doi:10.3969/j.issn.1000-436x.2014.11.013

基于 SAT 的安全协议惰性形式化分析方法

顾纯祥[1,2]，王焕考[1]，郑永辉[1,2]，辛丹[1]，刘楠[1]

(1. 解放军信息工程大学 网络空间安全学院，河南 郑州 450001；2. 数学工程与先进计算国家重点实验室，江苏 无锡 214125)

摘 要：提出了一种基于布尔可满足性问题的安全协议形式化分析方法 SAT-LMC，通过引入惰性分析的思想优化初始状态与转换规则，提高了安全性的检测效率。另一方面，通过在消息类型上定义偏序关系，SAT-LMC 能够检测出更丰富的类型缺陷攻击。基于此方法实现了一个安全协议分析工具，针对 Otway-Rees 协议检测出了一种类型缺陷攻击；针对 OAuth2.0 协议，检测结果显示对现实中存在的一些应用场景，存在一种利用授权码截取的中间人攻击。

关键词：安全协议；形式化分析；布尔可满足性；惰性分析；类型缺陷攻击

中图分类号：TN915.0 **文献标识码**：A **文章编号**：1000-436X(2014)11-0117-09

SAT-based lazy formal analysis method for security protocols

GU Chun-xiang[1,2], WANG Huan-xiao[1], ZHENG Yong-hui[1,2], XIN Dan[1], LIU Nan[1]

(1. Institute for Network Security, PLA Information Engineering University, Zhengzhou 450001, China;
2. State Key Laboratory of Mathematical Engineering and Advanced Computing, Wuxi 214125, China)

Abstract: A SAT-based security protocol formalization analysis method named SAT-LMC is proposed. The method introduces optimized the initial state and transformational rules with “lazy” idea. The efficiency of detection is significantly improved. Moreover, by adding support for strong type flaw attack defect, the attack detection becomes more comprehensive. A security protocol analysis tool is implemented based on the method; a type flaw attack is detected for protocol Otway-Rees. For OAuth2.0 protocol, analysis shows that there is a kind of man-in-the-middle attack of the authorization code in some application scenarios.

Key words: security protocols; formalization analysis; Boolean satisfiability problem; lazy analyze; type flaw attack

1 引言

随着计算机网络技术的飞速发展，网络信息安全问题越来越受到人们的重视。网络安全协议采用密码技术实现通信和数据安全，在整个网络安全保障体系中具有举足轻重的作用。随着协议的流程步骤和网络环境的日益复杂，安全协议的安全性分析评估具有非常重要的意义和价值。

近年来，安全协议形式化分析研究受到国内外研究人员越来越多的关注，并取得了很大的研究进展，一些成熟的形式化分析方法也在实际网络环境中得到应用，例如，Sun 等[1]利用形式化分析的方法检测到了 OpenID 单点登录协议存在的安全缺陷。而对于 OAuth 开放授权协议，文献[2]发现了 OAuth1.0 协议的一个漏洞，并给出相应改进。2004 年，Armando A 和 Compagna L[3]提出一种基于布尔可满足性问题（SAT, Booolean satisfiability problem）的安全协议形式化分析方法。Armando A 等[4]进一步设计开发了基于 SAT 的安全协议模型检测工具 SATMC，该工具成为欧盟联合开展的 AVISPA

收稿日期：2014-06-30；**修回日期**：2014-10-30

基金项目：河南省科技创新杰出青年基金资助项目（134100510002）；河南省基础与前沿技术研究基金资助项目（142300410002）；数学工程与先进计算国家重点实验室开放基金资助项目

Foundation Items: Henan Provincial Science and Technology Innovation Fund for Outstanding Young (134100510002); Foundation and Research in Cutting-edge Technologies in the Project of Henan Province (142300410002); Mathematical Engineering and Advanced Calculation State Key Laboratory Open Foundation

（automated validation of inter security protocols and applications）工程的 4 个分析终端之一。2011 年，Bradley[5]将 SAT 模型检测工具改进为无需对转换关系进行展开，大大提高了检测效率。Vizel 等[6]和 Johnson 等[7]对 SATMC 的优化和应用做了进一步的工作。

本文在前人工作基础上，提出了一种基于 SAT 问题的安全协议分析方法 SAT-LMC（SAT-based lazy model checking），通过加入惰性的思想优化初始状态与转换规则，并根据协议的复杂程度动态调用命题公式的范式形式，提高了模型检测的效率；通过在消息类型上定义偏向关系，SAT-LMC 能够检测出一些经典协议[8~11]中存在的类型缺陷攻击[12]。应用 SAT-LMC，本文检测出当前网络上广泛应用的 OAuth2.0[11]协议的一个有效攻击路径，而 SATMC 并不能检测出上述攻击。

2 背景知识

2.1 攻击者模型

攻击者模型是对一个攻击者具备的攻击能力的描述，本文采用现实中最常用的 Dolev-Yao 模型[13]，简称 DY 模型。在 DY 攻击者模型中，攻击者可以控制整个通信网络，并假定攻击者除了可以窃听、阻止、截获所有经过网络的消息等之外，还具备以下知识和能力。

1) 熟知参与协议的主体标识符和公钥等一系类公开信息；

2) 具有专业的密码分析知识和能力；

3) 熟悉加解密、散列、签名验证等密码基础运算，并且拥有自己的加解密密钥；

4) 具有进行各种攻击（例如重放攻击、并行会话攻击）的知识和能力。

2.2 符号定义

本文的符号定义如表 1 所示。

2.3 SAT 问题

SAT 问题在硬件测试、人工智能、计算机视觉等很多领域都有广泛应用，作为经典的 NP 完全问题，很多问题都可以转换成 SAT 问题从而利用 SAT 问题求解器进行求解。基于 SAT 的安全协议形式化分析的基本思想是：首先建立形式化分析模型，给出协议的初始状态、转换规则以及需要满足的安全目标，然后根据转换规则对协议进行状态变换，最后将状态以命题公式的形式输入 SAT 求解器求解可满足性问题，如果命题公式可满足，就可以找出协议存在的攻击路径。

