $A^{(1)}, A^{(3)}, \dots$  分别在阶段  $\pi^{(1)}, \pi^{(2)}, \dots$  上攻击  $\pi$ , 而  $A^{(2)}, A^{(4)}, \dots$  分别在阶段  $\varphi^{(1)}, \varphi^{(2)}, \dots$  上攻击  $\varphi$  的各个实例.

定理 3 将针对这类协议解决问题. 显然, 所谓  $\pi$  在  $\varphi$  上的子协议原子性本质上取决于  $\pi$  对子协议  $\varphi$  的调用方式而与子协议本身无关, 特别是, 若  $\pi$  在  $\varphi$  上有子协议原子性则  $\pi(\psi/\varphi)$  在  $\psi$  上也具有子协议原子性. 进一步不难看出子协议原子性是一个非常自然而常见的性质: 一方面, 很多协议具有固有的子协议原子性, 例如当  $\varphi$  是非交互式密码方案如加密、数字签名方案时, 任何协议  $\pi$  在这类  $\varphi$  上都显然具有子协议原子性; 另一方面, 如果  $\pi$  串行调用  $\varphi$  的多个实例并且每次都是在该实例完成之后才继续执行(事实上绝大多数密码协议都以这种方式调用子协议, 例如许多协议对零知识证明子协议的调用), 则不难加入一些附加机制保证  $\pi$  在  $\varphi$  上具有子协议原子性, 例如  $\pi$  在每次调用  $\varphi$  之前生成特定的会话号, 子协议  $\varphi$  在运行期间始终遵循这一会话号并且会话各方在每条消息中都附加对该会话号(及其它关联信息)的数字签名, 签名验证失败则终止协议运行. 总之, 作为一种技术性条件, 子协议原子性能够合理涵盖很大一部分协议.

**定理 3**(刚性的复合-稳定性质). 设  $\pi(\varphi)$  是基于子协议  $\varphi$  的复合协议且  $\pi$  在  $\varphi$  上具有子协议原子性, 协议  $\varphi$  和协议  $\psi$  都是自独立的. 若  $\pi(\varphi)$  和  $\psi$  都具有 Dolev-Yao 刚性, 则  $\pi(\psi/\varphi)$  也具有 Dolev-Yao 刚性.

证明. 以下将  $\pi(\psi/\varphi)$  简记为  $\pi(\psi)$ .  $A$  是  $\pi(\psi)$  的 P.P.T. 攻击算法, 以下  $A$  也表示其任何一个运行剖面, 即各个随机因素都取特定样本值时  $A$  的运行实例. 因为  $\pi$  具有子协议原子性, 如前所述,  $A$  与  $\pi(\psi)$  的相互作用能依次划分为一系列连续的阶段

$\pi^{(1)}, \psi^{(1)}, \pi^{(2)}, \psi^{(2)}, \dots, \psi^{(n-1)}, \pi^{(n)}$ , 在阶段  $\pi^{(i)}$  上  $A$  仅与  $\pi$  作用, 以下也用  $\pi^{(i)}$  表示  $\pi(\psi)$  (在这阶段上也就是  $\pi$ ) 在这一阶段上的运行过程, 而在阶段  $\psi^{(i)}$  上  $A$  与子协议  $\psi$  的一个完整实例(第  $i$  个实例)相互作用. 于是  $A$  等价于一系列攻击算法的实例  $A^{(1)}, A^{(2)}, A^{(3)}, A^{(4)}, \dots$  的串行复合, 其中  $A^{(1)}, A^{(3)}, \dots$  分别在阶段  $\pi^{(1)}, \pi^{(2)}, \dots$  上攻击  $\pi$ ,  $A^{(2)}, A^{(4)}, \dots$  分别在阶段  $\psi^{(1)}, \psi^{(2)}, \dots$  上攻击  $\psi$  的各个实例且这些  $\psi$ -实例相互独立.

设  $e_1, e_3, e_5, \dots$  分别是阶段  $\pi^{(1)}, \pi^{(2)}, \pi^{(3)}, \dots$  的最后一个事件,  $e_2, e_4, e_6, \dots$  分别是阶段  $\psi^{(1)}, \psi^{(2)}, \psi^{(3)}, \dots$  的最后一个事件. 记  $\tilde{\psi} = \text{symb}_P(\psi)$ ,  $\tilde{\pi}^{(i)} = \text{symb}_P(\pi^{(i)})$ ,  $\tilde{A}^{(i)}$  表示 Dolev-Yao 攻击者 strand,  $i = 1, 2, 3, \dots, u$  表示外部输入,  $\tilde{u} = \text{symb}_T(u) \cup K_A$ ,  $K_A = \{\tilde{x} = \text{symb}_T(x) : x \text{ 是 } A \text{ 决定入侵的对象的秘密参数(注意我们假定了攻击算法 } A \text{ 仅有非适应性入侵能力)}\}$ .

对任何环境  $Z$  做环境  $Z_1$ ,  $Z_1$  与  $Z$  相同但当接收到事件  $e_1$  时强迫  $\pi(\psi)$  终止, 于是有

$$\begin{aligned} \text{Exec}(\pi^{(1)}, A^{(1)}, Z; u) &= \text{Exec}(\pi(\psi), A^{(1)}, Z_1; u) \\ &= \text{Exec}(\pi(\varphi), A^{(1)}, Z_1; u), \end{aligned}$$

第 1 个等式是因为在环境  $Z_1$  之下  $\pi(\psi)$  仅运行于阶段  $\pi^{(1)}$ , 第 2 个等式是因为在阶段  $\pi^{(1)}$  上  $\pi(\psi) = \pi(\varphi)$ . 设  $\xi_1$  是  $A^{(1)}$  在这一阶段上与  $\pi^{(1)}$  相互作用所交换的所有消息,  $St_1$  是  $A^{(1)}(\xi_1, u)$  的输出. 由上式及  $\pi(\varphi)$  具有 Dolev-Yao 刚性, 故存在复杂性参数  $k$  的可忽略函数  $\delta_\pi(k)$  使

$$\begin{aligned} P[\tau \in \text{Exec}(\pi^{(1)}, A^{(1)}, Z; u) : \text{存在 } \tilde{A}^{(1)} \text{ 使} \\ \text{symb}_T(\tau) = \text{Trace}(\tilde{\pi}^{(1)}, \tilde{A}^{(1)})] > 1 - \delta_\pi(k), \end{aligned}$$

$\tilde{A}^{(1)}$  是 Dolev-Yao 攻击者 strand 意味着输出  $\tilde{St}_1 = \text{symb}_T(St_1)$  必可从其初始知识集合  $\tilde{u}$  通过 Dolev-Yao 形式演绎规则导出, 因此上式蕴涵  $P[\tau \in \text{Exec}(\pi^{(1)}, A^{(1)}, Z; u) : \tilde{u} \vdash_E \tilde{St}_1] > 1 - \delta_\pi(k)$ . 于是除去至多为  $\delta_\pi(k)$  的概率之外可以做出  $\tilde{\pi}^{(1)}$  和  $\tilde{A}^{(1)}$  相互作用的 strand-图, 该图中代表  $\tilde{A}^{(1)}$  的攻击者 strand 起始于负的消息作用点  $-\tilde{u}$  而终止于正的消息作用点  $+\tilde{St}_1$ , 同时其代表  $\tilde{\pi}^{(1)}$  的合法 strand 起始于初始输入事件而终止于事件  $e_1$ .

