一种基于语义的安全协议形式化模型

韩继红 范钰丹 王亚弟 郭渊博
(信息工程大学电子技术学院 郑州 450004)

摘 要 在分析实际网络环境中安全协议的运行特点之后,提出了安全协议建模分析的两点基本假设。在此基础上,提出了一种基于语义的安全协议形式化模型,具体包括基于角色事件的协议静态描述模型和基于运行状态的协议动态执行模型,给出了模型的基本语法及形式语义,明确了模型推理过程中涉及到的一些关键性概念,并以简化的NSL协议为例进行了说明,为实现自动化验证打下了必要的基础。

关键词 安全协议,形式化分析,模型,语义
中图法分类号 TP309 **文献标识码** A

Semantics Based Formal Model for Security Protocols

HAN Ji-hong FAN Yu-dan WANG Ya-di GUO Yuan-bo
(Institute of Electronic Technology, Information Engineering University, Zhengzhou 450004, China)

Abstract Two basic hypotheses for security protocols analysis were proposed after the character of protocols in actual network was analyzed. Then a semantics based formal model for security protocols was proposed, which includes a role event based static protocol description model and a operation state based dynamic execution model, the basic syntax and semantics were introduced, and some key concepts in the model deduction are presented. Finally, the example for the simplified NSL protocol was given. All of those works provide a necessary basis for realizing automatic verification for security protocols.

Keywords Security protocol, Formal analysis, Model, Semantics

1 引言

安全协议形式化分析模型是一种用于检验和评估安全协议的形式化语言描述。像所有分析模型一样,安全协议形式化分析模型是安全协议形式化分析的基础和依据,包括协议描述、目标特性描述、攻击者描述和环境描述等。由于其在协议分析中具有十分重要的作用,模型研究已成为了安全协议形式化分析领域的重要研究内容之一。目前,已提出了不少模型,如BAN逻辑[1]、Dolev-Yao模型[2]、CSP模型[3,4]、Brutus模型[5]、Woo-Lam模型[6,7]、Strand Space模型[8-10]、多重集重写模型[11-13]、Spi演算[14,15]等,每种模型各有侧重点和局限性[16]。本文将提出一种基于语义的安全协议形式化模型,重点研究其中的静态描述模型及动态执行模型,力求达到如下目标:

(1)能够对安全协议进行精确的形式化描述;
(2)具有合理可靠的操作语义;
(3)便于实现自动化验证。

2 基本假设

本文在建模和分析安全协议时,将基于以下基本假设。

2.1 环境假设

安全协议运行在一个具有攻击者的开放网络环境中。要想全面地刻画安全协议的特性和行为,必须综合考虑协议主体、网络和攻击者的行为和状态。如图1所示,协议诚实主体A,B按照协议规范定义的步骤进行消息计算,将发送给协议另一方的消息组装成规定格式后发送到网络上。从网络上接收到发给自己的消息后,进行消息格式匹配检验。若收到的协议符合协议要求,与其期望在该状态下收到的消息形式一致,则继续进行下一轮的消息准备和发送,或完成协议。随着协议步骤的不断推进,主体的状态也不断变化,在没有外界干扰的情况下,最终在进入终止状态时会实现预期的安全目标。

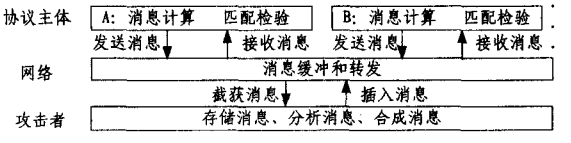

图1 安全协议系统

网络在协议运行中起着消息缓冲和转发的作用,不对消息进行任何修改和过滤。因此,网络可以被建模成两个缓冲器[20]:协议主体发送的消息进入输出缓冲器,主体接收的消息来自于输入缓冲器。正常情况下,消息会被从输出缓冲器自然传递到输入缓冲器。攻击者存在时,这种情况会发生变化。攻击者可以从网络上窃听消息,存储所有截获的消息,对消息进行分析,并根据已有的知识执行合成消息、分解消息的

到稿日期:2008-03-20 本文受863国家重点基金项目(2007AA01Z405),国家自然科学基金(60503012)资助。
韩继红(1966—),女,教授,主要研究方向为计算机网络安全、信息系统安全,E-mail:Hnhanjh@163.com;范钰丹(1982—),女,讲师,主要研究方向为信息安全;王亚弟(1953—),男,教授,主要研究为信息安全等;郭渊博(1975—),副教授,主要研究方向为计算机应用技术、信息安全等。

操作;可以在恰当的时间,向网络中插入重放或篡改的消息。

对于协议诚实主体来讲,网络和攻击者都属于外部环境。因此,可以将网络和攻击者合二为一,将网络的转发能力也归到攻击者的能力之中,构成如图 2 所示的理想系统。

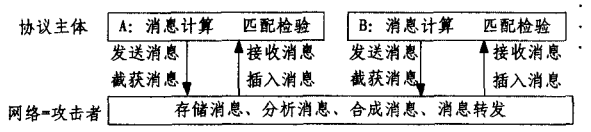

图 2 理想协议系统

在这种情况下,协议主体能否收到符合协议规范的消息,完全取决于攻击者是否发送(转发或伪造)了符合格式要求的消息。

本文将基于如下环境假设对协议进行分析:

(1)协议主体发出的消息全部直接被攻击者截获;

(2)合法主体接收到的所有消息均来自于攻击者。

在建立安全协议形式化模型时,只需描述协议诚实主体行为、攻击者行为和安全目标特性即可。

2.2 主体建模假设

在如图 2 所示的协议系统中,消息计算和匹配检验等都属于协议诚实主体的内部活动。我们只关注协议呈现出来的外部安全特性,因此不对其内部正常活动进行建模,而只考虑其与外界进行消息交互的行为。该假设不会影响安全协议分析的正确性和完备性。

但是,攻击者的内部活动(如基于其计算能力的消息分解和合成行为)会直接影响协议的安全性,因此在建模攻击者时要尽可能细致地反映其最大攻击行为能力。

另外,由于某些主体可能被攻击者攻破或与攻击者合谋,因此将协议主体分为可信的诚实主体和不可信主体两类。攻击者可以得到不可信主体的初始知识及其所有实例。

3 静态描述模型

一个安全协议由执行协议的主体集合、主体的行为集合以及主体之间交互的消息集合组成。从另一个角度讲,可以将安全协议抽象为一个"角色"集合,如{协议发起者,协议响应者},每个角色可以看成是一个"发送"和"接收"消息的有限事件序列。协议主体执行协议,实际上是被赋予了某个"角色",从而可以实施协议规定的行为。安全协议静态模型即描述了协议中每个角色的行为。

3.1 项

安全协议中交互的协议消息可以用项表示。

3.1.1 项的基本语法结构

定义 1(项) 令 $\mathcal{F}$ 为一个项操作符集合,每个项操作符 $f\in\mathcal{F}$ 至少操作一个项,即其元数 $ar(f)$ 至少为 1。$\mathcal{FC}$ 为与密码操作有关的项操作符集合,$\mathcal{FO}$ 为一般操作符集合,$\mathcal{F}=\mathcal{FC}\cup\mathcal{FO}$,且 $\mathcal{FC}\cap\mathcal{FO}=\emptyset$。令 $\mathcal{C}$ 为一个常量(首字母用大写表示,如主体名 A,B,I,随机数 Na,Nb 和各种标识符等)集合,$\mathcal{V}$ 为一个变量(用小写字母表示)集合,则 $\mathcal{T}(\mathcal{F},\mathcal{C},\mathcal{V})$ 表示构建于 $\mathcal{F},\mathcal{C},\mathcal{V}$ 的项集合,它是满足以下条件的最小集合:

1)$\mathcal{C}\subseteq\mathcal{T}(\mathcal{F},\mathcal{C},\mathcal{V})$;

2)$\mathcal{V}\subseteq\mathcal{T}(\mathcal{F},\mathcal{C},\mathcal{V})$;

3)若 $t_1,\cdots,t_n\in\mathcal{T}(\mathcal{F},\mathcal{C},\mathcal{V})$,且 $f\in\mathcal{F}$,则 $f(t_1,\cdots,t_n)\in\mathcal{T}(\mathcal{F},\mathcal{C},\mathcal{V})$。

$t\in\mathcal{T}(\mathcal{F},\mathcal{C},\emptyset)$ 称为基项。$var(t)$ 表示项 t 中包含的变量集合。攻击者的主体名称固定地用常量 I 表示。

定义 2(项类型) 项类型 Γ 是消息项的属性集合。各种类型具体定义如下:

$\tau,\tau_2::=msg$	消息
$\mid princ$	主体名
$\mid role$	角色
$\mid num$	数字
$\mid key$	密钥
$\mid compmsg$	合成消息
$princ::=princt$	可信主体
$\mid princu$	不可信主体
$num::=nonce$	随机数
$\mid ident$	标识
$ident::=runid$	运行标识
$\mid stepid$	步骤标识
$\mid sessionid$	会话标识
$key::=symk$	对称会话密钥
$\mid pubk$	公钥
$\mid privk$	私钥
$\mid longtk$	长期密钥
$compmsg::=H[\tau]$	哈希值
$\mid HK[\tau,\tau']$	密控哈希值
$\mid SC[\tau,\tau']$	对称密钥加密密文
$\mid AC[\tau,\tau']$	公钥加密密文
$\mid SN[\tau,\tau']$	数字签名
$\mid T[\tau,\tau']$	元素组
$\mid ALT$	代数运算项
$\mid OT$	其它复合项
$ALT::=XO[\tau_1,\tau_2,\cdots,\tau_n]$	异或项
$\mid AD[\tau_1,\tau_2,\cdots,\tau_n]$	加项
$\mid MU[\tau_1,\tau_2,\cdots,\tau_n]$	乘项
$\mid EX[\tau,\tau_2]$	幂

$type(m)$ 表示项 m 的类型,一个项具有类型表示为 $t:\tau$。用 $<_{tp}$ 表示项类型之间的偏序关系,如:$princ<_{tp}msg$,$pubk<_{tp}key<_{tp}msg$,$runid<_{tp}ident<_{tp}num<_{tp}msg$。

$f\in\mathcal{FC}$ 作用于项集合可生成如下具体的项形式:

• $[t_1,t_2]$:项的连接,又称为对。我们假定对具有右结合性,这样即可递归得出任意个项的连接 $[t_1,t_2,\cdots,t_{n-1},t_n]=[t_1,[t_2,\cdots,[t_{n-1},t_n]\cdots]]$;