表 1 符号定义

符号	定义
$A \to B : Message$	实体 A 向实体 B 发送消息 $Message$
$\{Message\}_K$	使用密钥 K 加密消息 $Message$
$\{Message1.Message2\}$	$Message1$ 与 $Message2$ 连接
$new()$	生成一个具有随机性的变量值
$cons(k,S)$	将变量 k 放入集合 S 中
$in(k,S)$	从集合 S 中按堆栈的顺序取出一个元素并赋值给 k
$delete(k,S)$	将 k 从集合 S 中删除
$ik(m)$	攻击者知道消息 m
$msg(j,s,r,m)$	在协议的第 j 步，实体 s 向实体 r 发送了消息 m
$secret(m,[s_1,\cdots,s_i])$	消息 m 在实体 $[s_1,\cdots,s_i]$ 之间秘密保存
$witness(s,r,o,m)$	实体 s 将变量 o 赋值为 m，并请求与实体 r 认证
$request(r,s,o,m,c)$	在会话 c 中，实体 r 接收变量 o 的值为 m 并实现与 s 的认证

3 基于 SAT 的惰性模型检测方法 SAT-LMC

基于模型检测的安全协议形式化分析是当前协议分析的重要方法，本文根据有界模型检测的方法，提出了一种基于可满足性的安全协议模型检测方法 SAT-LMC。在有界模型检测系统中，定义一个转换系统 M 和一个界限 k，其中，$M=\{S, I, T, L\}$，S 是一个有限的状态集合，I 是初始状态的命题集合，$T\subseteq S\times S$ 表示状态间的转换关系，L：$S\to 2^{AP}$ 是标记函数，AP 是表示具有某些性质的原子命题的集合，L 标识每个状态上哪些原子命题成立；k 是一个界限，表示搜索的深度不超过时间系数 k。SAT-LMC 方法的思想就是高效地写出集合 I 和 T，给出协议需要满足的目标 g，然后从初始状态集合 I 出发，根据状态迁移关系 T 中的迁移路径将状态迁移 k 次，利用 SAT 求解器来判断安全协议在界限 k 内是否存在攻击路径。

惰性分析的思想主要体现在 2 个方面：一是在刻画状态集合 I 时，以变量代替协议主体，在状态转化过程中再逐步对变量进行赋值，以减少状态集合中状态数量；二是在状态转换时只考虑协议流程重写规则，只有在更新攻击者知识域是考虑攻击者能力规则，进一步缩减状态空间规模

的增长速度。

接下来以 Otway-Rees 协议为例，具体介绍该方法形式化分析安全协议的过程。Otway-Rees 协议流程如下。

1) $A \to B : M, A, B, \{N_A, M, A, B\}_{K_{AS}}$。

2) $B \to S : M, A, B, \{N_A, M, A, B\}_{K_{AS}}$, $\{N_B, M, A, B\}_{K_{BS}}$。

3) $S \to B : M, \{N_A, K_{AB}\}_{K_{AS}}, \{N_B, K_{AB}\}_{K_{BS}}$。

4) $B \to A : M, \{N_A, K_{AB}\}_{K_{AS}}$。

3.1 初始状态的形式化描述

3.1.1 基本初始状态

SAT-LMC 方法在输入具体的协议流程后,首先要做的就是将协议形式化，形式化的第一步就是写出协议主体的初始状态。主体的初始状态用 $state(j, X, Y, [M_1 \cdots], Se)$ 表示，j 为状态编号，在会话 Se 中，主体 Y 有 $[M_1 \cdots]$ 的知识域，并期待 X 发送的消息，如果 Y 为协议发起者，那么 X=Y。针对 Otway-Rees 协议在会话 Se 中参与协议的主体 A、B、S，其初始状态可以表示为

$$state(0, A, A, [A, B, S, K_{AS}], Se)$$

$$state(1, A, B, [A, B, S, K_{BS}], Se)$$

$$state(2, B, S, [A, B, S, K_{AS}, K_{BS}], Se)$$

3.1.2 惰性的初始状态

模型检测的效率与初始状态的个数是线性负相关的，对初始状态的优化可以很好地解决模型检测的效率问题，为了减少初始状态的数量，首先考虑采用保持变量的惰性方法，不对 A、B、S 赋值，仍使其保持变量的性质，这样初始状态就能保持如下

$$state(0, A, A, [A, B, S, K_{AS}], f_1(A, B, S))$$

$$state(1, A, B, [A, B, S, K_{BS}], f_1(A, B, S))$$

$$state(2, B, S, [A, B, S, K_{AS}, K_{BS}], f_1(A, B, S))$$

其中，$f_1(A, B, S)$ 是一个函数，返回与变量 A、B、S 的具体赋值相关的一个会话编号。这样优化之后，协议能暂时保持 3 个初始状态，在后续的步骤中利用转换规则进行状态转换，并仅当在需要给出某一个状态的每个主体状态时在分别将 A、B、S 对应的具体主体代入即可。

3.1.3 协议主体与攻击者能力

以上优化虽然减少了初始状态的个数，但是需要在每一个协议状态分别将实体代入，本质上复杂度和原来是一样的，于是需要进一步优化。对于协议中攻击者可以扮演的角色（例如 Otway-Rees 协议攻击者可以作为 A、B），SAT-LMC 赋予它们攻击者的能力，对于这些主体获得的知识 k，如果 k 没有随机性（随机数或与主体标识符相关）并且 k 不属于主体的初始知识域，那么攻击者也可以获知 k，对于具有随机性的知识 k，将 $\{k, f_1(A, B, S)\}$ 加入攻击者知识域，表示 k 是与会话相关的知识。这样 Otway-Rees 协议的初始状态就可以真正意义上地约化为 3 个。

