

基于智能合约的高对抗性僵尸网络研究

赵昊, 舒辉, 康绯, 邢颖

(数学工程与先进计算国家重点实验室, 河南 郑州 450001)

摘要: 区块链技术的发展与应用使构建更为鲁棒和灵活的僵尸网络命令控制信道成为可能。为了更好地研究这类潜在的新型僵尸网络威胁, 提出基于区块链智能合约的高对抗性僵尸网络模型——SCBot。SCBot 模型采用分层混合拓扑结构, 在僵尸子网层构建基于智能合约的命令传递信道, 并建立可信度评估机制判别节点真实性, 从流量和终端两大层面提升网络的对抗性。模拟构建小型的僵尸网络集群, 对 SCBot 的命令传递效率和鲁棒性进行了对比实验, 并从经济成本角度分析其在现实环境中的可行性。最后对该类型僵尸网络的防御策略做简要分析和讨论。

关键词: 网络安全; 僵尸网络; 命令控制; 区块链; 智能合约

中图分类号: TP309

文献标识码: A

doi: 10.11959/j.issn.2096-109x.2021070

High resistance botnet based on smart contract

ZHAO Hao, SHU Hui, KANG Fei, XING Ying

State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou 450001, China

Abstract: The development and application of blockchain technology makes it possible to build a more robust and flexible botnet command and control channel. In order to better study this type of potential new botnet threats, a highly confrontational botnet model based on blockchain smart contracts-SCBot was proposed. The SCBot model adopts a hierarchical hybrid topology structure, builds a command transmission channel based on smart contracts at the zombie subnet layer, and establishes a credibility evaluation mechanism to determine the authenticity of nodes, and enhances the confrontation of the network from the two major levels of traffic and terminals. The construction of small botnet clusters were simulated, comparative experiments on SCBot's command transmission efficiency and robustness were conducted, and its feasibility in the real environment from the perspective of economic costs was analyzed. Finally, a brief analysis and discussion of the defense strategy of this type of botnet were given.

Keywords: cyber security, botnet, command and control, blockchain, smart contract

收稿日期: 2021-01-11; 修回日期: 2021-05-17

通信作者: 舒辉, shuhui123@126.com

基金项目: 国家重点研发计划 (2019QY1305)

Foundation Item: The Key R & D Program of China (2019QY1305)

论文引用格式: 赵昊, 舒辉, 康绯, 等. 基于智能合约的高对抗性僵尸网络研究[J]. 网络与信息安全学报, 2021, 7(4): 30-41.

ZHAO H, SHU H, KANG F, et al. High resistance botnet based on smart contract[J]. Chinese Journal of Network and Information Security, 2021, 7(4): 30-41.

1 引言

僵尸网络被视为网络空间安全的重大威胁之一^[1], 用于发动 DDoS、垃圾邮件、点击欺诈和敏感信息窃取等大规模网络攻击活动。僵尸网络本质上可看作发动大规模网络攻击的基础设施平台, 由数以万计的受控计算机或物联网设备组成。僵尸控制者通过命令控制 (C&C, command and control) 信道, 实现对僵尸成员节点的一对多远程管理。实际上, C&C 机制是僵尸网络的核心所在, 唯一决定一种僵尸网络。从最传统的中心结构到分布式的 P2P 结构, 僵尸网信道的拓扑架构不断演变。虽然出现了 Domain-Fluxing、Fast-Fluxing 和 Tor 网络等方法用于保护 C&C 服务器, 但中心化的本质避免不了“单点失效”的固有缺陷。而更为复杂的 P2P 网络结构, 存在难以部署、易受监视和索引污染等问题。从协议层面看, 所有的应用层协议 (如 IRC、HTTP、DNS) 或者自定义 P2P 协议等可作为僵尸网络通信协议。近年来, 僵尸网络设计者尝试使用如云平台^[2]和社交网络^[3]等公共服务来提升 C&C 信道的弹性和鲁棒性。

比特币^[4]作为区块链技术的首个大规模应用, 取得了空前的成功, 成为众所周知的数字加密货币。区块链技术展现了匿名性、不可篡改性 and 去中心化等优势特性, 不仅在数字货币领域广泛应用, 而且引起了程序开发人员的关注, 将其作为安全解决方案应用于软件^[5]、汽车^[6]和化工行业^[7]等领域。然而, 区块链技术本身吸引了网络攻击者的注意。其中最明显的例子是区块链使攻击者能够在现有公共区块链的基础上建立更具弹性和可用性的僵尸网络^[8]。随着区块链技术的普及, 网络中与之相关的流量变得越来越普遍, 而区块链的底层网络则提供了分布式 P2P 通道, 僵尸网络可直接基于此构建命令控制信道而无须自定义 P2P 协议。因此, 基于区块链的僵尸网络 C&C 机制不同于传统僵尸网络拓扑结构或通信协议, 具有高度匿名、流量隐蔽和强认证性等特点, 难以被检测、发现和接管, 是潜在的新型网络安全威胁。在国内尚缺乏针对此类僵尸网络的系统性研究的背景下, 开展这方面的研究工作,

具有十分重要的理论价值和现实意义。

本文的主要贡献如下: ① 综述现有的基于区块链构建僵尸网络命令控制信道的相关研究, 总结现有方案的原理和实现过程, 分析这类方案存在的不足和局限性; ② 简要阐述区块链智能合约的技术原理, 从理论上说明该技术用于僵尸网络命令控制机制的优势特性; ③ 提出基于区块链智能合约的僵尸网络信道模型 SCBot, 该模型采用分层混合 P2P 结构, 构建基于智能合约进行命令传递的僵尸子网层, 并加入节点可信度评估机制提升对抗性; ④ 模拟构建小型的僵尸网络集群, 从命令传递效率和鲁棒性等方面对 SCBot 做对比试验, 并从经济成本角度分析其可行性; ⑤ 对该类型僵尸网络的防御策略做简要分析和讨论。