• $senc(t_1,t_2)$:用对称密钥 $t_2:symk$ 加密 $t_1:msg$ 产生的项;

• $penc(t_1,t_2)$:用公钥 $t_2:pubk$ 加密 $t_1:msg$ 产生的项;

• $sign(t_1,t_2)$:用私钥 $t_2:privk$ 签名 $t_1:msg$ 产生的项;

• $h(t_1):t_1:msg$ 的哈希值;

• $hk(t_1,t_2)$:在密钥 $t_2:key$ 控制下对 $t_1:msg$ 的哈希值。

$f\in\mathcal{FO}$ 作用于项集合可生成如下具体的项形式:

• $xor(t_1,t_2,\cdots,t_n)$:项 $t_1:msg,t_2:msg,\cdots,t_n:msg$ 的二进制异或;

• $add(t_1,t_2,\cdots,t_n)$:项 $t_1:msg,t_2:msg,\cdots,t_n:msg$ 的二进制和;

• $mul(t_1,t_2,\cdots,t_n)$：项 $t_1:msg,t_2:msg,\cdots,t_n:msg$ 的二进制乘积；

• $expo(t_1,t_2)$：项 $t_1:msg$ 和 $t_2:msg$ 的模指数运算结果；

• $inc(t_1)$：项 $t_1:num$ 的自加 1，又称后继；

• $inv(t_1)$：项 $t_1:num$ 的乘法逆；

• $opp(t_1)$：项 $t_1:num$ 的相反数。

• $shk(t_1,t_2)$：主体 $t_1:princ$ 和 $t_2:princ$ 的共享密钥；

• $pk(t_1)$：主体 $t_1:princ$ 的公钥；

• $sk(t_1)$：主体 $t_1:princ$ 的私钥；

• $host(t_1)$：主体 $t_1:princ$ 的身份；

• $fo(t_1,t_2,\cdots,t_n)$：其它项操作函数。

定义 3(项深度) 项 $t\in\mathcal{T}(\mathcal{F},\mathcal{C},\mathcal{V})$的深度$\partial(t)$递归定义如下：

• 若 $t\in\mathcal{C}$，则$\partial(t)=1$；

• 若 $t\in\mathcal{V}$，则$\partial(t)=1$；

• 若 $t=f(t_1,t_2,\cdots,t_n)$，则$\partial(t)=1+\max(\partial(t_1),\partial(t_2),\cdots,\partial(t_n))$。

3.1.2 项的操作语义

定义 4(子项) 项 t 的子项 $St(t)$是满足以下条件的最小集合：

(1)$t\in St(t)$；

(2)对于 1≤i≤n，若 $f(t_1,\cdots,t_n)\in St(t)$，则 $t_i\in St(t)$。

子项关系记为≤和≥。$t'\leq$t 表示项 t'在项 t 中出现。对于满足第二个条件的项有 $f(t_1,\cdots,t_n)>t_i$ 或 $t_i<f(t_1,\cdots,t_n)$，此时称 t_i 为项 $t=f(t_1,\cdots,t_n)$的真子项。

定义 5(位置) 令 λ 为一个空序列，$t\in\mathcal{T}(\mathcal{F},\mathcal{C},\mathcal{V})$的位置集合 $\mathcal{O}(t)$定义如下：

• 若 $t\in\mathcal{C}$，则 $\mathcal{O}$(t)=$\{\lambda\}$；

• 若 $t\in\mathcal{V}$，则 $\mathcal{O}(t)=\{\lambda\}$；

• 若 $t=f(t_1,\cdots,t_n)$，则 $\mathcal{O}(t)=\{\lambda\}\cup\{i.p\mid 1\leq i\leq p,p\in\mathcal{O}(t_i)\}$。

t 中在位置 $p\in\mathcal{O}(t_i)$处的子项记为 $t|_p$。对于 t 的一个真子项 t'，存在一个 $p\in\mathcal{O}(t)$，使得 $t'=t|_p$，且 $p\neq\lambda$。在 t 中用 u 替代 $t|_p$ 得到的项记为 $t[u]_p$。若项具有关系 $t'\in t$，则 t 的位置 p 和 t'的位置 p'的关系为 $p'\geq t'$。

定义 6(置换) 置换是一个函数 $\sigma:\mathcal{V}\mapsto\mathcal{T}(\mathcal{F},\mathcal{C},\mathcal{V})$，$\sigma(x)$可简记为 $x\sigma$。$Dom(\sigma)=\{x\mid\sigma(x)\neq x\}$称为 σ 的域，域中元素是有限的。

• 置换 σ 可以连续作用于项 $\mathcal{T}(\mathcal{F},\mathcal{C},\mathcal{V})$，且有 $\sigma(f(t_1,\cdots,t_n))=f(\sigma(t_1),\cdots,\sigma(t_n))$。

• 满足 $\sigma(Dom(\sigma))=\mathcal{T}(\mathcal{F},\mathcal{C},\emptyset)$的置换 σ 称为基置换。

• 对于所有 $x\in\mathcal{V}$，两个置换 σ 和 τ 的复合为 $\sigma\circ\tau(x)=\sigma(\tau(x))$。若对于一个置换 τ，有 $\sigma\circ\tau=\sigma'$，则称置换 σ 比置换 σ'更一般。若 σ'不比 σ 更一般，则称 σ 比 σ'严格更一般。