令环境  $Z_2$  是  $Z$  与  $A^{(1)}$  的复合, 当接收到事件  $e_2$  时强迫  $\pi(\psi)$  终止. 由  $\psi$  的 Dolev-Yao 刚性, 存在复杂性参数  $k$  的可忽略函数  $\delta_\psi(k)$  使

$$\begin{aligned} P[\tau \in \text{Exec}(\psi^{(1)}, A^{(2)}, Z_2; u) : \text{存在 } \tilde{A}^{(2)} \text{ 使} \\ \text{symb}_T(\tau) = \text{Trace}(\tilde{\psi}, \tilde{A}^{(2)})] > 1 - \delta_\psi(k), \end{aligned}$$

因为  $A^{(2)}$  与  $\psi$  的实例相互作用而  $\varphi$  和  $\psi$  都是自独立

的,故在符号形式上等价于考虑  $\tilde{A}^{(2)}$  与  $\tilde{\psi}$  的相互作用. 设在阶段  $\psi^{(1)}$  上  $A^{(2)}$  所交换的所有消息为  $\eta_1$  且显然  $A^{(2)}$  以  $A^{(1)}$  的输出  $St_1$  为输入,故  $A^{(2)}$  的输出  $St_2 = A^{(2)}(St_1, \eta_1)$ . 令  $\tilde{St}_2 = \text{symp}_T(St_2)$ , 基于与上一段相同的理由,  $P[\tau \in \text{Exec}(\psi^{(1)}, A^{(2)}, Z; u) : \tilde{St}_1 \models_E \tilde{St}_2] > 1 - \delta_\psi(k)$ . 于是除去至多为  $\delta_\psi(k)$  的概率之外可以做出  $\tilde{\psi}$  和  $\tilde{A}^{(2)}$  相互作用的 strand-图, 该图中代表  $\tilde{A}^{(2)}$  的攻击者 strand 起始于负的消息作用点  $-\tilde{St}_1$  而终止于正的消息作用点  $+\tilde{St}_2$ , 同时其代表  $\tilde{\psi}$  的合法 strand 起始于事件  $e_1$  而终止于事件  $e_2$ .

将 strand-图  $(\pi^{(1)}, \tilde{A}^{(1)})$  和  $(\tilde{\psi}, \tilde{A}^{(2)})$  中的合法 strand 在事件  $e_1$  上粘合起来、同时将攻击者 strand 在消息作用点  $\pm \tilde{St}_1$  上粘合起来, 就得到这样一个 strand-图, 其中的合法 strand 代表  $\pi(\psi)$  在阶段  $\pi^{(1)}, \psi^{(1)}$  上运行并终止于事件  $e_2$ , 攻击者 strand 则代表对这一过程进行的一个 Dolev-Yao 攻击.

如此继续下去, 在每个阶段上除去至多为复杂性参数  $k$  的一个可忽略函数的概率之外总可以做出一个 strand-图, 这些 strand-图中的合法 strand 的起始事件恰是前一阶段的合法 strand 的终止事件, 攻击者 strand 的起始点(负号)也恰是前一阶段的 strand-图中攻击者 strand 的终止点(正号). 将各阶段所对应的这些 strand-图中的合法 strand 依次在相同的事件点上、攻击者 strand 则在(除正负号之外)相同的消息作用点上粘合起来, 仍然得到一个 strand 图. 这一图中完整的合法 strand 就是由所有各阶段上的合法 strand 粘合而成, 并且恰代表着协议  $\pi(\psi)$  的一个运行实例, 而完整的攻击者 strand 由所有各阶段上的攻击者 strand 粘合而成, 代表对  $\pi(\psi)$  的一个运行实例, 记作  $\tilde{A}$ . 由于各个攻击者 strand 都是 Dolev-Yao 的, 显然这样构造的  $\tilde{A}$  也是 Dolev-Yao strand.

现在估计 Dolev-Yao 刚性定义中的概率函数. 对任意的  $A, Z$  和  $u$  以及  $\tau \in \text{Exec}(\pi(\psi), A, Z; u)$ , 设  $\tau = \tau(1) \parallel \tau(2) \parallel \tau(3) \parallel \tau(4) \parallel \dots$ , 其中  $\tau(i) \in \text{Exec}(\pi^{(i)}, A^{(i)}, Z_i; u), i=1, 3, 5, \dots, \tau(i) \in \text{Exec}(\psi^{(i)}, A^{(i)}, Z_i; u), i=2, 4, 6, \dots$ , 于是

$$\text{symp}_T(\tau) = \text{symp}_T(\tau(1)) \parallel \text{symp}_T(\tau(2)) \parallel \text{symp}_T(\tau(3)) \parallel \text{symp}_T(\tau(4)) \parallel \dots$$

记  $\tilde{\theta}^{(i)} \equiv \pi^{(i)}$ , 若  $i=1, 3, 5, \dots$ ;  $\tilde{\theta}^{(i)} \equiv \tilde{\psi}^{(i)}$ ; 若  $i=2, 4, 6, \dots$ , 定义概率事件

$\Gamma \equiv$  “对每个  $i$  都存在 Dolev-Yao 攻击者 strand  $\tilde{A}^{(i)}$  使  $\text{symp}_T(\tau(i)) = \text{Trace}(\tilde{\theta}^{(i)}, \tilde{A}^{(i)})$ ”,

显然条件  $\Gamma$  意味着

$$\begin{aligned} \text{symp}_T(\tau) &= \text{symp}_T(\tau(1)) \parallel \text{symp}_T(\tau(2)) \parallel \text{symp}_T(\tau(3)) \parallel \text{symp}_T(\tau(4)) \parallel \dots \\ &= \text{Trace}(\pi^{(1)}, \tilde{A}^{(1)}) \parallel \text{Trace}(\tilde{\psi}^{(1)}, \tilde{A}^{(2)}) \parallel \text{Trace}(\pi^{(2)}, \tilde{A}^{(3)}) \parallel \text{Trace}(\tilde{\psi}^{(2)}, \tilde{A}^{(4)}) \parallel \dots \\ &= \text{Trace}(\text{symp}_P(\pi(\psi)), \tilde{A}). \end{aligned}$$

不难直接验证最后一个等式源于  $\pi$  的子协议原子性和  $\text{symp}_P(\cdot)$  的构造. 设  $M(k)$  和  $N(k)$  分别是  $\pi^{(i)}$  阶段和  $\psi^{(i)}$  阶段数目的上界, 显然两者都是  $k$  的多项式函数, 于是

$$\begin{aligned} P[\tau \in \text{Exec}(\pi(\psi), A, Z; u) : \text{存在 } \tilde{A} \text{ 使} \\ \text{symp}_T(\tau) = \text{Trace}(\text{symp}_P(\pi(\psi)), \tilde{A})] &\geq P[\Gamma] \\ &\geq \prod_{i=1, 3, \dots} P[\tau(i) \in \text{Exec}(\pi^{(i)}, A^{(i)}, Z_i; u) : \\ &\quad \text{symp}_T(\tau(i)) = \text{Trace}(\pi^{(i)}, \tilde{A}^{(i)})] \times \\ &\quad \prod_{i=2, 4, \dots} P[\tau(i) \in \text{Exec}(\psi^{(i)}, A^{(i)}, Z_i; u) : \\ &\quad \text{symp}_T(\tau(i)) = \text{Trace}(\tilde{\psi}^{(i)}, \tilde{A}^{(i)})] \\ &\geq (1 - \delta_\pi(k))^{M(k)} (1 - \delta_\psi(k))^{N(k)} \\ &\geq (1 - M(k)\delta_\pi(k))(1 - N(k)\delta_\psi(k)) \\ &\geq 1 - \delta(k), \end{aligned}$$

其中  $\delta(k) = M(k)\delta_\pi(k) + N(k)\delta_\psi(k)$ . 因为  $\delta_\pi(k)$  和  $\delta_\psi(k)$  都是  $k$  的可忽略函数, 而  $M(k)$  和  $N(k)$  都是  $k$  的多项式函数, 因此  $\delta(k)$  是  $k$  的可忽略函数. 证毕.