3.2 协议转换规则的描述

3.2.1 基本的协议转换规则

给定了协议的初始状态，就需要有协议转换规则来对初始状态进行变换，从而得到在不同时间步骤协议的不同状态。协议的转换规则一般分为 2 类：攻击者能力规则和协议流程重写规则。一条转换规则可以表示为 $Left \xrightarrow{act} Right$，表示协议可以经过动作 act 将状态从 $Left$ 转换为 $Right$。

攻击者能力规则的作用是给出攻击者能力的形式化描述，SAT-LMC 方法采用现实中最常见的 Dolev-Yao 模型，DY 模型的攻击者主要有窃听、阻止、截获消息、加解密、消息的连接分割操作等能力。 基于 DY 模型的攻击者能力规则可以描述为

$$msg(J, A, B, M) \xrightarrow{divert(A,B,J,M)} ik(M)$$

$$ik(M).ik(A).ik(B) \xrightarrow{fake(A,B,M)} ik(M).ik(A).ik(B).msg(j, A, B, M)$$

$$ik(M).ik(K) \xrightarrow{encrypt(K,M)} ik(M).ik(K).ik(\{M\}_K)$$

$$ik(M).ik(K) \xrightarrow{sencrypt(K,M)} ik(M).ik(K).ik(\{M\}_K^S)$$

$$ik(M_1).ik(M_2) \xrightarrow{pairing(M_1,M_2)} ik(< M_1, M_2 >)$$

$$ik(\{M\}_K).ik(K^{-1}) \xrightarrow{decrypt(K,M)} ik(\{M\}_K).ik(K^{-1}).ik(M)$$

$$ik(\{M\}_K^S).ik(K) \xrightarrow{sdecrypt(K,M)} ik(\{M\}_K^S).ik(K).ik(M)$$

$$ik(< M_1, M_2 >) \xrightarrow{decompose(M_1,M_2)} ik(M_1).ik(M_2)$$

协议流程重写规则是根据协议的具体流程写出的，每一步流程对应一个转换规则，Otway-Rees 协议的协议流程重写规则可以表示为

$$state(0, A, A, [A, B, S, K_{AS}], f_1(A, B, S)) \xrightarrow{step_1(A,B,S,K_{AS},K_{BS},f_1(A,B,S))} \exists M. \exists N_A. state(3, B, A, [A, B, S, K_{AS}, M, N_A], f_1(A, B, S)).msg(1, A, B, < M, A, B, \{N_A, M, A, B\}_{K_{AS}} >)$$

$$msg(1,A,B,M.A.B.X_1)$$
$$.state(1,A,B,[A,B,S,K_{BS}],f_1(A,B,S))$$
$$\xrightarrow{step_2(A,B,S,K_{AS},K_{BS},M,N_A,f_1(A,B,S))}$$
$$\exists N_B.state(4,S,B,[A,B,S,K_{BS},M,N_B],$$
$$f_1(A,B,S)).msg(2,B,S,<M,A,B,X_1,$$
$$\{N_B,M,A,B\}_{K_{BS}}>)$$
$$msg(2,B,S,<M,A,B,\{N_A,M,A,B\}_{K_{AS}},$$
$$\{N_B,M,A,B\}_{K_{BS}}>)$$
$$.state(2,B,S,[A,B,S,K_{AS},K_{BS}],f_1(A,B,S))$$
$$\xrightarrow{step_3(A,B,S,K_{AS},K_{BS},M,N_A,N_B,f_1(A,B,S))}$$
$$\exists K_{AB}.state(5,S,S,[A,B,S,K_{AS},K_{BS},$$
$$K_{AB},M,N_A,N_B],f_1(A,B,S))$$
$$.msg(3,S,B,<M,\{N_A,K_{AB}\}_{K_{AS}},$$
$$\{N_B,K_{AB}\}_{K_{BS}}>)$$
$$msg(3,S,B,<M,X_2,\{N_B,K_{AB}\}_{K_{BS}}>)$$
$$.state(4,S,B,[A,B,S,K_{BS},M,N_B],f_1(A,B,S))$$
$$\xrightarrow{step_4(A,B,S,K_{AS},K_{BS},K_{AB},M,N_A,N_B,f_1(A,B,S))}$$
$$state(6,A,B,[A,B,S,K_{BS},K_{AB},M,N_B],f_1(A,B,S))$$
$$.msg(4,B,A,<M,X_2>)$$

其中，X_1、X_2 表示未知消息类型，由于 B 有 A 与 S 的密钥，无法精确匹配这 2 个实体间的加密消息，只能当作未知消息类型处理。这样做就使本文的模型检测方法能够检测针对以上 2 个密文的类型缺陷攻击，需要做进一步的修改来优化本文的检测工具并使系统支持完全类型缺陷攻击检测。

3.2.2 类型缺陷攻击支持

类型缺陷攻击 *A*：实体 *A* 发送消息 *msg* 给实体 *B*，如果 *B* 没有事先与 *A* 商定好消息的类型，那么 *B* 可以接收任意消息类型，攻击者针对这种现象的攻击叫做类型缺陷攻击 *A*。

类型缺陷攻击 *B*：实体 *A* 发送消息 *msg* 给实体 *B*，如果 *B* 与 *A* 事先约定好消息的类型为 *T* 但是没有约定消息的具体内容，并且 *B* 无法判断消息 *msg* 是否是合法的 *T* 类型消息，攻击者针对这种现象的攻击叫做类型缺陷攻击 *B*。

