ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.17, No.1, January 2006, pp.148−156
DOI: 10.1360/jos170148

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

基于认证测试的安全协议分析*

杨 明+, 罗军舟

(东南大学 计算机科学与工程系 网络室,江苏 南京 210096)

Analysis of Security Protocols Based on Authentication Test

YANG Ming+, LUO Jun-Zhou

(Network Laboratory, Department of Computer Science and Engineering, Southeast University, Nanjing 210096, China)

+ Corresponding author: Phn: +86-25-83791010, Fax: +86-25-83794838, E-mail: yangming2002@seu.edu.cn, http://www.seu.edu.cn

Yang M, Luo JZ. Analysis of security protocols based on authentication test. *Journal of Software*, 2006,17(1): 148−156. http://www.jos.org.cn/1000-9825/17/148.htm

Abstract: Authentication Test is a new type of analysis and design method of security protocols based on Strand space model, and it can be used for most types of the security protocols. However, as a Strand space model, it is inclined to be used for the proof of correctness, and is relatively weaker for incorrectness analysis. This paper proposes the concepts of Enhanced Authentication Test (EAT) and the correspondence function that can solve the problem. Compared with the original concept, the new approach is more formal and can make protocol analysis easier both by hand and automatically.

Key words: network security; protocol analysis; Strand space model; authentication test; authentication logic

摘 要: 认证测试是一种新型的在 Strand 空间模型基础上发展而来的安全协议分析与辅助设计技术,可用于大部分协议的关联属性的分析;但是与 Strand 空间模型一样,它主要用于协议正确性证明,在协议为何不正确以及如何进行改进这个问题上处理分析能力较弱.在认证测试概念的基础上,结合逻辑分析的优点,提出了增强型认证测试 EAT(enhanced authentication test)和 Correspondence 函数等概念来对安全协议进行关联属性的分析,很好地解决了这一问题.与原有技术相比,该方法更为形式化,协议分析人员可以很方便地进行手动分析,并且更有利于协议分析自动化工具的实现.

关键词: 网络安全;协议分析;Strand 空间模型;认证测试;认证逻辑
中图法分类号: TP309 文献标识码: A

安全协议,又称为密码协议,它的形式化分析是网络安全研究领域的重要内容.本文在已有的国内外研究成果的基础上,提出了增强性认证测试(enhanced authentication test,简称 EAT)和 Correspondence 函数的概念,能够很好地解决协议关联性(correspondence,也有称一致性、对应性)的分析问题.

本文第 1 节详细分析了研究背景和国内外已有的一些研究成果,以及本文工作与这些研究的关系和差别.

* Supported by the National Natural Science Foundation of China under Grant No.90412014 (国家自然科学基金); the Jiangsu Provincial Key Laboratory of Network and Information Security under Grant No.BM2003201 (江苏省"网络与信息安全"重点实验室); the Jiangsu Provincial High-Tech Research Program under Grant No.BG2004036 (江苏省高技术研究项目)

Received 2004-06-03; Accepted 2005-07-28

第 2 节简单介绍了认证测试(authentication tests,简称 AT)的基本概念.第 3 节详细描述了 EAT 和 Correspondence 函数以及相应的算法.第 4 节以 Kerberos 协议和 OR 协议为例,演示了如何使用本文提出的方法进行协议分析,并给出了 Kerberos 协议的正确性证明和 OR 协议的错误根源分析.文章最后简要总结了 EAT 和 Correspondence 函数在协议分析中的意义和作用,并给出了下一步工作的设想.

1 研究背景

1.1 概 述

Strand 空间模型[1,2](Strand space model,简称 SSM)是 Thayer Fàbrega,Herzog 和 Guttman 等人在 1998 年提出的安全协议分析模型,它借助图论的方法描述协议的执行过程.协议主体的行为序列构成 Strand,而 Strand 空间则是所有 Strand 的集合;不同协议主体的 Strand 之间通过消息数据的收发相互关联从而形成线束(bundle).在线束的基础上,不同节点之间的偏序关系使得存在极小元,从而又产生了一种类似于归纳法[3]的协议安全性的证明方法.Strand 空间模型能够准确描述协议执行过程中事件的先后顺序关系,为研究人员提供了一种全新的协议分析方法,但是它主要用于协议正确性的证明,研究人员无法应用它来查找协议中存在的漏洞,并进一步获取为何存在漏洞以及如何进行改进等方面的有用信息.

认证测试[4,5]是在 Strand 空间模型的基础上发展而来的一种基于挑战-响应概念的协议分析技术.它的基本思想是,如果协议的一个主体发送了包含某个特定值 a(明文或者密文形式)的消息,并在之后收到改变了形式后的 a 值(被加密或者解密),那么可以肯定存在一个持有相应密钥的普通协议主体参与了该协议的执行.在文献[4]中,对应于挑战方-响应方对值 a 不同的加解密处理,作者 Guttman 和 Thayer Fàbrega 将认证测试划分为 3 种类型:Incoming Test(IT),Outgoing Test(OT)和 Unsolicited Test(UT).AT 概念提供了简洁、强大的协议分析能力,并且能够应用于安全协议的设计[6];但是,与 Strand 空间模型一样,它对协议不正确的判断基于 Strand 参数的不一致性,而如何判断参数的不一致并没有提供明确而可行的分析方法.

