基于SmartVerif的比特币底层协议算力盗取漏洞发现

包象琳[1,2],熊 焰[2],黄文超[2],陈凯杰[2],汪万森[2],孟昭逸[2],徐晓峰[1],方贤进[3]
(1. 安徽工程大学计算机与信息学院,安徽芜湖 241000; 2. 中国科学技术大学计算机科学与技术学院,安徽合肥 230026;
3. 安徽理工大学计算机科学与工程学院,安徽淮南 232001)

摘 要: 比特币引入了一种新的P2P(Peer to Peer)交易方法,并依靠其底层协议实现去中心化交易.然而,由于目前缺乏对比特币各底层协议的细粒度形式化分析和系统建模,比特币安全性并未被保证.本文通过设计多维度的比特币安全模型引理和细粒度的比特币模型规则,系统地抽象了多协议组合运行考虑下的比特币协议实体交互,完成了对比特币的形式化符号建模与自动化安全分析.与以前的工作相比,本文更细粒度地建模了比特币协议实体及其相关操作,并全面设计了满足比特币各实体需求的安全属性.此外,本文利用自动化形式化验证系统SmartVerif实现了无需额外手工推导证明的形式化验证实验,通过将本文所建模的符号模型规则与引理作为SmartVerif的输入,发现了比特币底层协议算力盗取攻击.

关键词: 比特币; 区块链; 协议安全; 符号模型; 形式化分析
中图分类号: TP309 **文献标识码:** A **文章编号:** 0372-2112(2021)12-2390-09
电子学报URL: http://www.ejournal.org.cn **DOI:** 10.12263/DZXB.20201194

Detection of the Computational Power Stealing Attack in Bitcoin Protocols Based on SmartVerif

BAO Xiang-lin[1,2], XIONG Yan[2], HUANG Wen-chao[2], CHEN Kai-jie[2], WANG Wan-sen[2], MENG Zhao-yi[2],
XU Xiao-feng[1], FANG Xian-jin[3]
(1. *School of Computer and Information, Anhui Polytechnic University, Wuhu, Anhui* 241000, *China*;
2. *School of Computer Science and Technology, University of Science and Technology of China, Hefei, Anhui* 230026, *China*;
3. *College of Computer Science and Engineering, Anhui University of Science and Technology, Huainan, Anhui* 232001, *China*)

Abstract: Bitcoin introduces a new P2P(Peer to Peer) trading method to the public and relies on its protocols to achieve decentralization. However, due to the lack of fine-grained formal analysis and systematic modeling of bitcoin, the security of bitcoin protocols is not guaranteed. We provide a comprehensive symbolic analysis of bitcoin protocols. Our work develops bitcoin model rules and model lemmas with the abstraction of in-protocol interactions and inter-protocols interactions. We refine the modeling of bitcoin protocol entities and their related protocol operations compared to the previous work, along with the comprehensive design of the security lemmas that bitcoin protocols should guarantee. The symbolic model rules and lemmas developed in our work are inputted into the automatic formal verification system SmartVerif. We find the computational power stealing attack and no additional manual work is needed for the model verification with the help of the SmartVerif.

Key words: bitcoin; blockchain; protocol security; symbolic model; formal analysis

1 引言

比特币[1~4]是一种近年来十分流行的去中心化电子加密货币,其安全性和可用性依赖于比特币底层协议.目前比特币总市场价值约为1700亿美元,总日营业额约为800亿美元.与此同时,比特币的高价值也吸引着众多黑客对比特币实施恶意攻击[5],这些攻击可对电子加密货币平台及比特币使用者造成巨大损失:加密货币结算平台Cryptsy在2016年因遭受黑客攻击而损失了当时

收稿日期:2020-10-26;修回日期:2021-04-19;责任编辑:梅志强
基金项目:国家自然科学基金(No.61972369, No.61572453, No.61572454); 国家重点研发计划(No.2018YFB2100300); 安徽工程大学校级科研项目(No.XJKY2020122, No.XJKY2020119, No.XJKY2020120); 安徽工程大学引进人才科研启动基金项目(No.2020YQQ062)

价值约473万美元的比特币,比特币交易所CoinSecure在2018年因受恶意攻击损失了当时市值约337万美元的比特币,一名匿名比特币投资者在2020年2月因黑客攻击而损失了当时价值超1440万美元的比特币.

学术界近年来对比特币展开了激烈讨论,众多研究人员开始着眼于对比特币安全性的探讨. Bastiaan研究了比特币的双重支付攻击[6];Miller讨论了比特币网络中区块链分叉的可能性与比特币矿工算力分布之间的关系[7];Eyal提出了用于攻击者分叉比特币区块链的最优自私挖掘策略[8]. 在比特币实体交互安全性的形式化验证方面,现有工作基于计算方法[9]分析了破坏比特币共识安全性的恶意攻击. Garay为比特币共识协议提供了一个基于计算方法的形式化模型,并手工证明了比特币的共识安全[10,11];Kiayias基于Garay的形式化建模工作,对比特币区块链分叉概率与区块链链增长属性间的关联性进行研究[12]. Poulami则面向比特币确定性钱包,基于计算方法形式化证明了钱包安全性[13]. 然而基于计算方法构建的比特币模型验证需人工参与,受限于模型复杂度和状态空间. 在形式化分析比特币交互安全性时,现有工作[10~12]着眼于共识安全验证,对不同实体功能进行合并,构建粗粒度比特币模型,此种建模会导致协议设计缺陷无法被有效发现[14,15].

基于以上考虑,本文利用符号方法[9]面向比特币协议交互进行了细粒度的模型构建及其安全属性验证. 本文从四个不同维度对比特币安全属性进行了符号化定义,所建模的形式化模型不仅抽象了各个比特币协议独立执行下的实体状态转换,还对协议间共享协议项的交互进行了定义,实现了对比特币协议的细粒度形式化建模,并借助自动化形式化验证系统SmartVerif[16]发现了比特币底层协议算力盗取攻击. SmartVerif是一种具备通用性的全自动化形式化验证系统,需要安全形式化分析人员输入已被合理建模的形式化模型引理和模型规则,将对已输入的模型进行全自动化的推理,最终输出模型引理的验证结果和被发现的攻击路径. SmartVerif使用了强化学习的手段,突破了传统形式化验证系统所面临的状态空间爆炸问题,已成功对多个目前最具挑战的安全协议进行了全自动证明.

