基于免疫的区块链eclipse攻击的异常检测

吕婧淑[1] 杨 培[2] 陈 文[3] 操晓春[1] 李 涛[3]
(中国科学院信息工程研究所信息安全国家重点实验室 北京 100093)[1]
(32081部队 北京 100093)[2] (四川大学计算机学院 成都 610065)[3]

摘 要 区块链的eclipse攻击具有并发性、隐蔽性的特点,且往往依赖多节点协作完成垄断受害节点网络连接的攻击;相应地,计算机免疫系统具有分布式、自学习和自适应能力强的特点,能够良好地适应区块链多节点P2P分布式网络连接的环境。因此,为了检测区块链是否受到eclipse攻击,提出了一种基于免疫的区块链eclipse攻击的新型检测模型,并且建立了该模型的架构,给出了模型中各要素的形式定义及各模块的执行流程。根据模型进行了仿真实验,结果表明该模型具有较高的准确性和效率。

关键词 计算机免疫系统,区块链,P2P网络,eclipse攻击
中图法分类号 TP393 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2018.02.002

Abnormal Detection of Eclipse Attacks on Blockchain Based on Immunity

LV Jing-shu[1] YANG Pei[2] CHEN Wen[3] CAO Xiao-chun[1] LI Tao[3]
(State Key Laboratory of Information Security, Institute of Information Engineering, CAS, Beijing 100093, China)[1]
(32081 Army, Beijing 100093, China)[2]
(College of Computer Science, Sichuan University, Chengdu 610065, China)[3]

Abstract The eclipse attack against the blockchain has the characteristics of concurrency and concealment, and often relies on multi-node to collaboratively complete the attack of monopolizing victim's network connections. Correspondingly, the computer immune system has the characteristics of distribution, self-learning and strong adaptive ability. To detect whether the blockchain suffers from eclipse attacks, this paper proposed a new immunity based model to detect eclipse attacks on blockchain. At the same time, this paper established the architecture of the detection model, and presented the formal definitions of each element and the execution processes of each module in this model. The simulated experiments were carried out according to the proposed detection model. The experimental results show higher accuracy and efficiency of this model.

Keywords Computer immune system, Blockchain, P2P network, Eclipse attacks

1 简介

区块链系统基于密码学原理而并不基于信用,使得任何达成一致的双方都能够直接进行交易,而不需要第三方中介的参与。具体地,区块链系统使用了点对点分布式[1]的时间戳服务器来生成依照时间顺序排列并加以记录的电子交易证明[2]。因此,作为一种新的分布式记账技术,区块链因具有去中心化、分布式、货币统一性、抗抵赖[3]等优势而受到普遍重视;然而,去中心化、分布式的特点,也使得其在共识机制、网络方面面临诸多安全问题。

51%攻击[4]是针对区块链网络采矿过程的主要威胁之一。当任何共谋的用户或用户组在采矿过程中获得50%以上的计算能力时,该用户或用户组可以通过阻止挖掘某些或全部有效块来获得不法利益。文献[5]表明,即使仅拥有40%的计算资源,攻击者也可以达到50%的成功概率。

双花攻击[6]也是对区块链交易的严重威胁。攻击者使用同一笔比特币支付了两次交易,通过快速支付其中一个交易而造成另一个交易变为无效。因此,其中一个卖家将免费提供服务。攻击者还可以使用同一笔比特币去创建两个交易,并将该笔比特币发送给自己。如果攻击成功,他将不花费任何比特币就能获得一位卖家的服务[7]。

除以上这些攻击外,还有一种会对区块链网络造成严重后果的攻击——eclipse攻击[8]。eclipse攻击的流程如下:攻击节点在区块链受害节点重启前,恶意地填充受害节点的路

到稿日期:2017-11-07 返修日期:2017-12-13 本文受国家重点研发计划(2016YFB0800603)资助。
吕婧淑(1992-),女,硕士,工程师,CCF会员,主要研究方向为信息安全、数据挖掘;杨 培(1981-),女,硕士,工程师,主要研究方向为网络空间安全、数据挖掘;陈 文(1983-),博士,副教授,主要研究方向为分布式网络、信息安全;操晓春(1980-),男,博士,研究员,博士生导师,主要研究方向为多媒体内容安全、计算机视觉,E-mail:caoxiaochun@iie.ac.cn(通信作者);李 涛(1965-),男,博士,教授,博士生导师,主要研究方向为网络安全、人工免疫。

