第 36 卷 第 7 期 计 算 机 工 程 2010 年 4 月
Vol.36 No.7 Computer Engineering April 2010
·软件技术与数据库· 文章编号：1000—3428(2010)07—0082—02 文献标识码：A 中图分类号：TP309

一种安全协议的形式化分析方法

王 昕，袁超伟
(北京邮电大学信息与通信工程学院，北京 100876)

摘 要：对快速、高效的形式化分析安全协议进行研究，提出“信任域”的概念。采用与图形化相结合的分析方法，使得协议流程的推导过程清晰、直观。该方法直接分析协议参与主体的信任域，简化分析过程和步骤。实验结果表明，与传统方法相比，该方法更快速、直观，并能为分析协议的冗余性提供具体方法和依据。

关键词：形式化分析；安全协议；BAN 逻辑；NSSK 协议

Formal Analysis Method of Security Protocol

WANG Xin, YUAN Chao-wei
(School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876)

【Abstract】This paper researches on quick and efficient formal analysis of security protocol, and presents the concept of trust domain. Diagrammatic analysis is adopted, which makes deduction of protocol more clear and intuitive. This method analyzes the trust domain of protocol entity directly, which makes analysis easier. Experimental results show that the method is faster and more intuitive compared with traditional methods. It provides ways and basis for finding redundancies of security protocols.

【Key words】formal analysis; security protocol; BAN logic; NSSK protocol

1 概述

安全协议的设计和分析十分复杂，采用形式化分析方法可以有效地降低分析过程的复杂度，使得安全协议的分析简单、规范和实用。GNY, AT, VO 和 SVO 等 BAN 类逻辑方法[1]在安全协议的形式化分析中被广泛采用[2]，新的形式化分析方法也不断出现[3-4]。但是，BAN 类逻辑分析方法将论证的空间落在了协议的参与主体上，论证的焦点是参与协议的主体，这样容易造成分析过程中对于主体的逻辑判断加入人为的误判，与事实不符[5]。文献[6]中在构造类别代数的基础上引入了“规则集合”、“知识集合”的概念，对于协议的规范化描述起到重要作用，但对其在协议形式化分析的应用并未涉及。文献[7-8]以 BAN 类逻辑为基础，采用逻辑分析与模型检测相结合的方式完成分析，扩充了分析范围，但形式化方面仍较欠缺。实际上，协议设计和分析的目标应是参与协议主体的信任状态，即分析过程中应更关注主体所信任的事物，随协议的执行而发生变化。原有 BAN 类逻辑初始假设人为设定，是否合适需要在分析完成后进行人工检验。文献[9]同样以逻辑分析为基础，将安全协议看作是若干次带有参数的挑战-响应交互，分析的焦点是协议参与主体是否能够成功响应，该方法成功分析 NSL(Needham-Shroeder-Lowe)协议[10]，首次从参与者的角度分析协议，沿用了 BAN 类逻辑的符号推导，但在协议分析方面仍较抽象和繁琐。

本文提出的形式化方法就是以协议主体信任域为焦点，在基本规则基础上通过证明推导出常用定理，使得协议分析不仅能够更直观和快速的完成，还能使得协议的冗余和错误假设在协议分析之后直观地显现。

2 语法、语义和规则

信任域是协议主体所信任事物构成的集合。该集合随着协议的执行不断扩张，直至协议结束。

(1)语法和语义

X,Y：参与协议的主体。

XY：X 与 Y 之间共享对称密钥。

$\overline{M}$：M 是新鲜的。

$(M)_{XY}$：用 X 与 Y 共享的对称密钥加密的消息 M。

$[M]^X$：主体 X 相信 M。

$|M|_X$：主体 X 控制 M。

$(M)^X$：主体 X 曾发送过 M。

~~M~~：被使用过的条件 M。

$\boxed{\overset{X:}{\overline{\overline{\Psi}}}}$：接收到新消息前，主体 X 的信任域为 Ψ，Ψ 随着协议分析的进行不断扩充。

$\boxed{\overset{X:}{\overline{\Phi}}}$：接收到新消息后，$\Phi$ 为主体 X 的分析推导。

$\boxed{\overset{X:}{\overline{\overline{\Omega}}}}$：经过推导之后，主体 X 得出的结论为 Ω。Ω 会在下一次推导中合并到 Ψ。