2　比特币敌手模型和安全属性的设计

2.1　比特币协议实体及攻击者模型设计

本文对比特币协议实体间的通信信道进行了不同安全级别的假设,并且对可能的恶意比特币协议实体进行了符号化定义. 本文假设拥有比特币的买家P和支持比特币支付的商家M之间协商订单支付的通信信道是认证信道,假设矿池服务器F与节点N之间的通信信道是安全信道,假设交易双方和具备跨账本功能的比特币节点N之间的通信信道是完成身份验证的. 与此同时,本文假设商家商品价格是一个公开的常数;假设买家P在进行初始化时已完成待支付订单的秘密发送;假设商家M在初始化时,已完成商家标识及其电子货币钱包地址的声明.

本文基于Dolev-Yao敌手模型赋予比特币攻击者计算能力,假设比特币攻击者可以通过其侵蚀的恶意协议实体,获得恶意实体的初始化信息、应当被保护的秘密协议项及其通信信道的控制权,这些恶意实体会主动将所有可掌握的协议消息分享给比特币攻击者. 本文在建模比特币协议执行的不同安全情形时,设置了不同的参与协议执行的恶意实体,包括已被攻击者侵蚀的恶意买家、恶意商家、恶意矿池服务器、恶意矿池矿工以及恶意的具备跨账本功能的比特币节点.

2.2　比特币安全属性的设计

本文形式化定义了覆盖五个比特币协议实体安全需求的安全属性,提供了四个不同安全维度的比特币安全属性符号定义:基于洛氏分类法[17]对比特币认证属性进行了符号化定义;基于Ralf对可问责性的研究[18],完成了对比特币可问责性的符号化抽象及形式化定义;参考Jannik对隐私性和金额一致性的定义[19],完成了比特币隐私属性和金额一致性的符号化定义. 本文将这些符号化定义的安全属性开发为可作为自动化验证系统SmartVerif输入的形式,验证过程无需额外手工推理. 接下来,本文对比特币认证性、可问责性、隐私性和金额一致性这四个维度定义的比特币安全引理进行了解释,其中:变量前缀~用于标识新生成的变量;变量前缀$用于标识公开变量,变量前缀#用于标识时间变量;@符号后接时间变量,表示对动作的时间约束,@后的时间变量可省略前缀#.

认证性　本文认证性表示比特币需要保证协议执行的单射一致性. 下方引理定义了比特币支付协议中买家与商家间的单射一致性,如果一个已注册比特币身份m的商家通过协议动作Claim_pmt宣称已与拥有身份标识p的买家完成对订单商品的协商一致,那么买家p必定也通过协议动作Claim_runpmt宣称自己已执行与对应商家协议执行的支付协议来对订单商品进行付款操作. 此外,对于同一待支付订单,协议动作Claim_pmt只能发生一次,否则本次执行对应的交互实体商家m和买家p中至少有一个已被攻击者侵蚀.

$$\begin{aligned}&\forall p,m,x,\#i_1.\mathrm{Claim\_pmt}(p,m,x)@i_1\Rightarrow\\&(\exists\#j.\mathrm{Claim\_runpmt}(p,m,x)@j)\&\neg\\&(\exists\#i_2.\mathrm{Claim\_pmt}(p,m,x)@\#i_2\\&\&\neg(i_1=i_2))\vee(\exists\#r.\mathrm{Corrupt}(p)@r)\vee\\&(\exists\#r.\mathrm{Corrupt}(m)@r)\end{aligned}$$

相似地,本文定义了矿池矿工与矿池服务器之间所

需满足的单射一致性，拥有身份标识w的矿池矿工与拥有身份标识f的矿池服务器之间需要对合作挖矿任务s协商一致. 为实现对比特币模型共识安全属性的定义以及对模型的自动化形式化验证，本文假设比特币节点角色是一个参与比特币共识但没有独立节点身份标识的整体，因此本文没有对独立比特币节点的认证性进行定义.

可问责性 本文可问责性表示任意恶意比特币协议实体可被正确识别、任意诚实比特币协议实体不被错怪. 比特币攻击者侵蚀诚实比特币协议实体e的协议动作Corrupt(e)会在对应的本文所设置的对应的不安全情景里发生. 下面的引理为本文定义的恶意矿池服务器可识别性安全引理，如果身份标识为f的矿池服务器通过协议动作Claim_sol，为身份标识为w的矿池矿工支付求解合作挖矿任务s的薪酬，则被支付薪酬的矿池矿工应该是合作挖矿任务s的真正求解者，并已通过协议动作Claim_smbsol将任务解答提交给矿池服务器f，否则矿池服务器f是恶意的.

$$\forall w,f,s,\#i.\text{Claim\_sol}(w,f,s)@i \Rightarrow$$
$$(\exists \#j.\text{Claim\_smbsol}(w,f,s)@j|\exists \#r.\text{Corrupt}(f)@r)$$

相似地，本文定义了可正确找出抵赖已确认支付订单的恶意商家可识别性、产生冲突交易的恶意买家可识别性、拒绝更新账本记录账户余额的恶意跨账本节点可识别性以及提交错误求解的恶意矿池矿工可识别性安全引理.

隐私性 观察等价可以用来推理两个系统(例如协议的两个实例)是否无法被区分，通常用来证明隐私性[19]. 由于观察等价是定义隐私性的标准选择，本文利用观察等价对比特币隐私性进行定义. 非形式化地，如果攻击者与协议实例进行任意交互，且无法对协议实例进行区分，则协议是观察等价的. 下方引理将比特币交易者标识用p表示，本文定义若比特币协议执行中存在痕迹Secret(p)，则p是秘密的，那么在协议执行中不存在显示比特币攻击者成功得知比特币交易者标识p的协议动作痕迹$K(p)$. 只有当下方引理在比特币交易者标识作为区分项的协议执行实例中都成立时，比特币满足观察等价. SmartVerif的使用者可利用diff运算符建模协议执行实例的区分项，本文通过diff$(\sim p_1,\sim p_2)$告知工具协议执行实例的区分项为比特币交易者标识$\sim p_1$及$\sim p_2$.