定理 3 有一个有用但很容易证明的推论.

**定理 4.**  $\pi(\varphi_1, \dots, \varphi_n)$  是基于子协议  $\varphi_1, \dots, \varphi_n$  的复合协议且  $\pi$  在  $\varphi_1, \dots, \varphi_n$  上都具有子协议原子性, 子协议  $\varphi_1, \dots, \varphi_n$  和子协议  $\psi_1, \dots, \psi_n$  都是自独立的且相互独立. 若  $\pi(\varphi_1, \dots, \varphi_n)$  和  $\psi_1, \dots, \psi_n$  都具有 Dolev-Yao 刚性, 则  $\pi(\psi_1/\varphi_1, \dots, \psi_n/\varphi_n)$  也具有 Dolev-Yao 刚性.

证明. 从  $\pi(\varphi_1, \dots, \varphi_n)$  和  $\psi_1$  的 Dolev-Yao 刚性及子协议的独立性出发, 运用定理 3 相同的论证即可证明  $\pi_1(\varphi_2, \dots, \varphi_n) \equiv \pi(\psi_1/\varphi_1, \varphi_2, \dots, \varphi_n)$  具有 Dolev-Yao 刚性. 显然  $\pi_1(\varphi_2, \dots, \varphi_n)$  也满足定理 3 的所有条件, 因此同理证明  $\pi_2(\varphi_3, \dots, \varphi_n) \equiv \pi_1(\psi_2/\varphi_2, \varphi_3, \dots, \varphi_n)$  具有 Dolev-Yao 刚性. 如此继续下去直到  $\pi_n \equiv \pi_{n-1}(\psi_n/\varphi_n) = \pi(\psi_1/\varphi_1, \dots, \psi_n/\varphi_n)$ . 证毕.

利用定理 3、定理 4 能立刻证明相当大范围的密码协议对第 1 类主动攻击者都具有 Dolev-Yao 刚性. 实际上, 首先注意到不含任何密码方案的协议  $\pi$  显然对任何主动攻击者都具有 Dolev-Yao 刚性; 同时注意到典型的(非交互式)密码方案如对称/非对称加密、数字签名等的理想 UC-模型都具有 Dolev-Yao 刚性(这一点从 UC-理想模型本身最容易看清楚, 参见文献[22-23, 27]), 而且  $\pi$  在这些非交互式

方案上显然都是原子性的(参见本小节开头一段), 因此从  $\pi$  的 Dolev-Yao 刚性出发应用定理 4 便得出基于任意多个这类 UC-理想密码方案的复合协议都具有 Dolev-Yao 刚性. 再运用下一小节将要证明的刚性的相似-遗传定理, 便能得出结论: 基于任意多个这类(并非 UC-理想)密码方案的复合协议都具有 Dolev-Yao 刚性, 其中加密方案要求具有 IND\_CCA 保密性、数字签名方案要求具有 UF\_CMA 抗伪造性质等.

#### 4.2 Dolev-Yao 刚性的相似-遗传定理

这一小节将 Dolev-Yao 刚性概念与强有力的 UC-相似性概念联系起来, 证明相似-遗传性定理, 并且借此导出刚性的复合-稳定性定理的一个略微不同的形式.

**定理 5**(刚性的相似-遗传性质).  $\pi_2, \pi_1$  是两个(计算形式的)密码协议且  $\pi_2 \rightarrow^{\text{UC}} \pi_1$ . 若  $\pi_1$  具有 Dolev-Yao 刚性且表达形式协议模型的消息代数上的  $\models_E$  问题多项式可解, 则  $\pi_2$  也具有 Dolev-Yao 刚性.

证明. 假设不然, 即存在 P.P.T. 算法  $A^*$  和  $Z^*$ 、输入  $u$  和复杂性参数  $k$  的多项式  $p(k)$  使得在无穷多个  $k$  上都有

$P[\tau \in \text{Exec}(\pi_2, A^*, Z^*; u): \text{不存在 Dolev-Yao 攻击者 strand } \tilde{A} \text{ 使 } \text{symp}_T(\tau) = \text{Trace}(\tilde{\pi}_2, \tilde{A})] > 1/p(k)$ . 从 3.4 小节关于  $\text{symp}_T()$  的构造及 3.2~3.3 小节关于 strand-图的阐述不难看出, 若存在 Dolev-Yao 攻击者 strand  $\tilde{A}$  使  $\text{symp}_T(\tau) = \text{Trace}(\tilde{\pi}, \tilde{A})$ , 则必有  $\text{symp}_T(\tau) \neq \perp$ , 故  $P[\tau \in \text{Exec}(\pi_2, A^*, Z^*; u): \text{symp}_T(\tau) = \perp] > 1/p(k)$  在无穷多个  $k$  上成立.

做环境  $Z_0; Z_0$  调用  $Z^*$ , 对  $\tau \in \text{Exec}(\pi, A, Z^*; u)$  ( $\pi$  是  $\pi_1$  或  $\pi_2, A, u$  任意) 计算出  $\tilde{\tau} = \text{symp}_T(\tau)$ , 若  $\tilde{\tau} \neq \perp$ , 则输出 1; 否则输出 0. 由于  $\models_E$  问题多项式可解, 因此  $\text{symp}_T()$ 、从而  $Z_0$  是 P.P.T. 算法.

对任何攻击算法  $A_1$ , 环境  $Z_0$  能将  $\pi_2, A^*$  之间的相互作用与  $\pi_1, A_1$  之间的相互作用有效区分开. 事实上, 显然存在无穷多个  $k$  使  $P[Z_0^{\text{exec}}(\pi_2, A^*, u) = 0] > 1/p(k)$ ; 另一方面, 由于  $\pi_1$  具有 Dolev-Yao 刚性, 因此对  $u, Z_0$  和任何攻击算法  $A_1$  存在  $k$  的可忽略函数  $\epsilon(k)$  使得

$P[\tau \in \text{Exec}(\pi_1, A_1, Z_0; u): \text{存在 Dolev-Yao 攻击者 } \tilde{A}_1 \text{ 使 } \text{symp}_T(\tau) = \text{Trace}(\tilde{\pi}_1, \tilde{A}_1)] > 1 - \epsilon(k)$ , 从而  $P[Z_0^{\text{exec}}(\pi_1, A_1, u) = 1] > 1 - \epsilon(k)$ , 即  $P[Z_0^{\text{exec}}(\pi_1, A_1, u) = 0] < \epsilon(k)$ , 特别是对充分大的  $k$  有  $P[Z_0^{\text{exec}}(\pi_1,$

$A_1, u) = 0] < 1/2p(k)$ . 于是对无穷多个  $k$  有

$$\begin{aligned} \delta_{A.S.}^{\psi, \varphi}(k) &\equiv |P[Z_0^{\text{exec}}(\pi_2, A^*, u) = 1] - \\ &\quad P[Z_0^{\text{exec}}(\pi_1, A_1, u) = 1]| \\ &= |P[Z_0^{\text{exec}}(\pi_2, A^*, u) = 0] - \\ &\quad P[Z_0^{\text{exec}}(\pi_1, A_1, u) = 0]| \\ &> 1/p(k) - 1/2p(k) = 1/2p(k), \end{aligned}$$

与条件  $\pi_2 \rightarrow^{\text{UC}} \pi_1$  相矛盾.