• 对于所有 x，满足 $\sigma(x)=y,y\in\mathcal{V}$的双射置换 σ 称为 $\mathcal{V}$ 的变量重命名。

定义 7(置换的类型保持性) 一个置换称为类型保持的，若 $\forall v\in Dom(\sigma)$，使得 $type(\sigma(v))=type(v)$。

定义 8(匹配，合一化子) 令 $s,t\in\mathcal{T}(\mathcal{F},\mathcal{C},\mathcal{V})$为两个项。则

• 满足 $\sigma(s)=t$ 的置换 σ 称为 s 到 t 的匹配。若该置换存在，称 s 比 t 更一般，t 称为 s 的实例。若 s 比 t 更一般，反之不然，则称 s 比 t 严格更一般。

• 若存在置换 σ，使得 $\sigma(s)=\sigma(t)$，称 s 和 t 可合一，σ 为 s 和 t 的合一化子；若不存在比 σ 严格更一般的合一化子，称 σ 为 s 和 t 的最一般合一化子 $mgu(s,t)$；若两个项的最一般合一化子存在，则它是可计算的且模变量重命名唯一的[21]。

定义 9(方程、同余、E-等式)

• 一个方程是一个项对 $(s,t)\in\mathcal{T}(\mathcal{F},\mathcal{C},\mathcal{V})\times\mathcal{T}(\mathcal{F},\mathcal{C},\mathcal{V})$，记为 $s=t$。方程是对称的，即 $s=t$ 与 $t=s$ 等价。

• 项上的同余关系～是一种适应项结构和置换的等价关系：

• 若对于所有的 $s,t,t'\in\mathcal{T}(\mathcal{F},\mathcal{C},\mathcal{V})$和 $p\in\mathcal{O}(s)$，都有 $s[t]_p\sim s[t']_p$，则称 $t\sim t'$。

• 若对于所有 $s,t\in\mathcal{T}(\mathcal{F},\mathcal{C},\mathcal{V})$和置换 σ，都有 $\sigma(s)\sim\sigma(t)$，则称 $s\sim t$。

• 对于 $\mathcal{T}(\mathcal{F},\mathcal{C},\mathcal{V})$上的一个方程集合 E，E-等式 $=_E$ 是涵盖 E 的最小同余。即对于所有 $s=t\in E$ 和置换 σ，都有 $\sigma(s)=_E\sigma(t)$。

该定义用于扩展 Dolev-Yao 模型中的攻击者演绎能力，使之可以实施基于代数特性的攻击。

3.2 事件

一个安全协议中包含两类事件：通信事件和目标声明事件。即

$Event=CommEvent\mid GoalClaimEvent$

定义 10(通信事件) 通信事件是协议主体收发消息的行为描述，包括发送事件和接收事件。具体定义如下：

$$CommEvent=send(sid,p,p',m)\mid receive(sid,p',p,m),\quad sid:stepid,p:princ,p':princ,m:msg$$

其中 $send(sid,p,p',m)$表示在协议的第 sid 步 p 向 p'发送消息 m。$receive(sid,p',p,m)$表示在协议的第 sid 步 p'收到看似来自 p 的消息 m。事件中涉及的两个主体中，排在前面的被称为主动主体，记为 $aprinc(CommEvent)$，紧随其后的为被动主体，记为 $pprinc(CommEvent)$。

在协议设计时，随机数经常被用来保证消息的新鲜性，防止重放攻击。在此，我们隐含地认为随机数在它第一次出现在某个事件之前产生。

定义 11(目标声明事件) 目标声明事件表示主体在执行协议后期望达到的安全目标。

这里只讨论秘密性和不同层次的认证性，如存在性、一致性(弱一致性、非单射一致性和单射一致性)和同步性(非单射同步性和单射同步性)。具体目标声明事件 $GoalClaimEvent$ 定义如下：

$$\begin{aligned}GoalClaimEvent=&Secret(p,m)\\&\mid Alive(p,p')\mid WAgree(p,p')\\&\mid NiAgree(p,p')\mid IAgree(p,p')\\&\mid NiSynch(p,p')\\&\mid ISynch(p,p'),p:princ,p':princ,\\&m:msg\end{aligned}$$

3.3 角色

定义 12(角色) 协议角色是一个二元组 $Role=\langle InitK,EvenQ\rangle$，其中 $InitK$ 为角色的初始知识集合，$EvenQ$ 为主动主体均相同的通信事件和目标声明事件序列。$EvenQ$ 中事件所对应的角色记为 $roleof(Event)$。

在此，将协议安全特性的声明放在了角色定义中。这样做的主要原因是由于安全性目标是协议主体的期望，从不同主体的角度看，协议安全性往往不同。如在一个双方均为可信主体的协议中，A 采用 B 的公钥加密一个随机数 Na 传给 B，则 Na 对于 A 来讲是具有秘密性的，因为只有他和 B 知道 Na，攻击者由于没有 B 的私钥而不会得到 Na；但是站在 B 的角度，Na 的秘密性就得不到保证，因为 B 无法确定只有 A 和他自己知道 Na，任何人都可以得到 B 的公钥，用其加密一个自己产生的随机数发给 B。我们在分析 Denning-Sacco[17] 和 Otway-Rees-Paulson[18] 协议时也得出了主体双方秘密性不一致的结论。

