OLSR 协议的 AVISPA 分析研究

林春平
（福建工程学院，福建 福州 350108）

摘 要：协议的安全目标分为认证性、非否认性、可追究性、公平性四种，其中，认证性应用最为广泛和重要，是网络安全性的基础。分析了先应式链路状态路由协议 OLSR 及其安性，并采用 AVISPA 工具对 OLSR 协议的既定目标进行分析验证，验证结果表明，该设计是安全的。

关键词：OLSR；安全；AVISPA；认证

Research and Analysis on OLSR of AVISPA

LIN Chun-ping
(Fujian University of Technology, Fuzhou,Fujian 350108, China)

Abstract:The security goals of Protocol are divided into four kinds, authentication, non-repudiation,non-repudiation,accountability,fairness.In which the application of authentication is widely used and most inportant.It is the basis of network security.In this paper we analyse security problems of the Proactive Routing Protocol of link-state .Then the tool of AVISPA to is introduced to analyse and verify the security of established goals. The results show that the design is safe.

Key words:OLSR；Security；AVISPA；Authentication

1 引言

无线自组网（Ad Hoc）是一种新兴的网络技术，具有单独组网能力和自组织的特点，OLSR[1]（Optimized Link State Routing）协议是 Ad Hoc 网络中的一种先应式路由协议，文中阐述了 OLSR 协议的基本机制及分析了该协议存在的安全漏洞，并采用 AVISPA[2] 工具对 OLSR 协议设计的安全策略进行分析验证。

2 OLSR 协议安全分析

2.1 OLSR 协议运行机制

OLSR 协议是一种先应式的链路状态路由协议，对纯链路状态算法进行优化，在这种路由协议机制中，无论是否有通信需求，每个节点周期性的广播交换路由分组信息，维护一张包含到达其他节点的路由信息的路由表。在 OLSR 协议中，通信节点间交互的消息有两种类型：HELLO 消息包和 TC（Topology control）消息包。HELLO 消息执行链路检测、邻居发现功能；TC 消息执行多点中继（Multipoint Relay，MPR）声明功能。

2.2 OLSR 协议安全漏洞

在 OLSR 协议中，每个节点周期发送 HELLO 消息包和发送 TC 消息包来维护整个网络的拓扑结构，如果一些节点成为恶意节点，篡改一些消息及路由信息，导致一些正常节点无法进行通信，网络的完整性和安全性将受到威胁[3]，存在的安全漏洞有：

(1) 节点间没有进行身份的相互认证，一旦恶意节点成为某个节点的邻居，就有可能成为该节点的中继代理，对拓扑信息进行不转发、或者有选择性的转发等从而干扰拓扑图的形成。

(2) 攻击者发送一个目的主机已接收过的包，进行重放攻击，来达到欺骗系统的目的。

(3) 网络上传输的路由报文信息都是明文传输，容易被恶意节点进行篡改，导致错误路由信息，从而形成攻击。

针对这些潜在的威胁，进行安全策略的设计：在 OLSR 协议中每个节点所发送的报文都具有一个惟一的序列号，在完整性保护中序列号保证不被窜改，就可以有效抵抗报文重放攻击；对发送的信息利用数字签名及 hash[4] 函数进行身份验证及保持报文的完整性。文中对这设计的安全策略思想，采用自动验证互联网安全协议工具 AVISPA，根据 HLPSL 语言标准，进行代码编写，然后进行验证该设计思想是否达到既定的安全目标。

3 分析工具 AVISPA 及 OLSR 的 HLPSL 编码

3.1 AVISPA

AVISPA（Automated Validation of Internet Security Protocols and Applications）是一套建立和分析安全协议模型的工具，用于验证各种安全相关网络协议。利用 AVISPA 强大的自动化检测功能可以快速的对协议安全性进行验证，加快协议的开发过程。

AVISPA[5]融合了四种侧重点均有不同的分析终端，采用高层次协议规范语言 HLPSL，首先，使用 AVISPA 的标准协议描述语言 HLPSL 来定义用于验证的协议和被检验的安全目标。编写的代码要通过 HLPSL2IF 翻译工具转换成 IF(Intermediate Format) 语言，AVISPA 工具集中的分析终端可以直接读取 IF 语言，分析出安全目标是否达到。如果协议不安全，分析终端会给出导致此事件发生的攻击轨迹，并采用相应的安全策略进行防御。结构图如图 1 所示。

