建进程, 表述变量, 通过进程间信息传输等对模型进行描述[13]. 使用 Promela 语言对 Delta 协议进行如下描述.

Delta 协议的 Promela 描述

```
1.  chan notifFile=[1]of{byte};
                …                          /*定义全局消息通道*/
2.  chan deltaData=[1]of{byte};
3.  active proctype Library(){
4.  byte nF=1, sF=1, sD=1, dF=1, dD=1;
5.  byte rubbish;
6.  do                                     /*Library 进程, 模拟文
7.  ::notifFile!nF                         件生成*/
                …
8.  ::deltaData!dD
9.  od}
10. active proctype RP(){
11. byte getNoti;
                …
12. byte deltaState;
13. do
14. ::notifFile?getNoti;                   /*RP 进程变量定义和
15. ::(getNoti==0)->goto continue          状态转移*/
16. if
17. fi
18. ::(notiState==0)->goto refuse
                …
19. ::(notiState==2)->goto proDelta
20. proSnapsh:
                …
21. ::snapshFile?getSnapsh;
22. ::(getSnapsh==0)->goto continue
23. if
                …                          /*RP 进程中的
24. fi                                     Snapshot 处理状态*/
25. ::(snapshState==0)->goto refuse
26. progress1:
    snapshData?getSnapshData
27. ::(getSnapshData==0)->goto continue
28. goto continue
29. proDelta:
                …
30. ::deltaFile?getDelta;
31. ::(getDelta==0)->goto continue
32. if                                     /*RP 进程中的 Delta 处
                …                          理状态*/
33. fi
34. ::(deltaState==0)->goto refuse
35. progress2:deltaData?getDeltaData
36. ::(getDeltaData==0)->goto continue
37. goto continue
                …
38. refuse:
                …                          /*RP 进程中出错或循
39. continue:                              环转移状态*/
                …
40. od}
```

图 5 为 Promela 模型中各进程间信息传递过程及状态转移图. 在 Promela 模型中, 构建了两个进程 proctype_Library 和 proctype_RP 分别用以模拟 Delta 协议中 RPKI 资料库端和 RP 依赖方的运行状态. proctype_Library 进程对控制文件的生成和数据打包进行了模拟, 此进程为循环进程, 若产生文件生成或数据打包失败则循环, 生成的文件和数据都将被公用全局通道变量 notifFile、snapshFile、snapshData、deltaFile、deltaData 负载以供 proctype_RP 进程获取. proctype_RP 为 Delta 主要的协议逻辑模拟进程, 表 2 中为进程中变量与之对应的模拟状态和模型中取值.

表 2 Delta 协议模型进程内变量

进程内变量	模拟状态	值
getNoti	Notification 文件获取状态	
notiState	Notification 文件验证状态	
getSnapsh	Snapshot 文件获取状态	
getSnapshData	Snapshot 数据获取状态	0/1
snapshState	Snapshot 文件验证状态	
getDelta	Delta 文件获取状态	
getDeltaData	Delta 数据获取状态	
deltaState	Delta 文件验证状态	0/1/2

proctype_RP 进程中主要的循环逻辑在 do…od 循环体中, 表 2 中的状态变量则由 if…fi 结构内的语句进行随机的数值变换, 以表述文件验证或数据获取的成功与否, 根据状态数据表述的结果在逻辑处理之间使用 goto 语句进行跳转, 主要的三个处理逻辑部分分别为主循环体中的 Notification 文件处理逻辑、Snapshot 文件处理逻辑 proSnapsh 和 Delta 文件处理逻辑 proDelta. 同时在 Promela 模型中标注了下述语句: ::(1)->progress1: snapshData?getSnapshData 和::(1)->progress2:deltaData?getDeltaData 分别使用模型标记关键字 progress 用于指示 SPIN 在验证过程不允许出现从不执行语句 snapshData?getSnapshData 和 deltaData?getDeltaData 的循环发生, 因为此两条语句所表示的模型意义分别是从 proctype_Library 进程获取 Snapshot 数据和 Delta 数据, 为该验证模型必须可达的“可接受”状态.

(3) Delta 协议模型的 SPIN 验证

图 6 所示是由 Promela 模型生成的 Delta 协议逻辑自动机, 其本质与图 4 自动机相同, 只不过在 Promela 描述中加入了循环用转移状态, 所以略有差异.