2 相关工作

区块链技术的出现引起了僵尸网络设计者的注意, 使建立更加鲁棒和更具弹性的僵尸网络命令控制机制成为可能, 这促使学术界对这一问题开展研究, 以应对可能到来的新型僵尸网络安全威胁。Ali 等^[9]在 2015 年提出的 ZombieCoin 中讨论了使用公共区块链构建僵尸网络 C&C 信道的可能性, 文中列举了在比特币区块链中存储和隐藏僵尸网络命令的 4 种方法: 一是利用交易中的 OP_RETURN 字段, 该字段被允许写入任意数据, 长度限制为 80 B; 二是利用 unspent output, 由数据本身直接生成实际不存在的账户地址, 并向其发送少量比特币, 最多可保存 20 B 的数据; 三是利用比特币中的椭圆曲线数字签名算法 (ECDSA), 通过选择相同的随机因子派生私钥来传播最多 32 B 的数据, 但每次都需要发起两笔交易来实现; 四是利用签名算法的随机性, 通过重复创建签名, 使前 n 位与预编码命令字符串一致, 这种方法与正常交易无异, 隐蔽性更好。Frkat 等^[10]深入研究了 ZombieCoin 中提及的利用比特币数字签名嵌入隐藏信息的思想, 提出了 ChainChannels 方案, 用更隐蔽和先进的方法隐藏僵尸网络 C&C 活动。此外, Curran 和 Geist^[11]提出利用区块链的 OP_RETURN 域作为中心化僵尸网络的备份机制——Floating C&C Server, 在原 C&C 服务器不可用的情况下, 在比特币区块链上

发布备用 C&C 服务器的地址信息。

利用比特币作为僵尸网络 C&C 机制的方案受限于字段长度,不能够完全适应 C&C 信道的需求。除了比特币,在以太坊平台上利用智能合约构建僵尸网络 C&C 通道也是一种思路。与比特币交易不同,智能合约具有图灵完备性,可承载较大的信息容量,提供更丰富的应用场景,允许程序功能直接部署到区块链上。Majid 等^[12]讨论了基于以太坊智能合约构建僵尸网络 C&C 的可能性,提出了 Bottract 这一概念,具体说明了如何编写智能合约实现僵尸网络命令的发布与获取。Zohar^[13]提出把僵尸控制者和僵尸节点之间的所有通信都通过以太坊智能合约来完成,并完成了概念验证——Unblockable Chains。然而,这两种基于智能合约的方案仅停留在理论验证阶段,并没有结合僵尸网络实际特性进行测试或仿真,智能合约作为真实僵尸网络 C&C 机制的可行性和实用性有待进一步论证和研究。

3 基本概念

3.1 区块链

区块链是一种去中心化的分布式账本技术,记录了 transaction(交易)数据的 block(区块)按照时间顺序前后连接组成区块链^[14]。在区块链的 P2P 网络中,每个 full node(全节点)都在本地维护完整的区块链数据副本。区块链综合运用了包括数据库、分布式系统和密码算法等成熟技术,确保 transaction 的安全性和不可篡改性。

区块链系统的运行主要由以下几个步骤组成。

(1)发起 transaction,然后取 hash 值,用 private key(私钥)签名后提交发送至区块链网络。

(2)由网络中的其他 node 共同验证和审核 transaction 的合法性。

(3)获得记账权的 node 打包 transaction 并发布新 block。

(4)新的 block 被广播到 P2P 网络,通过共识算法同步到每一个 node。

3.2 形式化描述

基于区块链的僵尸网络命令控制信道可以形

式化表示为

$$\text{Blockchain.C\&C} = (\text{Botmaster}, \text{Bot}, D, S) \quad (1)$$

其中, Botmaster 是僵尸网络的控制者,即信息的发布者; Bot 表示受控僵尸主机上的僵尸程序集合,即信息的接收方,在区块链网络中可看作客户端节点; D 表示传递的信息数据; S 表示信息的传输过程,在区块链网络中也可视为节点同步过程。

假设基于区块链的僵尸网络中共有 x 个僵尸节点,并且考虑到僵尸网络控制者 Botmaster 需要构造并发送交易,因此可视为区块链节点,则 Botmaster 和 Bot 可分别表示为

$$\text{Botmaster} = (n_0) \quad (2)$$

$$\text{Bot} = (n_1, n_2, n_3, \dots, n_t, \dots, n_x) (1 \leq t \leq x) \quad (3)$$

命令控制信道的目标是把 Botmaster 发布的指令信息数据 D 通过传输过程 S 发送至各个 Bot 节点,如式(4)所示。

$$\text{Botmaster} \xrightarrow{D} \text{Bot} = S[(n_0) \xrightarrow{D} (n_1, n_2, n_3, \dots, n_t, \dots, n_x)] (1 \leq t \leq x) \quad (4)$$

当 Botmaster 将发布的指令信息以交易的形式发送至区块链网络后,假设节点 n_t 最先接收到该信息,那么之后该信息在区块链底层 P2P 网络中传播的对等通信过程 S 可表示为

$$S = \left\{ \left(n_0 \xrightarrow{D} n_t \right) \Rightarrow \left[(n_t \xrightarrow{D} n_i, i \neq t) \Rightarrow (n_i \xrightarrow{D} n_j, j \neq i) \right] \right\} \quad (5)$$

3.3 智能合约

智能合约是区块链技术最重要的应用领域之一。从技术角度来说,智能合约可看作可自我执行和自我验证的计算机程序^[15]。在部署智能合约之前,与合约相关的所有条款的逻辑流程已经被制定。以太坊是用来设计和部署智能合约的重要平台,通过外部账户和合约账户的机制实现智能合约的部署和调用。某种程度上,智能合约可视为托管在区块链上自动运行的程序。开放的社区允许任意合约无须审查就可部署在区块链上,但也给恶意利用提供了可能性。智能合约中专用名词解释如表1所示。

表1 智能合约中专用名词解释
Table 1 Explanation of terms for smart contracts

专用名词	定义
account	一般账户，也称为外部账户
contract account	合约账户，唯一标识一个合约
address	特指账户地址的数据类型
int, uint, string, struct	常见数据类型
mapping	用来构建“键-值对”存储的数据类型
function	合约中实现具体功能的函数模块

3.4 优势特性

随着网络空间对抗的升级，传统的僵尸网络命令控制机制已不再满足攻击者的需求，构建更为鲁棒、隐蔽和灵活的僵尸网络命令控制信道成为趋势，而区块链的智能合约的普及与应用为此提供了新的可能性。本节总结基于区块链的智能合约用于僵尸网络命令控制机制的优势如下。

(1) 高可用性：区块链的分布式记账机制使每个节点维护和同步相同的数据，因此僵尸程序无论连接哪个节点都能调用合约获得 C&C 服务器信息或者发布的命令。只要智能合约所在的区块链存在，僵尸网络 C&C 机制就能一直运行。

(2) 易部署性：相比构建自定义的 P2P 协议，借助于强大的社区生态和开源的软件工具，直接在智能合约上部署僵尸网络 C&C 通道是更为高效和简便的，而且网络的组建更为快捷。