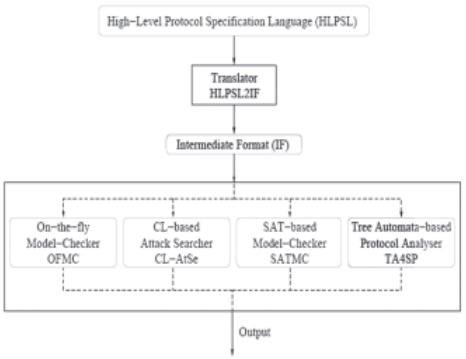

图 1 AISPA 分析工具结构图

3.2 HLPSL（高层次协议规范语言）

HLPSL 是基于角色 (role) 的语言，具有基于行为时序逻辑 TLA(Temporal Logic of Acitons) 的形式化语义，允许报文盲转发等，并且提供了控制流、强数据类型、可替换的入侵者模型以及自定义安全目标等强大功能。HLPSL 中定义的基本角色有协议参加的各方和代表场景的角色。在各个角色中可以设定一些初始参数。每一个角色都独立于其他角色，角色之间通过信道来通信。下面对使用 HLPSL 语言编写的协议脚本作简要介绍：

Roles（角色）：协议中角色包括两种类型：基本角色和构造角色。

Communication（通信）：HLPSL 中的通信式同步且发生在“channel”信道上，通常把信道命名为“SND”和“RCV”，分别表示发送信道和接收信道；用“SND(Msg)”和“RCV(Msg)”来表示发送和接收信息 Msg。

Intuder Models（攻击模型）：例如“SND,RCV：channel(dy)”表示协议的攻击模型，既攻击者可以通过信道窃听，截取，重放和修改协议中的信息。{ } _XXX 表示用 XXX 对 { } 进行加密／签名。

3.3 编写 OLSR 的 HSPSL 脚本

以 HLPSL 语言描述设计的安全 OLSR 协议代码定义了发送者 (sender)、接收者 (receiver) 角色，会话 (session) 角色，场景 (environment) 角色及代表验证目标的目标区段 (goal)（代码已简化）：

%% 组成场景的 session 角色 .

role session(S, R: agent, Hash: hash_func, PK_S: public_key, Thop1, Thop2: text)

def=

local SND, RCV, SNDS, RCVS: channel (dy)

composition

sender(S, R, SND, RCV, Hash, PK_S, Thop1, Thop2)

/\ receiver(R, S, SND, RCV, Hash, PK_S, Thop1, Thop2)

end role

%% 定义场景角色，整个脚本入口及入侵知识库

role environment ()

def=

const s,r:agent,f:hash_func,msg:protocol_id,pk_s,pk_r,pk_i:public_key,thop1,thop2:text

intruder_knowledge= {s, r, f, pk_s, pk_r, pk_i, inv(pk_i)}

composition

session(s, r, f, pk_s, thop1, thop2)

/\session(s, r, f, pk_r, thop1, thop2)

/\session(i, r, f, pk_i, thop1, thop2)

/\session(s, i, f, pk_s, thop1, thop2)

end role

goal

%% 表示对发送和接收者间的消息 msg 进行强认证

authentication_on msg

end goal

3.4 使用 AVISPA 验证 OLSR

运行脚本前，在 Linux 环境下先进行环境变量的设置：

export AVISPA_PACKAGE=/home/lcp/avispa

export PATH=$PATH：$AVISPA_PACKAGE

协议仿真分析终端使用分析引擎 OFMC(On-the-Fly Model-Checker)。协议仿真流程如图 2 所示。

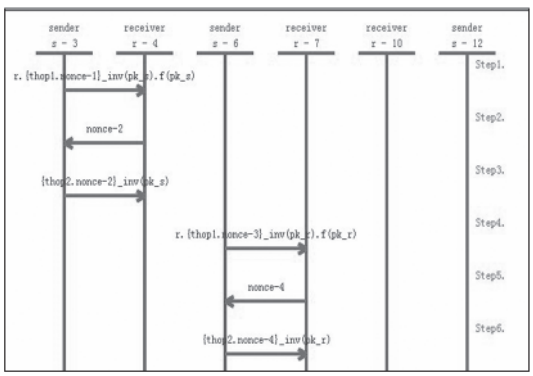

图 2 协议仿真事件流程

发送者与接收者间引入事件为：Past events:

(sender, 3)-> (receiver, 4):R. {Thop1.Msg}_inv(PK_S).Hash (PK_S)

(receiver, 4)-> (sender, 3): Nonce

(sender, 3)-> (receiver, 4) :{Thop2.Nonce}_inv(PK_S)

（下转第 34 页）

码，保证了Agent平台间的通信安全，也为其他系统应用SSL协议提供了参考。

参考文献:

[1] 张云勇，刘锦德．移动agent技术[M]．北京：清华大学出版社，2003:218-225.

[2] Dierks Allen C.The TLS Protocol(Version 1.0)[S]. RFC 2246,1999-01.

