Kerberos 5 协议的形式化分析综述

赵倩倩[1] 李舟军[1] 周 倜[2]
(北京航空航天大学计算机学院 北京 100083)[1] (国防科技大学计算机学院 长沙 410073)[2]

摘 要 网络认证协议 Kerberos 5 提供三方认证机制,允许客户在单次登录的前提下实现对多个网络应用服务器的身份认证,目前该协议已得到广泛应用。FreeBSD,Linux 服务器以及微软公司的 Windows 系列均采用该协议提供网络安全认证,因而该协议自身的安全性引起人们的广泛关注。由于该协议采用时间戳机制,同时涉及 4 个参与方,协议的复杂度较高,如何对其安全性进行全面的形式化分析与验证,一直是安全协议分析领域的研究热点与难点。目前国际上对其验证的方法主要分为两类,分别是基于符号模型的验证方法和基于计算模型的验证方法。全面系统地介绍和分析了目前国际上对该协议的形式化验证工作,在此基础上简要介绍作者目前的研究工作。

关键词 Kerberos 5,形式化分析,认证性,保密性

Survey on Formal Analysis of Kerberos 5

ZHAO Qian-qian[1] LI Zhou-jun[1] ZHOU Ti[2]
(Department of Computer, Beihang University, Beijing 100083, China)[1]
(Department of Computer Science, National University of Defence Technology, Changsha 410073, China)[2]

Abstract Network Authentication Protocol Kerberos 5 provides us with a third-party authentication mechanism. The protocol was designed to allow a client to repeatedly authenticate herself to multiple network servers based on a single login. Nowadays this protocol has been widely used in Windows serials as well as FreeBSD and Linux system for network authentication. As a result, the security properties of it attract great attention. Since the protocol adopts timestamp and involves four participants, how to formally analyze and verify its security comes to be a hot and hard research point. Many methods were carried out internationally, which can be divided into two main approaches: one based the symbolic model and another based on the computational model. This paper introduced and analyzed typical formally work on analyses of Kerberos 5 over the world across-the-board and systematically, finally showed out our current research effort.

Keywords Kerberos 5, Formally analysis, Authentication, Confidentiality

1 引言

伴随开放网络系统的飞速发展,网络系统的安全性及可靠性面临严峻的挑战。为了满足开放式系统的认证需求,安全协议 Kerberos[1] 应运而生。Kerberos 是由 MIT 在其 Athena 项目中开发的一种为网络通信提供可信第三方服务的面向开放系统的认证机制,目前已开发到第五版,即 Kerberos 5[2]。鉴于其有效的身份认证机制,目前该协议已得到广泛应用,FreeBSD,Linux 服务器以及微软公司的 Windows 系列均采用该协议提供网络安全认证。鉴于该协议采用时间戳机制,并涉及多个协议参与方,协议的复杂度较高,因此该协议自身的安全性受到人们的广泛关注。如何对其安全性进行全面的形式化分析和验证,一直是安全协议分析领域的研究热点和难点。

目前国际上对其形式化分析的方法可归结为两类:基于符号模型的验证方法和基于计算模型的验证方法。本文将基于如上分类,全面系统地分析自 Kerberos 5 问世以来国际上对它的形式化验证工作,在此基础上指出进一步的研究方向。

2 Kerberos 5 协议

Kerberos 5 的设计目标:允许一个用户在单次登录的前提下,可实现与多个网络服务器的交互以获取服务。协议通过支持票据在其有效期内的重复使用来实现上述目标。该协议的基本版本如图 1 所示,客户在第一轮得到的票据授权票据有效期一般可达一天,第二轮得到的服务票据有效期一般为若干分钟,每个票据在有效期内均可被多次使用。

The first round: Authentication

(1) $C \rightarrow KAS: \{C, TGS, n_1\}$

(2) $KAS \rightarrow C: C, TGT, \{AK, n_1, t_k, TGS\}_{K_C}$

The second round: Authorisation

(3) $C \rightarrow TGS: TGT, \{C, t_C\}_{AK}, S, n_2$

(4) $TGS \rightarrow C: C, ST, \{SK, n_2, t_T, S\}_{AK}$

到稿日期:2008-06-13 本文研究得到国家自然科学基金(60473057,90604007,60703075,90718017)和高等学校博士学科专项科研基金资助课题(20070006055)的支持。

赵倩倩 硕士研究生,研究方向为安全协议的形式化验证;**李舟军** 教授,博士生导师,研究方向为进程代数理论、安全协议的形式化验证及数据挖掘;**周 倜** 博士研究生,研究方向为安全协议的形式化验证。

The third round: Service

(5) C→S: ST, $\{C, t_C'\}_{SK}$

(6) S→C: $\{t_C'\}_{SK}$

图 1 Kerberos 5 基本版本消息流图

其中 $TGT=\{AK, C, t_K\}k_T$, $ST=\{SK, C, t_T\}\ k_S$

(1) $C->KAS$: 客户 C 向认证服务器 KAS 发出认证请求 KAS_REQ, 请求访问票据授权服务器 TGS, n_1 为 C 产生的随机数;

