文章编号: 1001-0920 (2014) 12-2144-07 DOI: 10.13195/j.kzyjc.2013.1177

基于有色 Petri 网的时钟同步协议安全性分析

冯冬芹, 沈佳骏, 褚 健

(浙江大学 a. 工业控制技术国家重点实验室, b. 智能系统与控制研究所, 杭州 310027)

摘 要: 随着工业以太网的发展, 作为其实时性保障核心技术的时钟同步协议的安全性变得至关重要. 针对时钟同步协议的安全性问题, 首先提出一种基于有色 Petri 网的时钟同步协议安全性分析方法; 然后通过建立协议的有色 Petri 网模型, 利用状态方程等工具针对不安全状态的可达性进行判断分析, 从而实现时钟同步协议的安全性分析; 最后具体分析了一种基于精密时钟同步协议 (PTP) 的时钟同步协议以及针对该协议的主时钟欺骗攻击, 验证了所提出方法的有效性.

关键词: 工业以太网; 时钟同步协议; 有色 Petri 网; 状态方程; 主时钟欺骗

中图分类号: TP393 **文献标志码:** A

Analysis of clock synchronization protocol security using colored Petri net

FENG Dong-qin, *SHEN Jia-jun*, *CHU Jian*

(a. State Key Laboratory of Industrial Control Technology, b. Institute of Cyber-Systems and Control, Zhejiang University, Hangzhou 310027, China. Correspondent: FENG Dong-qin, E-mail: dqfeng@iipc.zju.edu.cn)

Abstract: With the development of the industrial ethernet, the clock synchronization protocol which is the core technology of industrial ethernet real-time has become crucial. For the problem of the clock synchronization protocol security, a method for analyzing the clock synchronization protocol security using colored Petri net is proposed. Firstly, the protocol is modeled by using colored Petri net. Then the reachability of possible insecurity state is judged and analyzed by using the model through state equation, thus the security analysis of the clock synchronization protocol is realized. Finally, a clock synchronization protocol based on PTP and the main-clock spoofing attack aiming at this protocol are analyzed by using this method, and the result shows the effectiveness of the proposed method.

Key words: industrial ethernet; clock synchronization protocol; colored Petri net; state equation; main-clock spoofing

0 引 言

以太网以其开放性好、应用广泛、价格低廉等优势逐步在过程控制领域的中上层 (如过程控制层、信息管理层) 网络中起到了主导作用, 并有进一步向下发展至下层 (如现场设备层) 网络的趋势, 在各类重要基础设施中起到了关键作用. 但是, 由于其自身的 CSMA/CD 机制以及设备层和 I/O 层上的数据采集与传输等问题, 使其暴露了在实时性上的严重缺陷[1-2]. 针对以太网实时性不足的问题, 出现了一些网络时钟同步技术, 例如 NTP 和 SNTP 协议, 但是以上两种协议均是针对毫秒级同步需求的大型分布式系统[3], 其同步精度以及收敛速度尚不能满足一些对时钟同步要求十分严格的分布式系统. IEEE 1588 精确时钟同步协议 (PTP) 的出现使得基于工业以太网的高精度同步控制成为可能, 该协议能达到微秒级至亚微秒级的同步精度[4]. 在时钟同步协议广泛应用于工业以太网领域的同时, 其安全性问题不容忽视. 目前, 针对 NIST-80 系列规范[5-6]中涉及的协议漏洞的相关研究尚未形成成熟体系, 因为许多工业通信协议在设计之初并没有考虑安全问题, 大部分通信建立在 IP 信任的基础上, 所以不可避免地会遭受安全攻击. 常见的如 TCP/IP 协议容易遭受 IP 欺骗、ARP 欺骗等攻击[7]. 同样的, 时钟同步协议也容易遭受主时钟欺骗等攻击, 容易造成关键设备的实时时钟被控制, 继而引起终端设备被破坏的严重后果.

基于上述问题, 本文将有色 Petri 网引入时钟同步协议安全性分析, 提出一种新的时钟同步协议安全性分析方法, 该方法属逆向状态分析方法, 主要分为

收稿日期: 2013-08-27; **修回日期:** 2013-12-12.

基金项目: 国家自然科学基金项目(61223004); 国家 863 计划 项目(2012AA041102).

作者简介: 冯冬芹(1968–), 男, 教授, 从事现场总线、工业控制系统等研究; 沈佳骏(1990–), 男, 硕士生, 从事现场总线、Petri网应用的研究.

以下4步: 1) 建立时钟同步协议的有色Petri网模型; 2) 分析时钟同步协议不安全状态; 3) 构建加入攻击者的有色Petri网模型与状态方程; 4) 分析攻击可达性. 在不安全状态分析过程中需要注意: 为了提升分析结果的真实性和可信性, 应尽可能将攻击者所具备的知识及能力最大化, 因此在加入攻击者的时钟同步协议模型的建立过程中, 本文默认攻击者能够截获、存储时钟同步过程中发送至公共信道的任意报文, 并有能力对截获报文中的关键数据进行修改, 同时重新组包发送, 实现伪造报文的目的. 在确定了不安全状态后, 可以根据加入攻击者行为的协议有色Petri网模型建立状态方程

$$M_n = M_0 + C^{\mathrm{T}} \times \sigma. \tag{1}$$

其中: M_0 表示初始状态; M_n 表示不安全状态的终止状态; C 表示该有色Petri网的关联矩阵; σ 表示变迁实施过程的向量. 据此方程可以求得该方程的解 σ, 此时, 若 σ 为非负整数解, 则能够由 σ 进一步确定一个攻击执行序列. 通过证明该攻击执行序列的合法性, 可以最终确定不安全状态 M_n 是否可达, 即攻击者能否借由该攻击执行序列(攻击路径)成功实施攻击(达到不安全状态 M_n).