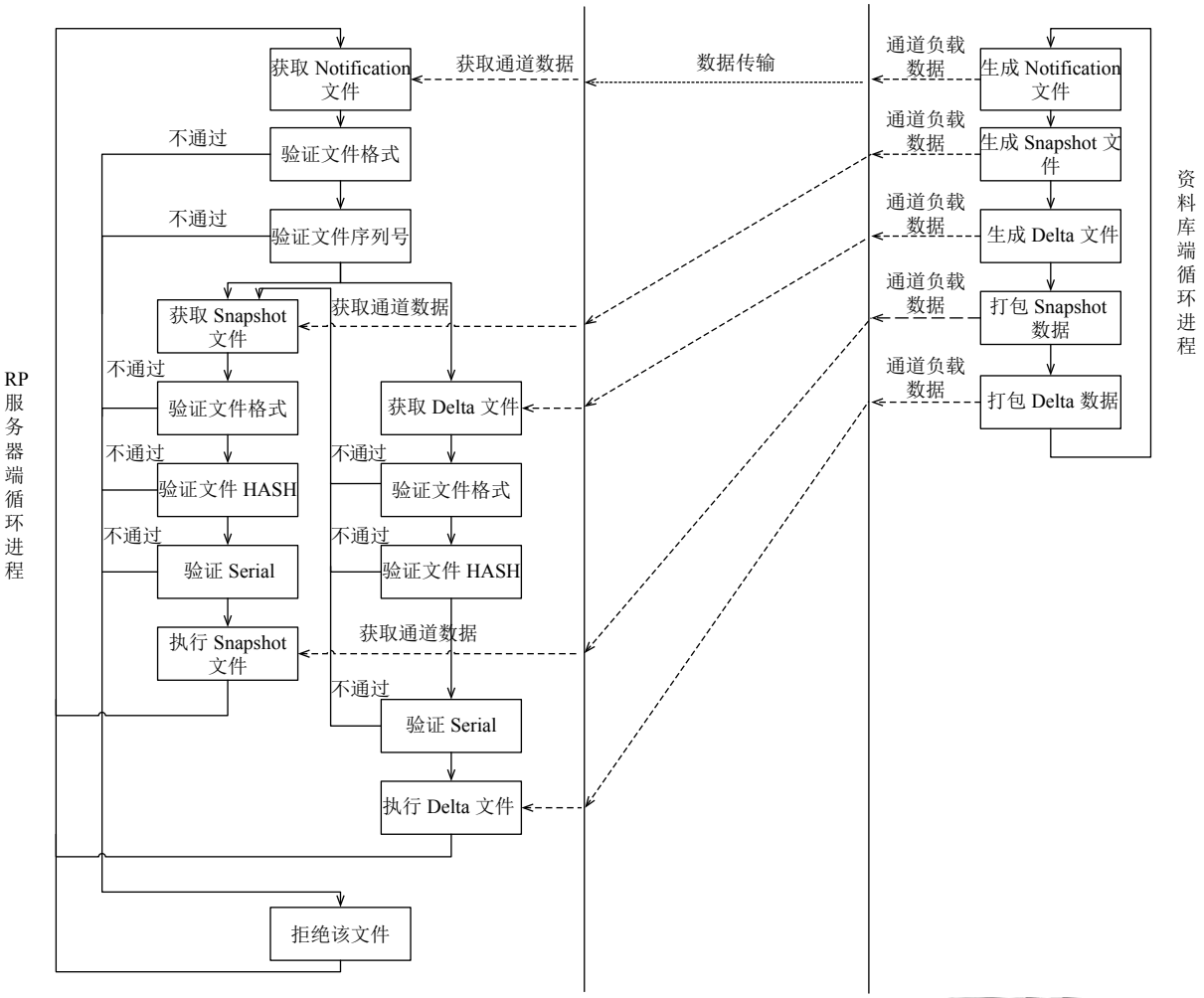

图 5 Delta 协议 Promela 模型

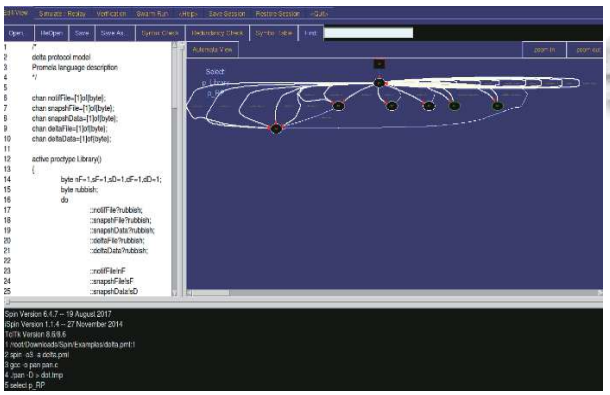

图 6 Promela 模型生成的自动机

图 7 为使用 SPIN 对 Delta 协议的 Promela 模型进行验证的结果, 验证结果表示 Delta 协议不存在“死锁”、“无效循环”等不安全协议特性, 同时其协议逻辑完全可达.

图 8 为 Promela 模型的模拟运行, 共进行 10 000 步模拟运行, 无任何报错, 协议稳定性较高.

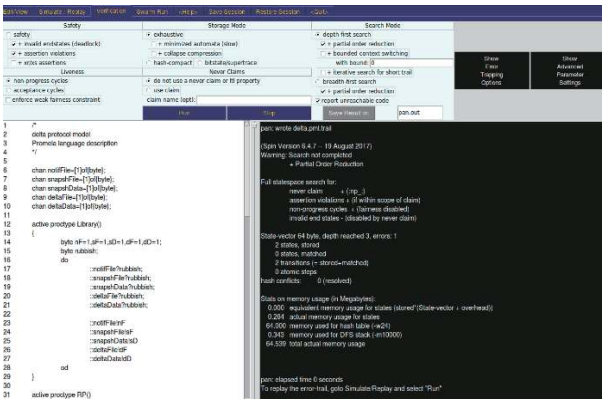

图 7 Promela 模型验证结果

通过上述验证过程, 可以从逻辑层面非常严密地

© 中国科学院软件研究所 http://www.c-s-a.org.cn

证明: Delta 协议不存在“死锁”、“无效循环”等不安全协议特性, 同时其协议逻辑完全可达, 具有非常高的协议安全性. 通过模拟运行则可以体现出其具备极高的稳定性.

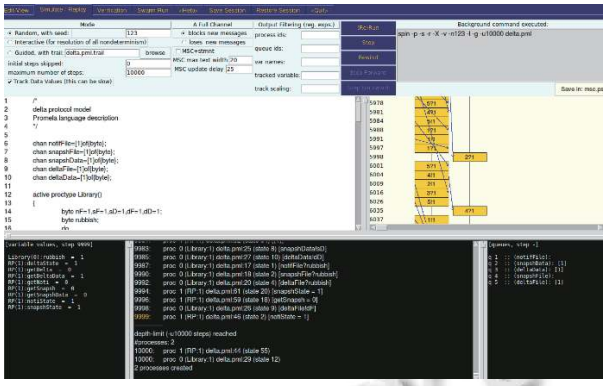

图 8 Promela 模型模拟运行

3 Delta 协议实现