(2) $KAS->C$: KAS 回送认证回应 KAS_REP, 包括随机生成的 C 与 TGS 之间会话密钥 AK, 及 C 用于访问 TGS 的票据授权票据 TGT, 其中 K_C 和 K_T 为长期共享密钥;

(3) $C->TGS$: C 向 TGS 发送授权请求 TGS_REQ, 将 KAS 为其发放的 TGT 出示给 TGS, 同时将自己的认证信息与期望通信的应用服务器 S 的身份标识一并发送, n_2 为 C 产生的随机数;

(4) $TGS->C$: TGS 向客户发送授权回应 TGS_REP, 发送随机产生的用于 C 和 S 通信的会话密钥 SK 及服务票据 ST;

(5) $C->S$: C 向 S 发送应用服务请求 AP_REQ, 将 TGS 为其发放的服务票据 ST 以及自己的认证信息出示给应用服务器 S, 请求获得服务;

(6) $S->C$: 应用服务器回送认证回应 AP_REP, 用于客户确认服务器已通过对其的认证。

PKINIT[3] 为基本 Kerberos 5 的扩展版本, 仅改变协议的第一轮消息流, 采用公钥加密机制取代 C 和 KAS 之间的长期共享密钥机制, 如图 2 所示。其它两轮与原协议同。

C→KAS: $Cert_C, [t_C'', n_2]_{sk_C}, C, T, n_1$

KAS→C: $\{\{Cert_K, [k, ck]_{sk_K}\}\}_{pk_C}, C,$

$TGT, \{AK, n_1, t_K, T\}_K$

图 2 PKINIT 版本中第一轮消息流图

其中 $TGT=\{AK, C, t_K\}k_T$

3 Kerberos 5 的形式化分析现状

自 Kerberos 5 问世以来, 国际上学者应用多种形式化方法分析该协议的安全性。主要关注该协议的保密性和认证性, 包括 AK 及 SK 的保密性、消息源及票据的认证性。纵览目前国际上对其形式化分析的方法, 可归结为两类, 一类基于符号模型, 即 Dolev-Yao 模型[4], 该模型使用代数项建模协议, 消息被抽象为项。在此模型下利于发现协议的攻击反例。另一类基于计算模型, 基于概率及复杂性理论, 将消息描述为位串, 加/解密操作以概率算法实现。在此模型下证明的安全属性有可靠的安全保证。符号模型可视为计算模型的理想化, 计算模型依赖人工参与, 符号模型下可实现自动化验证。

3.1 基于符号模型的形式化分析方法

3.1.1 归纳法建模, Isabelle 工具验证

G. Bella 等采用归纳法[5] 建模 Kerberos 5 的基础版本, 将协议的安全属性描述为定理, 在此基础上利用 Isabelle 定理证明器[6] 证明协议的安全属性[7,8]。

3.1.1.1 归纳法建模

归纳法的核心概念是事件, A 给 B 发送一个消息 X 对应一个事件: Says A B X, 归纳法将安全协议建模为所有可能的迹的集合, 其中迹是连续的事件序列。

G. Bella 用迹的长度建模网络的当前时间, 利用归纳法建模基本 Kerberos 5 协议, 归纳基础是空规则, 6 条规则(KV1 至 KV6)建模协议参与者的行为, 两条 Oops 规则分别用于描述认证密钥 AK 及会话密钥 SK 失效后的偶然丢失, 遵循 Dolev-Yao 模型用 Fake 规则描述攻击者的能力。

3.1.1.2 Isabelle 工具验证

在一个典型的证明过程中, 专家指导 Isabelle 执行一定的归纳, 再对归纳结果进行筛选, 剔除冗余信息, 确定子目标, 然后将子目标递交自动定理证明器或根据引理将其分解为更小的子目标进行处理。

G. Bella 利用辅助定理证明器 Isabelle 通过证明协议的保密性定理、认证性定理、唯一性定理及密钥分发定理来验证 Kerberos 5 协议, 具体验证如下安全属性。

(1) 保密性: 分别从产生该密钥的服务器及利用该密钥通信的参与者的角度验证了 AK 及 SK 的保密性。

(2) 认证性: 分别验证协议中通信方之间的认证性、票据 TGT 及 ST 的认证性、客户身份信息的认证性。

(3) 唯一性: 分别验证了 AK 及 SK 的唯一性, 以及协议消息中时间戳的唯一性。G. Bella 定义会话密钥的唯一性为: 任意一个该协议运行迹中同一个服务器产生的多个会话密钥互异; G. Bella 称协议满足消息中时间戳的相对唯一性, 若任意一个该协议运行迹中不会出现两个事件, 记录同一个参与者发出的两个消息, 它们对应协议中的同一个消息格式, 两条消息中该参与者加盖的时间戳相同, 但消息内容却不同。

(4) 密钥分发性: 若 A 拥有证据, 可证明 B 获得与其共享的会话密钥 K, 则 A 对 B 满足密钥 K 分发性。G. Bella 分别验证了 A 与 TGS、A 与 B 之间的密钥分发性。