由表，迫使受害节点重启后与路由表中的攻击地址建立传出连接；同时，攻击节点不断与受害节点建立传入连接；最终，达到垄断受害节点的信道、控制其信息流的目的，使其仅能接收攻击节点发送的无用甚至恶意信息。攻击节点若能逐渐对更多节点成功实施 eclipse 攻击，则能够控制更多节点的区块链信道和信息流，进而逐渐控制大部分的区块链网络；攻击者甚至可以在此基础上发起 51%攻击和双花攻击，造成更加严重的后果。

鉴于 eclipse 攻击的严重性及长期性，本文将对该种攻击进行检测。然而，该攻击同时具有隐蔽性和并发性，使得传统的异常检测手段难以有效应对。因此，需要寻求新的检测手段。目前，生物免疫系统非自体抗原的检测机制已经被广泛应用于计算机网络安全领域以进行错误诊断、病毒检测[9]、入侵检测[10]等。生物免疫系统具有的并发性、自组织、自学习、自适应和免疫记忆等特点，使得其非常适合于区块链系统分布式 P2P 环境下的 eclipse 隐蔽攻击的检测。

本文第 2 节介绍相关背景知识和研究现状；第 3 节介绍模型架构；第 4 节介绍检测模型的组件设计；第 5 节为实验及分析；最后总结全文。

2　相关工作

2.1　免疫检测

生物免疫系统[12]是生物体体内一系列的生物学结构和进程所组成的疾病防御系统。免疫系统可以检测小到病毒、大到寄生虫的各类病原体和有害物质，并且在正常情况下能够将这些物质与生物体自身的健康细胞和组织区分开。

同样，计算机也能利用免疫的原理来监测和抵抗有威胁的信息等[13]。因此，将生物免疫系统运用到网络空间安全中进行错误诊断、病毒检测、入侵检测等，可以提高检测的正确性和效率[14]。

具体地，传统的免疫检测模型将数据、操作行为和 IP 包等抽象为抗原，将检测系统中的检测器抽象为免疫细胞，将匹配器抽象为抗体。那么，发现安全威胁这一过程就变为通过免疫检测器将输入的抗原分类为自体和非自体的过程。通过以上这些元素和过程总结出安全威胁自适应感知模型，如图 1 所示。

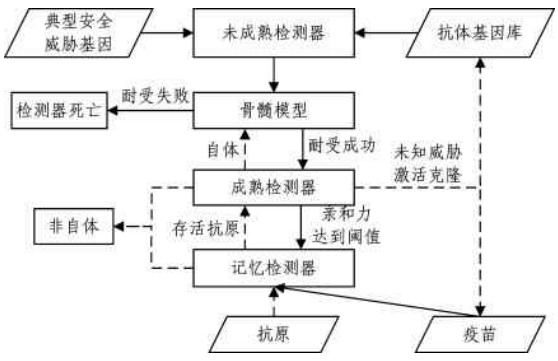

图 1　安全威胁自适应感知模型

Fig. 1　Security threat adaptive perception model

图 1 中，箭头由上至下（实线箭头）代表检测器的生成、进化、学习过程，箭头从下至上（虚线箭头）代表非自体抗原、抗体的发现过程，其中关键步骤包括如下过程。

检测器的生成、进化学习过程：1）通过典型的安全威胁基因、抗体基因库以及随机生成等方法，产生未成熟检测器，并将其送入骨髓模型进行训练；2）将训练的未成熟检测器进化为成熟检测器参与威胁发现工作，在生命周期中一旦发现威胁则其进化为记忆检测器，否则将被淘汰。