证毕.

以上定理要求  $\models_E$  多项式可解. 虽然一般来说甚至存在  $\models_E$  不可解的消息代数, 但文献[19]已经证明对很大一类非自由消息代数, 即其中的约束集合  $E$  是所谓收敛的子项理论形式,  $\models_E$  问题不仅可解而且多项式复杂度可解(定理 3.4). 文献[19-20]的许多实例充分表明这一情形能涵盖实际应用中的很大一部分情形, 例如含(任意多项)对称或非对称加密算子、数字签名算子、群算术或  $GF(2)$  算术的非自由代数, 因此  $\models_E$  多项式可解是一个足够广泛的充分条件.

以上定理表明 UC-相似保持刚性. 从 UC-理论的观点, 证明一个密码协议  $\pi$  具有某种安全性质本质上就是要证明其与某个具有理想安全性质的协议  $\pi^*$  UC-相似, 而  $\pi^*$  往往由其定义就很容易验证具有刚性(例如文献[27]中的大量实例, 特别是非交互式密码方案  $F_{\text{PKE}}, F_{\text{SIG}}$  等), 由此能立刻得出很多类型的协议(或方案)  $\pi$  具有刚性, 特别是 IND\_CCA 保密的加密方案、UF\_CMA 抗伪造的数字签名方案等, 这样我们就从定理 5 再次推出文献[18, 22-23]中的结论, 但这里的概念更明确、方法更普遍. 最后, 定理 5 有以下有用的推论.

**定理 6.**  $\pi(\varphi)$  具有 Dolev-Yao 刚性,  $\psi \rightarrow^{\text{UC}} \varphi$ , 协议  $\varphi$  和协议  $\psi$  都是自独立的、 $\varphi$  和  $\psi$  之间也相互独立. 设表达形式协议模型的消息代数上的  $\models_E$  问题多项式可解, 则  $\pi(\psi/\varphi)$  也具有 Dolev-Yao 刚性.

证明. 在定理的条件下由定理 1 有  $\pi(\psi) \rightarrow^{\text{UC}} \pi(\varphi)$ , 再由定理 5 立得  $\pi(\psi/\varphi)$  也具有 Dolev-Yao 刚性.

证毕.

## 5 语法-语义相似性的对偶关系

从本质上看, Dolev-Yao 相似性概念与 UC-相似性概念分别从形式语法和计算语义两个范畴表达协议之间的关系. UC-相似性是一个强有力的概念, 在密码协议的安全分析和证明中起着实质性的作用. 这一节将 Dolev-Yao 相似性与 UC-相似性联系

起来,证明两者之间近乎充分必要程度的对偶关系.

## 5.1 主要结论

**定理 7.**  $\pi_2, \pi_1$  是两个(计算形式的)密码协议且  $\pi_2 \xrightarrow{\text{UC}} \pi_1$ ,  $\tilde{\pi}_i = \text{symp}_P(\pi_i)$ ,  $i = 1, 2$ . 若  $\pi_1$  具有 Dolev-Yao 刚性、形式协议模型的消息代数上的  $\models_E$  问题和  $\cong_E$  问题均多项式可解, 则  $\tilde{\pi}_2 \xrightarrow{\text{DY}} \tilde{\pi}_1$ .

证明. 假若不然, 即存在 Dolev-Yao 攻击者  $\tilde{A}_2$ , 对任何 Dolev-Yao 攻击者  $\tilde{A}_1$  都没有  $\text{Trace}(\tilde{\pi}_2, \tilde{A}_2) \cong_E \text{Trace}(\tilde{\pi}_1, \tilde{A}_1)$ . 按照第 3.4 小节指出的方法构造  $\tilde{A}_2$  的计算语义, 得 P.P.T. 攻击算法  $A_2$ , 以下将证明存在 P.P.T. 环境  $Z^*$ , 对任何 P.P.T. 攻击算法  $A_1$  和输入  $u$  使

$$|P[Z^* \text{exec}(\pi_2, A_2, u) = 1] -$$

$$P[Z^* \text{exec}(\pi_1, A_1, u) = 1]| > 1 - \delta(k),$$

其中  $\delta(k)$  是复杂性参数  $k$  的可忽略函数, 从而与条件  $\pi_2 \xrightarrow{\text{UC}} \pi_1$  相矛盾.

记  $\tilde{\tau}_2 \equiv \text{Trace}(\tilde{\pi}_2, \tilde{A}_2)$ .  $Z^*$  构造如下: 对任何  $\tau \in \text{Exec}(\pi, A, Z^*; u)$  ( $\pi$  是  $\pi_1$  或  $\pi_2$ ,  $A, u$  任意),  $Z^*$  (在所有相互作用完成之后) 计算出  $\tilde{\tau} = \text{symp}_T(\tau)$ , 若  $\tilde{\tau} \cong_E \tilde{\tau}_2$ , 则输出 1; 否则输出 0. 由于  $\models_E$  问题多项式可解故  $\text{symp}_T$  是 P.P.T. 算法; 又  $\cong_E$  问题多项式可解, 故  $Z^*$  是 P.P.T. 算法.

显然  $P[Z^* \text{exec}(\pi_2, A_2, u) = 1] = 1$ . 另一方面, 由于  $\pi_1$  具有 Dolev-Yao 刚性因此对任何 P.P.T. 攻击算法  $A_1$  和输入  $u$  有可忽略函数  $\delta(k)$  使得

$P[\tau \in \text{Exec}(\pi_1, A_1, Z^*; u) : \text{存在 Dolev-Yao 攻击者 } \tilde{A}_1 \text{ 使 } \text{symp}_T(\tau) = \text{Trace}(\tilde{\pi}_1, \tilde{A}_1)] > 1 - \delta(k)$ , 进而由  $\tilde{A}_2$  和  $\tilde{\tau}_2$  的性质有

$$\begin{aligned} &P[Z^* \text{exec}(\pi_1, A_1, u) = 1] \\ &= P[\tau \in \text{Exec}(\pi_1, A_1, Z^*; u) : \text{symp}_T(\tau) \cong_E \tilde{\tau}_2] \\ &\leq P[\tau \in \text{Exec}(\pi_1, A_1, Z^*; u) : \text{不存在 Dolev-Yao 攻击者 } \tilde{A}_1 \text{ 使 } \text{symp}_T(\tau) = \text{Trace}(\tilde{\pi}_1, \tilde{A}_1)] \\ &< \delta(k), \end{aligned}$$

从而  $P[Z^* \text{exec}(\pi_2, A_2, u) = 1] - P[Z^* \text{exec}(\pi_1, A_1, u) = 1] > 1 - \delta(k)$ . 证毕.