另外，在提到认证性时，我们也是指一个主体对另一个主体的认证，如 A 认证 B，或 B 认证 A。即一个主体想要确认他是否确实是和其所期望的主体进行了协议交互。

本文规定在角色描述中目标声明事件序列必须在通信事件序列之后，这是符合常规的，因为实际中我们都是在协议运行完成后才考察协议安全性的。

由角色表示可以看出，协议发起者发送的消息要被协议响应者接收，协议响应者发送的消息要被协议发起者接收，为表示它们之间的对应关系，引入通信关系的概念。

定义 13（通信关系） 对于所有的协议角色，通信关系 $<_{\pi}$ 定义为

$e1<_{\pi}e2$ 当且仅当 $\exists sid:stepid, P:princ, P':princ, m:msg$，使得

$e1=send(sid,P,P',m) \wedge e2=receive(sid,P',P,m)$。

为研究安全协议的认证特性，引入下面两个概念。

定义 14（角色事件序） 角色事件序 $<_R$ 为一个角色中事件上的全序，表示角色内部动作的先后次序。

定义 15（协议前序） 一个协议 P 的因果关系前序 $<_P$ 是 P 中所有角色事件序和通信关系的传递闭包，即 $<_P=(\bigcup_{R\in P}<_R \cup <_{\pi})^+$。

协议中所有角色的集合就构成了协议的定义：

定义 16（协议） 一个协议 P 是参数化角色的集合。

如某个协议有协议发起者和协议响应者两个角色，则 $P=\{Rinit, Rresp\}$。

4 协议执行模型

上一节形式化了协议的静态描述，一个协议可以描述为一个角色集合，这些角色规定了实际主体在系统中能够做什么。当协议投入实际运行时，一个角色可以被指定给多个主体运行多次，一个主体也可以并行或串行地运行多个角色。将抽象角色与具体主体关联起来的过程是参数角色的实例化过程。

定义 17（角色实例） 一个角色实例是参数化角色的实例化。即 $RoleInst=\langle InitKInst, EvenQInst\rangle$，其中 $InitKInst$ 为角色知识关于具体主体的实例，$EvenQInst=EvenInst_1 \cdot EvenInst_2 \cdot \cdots$ 为角色事件序列的实例，其中通信事件序列的实例又被称为角色运行 Run。定义 $Inst$ 为实例化函数，则 $Inst(InitK)$，$Inst(CommEvent)$ 和 $Inst(GoalClaimEvent)$ 分别表示角色初始知识、通信事件和目标声明事件的实例。函数 $roleof(EvenInst)$ 表示事件实例对应的角色实例。

定义 18（角色运行） 一个角色的通信事件序列的一次参数实例化结果称为该角色的一次运行，即 $Run=Inst\times CommEvent^*$，$Inst=RID\times\sigma$。其中 σ 为角色参数中部分变量的置换，RID 为该角色运行的唯一标识，可用符号 $rid(Inst)$ 表示实例化过程 $Inst$ 对应的角色运行标识。在角色运行中的事件称为运行事件 $RunEvent$，其形式为 $\sigma(e)$，$e\in CommEvent$。函数 $role(RunEvent)$ 表示运行事件对应的角色实例，$rid(RunEvent)$ 表示运行事件对应的角色运行标识。$aprinc(Run)$ 表示角色运行的主动主体，$aprinc(RunEvent)$ 和 $pprinc(RunEvent)$ 分别表示运行事件的主动主体和被动主体。

$$Init(A,b,N_a,n_b)\#1=send(1,A,b,penc([N_a,A],pk(b))) \cdot receive(2,A,b,penc([N_a,n_b,b],pk(A))) \cdot send(3,A,b,penc(n_b,pk(b)))$$

即为 NSL 协议的协议发起者角色 $Init(a,b,n_a,n_b)$ 在置换 $\{a\mapsto A\}\cup\{n_a\mapsto N_a\}$ 下的一次运行，主体 A 使用其产生的随机数 N_a 执行了协议发起者的角色，角色运行标识为 1，它自动标于角色名后，以 # 号连接。角色运行标识为全局符号，即所有角色的运行统一编号。

定义 19（事件内容） 运行事件中操作的项称为事件内容，具体表示为：

$cont(\sigma(send/receive(sid,p,p',m)))=\sigma(m)$。

定义 20（协议会话） 协议会话是完成一次协议所需的一个角色运行集合。

定义 21（协议场景） 协议场景是一个协议会话集合，也是一个角色运行的多重集，用于确定哪几个协议会话同时运行，各主体都扮演哪些角色。

协议场景中的每个角色运行应满足“变量起源假设”，即未实例化的变量首次一定出现在接收事件的消息项中，因为一个协议主体只在发送消息之前产生随机数等新项，自己产生的项对他来讲肯定是已知的常量，而接收到的消息中才会有不能确定的变量。