(3) 鲁棒性：区块链作为公共服务，不会受到权威组织审查，其流量不会被阻断和屏蔽。分布式特性使部署在区块链上的僵尸网络 C&C 机制能够免疫单点失效攻击。因此，基于智能合约的 C&C 机制具有良好的抗关停能力。

(4) 隐蔽性：由于区块链应用的普及，基于智能合约的僵尸网络 C&C 活动混杂在大量正常的交易流量中不会引起注意，僵尸节点隐藏在合法用户中获取命令，不易被识别，可以实现通信过程的完全加密和无数据泄露。

(5) 匿名性：区块链本身就具有匿名功能，可以任意生成账户地址，难以追溯命令信息的发布源头，无法确定僵尸网络控制者的真正身份。

(6) 强认证性：区块链的加密机制可以为僵尸网络 C&C 活动提供完整的认证功能，僵尸网

络控制者和僵尸节点间可建立双向认证机制，可抵御重放攻击和请求伪造，提供良好的防接管能力。

(7) 可编程和可扩展性：区块链是开放的公共平台，智能合约无须经过审查就可部署运行，相当于在区块链上运行托管程序，为僵尸网络 C&C 机制提供了良好的可编程和可扩展性。

4 SCBot 体系结构

4.1 命令控制信道

为了克服传统中心结构的单点失效缺陷，提高 P2P 结构下命令传递和节点控制的效率，实现更加鲁棒和隐蔽的命令控制信道，本文提出了一种分层混合结构的信道模型，该模型通过增加 C&C 服务器层，将僵尸网络划分为独立子网，缓解针对命令控制服务器的打击和关停等防御手段。同时，C&C 服务器之间和僵尸子网内部均采用 P2P 方式连接，有效提升僵尸网络的鲁棒性。

SCBot 的体系结构可从上至下分为 4 个层次，在图 1 中由三条虚线划分，各部分的具体组成如下。

(1) 僵尸网络控制中心：第一层是僵尸控制者所在的控制中心，只与 C&C 服务器进行交互。真实僵尸网络案例中，僵尸网络所有者为了增加经济利益，往往提供租赁服务。因此，SCBot 在该层加入账户控制模块，允许在管理员之外，添加具有普通使用权限的一般账户。

(2) 命令控制服务器层：第二层是 C&C 服务器部分，为了避免单一中心服务器的固有缺陷，采用部署多台 C&C 服务器的方案，C&C 服务器之间以 P2P 方式相连。事实上，每台 C&C 服务器均可以独立完成命令控制功能，指挥任一子网发起大规模协同攻击活动。SCBot 采取这样的备份冗余机制，提升了僵尸网络的鲁棒性。该层部署在匿名网络中，C&C 服务器之间通过 Tor 隐藏服务 (hidden service) 互相访问。

(3) 超级节点层：第三层是直接 C&C 服务器相连的僵尸主机，标以“S”表示这是超级僵尸节点。SCBot 僵尸网络通过特定的检测方法筛选高可信节点作为超级僵尸节点。作为僵尸子网的实际控制者，超级节点通过 Tor 网络访问 C&C 服务器获取命令，并负责将僵尸网络控制者的命

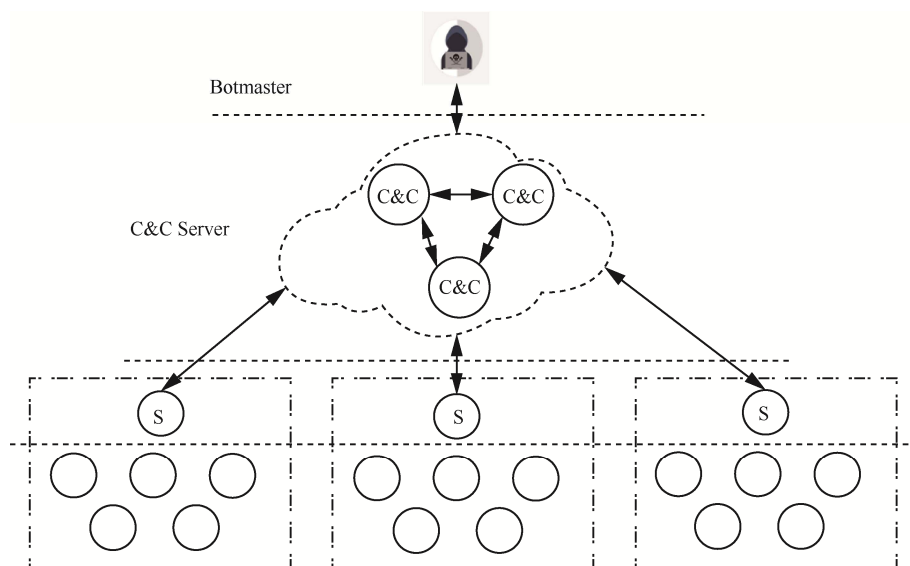


图 1 SCBot 体系结构
Figure 1 Architecture of SCBot

令下达至子网所有僵尸节点。

(4) 僵尸子网层：第四层由数量众多的普通僵尸主机组成，是攻击行动的具体执行层。为了提高僵尸网络行为的隐蔽性和抗检测能力，SCBot 运用区块链上的智能合约机制进行命令的发布与传递。

4.2 基于智能合约的命令传递方法

SCBot 僵尸网络子网层的组成结构如图 2 所示，各部分组成定义如表 2 所示。将图中的节点分为全节点（FN、FNB）和轻节点（LN、LNB）

两类，其中，FN 和 FNB 与区块链核心网络直接相联，而 LN 和 LNB 需要联接到全节点才能同步最新的数据。僵尸网络在扩展感染规模的过程中，较容易获得的是硬件资源有限的节点，因此适合运行轻节点，且通过联接 FN 和 FNB 可以同步获取最新命令数据。计算资源和带宽资源更丰富的节点数量较少，但由其作为 FNB 加入区块链核心网络，保证了 C&C 信道的稳定性和可靠性。另外，回传信道标注了两种方式：①回传信息与数据同样通过智能合约调用从区块链网络传回僵尸