本文在 AT 的基础上,提出了增强性认证测试 EAT 和 Correspondence 函数的概念.前者提供了具有很强可操作性的 Strand 参数一致性判断和协议主体之间关联性分析的方法,后者提出了协议关联性的量化表示.本文的基本思路是:将协议的执行过程以 EAT 为基本单位分段,然后在协议主体两两之间根据 Strand 参数一致性和认证测试概念逐段分析其关联性,从而最终达到分析整体关联性的目的.对于正确的安全协议,给出其正确性的证明;对于不能满足关联性的协议,分析出错误的具体原因,即哪一个参数在哪一段 EAT 上不满足一致性.需要指出的是,安全协议分析通常把协议的安全性归结为保密性、关联性[7]以及其他一些与电子商务相关的属性(如不可否认性、公平性等),本文只讨论其中的关联属性,因此,在论文中提及的正确性都是针对协议的关联性而言的.

1.2 相关工作

Strand 空间模型论文的发表引起了广泛的关注,研究人员纷纷提出自己的改进或者应用方案,除了前面提到的认证测试技术之外,具有代表性的还有以下几种:(1) 将 Strand 空间模型与状态检测技术相结合,如 Song,Berezin 和 Perrig 等人提出的高性能自动化安全协议分析技术 Athena[8,9],该技术对 Strand 空间模型进行了扩展,并综合应用了模型检测和定理证明的方法,很好地解决了状态爆炸问题,可以完全自动化地证明许多安全协议的正确性.当 Athena 对协议分析完毕之后,或者给出其正确性证明,或者给出反例;(2) 利用 Strand 模型能够准确描述协议执行过程中事件先后关系的优势,将其作为其他分析方法的语义模型,例如 Paul Syverson 在文献[10]中将 Strand 概念应用于 BAN 类逻辑,替代了原有的 BAN 逻辑的语义模型;(3) 转换 Strand 的描述形式,以期采用一些现有的成熟的分析技术.例如 Cervesato,Durgin,Kanovich 和 Scedrov 等人在文献[11]中提出用先序线性逻辑(first-order linear logic)公式表达 Strand 结构,可以很好地描述协议执行过程中的状态、事件和资源;(4) 对 Strand 空间模型本身进行扩展.例如国内学者季庆光、卿斯汉、周永彬和冯登国等人在文献[12]中提出了通用 Strand 空间模型(GSSM),以及 Oracle Strands 等新的 Strand 类别概念,不仅可以证明协议的正确性,还可以高

效地构造出对错误协议的攻击.

与这些工作相比,本文提出的协议分析技术充分利用 Strand 空间模型对协议正确性的证明能力与认证测试技术使用的简洁性,从另一个角度来对这些分析方法进行了完善和改进.当采用 EAT 和 Correspondence 函数分析完一个安全协议后,我们或者可以证明该协议的正确性,或者可以指出协议在哪一步 EAT 上的参数不一致,从而可以辅助协议设计人员对存在的缺陷进行修正.主流的安全协议分析方法可以分为 3 类[13]:基于推理结构性方法(如 BAN 逻辑[14])、基于攻击结构性方法(如线性逻辑[11])和基于证明结构性方法(如 Strand 空间模型).本文提出的方法与 Strand 空间模型一样,属于证明结构性方法,但是它不仅可以用于正确性证明,还能向分析人员提供直观明了的协议缺陷信息.此外,将协议以 EAT 为基本单位分段进行分析,该方法像 BAN 逻辑一样简单、易用,并且不存在协议正确性不能得到保证的问题,研究人员可以很方便地使用该方法对安全协议进行手动分析.

2 认证测试

为了简化描述,本文直接引用 Strand 空间模型的一些概念,它们的定义可以参阅文献[1,2].下面对认证测试作简单介绍[4,5].

2.1 基本概念

定义 **1**(消息组件 **Component**).

项 t_0 是 t 的组件,当且仅当 $t_0 \subset t$, t_0 不属于级联类型,并且对于任意 $t_1 \neq t_0$,如果有 $t_0 \subset t_1 \subset t$,那么, t_1 属于级联类型.消息组件或者是原子数据类型,或者是加密类型.

t_0 是节点 $n=\langle s,i\rangle$的新组件,当且仅当 t_0 是 $term(n)$ 的组件,并且不是任意其他节点 $n=\langle s,i\rangle$,$(j<i)$的组件.

定义 **2**(**Transformed Edge** 与 **Transforming Edge**).