协议运行时，场景中不同的角色运行通过网络进行异步通信。根据我们的协议环境假设，主体发送的消息全部发给了攻击者，而主体接收到的消息全部来自于攻击者。令 IK 表示攻击者的知识，则当协议主体执行发送事件 $send(sid,P,P',m)$ 时，其发出的消息 m 被加入到攻击者的知识库中，即有 $IK=IK\cup\{m\}$；当协议主体执行接收事件 $receive(sid,P,P',m)$ 时，其接收的消息 m 是攻击者转发或伪造的，即 m 必须满足协议规定的格式要求，而且是攻击者基于其已有知识和推理计算能力导出的，记为 $IK\vdash m$。

定义 22（迹） 协议的一个运行迹是已被执行的运行事件序列，记为 $TR=\langle ev_1 \cdot ev_2 \cdot \cdots\rangle$。一个协议 P 所有迹的集合记为 $TRs(P)$。

将事件 ev 加入迹 TR 生成新迹 $\langle TR \cdot ev\rangle$。包含前 i 个事件的前缀迹表示成 TR_i，$TR_0=\langle\rangle$。函数 $index(ev)$ 返回一个运行事件 ev 在迹中的索引值，迹中存在序 $<_{\pi}$。函数 $last$ 和 $length$ 分别表示迹中的最后一个运行事件和迹的长度。即 $last(\langle TR \cdot ev\rangle)=ev$，空迹的 $last(\langle\rangle)$ 无定义；$length(\langle\rangle)=0$，$length(\langle TR \cdot ev\rangle)=length(\langle TR \cdot ev\rangle)+1$，且对于 $m\geqslant length(TR)$，有 $TR_m=TR$。

定义 23（协议运行状态） 协议运行某时刻的状态 S 定义为一个三元组：$S=\langle Sc, IK, UE\rangle$，其中 Sc 为一个协议运行

场景，IK 为此刻攻击者的知识，UE 为场景中尚未执行的角色运行。协议运行初始状态为 $S_0=\langle\emptyset,IK,\emptyset\rangle$。

定义 24(协议运行状态迁移) 导致协议从一个状态迁移至另一个新状态的迁移规则定义为：

角色运行创建(*CreateRun*)：

$$\frac{u=inst(R.evenQ),R\in P,u\notin Sc}{\langle Sc,IK,UE\rangle\Rightarrow\langle Sc\cup\{u\},IK\cup\{inst.\sigma(R.InitK)\},UE\cup\{u\}\rangle}$$

消息发送(*SendMsg*)：

$$\frac{u=\sigma(send(sid,p,p',m)\cdot s)\in UE}{\langle Sc,IK,UE\rangle\Rightarrow\langle Sc,IK\cup\{\sigma(p,p',m)\},UE\backslash\{u\}\cup\{\sigma(s)\}\rangle}$$

消息接收(*ReceiveMsg*)：

$$\frac{u=\sigma(receive(sid,p,p',m)\cdot s)\in UE\wedge fakeinsert(sid,p\sigma,p'\sigma,m\sigma)}{\langle Sc,IK,UE\rangle\Rightarrow\langle Sc,IK\cup\{\sigma(sid,p,p',m)\},UE\backslash\{u\}\cup\{\sigma(s)\}\rangle}$$

目标声明(*GoalClaim*)：

$$\frac{u=\sigma(e\cdot s)\in UE,e\in GoalClaimEvent}{\langle Sc,IK,UE\rangle\Rightarrow\langle Sc,IK,UE\backslash\{u\}\cup\{\sigma(s)\}\rangle}$$

其中 $IK\cup\{\sigma(p,p',m)$ 刻画了攻击者从网络获取信息的能力，事件 $fakeinsert(sid,\sigma p,\sigma p',\sigma m)$ 表示攻击者向网络插入了消息，其成功的前提条件是 $IK\vdash\sigma m$。若由于 $\sigma m\in IK$ 而使得 $IK\vdash\sigma m$，此时 $fakeinsert(sid,\sigma p,\sigma p',\sigma m)$ 表示攻击者转发消息的能力，否则 $IK\vdash\sigma m(fakeinsert(sid,\sigma p,\sigma p',\sigma m)$ 表示攻击者伪造消息的能力。

在攻击者从 IK 构造 σm 时，需要一个攻击者演绎系统，攻击者的演绎能力直接决定着攻击者是否可以成功伪造消息，从而决定针对协议的攻击是否成功。在此，将协议主体在收到消息后对消息进行格式检查的责任另派给了攻击者。即攻击者在由 IK 推导 m 时必须考虑格式和类型匹配问题。

5 应用实例

下面以简化的 NSL 协议为例对本文提出的模型进行具体说明。NSL 协议是由 G. Lowe 改进的 Needham-Schroeder 公钥协议，该协议的意图是利用可信密钥服务器和公钥实现两个主体之间的相互认证[19]。具体形式为

1) $A\rightarrow S:A,B$

2) $S\rightarrow A:\{pk(B),B\}sk(S)$

3) $A\rightarrow B:\{Na,A\}pk(B)$

4) $B\rightarrow S:B,A$

5) $S\rightarrow B:\{pk(A),A\}sk(S)$

6) $B\rightarrow A:\{Na,Nb,B\}pk(A)$

7) $A\rightarrow B:\{Nb\}pk(B)$

主体 A,B 与可信服务器之间只交换主体身份和公钥等公开信息，假定任何人都可以随时得到这些信息，因此可以将第 1,2,4,5 步从协议中略去，得到如下简化的 NSL 协议：