为了使系统能够支持 2 个类型缺陷攻击检测，需要借助一个偏序集合。设 *R* 为非空集合 *A* 上的关系，如果 *R* 是自反的、反对称的和传递的，则称 *R* 为 *A* 上的偏序关系，简称偏序。定义一组偏序集 <*A*,*R*>，*A* 表示消息类型集合，其中的元素有：MSG（消息）、AGENT（实体）、NUM（自然数）、KEY（密钥）、STEPID（协议步骤）、SESSID（会话 ID）、CPT(密文消息)、RAN（随机值）；偏序关系 *R* 定义为：AGENT≤MSG、NUM≤MSG、KEY≤MSG、STEPID≤NUM、SESSID≤NUM、CPT≤MSG、RAN≤MSG。

对于实体不能精确匹配消息 *msg* 的内容而只能匹配消息类型的情况，SAT-LMC 使用 *msg'* 表示类型确定但内容不确定的变量消息。例如，*B* 接收 *msg'* 表示 *B* 可以接收类型为 TYPE(*msg*)的所有消息。

send(A,B) 表示 *A* 向 *B* 发送消息，*receive*(B,A) 表示 *B* 接收 *A* 发送的消息，修改 Otway-Rees 协议流程如下

$$send(A,B):M,A,B,\{N_A,M,A,B\}_{K_{AS}}$$
$$receive(B,A):M',A',B,X_1'$$
$$send(B,S):M,A,B,X_1,\{N_B,M,A,B\}_{K_{BS}}$$
$$receive(S,B):M',A',B',\{N_A',M',A',B'\}_{K_{AS}},$$
$$\{N_B',M',A',B'\}_{K_{BS}}$$
$$send(S,B):M,\{N_A,K_{AB}\}_{K_{AS}},\{N_B,K_{AB}\}_{K_{BS}}$$
$$receive(B,S):M,X_2',\{N_B,K_{AB}'\}_{K_{BS}}$$
$$send(B,A):M,X_2$$
$$receive(A,B):M,\{N_A,K_{AB}'\}_{K_{AS}}$$

由此得到的协议流程重写规则可以表示为

$$state(0,A,A,[A,B,S,K_{AS}],$$
$$f_1(A,B,S))\xrightarrow{step_1(A,B,S,K_{AS},K_{BS},f_1(A,B,S))}$$
$$\exists M.\exists N_A.state(3,B,A,[A,B,S,K_{AS},M,N_A],$$
$$f_1(A,B,S)).msg(1,A,B,<M,A,B,$$
$$\{N_A,M,A,B\}_{K_{AS}}>)$$
$$msg(1,A,B,M'.A'.B.X_1')$$
$$.state(1,A,B,[A,B,S,K_{BS}],f_1(A,B,S))$$
$$\xrightarrow{step_2(A,B,S,K_{AS},K_{BS},M,N_A,f_1(A,B,S))}$$
$$\exists N_B.state(4,S,B,[A,B,S,K_{BS},M,N_B],f_1(A,B,S))$$
$$.msg(2,B,S,<M,A,B,X_1,\{N_B,M,A,B\}_{K_{BS}}>)$$
$$msg(2,B,S,<M',A',B',\{N_A',M',A',B'\}_{K_{AS}},$$
$$\{N_B',M',A',B'\}_{K_{BS}}>)$$
$$.state(2,B,S,[A,B,S,K_{AS},K_{BS}],f_1(A,B,S))$$
$$\xrightarrow{step_3(A,B,S,K_{AS},K_{BS},M,N_A,N_B,f_1(A,B,S))}$$
$$\exists K_{AB}.state(5,S,S,[A,B,S,K_{AS},K_{BS},K_{AB},$$
$$M,N_A,N_B],f_1(A,B,S))$$
$$.msg(3,S,B,<M,\{N_A,K_{AB}\}_{K_{AS}},\{N_B,K_{AB}\}_{K_{BS}}>)$$

$$msg(3,S,B,<M,X_2',\{N_B,K_{AB}'\}_{K_{BS}}>)$$
$$.state(4,S,B,[A,B,S,K_{BS},M,N_B],f_1(A,B,S))$$
$$\xrightarrow{step_4(A,B,S,K_{AS},K_{BS},K_{AB},M,N_A,N_B,f_1(A,B,S))}$$
$$state(7,A,B,[A,B,S,K_{BS},K_{AB},M,N_B],$$
$$f_1(A,B,S)).msg(4,B,A,<M,X_2>)$$
$$msg(4,B,A,<M,\{N_A,K_{AB}'\}_{K_{AS}}>)$$
$$.state(3,B,A,[A,B,S,K_{AS},M,N_A],f_1(A,B,S))$$
$$\xrightarrow{step_5(A,B,S,K_{AS},K_{BS},K_{AB},M,N_A,N_B,f_1(A,B,S))}$$
$$state(6,B,A,[A,B,S,K_{AS},K_{AB},M,N_A],f_1(A,B,S))$$

偏序关系的解释如下：如果 2 个消息有关系 $msg1 \leqslant msg2$，那么如果实体 B 期望接收 TYPE($msg2$)类型的消息，那么 B 也会接收 $msg1$，如果实体 B 期望接收 TYPE($msg1$)类型的消息，那么 B 也会接收某些 $msg1$（即类型为 TYPE($msg1$)的 $msg2$），$msg1$ 可以看作 $msg2$ 的子集；相反，如果 $msg1$、$msg2$ 没有偏序关系，那么 B 在以上 2 种情况都不会接收；而对于 3 个消息，例如 $msg1 \leqslant msg3$、$msg2 \leqslant msg3$，如果实体 B 期望接收 TYPE($msg3$)类型的消息，那么 B 也会接收某些 $msg1.msg2$ ($msg1$ 与 $msg2$ 的结合)。