$$\forall p,\#i.\ \text{Sec ret}(p)@i \Rightarrow (\neg(\exists \#k.K(p)@k)).$$

金额一致性 正文如下方引理所示，本文定义的金额一致性意味着只要比特币节点执行了一条比特币交易，那么该节点需要通过协议动作Exe_tx完成支付x_1所需比特币交易的执行，即减少地址a对应账户余额、增加地址b对应账户余额，与此同时，任意协议执行中不存在重复执行待转账金额来自同一地址a的比特币交易所对应协议动作痕迹Exe_tx，可检测双重支付攻击是否存在.

$$\forall a,b,x_1,\#i.\text{Exe\_tx}(a,b,x_1)@i \Rightarrow$$
$$(\neg(\exists c,x_2,\#k.\text{Exe\_tx}(a,c,x_2)@k)\&\neg(\#i=\#k)).$$

3 比特币协议形式化建模

3.1 比特币网络划分

本文细粒度建模了五个比特币协议实体以及比特币生命周期所涉及的四个比特币协议. 如图1所示，本文将比特币网络划分为四个比特币子网络并分别命名为比特币交易网络、比特币共识网络、区块计算网络以及其他货币系统网络. 四个比特币子网络中的协议实体通过执行四个待验证的比特币协议实现与其他实体的交互. 在比特币交易网络中，买家P与商家M之间执行比特币支付协议以实现安全的比特币支付. 在比特币共识网络中，比特币共识节点N通过执行比特币共识协议以实现一致的比特币交易确认. 在区块计算网络中，矿池服务器F与矿池矿工W执行矿池协议以实现公平的合作性挖矿. 在其他货币系统网络中，具备多重身份的比特币共识节点N执行跨账本协议与交易者以实现安全的跨账本交易. 相关形式化分析工作定义了比特币共识协议的交互细节以及共识节点实体功能涉及的协议操作，但未对交易涉及的其他协议进行细粒度形式化建模.

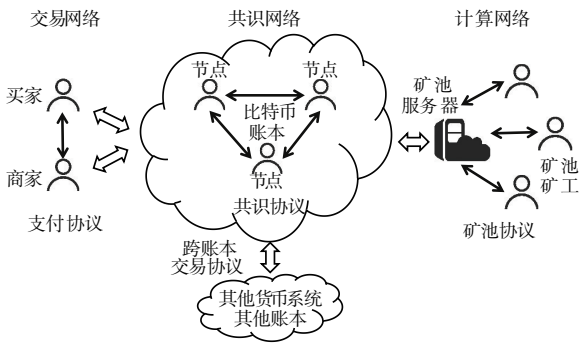

图1 比特币协议及其实体组成

3.2 协议实体交互

图2为本文SmartVerif模型的完整实体交互及状态转换图，圆圈里标注了模型的五个协议实体，水平虚线箭头标识了实体交互方向，箭头上方标注了协议交互消息，垂直箭头标识实体状态转换方向，箭头两侧标注了状态转换对应模型规则. 图中加粗标注了跨账本交易协议涉及状态转换. 下面本文将结合图2介绍比特币实体交互的消息细节.

买家将支付协议订单消息Order(A_p, A_M, v, t_v, o)发送给商家，其中A_p为买家新生成的钱包支付地址，A_M为收款地址，v为订单金额，t_v为支付有效期，o为订单备注信息. 商家在收到订单消息后，创建支付请求并向买家发送消息PaymentRequest(A_M, v, t_v, o, S_M)，其中S_M为商家使用商家服务器公开证书对整个支付请求的签名. 买家在收到支付请求消息后，向共识节点广播消息