1) $A\rightarrow B:\{Na,A\}pk(B)$

2) $B\rightarrow A:\{Na,Nb,B\}pk(A)$

3) $A\rightarrow B:\{Nb\}pk(B)$

该协议有协议发起者和协议响应者两个角色，在仅考虑秘密性的情况下，用前面的项定义形式表示这些角色，得到协议的静态描述：

$P=\{Rinit,Rresp\}$

$Rinit=\langle\{a,b,n_a,pk(a),sk(a),pk(b)\},Init(a,b,n_a,n_b)\rangle$

$Init(a,b,n_a,n_b)=send(1,a,b,penc([n_a,a],pk(b)))\cdot receive(2,a,b,penc([n_a,n_b,b],pk(a)))\cdot send(3,a,b,penc(n_b,pk(b)))$

A 的秘密性目标声明为 $GoalClaimEvent=Secret(a,n_a)$

$Rresp=\langle\{a,b,n_b,pk(a),sk(b),pk(b)\},Resp(a,b,n_a,n_b)\rangle$

$Resp(a,b,n_a,n_b)=receive(1,b,a,penc([n_a,a],pk(b)))\cdot send(2,b,a,penc([n_a,n_b,b],pk(a)))\cdot receive(3,b,a,penc(n_b,pk(b)))$

B 的秘密性目标声明为 $GoalClaimEvent=Secret(b,n_b)$

协议运行初始状态为 $S_0=\langle\emptyset,IK_0,\emptyset\rangle$，其中，攻击者的初始知识为 $IK_0=\{I,pk(I),sk(I),1:stepid,2:stepid,3:stepid\}\cup\{A,B,pk(A),pk(B)\}$。此时，主体 A,B 均为可信主体，若 B 为不可信主体，IK_0 中还应包含 $\{N_b,sk(B)\}$。IK_0 中包含 $\{1:stepid,2:stepid,3:stepid\}$ 是因为攻击者知道协议的步骤数。

从这个初始状态出发，通过前面定义的协议执行模型得到该协议的动态运行描述。最后，通过检查每个协议主体的运行迹中 n_a(或 n_b)是否在攻击者的知识 IK 中出现来判断对于 A(或 B)来讲 n_a(或 n_b)的秘密性能否被满足。

结束语 本文分析了实际网络环境中安全协议的运行特点，提出了安全协议建模分析的两点基本假设，将攻击者和网络环境合一，强化了攻击者能力，并对协议主体进行了划分。在此基础上，提出了基于语义的安全协议形式化模型，重点研究了其中的静态规范描述模型和动态运行模型。该模型具有描述精确、具有合理可靠的可证明语义、便于实现自动化验证的特点。在下一步的研究工作中，将重点研究具有强大计算能力和网络控制能力的攻击者模型，对秘密性、存在性、弱一致性、非单射一致性、单射一致性、非单射同步性和单射同步性等安全属性进行形式定义，最终实现安全协议的自动化验证。

参 考 文 献

[1] Burrows M, Abadi M, Needham R. A Logic of Authentication. ACM Transactions in Computer Systems, 1990, 8(1): 18-36

[2] Dolev D, Yao A C. On the Security of Public Key Protocols. IEEE Transactions on Information Theory, 1983, 29(2): 198-208

[3] Schneider S. Verifying Authentication Protocols in CSP. IEEE Transaction on Software Engineering, 1998, 24(9): 741-758

[4] Broadfoot P J, Roscoe A W. Internalising Agents in CSP Protocol Models//Proceedings of Workshop on Issues in the Theory of Security. 2002

[5] Clarke E M, Jha S, Marrero W. Verifying Security Protocols with Brutus. ACM Transactions on Software Engineering and Methodology, 2000, 9(4): 443-487

[6] Woo T Y C, Lam S S. A semantic model for authentication protocols//Proceedings of the IEEE Symposium on Research in Security and Privacy. Oakland, CA, 1993: 178-194

[7] 赵宇，袁霖，王亚弟，等. 一种改进的 Woo-Lam 安全协议模型. 计算机应用，2006，26(9)：2116-2120

[8] Fabrega F J T, Hertzog J, Guttman J. Strand Spaces: Proving Security Protocols Correct. Journal of Computer Security, 1999, 7(2/3): 191-230

(下转第 136 页)

结束语 本文指出了传统的代理环签名所存在的问题，同时以RSA算法为基础，提出了一种新型的代理环签名方案。本文详细阐述了系统建立、密钥生成、签名生成及签名认证的过程，并给出详细的分析及证明。本签名方案具有无条件匿名性、不可伪造性、易于验证、可区分性等诸多代理环签名的良好特性，较好地满足了代理环签名要求的所有性质。本方案采用了无证书公钥体制，将用户的公钥与本人身份相结合，从而解决了密钥托管和证书管理问题，大大减轻了系统中Trent的工作量。与传统的方案相比，本方案具有更加良好的高效性、可计算性以及可扩展性、自证明性等特点。本签名方案可以广泛应用于电子投标、电子支付、移动自组网认证等领域。

参 考 文 献

[1] Rivest R L, Shamir A, Tauman Y. How to Leak a Secret // 7th International Conference on the Theory and Application of Cryptology and Information Security. LNCS 2248. Springer-Verlag, 2001: 552-565