边$\langle s,i\rangle \Rightarrow^{+} \langle s,j\rangle$是关于值 a 的 Transformed Edge,如果 a 在节点$\langle s,i\rangle$发送,并在节点$\langle s,j\rangle$从新组件中接收;边$\langle s,i\rangle \Rightarrow^{+} \langle s,j\rangle$为 Transforming Edge,如果 a 在节点$\langle s,i\rangle$接收,并在节点$\langle s,j\rangle$存在于新组件中发送.

定义 **3**(测试组件 **Test Component** 与测试 **Test**).

$t=\{h\}_K$ 是节点 n 关于 a 的测试组件,如果:① $a \subset t$,并且 t 是节点 n 的组件;② t 不是 Strand 空间Σ中任意其他常规节点的组件的子项.

如果值 a 在节点 n_0 唯一生成(Strand 空间模型中的“唯一生成”,相当于 BAN 类逻辑的 Fresh 概念),并且边 $n_0 \Rightarrow^{+} n_1$ 关于 a 的 Transformed Edge,那么我们称边 $n_0 \Rightarrow^{+} n_1$ 是 a 的一个测试.

定义 **4**(**outgoing test**,简称 **OT**).

边 $n_0 \Rightarrow^{+} n_1$ 是项 $t=\{h\}_k$ 关于值 a 的 OT,如果,(i) $n_0 \Rightarrow^{+} n_1$ 是 a 的一个测试;(ii) $K^{-1} \notin P$;(iii) a 不在节点 n_0 的除 t 以外的任何其他组件中出现;(iv) t 是节点 n_0 关于 a 的一个测试组件.

定义 **5**(**incoming test**,简称 **IT**).

边 $n_0 \Rightarrow^{+} n_1$ 是项 $t=\{h\}_k$ 关于值 a 的 IT,如果,(i) $n_0 \Rightarrow^{+} n_1$ 是 a 的一个测试;(ii) $K \notin P$;(iii) t 是节点 n_1 关于 a 的一个测试组件.

定义 **6**(**unsolicited test**,简称 **UT**).

接收节点(负节点) n 是项 $t=\{h\}_k$ 关于值 a 的 UT,如果:(i) t 是节点 n 中 a 值的测试组件;(ii) $K \notin P$.

2.2 3种认证测试

AT1. 假设 C 是某协议 Strand 空间的线束(bundle),节点 $n' \in C$,边 $n \Rightarrow^{+} n'$ 是项 t 关于值 a 的 OT,那么我们有:(i) 必然存在节点 $m, m' \in C$ 满足 t 是 m 的消息组件,并且边 $m \Rightarrow^{+} m'$ 是值 a 的 Transforming Edge;(ii) 如果假设值 a 只在节点 m' 的组件 $t_1=\{h_1\}_{K_1}$ 中出现, t_1 不是任何其他常规节点组件的子项,并且 $K_1^{-1} \notin P$, 那么必然存在一个包含 t_1 为组件的常规节点(负节点).

AT2. 假设线束 C 包含节点 n',边 $n \Rightarrow^{+} n'$ 是项 t' 关于值 a 的 IT,那么必然存在节点 $m, m' \in C$ 满足 t' 是 m' 的消息组件,并且边 $m \Rightarrow^{+} m'$ 是值 a 的 Transforming Edge.

AT3. 假设线束 C 包含节点 n,并且 n 是项 $t = \{h\}_K$ 的 UT,那么必然存在一个常规发送节点(正节点) $m \in C$ 满足 t 是 m 的消息组件.

3 EAT 与 Correspondence 函数

3.1 Strand参数一致性问题

在采用认证测试分析 NS 和 NSL 协议的关联属性时我们可以发现:两个协议分析获得的结论是基本相同的;二者的差别在于 Init Strand 的第 2 个参数,NSL 能够确定为 *B*,而 NS 无法确定,见表 1.为何会造成这种差别,如何分析出这种差别,在文献[4]中并没有给出说明.诚然,NSL 协议第 2 条消息中包含 *B* 能确定 Init Strand 的参数,但是不包含 *B* 也并不能直接说明此参数无法确定.

Table 1 Analysis of NS and NSL protocol using AT

表 1 AT 对 NS 和 NSL 协议的分析

Protocol	Difference	Conclusion with AT	Extra analysis results
NSL	{*Na*,*Nb*,*B*}*Ka*	Exists corresponding regular Strand, height=3	Init (*A*,*B*,*Na*,*Nb*)
NS	{*Na*,*Nb*}*Ka*	Exists corresponding regular Strand, height=3	Init (*A*,*,*Na*,*Nb*)

事实上,认证测试通过 Challenge-Response 方式来验证对方的身份,AT1~AT3 只明确地解决了“挑战方能够确认是自己希望的主体作出了响应”的问题,而通过对上面两个协议的分析,我们发现要保证协议的关联性,还进一步要求“挑战方能够确定该主体是针对自己作出的响应”.如果采用类似 BAN 逻辑的表示方法,我们可以把 NSL 和 NS 协议达到的认证效果分别表示为“B believes A response recently”和“B believes A response to B recently”,正是这些细微的差别造成了 NS 协议的失败.因此,要增强 AT 的查错能力,必须提供形式化的方法来进行 Strand 参数一致性的判断.