G. Bella 据此得出 Kerberos 5 基本版本满足上述安全属性。但 G. Bella 在验证 AK 保密性时仅从 KAS 的角度考察, 未从 A 及 TGS 的角度考察; 同时由于归纳法本身建模时间戳上的缺陷, 故 G. Bella 虽证明了协议满足其定义的消息中时间戳的唯一性, 但却无法验证消息中时间戳的绝对唯一性, 即无法证明多个迹中同一参与者对同一格式下的消息加盖的时间戳互异, 这归因于归纳法用迹的长度表示网络的当前时间, 故无法阻止同一参与者在两个长度相同的迹上分别发送同一消息格式下加盖相同时间戳而内容不同的两条消息。

3.1.2 MSR 建模, 归纳法证明

F. Bulter 等采用安全协议建模语言 MSR[9,10] 分别在多个层次上建模 Kerberos 5 协议, 在符号模型下采用归纳证明方法手工验证了协议的部分安全属性[11,12]。

3.1.2.1 MSR 建模

F. Bulter 等在 3 个层次上建模 Kerberos 5, 以图 1 中协议的基本版本为准线, A 层次模型涵盖该版本下的所有信息, 剔除前两轮交换中的时间戳信息。该层次作为下述层次模型建模及验证的基础; B 层次模型涵盖该版本下的所有信息, 并且前两轮交换中 KAS 及 TGS 回送的应答中, 除包含原版本中的时间戳外, 还添加对应的生存期信息。该层次下主要关注时间属性; C 层次模型涵盖 A 模型的所有信息并添加加密类型、标记、可选项、校验和及错误消息机制。A 和 C 层次模型中均剔除了绝大部分的时间戳信息, 仅考虑协议最后一轮中的时间戳。而实际处理又将其作为随机数处理, 因此未考

虑协议中的时间约束条件；B 层次模型中以局部时钟建模时间戳。

F. Bulter 采用 MSR 分别在上述层次上建模协议为对应规则集，其中规则引发事实多集之间的变迁。事实多集即为多个事实序列，事实包括网络消息、储存信息、协议参与者的中间状态。

在每个层次下建模 Dolev-Yao 攻击者模型为如下三类规则集：网络消息截获/传输、消息分解/组合及已知密钥前提下对消息的加密/解密规则；数据生成规则；数据访问规则。F. Bulter 建模 B 层次下攻击者模型同 A 层次下攻击者模型，C 层次下攻击者模型仅对 A 层次下攻击者模型中的上述第一类规则进行了改动，添加加密类型信息。

3.1.2.2 归纳法证明

F. Bulter 通过引入 rank 函数和 corank 函数描述验证属性，其中 rank 用于数据源的认证性验证，corank 用于保密性验证。

给定密钥 k 和消息 m_0，定义项 t 对应于消息 m_0 的 k-rank 值为项 t 中使用密钥 k 对消息 m_0 加密的层数。

给定密钥集 E 及消息 m_0，定义项 t 对应于消息 m_0 的 E-corank 值。即为了从 t 中提取出消息 m_0 而使用 E 中密钥的最少次数。

认证性：如果一个特定的协议参与者产生一个项 F，其对应于消息 m_0 的 k-rank 为 1，即 $F=\{m_0\}_k$，而对于所有其他协议参与者包括入侵者，根据协议规则重写一集事实多集 M 为 M' 后，M' 中不存在项对应于消息 m_0 的 k-rank 大于 M 中每一项对应值。若初始迹对应的事实集中每一项对应于消息 m_0 的 k-rank 均为 0，而迹的后续事实集中存在大于 0 的项，则证明前述协议参与者在迹的某个时间点上产生了项 F，即实现了对 F 的数据源认证。

保密性：给定密钥集 E，若协议轨迹中不会出现项对应于消息 m_0 的 E-corank 值为 0，也即协议参与者必须使用 E 中密钥才能得到消息 m_0，则 m_0 满足保密性。

F. Bulter 证明了在 A 层次协议模型满足如下安全属性：

(1) AK 的保密性，即若攻击者无法获得用于加密 AK 的长期密钥 K_C 及 K_T，则攻击者无法获得 AK。F. Bulter 通过证明迹中不存在某个事实对应于 AK 的 $\{K_C, K_T\}$-corank 值为 0 来实现。

(2) *TGS_REQ* 中 *TGT* 及客户身份信息的认证性，即若攻击者无法获得长期密钥 K_T，同时该 *TGT* 初始不存在，则若 *TGS* 收到一个标志为 C 发出的请求，包含 *TGT* 及 AK，则必有某个 *KAS* 产生了此 *TGT* 及 AK。同时，若攻击者无法获得用于加密 AK 的长期密钥 K_C，则客户认证信息确由该客户发出。F. Bulter 通过证明，对应于 $\{AK, C\}$ 的 K_T-rank 值只有特定的 *KAS* 才可增加此值，验证了 *TGT* 的认证性；假定初始迹中无事实 $I(K_C)$，即攻击者 I 无法获得 K_C，则 I 无法获得 AK，因而 I 不能增加对应于 $\{C\}$ 的 AK-rank 值，协议参与者中仅有 C 可执行此操作。故 $\{C\}_{AK}$ 确由该 C 发出，据此实现客户身份信息的认证性。