3.2.3　惰性的协议转换规则

在协议的安全检测中，攻击者能力规则并不是时刻都需要运用到状态上的，比如说，攻击者在某个状态获得了一个加密密钥 K，根据攻击者能力的加密规则，攻击者需要对其知道的所有消息用密钥 K 加密求得密文，并加入到自己的知识域，这显然是没有必要的。于是本文加入如下假设。

1) 所有主体发送的消息都被攻击者 i 接收。

2) 所有主体接收的消息都是由攻击者 i 生成发送的。

3) 攻击者 i 维护一个知识域的最小集合，相当于攻击者知识域的一组基，每当攻击者或者新的知识，只有在不能用集合中元素表示时才加入集合；每当判定攻击者是否具有某个知识，如果这个知识能用集合表示，那么攻击者具有这个知识。

通过以上假设，本系统可以在状态转换时只考虑协议流程重写规则，只有在更新攻击者知识域是考虑攻击者能力规则，这是一种对攻击者能力动态调用的惰性思想。而对于转换规则中的存在量词结构 $\exists m$，定义如下结构代替：$Agent(m)$，表示主体 $Agent$ 生成了 m。此外，由于初始状态简化为了 3 个，如果按照上述正常的协议转换流程，那么只能检测单会话的攻击，因为某个状态经过转换后得到新状态，原状态就不存在了，这显然是不符合系统预期的，在 SAT-LMC 工具中，通过在转换规则右边加入左边的状态来解决这个问题，表示状态转换后原状态仍然保留，可以由其他会话继续转换。最终的转换规则如下

$$state(0,A,A,[A,B,S,K_{AS}],$$
$$f_1(A,B,S))\xrightarrow{step_1(A,B,S,K_{AS},K_{BS},f_1(A,B,S))}$$
$$A(M,N_A).state(3,B,A,[A,B,S,K_{AS},M,N_A],$$
$$f_1(A,B,S)).state(0,A,A,[A,B,S,K_{AS}],f_1(A,B,S))$$
$$.ik(M).ik(A).ik(B).ik(\{N_A,M,A,B\}_{K_{AS}})$$
$$i(M',A',X_1').ik(B).state(1,A,B,[A,B,S,K_{BS}],$$
$$f_1(A,B,S))\xrightarrow{step_2(A,B,S,K_{AS},K_{BS},M,N_A,f_1(A,B,S))}$$
$$B(N_B).state(4,S,B,[A,B,S,K_{BS},M,N_B],f_1(A,B,S))$$
$$.state(1,A,B,[A,B,S,K_{BS}],f_1(A,B,S))$$
$$.ik(M).ik(A).ik(B).ik(\{N_B,M,A,B\}_{K_{BS}}).ik(X_1)$$
$$i(M',A',B').ik(\{N_A',M',A',B'\}_{K_{AS}}$$
$$.ik(\{N_B',M',A',B'\}_{K_{BS}}$$
$$.state(2,B,S,[A,B,S,K_{AS},K_{BS}],f_1(A,B,S))$$
$$\xrightarrow{step_3(A,B,S,K_{AS},K_{BS},M,N_A,N_B,f_1(A,B,S))}$$
$$S(K_{AB}).state(5,S,S,[A,B,S,K_{AS},K_{BS},$$
$$K_{AB},M,N_A,N_B],f_1(A,B,S))$$
$$.state(2,B,S,[A,B,S,K_{AS},K_{BS}],f_1(A,B,S))$$
$$.ik(M).ik(\{N_A,K_{AB}\}_{K_{AS}}).ik(\{N_B,K_{AB}\}_{K_{BS}})$$
$$ik(M).i(X_2').ik(\{N_B,K_{AB}'\}_{K_{BS}})$$
$$.state(4,S,B,[A,B,S,K_{BS},M,N_B],f_1(A,B,S))$$
$$\xrightarrow{step_4(A,B,S,K_{AS},K_{BS},K_{AB},M,N_A,N_B,f_1(A,B,S))}$$
$$state(7,A,B,[A,B,S,K_{BS},K_{AB},M,N_B],f_1(A,B,S))$$
$$state(4,S,B,[A,B,S,K_{BS},M,N_B],$$
$$f_1(A,B,S)).ik(M).ik(X_2)$$
$$ik(M).ik(\{N_A,K_{AB}'\}_{K_{AS}})$$
$$.state(3,B,A,[A,B,S,K_{AS},M,N_A],f_1(A,B,S))$$
$$\xrightarrow{step_5(A,B,S,K_{AS},K_{BS},K_{AB},M,N_A,N_B,f_1(A,B,S))}$$
$$state(6,B,A,[A,B,S,K_{AS},K_{AB},M,N_A],f_1(A,B,S))$$
$$.state(3,B,A,[A,B,S,K_{AS},M,N_A],f_1(A,B,S))$$

3.3　给定协议安全目标

协议的安全目标规定了协议在每个状态必须满足的属性，安全目标可以分为秘密性和认证性 2 类。秘密性是指某些消息只能由指定集合内的实体获知，写作 $secret(m,[s_1,\cdots,s_i])$，表示为消息 m 只能由实体 $s_1,\cdots,s_i$ 获知；认证性是指 $witness(s,r,o,m)$ 和