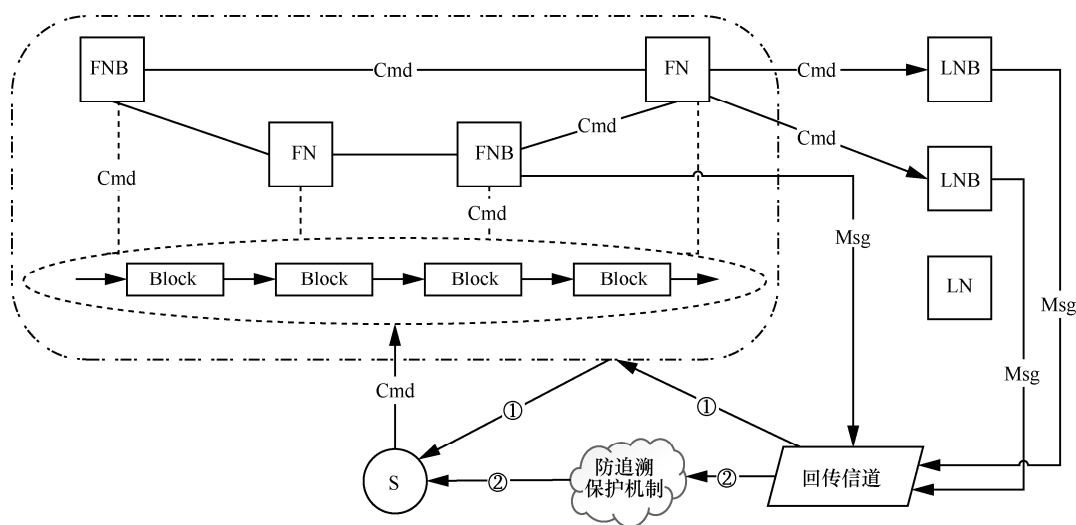


图 2 SCBot 僵尸网络子网层的组成结构
Figure 2 Subnet layer composition structure of SCBot

控制节点 S；②使用一定的防溯源反追踪机制建立额外的回传信道。显然，前者相比后者具有十分明显的优势，之所以把后者作为备选项列出，是因为使用区块链构建信道需要考虑经济成本。

表2 SCBot 僵尸子网组成定义
Table2 Subnet composition definition of SCBot

符号	含义
S	高可信的超级僵尸节点 S, 即僵尸子网的实际控制节点
Block	区块链中的区块, 前后链接组成完整的区块链 (图中椭圆虚线框内)
FN	维护完整区块链数据副本的节点, 可称为全节点 (full node)
FNB	被僵尸网络感染后成为全节点的僵尸节点, 即僵尸全节点 (full node bot)
LN	只维护完整区块链数据索引的节点, 可称为轻节点 (light node)
LNB	被僵尸网络感染后成为轻节点的僵尸节点, 即僵尸轻节点 (light node bot)
Cmd	由超级节点 S 转发的来自僵尸控制者的攻击命令或控制指令
Msg	各僵尸节点 (FNB 和 LNB) 向僵尸控制者上报的节点状态信息或攻击执行反馈
回传信道	由僵尸节点向僵尸控制节点 S 回传数据的通信信道

在僵尸子网层面, 为了增强隐蔽性和匿名性, SCBot 在区块链底层网络上利用智能合约机制进行命令的发布与传递。命令发布与传递的具体过程如图 3 所示。

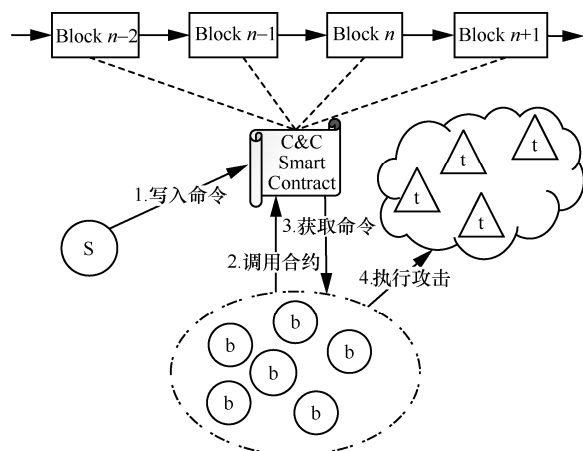


图3 基于智能合约的命令发布与传递

Figure 3 Command issuance transmission based on smart contract

(1) 高可信节点 S 作为超级僵尸节点, 负责创建 contract account (合约账户)。contract 创建

并发布至区块链后, 节点 S 发起交易, 调用 contract 中对应命令写入的 function (函数), 函数运行后首先验证调用者身份, 然后将接收到的命令作为参数写入 smart contract。

(2) 子网内各僵尸节点周期性访问智能合约, 发起交易, 轮询调用对应命令获取的 function。

(3) 僵尸节点 b 通过调用相应的 function 读取 contract 内当前的命令参数, 判断是否为新命令。

(4) 获取新命令后, 僵尸子网执行命令, 向目标 t 发起大规模协同攻击。

实现 SCBot 僵尸网络 C&C 信道的智能合约的组成结构如图 4 所示, 智能合约中涉及的变量和函数的说明见表 3。基于以太坊智能合约构建加密认证的 SCBot 僵尸子网命令控制信道的实现过程如下。

(1) 明确具体数据结构和基本的函数功能:

①自定义命令数据结构体, 内容包含攻击方式 (attack)、目标地址 (target)、持续时间 (duration) 等; ②定义僵尸网络控制者账户和合约账户地址; ③命令类型包含广播和单播两种。

(2) 结合智能合约的固有特性, 实现对僵尸网络控制者身份的认证, 并建立信道动态更新机制: ①合约一旦发布后, 编写者不具有特权或控制权, 这与僵尸网络的设计要求不符, 因此需要在构造函数中实现合约所有者认证; ②建立地址获取和合约更新机制。

(3) 这样的智能合约需要满足僵尸网络通信的基本需求, 即命令的发送和接收。①发送: 验证命令发布者的身份, 然后调用函数向合约变量中写入相应的命令, 单播模式下还需指定发送目标地址。②获取: 调用函数返回命令变量的字符串, 单播模式下通过当前调用账户地址检索对应命令。

基于区块链的智能合约作为僵尸网络通信协议的优势在于匿名、防篡改、强认证和稳健性, 这使下发指令到僵尸节点的过程十分便捷和高效, 但无论比特币或是以太坊等平台, 发起交易都需要支付成本。从僵尸控制者到僵尸节点的命令下发通道, 其交易成本与下发次数成正比, 虽然各个节点都需要同步数据, 但只需计算一次费用, 单次成本并不算多。如果各节点的回传数据