本文所提方法解决了传统的Petri网状态分析方法(如可达树分析方法)经常出现的可达性分析不准确、状态空间爆炸、动态特性分析能力不足等方面的问题. 通过引入状态方程、状态矩阵等线性代数概念, 将可达性问题转化为线性方程求解问题, 进而借助线性代数计算工具, 使得计算复杂性得到一定程度的降低. 针对一种基于IEEE 1588 PTP协议的时钟同步协议及其遭受的主时钟欺骗攻击进行的具体分析证明了该方法的有效性.

1　时钟同步协议描述

时钟同步协议的基本运行机制: 在当前的网络或网段中, 通过特定的竞争条件(如时钟源的精度, 时钟源的IP、MAC等组态信息), 决选出当前网络或网段中最优的时钟源作为主时钟节点, 该节点按照设定的组态信息, 周期性发送同步报文, 该同步报文带有时间戳、目的IP等关键信息, 同一网络或网段中的剩余节点以从时钟节点的身份接收同步报文, 并提取报文中的关键数据进行相应补偿时间的计算, 同时将计算结果、时间戳以及本地IP等关键信息打包, 以相应报文的形式周期性地发送给主时钟, 最终实现通过同步算法进行对自身时钟的调整[8]. 协议简要描述如图1所示.

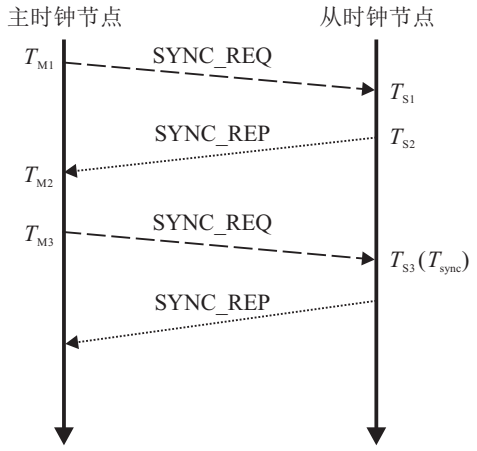


图 1　时钟同步协议描述

定义 1　SYNC_REQ为主时钟节点发送的同步请求报文, 包含主时钟节点IP、轮询节点IP、时间戳等信息.

定义 2　SYNC_REP为从时钟节点发送的同步响应报文, 包含相应节点IP及时间戳等信息.

定义 3　T_{M1}、T_{M2}、T_{M3} 为主时钟节点各个重要时间节点的时间戳, T_{M1} 为主时钟节点第1次发送同步请求报文的时间戳, T_{M2} 为主时钟节点第1次接收同步响应报文的时间戳, T_{M3} 为主时钟节点第2次发送同步请求报文的时间戳.

定义 4　T_{S1}、T_{S2}、T_{S3} 为从时钟节点各个重要时间节点的时间戳, T_{S1} 为从时钟节点第1次接收同步请求报文的时间戳, T_{S2} 为从时钟节点第1次发送同步响应报文的时间戳, T_{S3} 为从时钟节点第2次接收同步请求报文的时间戳.

定义 5　Delay为同步报文传输过程中的线路延时时间, Offset为补偿时间.

$$\mathrm{Delay} = \frac{(T_{S1} - T_{M1}) + (T_{M2} - T_{S2})}{2}, \tag{2}$$

$$\mathrm{Offset} = T_{S1} - T_{M1} - \mathrm{Delay}. \tag{3}$$

定义 6　T_{sync} 为从时钟节点同步后在原 T_{S3} 时间节点处的准确时间.

$$T_{\mathrm{sync}} = T_{M3} + \mathrm{Offset} + \mathrm{Delay}. \tag{4}$$

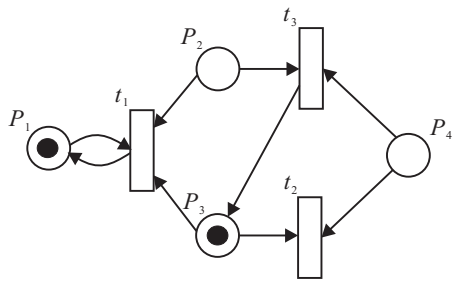
由主从时钟节点之间同步请求及同步响应报文交互可以得到4个相应的时间戳, 分别为 T_{M1}、T_{S1}、T_{S2}、T_{M2}. 由式(2)和(3), 可以得到主从时钟节点之间的线路延时时间以及补偿时间, 进而计算出在时间节点 T_{S3} 处的从时钟节点同步后的准确时间 T_{sync}, 随后的每个通信周期均按式(3)和(4)进行同步维护, 从而完成时钟同步过程.