(3) SK 的保密性，即若攻击者无法获得 K_S 及 AK，则攻击者无法获得 SK。F. Bulter 通过证明迹中不存在某个事实对应于 SK 的 $\{K_S, AK\}$-corank 值为 0 来实现。

(4) *AP_REQ* 中 *ST* 及客户身份信息的认证性，即若攻击者无法获得 K_S，同时此票据初始不存在，若 S 收到一个标记为 C 发出的请求，则某个 *TGS* 产生了此 *ST* 及 SK，同时若攻击者无法获得 AK，则客户认证信息确由该客户发出。证明过程与(2)中类似。

B 层次模型下的验证结果与 A 层次下相同，C 层次模型下仅证明了 *TGS_REQ* 中 *TGT* 及客户身份信息的认证性。

在 A 层次的验证过程中，F. Bulter 发现了票据异常，即在协议第一轮攻击者将 K 回送的 *TGT* 替换为无用消息。当 C 向 *TGS* 发出请求后，攻击者将其中的无用信息替换为 *TGT*。此时协议虽分别满足了 *TGS_REQ* 中 *TGT* 的认证性及客户身份信息的认证性，但不满足整个 *TGS_REQ* 消息的认证性。

在 B 层次验证下未发现新的异常。

在 C 层次的验证过程中，F. Bulter 发现了如下 3 种异常：ⅰ)匿名票据交换异常，即客户向 *TGS* 请求两张用于与 S 通信的服务票据：一张为匿名票据，另一张为非匿名票据。攻击者 I 将截获 *TGS* 回送的两个消息，将其中的服务票据互换。当客户向 S 发送两个请求后，I 截获两个消息并将其中匿名客户认证信息替换为另一个消息中的非匿名客户认证信息，然后将消息发送给 S。S 将接受其中非匿名服务请求，同时拒绝匿名服务请求，回送错误信息(其中包括客户提供的身份认证信息)，则客户认为自己获得匿名服务而实际上获得的是非匿名服务。ⅱ)票据重放异常，攻击者 I 截获客户 C 发送给 S 的服务请求，重放该请求以获得 S 回送的重放错误消息。I 将此消息替换为其它错误消息类型，并将此消息回送给 C。ⅲ)加密类型异常，假定客户 C 对于不同的加密方法拥有不同的长期密钥，攻击者 I 获得了其中一个密钥。若 C 向 *KAS* 发出认证请求消息，I 将该消息中的加密类型域改变为他所掌握的密钥对应的方法，然后假冒 C 获取服务。

F. Bulter 在 3 个层次上手工建模及验证了 Kerberos 5 的部分安全属性。首先实现验证 A 层次模型下协议的部分安全属性并发现了一个异常。在此基础上，分别扩展到 B 层次和 C 层次，B 层次旨在关注协议的时间属性。但在该层次下 F. Bulter 提供的攻击者模型与 A 层次下相同，未考虑密钥在失效后可被攻击者获取这一规则，同时时间属性考虑不全面，因此实际上 B 层次模型下验证未获得有效结果。而 C 层次在 A 的基础上附加关注了加密类型、可选项等信息，发现了 3 种异常。但 F. Bulter 发现的上述异常均不具有威胁性。

3.2 从符号模型到计算模型的过渡

3.2.1 基于 BPW 模型

M. Backes 等基于 BPW 模型[13]首次在计算层次下手工地验证 Kerberos 5 基础版本及 PKINIT 版本[14]。

3.2.1.1 BPW 建模

BPW 模型为安全协议提供一个确定的 Dolev-Yao 攻击者模型，在此基础上建模协议，在该模型下协议的验证结果可迁移到计算模型上来。

在 BPW 模型下，每个协议参与者对应一个 I/O 机，I/O 机之间通过端口相连，参与者通过与其他 I/O 机进行交互来执行协议，敌手机将会与所有诚实机通过端口进行交互。在这种交互式场景下，语义是基于状态的，状态用抽象数据库 D 及对 D 中项的控制描述。数据库中的每一项均有其类型属性及指向其参数的指针，同时有对该项的控制。仅当一个参

与者直接或者间接获得对该项的控制时才可访问该项，协议参与者之间通过发送命令来传送控制。每个协议机在输入端口获得某个输入信息后，执行该参与者对应的协议段，然后从输出端口输出对应信息。在此过程中，协议机对 D 进行读写操作。

M. Backes 据此建模了图 1 和图 2 中协议的两个版本。

3.2.1.2 协议验证

M. Backes 定义协议的安全性如下。

(1) 密钥保密性：对任何诚实客户 C 和诚实服务器 S，若 TGS T 产生了 C 和 S 共享的会话密钥 SK，则攻击者无法获悉 SK；

(2) 认证性：ⅰ)若服务器 S 完成协议的一次运行，认为是与 C 的通信，则在此之前，C 发起了协议，从某个 KAS 处获得了 TGT，然后从某个 TGS 请求了服务票据；ⅱ)若 C 完成一次协议运行，认为是与 S 通信，则 S 给 C 发送了一个有效的 AP_REP 消息。