Promela 构建的协议模型不仅可以对协议验证进行模拟, 同时由于具备完整的协议结构, 也可以在协议的实现中进行指导. 本文基于 Delta 的 Promela 模型, 使用 Python 对 Delta 协议进行了实现开发. 截止本文撰写, 该 Delta 功能为国内首次实现, 源码已呈现于 GitHub 供开源使用 https://github.com/sihaolin/RPKI-Delta-Protocol. 表 3 为该协议实现的各主要功能函数, 可以从逻辑上完整搭建 Delta 协议的工程架构, 望能对其他有需求的开发者提供参考和帮助.

4 总结

通过上文阐述, 可以看出 Delta 协议具有较高协议安全特性, 且其协议逻辑稳定. 相较于 RPKI 体系中早期使用的 Rsync 同步工具, Delta 协议的同步可控性得到大幅提升, 增量更新的方式也使得其更新效率大幅提高, 严密的控制文件格式验证和 HTTPS 协议对传输数据的加密也使得数据同步的安全性得到保障, Delta 协议对资料库服务器更少的资源占用则使得服务器在遭受 DDOS 攻击时具有更高的抵御力. Delta 协议已经较为成熟, 且具备 RPKI 体系所需的优秀特性, 在未来一段时间内将会完全替代 Rsync, 成为组成 RPKI 体系的重要组件.