2　可达树分析方法

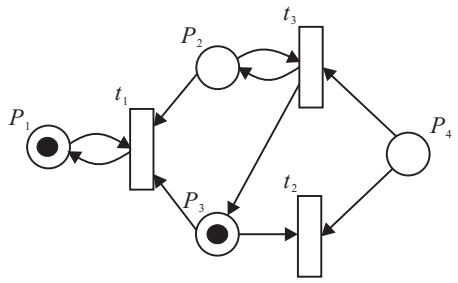
Petri网的分析手段一般可分为正向分析方法和逆向状态分析方法, 正向分析方法也是常用的可达树分析方法, 与本文中采用的逆向可达性分析方法相反, 该方法通过正向分析, 穷举出可能出现的库所及变迁, 建立Petri网模型, 转换得到相应的可达树模型. 在模

型中, 可达性分析一般通过判断可达树模型中某个树节点的可达性进而判定Petri网模型中对应标识的可达性. 该方法相较逆向状态分析方法, 在可达性分析准确性、状态空间爆炸、动态特性分析等方面仍存在一定缺陷.

图2为两个不同的Petri网模型, 在分别进行可达树模型转换后可以发现两者均对应于同一个可达树模型, 如图3所示. 但是, 图2(b)模型中 P_2 库所在变迁 t_1 发生以前, 其中的托肯数必定为偶数, 而图2(a)模型中 P_2 库所中的托肯数则可以是任意数. 虽然这两个Petri网模型具有相同的可达树模型, 但是其可达集显然是不同的, 这就表明单从可达树模型角度无法准确地对相应的Petri网进行可达性分析.



(a) Petri网模型a



(b) Petri网模型b

图 2 两个不同的Petri网模型

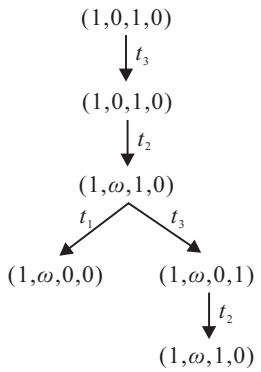


图 3 模型a与模型b相同的可达树模型

在实际应用中, 尤其是在针对复杂对象(如时钟同步协议)进行分析时, 由于对象运行机制的复杂性, 会产生大量的库所、变迁以及有向弧, 类似于图2(b)中所涉及到的一个库所同时连接 M 个(在图2(a)中, P_2 库所的 M 值为"2")来自相同变迁的有向弧的情景也会大量出现. 此时针对该库所中托肯的变迁过程分析的计算量将呈 $(N \times M)^P$ 倍增长, 其中 N 为该库所运行路径数(在图2(a)中, P_2 库所的值为"1"), P 为该类型库所的个数(在图2(a)中, 该类型的库所只有 P_2 一个, 所以值为"1"). 在实际的复杂系统中, 托肯的数量往往较大, 容易造成可达树模型中的可达性分析计算量呈指数倍增长. 此外, 通过穷举产生大量库所不仅是一个复杂的工作, 同时也会导致Petri网的状态空间爆炸问题, 从而使得相应的可达树模型规模过大, 引起无界性问题. 虽然目前已经存在通过引入无界符号 ω 将一个无界Petri网模型有限化以得到有限可达树模型的方法, 但该方法仍不能有效减小可达树模型的规模, 同时 ω 的引进也造成运行过程中一些动态信息的丢失, 极大地限制了可达树模型对Petri网模型动态特性的分析能力[9].

针对上述缺陷, 本文采用的逆向可达性分析方法是通过建立分析对象的有色Petri网模型和状态方程, 确定并记录模型运行过程中的任一状态(即动态信息), 这既有助于准确判定模型中任一状态的可达性, 又使该方法在模型动态特性方面的分析能力较前述可达树分析方法有较大提升. 此外, 逆向可达性分析方法采用了倒推的思想, 从结果出发构建模型, 规避了大量中间过程库所, 有效地解决了前述可达树分析方法中遇到的状态空间爆炸问题. 虽然该方法在针对复杂对象进行分析时, 存在状态转移矩阵规模较大的问题, 使得相关计算(如逆矩阵计算)过于繁琐, 但是相比于可达树模型尚无成熟的数学分析计算工具, 逆向状态分析方法由于引入了状态方程、状态矩阵等线性代数概念, 将可达性问题转化为线性方程求解问题, 进而借助成熟的线性代数计算工具(如Matlab、Mathematica等), 在一定程度上降低了计算的复杂性.

3 时钟同步协议的有色Petri网建模

有色Petri网是从经典Petri网转变而来的高级网系统之一, 其将经典Petri网系统中的托肯赋予一定的颜色以代表不同的实物, 大大减少了库所和变迁的数目. 与同样是高级网系统的谓词/变迁系统相比, 两者的区别在于对个性托肯的描述方法不同, 谓词/变迁系统为每个个体命名, 有色Petri网为不同个性的托肯染上不同颜色. 显然, 染色的方法既能体现个性(不同色), 又能体现共性(同色), 相较于一一命名的方法更优. 这也使得有色Petri网系统成为Petri网应用的主流模型之一.