M. Backes 通过验证保密性需求及认证性需求来获得协议的安全性。譬如定义保密性需求：

t_1 : KA_out_s ! (OK, Kerb, u, SK^{hnd})

∨ t_2 : KA_out_u ! (OK, Kerb, S, SK^{hnd})

⇒ t_3 : D[$hnd_u = SK^{hnd}$]. $hnd_a = \downarrow$,

即 t_1 时刻服务器 S 对应的 I/O 机在端口 KA_out_S ! 输出消息(OK, Kerb, u, SKhnd)，即 S 结束与客户机 u 的会话，拥有对 SK 的控制。或 t_2 时刻客户机 u 对应 I/O 机在端口 KA_out_u ! 输出(OK, Kerb, S, SK^{hnd})，结束与 S 的会话，拥有对 SK 的控制，则在任意 t_3 时刻下，D 中对应 SK 的项，攻击者无法获得对其的控制。认证性需求描述类似。

证明过程从一个结束状态开始，反向重建协议的一个迹。据此 M. Backes 证明了保密性需求和认证性需求，得出 Kerberos 5 两个版本的保密性和认证性在此模型下均满足。

M. Backes 将 BPW 模型下协议的验证结果迁移到计算模型下，得出两个版本协议的认证性在加密实现下依然满足，但保密性不保持。

M. Backes 第一次将 Kerberos 5 符号模型下的部分验证结果迁移到计算模型下，但整个验证过程均通过手工实现。

3.3 基于计算模型的形式化分析方法

3.3.1 利用 CryptoVerif 工具验证 PKINIT

A. D. Jaggard 等在 M. Backes 工作[14]的基础上，利用 CryptoVerif 工具[15,16]，机械化地验证 PKINIT 对应的 Kerberos 5 协议第一阶段的保密性和认证性[17]。

CryptoVerif 工具将协议用概率多项式时间下的进程代数建模，消息用位串标识，加密原语用对位串的操作描述。每个进程用数组记录核心信息，Events $e(M_1, \cdots, M_k)$ 用于记录一个特定的程序点已抵达。给定一个安全参数 η，工具证明在有敌手参与的情况下多个协议会话是否满足 η。安全属性的证明过程用以进程代数描述的游戏序列表示，初始游戏格局为该安全协议对应的角色进程，涉及的所有变元均存储在数组中。游戏之间的变迁通过加密原语的安全定义或合成转换等操作实现。工具采用交互式模型，需要用户输入命令指导工具验证，该工具具有可靠性，但不具备完备性。

工具在一系列假定前提下执行证明过程，A. D. Jaggard 利用工具分别证明 PKINIT 下 AK 的保密性，通过输入 query secret AK 命令实现；C 对 KAS 的认证性，通过输入如下命令实现：

```
query x:bitstring, k:key;
event inj:fullC(KAS, k, x) ==>
      inj:fullKAS(C, k, x)
```

其中 fullC 为 C 结束第一阶段会话后产生的一个事件，fullKAS 为认证服务器 KAS 完成该阶段会话后产生的一个事件，表示若 C 结束了第一阶段的会话，则 KAS 必定参与了此会话。工具未证明出 PKINIT 满足 AK 的保密性及 C 对 KAS 的认证性。

B. Blanchet 首次尝试了在计算模型下机械化地验证 PKINIT，但仅验证了协议第一阶段安全性。

3.3.2 利用 CryptoVerif 工具验证 Kerberos 5

B. Blanchet 在 M. Backer[14]及 A. D. Jaggard[17]工作的基础上，利用 CryptoVerif 工具，首次在计算模型下机械化地分析了 Kerberos 5 的两个版本[18]。

CryptoVerif 工具对协议的建模方法如 3.3.1 节中所述，在此不赘述。

工具在一系列加密假定前提下进行安全性的验证，包括假定公钥加密机制在适应性选择密文攻击下不可区分 IND-CCA2、签名机制在选择消息攻击下不可伪造 UF-CMA、对称加密机制应满足密文完整性 INT-CTXT 及选择明文攻击不可区分性 IND-CPA 等。

工具利用事件记录协议中某个特定程序点已抵达，将认证性描述为若某个事件已执行，则在此之前某些特定的事件必已执行。

对于基本 Kerberos 5 版本，在认证性证明过程中，对于每一个对称密钥，应用 INT-CTXT 及 IND-CPA 前提，进行加密转换，逐渐简化证明游戏，最终证明了对应性属性。对于 PKINIT 版本，在认证性的证明过程中，应用 IND-CCA2 及 UF-CMA 前提，指导证明游戏的运行，最终获证该版本下的认证属性。

B. Blanchet 据此得出 Kerberos 5 两个版本均满足：C 对 KAS 的认证性、服务票据请求消息的认证性、C 对 TGS 的认证性、服务请求消息的认证性以及 C 对 S 的认证性。

工具考察同一角色进程的多次运行，将所有涉及变元存储在数组中，以进程 id 取对应进程复本中的对应变元。B. Blanchet 考察如下保密性：进程 Q 保持数组 x 的保密性，当且仅当以绝对的可能性，多次测试查询中攻击者同 Q 交互无法将数组 x 中的成员同独立均匀分布下的随机数区分开来。