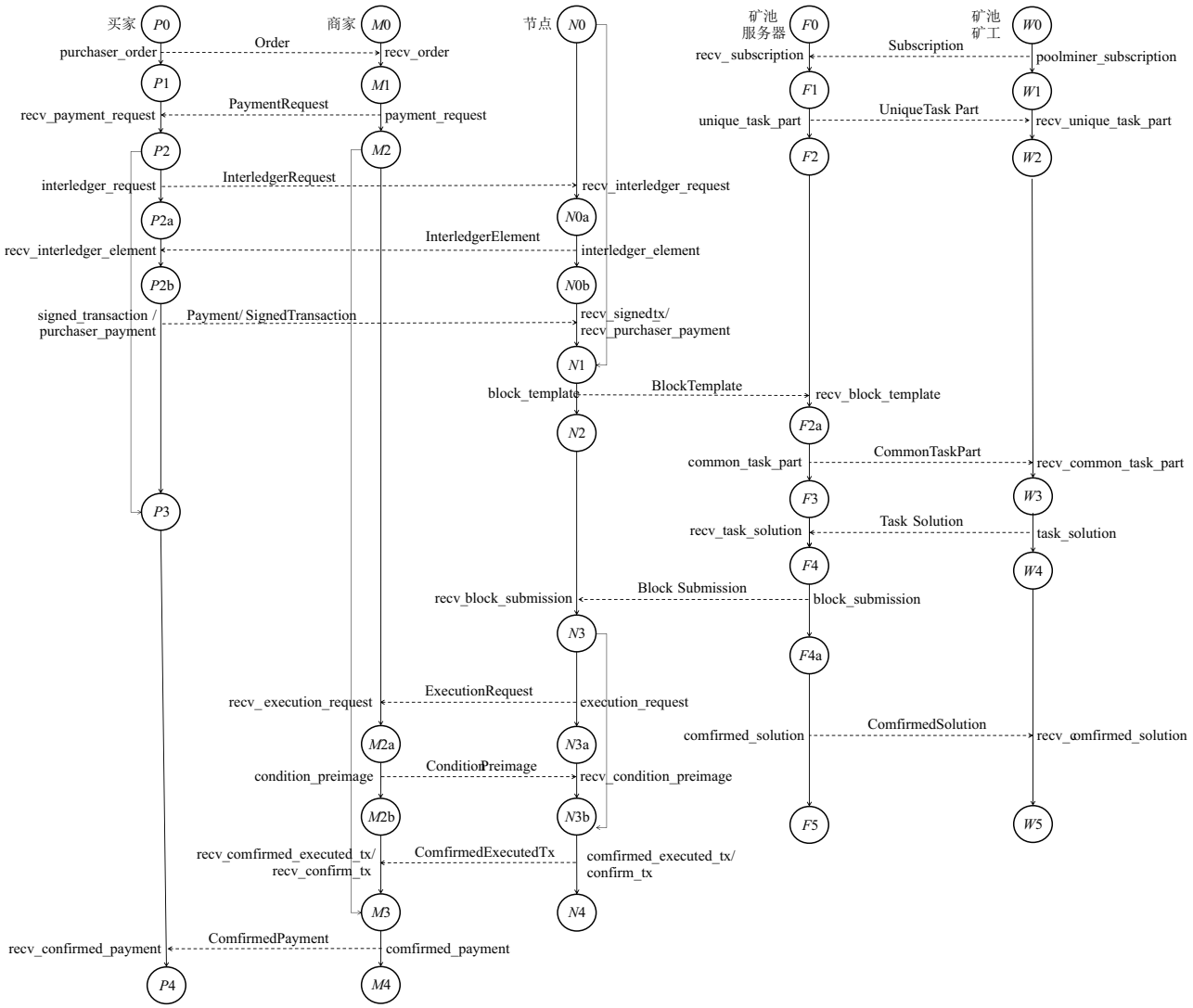

图2　比特币协议及其实体组成

Payment(U_P, A_P, v, A_M, S_P),其中U_P为买家存储的未花费交易标识,S_P是买家比特币钱包签名,商家在确认交易后,发送 ConfirmedPayment(U_P, A_P, v, A_M, S_P, o),并授权发货已完成支付的订单.

若其订单支付的完成需要跨链,则需生成跨链交易条件原象$h^{-1}(c)$并交互对应交易执行条件c. 买家向支持跨链协议的跨链节点发送消息 InterledgerRequest (A_P, A_M),支持跨链协议的节点将 InterledgerElement (A_N, r, i)反馈给等待跨链交易的买家,其中A_N为跨链节点钱包地址,r为跨链汇率,i为跨链手续费. 买家构造带签名交易消息 SignedTransaction(U_P, A_P, $v\times r$, U_N, c, S_P),以实现跨链交易执行有条件触发. 此时跨链节点记录跨链交易并构造及发布跨链子交易. 在区块链更新后,向商家发送执行请求 ExecutionRequest(U_N, A_N, v, A_M, c, S_N),其中U_N为跨链节点存储的未花费交易标识. 跨链商家在收到交易执行请求消息后,查询对应买家的跨链协议交易执行条件原象,并发布消息 ConditionPreimage(U_N, A_N, v, A_M, c, $h^{-1}(c)$),随后跨链节点完成对应交易的执行.

收到 Payment/SignedTransaction(U_P, A_P, v, A_M, S_P)后,节点广播 BlockTemplate(B_h, B_{pre}, B_r, B_d, T_P, T_M),其中B_{h}为区块高度,B_{pre}为父区块哈希,B_r为区块奖励,B_d为区块难度,T_P为U_P对应买家交易,T_M为U_M对应商家未花费交易;节点收到消息后,若非矿池,则独立挖矿并发布 BlockSubmission(B_{com}, B_h, B_{pre}, B_{merkle}, B_{nonce}, B_{time}, B_d, B_{tx}),其中B_{com}为被提交区块的支持算力,B_{merkle}为交易集合默克尔树根,B_{nonce}为区块挖掘任务的正确求解,B_{time}为区块实际出块时间,B_{tx}为被记录在区块的交易集合. 若节点具备矿池功能,则通过合作挖矿计算区块,广播消息 BlockSubmission(B_r, B_h, B_{pre}, B_{merkle}, B_{nonce}, B_{time}, B_d, B_{tx}),并在收到执行触发消息后执行跨链交易,随后共识节点广播 ConfirmedExecutedTx(U_P, A_P, v, A_M, S_p).

在进行合作挖矿之前，矿池矿工首先需要向矿池服务器发送矿池协议任务消息Subscription(W, W_{cab})，其中W为矿池矿工公共标识，W_{cab}为计算能力．矿池服务器发送独立任务部分消息UniqueTaskPart(W, S_{id}, I_{fix})，其中S_{id}为会话标识，固定任务信息I_{fix}=〈F, S_{id}, W_{cab}〉，F为矿池公钥标识，矿池发送CommonTaskPart(F, W, I_{change})，其中变化任务信息I_{change} = chptsk(B_h, B_{pre}, B_{mtb}, B_{time}, B_d)，chptsk为任务公共部分生成函数，B_{mtb}为交易默克尔树树枝．矿池矿工构造并计算解答后，发送矿池任务求解消息TaskSolution(W, F, I_{share})，其中I_{share} = share(F, S_{id}, W_{cab}, I_{change}, B_{en2}, B_{time}, B_{nonce})，share为任务求解函数，B_{en2}为矿池协议合作挖矿任务额外求解空间对应随机数解答．矿池服务器在收到任务解答消息后，验证区块合法性并提交比特币新区块，发布矿池任务解答确认消息ConfirmedSolution(F, W, I_{fix}, I_{change})，对应矿池矿工获得任务奖励．

3.3 比特币符号模型设计

本文为比特币交互构建了SmartVerif模型，图2描述了模型所抽象的全部状态转移和实体交互，其中交互的消息项详见3.2．图2所示模型规则的执行需要对应规则前件的状态事实存在于比特币当前状态集合，SmartVerif工具将存储规则的执行痕迹以验证模型安全属性．模型规则执行后，对应结果状态事实将被SmartVerif添加到当前状态集合中，前件状态事实则被移除．为便于读者复现模型，下面本文将分别给出支付协议、跨链交易协议、共识协议、矿池协议以及协议间实体交互的典型规则代码及其解释，其余规则代码可通过图2所示模型状态转移及3.2节所述消息项的定义，结合下方典型规则代码及解释推知．