$\xrightarrow{M}$：发送消息 M。

(2)推理规则

与 BAN 类逻辑[5,11]的推理规则类似，在主体 X 信任域中有以下 4 条规则：

(1) $XY+(M)_{XY}\Rightarrow(M)^Y$，主体 X 相信 X 与 Y 共享密钥 XY，且收到了用 XY 加密的消息 M，则 X 相信 Y 发送过 M。

(2) $(M)^X+\overline{M}\Rightarrow[M]^X$，主体 X 相信 Y 曾经发送过 M，且 M 是新鲜的，则主体 X 认为 Y 相信 M。

作者简介：王 昕(1982－)，男，博士研究生，主研方向：信息安全，移动支付；袁超伟，教授、博师生导师

收稿日期：2009-12-10 **E-mail：**wangxinsmile@gmail.com

(3) $[M]^X+|M|_X \Rightarrow M$，主体 X 认为 Y 相信 M，且认为 Y 控制 M，则 X 也相信 M。

(4) $\overline{M_1}+(M_1,M_2,\cdots)\Rightarrow(\overline{M_1},\overline{M_2},\cdots)$，新鲜性具有“传染”性。

但与 BAN 类逻辑不同的是，以上 4 条推理规则直接应用在参与协议主体的信任域中。

定理 $(M_1,M_2,\cdots)^X+\overline{M_i}\Rightarrow[M_1,M_2,\cdots]^X$，($i$=1,2,…)，即主体 X 收到消息 $M_1,M_2,\cdots$，若其中任意一个消息 M_i (i=1,2,…) 是新鲜的，则主体 X 相信 $M_1,M_2,\cdots$。

证明如下：

(1) $(M_1,M_2,\cdots)+\overline{M_i}\Rightarrow(\overline{M_1},\overline{M_2},\cdots)$，($i$=1,2,…)，由推理规则(4)可得。

(2) $(\overline{M_1},\overline{M_2},\cdots)+(M_1,M_2,\cdots)^X\Rightarrow[M_1,M_2,\cdots]^X$，由推理规则(2)可得。