定义 7 一个有色Petri网满足下列条件[10]:

$$\mathrm{CPN} = (P, T, F, C, W, M). \tag{5}$$

其中: P 是一个有限集合, 称为库所集, 在有色 Petri 网模型中通常用小圆圈表示; T 是一个有限集合, 称为变迁集, 在有色 Petri 网模型中通常用小矩形表示; $F \subseteq (P \times T) \bigcup (T \times P)$ 是有向弧的集合, 在有色 Petri 网模型中通常用有向弧来表示; C 是一个非空有限类型集, 称为颜色集, $C = \{c_1, c_2, \cdots, c_k\}$; $W : F \to \{0, 1, \cdots\}^k$, 即映射是对每条有向边赋予一个 k 维非负整数向量; $M : P \to \{0, 1, \cdots\}^k$, 即映射 M 对每个库所赋予一个 k 维非负整数向量.

据此建立时钟同步协议的有色 Petri 网模型, 如图 4 所示.

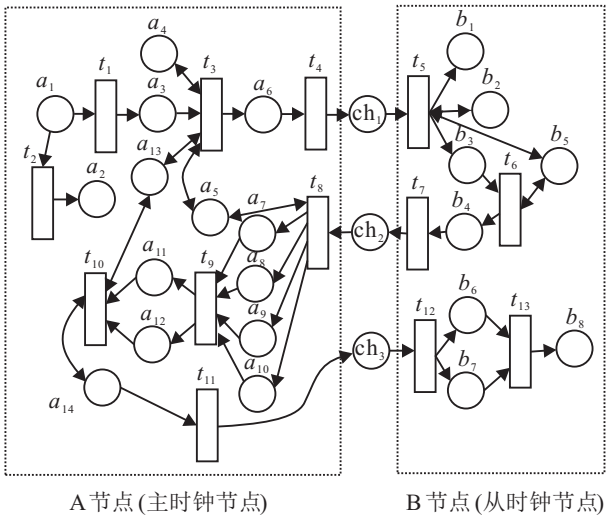


图 4　时钟同步协议有色 Petri 网模型

图 4 中, A 节点为主时钟节点, B 节点为从时钟节点. 各个节点的库所中均含有各自相关的颜色托肯, 比如主时钟标志位库所 a_1 含有用以表示主时钟状态标志位的颜色托肯, 同步请求报文库所 a_6 含有用以表示具体同步请求报文数据的颜色托肯. 由于颜色托肯数量较大, 并未一一标注, 由颜色托肯组成的颜色集 $C = \{c_1, c_2, \cdots, c_{16}\}$, 即含有 16 个不同的颜色托肯.

具体颜色集元素为

$$c_1 = \mathrm{Flag}_{\mathrm{main\_clk}},\ c_2 = D_{\mathrm{A}},\ c_3 = \mathrm{IP}_{\mathrm{A}},$$

$$c_4 = \mathrm{IP}_{\mathrm{B}},\ c_5 = \mathrm{CLK}_{\mathrm{A}},\ c_6 = P\{D_{\mathrm{A}}, \mathrm{IP}_{\mathrm{B}}\},$$

$$c_7 = [\mathrm{A\_SEND\_REQ}]_{1\mathrm{st}},\ c_8 = \mathrm{CLK}_{\mathrm{B}},$$

$$c_9 = P\{[\mathrm{A\_SEND\_REQ}]_{1\mathrm{st}}, \mathrm{CLK}_{\mathrm{B}}\},$$

$$c_{10} = [\mathrm{I\_RECV\_REQ}]_{1\mathrm{st}},$$

$$c_{11} = [\mathrm{I\_SEND\_REP}]_{1\mathrm{st}},$$

$$c_{12} = [\mathrm{A\_RECV\_REP}]_{1\mathrm{st}},$$

$$c_{13} = \mathrm{Delay}_{\mathrm{AB}},\ c_{14} = \mathrm{Offset}_{\mathrm{AB}},$$

$$c_{15} = P\{\mathrm{Delay}_{\mathrm{AB}}, \mathrm{Offset}_{\mathrm{AB}}, \mathrm{IP}_{\mathrm{A}}\},$$

$$c_{16} = \mathrm{SYNC\_CLK}_{\mathrm{BA}}.$$

为了便于颜色集的描述, 定义了如下表述方式.

1) Flag 表示标志位, [·] 表示取方括号内相关操作发生时刻的时间戳.

2) D 表示组包数据.

3) CLK 表示实时时间戳, 即节点时钟.

4) SYNC_CLK 表示同步后的实时时间戳.

5) $P\{\cdot\}$ 表示将括号内的数据元素进行组包.

6) Delay 表示线路延时时间.

7) Offset 表示补偿时间.

8) 单字母下标表示所属对象, 例如 D_A 表示 A 节点的组包数据; 双字母下标表示两者间对应关系, 例如 $\mathrm{Delay}_{\mathrm{AB}}$ 表示 A 节点相对于 B 节点的线路延时时间, 1_{st} 下标表示第 1 次发送.