图3为矿池协议建模的比特币模型规则task_solution及recv_block_submission的代码，其中task_solution模型规则符号化表示了矿池矿工与矿池服务器的交互和状态转换．该规则前件要求矿池矿工处于状态事实St_W3(对应图2状态W3)，其中状态事实St_W3中存储了矿池服务器密钥k、矿池服务器标识F、矿池矿工标识W、合作挖矿任务固定部分fixtaskinfo(对应3.2节变量I_{fix})、会话标识sbid(对应3.2节变量S_{id})、矿池任务可变部分changetaskinfo(对应3.2节变量I_{change})，其中任务固定部分由F、sbid以及矿工计算能力cab(对应3.2节变量W_{cab})组成，任务可变部分由区块高度bh、父区块哈希hpb、交易集合默克尔树根MTB、区块时间bt以及区块难度bd决定(分别对应3.2节变量B_h、B_{pre}、B_{mtb}、B_{time}、B_d)，本文定义chptsk五元函数用于生成任务可变部分；该规则前件要求生成矿池任务额外求解域解答~en2、生成区块时间状态信息~st、生成矿池任务区块求解域解答~nonce(分别对应3.2节变量B_{en2}、B_{time}、B_{nonce})．该规则在执行后，矿池矿工转移到状态事实St_W4(对应图2状态W4)，状态事实St_W4存储了F、W、fixtaskinfo、sbid、changetaskinfo以及对应矿池任务的解答结果shareinfo(对应3.2节变量I_{share})，本文定义了五元函数share用于生成矿池任务解答结果；规则执行后矿池矿工向对应矿池服务器发送任务解答消息．该规则定义规则被触发时，矿池解答提交者不能为矿池服务器；定义SmartVerif工具存储的规则执行痕迹为任务解答消息TaskSolution以及解答提交动作Claim_subsol所包含的消息项W、F、shareinfo．

```
rule task_solution:
    let subscrinfo =〈W,cab〉
      fixtaskinfo =〈F,sbid,cab〉
      hpb = h1(pb)
      changetaskinfo = chptsk(bh,hpb,MTB,bt,bd)
      shareinfo = share(fixtaskinfo,changetaskinfo,
                    ~en2,~st,~nonce)
      txnminfo =〈uidP,pkP,sV,pkM,sigP〉
      MTB = h1(txnminfo)
      utxoinfo =〈uidP,pkP,Vp〉
      chainmsg =〈bh,pb,rwd,bd,utxoinfo〉 in
    [ St_W3(k,F,W,fixtaskinfo,sbid,changetaskinfo)
    , Fr(~en2), Fr(~st), Fr(~nonce)]
    --[ Claim_subsol(W,F,sbid,shareinfo)
    , TaskSolution(W,F,sbid,shareinfo)
    , InEq(F,W)]->
    [ Out( 〈W,F,shareinfo〉 )
    , St_W4(k,F,W,fixtaskinfo,sbid,changetaskinfo,shareinfo)]
rule recv_block_submission:
    let fixtaskinfo =〈F,sbid,cab〉
      utxoinfo =〈uidP,pkP,Vp〉
      chainmsg =〈bh,pb,rwd,bd,utxoinfo〉
      hpb = h1(pb)
      txnminfo =〈uidP,pkP,sV,pkM,sigP〉
      MTB = h1(txnminfo)
      changetaskinfo = chptsk(bh,hpb,MTB,bt,bd)
      shareinfo = share(F,sbid,cab,changetaskinfo,
                    en2,bt,nonce)
      TID0 = uidP
      txcbinfo = txcb(TID0,FIX,rwd,F,shareinfo)
      MTR = h2(h5(TID0,FIX,rwd,F,shareinfo),MTB)
      txsinfo =〈TID0,FIX,rwd,F,sbid,en2,txnminfo〉
      block =〈bh,hpb,MTR,nonce,bt,bd,txsinfo〉 in
    [ St_N2(CN,pkP,pkM,sbr,chainmsg)
    , In( 〈br,block〉 )]
    --[ RecvBlockSubmission(block)
    , NotSmaller(br,sbr)]->
    [ St_N3(CN,pkP,pkM,br+sbr,block)]
```

图3 task_solution及recv_block_submission规则代码

相较于需要手工推理的现有的形式化分析工作，本文基于符号化方法完成了对比特币协议的细粒度系统化建模，并且完成了对本文比特币符号模型待验证安全引理的自动化形式化验证．本文不仅抽象了由单一比特币协议执行状态转换，还抽象了多比特币协议组合执行下考虑协议间交互的状态转换对应规则，下面以图2所示recv_block_submission模型规则代码为例进行说明，该模型代码涉及对协议模块间实体交互的建模，对矿池协议中共识网络与矿池服务器的交互与状态转换进行了符号化表示．该规则前件要求共识网络处于状态事实St_N2(对应图2状态N2)，状态事实St_N2中存储了共识网络标识CN、交易发送者公钥地址pkP、交易接收者公钥地址pkM、当前区块算力sbr(分别对应3.2节变量N、A_P、A_M、B_{com})以及本地存储区块链信息chainmsg(该元组

包含3.2节变量B_h、B_{pre}、B_r、B_d、B_{tx});要求共识网络必须接收到区块提交消息. 该规则在执行后,共识网络转移到状态事实St_N3(对应图2状态N3),状态事实St_N3存储了CN、pkP、pkM、新区块支持算力br+sbr以及被提交新区块block,其中block包含的消息项有区块高度bh、父区块哈希hpb、交易集合默克尔树根MTR、区块求解域解答nonce、区块时间bt、区块难度bd以及待共识交易集合txsinfo(分别对应更新后的3.2节变量B_h、B_{pre}、B_{merkle}、B_{nonce}、B_{time}、B_d、B_{tx}). 该规则定义规则被触发时,新区块支持算力不能小于本地存储区块的支持算力;定义验证工具存储的模型规则执行痕迹为比特币新区块提交消息接收动作RecvBlockSubmission所涉及的消息项block.

3.4　对交易收集时间间隔及算力分布的处理

在实际的比特币区块链共识过程中,上一区块共识确认时间与已授权交易公布时间的间隔长短会影响新挖掘比特币区块的接受率. 然而,为了实现对本文符号模型的自动化形式化验证,本文需要规避对时间变量以及概率的讨论. 因此,本文规则common_task_part将待记录比特币交易集合建模为CommonTaskPart消息的元素,并且不在规则中对待记录交易集合的大小进行检查,以覆盖在区块挖矿任务分配过程中所涉及的可能的交易集合情形. 在比特币算力均匀分布、所有被提交新区块的支持算力大小皆相同的情况下,本文建模的用于更新当前区块链信息的模型规则recv_block_submission被定义为只记录第一个被提交的区块所含交易集合,该规则将对交易集合进行检查.