[3] Penserini L,Perini A,Susi A, et a1. From Stakeholder Intentions to Software Agent Implementations[C]f.Proc of the 18th Conf on Advanced Information Systems Engineering, 2006:465-479.

[4] 赵纪平．移动Agent安全方案的分析与研究[J]．通信技术，2008,12:275.

作者简介：**赵纪平**(1957-)，女，徐州师范大学物理与电子工程学院，高级实验师，主要研究方向：网络技术与安全，软件工程。

收稿日期：2010-05-26

（上接第31页）

(sender, 6)-> (receiver, 7):R. {Thop1.Msg}_inv (PK_S).Hash (PK_S)

(receiver, 7)-> (sender, 6): Nonce

(sender, 6)-> (receiver, 7) :{Thop2.Nonce}_inv (PK_S)

使用分析引擎OFMC在协议仿真过程中产生的入侵过程：

(sender, 12)-> (Intruder_, 0):R. (Thop1.Msg)_inv (PK_S).Hash (PK_S)

(Intruder_, 0)-> (sender, 12): nonce-1

(sender, 12)-> (Intruder_, 0) :{Thop2.Nonce}_inv (PK_S)

(sender, 3)-> (Intruder_, 0):R. (Thop1.Msg)_inv (PK_S).Hash (PK_S)

(Intruder_, 0)-> (receiver, 4):r. {thop1.nonce-2}_inv (pk_s).f (PK_S)

(Intruder_, 0)-> (sender, 3): nonce-1

(receiver, 4)-> (Intruder_, 0): nonce

(sender, 3)-> (Intruder_, 0) :{Thop2.Nonce}_inv(PK_S)

(sender, 6)-> (Intruder_, 0):r. {thop1.nonce-4}_inv (pk_r).f(pk_r)

(Intruder_, 0)-> (receiver, 7):r. {thop1.nonce-4}_inv (pk_r).f(pk_r)

(Intruder_, 0)-> (sender, 6): nonce-1

(receiver, 7)-> (Intruder_, 0): Nonce

(sender, 6)-> (Intruder_, 0) :{Thop2.Nonce}_inv (PK_S)

入侵仿真跟踪各个步骤的时间流程图如图3所示。

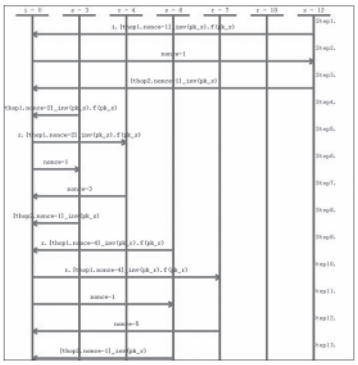

图3 入侵事件流程图

运行avispa OLSR.hlpsl－ofmc；输出结果下：

```
[root@localhost lcp]# cd avispa/testsuite/hlpsl
[root@localhost hlpsl]# export AVISPA_PACKAGE=/home/lcp/avispa
[root@localhost hlpsl]# export PATH=$PATH:$AVISPA_PACKAGE
[root@localhost hlpsl]# avispa OLSR.hlpsl --ofmc
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/lcp/avispa/testsuite/results/OLSR.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 2.55s
  visitedNodes: 2964 nodes
  depth: 12 plies
[root@localhost hlpsl]#
```

运行结果表明协议是安全的（SUMMARY：SAFE），达到了既定的安全目标；结果中的统计量还包含运行花费的搜索时间、访问的节点总数、深度及采用的入侵模型等。

4 结束语

随着网络的不断发展，对网络协议的安全性验证更加广泛，AVISPA工具使用高层次协议规范语言(HLPSL)，直观简便对协议进行形式化语言建模、提供了控制流、可替换的入侵者模型以及自定义安全目标等强大功能，能够快速准确的推导出分析结果和分析轨迹，加快了下一代网络协议安全性研究步伐。

参考文献:

[1]RFC3626 Optimized Link State Routing October 2003 http://hipercom.inria.fr/olsr/rfc3626.txt

[2]AVISPA Project http://www.avispa-project.org

[3]洪帆，洪亮，付才．一种安全移动自组网链路状态路由协议SOLSR.计算机科学，2005Vol.32No.11.

[4]Bart Preneel.The State of Cryptographic Hash Functions. Proceedings,ERUOCRYPT 96,1996；Published by Springer-Verlag.

[5]徐梦茗，李斌，肖聪．怎样用好AVISPA工具．信息安全，2009,8：154-158.

[6]BASIN D, MCDERSHEIM S.VIGANOL OFMC: a symbolic model checker for security protocols[J].International of information Security, 2004, 4(3):181-208.

作者简介：**林春平**(1978-)，男，硕士，助理工程师，研究方向：计算机网络、信息安全。

收稿日期：2010-06-12