主时钟标志位库所 a_1 经变迁 t_1 验证为"1"后, 将轮询 IP 值库所 a_4 中的 IP 值、A 节点实时时间戳库所 a_{13} 中的同步请求报文发送时间以及其余组包数据库所 a_3 中的数据封装成同步请求报文 (即 SYNC_REQ), 报文经总线发送至 B 节点. B 节点收到报文后, 解析得到主时钟的 IP 值库所 b_1 以及 A 节点第 1 次发送时间的时间戳库所 b_3, 同时通过 B 节点自身的实时时间戳库所 b_5 记录接收到报文的时间, 重新组包并以同步响应报文 (即 SYNC_REP) 的形式由 B 节点发回 A 节点. A 节点经过变迁 t_8 对接收到的同步响应报文进行解析, 得到 A 节点发送同步请求报文时间戳库所 a_7、B 节点接收同步请求报文时间戳库所 a_9 以及 B 节点发送同步响应报文时间戳库所 a_{10}, 同时记下 A 节点自身接收同步响应报文的时间戳库所 a_8, 经过变迁 t_9 的计算可以得到线路延时时间库所 a_{11} 和补偿时间库所 a_{12}, 再将计算结果发送至 B 节点, 最终算出 B 节点同步后时间库所 b_8, 完成整个时钟同步过程.

4　协议安全风险及不安全状态可达性分析

由时钟同步协议有色 Petri 网模型可以发现, 攻击者如果能够截获主从时钟节点之间的重要通信报文 (即 A 节点发送 SYNC_REQ 报文库所 ch_1、B 节点发送 SYNC_REP 报文库所 ch_2 以及 A 节点再次发送 SYNC_REQ 报文库所 ch_3 这 3 个主要通信交互库所中的报文), 就可以通过不同的手段实现对该时钟同步协议的攻击, 实施主时钟欺骗行为, 切断真正主从时钟节点之间的通信.

具体实施的主时钟欺骗攻击主要可以分为中间人攻击、拒绝服务攻击以及克隆攻击. 在这 3 种攻击方式中, 拒绝服务攻击和克隆攻击均会造成一个或多个时钟同步节点的运行异常, 虽然能成功实施攻击行为, 达到攻击目的, 但是隐蔽性不足, 相较这两种攻击

方式, 中间人攻击威胁较大, 且隐蔽性好, 不易被探测.

中间人攻击是攻击者通过自己的主机插入两个目标主机通信路径之间, 使其成为两个目标主机相互通信的一个中继[11-12]. 为了不中断通信, 攻击者将设置自己的主机转发来自两个时钟同步节点之间的同步报文, 可以实现在不被察觉的情况下实施攻击, 即通过在主从时钟节点之间插入攻击者节点, 截获并转发图4所描述模型里关键通信库所 ch_1、ch_2、ch_3 中的报文, 以达到切断主从时钟节点通信而不被察觉的目的. 在本文针对主时钟欺骗攻击具体模型的构建过程中, 主要以中间人攻击作为研究对象进行建模.

结合协议安全风险分析和已建立的协议有色Petri网模型, 可以得到加入中间人攻击模型的协议有色Petri网模型, 如图5所示.

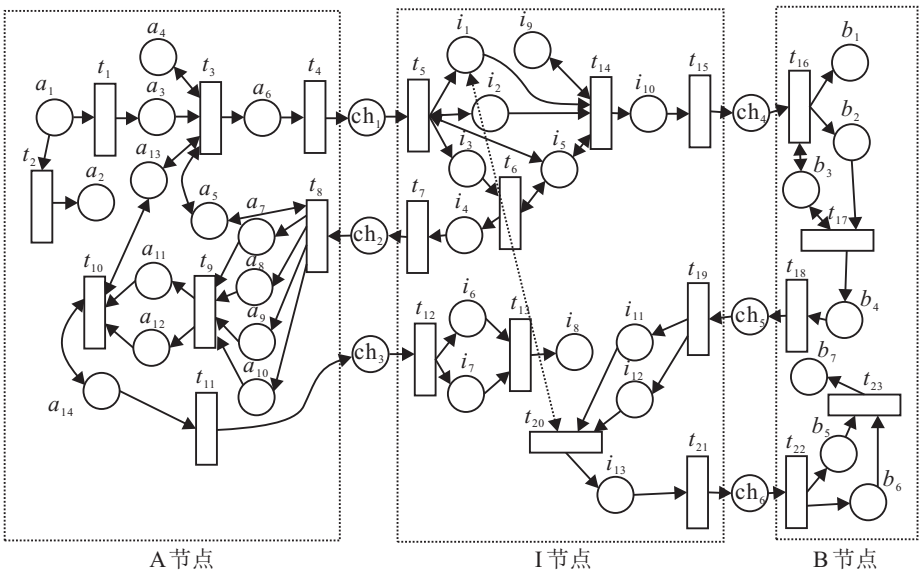


图5 加入中间人攻击模型的时钟同步协议有色Petri网模型

在图5中加入了I节点(即中间人攻击节点), 该节点通过对A节点同步请求报文和B节点同步响应报文的拦截, 切断两者之间的通信, 同时伪造发送给B节点的同步请求报文和发送给A节点的同步响应报文, 最终造成A、B节点均误以为同步完成(即攻击不被察觉), 实际上B节点的同步时钟是由I节点的伪造线路延时时间 i_{11} 和伪造补偿时间 i_{12} 计算得到, 即攻击者可以任意控制B节点的实时时钟.