本文在考虑不均匀分布的比特币算力时,拥有最大比特币支持算力的合法区块才能通过共识模块规则的检查,模型将每个历史区块的支持算力作为比特币共识节点的实体状态元素存储起来,并且拥有更大算力的区块的提交将触发模型规则对当前区块链合法区块的替换. 在算力分布不均匀时,规则recv_block_submission用于更新当前区块链最尾部区块信息,定义了将历史最尾部区块更新为当前拥有更大支持算力的新提交合法区块的协议操作. 与此同时,矿池服务器状态存储了当前已被分配的合作挖矿任务数,由于本文假设一个单位的比特币算力可以求解一个已分配的合作挖矿任务,因此已被接受的任务求解数量实际上即为矿池服务器分配给已挖掘区块的算力大小.

4　实验与分析

4.1　SmartVerif

本文使用SmartVerif[14]作为比特币底层协议形式化模型通用全自动化验证实验的工具,SmartVerif是一种无需专家干预的形式化验证系统,可完成复杂协议的自动化形式化安全性分析. 由于复杂协议的状态空间很大或者无限大,SmartVerif可用其内部的动态策略来智能地搜索证明路径,其动态策略设计简单而灵活,可根据模型自动进行优化,而无需任何人工干预. SmartVerif已经通过充分的案例分析验证了其动态策略的有效性,并且已成功对多个目前最具挑战的安全协议进行了全自动证明. 安全研究人员在使用SmartVerif进行自动化形式化验证时,需要输入已被合理建模的形式化模型,SmartVerif会对模型所需验证的安全引理进行自动化证明,最终输出模型引理的验证结果及导致引理被破坏的攻击路径.

4.2　安全属性验证结果

本文完成了对比特币形式化模型的自动化验证实验,不仅验证了独立执行下的比特币协议支付模块PP(Payment Protocol)、共识模块CP(Consensus Protocol)、合作挖矿模块SP(Stratum Protocol)以及跨账本模块IP(Interledger Protocol)的安全性,还验证了比特币协议支付模块与共识模块(PP&CP)、共识模块与合作挖矿模块(CP&SP)以及支付模块与跨账本模块(PP&IP)组合执行安全性,本节具体验证结果见表1.

表1　比特币协议的安全属性验证结果

	可认证性	隐私性	可问责性	金额一致性
PP	PM:√	×	√	√
CP	NF: −	-	-	×
SP	WF:×	-	×	-
IP	PN: −	-	√	-
PP&CP	PM:× NF: −	×	-	×
CP&SP	NF: − WF:×	-	×	×
PP&IP	PM:√ PN: −	×	×	-

注:可认证性表明各协议模块交互实体间是否满足比特币单射一致性级别认证属性(Injective Agreement),其中:P为买家实体、M为商家实体、N为节点实体、F为矿池服务器实体、W为矿池矿工实体,PM表示协议是否保证买家实体P与商家实体M是否满足单设一致性,依此类推;√表示该列属性可被满足,×表示该列属性不能被满足,-表示未考虑或不相关安全属性.

4.3　算力盗取攻击

本文借助形式化自动化验证系统SmartVerif发现了可在实际世界复现的比特币底层协议算力盗取攻击. 该攻击可以使得挖矿任务求解奖励会被错误地支付,矿池服务器的诚实性会被错判. 诚实的矿池服务器将为未贡献算力进行合作挖矿任务求解的比特币攻击者支付薪酬,且真正贡献算力的诚实矿池矿工也无法被支付应得薪酬. 算力盗取攻击破坏了涉及共识协议模块执行的矿池服务器与矿池矿工间单射一致性级别

认证性以及比特币认证属性.

4.3.1 原始协议

想要加入矿池的矿工旨在通过自己的算力对满足合作挖矿任务难度的哈希值进行暴力查找. 矿池矿工发送包含矿池矿工身份信息的Subscription消息给矿池服务器. 此后,矿池将生成作为会话标识的随机数以及由S_{id}生成特殊部分任务、设置难度,并且将会话标识与矿工身份信息进行绑定. 矿池发送UniqueTaskPart给矿池矿工. 当区块链更新后,矿池服务器设置合作挖矿公共部分任务的消息元素,向矿池矿工发布CommonTaskPart消息. 在矿池矿工接收到合作挖矿特殊部分任务以及公共部分任务后,矿池矿工可组合两个消息生成待解决的完整合作挖矿任务. 矿池矿工借助算力暴力地找到属于自己的合作挖矿任务求解,将SolutionSubmission消息发送给矿池. 矿池接收到任务求解后,比较求解对应的区块哈希值与当前任务信息、当前区块求解与历史求解,检查区块求解的合法性和唯一性,宣称将支付对应矿工指定金额求解奖励.

4.3.2 模型验证结果分析

本文借助SmartVerif完成对模型的自动化形式化验证,通过分析验证结果发现了可在比特币计算网络中实施的攻击,本文称之算力盗取攻击. SmartVerif结果输出的攻击路径含义为:矿池服务器F通过动作Claim_sol向矿池协议恶意矿池矿工W提交了被窃听任务应得的矿工任务求解薪酬,而协议执行中并不存在对应的任务求解提交动作痕迹Claim_subsol,并且受害矿池矿工即使贡献了算力并提交了正确求解,也没有求解的确认动作痕迹,因此也无法获得被窃听任务应得的矿工任务求解薪酬. 验证结果表明该协议执行可破坏矿池协议模型中矿池服务器与矿池矿工之间的单射一致性级别认证属性,并且即使矿池服务器是诚实的,即协议执行在不存在恶意矿池服务器的情形下,存在诚实矿池矿工无法获得薪酬奖励的情况,因此诚实的矿池服务器会被判别为恶意矿池服务器,危害可追溯性.