$request(r,s,o,m,c)$ 构成的一对实体 r 对 s 的认证。

协议的安全目标保证了协议在实际网络环境中的安全性，一旦安全目标不满足，那么就有相应的攻击路径使协议不安全。对于 Otway-Rees 协议的安全目标可以归结为

$$g_1 : secret(K_{AB},[A,B,S])$$

$$g_2 : secret(K_{AS},[A,S])$$

$$g_3 : secret(K_{BS},[B,S])$$

$$g_4 : witness(A,S,N_A,n_a) \wedge request(S,A,N_A,n_a,Se)$$

$$g_5 : witness(B,S,N_B,n_b) \wedge request(S,B,N_B,n_b,Se)$$

$$g_6 : witness(S,A,K_{AB},k_{ab}) \wedge request(A,S,K_{AB},k_{ab},Se)$$

$$g_7 : witness(S,B,K_{AB},k_{ab}) \wedge request(B,S,K_{AB},k_{ab},Se)$$

3.4 有界模型检测命题公式

本文介绍的有界模型检测系统 SAT-LMC 是基于计算树逻辑（CTL, computational tree logic）的，CTL 公式由命题逻辑公式和时序模态词两部分构成。命题公式用来表示系统的状态，时序模态词表示系统运行将来能够经过的路径。

CTL 的时序模态词由一对符号构成。这对符号的第一个符号∈{A，E}，其中，A 表示必然的，E 表示可能的；这对符号的第二个符号∈{X,G,F,U}，其中，X 表示下一个，G 表示所有的将来，F 表示一些将来，U 表示“直到…才…”。例如，EF(p)为真当且仅当从状态 S 开始的所有路径中，至少存在一条路径，该路径至少存在一个状态，命题 p 在该状态中为真。

在 SAT-LMC 方法中，对于转换系统 M={S，I，T}和界限 k，可以构造如下的可满足性命题公式。

$$[\![\Xi]\!]_k = I(s_0) \wedge \bigwedge_{i=0}^{k-1} T(s_i,s_{i+1}) \wedge [\![\mathrm{EF}(\neg g)]\!]_k$$

其中，$I(s_0)$ 表示初始状态的命题集合，Otway-Rees 协议中即为 3 个初始状态的合取形式；$T(s_i,s_{i+1})$ 表示状态 s_i 通过状态转化关系转换为 s_{i+1} 后 s_{i+1} 中的所有命题集合，Otway-Rees 协议中的状态转化关系即为协议转换规则；$[\![\mathrm{EF}(\neg g)]\!]_k$ 表示从初始状态 s_0 到状态 s_k 的所有路径中，至少存在一条路径，该路径中存在一个状态，命题 g 不满足，而命题 g 表示所有安全目标的集合，$[\![\mathrm{EF}(\neg g)]\!]_k$ 可以表示为 $\bigvee_{i=0}^{k} \neg g_i$。

如果命题公式 $[\![\Xi]\!]_k$ 是可满足的，则说明在时间界限 k 内，至少存在一条攻击路径，使安全目标不能满足，即实现了协议的攻击。

3.5 规约为 SAT 问题求解

通过以上流程，SAT-LMC 方法将协议的安全问题规约为求解命题公式 $[\![\Xi]\!]_k$ 的可满足性问题。本节将通过一个简单的“2 bit 计数器”的例子简要说明如何调用 SAT 求解器求解协议安全问题。

“2 bit 计数器”初始状态以及状态转化关系如图 1 所示。

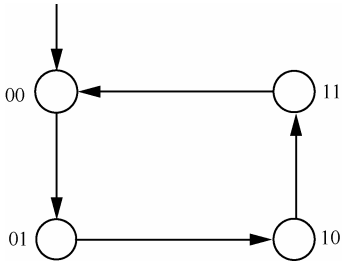


图 1 2 bit 计数器

如果 p 代表高位，q 代表低位，那么状态转化关系可以写作：$p_{i+1} \leftarrow p_i \oplus q_i$ 和 $q_{i+1} \leftarrow \neg q_i$。取 k=2，安全目标为 p*q=false，本文设计了 2 种不同的 SAT 求解规约方法。

方法 1 合取范式求解

SAT 求解器的输入是一个标准合取范式，因此 SAT-LMC 方法只需构造 $[\![\Xi]\!]_k$，并将其作为 SAT 求解器的输入，如果 $[\![\Xi]\!]_k$ 可满足，则说明协议不安全，存在攻击路径；如果 $[\![\Xi]\!]_k$ 不可满足，只能说明协议在时间界限 k 内不存在攻击路径，不能说明协议是安全的。对于以上例子，就是求解 $[\![\Xi]\!]_2 = I(s_0) \wedge T(s_0,s_1) \wedge T(s_1,s_2) \wedge (\neg g_0 \vee \neg g_1 \vee \neg g_2)$ 的可满足性问题。其中

$$I(s_0) = (\neg p_0 \wedge \neg q_0)$$

$$T(s_0,s_1) = \neg p_1 (p_0 \oplus q_0) \wedge q_1 (\neg q_0)$$

$$T(s_1,s_2) = p_2 (p_1 \oplus q_1) \wedge \neg q_2 (\neg q_1)$$

$$g_0 = p_0 \wedge q_0, g_1 = p_1 \wedge q_1, g_2 = p_2 \wedge q_2$$

方法 2 析取范式求解

通过观察方法 1 的命题公式，发现其中有太多的重复命题以不同的时间系数出现，这样使合取范式长度大大增加，有可能使 SAT 求解器无法求解。基于此 SAT-LMC 采用了动态调节的方法，对于 k 比较大的情况，采用构造析取范式的方法求解，流程如下。

i 从 0 到 k 循环：

如果 i=0，$[\![\Xi]\!]_i = I(s_0) \wedge \neg g$；

否则，计算 $[\![\Xi]\!]_i = T(s_i, s_{i+1}) \wedge \neg g$；

将 $[\![\Xi]\!]_i$ 输入 SAT 求解器，如果 $[\![\Xi]\!]_i$ 可满足，则输出协议不安全并退出，否则继续循环；

本系统需要将空转化规则加入到 $T(s_i, s_{i+1})$ 中，即如果一个命题属于状态 s_i，那么如果状态 s_i 到 s_{i+1} 没有删除该命题的行为，那么该命题仍属于 s_{i+1}，以便保证 $T(s_i, s_{i+1})$ 的命题完整性。对于 k=2 的“2 bit 计数器”例子，总结来说就是求 $[\![\Xi]\!]_2 = (I(s_0) \wedge \neg g) \vee (T(s_0, s_1) \wedge \neg g) \vee (T(s_1, s_2) \wedge \neg g)$ 的可满足性。其中