在该有色Petri网模型中, 颜色集 $C = \{c_1, c_2, \cdots, c_{26}\}$, 即含有26个不同的颜色托肯, 具体颜色集元素为

$$c_1 = \mathrm{Flag}_{\mathrm{main\_clk}},\ c_2 = D_{\mathrm{A}},\ c_3 = \mathrm{IP}_{\mathrm{A}},$$
$$c_4 = \mathrm{IP}_{\mathrm{B}},\ c_5 = \mathrm{CLK}_{\mathrm{A}},\ c_6 = P\{\mathrm{D}_{\mathrm{A}}, \mathrm{IP}_{\mathrm{B}}\},$$
$$c_7 = [\mathrm{A\_SEND\_REQ}]_{1\mathrm{st}},\ c_8 = \mathrm{CLK}_{\mathrm{I}},$$
$$c_9 = P\{[\mathrm{A\_SEND\_REQ}]_{1\mathrm{st}},\ \mathrm{CLK}_{\mathrm{I}}\},$$
$$c_{10} = [\mathrm{I\_RECV\_REQ}]_{1\mathrm{st}},$$
$$c_{11} = [\mathrm{I\_SEND\_REP}]_{1\mathrm{st}},$$
$$c_{12} = [\mathrm{A\_RECV\_REP}]_{1\mathrm{st}},$$
$$c_{13} = \mathrm{Delay}_{\mathrm{AI}},\ c_{14} = \mathrm{Offset}_{\mathrm{AI}},$$
$$c_{15} = P\{\mathrm{Delay}_{\mathrm{AI}},\ \mathrm{Offset}_{\mathrm{AI}},\ \mathrm{IP}_{\mathrm{A}}\},$$
$$c_{16} = \mathrm{SYNC\_CLK}_{\mathrm{IA}},\ c_{17} = D_{\mathrm{I}},$$
$$c_{18} = P\{\mathrm{IP}_{\mathrm{A}}, \mathrm{IP}_{\mathrm{B}}, D_{\mathrm{I}}, \mathrm{CLK}_{\mathrm{I}}\},\ c_{19} = \mathrm{IP}_{\mathrm{I}},$$
$$c_{20} = [\mathrm{I\_SEND\_REQ}]_{1\mathrm{st}},\ c_{21} = \mathrm{CLK}_{\mathrm{B}},$$
$$c_{22} = P\{[\mathrm{I\_SEND\_REQ}]_{1\mathrm{st}},\ \mathrm{CLK}_{\mathrm{B}}\},$$
$$c_{23} = \mathrm{Delay}_{\mathrm{IB}},\ c_{24} = \mathrm{Offset}_{\mathrm{IB}},$$
$$c_{25} = P\{\mathrm{Delay}_{\mathrm{I}},\ \mathrm{Offset}_{\mathrm{I}}, \mathrm{IP}_{\mathrm{A}}\},$$
$$c_{26} = \mathrm{SYNC\_CLK}_{\mathrm{BI}}.$$

同时可以得到该有色Petri网中的 K (K 为颜色集中元素的个数, 该模型中为26) 维向量 $x_1 \sim x_{26}$, 即经过映射后用以表示各有向弧的非负整数向量. 正如Petri网的一个标识可以表示成一个向量一样, 同样能够以矩阵的形式对Petri网的结构进行描述, 进而通过引入状态方程、状态矩阵等线性代数概念, 将整个Petri网的性质分析过程转化为线性代数求解问题, 这样, Petri网运行过程中的动态信息也能够通过向量的形式进行描述, 提升了针对模型动态特性的分析能力.

定义8 设 $\Sigma = (P, T, F, M_0)$ 为一个Petri网, $P = \{p_1, p_2, \cdots, p_m\}$, $T = \{t_1, t_2, \cdots, t_n\}$, 则 Σ 可以用一个 n 行 m 列的矩阵 $A = [a_{ij}]_{n\times m}$ 来表示, 并称 A 为 Σ 的关联矩阵[13].

$$a_{ij} = a_{ij}^{+} - a_{ij}^{-},$$
$$i \in \{1, 2, \cdots, n\},\ j \in \{1, 2, \cdots, m\}. \qquad (6)$$

其中

$$a_{ij}^{+}=\begin{cases}1,\ (t_i,p_j)\in F;\\ 0,\ \text{else};\end{cases}\tag{7}$$

$$a_{ij}^{-}=\begin{cases}1,\ (p_j,t_i)\in F;\\ 0,\ \text{else}.\end{cases}\tag{8}$$

定义8仅描述了普通Petri网中关联矩阵的定义, 对于CPN $=(P,T,F,C,W,M)$, 同样能够引入关联矩阵的相关概念. 设$|P|=m$, $|T|=n$, 此时关联矩阵为n行m列矩阵, 与普通Petri网中不同的是, CPN中A的各个元素均为K维向量.

若$M\in R(M_0)$, 则存在n维非负整数向量X满足状态方程

$$M=M_0+A^{\mathrm{T}}X.\tag{9}$$

此时,

$$A^{\mathrm{T}}X=[A_{*1}^{\mathrm{T}}X,A_{*2}^{\mathrm{T}}X,\cdots,A_{*m}^{\mathrm{T}}X]^{\mathrm{T}}=\Big[\sum_{i=1}^{n}A_{i1}X(i),\sum_{i=1}^{n}A_{i2}X(i),\cdots,\sum_{i=1}^{n}A_{im}X(i)\Big]^{\mathrm{T}}.\tag{10}$$

其中: A_{*j}代表矩阵A的第j列, $A^{\mathrm{T}}X$的第j个分量值为$\sum_{i=1}^{n}A_{ij}X(i)$.