回顾第 3.3 小节对形式表达式的一般性的语义赋值构造, 下面的语义赋值都指这一语义.  $\cong_E$  关系、 $\cong_E$  相容性和 \*Dolev-Yao 相似性概念参见 3.3 和 3.5 小节.

**定理 8.**  $\pi_2, \pi_1$  是两个(计算形式的)密码协议,  $\tilde{\pi}_i = \text{symp}_P(\pi_i)$ ,  $i = 1, 2$ . 若  $\pi_2$  具有 Dolev-Yao 刚性、 $\tilde{\pi}_2 \xrightarrow{* \text{DY}} \tilde{\pi}_1$ 、形式协议模型的消息代数上的  $\cong_E$  问题多项式可解且其语义赋值是  $\cong_E$  相容的, 则必有

$$\pi_2 \xrightarrow{\text{UC}} \pi_1.$$

证明. 设  $Z$  是 P.P.T. 环境、 $u$  是  $Z$  的输入,  $A_2$  是对  $\pi_2$  的 P.P.T. 攻击算法,  $R_2$  是  $A_2$  的内部随机因素样本的生成算法. 既然  $\tilde{\pi}_2 \xrightarrow{* \text{DY}} \tilde{\pi}_1$ , 设  $J^*$  是定义 7 中所出现的算法.

构造  $\pi_1$  的攻击算法  $A_1$ ,  $A_1$  由如下步骤组成: 第 1 步, 调用  $R_2(k)$  生成  $r$ 、计算  $\tau \leftarrow \text{Exec}(\pi_2, A_2(r), Z; u)$ . 第 2 步, 调用  $J^*$  计算  $\tilde{A}_1(r) \leftarrow J^*(\tilde{\pi}_1, \text{symp}_T(\tau))$ . 由于  $\pi_2$  有刚性, 除去不超过某个可忽略函数  $\delta(k)$  的概率之外必有  $\tilde{\pi}_2$  的 Dolev-Yao 攻击者  $\tilde{A}_2(r)$  使  $\text{symp}_T(\tau) = \text{Trace}(\tilde{\pi}_2, \tilde{A}_2(r))$ , 从而  $\tilde{A}_1(r) = J^*(\tilde{\pi}_1, \text{Trace}(\tilde{\pi}_2, \tilde{A}_2(r)))$ , 再由  $J^*$  的定义 ( $\tilde{\pi}_2 \xrightarrow{* \text{DY}} \tilde{\pi}_1$ ) 使得  $\tilde{A}_1(r)$  是  $\tilde{\pi}_1$  的 Dolev-Yao 攻击者并且

$$P[r \leftarrow R_2(k) : \text{Trace}(\tilde{\pi}_2, \tilde{A}_2(r)) \cong_E \text{Trace}(\tilde{\pi}_1, \tilde{A}_1(r))] \geq P[r \leftarrow R_2(k) : \text{symp}_T(\text{Exec}(\pi_2, A_2(r), Z; u)) = \text{Trace}(\tilde{\pi}_2, \tilde{A}_2(r))] > 1 - \delta(k) \quad (1)$$

将  $\text{Trace}(\tilde{\pi}_2, \tilde{A}_2(r))$  和  $\text{Trace}(\tilde{\pi}_1, \tilde{A}_1(r))$  中由环境的输入/输出事件构成的观测式(定义参见 3.3 节)分别记作  $\varphi_2 \equiv \nu \bar{a}_2. \{z_0^{(2)}/y_0, z_1^{(2)}/y_1, \dots, z_{N_2}^{(2)}/y_{N_2}\}$  和  $\varphi_1 \equiv \nu \bar{a}_1. \{z_0^{(1)}/y_0, z_1^{(1)}/y_1, \dots, z_{N_1}^{(1)}/y_{N_1}\}$ . 对观测式中的项数做数学归纳, 容易证明下面的辅助性结果.

**引理 1.**  $\text{Trace}(\tilde{\pi}_2, \tilde{A}_2(r)) \cong_E \text{Trace}(\tilde{\pi}_1, \tilde{A}_1(r))$  且语义赋值  $\cong_E$  相容, 则  $N_1 = N_2$  且  $z_i^{(2)} = z_i^{(1)}, i = 0, 1, \dots, N_1$ .

将 strand-图  $(\tilde{\pi}_2, \tilde{A}_2(r))$ ,  $(\tilde{\pi}_1, \tilde{A}_1(r))$  分别记为  $G_2, G_1$ , 其输出观测式  $\varphi_2 \equiv \text{Trace}(G_2) = \nu \bar{a}_2. \{z_0^{(2)}/y_0\} \omega_1^{(2)} \{z_1^{(2)}/y_1\} \omega_2^{(2)} \{z_2^{(2)}/y_2\} \dots \omega_N^{(2)} \{z_N^{(2)}/y_N\} \omega_{N+1}^{(2)}$ ,  $\varphi_1 \equiv \text{Trace}(G_1) = \nu \bar{a}_1. \{z_0^{(1)}/y_0\} \omega_1^{(1)} \{z_1^{(1)}/y_1\} \omega_2^{(1)} \{z_2^{(1)}/y_2\} \dots \omega_N^{(1)} \{z_N^{(1)}/y_N\} \omega_{N+1}^{(1)}$ , 其中  $z_i^{(2)}, z_i^{(1)}$  是前面  $\varphi_2, \varphi_1$  中的各项, 分别表达 strand-图  $G_2, G_1$  中环境的全部输入/输出事件,  $\omega_i^{(2)}, \omega_i^{(1)}$  是除这些项以外的项构成的子观测式, 并设这些子式中的变元符号是  $x_i, i = 1, \dots, N+1$  (一般地说,  $x_i$  是矢量符号但这里简记作标量). 将  $G_1$  中表达环境行为的子图用  $G_2$  中表达环境  $Z$  的子图加以替换, 得  $G_1^*$ , 由引理 1 知  $G_1^*$  是 strand-图. 显然,

$$\begin{aligned} \varphi_1^* &\equiv \text{Trace}(G_1^*) \\ &= \nu \bar{a}_1. \{z_0^{(2)}/y_0\} \omega_1^{(1)} \{z_1^{(2)}/y_1\} \omega_2^{(1)} \{z_2^{(2)}/y_2\} \dots \\ &\quad \omega_N^{(1)} \{z_N^{(2)}/y_N\} \omega_{N+1}^{(1)}. \end{aligned}$$

我们证明  $\varphi_1^* \cong_E \varphi_1$ . 首先注意到  $\varphi_2 \cong_E \varphi_1$  (即  $\text{Trace}(\tilde{\pi}_2, \tilde{A}_2(r)) \cong_E \text{Trace}(\tilde{\pi}_1, \tilde{A}_1(r))$ ) 蕴涵  $\omega_i^{(2)} \cong_E \omega_i^{(1)}, i = 1, \dots, N+1$  (事实上, 对任何项  $M$