$I(s_0) = (\neg p \wedge \neg q)$，

$T(s_0, s_1) = \neg p(p \oplus q) \wedge q(\neg q)$

$T(s_1, s_2) = p(p \oplus q) \wedge \neg q(\neg q)$，

$g = p \wedge q$

方法 1 与方法 2 均可求解有界模型检测的可满足性问题，方法 1 更适用于简单的协议实例，在构建的命题公式不会太长的情况下只需调用一次 SAT 求解器；方法 2 适用于复杂的协议实例，对于相对复杂的协议，如果使用方法 1，有可能引起命题公式长度过长，并使 SAT 求解器效率低下，而通过反复构造命题公式，并大大减少每一个命题公式的长度，有利于提高 SAT 求解器的效率。

4　实验结果分析与比较

4.1　实验结果分析

按照本文提出的 SAT-LMC 方法实现了安全协议模型检测工具，针对 Otway-Rees 协议运用本文设计的模型检测系统，SAT-LMC 能够检测到以下攻击路径，这个攻击是一种类型缺陷攻击，攻击完成后协议主体 A 和 B 会使用共同的密钥($M.A.B$)，这些信息都是在攻击者知识域内的，所以协议安全目标 g_3 被攻破，协议是不安全的。

1)　$A \to i(B) : M, A, B, \{N_A, M, A, B\}_{K_{AS}}$

2)　$i(B) \to B : M, A, B, \{N_A, M, A, B\}_{K_{AS}}$

3)　$B \to i(S) : M, A, B, \{N_A, M, A, B\}_{K_{AS}}, \{N_B, M, A, B\}_{K_{BS}}$

4)　$i(S) \to B : M, \{N_A, M, A, B\}_{K_{AS}}, \{N_B, M, A, B\}_{K_{BS}}$

5) $B \to i(A) : M, \{N_A, M, A, B\}_{K_{AS}}$

6)　$i(A) \to A : M, \{N_A, M, A, B\}_{K_{AS}}$

近年来，OAuth2.0 协议成为互联网上账号通用和资源共享解决方案中重要的认证授权协议之一。OAuth2.0 协议有 5 个主体：Resource Owner（资源拥有者）；User Agent（用户代理）；Resource Server（资源服务器）；Client（客户端）；Authorization Server（授权服务器），协议详细规范见 RFC6749。将协议简化为 3 个实体，User、Client 和 Server，资源拥有者和用户浏览器归结为 User 一个实体，认证服务器和资源服务器归结为 Server 一个实体。利用 SAT-LMC 方法分析 OAuth2.0 协议，发现如果不按照协议规范推荐的协议三方均使用安全信道通信，则会存在一种利用授权码截取的中间人攻击，具体攻击路径如下

1.1. $User \to i$: U

2.1. $i \to Client : i$

2.2. $Client \to i$: Response_type.Client_id.Redirect_uri.Scope

1.2. $i \to User$: Response_type.Client_id.Redirect_uri.Scope

1.3. $User \to i$: {Response_type.Client_id.Redirect_uri.Scope.Credentials.Authorization}k_{us}

1.4. $i \to Server$: {Response_type.Client_id.Redirect_uri.Scope.Credentials.Authorization}k_{us}

1.5. $Server \to i$: {Code}k_{us}

1.6. $i \to User$: {Code}k_{us}

1.7. $User \to i$: Code

2.3. $i \to Client$: Code

2.4. $Client \to i$: {Grant_type.Code.Redirect_uri.Client_id.Client_key}k_{cs}

1.8. $i \to Server$: {Grant_type.Code.Redirect_uri.Client_id.Client_key}k_{cs}

1.9. $Server \to i$: {Token.Token_type.Expires}k_{cs}

2.5. $i \to Client$: {Token.Token_type.Expires}k_{cs}

2.6. $Client \to i$: {Token.Token_type.Scope}k_{cs}

1.10. $i \to Server$: {Token.Token_type.Scope}k_{cs}

1.11. $Server \to i$: {Resource}k_{cs}

2.7. $i \to Client$: {Resource}k_{cs}

这是针对 OAuth2.0 的授权码授权模式的一种攻击，步骤 1.x 与 2.x 分属于 2 个不同的会话，攻击者通过窃取通信中的授权码（Code），最终达到从服务器非法获取到受保护资源的目的。这是一种利

用授权码截取的中间人攻击，通过对国内访问量前100 的 OAuth2.0 协议服务端网站做统计，结果显示有 64%的网站存在该安全隐患，协议实现仍不规范不安全。

4.2 与其他工具比较

目前，利用可满足性来判定协议安全性的最为经典的工具是 SATMC，下面从检测效率和能力等方面将 SAT-LMC 和 SATMC[4~6]进行比较。

4.2.1 检测效率更高

通过对初始状态的优化，大大减少了初始状态的个数，对于 Otway-Rees 协议初始状态从 18 个减少为 3 个;通过对转换规则的优化，SAT-LMC 方法只在必要时调用攻击者能力规则；通过提出命题公式的 2 种形式的范式，SAT-LMC 可以根据协议复杂程度动态调用不同类型的范式。通过以上优化，增强了 SAT-LMC 方法模型检测的效率，相较于 SATMC 方法在检测速度上得到了明显的提升。通过设置不同的深度 k，分别 SATMC 与 SAT-LMC 分析 Fischer 协议[14]，时间比较如图 2 所示。