据此可以得到该有色Petri网模型的关联矩阵. 由于在求解过程中使用的均为关联矩阵的逆矩阵, 为了便于分析, 本文中直接列出关联逆矩阵, 并根据库所分类, 将其拆分为4个逆子矩阵, 分别对应A节点库所子矩阵、B节点库所子矩阵、I节点库所子矩阵以及子矩阵, 如表1～表6所示(篇幅所限, 未列出全零行、列).

表 1　A节点库所关联逆子矩阵 C_1^{T}

	t_1	t_2	t_3	t_4	t_8	t_9	t_{10}	t_{11}
a_1	$-x_1$	$-x_1$						
a_2		x_1						
a_3	x_1							
a_6			x_6	$-x_6$				
a_7					x_7	$-x_7$		
a_8					x_{12}	$-x_{12}$		
a_9					x_{10}	$-x_{10}$		
a_{10}					x_{11}	$-x_{11}$		
a_{11}						x_{13}	$-x_{13}$	
a_{12}						x_{14}	$-x_{14}$	
a_{14}							x_{15}	$-x_{15}$

表 2　I节点库所关联逆子矩阵 C_1^{T}-1

	t_5	t_6	t_7	t_{12}	t_{13}
i_3	x_7	$-x_7$			
i_4		x_9	$-x_9$		
i_6				x_{13}	$-x_{13}$
i_7				x_{14}	$-x_{14}$
i_8					x_{16}

表 3　I节点库所关联逆子矩阵 C_1^{T}-2

	t_5	t_{14}	t_{15}	t_{19}	t_{20}	t_{21}
i_1	x_3	$-x_3$				
i_2	x_4	$-x_4$				
i_{10}		x_{18}	$-x_{18}$			
i_{11}				x_{23}	$-x_{23}$	
i_{12}				x_{24}	$-x_{24}$	
i_{13}					x_{25}	$-x_{25}$

表 4　$\mathrm{ch}_1\sim\mathrm{ch}_6$ 库所关联子矩阵 C_3^{T}-1

	t_4	t_5	t_7	t_8	t_{11}	t_{12}
ch_1	x_6	$-x_6$				
ch_2			x_9	$-x_9$		
ch_3					x_{15}	$-x_{15}$

表 5　$\mathrm{ch}_1\sim\mathrm{ch}_6$ 库所关联子矩阵 C_3^{T}-2

	t_{15}	t_{16}	t_{18}	t_{19}	t_{21}	t_{22}
ch_4	x_{18}	$-x_{18}$				
ch_5			x_{22}	$-x_{22}$		
ch_6					x_{25}	$-x_{25}$

表 6　B节点库所关联子矩阵 C_4^{T}

	t_{16}	t_{17}	t_{18}	t_{22}	t_{23}
b_1	x_{19}	$-x_{19}$			
b_2	x_{20}	$-x_{20}$			
b_3		x_{22}	$-x_{22}$		
b_4				x_{23}	$-x_{23}$
b_5				x_{24}	$-x_{24}$
b_6					x_{26}

初始状态为$M_0^{\mathrm{T}}=[0,0,\cdots,0]$, 共计34个元素, 向量的维数由总的库所个数(在该模型中为40)减去自锁库所(即仅与双向有向弧连结的库所)的个数(在该模型中为6)计算所得. 该34维列向量分别代表了$a_1\sim a_3$, $a_6\sim a_{12}$, a_{14}, $i_1\sim i_4$, $i_6\sim i_8$, $i_{10}\sim i_{13}$, $b_1\sim b_2$, $b_4\sim b_7$以及$\mathrm{ch}_1\sim\mathrm{ch}_6$共34个库所的初始状态.

不安全的终止状态为: $M_n^{\mathrm{T}}=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,x_{16},0,0,0,0,0,0,0,0,0,0,x_{26},0,0,0,0,0,0]$, 即攻击者节点(I节点)以从时钟节点(B节点)的身份与主时钟节点(A节点)同步成功, 并捕获相应IP及时间戳信息, 同时该攻击节点还以主时钟节点身份与从节点同步成功, 通过可任意设定的线路延时时间与补偿时间实现对从时钟节点实时时钟的控制.