据上保密性定义工具证明了协议仅在在第一轮保持了 AK 的保密性，仅在前两轮保持了 SK 的保密性，但工具证明协议满足可选子会话密钥保密性。

密钥可用性指一个交换密钥，即使它同随机数可分，但依然可安全地用于某些加密操作。密钥可用性保证了即使密钥可同随机数相区分，攻击者依然无法获得协议后期用该密钥加密的信息。A. Datta 定义密钥可用性如下[19]：给定一个密钥交换协议 Σ，一类应用 S 及两相攻击者 $A = (A_c, A_c)$，在密钥交换阶段，诚实参与者在 A_c 可控制的网络上运行多个协议会话。然后 A_c 从其中选择一个会话，将其会话 id 及攻击者所收集的信息传递给 A_c，进入挑战阶段。A_c 根据 A_c 传递的密钥信息去攻破 S 中的某个机制，若不成功，则密钥可用性

满足。由于 CryptoVerif 不支持顺序建模两个阶段，同时不支持两相攻击者模型及选择会话 id 及其密钥进行攻击操作，B. Blanchet 扩展密钥可用性概念，提出强密钥可用性概念如下：

给定对称加密下的一类应用 S，给定 $b\in\{0,1\}$、一个密钥交换协议 $\sum$ 和攻击者 A，进行如下试验：

ⅰ）给定 A 一个安全参数 η，授权 A 可与 $\sum$ 的多项式级会话进行交互；

ⅱ）在某个执行点上，在 A 的请求下，随机地给 A 一个会话 id，同时给其对 k 加密预言机 E_k 及 k 解密预言机 D_k 的访问权；

ⅲ）敌手执行 IND-CCA2 攻击游戏。在此过程中，A 可向 E_k 提交相同长度的消息对 $\{m_0,m_1\}$，得到对 m_b 的加密输出。但 A 绝不会将 E_k 加密得到的消息输入 D_k，A 可与执行中的协议会话进行交互，所有协议会话在抵达一个执行点后将不接受 E_k 输出的密文；

ⅳ）最后在某个时间点，A 猜测位值 $d\in\{0,1\}$ 作为实验输出结果。

若对于 S 中的所有机制，在任意多项式级的攻击者 A 下，b 取 0 时的上述实验输出 1 的概率，近似等于 b 取 1 时上述实验输出 1 的概率则称 $\sum$ 中的交换密钥对 S 中的机制是强可用的。

据上定义，B. Blanchet 采用辅助结构证明了在两个版本下对于对称加密机制、AK 及 SK 均满足 IND-CCA2 强密钥可用性。

B. Blanchet 利用 CryptoVerif 工具，在计算模型下对 Kerberos 5 协议的两个版本进行了建模和验证。但由于工具工作在交互式模式下，需要人工输入一系列命令来指导验证过程，依赖专家意见。

结束语 基于计算模型的验证方法提供可靠性但不具备完备性，而基于符号模型的验证方法易于实现自动化同时利于发现攻击反例，因此两类方法相互补充，缺一不可。

目前国际上已在计算模型下对 Kerberos 5 协议的两个版本进行了完整建模，并在交互式验证工具辅助下完成了相应的验证工作。但在符号模型下，协议的验证尚未完善，已有的工作或者仅是手工证明，或者仅利用工具验证了协议的部分安全属性。国际上尚未在该模型下开发出有效工具，以实现自动化验证 Kerberos 5。

我们课题组长期致力于安全协议的形式化分析和验证工作，成功开发了安全协议验证工具 SPVT[20]，该工具基于 Objective Caml 语言，支持安全协议的自动验证和反例的自动生成[21,22]。目前该工具已实现对大量安全协议的自动化验证，但对 Kerbero 5 的验证尚未实现。

针对安全协议逻辑程序模型的不动点计算不一定终止（故安全协议的验证过程也不一定停机）的问题，基于抽象解释理论，我们提出了对安全协议保密性和认证性进行抽象验证的一个抽象精化框架，构造出了一个停机的安全协议抽象验证过程[23]。为进一步提高安全协议的验证效率，我们提出了一种安全协议解形式不动点停机性的动态刻画方法和相应的预测方法，并提出了一种将停机的抽象验证方法和不停机的精确验证方法结合起来的组合验证方法[24]。

为了解决时间敏感安全协议自动化验证问题，我们提出了带时间约束的安全协议的扩展 Horn 逻辑模型，给出了从描述模型到验证模型的抽象规则，同时给出了时间敏感安全协议的验证方法，为分析和验证时间敏感安全协议提供了一条新的、高效的途径[25]。

目前我们课题组致力于在上述工作的基础上对 SPVT 工具进行扩展，以实现在符号模型下自动化验证 Kerberos 5 协议。

参 考 文 献

[1] Steiner J G, Neuman B C, Schiller J I. Kerberos: An Authentication Service for Open Network Systems[C]// Proc. of the Winter 1988 Usenix Conference. 1988