非自体抗原的发现过程：1）对数字虚拟资产数据、行为操作等进行特征提取和形式化表达，从而形成抗原；再将抗原依次送入记忆检测器和成熟检测器进行威胁检查，其中记忆检测器发现的是已知威胁，而成熟检测器发现的是未知威胁，其克隆体作为疫苗，被立即注入到其他免疫系统的记忆检测器集合，以迅速使其他系统具备抵御类似威胁的能力，防止类似威胁蔓延。2）匹配自体的错误记忆检测器最终将通过免疫反馈机制予以清除。

2.2　区块链 P2P 网络

传统 P2P 网络——对等式网络（Peer-to-Peer，P2P），又称点对点技术，是无中心服务器且依靠用户群（peers）交换信息的网络体系。它的作用在于减少以往网络传输中的中心节点，以降低资料遗失的风险。与有中心服务器的中央网络系统不同，对等网络的每个用户端既是一个节点，也具有服务器的功能；任何一个节点若希望找到别的节点，则需要依靠查询 DNS 服务器或接收 P2P 网络的广播消息来发现网络中的对等节点。

区块链 P2P 网络中的节点由其 IP 地址标识。具有公共 IP 的节点可以启动与其他区块链节点多达 8 个的传出连接，并且可以接受多达 117 个[8]的传入连接。连接通过 TCP 协议达成[8]。

2.2.1　传播网络信息

区块链网络信息通过其 P2P 网络的 DNS 播种机和 ADDR 消息传播。

DNS 播种机：DNS 播种机是一个服务器，可以通过（非加密认证的）服务器端存储的节点的 IP 地址列表来响应来自节点的 DNS 查询。IP 地址列表的大小受 DNS 的限制；单个 DNS 查询可以返回的最大可能 IP 地址数量约为 4000 个，DNS 播种机通过定期爬取区块链网络来获得这些 IP 地址。区块链网络有 6 个播种机，仅在两种情况下进行查询：1）新节点首次加入网络时，它尝试连接到 DNS 播种机以获取活动节点的 IP 列表；2）现有节点重新启动并重新建立与新节点的传出连接时。这里，只有同时满足节点已有传出连接少于 2 个且节点开始尝试建立连接的时间大于 11 s，才能查询播种机[8]。

ADDR 消息：包含最多 1000 个 IP 地址及其时间戳的 ADDR 消息，用于从对等节点中获取网络信息。如果 ADDR 消息中包含的地址超过了 1000 个，则发送消息的节点被列入黑名单。节点只有在建立了与对等节点的传出连接后，才能发起 ADDR 消息的请求；对等节点最多响应 3 个 ADDR 请求，每个消息包含从对等节点表中随机选择的最多 1000 个地址。

2.2.2　存储网络信息

公共 IP 地址存储于节点的 tried 表和 new 表中。tried 表包含了 64 个存储桶，每个桶为已经建立传入或传出连接的

节点存储地址，最多能存储 64 个不同的 IP 地址。new 表包含了 256 个存储桶，每个桶可以存储还没有成功建立连接的节点的 64 个地址。

2.3 eclipse 攻击

在区块链系统中，P2P 网络是其能够成为去中心化账本系统的核心所在，每个节点在该网络中都是一个中心，从而无需第三方的参与，并且节点之间可以通过 P2P 网络交换信息。若攻击者能够控制该 P2P 网络，从而控制区块链系统的信息流，那么攻击者能够向任意节点发送恶意信息，进而控制整个区块链系统。

eclipse 攻击针对具有公共 IP 的受害者，其详细步骤如下：1)攻击者通过控制多个傀儡节点，向受害节点发起大量持续性 TCP 传入连接，将傀儡节点的 IP 地址填充到受害节点的 tried 表中。2)在建立 TCP 传入连接的基础上，傀儡节点向受害节点发送 ADDR 消息，该消息包含了大量不属于区块链网络的“垃圾”IP 地址。然而，受害节点默认将此类还未成功建立连接的地址存储在其 new 表中。攻击者就是利用受害节点的此特性来达到覆盖受害节点的 new 表地址的目的。其中，“垃圾”地址是未分配的或保留以供将来使用的地址(如 252.0.0.0/8)。3)攻击持续到受害者节点重新启动，并从其永久存储的 tried 表和 new 表中选择新的传出连接。受害者很大概率地将建立的所有 8 个传出连接与攻击者地址相连(所有的 8 个地址均来自 tried 表，因为受害者无法连接到 new 表中的“垃圾”地址)。4)攻击者不断用其攻击地址与受害者相连，从而最终占据受害者剩下的 117 个传入连接。eclipse 攻击模型如图 2 所示。