表 3 Delta 实现的主要功能函数

Delta 协议主要功能函数	功能示意
delta_Read_Conf()	解析配置文件, 获取控制文件的 CA 分发点
obtain_File(current_Node)	获取当前控制文件
obtain_H_Sha256(file_Name)	生成文件的 HASH 值
validate_Notifica_File_Fo()	验证 Notification 格式
validate_Snapsh_File_Fo()	验证 Snapshot 格式
validate_Delta_File_Format()	验证 Delta 文件格式
parse_Notification_File()	解析 Notification 文件, 获取文件负载的信息
parse_Snapshot_File()	解析 Snapshot 文件, 构建元素树 s_E, 并返回数据
parse_Delta_File()	解析 Snapshot 文件, 构建元素树 d_E, 并返回数据
rsync_File(URI)	通过 URI 远程获取证书数据
processing_Snap_File(s_E)	传入元素树 s_E, 执行 Snapshot 文件
processing_Delta_File(d_E)	传入元素树 d_E, 执行 Delta 文件
main()	主体循环, 调用其他功能函数, 构建各个文件验证的主要协议逻辑, 并获取数据

参考文献

1 马迪, 沈烁. 基于本地信任锚点管理的 RPKI 安全运行机制研究. 电信科学, 2013, 29(9): 55–59. [doi: 10.3969/j.issn.1000-0801.2013.09.010]

2 贾佳, 延志伟, 耿光刚, 等. BGP 路由泄露研究. 网络与信息安全学报, 2016, 2(8): 54–61.

3 Ballani H, Francis P, Zhang XY. A study of prefix hijacking and interception in the internet. Proceedings of the 2007 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications. Kyoto, Japan. 2007. 265–276. [doi: 10.1145/1282427.1282411]

4 许圣明, 马迪, 毛伟, 等. 基于有序哈希树的 RPKI 资料库数据同步方法. 计算机系统应用, 2016, 25(6): 141–146. [doi: 10.15888/j.cnki.csa.005203]

5 刘晓伟, 延志伟, 耿光刚, 等. RPKI 中 CA 资源分配风险及防护技术. 计算机系统应用, 2016, 25(8): 16–22. [doi: 10.15888/j.cnki.csa.005313]

6 王娜, 杜学绘, 王文娟, 等. 边界网关协议安全研究综述. 计算机学报, 2017, 40(7): 1626–1648.

7 肖美华, 薛锦云. 基于 SPIN/Promela 的并发系统验证. 计算机科学, 2004, 31(8): 201–203, 208. [doi: 10.3969/j.issn.1002-137X.2004.08.059]

8 韩德帅, 杨启亮, 邢建春. 一种软件自适应 UML 建模及其形式化验证方法. 软件学报, 2015, 26(4): 730–746. [doi: 10.13328/j.cnki.jos.004758]

9 郭伟, 缪力, 张大方, 等. 基于 Spin 的 UML 状态图模型检

© 中国科学院软件研究所 http://www.c-s-a.org.cn

查的设计与实现. 计算机工程与应用, 2008, 44(10): 43–47. [doi: 10.3778/j.issn.1002-8331.2008.10.013]

10 Yang QL, Lv J, Tao XP, *et al*. Fuzzy self-adaptation of mission-critical software under uncertainty. Journal of Computer Science and Technology, 2013, 28(1): 165–187. [doi: 10.1007/s11390-013-1321-9]

11 Holzmann G J. The model checker SPIN. IEEE Transactions on Software Engineering, 1997, 23(5): 279–295. [doi: 10.1109/32.588521]

12 Yang WH, Xu C, Liu YP, *et al*. Verifying self-adaptive applications suffering uncertainty. Proceedings of the 29th ACM/IEEE International Conference on Automated Software Engineering. Vasteras, Sweden. 2014. 199–210. [doi: 10.1145/2642937.2642999]

13 Huston G, Michaelson G. RFC 6483 - Validation of route origination using the resource certificate Public Key Infrastructure (PKI) and Route Origin Authorizations (ROAs). BMC Public Health, 2012, 11(1): 1–6.

www.c-s-a.org.cn

www.c-s-a.org.cn

© 中国科学院软件研究所 http://www.c-s-a.org.cn