将M_0、M_n以及关联矩阵代入式(1), 经移项可得

$$M_n-M_0=C^{\mathrm{T}}\times\sigma.\tag{11}$$

由于M_0为零向量, 式(12)可以化简为

$$M_n=C^{\mathrm{T}}\times\sigma.\tag{12}$$

线性代数中非齐次线性方程组解的判别定理: 线性方程组$Ax=b$有解的充分必要条件是系数矩阵A的秩与增广矩阵$\widetilde{A}$的秩相同, 即

$$\mathrm{rank}(A)=\mathrm{rank}(\widetilde{A}).\tag{13}$$

本例中, 系数矩阵 A 为 C^{T}, 增广矩阵 $\widetilde{A}$ 为 (C^{T}, M_n), $\mathrm{rank}(C^{\mathrm{T}}) = \mathrm{rank}(C^{\mathrm{T}}, M_n)$, 据此可以判定该方程有解, 但此时并不足以判定该不安全终止状态是可达的. 因此进一步求得式 (13) 的解, 即 $\sigma = (C^{\mathrm{T}})^{-1} \times M_n = [0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1]^{\mathrm{T}}$, 由此非负整数解可以确定攻击者的攻击实施序列 (即有色 Petri 网模型中的变迁可执行序列) 为 $t_3t_4t_5t_6t_7t_8t_9t_{10}t_{11}t_{12}t_{13}t_{14}t_{15}t_{16}t_{17}t_{18}t_{19}t_{20}t_{21}t_{22}t_{23}$, 该攻击序列的实际运行从A节点第1次组SYNC_REQ报文并发送 (即变迁 t_3、t_4) 开始, 至 B 节点与伪装成 A 节点的 I 节点完成同步 (即变迁 t_{23}) 结束. 期间, A 节点与伪装为 B 节点的 I 节点完成同步 (即变迁 t_{13}). 至此可以确定该攻击序列的合法性, 进而判定该不安全终止状态 M_n 是可达的, 最终证明该时钟同步协议中存在的不安全性.

5 结 论

从上述的分析可以看出, 攻击者能够通过截获、篡改并转发图 4 所描述模型里关键通信库所 ch_1、ch_2、ch_3 中的关键报文, 成功实施中间人攻击, 从而达到切断主从时钟节点同步通信以及操纵从时钟节点实时时钟的攻击目的. 为了防止上述攻击的实现, 可以在主从节点进行通信时增加身份认证环节, 该环节的增加可以通过在本协议执行前, 先执行另一个身份认证协议的方式实现.

在工业以太网已被广泛应用于大量重要基础设施的大背景下, 时钟同步协议作为工业以太网中的关键核心技术, 其安全性问题至关重要. 针对此问题, 本文基于有色 Petri 网, 结合逆向状态分析方法的思想, 引入线性代数概念, 提出了一种新的时钟同步协议安全性分析方法, 该方法有效解决了传统分析方法中存在的可达性分析不准确、状态空间爆炸、动态特性分析能力不足等问题, 并在一定程度上降低了计算的复杂性. 同时将其应用于一种针对时钟同步协议的主时钟欺骗攻击, 建立相应模型并进行分析, 分析结果验证了本文所提方法的有效性.

参考文献(References)

[1] 缪学勤. 论六种实时以太网的通信协议[J]. 自动化仪表, 2005, 26(4): 1-3.
(Miu X Q. Expositionof six types of communication protocols of real-time Ethernet[J]. Instrument and Meter for Automation, 2005, 26(4): 1-3.)

[2] 王燕山, 李运华, 刘恩朋, 等. 以太网时间同步技术的研究进展及其应用[J]. 测控技术, 2007, 26(4): 4-6.
(Wang Y S, Li Y H, Liu E P, et al. Research and application of Ethernet time synchronization[J]. Measurement and Control Technology, 2007, 26(4): 4-6.)

[3] 王飞雪, 盛红岩, 常勇. 基于 Linux 系统的 SNTP 协议的研究与应用[J]. 现代电视技术, 2007(2), 102-104.
(Wang F X, Sheng H Y, Chang Y. The study and application of SNTP protocol under Linux[J]. Advanced Television Engineering, 2007(2), 102-104.)

[4] Hans Weibel. IEEE 1588 Tutorial[C]. Conf on IEEE 1588. Maryland, 2006: 1-56.

[5] NIST 800-82, Guide to industrial control systems (ICS) Security[S].

[6] NIST 800-42, Guideline on network security Testing[S].

[7] 郑文兵, 李成忠. ARP 欺骗原理及一种防范算法[J]. 江南大学学报, 2003, 2(6): 574-577.
(Zheng W B, Li C Z. An algorithm against attacks based on ARP spoofing[J]. J of Jiangnan University, 2003, 2(6): 574-577.)

[8] Eidson J, Agilent L, Lee K. IEEE standard for a precision clock synchronization protocol for networked measurement and control systems[C]. Sensors for Industry Conference. Houston: IEEE, 2002: 98-105.

[9] 汤利杰. 基于可达树对无界 Petri 网性能检测的方法研究[D]. 成都: 成都理工大学信息科学与技术学院, 2012.
(Tang L J. Research on property checking for unbounded Petri nets based on reachability trees[D]. Chengdu: College of Information Science & Technology, Chengdu University of Technology, 2012.)

[10] 吴哲辉. Petri 网导论[M]. 北京: 机械工业出版社, 2006: 181-188.
(Wu Z H. The introductory theory of Petri net[M]. Beijing: China Machine Press, 2006: 181-188.)

[11] Sean Whalen. An introduction to ARP spoofing revision 1.82[EB/OL]. (2002-05-27)[2003-07-15]. http://packetstormsecurity.org/papers/protocol/.

[12] Yuri Volobuev. ARP and ICMP redirection games[EB/OL]. (1997-09-19)[2000-03-05]. http://www.insecure.org/.

[13] Muratu T. Petri nets: Properties, analysis and application[J]. Proc of The IEEE, 1989, 77(4): 541-580.

(责任编辑: 齐 霁)