4.3.3 不安全的协议执行

下面本文将描述基于自动化验证结果所发现不安全协议执行,为便于读者理解算力盗取攻击过程,本文通过图4展示了基于SmartVerif验证结果所发现的算力盗取攻击. 如果一个攻击者正在监听矿池服务器与矿池矿工之间的信道,窃听到Subscription消息,攻击者篡改Subscription消息中的元素W为自己的身份信息A. 随后,矿池生成与攻击者的会话标识S_{id},并且基于绑定A的S_{id}生成合作挖矿特殊部分任务. 在从矿池中获得包含A和S_{id}的UniqueTaskPart消息后,比特币攻击者将冒充诚实矿池服务器并且发送恶意的UniqueTaskPart消息给矿池矿工,该消息包含W和S_{id}. 受害者矿池矿工错误地花费算力计算基于S_{id}生成攻击者身份对应的特殊部分任务.

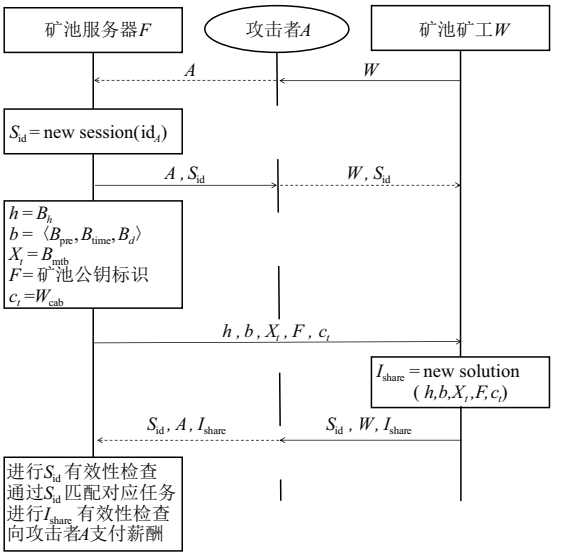

图4 算力盗取攻击

诚实矿池服务器随后发送包含消息项(h, b, X_t, F)以及被分配任务数c_t的CommonTaskPart消息,其中消息中包含的挖矿任务难度被设置为单位难度. 矿池矿工在收到合作挖矿公共部分任务消息CommonTaskPart以及特殊部分任务消息UniqueTaskPart后,根据本地状态存储的S_{id}以及(h, b, X_t, P_F)计算合作挖矿任务的求解I_{share}. 受害者矿池矿工在计算出求解I_{share}后,向诚实矿池发送包含消息项S_{id}、W以及I_{share}的SolutionSubmission消息. 同时,攻击者将SolutionSubmission消息的W消息项替换为自己的身份标识A,并将修改后的SolutionSubmission消息发送给诚实矿池. 诚实矿池矿工在收到SolutionSubmission消息后,将按照协议规则基于S_{id}以及A对被提交的任务求解进行检查. 诚实矿工通过S_{id}检索到本地状态存储的完整的任务信息,通过计算收到的任务求解I_{share}. 正如攻击者所期望的,诚实矿池向标识为A的攻击者支付了任务求解I_{share}所需的薪酬,而真正付出算力计算出sol_t的矿池矿工却无法获得应得任务求解薪酬.

4.3.4 被破坏的安全属性

在上述的不安全比特币协议执行中,矿池服务器通过动作Claim_sol(A, F, I_{share}),向提交了被窃听任务求解I_{share}身份标识为id_A的协议攻击者支付计算该任务应得的任务求解薪酬,而协议执行中并不存在对应求解确认动作痕迹Claim_sol(A, F, I_{share})的任务求解提交动作痕迹Claim_subsol(A, F, I_{share}),而身份标识为W矿池矿工即使贡献了算力并提交了正确求解,也没有任务求解提交动作痕迹Claim_subsol(W, F, I_{share})所对应的求解确认动作痕迹Claim_sol(W, F, I_{share}). 从上述描述可以看出,算力盗取攻击破坏了涉及合作挖矿协议模块执行的矿池服务器与矿池矿工之间的单射一致性

级别的认证属性. 与此同时,即使矿池服务器是诚实的、协议执行在不存在矿池服务器被侵犯动作痕迹Corrupt(F)的情形下,仍在存在诚实矿池矿工无法获得应得的任务求解薪酬奖励的情况,因此诚实的矿池服务器会被判别为恶意矿池服务器,危害可追溯性.

4.4 其他攻击

本文通过形式化验证实验验证了现有形式化分析所讨论的比特币双重支付攻击和比特币隐私攻击,并且本文实验结果所显示的比特币双重支付攻击路径与现有工作所展示的比特币攻击逻辑是一致的,该攻击破坏了与共识协议执行的金额一致性属性以及比特币支付模块与共识模块在组合执行过程中的买家与商家之间单射一致性级别的协议认证属性. 本文也发现了现有工作研究的隐私攻击,该攻击破坏了本文利用观察等价进行验证的隐私性,攻击者只需要在支付协议模块执行过程中,简单地获取商户在初始化过程中所需对外发送的消息(包括商户标识消息项和地址消息项),并将地址项与已发布交易地址项比较,即得知身份相关性.

此外,本文工作发现了一个不安全场景:被侵蚀的具备多重身份的共识节点可以成功发起支付条件重放攻击. 支付条件重放攻击破坏了基于哈希锁定技术的跨账本交易模块与比特币支付协议模块组合执行下的比特币可问责性. 在这种攻击下,如果一个买家在短时间内重复地向跨账本商家购买某商品,被侵蚀的跨链节点则可对历史跨账本交易条件进行重放,通过提供历史原象,直接触发交易执行,不需将跨账本资产转移到商家,而转移到自己账户. 此时,诚实的商家会因为未收到跨账本交易资产而拒绝确认买家对订单的支付.

5 结论

本文多维度地设计了比特币安全属性的符号化定义,细粒度而系统地形式化建模了四种被广泛部署的比特币协议及其五类协议实体的交互过程,并且借助自动化形式化验证系统SmartVerif实现了对比特币安全性全面的形式化分析. 本文的未来工作将对其他加密数字货币进行全面形式化分析以及安全性的比较. 此外,还将进一步对各区块链协议的实体功能及其安全需求进行细化,基于符号方法完成对主流区块链应用安全属性的形式化分析,并为公众提供全面可靠的形式化安全验证报告.

参考文献

[1] Nakamoto S. Bitcoin: A peer-to-peer electronic cash system[EB/OL]. https://mronline.org/wp-content/uploads/2018/06/bitcoin.pdf,2020-2-13.