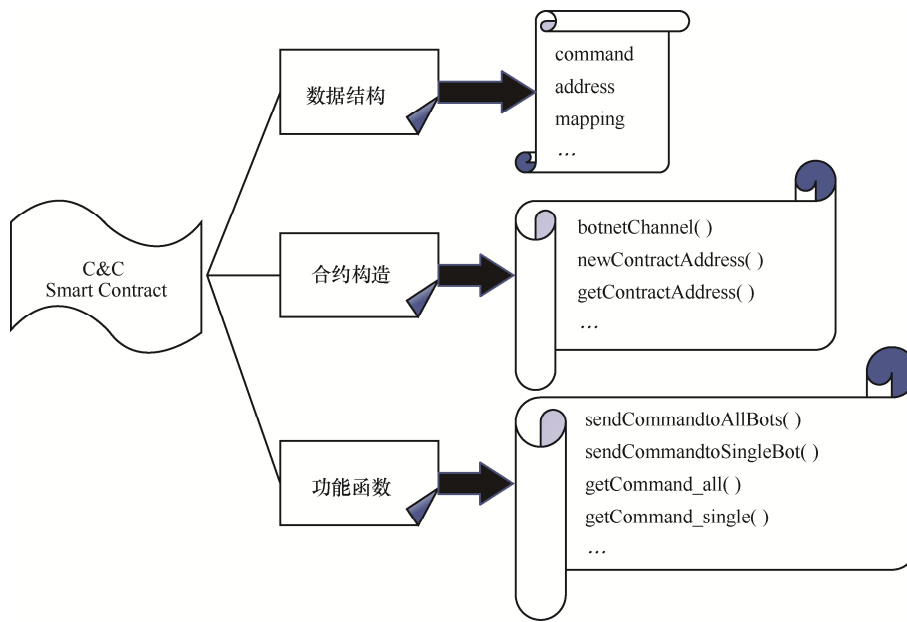


图 4 SCBot 智能合约组成结构

Figure 4 Smart contract composition structure of SCBot

表 3 SCBot 智能合约说明
Table 3 Description of SCBot smart contract

序号	变量或函数名	解释说明
1	command	传递具体命令的结构体变量，由 2,3,4 组成
2	attack	string 类型，表示攻击方式，如 syn flood、http get
3	target	string 类型，表示攻击目标的地址或域名
4	duration	string 类型，表示攻击活动的持续时间
5	botmaster	address 类型，表示僵尸网络控制者账户地址，即 S 节点的账户
6	contractAddress	address 类型，表示合约账户地址
7	toAllBots	command 结构体类型，表示广播给全体僵尸节点的命令
8	singleBot	command 结构体类型，表示单播给指定僵尸节点的命令
9	botnetChannel(address b)	合约构造函数，指定合约发起者
10	newContractAddress(address n)	合约地址更新函数
11	getContractAddress()	合约地址获取函数
12	sendCommandtoAllBots(string a, string t, string d)	广播命令发送函数，函数运行后会先验证调用者身份，确认命令真实性
13	sendCommandtoSingleBot(string a, string t, string d, address s)	单播命令发送函数，函数运行后会先验证调用者身份，确认命令真实性
14	getCommand_all()	广播命令获取函数
15	getCommand_single()	单播命令获取函数

(状态信息或执行反馈) 也通过区块链传输，所产生的交易成本与节点数量成正比，随着僵尸网络规模的扩大，回传信道需要支付的费用是巨大的。除非僵尸控制者只想发动一次性的大规模攻击活动而不考虑节点维护和信息反馈，否则对于僵尸网络攻击活动而言，回传信道是

不可或缺的。

4.3 节点可信度评估

对于僵尸网络控制者而言，来自防御方的威胁主要体现在两方面：一是流量分析和检测；二是样本捕获和分析。SCBot 利用智能合约进行命令的发布与传递，将僵尸网络的通信流量混杂在

正常的区块链网络中，能够免疫防御针对流量层面的对抗手段。任何暴露在互联网上的终端节点都是僵尸网络的潜在感染目标，这给防御方提供了捕获僵尸网络程序样本的机会，如通过部署蜜罐节点，从而进一步了解僵尸网络通信机理及潜在脆弱点。因此，准确判断终端身份，提高对蜜罐节点或虚拟环境的识别能力，是增强僵尸网络对抗性的重要方面。根据可信度评估机制，SCBot从可感染目标中剔除不可信节点，并选取高可信度节点作为各僵尸子网的实际控制节点S。

从默认配置 A_1 、系统指令 A_2 和异常行为 A_3 这3方面对节点进行评估，构建节点可信度评估矩阵 M ，如表4所示。显然，覆盖的检测指标 a 越多，非真实节点的可能性越大，然而，如网络时延、端口开放数量、调用时延等指标会存在误差，正常系统在某些情况下也可能表现出与蜜罐类似的特征。为了提升检测的准确性，本文引入加权参数 w_{ij} 以反映检测指标 a_{ij} 的重要程度。

表4 可信度评估矩阵 M
Table 4 Credibility Evaluation Matrix M

类别 A	指标 a	权重 w
A_1 (默认配置)	a_{11} : 检查 ping 程序的返回结果 (蜜罐中往往返回相同结果)	w_{11}
	a_{12} : 检查常见用户名口令是否能够登录	w_{12}
	a_{13} : 检查主机名是否正常	w_{13}
	a_{14} : 检查 Telnet、OpenSSH、Apache 等软件版本	w_{14}
		...
A_2 (系统指令)	a_{21} : 检查常见 Linux 命令行操作 (如 “cd ~/, \$?”) 的执行结果	w_{21}
	a_{22} : 检查循环结构指令的执行结果 (蜜罐中会报错)	w_{22}
	a_{23} : 检查带选项 (如 -help) 的命令行操作结果	w_{23}
		...
		...
A_3 (异常行为)	a_{31} : 检查网络时延	w_{31}
	a_{32} : 检查系统组件调用的时延	w_{32}
	a_{33} : 检查端口开放数量 (正常系统一般不超过5个)	w_{33}
	a_{34} : 检查重新连接前后创建的文件是否存在	w_{34}
	a_{35} : 检查监控软件标志	w_{35}
		...

$$P(t) = \sum_i A_i W_i = \sum_{i,j} a_{ij} w_{ij} \quad (6)$$

$$\sum_{i,j} w_{ij} = 1 \quad (7)$$

由此得出，针对任一待检测目标 t ，可信度 $P(t)$ 的计算方法如式(6)所示，且权重系数满足式(7)。计算得到可信度数值后，设定阈值 β 和 γ ，节点可信度评估结果由式(8)给出。

$$\begin{cases} 0 \leq P(t) < \beta, & t \text{ 为高可信节点} \\ \beta \leq P(t) < \gamma, & t \text{ 为普通可信节点} \\ \gamma \leq P(t), & t \text{ 为不可信节点} \end{cases} \quad (8)$$