通过研究,我们把 Strand 参数划分为 3 种类型:Nonce、协商数据和主体标志.具体定义如下:

- Nonce

这里的 Nonce 通常包含两类:一类是由协议主体生成随机数值;另一类是时间戳.对于前者来说,参与协议执行的一方 *A* 生成某个随机 nonce,并将其以某种形式(加密或明文)发送给另一方 *B*,然后再接收改变了形式的 nonce,而这实际上就是前面所描述的 IT 和 OT;另一方面,根据第 2.2 节 UT 的定义,时间戳在协议中是其必要组成部分.因此,我们可以得出结论,Nonce 是认证测试的基础,符合 AT 必然满足 Strand 在 Nonce 类型参数上的一致性.

- 协商数据

安全协议执行的目的通常是为了在不同主体之间协商数据,这种数据或者是会话参数,或者是共享密钥,等等.假设 $t = \{h\}_K$ 是某一 AT 的测试组件,那么要保证 AT 涉及的协商数据 x 的 Strand 参数一致性,那么必须有 $x \subset t \vee x = K$.

- 主体标志

主体标志是 Strand 参数一致性分析中最重要的一环,在 Strand 空间模型以及 AT 有关文献中都没有得到足够的重视.事实上,主体标志的不一致正是大部分回放攻击存在的原因.以三方的密钥分配协议(*S* 为 *A*,*B* 分配会话密钥)为例,我们不仅要考虑到 *A* 能确认是 *S* 在与自己会话,还要能确认 *S* 知道是为 *A* 和 *B* 分配密钥.综合 Nonce 和协商类型参数的一致性问题,那么 *A* 就需要确认 S 知道 *k*(协商数据)是为 *A* 和 *B*(主体标志)在当前轮(nonce)分配的密钥.

相对来说,Strand 在 Nonce 和协商类型参数上的一致性判断较为简单,下面的讨论主要围绕主体标志类型参数进行.

定义 7(身份标志 **id**).

在协议 Strand 空间Σ中,主体 A 确定原子数据 $t\in A$ 是 A 对于主体 S 的身份标志,如果 $t=A$ 或者 $t=K_A^{-1}\wedge t\notin P$ 或者 $t=K_{AS}\wedge t\notin P$ 或者 t 是 A 和 S 之间的共享秘密,记为 $id(A,A,S)$.

在协议 Strand 空间Σ中,主体 A 确定原子数据 $t\in A$ 是 B 对于主体 S 的身份标志,如果 $t=B$ 或者 t 是 A 能够确认被 S 所知的 A 与 B 之间的共享秘密,记为 $id(A,B,S)$.

表 2 给出了 A 发起与 S 的认证测试时,在不同加密方式下,各 Strand 的参数在某主体标志 X 上要达到一致所需满足的条件.需要指出的是:事实上,如果 A 与 S 的数据交互过程满足 AT 的要求,那么自然就满足在 $X=S$ 上的主体标志一致性.因此,后文的分析都针对 $X\neq S$ 的情况来进行.表 2 中第 1 列给出了 A 发出的 Challenge 的所有可能的加密情况(不加密、用 Kas 等密钥加密等等),行表示此时 AT 的类型以及 S 的 Response 在各种情况下所要满足的条件.

Table 2 Correspondence of Strand's principal parameters

表 **2** Strand 的主体参数一致性

A \ S		Type	1	2	3	4	5	6
			NULL	*Kas*	Ka^{-1}	*Ka*	Ks^{-1}	*Ks*
1	*NULL*	IT/UT	×	OK[R]	×	×	R	×
2	*Kas*	OT	OK[C]	OK[C/R]	×	OK[C/R]	OK[C/R]	×
3	Ka^{-1}	IT/UT	×	OK[C/R]	×	×	R	×
4	*Ka*	×	×	×	×	×	×	×
5	Ks^{-1}	×	×	×	×	×	×	×
6	*Ks*	OT	C	OK[C/R]	×	C	C/R	×

表中 C 表示 $id(A,X,S)$必须包含(这里的"包含"采用的是文献[2]中 $\subset'$ 的定义: $a\subset'\{g\}_k$, iff $a\subset' g\vee a=k\vee a=\{g\}_k$)在 Challenge 内;$R$ 表示 $id(A,X,S)$必须包含在 Response 内;OK 表示如果 $X=A$,那么直接符合要求,之所以会出现这种情况,是因为 Kas 或者 Ka^{-1} 本身就是 $id(A,A,S)$;方括号里的值表示,如果是其他主体标志($X\neq A$)需要满足的条件;× 表示不满足认证测试要求.下面,以第 1 行、第 2 列为例给出该表的简要说明.