[2] Neuman C, Yu T, Hartman S, et al. The Kerberos Network Authentication Service (V5)[OL]. July 2005. http://www.ietf.org/rfc/rfc4120

[3] IETF. Public Key Cryptography for Initial Authentication in Kerberos. RFC 4556. Sequence of Internet Drafts[OL]. http://tools.ietf.org/wg/krb-wg/draft-ietf-cat-kerberos-pk-init/

[4] Dolev D, Yao A. On the security of public-key protocols[J]. IEEE Transaction on Information Theory, 1983, 2(29): 198-208

[5] Paulson L C. Proving properties of security protocols by induction[C]// Proc. of the 10th CSFW. 1997: 70-83

[6] Paulson L C. Isabelle: A Generic Theorem Prover. LNCS 828. Springer, 1994

[7] Bella G, Paulson L C. Using Isabelle to Prove Properties of Kerberos Authentication System[C]// Proc. of DIMACS'97, Workshop on Design and Formal Verification of Security Protocols. New York, USA, 1997

[8] Bella G. Formal Correctness of Security Protocols. IS&C. Springer-Verlag Berlin Heidelberg, 2007

[9] Cervesato I, Durgin N A, Lincoln P, et al. A Meta-notation for Protocol Analysis[C]// Proc. of 12th IEEE CSFW. 1999: 55-69

[10] Cervesato I. Typed Multiset Rewriting Specifications of Security Protocols[C]// Proc. of the First Irish Conference on MFCSIT'00. Elsevier ENTCS 40, 2000

[11] Bulter F, Cervesato I, Jaggard A D, et al. An Analysis of Some Properties of Kerberos 5 Using MSR[C]// Proc. of the 15th CSFW. IEEE Computer Society, 2002: 175-190

[12] Bulter F, Cervesato I, Jaggard A D, et al. A formal analysis of some properties of kerberos 5 using MSR[R]. CIS-MS-04-04. Department of Computer&Information Science, University of Pennsylvania, 2004

[13] Backes M, Pfitzmann B, Waidner M. A universally composable cryptographic library[R]. IACR Cryptology ePrint Archive, 2003/015. January 2003

[14] Backes M, Cervesato I, Jaggard A D, et al. Cryptographically Sound Security Proofs for Basic and Public-key Kerberos[C]// ESORICS'06, LNCS 4189. Springer, 2006

[15] Blanchet B, Pointcheval D. Automated Security Proofs with Sequences of Games[C]// CRYPTO 2006, LNCS 4117. Springer, 2006

[16] Blanchet B. A Computationally Sound Mechanized Prover for Security Protocols[C]// IEEE Symposium on Security and Privacy. May 2006: 140-154

[17] Jaggard A D, Scedrov A, Tsay J-K. Computationally Sound Mechanized Proof of PKINIT for Kerberos// FCC'07

[18] Blanchet B, Jaggard A D, Scedrov A, et al. Computationally

Sound Mechanized Proofs for Basic and Public-key Kerberos[C]// ASIACCS'08. Tokyo, Japan, March 2008

[19] Datta A, Mitchell J, Warinschi B. Computationally Sound Compositional Logic for Key Exchanged Protocols[C]// Proc. of CSFW'06. July 2006

[20] 李梦君,李舟军,陈火旺. SPVT:一个有效的安全协议验证工具[J]. 软件学报,2006,17(4):898-906

[21] 李梦君,李舟军,陈火旺. 安全协议的扩展 Horn 逻辑模型及其验证方法[J]. 计算机学报,2006,29(9):1666-1678

[22] 周倜,李梦君,李舟军,等. 基于 Horn 逻辑扩展模型的安全协议反例的自动构造[J]. 计算机研究与发展,2007,44(9):1518-1531

[23] Li Mengjun, Zhou Ti, Li Zhoujun, et al. An Abstraction and Refinement Framework for Verifying Security Protocols Based on Logic Programming[C]// ASIAN 2007, LNCS 4846. 2007:166-180

[24] Li Mengjun, Li Zhoujun, Chen Huowang, et al. A Novel Derivation Framework For Define Logic Program[J]. Electronic Notes in Theoretical Computer Science, 2008, 212:71-85

[25] Zhou Ti, Li Mengjun, Li Zhoujun, et al. Modeling and Verifying Time Sensitive Security Protocols with Constraints[J]. Electronic Notes in Theoretical Computer Science, 2008, 212:103-118

(上接第 6 页)

[45] Jones J, Harrold M J, Stasko J. Visualization of Test Information to Assist Fault Localization[C]// Proc. of ICSE'02. 2002: 467-477

[46] 刘彦斌,朱小冬. 基于双轨迹差异分析法的软件故障定位[J]. 计算机工程,2007,33(9):43-48

[47] Renieris M, Reiss S P. Fault Localization with Nearest Neighbor Queries[C]// Proc. of ASE'03. 2003:30-39

[48] Wu Ji, Jia Xiaoxia, Liu Chang, et al. A Statistical Model to Locate Faults at Input Level[C]// Proc. of ASE'04. 2004:274-277