5 实验与分析

5.1 C&C 信道实现

实现 SCBot 僵尸子网 C&C 信道的智能合约的部署与调用过程如图5所示。在本地运行 geth 节点搭建测试用私有链网络，版本号为 v1.9.16，操作系统环境为 Ubuntu20.04。按4.2节编写的以太坊 (Ethereum) 智能合约 (.sol 文件) 用 Solidity 编译器的 solc 命令即可编译得到合约的 EVM 二进制代码和 JSONABI，然后在 JavaScript 控制台中手动进行部署。更简便的方法是直接运行 Remix 在线编译器自动生成的合约部署代码。智能合约发布完成后，发布和获取命令的过程等同于合约调用，以单播模式为例，相关合约函数的调用格式如下。

```
BotnetChannel.sendCommandtoSingleBot.  
send Transaction(attack,target,duration,botAddress,  
{from: botmaster});
```

```
BotnetChannel.getCommand_single.call  
({from: web3.eth.account[singleBot]}))
```

5.2 命令传递效率分析

对于僵尸网络命令控制信道而言，命令下发和传达的效率是重要的评价指标之一，体现在命令传达耗时和命令传达率两方面。命令传达耗时指的是命令从发出到被节点接收的时间，而命令传达率指的是收到命令的僵尸节点数占所有僵尸节点数的比例。为检验 SCBot 僵尸子网 C&C 信道的命令传达效率，利用 VMware 虚拟机软件生成规模化的可控节点，在多台服务器上同时运行上百个虚拟机系统，模拟构建小型的僵尸网络集

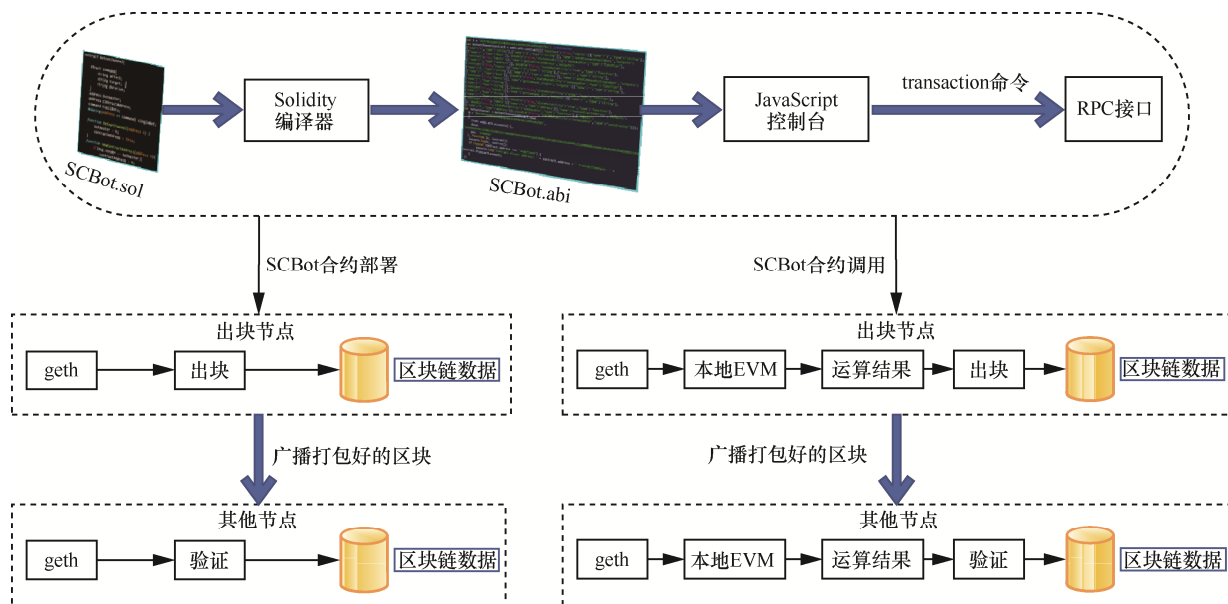


图 5 SCBot 智能合约部署与调用
Figure 5 Smart contract deployment and invocation of SCBot

群。作为对照,本文分别构建基于 IRC 和 Gnutella 的良性僵尸网络,前者是传统的中心结构,后者是典型的 P2P 结构。本文实验中的 SCBot 命令下发均采用广播模式,且僵尸节点均作为区块链中的全节点运行。同时,加入基于比特币 OP_RETURN^[16]的方案,作为同样基于区块链构建僵尸网络 C&C 信道的比较对象。需要指出的是,本文实验的目的是评估信道的命令传递效率,因此简化僵尸程序,只保留其接收命令的通信功能模块,避免产生危害真实互联网的恶意网络流量。

由于受到流量拥堵和路由选择等实际网络条件的影响,个别僵尸节点存在离线的概率,单个僵尸节点接收到命令的时间也可能存在时延。实验数据如图 6 所示,选取 120 s 内的数据进行分析。首先是基于 IRC 的中心结构网络,命令传达节点数与命令发出时间基本成正比,符合其以逐一单发模式进行命令下发的过程。而在基于 Gnutella 的 P2P 网络中,命令传达数近似以指数级增长,在初始阶段(0~30 s)较慢是因为需要以泛洪方式先查找邻居节点,随后各邻居节点均可进行转发,实现并发模式的命令下达,在 80 s 左右基本传达至所有在线节点。本文重点关注的基于区块链的命令传递方法,其底层网络本质上

仍然是 P2P 结构,数据曲线与 Gnutella 较为接近,同样具有“先慢后快”的特征,且区块链中需要执行共识算法以完成数据同步,因此同一时间段内命令传达节点数明显落后于 Gnutella。另外,在同样的时间段内,SCBot 中的命令传达节点数量少于比特币 OP_RETURN 的方法。这可以从原理的角度加以理解,在基于 OP_RETURN 的方法中,节点直接从同步后的交易数据中获取命令,而在 SCBot 中,节点需要调用合约并执行函数来获取命令,相对来说时延会高一些。总体来看,SCBot 的命令传递速率介于常见的中心结构和 P2P 结构的僵尸网络之间。