和  $N, f_v(M)$ ,  $f_v(N) \subseteq \text{dom}(\omega_i^{(2)}) \cup \text{dom}(\omega_i^{(1)})$ ,  $\omega_i^{(2)}(M) =_E \omega_i^{(2)}(N)$  显然意味着  $\phi_2(M) =_E \phi_2(N)$  从而  $\phi_1(M) =_E \phi_1(N)$ , 再由  $f_v(M), f_v(N)$  即知  $\omega_i^{(1)}(M) =_E \omega_i^{(1)}(N)$ ; 反之亦然, 因此  $\omega_i^{(2)} \sqsubseteq_E \omega_i^{(1)}$ . 考虑消息式  $M$  和  $N$ :  $\phi_1^*(M) =_E \phi_1^*(N)$ , 不失一般性  $f_v(M) = f_v(N) = \{y_0, \dots, y_N, x_1, \dots, x_{N+1}\}$ , 于是由  $z_i^{(2)} =_E z_i^{(1)}$  (引理 1) 和  $\omega_j^{(2)} \sqsubseteq_E \omega_j^{(1)}$  ( $i=0, 1, \dots, N, j=1, \dots, N+1$ ) 有  $\phi_1(M) =_E \phi_1(N)$ ; 反之, 由  $\phi_1(M) =_E \phi_1(N)$  同理证明  $\phi_1^*(M) =_E \phi_1^*(N)$ , 这就证明了  $\phi_1^* \sqsubseteq_E \phi_1$ , 即  $\text{Trace}(G_1^*) \sqsubseteq_E \text{Trace}(G_1)$ .

$A_1$  的第 3 步是按照 strand-图  $G_1^*$  中攻击者子图的语义赋值(第 3.4 小节)实施计算. 由上面的论断  $\text{Trace}(G_1^*) \sqsubseteq_E \text{Trace}(G_1)$  知  $\text{Exec}(\pi_1, A_1, Z; u)$  即为  $\text{Trace}(G_1^*)$  的语义值. 另一方面,  $\text{Trace}(G_2)$  的语义值显然是  $\text{Exec}(\pi_2, A_2, Z; u)$ . 于是由定理条件(语义  $\sqsubseteq_E$ -相容)和前面的式(1), 对任何 P.P.T. 算法  $D$ , 有

$$|P[D(u, \text{Exec}(\pi_2, A_2, Z; u))=1] - P[D(u, \text{Exec}(\pi_1, A_1, Z; u))=1]|$$

总是复杂性参数  $k$  的某个可忽略函数, 特别有  $Z^{\text{exec}}(\pi_2, A_2, u) \approx^{\text{PPT}} Z^{\text{exec}}(\pi_1, A_1, u)$  ( $Z^{\text{exec}}(\pi, A, u)$  的涵义见定义 1), 这恰意味着  $\pi_2 \rightarrow^{\text{UC}} \pi_1$ . 证毕.

这里对两个定理的条件给予适当解释. 与上一节一样, 消息代数上的  $\models_E$  问题和  $\sqsubseteq_E$  问题均多项式可解, 实际上是一个充分广泛的条件. 关于刚性条件, 正如 4.1 节所述, Dolev-Yao 刚性是现实的密码协议应该具备的必要条件, 因此在实际应用中常见而自然的情况是  $\pi_1, \pi_2$  都具有刚性. 关于  $^*\text{Dolev-Yao}$  相似, 尽管看起来较之单纯的 Dolev-Yao 相似性要强, 但在实际应用中这正是我们所期望的情况, 这是因为, (尽管条件略有不同) 定理 7 和定理 8 将证明计算形式的密码协议  $\pi_2$  和  $\pi_1$  UC-相似归结为证明相应的符号形式的密码协议  $\tilde{\pi}_2$  和  $\tilde{\pi}_1$  Dolev-Yao 相似, 而  $^*\text{Dolev-Yao}$  相似意味着可以有一个算法完成这一任务. 这时密码协议的安全性分析与证明就可以归结为一个可计算的过程, 这也正是我们的结论最有意义的情形. 总之,  $^*\text{Dolev-Yao}$  相似实际上是一个自然而有现实意义的条件.

对定理 8 所要求的语义的  $\sqsubseteq_E$ -相容性条件, 实际上有理由相信这是一个充分广泛的事实而非纯粹的假设. 因为这里要求的是 strand-图的迹的语义的  $\sqsubseteq_E$ -相容性, 而在构造和表达 strand-图的输出时(参见 3.2~3.5 小节), 我们明确包含了攻击者 strand 的输出, 以往关于被动攻击的处理方法(那些

工作中的迹虽不包含攻击者 strand, 但具体处理技术实际上与 strand 类型无关)可以移植到这里来处理主动攻击情形, 因此那里的结论对我们所要求的  $\sqsubseteq_E$ -相容性条件有支持性作用. 例如, 文献[20, 25] (文献[25]针对主动攻击)实际上表明含相当广泛一类对称加密算子和公钥加密算子的消息代数的语义具有  $\sqsubseteq_E$ -相容性, 文献[26]则在一个不同的形式模型之下, 实际上直接对主动攻击者证明了类似的结论, 其消息代数比前者更丰富. 从这些工作可以看出语义的  $\sqsubseteq_E$ -相容性实际上是一个广泛成立的事实, 然而目前为止我们并不满意于这些工作中的证明方法, 而期望有一个概念性更强、处理更系统和更一般性的证明方法. 出于严格性, 在找到令人满意的证明之前暂将语义的  $\sqsubseteq_E$ -相容性作为一个条件明确表述出来.

最后, 综合 UC-稳定性定理 1 和定理 8 便直接得到以下推论, 它表明下一小节将要建立的分析框架中的安全性结论具有所期望的复合-稳定性.

**定理 9.**  $\phi(\rho)$  是基于子协议  $\rho$  的复合协议,  $\pi_2, \pi_1$  是两个(计算形式的)密码协议且满足定理 1 和定理 8 的相应条件,  $\tilde{\pi}_i = \text{symb}_P(\pi_i), i=1, 2$ . 又设形式协议模型的消息代数上的  $\sqsubseteq_E$  问题多项式可解且其语义赋值  $\sqsubseteq_E$ -相容. 在这些条件下, 若  $\pi_2 \rightarrow^{*\text{DY}} \pi_1$ , 则必有  $\phi(\pi_2/\rho) \rightarrow^{\text{UC}} \phi(\pi_1)$ .

## 5.2 应用

至此已经建立起最重要的概念和理论性结果, 接下来在这一小节概要阐述如何基于这些结论发展出一种新的协议分析与证明技术, 它涵盖(非适应性)主动攻击者和广泛类型的消息代数, 并且其安全结论具有复合-稳定性.