S 返回用 A 与 S 之间的共享密钥 Kas 加密的响应数据,属于 IT 或者 UT(如在响应数据中包含时间戳).在此过程中,由于 Kas 的使用,A 能确定与 S 在主体参数 A 上的一致性,也就是说,A 能确定 S 是在和自己会话.如果 A 希望能确认在其他主体标志 B 上的一致性,那么 $id(A,B,S)$必须包含在 S 返回的数据中(一般是 B,或者是其他 A 确信 S 知道的 A 与 B 之间的共享秘密,例如此前 S 为 A 和 B 分配的会话密钥).

定理(认证测试的 **Strand** 参数一致性).

Strand 参数可以分为 Nonce、协商数据和主体标志 3 类,在主体 A 发起的与主体 S 之间的认证测试过程中,(i) A 能确定在 Nonce 类型参数上与 S 的一致性;(ii) 当且仅当认证测试的数据满足表 2 的要求,A 能确定与 S 在主体标志类型参数上的一致性;(iii) 当且仅当协商数据包含在测试组件中,A 能确定协商数据满足一致性.

利用认证逻辑和认证测试理论很容易证明 Strand 参数一致性定理的正确性.同时,由于在 3 类 Strand 参数的分析时已经给出相应的讨论,限于篇幅,这里就不再给出形式证明.

下面,我们应用该定理证明在 NS 协议中,主体 B 不能确定与 A 在主体参数 B 上的一致性.

$$\left(\left\langle +\{N_aA\}_{K_B},-\{N_aN_b\}_{K_A},+\{N_b\}_{K_B}\right\rangle,\left\langle -\{N_aA\}_{K_B},+\{N_aN_b\}_{K_A},-\{N_b\}_{K_B}\right\rangle\right).$$

证明:主体 B 向 A 发送 $\{N_aN_b\}_{K_A}$ 并接收 $\{N_b\}_{K_B}$ 的过程属于 OT,对应于表 2 的(6,4)项.由该项的值 C 我们知道,要满足在参数 B 上的一致性,必须有 $id(B,B,A)\subset'\{N_aN_b\}_{K_A}$.

显然, $id(B,B,A)\neq K_A\wedge id(B,B,A)\neq\{N_aN_b\}_{K_A}$,根据 $\subset'$ 的定义,必须有 $id(B,B,A)=N_a\vee id(B,B,A)=N_b$.

因为 N_b 是 B 新产生的随机数,要满足该条件,只有 $id(B,B,A)=N_a$;显然,$N_a\neq B, N_a\neq K_B^{-1}$ 并且 $N_a\neq K_{ab}$,由定义 7 可知,只有当 B 能确认 N_a 为 A 和 B 之间的共享秘密时才能满足条件.又因为 $N_a\subset'\{N_aA\}_{K_b}$ 并唯一产生于该节点,因此对于 B 来说,不能确定 N_a 是主体 A 发送给 B 的,因此也就不能确定 N_a 是 A,B 之间的共享秘密.

综上所述,NS 协议对于主体 B 不满足参数 B 的一致性. □

3.2 增强型认证测试(EAT)

我们在认证测试技术的基础上,结合上一节的 Strand 参数一致性定理,提出增强型认证测试 EAT 的概念.

EAT 1. 假设 C 是某协议 Strand 空间的线束,节点 $n' \in C$, 边 $n \Rightarrow^+ n'$ 是项 t 关于值 a 的 OT,那么必然存在节点 $m, m' \in C$ 满足 t 是 m 的消息组件,并且边 $m \Rightarrow^+ m'$ 是值 a 的 Transforming Edge;进一步地,如果能满足 Strand 参数的一致性,那么对于 AT 的发起方来说, $n \Rightarrow^+ n'$ 和 $m \Rightarrow^+ m'$ 满足关联性.

EAT 2. 假设线束 C 包含节点 n',边 $n \Rightarrow^+ n'$ 是项 t' 关于值 a 的 IT,那么必然存在节点 $m, m' \in C$ 满足 t' 是 m' 的消息组件,并且边 $m \Rightarrow^+ m'$ 是值 a 的 Transforming Edge;进一步地,如果能满足 Strand 参数的一致性,那么对于 AT 的发起方来说, $n \Rightarrow^+ n'$ 和 $m \Rightarrow^+ m'$ 满足关联性.

EAT 3. 假设线束 C 包含节点 n ,并且 n 是项 $t = \{h\}_K$ 的 UT,那么必然存在一个常规发送节点(正节点) $m \in C$ 满足 t 是 m 的消息组件;进一步地,如果能满足 Strand 参数的一致性,那么对于 AT 的发起方来说,n 和 m 满足关联性.

3.3 Correspondence函数

整个协议的关联性,建立在参与协议的不同主体之间的关联性基础上,本文在协议主体之间,以 EAT 为基本单位逐段分析其关联性.主体 B 对于 A 的关联度,记为 $Correspondence_{AB}$,其定义如下:

定义 **8(Correspondence** 函数).