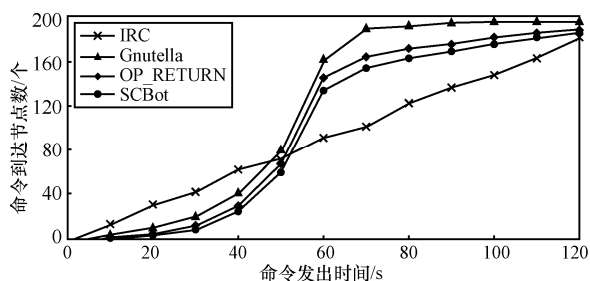


图 6 命令传达时间
Figure 6 Time of command transmission

对于发动大规模协同攻击活动而言,规定时间内的命令传达率是十分重要的指标。设定时间区段为命令发出后 60 s 内,以 Gnutella 和 OP_RETURN

作为对照,观察在不同节点规模下 SCBot 的命令传达率。实验结果如表 5 所示,可以看到,规定时间内的命令传达率随僵尸节点规模扩大而增加。同时,基于区块链的两种方法在相同规模下的命令传达率均低于典型 P2P 网络 Gnutella,但随着规模扩大,差距会逐渐缩小。由此得出结论,SCBot 在规定时间内(60 s)的命令传达率随僵尸节点规模扩大而不断增加,在实验条件下最高能接近 90%。

表 5 60 s 内命令传达率
Table 5 Rate of command transmission (60 s)

规模/台	传达率		
	Gnutella	OP_RETURN	SCBot
100	75.0%	56.0%	51.0%
200	81.5%	72.5%	67.0%
300	90.3%	89.7%	82.3%
400	96.4%	93.2%	89.5%

5.3 鲁棒性分析

僵尸网络命令控制信道的鲁棒性是十分重要的评估指标,可以理解为随机去除一定数量节点后,剩余僵尸节点的命令可达情况,即部分节点失去连接后,当前僵尸网络的连通性情况。因此,鲁棒性评估可理解为对僵尸网络抗毁能力的评估。实验节点规模为 400 台僵尸主机节点,依次随机关闭一定比例的节点,在 120 s 内观察剩余节点的可达率,并以 Gnutella 作为对照,实验数据如图 7 所示。

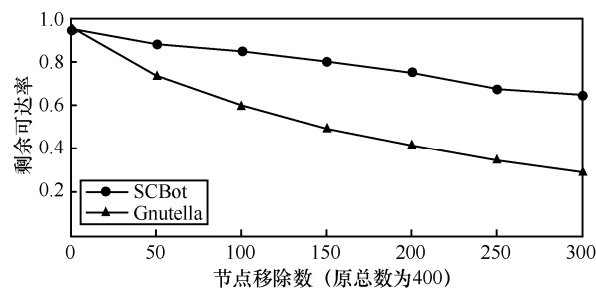


图 7 SCBot 鲁棒性评估
Figure 7 Robustness evaluation SCBot

结果表明,随机去除节点后,两种网络模型的连通性都受到了一定影响,随着节点数目的减少,剩余节点的可达率呈下降趋势。但在节点损失率相同的情况下,SCBot 的可达率相比基于原始 P2P 结构的 Gnutella 具有十分明显的优势。在

节点损失率将近 80%的情况下,SCBot 的可达率仍接近 70%。显然,底层的区块链网络为 SCBot 提供了良好的路由重构和网络恢复能力。

5.4 经济成本分析

式(9)给出了基于区块链的僵尸网络回传信道所需花费的计算公式,其中 Bot_number 表示僵尸节点个数, Fee 表示单次回传所需支付的费用,即发起单笔交易所需的交易费, Times 表示回传信息的次数,也可理解为频率。

$$\text{Spending} = \text{Bot_number} \cdot \text{Fee} \cdot \text{Times} \quad (9)$$

按照上述计算规则,表 6 列出了不同网络规模和通信频率下 SCBot 和比特币 OP_RETURN 方案^[16]的经济成本,其中比特币和以太坊的单价分别按 35 457.78 美元和 1046.6 美元计算(均为撰写本文时的市场实时均价)。从计算结果可以看到,即使是花费最小的方案,每日费用也超过 10 万美元。同时,由于原理机制的不同以及以太坊价格相对较低,SCBot 的经济成本明显小于比特币 OP_RETURN 的方案。文献[17]表明,僵尸网络的经济收益与僵尸网络规模正相关,僵尸网络控制者的日均收入不会超过 4 万美元,通常为数千美元。这些数据表明,黑客想要凭借自己的经济能力长期维护一个双向通信均基于区块链的僵尸网络是不可行的,除非获得了其他组织的资助。因此,在传统 C&C 的架构下,加入基于区块链的备份通道,即把区块链作为僵尸网络通信故障时的备用机制,可能是未来的发展方向之一。另外,为了方便开发者,如比特币或以太坊社区推出了对应的测试网络,在其中流通的都是测试币,不具有实际价值。在测试网上进行区块链僵尸网络 C&C 方案的概念验证是合适的,但如果将其用于实际部署,会存在规模较小、容易封堵和时延较大等缺陷,而且社区会不定时迭代更新测试网版本,这对于建立长期稳定运行的大规模僵尸网络是不利的。

6 防御策略分析

如果基于区块链智能合约的僵尸网络由于执行攻击活动或其他原因而暴露,采用何种措施将其移除或关停是需要重点讨论的问题。禁止网络内所有与区块链相关的流量有效但却不符合实

表 6 SCBot 和 OP_RETURN 方案的经济成本
Table 6 Economic cost of SCBot scheme and OP_RETURN scheme

单位: 万美元/日

频率	规模 (节点数)					
	5 万		10 万		20 万	
	OP_RETURN	SCBot	OP_RETURN	SCBot	OP_RETURN	SCBot
每 2 小时	96.36	13.90	192.75	27.83	385.50	55.63
每 1 小时	192.75	27.83	385.50	55.63	771.06	111.25
每 0.5 小时	385.50	55.63	771.06	111.25	1542.12	222.53

际。比特币等数字加密货币的流行使越来越多的普通用户安装并使用加密货币钱包,屏蔽所有区块链流量并不符合客观需求。Li 等^[18]认为常见的僵尸网络防御措施可总结为命令控制服务器打击、命令控制过程对抗和终端清除三大类,下面分别讨论这 3 种防御策略对基于智能合约的僵尸网络的影响和作用。