[49] Ren X, Ryder B G. Heuristic Ranking of Java Program Edits for Fault Localization[C]// Proc. of ISSTA'07. 2007:239-249

[50] Ostrand T J, Weyuker E J, Bell R M. Predicting the Location and Number of Faults in Large Software Systems[J]. IEEE Trans. on Soft. Eng., 2005, 31(4):340-355

[51] Khoshgoftaar T M, Pandya A S, Lanning D L. Application of Neural Networks for Predicting Faults[J]. Annals of Software Engineering, 1995, 1:141-154

[52] 罗云峰,贲可荣. 基于 BBNs 的软件故障预测方法[J]. 电子学报,2006,34(12A):2380-2383

[53] Pai G J, Dugan J B. Empirical Analysis of Software Fault Content and Fault Proneness Using Bayesian Methods[J]. IEEE Trans. on Soft. Eng., 2007, 33(10):675-686

[54] Denaro G, Pezze M. An Empirical Evaluation of Fault-proneness Models[C]// Proc. of ICSE'02. 2002:241-251

[55] Menzies T, Greenwald J, Frank A. Data Mining Static Code Attributes to Learn Defect Predictors[J]. IEEE Trans. on Soft. Eng., 2007, 33(1):2-13

[56] Ramanna S, Bhatt R, Biernot P. Software Defect Classification: A Comparative Study with Rough Hybrid Approaches[C]// Proc. of RSEISP'07. 2007:630-638

[57] Gall H, Hajek K, Jazayeri M. Detection of Logical Coupling Based on Product Release History[C]// Proc. of ICSM'98. 1998:190-198

[58] Rysselberghe F V, Demeyer S. Studying Software Evolution Information by Visualizing the Change History[C]// Proc. of ICSM'04. 2004:328-337

[59] Graves T L, Karr A F, Marron J S, et al. Predicting Fault Incidence Using Software Change History[J]. IEEE Trans. on Soft. Eng., 2000, 26(7):653-661

[60] Stoerzer M, Ryder B G, Ren X, et al. Finding Failure-Inducing Changes in Java Programs using Change Classification[C]// Proc. of FSE'06. 2006:57-68

[61] Livshits B, Zimmermann T. DynaMine: Finding Common Error Patterns by Mining Software Revision Histories[C]// Proc. of FSE'05. 2005:296-305

[62] Ying A T, Murphy G C, Ng R, et al. Predicting Source Code Changes by Mining Change History[J]. IEEE Trans. on Soft. Eng., 2004, 30(9):574-586

[63] Zimmermann T, Weibgerber P, Diehl S, et al. Mining Version Histories to Guide Software Changes[J]. IEEE Trans. on Soft. Eng., 2005, 31(6):429-445

[64] Bowman I T, Holt R C. Reconstructing Ownership Architectures To Help Understand Software Systems[C]// Proc. of IWPC'99:28-37

[65] Amrit C. Application of Social Network Theory to Software Development: The problem of Task Allocation[C]// Proc. of Computer Supported Activity Coordination. 2005:3-17

[66] Yanga H L, Tang J H. Team Structure and Team Performance in IS Development: A Social Network Perspective[J]. Information & Management, 2004(41):335-349

[67] Setamanit S, Wakeland W W, Raffo D. Exploring the Impact of Task Allocation Strategies for Global Software Development Using Simulation[C]// Proc. of SPW/ProSim'06. 2006:274-285

[68] Bahsoon R, Emmerich W. Economics-Driven Software Mining [C]// Proc. of the First International Workshop on the Economics of Software and Computation (ESC'07). 2007:3-7

[69] Ling C X, Sheng V S, Bruckhaus T F W, et al. Maximum Profit Mining and Its Application in Software Development[C]// Proc. of KDD'06:929-934

[70] Morisaki S, Monden A, Matsumura T, et al. Defect Data Analysis Based on Extended Association Rule Mining[C]// Proc. of MSR'07:3-10

[71] Zhang L, Wang Q, Xiao J, et al. A Tool to Create Process-Agents for OEC-SPM from Historical Project Data[C]// Proc. of ICSP'07:84-95

[72] Rubin V, Gunther C W, Aalst W M P, et al. Process Mining Framework for Software Processes[C]// Proc. of ICSP'07:169-181

[73] Brown A W, Booch G. Reusing Open-Source Software and Practices: The Impact of Open-Source on Commercial Vendors[C]// Proc. of the 7th International Conference Software Reuse: Methods, Techniques, and Tools. 2002:381-428

[74] Livieri S, Higo Y, Matsushita M, et al. Very-Large Scale Code Clone Analysis and Visualization of Open Source Programs Using Distributed CCFinder: D-CCFinder[C]// Proc. of ICSE'07: 106-115

[75] Simal-A Project Support Framework[OL]. http://simal.osswatch.ac.uk/index.html, 2007

[76] Asundi J. The Need for Effort Estimation Models for Open Source Software Projects[C]// Proc. of the Fifth Workshop on Open Source Software Engineering (5-WOSSE). 2005:1-3