在 Strand 空间中,$Correspondence_{AB}$ 表示为协议主体 B 相对于 A 的关联度,其值大于等于 0.如果 $Correspondence_{AB}$=0,那么我们可以认为协议主体 A 不能确认与主体 B 的关联性;否则 $Correspondence_{AB}$=n>0,表示协议主体 A 能确认与主体 B 的 Strand 边$\langle B,1\rangle \Rightarrow^+ \langle B,n\rangle$的关联性.

算法(关联度 $\boldsymbol{Correspondence_{AB}}$).

$Correspondence_{AB}$ 赋初值为 0,按 A 的 Strand 节点顺序依次考虑与 B 的认证测试(OT,IT 以及 B 向 A 提供的 UT).如果 B 的边$\langle B,i\rangle \Rightarrow +\langle B,j\rangle$满足 EAT_n(n=1,2 或 3),那么 $Correspondence_{AB}$=(j>$Correspondence_{AB}$)?j:$Correspondence_{AB}$,即如果 j>$Correspondence_{AB}$,那么 $Correspondence_{AB}$=j,否则 $Correspondence_{AB}$ 保持不变.

我们采用矩阵形式来描述整个协议的关联性,称为协议关联矩阵.若记矩阵为 M,有:

$$\begin{cases} M(A,B) = Correspondence_{AB}, & A \neq B \\ M(A,B) = height(Strand(A)), & A = B \end{cases}.$$

4 协议分析

安全协议的正确性主要体现为保密性和关联性,两者相比较,由于前者要简单得多,并且大部分协议都能确保其保密性,因此,本文不涉及到这方面的讨论.但是必须指出的是,安全协议的关联性必须建立在协议保密性的基础上,本文不作分析并非保密性不重要.下面我们给出协议关联性的分析过程:

1. 构造协议的 Strand 图(如图 1 所示),将协议的参与主体按任意顺序 $M1,M2,\ldots,Mn$ 排列.
2. 按第 1 步确定的顺序依次对主体 Mi 进行分析,过程如下:
   i. 考察 Mi 主动发起的认证测试(OT 或者 IT),以及其他协议主体向 Mi 提供的 UT,以此作为基本单位将 Mi 的 Strand 分段;
   ii. 按照 Mi 的 Strand 节点的顺序依次分析 Mi 的各个认证测试,根据 Strand 参数一致性定理以及 EAT 判断在认证测试中是否满足关联性,并将结果填充至关联矩阵.
3. 根据在第 2 步获得的关联矩阵,判断协议整体的关联性.

下面我们以 Kerberos 协议和 OR 协议为例,演示 EAT 概念、Correspondence 函数和 Strand 参数一致性定理在安全协议分析中的应用.

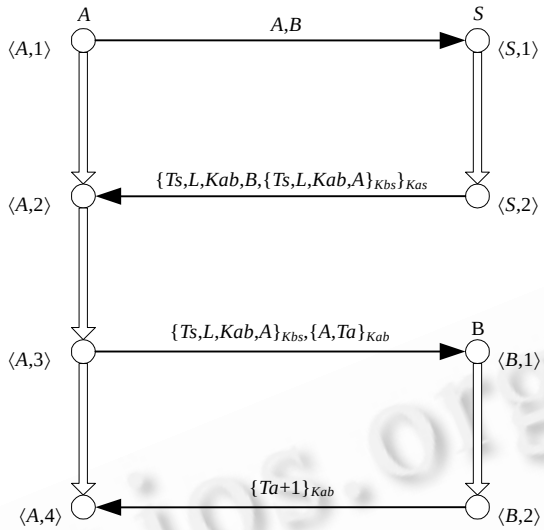

Fig.1 Strand graph of Kerberos protocol

图 1 Kerberos 协议 Strand 图

4.1 Kerberos协议分析

Kerberos 协议是一个三方的身份鉴别及密钥分配协议,其中 S 为密钥分配服务器,A,B 为另外两个协议主体.Kas,Kbs 分别是协议主体 A,B 与服务器 S 之间的共享密钥,Ts,Ta 是时间戳,L 是生命周期值.

协议的发起方 A 首先要求密钥分配服务器 S 为主体 A,B 分配会话密钥,在获取 S 的反馈信息后,A 解出所含的会话密钥,将转发信息与自己的测试信息一起发给 B,B 收到转发的会话密钥后,对 A 的测试信息作出反馈.

Kerberos 协议涉及的 Strand 参数可以进行如下分类:(i) Nonce:包含 Ts,Ta;(ii) 协商数据:包含 Kab,L,Kbs 以及 Kas;(iii) 主体标志:包含 A,B 和 S.

下面我们分别从主体 A,B,S 的角度分析该协议的关联性:

① 主体 A

主体 A 在节点$\langle A,1\rangle$发送明文数据 A,B 到服务器 S,仅仅起到告知服务器协议执行涉及主体的作用.