(1) 命令控制服务器打击。直接打击 C&C 服务器作为常见的僵尸网络对抗手段,具体方法为物理关闭和 DoS。但区块链底层网络中各个节点间地位对等,通过共识算法进行数据同步。基于区块链的僵尸网络命令控制信道中的通信通过底层 P2P 网络实现,不存在传统意义上的中心服务器,而且节点遍布全球互联网,难以进行有效的封堵和关停,因此命令控制服务器打击并不适用于对该僵尸网络的防御。

(2) 命令控制对抗。命令控制过程对抗主要利用僵尸网络通信中的脆弱点,如单点失效缺陷、身份认证缺失和域名硬编码等,从而影响和阻断僵尸网络的正常命令下发,甚至接管僵尸网络控制权。这方面的常见防御手段是黑名单机制。然而这对于基于区块链的僵尸网络却不可行^[19]。首先僵尸管理员可以使用数以百计的地址发布命令,因为在区块链中不需要花费代价就能够在本地申请到新的账户地址。从技术层面上讲,防御人员不可能做到封禁潜在的和未来的僵尸管理员地址。而且区块链的公钥认证机制能确保僵尸节点只受僵尸管理员的控制。另外,在基于区块链的数字加密货币中启用黑名单机制是违背其理念的,会引发社区的巨大动荡,甚至导致硬分叉的出现,如以太坊的 DAO 事件^[20]。

(3) 终端清除。终端清除是指清除已感染终

端的僵尸程序。防御人员可以通过挖掘并利用僵尸程序自身漏洞来实现清除^[21],或者联合大型厂商、安全公司和互联网服务提供商对感染终端进行定位,帮助用户进行清理^[18]。相对于前两种策略,这种手段对基于区块链的僵尸网络仍然是适用的。因此,在本地网络或设备终端上进行监视和检测,及时发现并定位异常活动和行为,是更为有效的策略。

7 结束语

僵尸网络攻击者始终致力于构建更为鲁棒和可用的 C&C 机制,为此不断尝试各种架构和协议。与此同时,研究人员通过预测模型,分析机理特性,试图提出应对新型僵尸网络威胁的防御解决方案。新兴技术如区块链的应用,催生了更高级的僵尸网络形态和命令控制机制,把攻防博弈推向了新阶段。本文提炼出一般模型,给出了基于区块链智能合约的僵尸网络 C&C 机制的原理及实现方法,并对其可行性和防御策略做了简单讨论。重视这一潜在的网络安全威胁,提出切实有效的应对措施,是目前亟须进一步开展的研究和工作。

参考文献:

- [1] BHATT P, THAKKER B. Isolating botnet attacks using bootstrap aggregating surflex-PSIM Classifier in IoT[J]. Journal of Intelligent & Fuzzy Systems, 2020, 38(2):1827-1840.
- [2] KEBANDE V R, VENTER H S. Obfuscating a cloud-based botnet towards digital forensic readiness[C]//10th International Conference on Cyber Warfare and Security (ICWS-2015). 2015.
- [3] YIN T, ZHANG Y, LI S. DR-SNBot: a social network-based botnet with strong destroy-resistance[C]//2014 9th IEEE International Conference on Networking, Architecture, and Storage (NAS). 2014.

- [4] UNDERWOOD S. Blockchain beyond bitcoin[J]. Communications of the ACM, 2016, 59(11): 15-17.
- [5] LEE B, LEE J H. Blockchain-based secure firmware update for embedded devices in an internet of things environment[J]. The Journal of Supercomputing, 2017, 73(3): 1152-1167.
- [6] DORRI A, STEGER M, KANHERE S S, et al. Blockchain: a distributed solution to automotive security and privacy[J]. IEEE Communications Magazine, 2017, 55(12): 119-125.
- [7] SIKORSKI J J, HAUGHTON J, KRAFT M. Blockchain technology in the chemical industry: machine-to-machine electricity market[J]. Applied Energy, 2017, 195: 234-246.
- [8] WANG W, MA X. Blockchain-based botnets for command-and-control Resilience[J]. Botnets: Architectures, Countermeasures, and Challenges, 2019: 217.
- [9] ALI S T, MCCORRY P, LEE H J, et al. ZombieCoin: powering next-generation botnets with bitcoin[C]//International Conference on Financial Cryptography and Data Security. 2015.
- [10] FRKAT D, ANNESSI R, ZSEBY T. ChainChannels: private botnet communication over public blockchains[C]//2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData). 2018.
- [11] CURRAN T, GEIST D. Using the bitcoin blockchain as a botnet resilience mechanism[EB].
- [12] MAJID N A I, MALAIKA A, IBRAHIM O A. Bottract: abusing smart contracts and blockchains for botnet command and control[EB].
- [13] ZOHAR O. Unblockable[EB].
- [14] SWAN M. Blockchain: blueprint for a new economy[M]. Blockchain: blueprint for a new economy. O'Reilly, 2015.
- [15] LUU L, CHU D H, OLICKEL H, et al. Making smart contracts smarter[C]//ACM SigSAC Conference. 2016: 254-269.
- [16] KAMENSKI D, SHAGHAGHI A, WARREN M, et al. Attacking with bitcoin: using bitcoin to build resilient botnet armies[M]. Chain: Springer. 2019.
- [17] U. S. leads multi-national action against “gameover zeus” botnet and “cryptolocker” ransom ware, charges botnet administrator[EB].
- [18] LI K, FANG B X, CUI X, et al. Study of botnets trends[J]. Journal of Computer Research and Development, 2016, 53(10): 2189-2206.
- [19] BCK L, ALEXOPOULOS N, SARACOGU E, et al. Assessing the threat of blockchain-based botnets[C]//2019 APWG Symposium on Electronic Crime Research (eCrime). 2020.
- [20] ATZEI N, BARTOLETTI M, CIMOLI T. A survey of attacks on ethereum smart contracts (SoK)[C]//International Conference on Principles of Security and Trust. 2017.
- [21] DITTRICH D. So you want to take over a botnet[C]// Proc of the 5th USENIX Workshop on Large-Scale Exploits and Emergent Threats. 2012.

[作者简介]



赵昊（1993—），男，浙江诸暨人，数学工程与先进计算国家重点实验室博士生，主要研究方向为信息安全、网络攻击行为分析。

舒辉（1974—），男，江苏盐城人，博士，数学工程与先进计算国家重点实验室教授、博士生导师，主要研究方向为网络安全、嵌入式系统分析与信息安全。

康绯（1972—），女，河南周口人，数学工程与先进计算国家重点实验室教授，主要研究方向为网络信息安全。

邢颖（1988—），女，河南周口人，数学工程与先进计算国家重点实验室博士生，主要研究方向为机器学习、安全检测。