[2] Tschorsch F, Scheuermann B. Bitcoin and beyond: A technical survey on decentralized digital currencies[J]. IEEE Communications Surveys & Tutorials, 2016, 18(3): 2084 – 2123.

[3] 秦超霞, 郭兵, 沈艳, 等. 区块链的安全风险评估模型[J]. 电子学报, 2021, 49(1): 117 – 124.
Qin C X, Guo B, Shen Y, et al. Security risk assessment model of blockchain[J]. Acta Electronica Sinica, 2021, 49 (1): 117 – 124.(in Chinese)

[4] 陈露, 相峰, 孙知信. 基于属性密码体制的区块链安全技术研究进展[J]. 电子学报, 2021, 49(1): 192 – 200.
Chen L, Xiang F, Sun Z X. A survey of blockchain security technologies based on attribute-based cryptography[J]. Acta Electronica Sinica, 2021, 49(1): 192 – 200.(in Chinese)

[5] Conti M, Sandeep Kumar E, Lal C, et al. A survey on security and privacy issues of bitcoin[J]. IEEE Communications Surveys & Tutorials, 2018, 20(4): 3416 – 3452.

[6] Bastiaan M. Preventing the 51%-attack: a stochastic analysis of two phase proof of work in bitcoin[EB/OL].http://referaat.cs. utwente. nl/conference/22/paper/7473/preventing-the-51-attack-a-stochasticanalysis-of-two-phase-proof-of-work in bitcoin.pdf, 2020-2-14.

[7] Miller A, LaViola Jr J J. Anonymous byzantine consensus from moderately-hard puzzles: A model for bitcoin[EB/OL]. https://socrates1024. s3. amazonaws. com/consensus. pdf, 2020-2-13.

[8] Eyal I, Sirer E G. Majority is not enough: Bitcoin mining is vulnerable[A]. International Conference on Financial Cryptography and Data Security[C]. Berlin, Heidelberg: Springer, 2014. 436 – 454.

[9] Blanchet B. Security protocol verification: Symbolic and computational models[A]. International Conference on Principles of Security and Trust[C]. Tallinn, Estonia: Springer, 2012. 3 – 29.

[10] Garay J, Kiayias A, Leonardos N. The bitcoin backbone protocol: Analysis and applications[A]. Annual International Conference on the Theory and Applications of Cryptographic Techniques[C]. Sofia, Bulgaria: Springer, 2015. 281 – 310.

[11] Garay J, Kiayias A, Leonardos N. The bitcoin backbone protocol with chains of variable difficulty[A]. Annual International Cryptology Conference[C]. Cham: Springer, 2017. 291 – 323.

[12] Kiayias A, Panagiotakos G. Speed-security tradeoffs in blockchain protocols[EB/OL]. https://eprint.iacr.org/2015/1019.pdf,2016.

[13] Das P, Faust S, Loss J. A formal treatment of determinis-

tic wallets[A]. Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security [C]. London, United Kingdom: ACM, 2019. 651 – 668.

[14] Cremers C, Horvat M, Hoyland J, et al. A comprehensive symbolic analysis of TLS 1.3[A]. Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security[C]. Dallas, Texas, USA: ACM, 2017. 1773 – 1788.

[15] Basin D, Dreier J, Hirschi L, et al. A formal analysis of 5G authentication[A].Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security[A]. Toronto, Canada: ACM, 2018. 1383 – 1396.

[16] Xiong Y, Su C, Huang W, et al. Smartverif: Push the limit of automation capability of verifying security protocols by dynamic strategies[A]. 29th {USENIX} Security Symposium ({USENIX} Security 20) [C]. USA: {USENIX} Association, 2020.253 – 270.

[17] Lowe G. A hierarchy of authentication specifications[A]. Proceedings 10th Computer Security Foundations Workshop[C]. Rockport, MA, USA: IEEE, 1997. 31 – 43.

[18] üsters R K, Truderung T, Vogt A. Accountability: definition and relation-ship to verifiability [A]. Proceedings of the 17th ACM conference on Computer and Communications Security[C]. Chicago, Illinois, USA: ACM, 2010. 526 – 535.

[19] Dreier J, Kassem A, Lafourcade P. Formal analysis of e-cash protocols[A]. 2015 12th International Joint Conference on e-Business and Telecommunications (ICETE) (Volum:04)[C]. Colmar, France: IEEE, 2015. 65 – 75.

作者简介

包象琳 女,1994年12月生于安徽芜湖,安徽工程大学计算机与信息学院教师.研究方向包括:区块链,形式化方法,网络与信息安全等.
E-mail:baoxianglin@ahpu.edu.cn

汪万森 男,1995年9月生于安徽合肥,中国科学技术大学计算机科学与技术学院在读博士生.研究方向包括:协议形式化验证,区块链.
E-mail:wangws@mail.ustc.edu.cn

熊 焰 男,1960年8月生于安徽合肥,中国科学技术大学计算机科学与技术学院教授、博士生导师.研究方向包括:计算机网络与信息安全,移动计算与移动网络,分布式处理等.
E-mail:yxiong@ustc.edu.cn

孟昭逸 男,1992年5月生于安徽合肥,中国科学技术大学计算机科学与技术学院博士后研究员.研究方向包括:安卓安全,软件形式化验证等.
E-mail:mzy516@ustc.edu.cn

黄文超(通讯作者) 男,1982年6月生于湖北宜昌,中国科学技术大学计算机科学与技术学院副教授.研究方向包括:信息安全,人工智能,移动计算,网络与系统安全自动化验证技术,Android系统安全等.
E-mail:hangwc@ustc.edu.cn

徐晓峰 男,1992年12月生于安徽芜湖,安徽工程大学计算机与信息学院讲师.研究方向包括:零样本学习,深度学习,机器学习,信息安全等.
E-mail:xuxiaofeng@ahpu.edu.cn

陈凯杰 男,1997年6月出生于安徽合肥,中国科技大学计算机科学与技术学院在读硕士研究生.研究方向包括:区块链,形式化方法.
E-mail:ckjkevin@mail.ustc.edu.cn

方贤进 男,1970年11月生于安徽舒城,安徽理工大学计算机科学与工程学院教授、博士生导师.研究方向包括:信息安全,数据挖掘等.
E-mail:xjfang@aust.edu.cn