根据 UC 理论, 安全证明的实质在于对 UC 相似性的证明, 因此上面所获得的普遍性结论至少有两种方式应用于密码协议的安全性证明. 对计算形式的密码协议  $\Pi$ , 要证明其 UC 相似于协议  $\Pi_0$  (例如  $\Pi_0$  是理想协议, 其定义蕴涵对攻击者能力的限制, 依照 UC 理论的概念, 正是这些限制内蕴地刻画了所期望的安全性质), 第 1 种方法是建立这样的形式安全性质  $\tilde{\Theta}$ , 使  $\Pi \rightarrow^{\text{UC}} \Pi_0$  当且仅当  $\tilde{\Pi} = \text{symb}_P(\Pi)$  满足  $\tilde{\Theta}$  (不妨称这类命题为 Canetti-Herzog 型定理), 于是证明  $\Pi \rightarrow^{\text{UC}} \Pi_0$  等价于验证  $\tilde{\Pi}$  满足  $\tilde{\Theta}$ , 而后者正可以利用现有工具实施自动分析(为使分析具有现实意义, 这些工具需支持非自由消息代数). 这一途径的关键在于 Canetti-Herzog 型定理, 实际上这是一类普遍性的命题, 一旦建立起来就可以被广



泛应用,并不只限于对特定协议的分析.以上方法是对文献[22]的推广,它(就十分特殊的一类自由消息代数情形)针对认证性质和密钥保密性质给出了具体实例,他们的工作表明协议  $\Pi$  的刚性(文献[22]仅讨论一类十分特殊的情形,参见 4.1~4.2 节的讨论及文献[22]的所谓映射引理)是建立 Canetti-Herzog 型定理的主要工具.需要指出的是,以上途径并不局限于某种特定的形式安全性质  $\tilde{\Theta}$ ,只要协议的刚性保证对应的 Canetti-Herzog 型定理成立,该方法就可以应用.此外,为应用该方法所建立的形式安全性质  $\tilde{\Theta}$  可能比纯粹形式分析领域所常用的形式特征更精确(例如文献[22]对保密性的刻画),在这一意义上,该方法有益地扩展了形式分析技术.

证明.  $\Pi \rightarrow^{\text{UC}} \Pi_0$  的第 2 种方法是对  $\tilde{\Pi} = \text{symb}_P(\Pi)$  和  $\tilde{\Pi}_0 = \text{symb}_P(\Pi_0)$  直接验证  $\tilde{\Pi} \rightarrow^{*\text{DY}} \tilde{\Pi}_0$ ,后者也是一个纯粹的形式分析过程.该方法所应用的自动分析风格与以往许多工作不同,在这里是验证两个形式协议之间是否具有某种关系,更类似于进程代数中的仿真分析技术.定理 8 保证这里的形式分析结果蕴涵  $\Pi \rightarrow^{\text{UC}} \Pi_0$ ,具体算法有待于在后续工作中进一步建立.

如果待分析/验证的协议比较简单,以上方法之一即可完成证明.对较复杂的密码协议  $\Pi$ ,要证明其 UC 相似于给定的理想协议  $\Pi^{\text{ideal}}$ ,可以考虑以下综合的分析程序.

1. 建立一系列具有 Dolev-Yao 刚性的协议  $\Pi_{n-1}, \dots, \Pi_1, \Pi_0$ ;
2. 按照 3.4 节的技术构造符号形式的协议  $\tilde{\Pi}_i = \text{symb}_P(\Pi_i), i = n, n-1, \dots, 1, 0$  (约定  $\Pi_n = \Pi$ );
3. 建立形式安全性质  $\tilde{\Theta}$ , 并证明  $\Pi_0 \rightarrow^{\text{UC}} \Pi^{\text{ideal}}$  当且仅当  $\tilde{\Pi}_0$  满足  $\tilde{\Theta}$ ;
4. 验证  $\tilde{\Pi}_0$  满足  $\tilde{\Theta}$ ;
5. 验证  $\tilde{\Pi}_i \rightarrow^{*\text{DY}} \tilde{\Pi}_{i-1}, i = n, n-1, \dots, 1, 0$ .

如果以上步骤均获成功,则有  $\Pi \rightarrow^{\text{UC}} \Pi^{\text{ideal}}$ .

在以上步骤中,第 1 步是启发式的,目的是把待分析/验证的复杂协议  $\Pi$  变形为越来越简单、越来越接近理想的协议,这些中间协议  $\Pi_i$  并不需要是现实的协议,如何构造它们完全出于方便分析与证明的目的,而且如第 4 节所述,构造具有 Dolev-Yao 刚性的协议  $\Pi_i$  常常并不困难而且自然;第 2 步是完全自动化的过程;第 3 步的核心工具是 Dolev-Yao 刚性,参见前一段的讨论;第 4 步和第 5 步也可以完全自动化.正是通过以上第 2、第 4 和第 5 步,我们的分析框架融入协议自动分析技术,并且定理 8 保证

第 5 步的分析结果蕴涵  $\Pi_i \rightarrow^{\text{UC}} \Pi_{i-1} (i = n, n-1, \dots, 1, 0)$ ,再由 UC-相似的传递性<sup>[27]</sup>即达到最终的证明目的.

## 6 总 结

本文建立了密码协议安全性分析与证明的一个新框架,融合计算密码学方法和形式演算方法,涵盖任何(非适应性的)主动攻击者和广泛类型的消息代数,并且该框架导出的安全结论具有复合-稳定性.概括起来说,建立这一分析框架的目的是尽可能地以形式演算过程代替计算密码学式的证明、同时保持两者之间精确的对应关系.正文已经对 UC-理论和 strand-图这一具体的 Dolev-Yao 类型的形式模型建立起该分析框架的基本概念、逻辑结构并证明了基础性结论,最后概要阐述了基于这些结论实现具体分析及证明技术的途径.

限于篇幅,本文仅重点解决了基础理论问题,进一步待解决的重要问题包括:建立有效的 Dolev-Yao 相似性判定算法;如何将具有适应性入侵能力的攻击者包涵进来;应用以上分析框架解决具体的密码协议分析问题;针对其它的 Dolev-Yao 类型的形式模型发展类似的理论和技术,例如对 Abadi 和 Gordan 所建立、Mitchell 和 Scedrov 等所发展的概率 spi 进程代数模型及我国学者建立的 CPA 模型等.更进一步的工作还包括将以上结果综合起来以形成一个较完整的协议分析环境,包括与现有自动分析技术相集成.

## 参 考 文 献

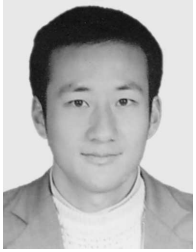
- [1] Bellare M, Canetti R, Krawczyk H. A modular approach to the design and analysis of authentication and key exchange protocols//Proceedings of the 30th Annual Symposium on the Theory of Computing. New York: ACM Press, 1998: 419-428
- [2] Canetti R, Krawczyk H. Analysis of key-exchange protocols and their use for building secure channels//Proceedings of the Advances in Cryptology-Crypto'01. Lecture Notes in Computer Science 2045. Berlin: Springer-Verlag, 2001: 453-474
- [3] Canetti R, Krawczyk H. Security analysis of IKE's signature-based key exchange protocol//Proceedings of the Advances in Cryptology—Crpto'02. Berlin: Springer-Verlag, 2002: 143-161
- [4] Bellare M, Namprempre C, Neven G. Security proofs for identity-based identification protocols and signature schemes