在负节点$\langle A,2\rangle$,主体 A 接收到数据 $term(\langle A,2\rangle)=\{Ts,L,Kab,B,\{Ts,L,Kab,A\}_{Kbs}\}_{Kas}$,该数据中的值 Ts(时间戳)是在协议 Strand 空间唯一生成的,因此,$term(\langle A,2\rangle)$是 Ts 的测试组件;又 $K_{as}\notin P$,所以节点$\langle A,2\rangle$是 Ts 的 UT. $B\subset' term(\langle A,2\rangle)$,根据表 2 中的(1,2)项,协议在主体参数 A,S,B 上达到一致;同时,因为协商数据 L,Kab 等包含于测试组件,因此在该类型参数上也能保证一致性.由 EAT3 我们得到结论:协议主体 A,S 在$\langle A,2\rangle$与$\langle S,2\rangle$处满足关联性,有 $Correspondence_{AS}=2$.

$\langle A,3\rangle\Rightarrow\langle A,4\rangle$ 是关于时间戳 Ta 的 OT,根据表 2 中的(2,2)项,协议在主体参数 A,B 上达到一致,并且在协商数据 Kab 上也能满足一致性.由 EAT1,我们有结论:协议主体 A,B 在 $\langle A,3\rangle\Rightarrow\langle A,4\rangle$ 与 $\langle B,1\rangle\Rightarrow\langle B,2\rangle$ 处满足关联性,且有 $Correspondence_{AB}=2$.

② 主体 B

主体 B 在负节点$\langle B,1\rangle$收到数据$\{Ts,L,Kab,A\}Kbs$ 和$\{A,Ta\}Kab$,因为 Ts,a 为协议 Strand 空间唯一生成的时间戳,并且 $K_{bs}\notin P, K_{ab}\notin P$,所以有节点$\langle B,1\rangle$是关于 Ts 和 Ta 的 UT,分别来自主体 S 和 A.

对于前者,根据表 2 中的(1,2)项,协议在主体参数 A,S,B 上达到一致;同时,因为协商数据 L 和 Kab 包含于测试组件,因此也能保持一致性.由 EAT3 与 Strand 空间的 $\leq$–$mimnimal$ 属性,我们有结论:协议主体 B,S 在$\langle B,1\rangle$与$\langle S,2\rangle$处满足关联性,且有 $Correspondence_{BS}=2$.

对于后者,在主体参数 A,B 以及协商数据 Kab 上达到一致,根据 EAT3 有:协议主体 B,A 在$\langle B,1\rangle$与$\langle A,3\rangle$处满足关联性,且有 $Correspondence_{BA}=3$.

③ 主体 *S*:主体 *S* 不能确定任意的关联性

综上所述,我们有结论:

$$Correspondence(Kerberos-A-B-S)=\begin{bmatrix}4 & 2 & 2\\ 3 & 2 & 2\\ 0 & 0 & 2\end{bmatrix}.$$

由 $Correspondence_{AS}$=2 与 $Correspondence_{AB}$=2,我们可以知道:主体 *A* 能确定 *Kab* 是服务器 *S* 为 *A*,*B* 分配的会话密钥,并且该密钥得到了主体 *B* 的确认;对于主体 *B*,结论完全相同.

4.2 OR协议分析

OR 协议的目标同样是密钥分配服务器为主体 *A*,*B* 分配会话密钥.这里,直接给出根据 EAT 与 *Correspondence* 函数的分析结果,然后,根据我们的结果分析其出错的原因.

$Correspondence(OR-A-B-S)=\begin{pmatrix}2 & 0 & 2\\ 0 & 4 & 2\\ 0 & 0 & 2\end{pmatrix}$,由此我们可以发现:对于主体 *A*,他能确认 *Kab* 是 *S* 为 *A*,*B* 分配的会话密钥;同样,对于 *B* 也能确认 *Kab* 是 *S* 为 *A*,*B* 分配的会话密钥;但是 *A*,*B* 两者之间没有任何关联性保障,也就是说,*A*,*B* 都不能确认 *S* 分配的 *Kab* 与 *Kab* 是同一个密钥.这也就是 OR 协议失败的原因.