3 协议分析步骤

协议分析的步骤如下：

(1)协议理想化，将协议的消息以符号形式描述。

(2)确定协议主体信任域初始化状态。

(3)将主体的信任域与收到的消息合并，应用上述规则在信任域下方进行协议分析推导。

(4)标记推导过程中使用到的初始状态条件，将推导出的重要结论加入主体信任域中。

(5)重复步骤(3)、步骤(4)，直到完成推导目标。

4 协议分析应用

例 1 为便于比较，用本文的方法对文献[5,11]的 NSSK 协议[12]进行分析。

协议理想化如下：

S->A: $(N_a,AB,(AB)_{BS})_{AS}$

A->B: $(AB)_{BS}$

B->A: $(N_b,AB)_{AB}$

A->B: $(N_b,AB)_{AB}$

文献[5,11]的 NSSK 协议分析过程如图 1 所示。

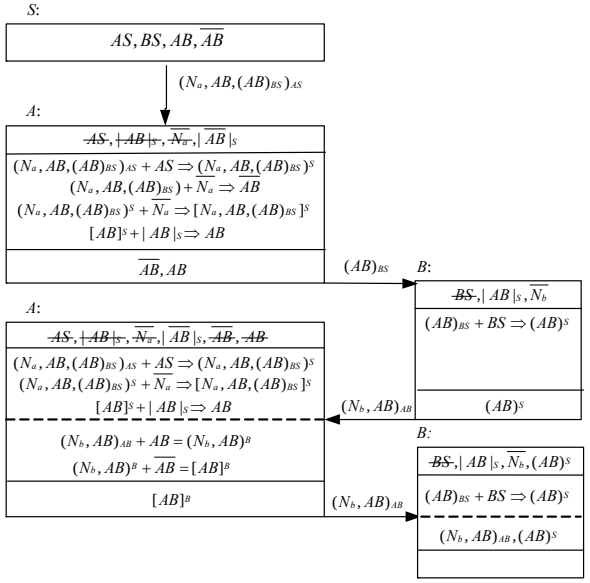

图 1 文献[5, 11]中的 NSSK 协议分析过程

通过观察分析过程能够发现，对于协议主体 B 的信任域只得到了 $(AB)^S$，即主体 B 只能相信可信第三方 S 曾发送过密钥 K_{AB}，但不能判断 K_{AB} 的新鲜性，存在受到重放攻击的可能。这与文献[5, 11]中对 NSSK 协议的分析结果是相符的。协议中止在交互的第 3 步，协议继续进行的条件是 B 的信任域中含有 AB，而这一条件在协议设计时被认为是在第 1 步执行后可以得到的，B 的信任域中未得出 AB 的原因是消息中缺少新鲜性，无法得到 $[AB]^S$ 的结论，进而无法推出 $AB\in\Omega_B$ 的结论。文献[11]的 NSSK 改进协议去掉了上述条件，但并没有给出形式化的理论依据。文献[13]中的形式化分析对于 NS 协议的改进并未提供直接帮助，改进是通过形式化分析后的人工分析得到的。从新方法对该协议的分析可以看到，相对于原有 BAN 类逻辑，新方法不仅可以发现协议缺陷，也可以较直观地找到缺陷原因。进一步分析不难发现，$|\overline{AB}|_S$ 的条件过强，对于协议的后续分析会产生误导，这也是在改进的 NSSK(1)中去掉其的原因。

例 2 为了进一步验证本文方法的有效性，下面用本文方法对文献[11]中提到的 NSSK(1)协议进行分析。

协议理想化如下：

S->A: $(N_a,AB,(N_a,AB,T_s)_{BS})_{AS}$

A->B: $(N_a,AB,T_s)_{BS}$

B->A: $(N_b,N_a,AB)_{AB}$

A->B: $(N_b,AB)_{AB}$

文献[11]中的 NSSK(1)协议分析过程如图 2 所示。

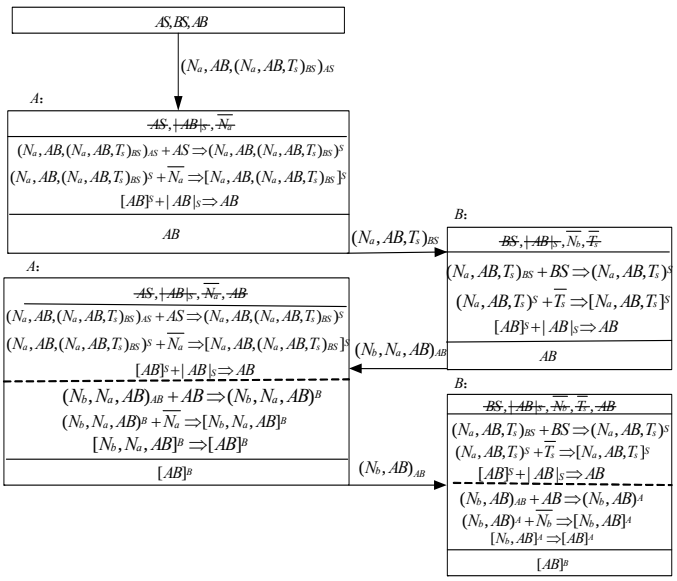

图 2 文献[11]中的 NSSK(1)协议分析过程

通过分析，可以直观地看到最终协议主体 A 与 B 的信任域 Ψ 都包含 AB，即 $AB\subset\Psi$。同时也得到了更强的信任关系：$[AB]^B\subset\Psi_A$，$[AB]^A\subset\Psi_B$，即达到文献[5]中所述的二级信仰。

5 本文方法与BAN类逻辑分析方法的比较

与 BAN 类逻辑分析方法相比，本文方法具有较强的可视化特点，并且对于一些可疑假设能够直观展现，具体比较如表 1 所示。

表 1 本文方法与 BAN 类逻辑分析方法的比较

项目	BAN 类逻辑	本文方法
语法与语义	仅限于符号	符号与图形结合
推理规则	基本规则	基本规则与定理结合
推导过程	抽象，与具体协议关联紧密	格式化，可与具体协议脱离
发现可疑假设	依赖推导后的人工分析	推导后能够直观体现
直观性	弱，二次检验较难	强，二次检验较易
推导速度	慢，逻辑判断多，公式化推导少	快，逻辑判断少，形式化推导多
自动化实现	适合于面向过程语言	适合于面向对象语言

（下转第 86 页）