- //Proceedings of the Advances in Cryptology—Eurocrypt’04. Lecture Notes in Computer Science 3027. Berlin: Springer-Verlag, 2004: 35-53
- [5] Feng Deng-Guo, Chen Wei-Dong. Modularized design and analysis of password-based security protocols. *Science in China, Series E*, 2007, 37(1): 223-237(in Chinese)  
(冯登国, 陈伟东. 基于口令的安全协议的模块化设计与分析. *中国科学, E 辑*, 2007, 37(1): 223-237)
- [6] Dolev D, Yao C A. On the security of public-key protocols. *IEEE Transactions on Information Theory*, 1983, 29(2): 198-208
- [7] Fabrega F J T, Herzog J C, Guttman J D. Strand spaces: Proving security protocols correct. *Journal of Computer Security*, 1999, 7(2): 191-230
- [8] Song D, Berezin S, Perrig A. Athena, a novel approach to efficient automatic security protocol analysis. *Journal of Computer Security*, 2001, 9(1): 47-74
- [9] Millen J, Shmatikov V. Constraint solving for bounded-process cryptographic protocol analysis//Proceedings of the 8th ACM Conference on Computer and Communications Security. New York: ACM Press, 2001: 166-175
- [10] Huai Jin-Peng, Li Xian-Xian. Algebraic model of cryptographic protocols and its security properties. *Science in China, Series E: Information Sciences*, 2003, 33(12): 1087-1106(in Chinese)  
(怀进鹏, 李先贤. 密码协议的代数模型及其安全性. *中国科学, E 辑*, 2003, 33(12): 1087-1106)
- [11] Li Jian-Xin, Li Xian-Xian, Zhuo Ji-Liang, Huai Jin-Peng. SPA: A novel highly efficient system for security protocol analysis. *Chinese Journal of Computers*, 2005, 28(3): 456-466(in Chinese)  
(李建欣, 李先贤, 卓继亮, 怀进鹏. SPA: 新的高效安全协议分析系统. *计算机学报*, 2005, 28(3): 456-466)
- [12] Meadows C. Formal methods for cryptographic protocol analysis: Emerging issues and trends. *IEEE Journal on Selected Areas in Communication*, 2003, 21(1): 44-54
- [13] Xue Rui, Feng Deng-Guo. Formal methods and techniques for security protocol analysis. *Chinese Journal of Computers*, 2005, 28(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*, 2005, 28(1): 1-20)
- [14] Abadi M, Rogaway P. Reconciling two views of cryptography: The computational soundness of formal encryption. *Journal of Cryptology*, 2002, 15(2): 103-127
- [15] Abadi M, Jurgens J. Formal eavesdropping and its computational interpretation//Proceedings of the 4th International Symposium on Theoretical Aspects of Computer Software. Lecture Notes in Computer Science 2215. Berlin: Springer-Verlag, 2001: 82-94
- [16] Micciancio D, Warinschi B. Completeness theorems for the Abadi-Rogaway logic of encrypted expressions. *Journal of Computer Security*, 2004, 12(1): 99-129
- [17] Bana G. Soundness and completeness of formal logics of symmetric encryption[Ph. D. dissertation]. Department of Computer Science, University of Pennsylvania. Philadelphia, PA, 2005
- [18] Herzog J. Computational soundness for standard assumptions of formal cryptography[Ph. D. dissertation]. Department of Computer Science, Massachusetts Institute of Technology, 2004
- [19] Abadi M, Cortier V. Deciding knowledge in security protocols under equational theories. *Theoretical Computer Science*, 2006, 2(1): 2-32
- [20] Baudet M, Cortier V, Kremer S. Computationally sound implementations of equational theories against passive adversaries//Proceedings of the ICALP’05. Lecture Notes in Computer Science 3580. Berlin: Springer-Verlag, 2005: 652-663
- [21] Millen J, Shmatikov V. Symbolic protocol analysis with products and Diffie-Hellman exponentiation//Proceedings of the 16th IEEE Computer Security Foundations Workshop. New York: IEEE Press, 2003: 47-61
- [22] Canetti R, Herzog J. Universally composable symbolic analysis of mutual authentication and key-exchange protocols//Proceedings of the Theory of Cryptography Conference. Lecture Notes in Computer Science 3876. Berlin: Springer-Verlag, 2006: 380-403
- [23] Patil A. On symbolic analysis of cryptographic protocols [Ph. D. dissertation]. Department of Computer Science, Massachusetts Institute of Technology. Cambridge, MA, 2006
- [24] Micciancio D, Warinschi B. Soundness of formal encryption in presence of active adversaries//Proceedings of the Theory of Cryptography Conference’04. Lecture Notes in Computer Science 2951. Berlin: Springer-Verlag, 2004: 133-151
- [25] Cortier V, Warinschi B. Computationally sound, automated proofs for security protocol analysis. *Logical Methods in Computer Science*, 2007, 3(3): 396-458
- [26] Backes M, Pfitzmann B, Waidner A. A composable cryptographic library with nested operations//Proceedings of the 10th ACM Conference of Computer and Communications Security. New York: ACM Press, 2003: 122-136
- [27] Canetti R. Universally composable security: A new paradigm for cryptographic protocols//Proceedings of the 42nd Annual Symposium on Foundations of Computer Science. New York: IEEE Press, 2001: 136-155
- [28] Canetti R, Krawczyk H. Universally composable notions of key-exchange and secure channels//Proceedings of the Advances in Cryptology—Eurocrypt’02. Lecture Notes in Computer Science 2332. Berlin: Springer-Verlag, 2002: 337-351
- [29] Canetti R, Rabin T. Universal composition with joint state//Proceedings of the Advances in Cryptology—Crypto’03. Lecture Notes in Computer Science. Berlin: Springer-Verlag, 2003: 565-582

- [30] Canetti R, Dodis Y, Pass R, Walfish S. Universally composable security with global setup//Proceedings of the Theory of Cryptography Conference'07. Lecture Notes in Computer Science 4392. Berlin; Springer-Verlag, 2007: 61-85
- [31] Blanchet B, Pointcheval D. Automated security proofs with

sequence of games//Proceedings of the Advances in Cryptology—Crypto'05. Berlin; Springer-Verlag, 2005: 537-554

- [32] Blanchet B. A computationally sound mechanized prover for security protocols//Proceedings of the IEEE Symposium on Security and Privacy. Oakland, CA, 2006: 96-120



**TIAN Yuan**, born in 1966, Ph. D., associate professor. His main research interests include computer cryptography and its applications in network security.

sor. Her main interests include Lie algebra, algebraic geometry and applications in computer science.

**JIN Feng**, born in 1984, Ph. D. candidate. His mainly interests include computer cryptographic protocols design and verification techniques.

**JIN Yue**, born in 1984, Ph. D. candidate. His mainly interests focus on network security.

**WANG Ying**, born in 1967, Ph. D., associate profes-

## Background

How to integrate computational and symbolic approaches to analyzing complicated cryptographic protocols is one of the most challenging problems in information security area, however, almost none existed approach can reach satisfactory composability in real-world aspects. Based-upon two novel concepts of “Dolev-Yao nonmalleability” and “Dolev-Yao emulation”, our approach establishes an analysis framework covering real-world non-free message algebras and against malicious adversaries via techniques of symbolic extraction and semantic assignment. Security properties proved in this framework are universally composable, i. e., all security properties are provably-preserved when combined with any malicious runtime environment. One of the novelties in this approach is that Canetti's concept of UC emulation and our concept of Dolev-Yao emulation are dual each other and this

analysis framework is both sound and complete. Based-on the above theoretical consequences, a new method for cryptographic protocol analysis is under systematic construction in the project supported by Chinese NFS which ultimate objective is to develop new efficient techniques to implement the automated system for semantic verification of cryptographic protocols. More Concretely, this research systematically applies this method to integrate computational setting and symbolic setting of strand model, process algebra model and process calculus model respectively, develops efficient analysis algorithms and automated verification tools and applies these tools to real-world cryptographic protocols design and analysis. The consequences in this paper contributes theoretical foundations and elementary techniques to this research.