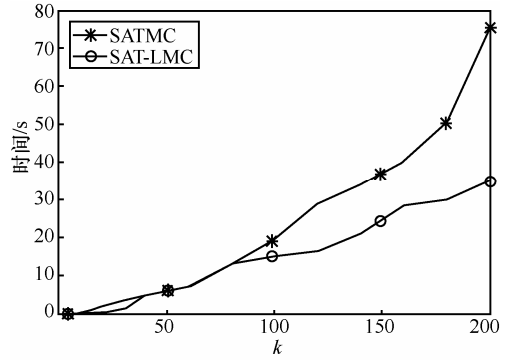


图 2 不同 k 下时间比较

4.2.2 攻击检测更全面

在 SAT-LMC 中，加入了强类型缺陷攻击检测，这正是 SATMC 缺少的，这直接导致本文的工具检测到的攻击路径在 SATMC 中是无法检测到的，SAT-LMC 方法更加准确，能够检测到更多的攻击。通过对几个协议的分析，SAT-LMC 检测到了类型缺陷攻击，如表 2 所示。其中 P/Q 表示 SATMC 的参数是 P，SAT-LMC 的参数改善到了 Q，TFA 表示类型缺陷攻击（type flaw attack），MITM 表示中间人攻击（man-in-the-middle attack），N 表示未发现攻击路径，M 表示在某些特定情况下有攻击路径。

表 2 协议攻击检测

协议	检测状态/个	发现攻击路径	时间消耗/s
Otway-Rees	71/11	N/TFA	0.56/0.22
NSPK[8]	21/8	MITM/MITM	0.31/0.14
CHAPv2[9]	547/125	N/N	1.78/0.92
RPC[10]	13/11	N/TFA	0.29/0.25
OAuth2.0[11]	12214/8562	N/M	4.53/3.13

5 结束语

本文提出了一种基于 SAT 问题的协议分析方法 SAT-LMC，通过加入惰性的思想优化初始状态与转换规则，并根据协议的复杂程度动态调用命题公式的范式形式，提高了模型检测的效率；另一方面，通过在消息类型上定义偏序关系，SAT-LMC 能够检测出更丰富的类型缺陷攻击。SAT-LMC 与经典的协议分析工具 SATMC 相比，具有速度更快、检测更加全面的优点。

参考文献：

[1] SUN S T, HAWKEY K, BEZNOSOV K. Systematically breaking and fixing OpenID security: formal analysis, semi-automated empirical evaluation, and practical countermeasures[J]. Computers & Security, 2012, 31(4):465-483.

[2] TASSANAVIBOON A, GONG G. OAuth and ABE based authorization in semi-trusted cloud computing: aauth[A]. Proceedings of the second international workshop on Data intensive computing in the clouds[C]. ACM, 2011.41-50.

[3] ARMANDO A, COMPAGNA L. SATMC: A SAT-based model checker for security protocols[M]. Logics in Artificial Intelligence. Springer Berlin Heidelberg, 2004.730-733.

[4] ARMANDO A, BASIN D, BOICHUT Y, *et al*. The AVISPA tool for the automated validation of internet security protocols and applications[A]. Computer Aided Verification[C]. Springer Berlin Heidelberg, 2005.281-285.

[5] BRADLEY A R. SAT-based model checking without unrolling[A]. Verification, Model Checking, and Abstract Interpretation[C]. Springer Berlin Heidelberg, 2011.70-87.

[6] VIZEL Y, GRUMBERG O, SHOHAM S. Lazy abstraction and SAT-based reachability in hardware model checking[A]. Proceedings of the 12th Conference on Formal Methods in Computer-Aided Design[C]. 2012.173-181.

[7] JOHNSON S, WILSON G, TANG Y, *et al*. SATMC: A Monte Carlo

Markov chain approach to SED fitting[A]. American Astronomical Society Meeting Abstracts[C]. 2013.221.

[8] LOWE G. An attack on the Needham-Schroeder public-key authentication protocol[J]. Information Processing Letters, 1995, 56(3): 131-133.

[9] ZORN G. Microsoft PPP CHAP extensions, version 2[EB/OL]. http://www.ietf.org/rfc/rfc2759.txt, 2000.

[10] BIRRELL A D, NELSON B J. Implementing remote procedure calls[J]. ACM Transactions on Computer Systems-TOCS, 1984, 2(1): 39-59.

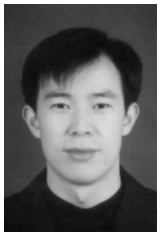
[11] HAMMER-LAHAV D E, HARDT D. The OAuth2. 0 Authorization Protocol[R]. IETF Internet Draft, 2011.

[12] HEATHER J, LOWE G, SCHNEIDER S. How to prevent type flaw attacks on security protocols[J]. Journal of Computer Security, 2003, 11(2): 217-244.

[13] DOLEV D, YAO A C. On the security of public key protocols[A]. Proceedings of the IEEE 22nd Annual Symposium on Foundations of Computer Science[C]. Nashville, TN, 1981, 350-357.

[14] DAWS C, OLIVERO A, TRIPAKIS S, *et al*. The Tool KRONOS[M]. Springer Berlin Heidelberg, 1996.208-219.

作者简介：



顾纯祥（1976-），男，安徽霍山人，博士，解放军信息工程大学副教授，主要研究方向为网络安全与密码学。

王焕孝（1988-），男，山东寿光人，解放军信息工程大学硕士生，主要研究方向为网络安全与密码学。

郑永辉（1976-），男，江西高安人，博士，解放军信息工程大学讲师，主要研究方向为网络安全与密码学。

辛丹（1990-），女，陕西西安人，解放军信息工程大学硕士生，主要研究方向为网络信息安全。

刘楠（1981-），女，湖北襄阳人，博士，解放军信息工程大学讲师，主要研究方向为网络信息安全。