[2] Chaum D, Heyst V E. Group signatures [A] // Proc. CRO-CRYPT'91[C]. Springer-Verlag, 1991:257-265

[3] Dodis Y, Kiayias A, Nicolosi A, et al. Anonymous identification in ad-hoc groups[A] // Proc. Eurocrypt'04[C]. Springer-Verlag, 2004:609-626

[4] Tsang P P, Wei V K. Short linkable ring signatures for E-voting, E-cash and attestation[A] // ISPEC 2005[C]. Springer-Verlag, 2005:48-60

[5] Masahiro MAMBO Keisuke USUDA Eiji OKAMOTO. Proxy Signatures: Delegation of the Power to Sign Messages. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, E79-A(9):1338-1354

[6] Lee Byoungcheon, Kim Heesun, Kim Kwangjo. Strong Proxy Signature and its Applications // Proc. of SCIS 2001. Japan: [s. n.], 2001:603-608

[7] Zhang F, Naini R, Lin C Y. New proxy signature, proxy blind signature and proxy ring signature schemes from bilinear pairings[EB/OL]. [2003-05-20]. Cryptology ePrint Archive. http://eprint.iacr.org/2003/

[8] Lang Weimin, Yang Zongkai, Cheng Wenqing, et al. A New ID-based Proxy Ring Signature Scheme. Journal of Harbin Institute of Technology, 2004, 6(2):10-15

[9] Shamir A. Identity-based cryptosystems and signature schemes, Advances in Cryptology // Proceedings of CRYPTO 84. volume 196 of Lecture Notes in Computer Science. Springer-Verlag, 1985:47-53

[10] Mao Wenbo. 现代密码学理论与实践 (Modern Cryptography: Theory and Practice). 电子工业出版社, 2004:294-295

[11] 禹勇, 等. 一个有效的代理环签名方案(An Efficient Proxy Ring Signature Scheme). 北京邮电大学学报, 2007, 30(3):23-26

[12] Bellare M, Rogaway P. The exact security of digital signatures - How to sign with RSA and Rabin // Maurer U. ed. Advances in Cryptology-Proceeding of EUROCRYPT'96. Lecture Notes in Computer Science 1070. Springer-Verlag, 1996:399-416

[13] Coron J S, Joye M, Naccache D, et al. Universal padding schemes for RSA // Yung M. ed. Advances in Cryptology-Proceedings of CRYPTO'02. Lecture Notes in Computer Science 2442. Springer-Verlag, 2002:226-241

[14] 饶方宇. 可证明安全密码系统之研究(Study of Provable Secure Cryptosystems and Signature Schemes). 台湾国立中山大学咨询工程学系硕士论文. 2005

[15] Bellare M, Rogaway P. Random oracles are practical: a paradigm for designing efficient protocols // First ACM Conference on Computer and Communications Security. New York, ACM Press, 1993:62-73

[16] Herranz J, Saez G. Forking Lemmas for Ring Signature Schemes // INDOCRYPT 2003. LNCS 2904. Springer-Verlag, 2003:266-279

(上接第118页)

[9] Fabrega F, Herzog J, Guttman J. Honest Ideals on Strand Spaces // Proceedings of the IEEE Computer Security Foundations Workshop XI. 1998

[10] Guttman J, Fabrega J T. Authentication Tests and the Structure of Bundles. Theoretical Computer Science, 2001

[11] Cervesato I, Durgin N, Lincoln P, et al. A Meta-notation for Protocol Analysis // Proceedings of 12th IEEE Computer Security Foundation Workshop. Mordano, Italy, 1999:55-72

[12] Cervesato I. A Specification Language for Crypto - protocols based on multiset rewriting, dependent types and subsorting // Delzanno G, Etalle S, Gabbrielli M, eds. Proc. of the Workshop on Specification, Analysis and Validation for Emerging Technologies-SAVE'01. Paphos, 2001:1-22

[13] 张畅, 王亚弟, 韩继红, 等. 一种改进的密码协议形式化模型. 软件学报, 2007, 18(7):1746-1755

[14] Abadi M, Gordon A D. A Calculus for Cryptographic Protocols: The Spi Calculus // Proceedings of 4th ACM Conference on Computer and Communications Security. Zurich, Switerland, 1997

[15] 赵宇, 王亚弟, 韩继红. 基于Spi演算的SSL3.0协议安全性分析. 计算机应用, 2005, 25(11):2515-2520

[16] 季庆光, 冯登国. 对几类重要网络安全协议形式模型的分析. 计算机学报, 2005, 28(7):1071-1082

[17] Denning D E, Sacco G M. Timestamps in Key Distribution Protocols. Commum. ACM, 1981, 24(8):533-536

[18] Paulson L C. The Inductive Approach to Verifying Cryptographic Protocols. Journal of Computer Security, 1998, 6(1):85-128

[19] Lowe G. Breaking and Fixing the Needham - Schroeder Public Key Protocol using FDR // Proceedings of TACAS, Lecture Notes in Computer Science 1055. Passau, Germany: Springer-Verlag, 1996:147-166

[20] Engels A G, Mauw S, Reniers M A. A hierarchy of communication models for message sequence. Charts Science of Computer Programming, 2002, 44(3):253-292

[21] Robinson J A. A Machine-oriented Logic Based on the Resolution Principle. Jourbal of the ACM, 1965, 12(1):23-41