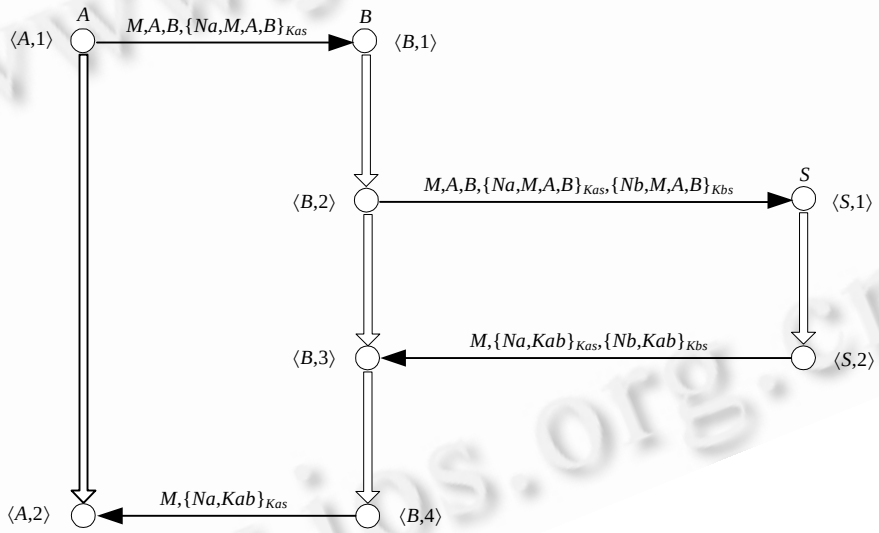

Fig.2 Strand graph of OR protocol

图 2 OR 协议 Strand 图

5 总 结

与认证逻辑等形式化方法相比,Strand 空间模型为协议分析人员提供了 Strand 这种更适合协议描述的工具,AT 则将不同协议主体 Strand 的若干有序状态集合逻辑地联系了起来.本文提出的 EAT 概念对 AT 进行了扩展,提出了 Strand 参数一致性定理,很好地解决了参数一致性的分析问题,从而更便于协议关联性分析自动化的实现,是协议分析合理、有效的基本单位,而 Correspondence 函数提供了从协议主体相互之间的关联性乃至协议整体的关联性的描述手段,将关联性这一逻辑概念进行了量化.

下一步的研究工作主要围绕两方面进行:首先是在本文提出的这些分析技术的基础上,进行安全协议分析自动化工具的实现;其次是进行将 EAT 应用于安全协议设计的研究.

References:

[1] Fàbrega FJT, Herzog JC, Guttman JD. Strand spaces: Why is a security protocol correct? In: Proc. of the 1998 IEEE Symp. on Security and Privacy. Los Alamitos: IEEE Computer Society Press, 1998. 160−171. http://ieeexplore.ieee.org/iel4/5528/14832/00674832.pdf?tp=&arnumber=674832&isnumber=14832

[2] Fàbrega FJT, Herzog JC, Guttman JD. Strand spaces: Proving security protocols correct. Journal of Computer Security, 1999, 7(2−3):191−230.

[3] Paulson LC. The inductive approach to verifying cryptographic protocols. Journal of Computer Security, 1998,6(1):85−128.

[4] Guttman JD, Fàbrega FJT. Authentication tests. In: Proc. of the 2000 IEEE Symp. on Security and Privacy. Los Alamitos: IEEE Computer Society Press, 2000. 96−109. http://ieeexplore.ieee.org/iel5/6864/18435/00848448.pdf?tp=&arnumber=848448&isnumber=18435

[5] Guttman JD, Fàbrega FJT. Authentication tests and the structure of bundles. Theoretical Computer Science, 2002,283(2):333−380.

[6] Guttman JD. Security protocol design via authentication tests. In: Proc. of the 2002 IEEE Computer Security Foundations Workshop. Los Alamitos: IEEE Computer Society Press, 2002. 92−103. http://ieeexplore.ieee.org/iel5/7957/21985/01021809.pdf?tp=&arnumber=1021809&isnumber=21985

[7] Woo TYC, Lam SS. A semantic model for authentication protocols. In: Proc. of the 1993 IEEE Computer Society Symp. on Research in Security and Privacy. Los Alamitos: IEEE Computer Society Press, 1993. 178−194. http://ieeexplore.ieee.org/iel2/902/7168/00287633.pdf?tp=&arnumber=287633&isnumber=7168

[8] Song DXD. Athena: A new efficient automatic checker for security protocol analysis. In: Proc. of the 12th IEEE Computer Security Foundations Workshop. Los Alamitos: IEEE Computer Society Press, 1999. 192−202. http://ieeexplore.ieee.org/iel5/6332/16921/00779773.pdf?tp=&arnumber=779773&isnumber=16921

[9] 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.

[10] Syverson P. Towards a strand semantics for authentication logic. Electronic Notes in Theoretical Computer Science, 1999,20: 143−157. http://citeseer.ist.psu.edu/syverson99towards.html

[11] Cervesato I, Durgin N, Kanovich M, Scedrov A. Interpreting strands in linear logic. In: Veith H, Heintze N, Clark E, eds. Proc. of the 2000 Workshop on Formal Methods and Computer Security. Chicago, 2000. http://theory.stanford.edu/~iliano/papers/fmcs00.ps.gz

[12] Ji QG, Qing SH, Zhou YB, Feng DG. Study on strand space model theory. Journal of Computer Science and Technology, 2003, 18(5):553−570.

[13] Fan H, Feng DG. Security Protocol Theory and Method. Beijing: Science Press, 2003. 41−42 (in Chinese).

[14] Burrows M, Abadi M. A logic of authentication. ACM Trans. on Computer Systems, 1990,8(1):18−36.

附中文参考文献:

[13] 范红,冯登国.安全协议理论与方法.北京:科学出版社,2003.41−42.

杨明(1979－),男,江苏常州人,博士生,主要研究领域为网络安全,协议分析.

罗军舟(1960－),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为协议工程,网络安全,网络管理,网格计算.