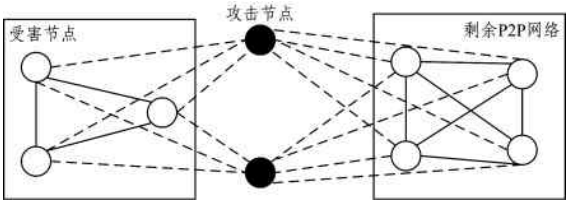

图 2 eclipse 攻击模型

Fig. 2 Eclipse attack model

如图 2 所示，eclipse 攻击成功后，攻击节点垄断了受害节点的传入连接与传出连接，此时攻击节点可以向受害节点发送任意信息并且截取剩余 P2P 网络向其广播的真实信息。

2.3.1 填充 tried 表和 new 表

来自未经请求的传入连接的地址会被存储在 tried 表中。因此，攻击者可以通过从一个地址连接受害节点来将该地址插入受害节点的 tried 表中。此外，节点对新地址的偏好规则意味着攻击者拥有更新的地址时会驱逐旧的合法地址。

区块链节点能够接收未经请求的 ADDR 消息。ADDR 消息中包含的地址可以被直接插入到 new 表中，而节点不会测试它们的连通性。因此，攻击节点通过攻击地址与受害节点相连时，攻击节点能够向受害节点发送包含大量无效的“垃圾”IP 地址的 ADDR 消息。“垃圾”地址将逐渐地覆盖 new 表的所有合法地址。

节点很少从其邻居节点和 DNS 播种器中获取网络信息。因此，当攻击者覆盖受害节点的 tried 表和 new 表时，受害节点几乎从来没有通过查询合法的同伴或播种者来验证其真实性。

2.3.2 受害节点重启

eclipse 攻击要求受害节点重启，节点重启后受害节点才能与攻击地址相连。导致比特币节点重启的原因包括：ISP 停机[8]、关机、矿机的操作系统升级等。

2.3.3 选择传出连接

所有 new 表中的地址都是垃圾地址，即受害节点重启后，若从 new 表中挑选地址建立传出连接，则所有的连接都是失败的。因此，受害节点被迫只能从 tried 表中挑选地址，又由于受害节点偏向选择更新的地址(保证合法地址变得越来越陈旧且攻击地址是新鲜的)，导致受害节点的传出连接全部与攻击地址相连。

2.3.4 垄断 eclipse 受害节点

若 2.3.3 节中的攻击成功，则受害节点有 8 个与攻击地址相连的传出连接；接下来，攻击者必须占据受害节点的所有传入连接才能真正垄断受害节点。为了防止他人连接到受害者，这些 TCP 传出连接可以维持 30 天，受害节点的地址在这个时期内被 P2P 网络所遗忘。

实验证实[8]，区块链节点能够接受来自相同 IP 地址的所有传入连接的请求。维护默认的 117 个传入 TCP 连接的成本为 $4(\frac{164\times117}{80\times60}\approx4)$ 字节/秒，这使得一台计算机很容易同时垄断多个受害者。

若上述攻击成功，那么受害节点的 8 个传出连接和 117 个传入连接分别与攻击地址和“垃圾”地址相连，受害节点的网络通信全都被攻击节点控制，例如：攻击节点可以向受害节点发送无用信息甚至是虚假信息，攻击节点可以截取区块链网络向其广播的信息。

eclipse 攻击的流程如图 3 所示。

图 3 eclipse 攻击流程

Fig. 3 Process of eclipse attack

3 模型架构

本文构建了一个基于免疫的 eclipse 攻击检测模型，其整体架构如图 4 所示。模型由 3 个主要模块构成：eclipse 攻击模块、检测器初次生成模块和免疫应答模块，如图 4 虚线框所示。每个模块的流程如下：

(1)图 4 左半部分为 eclipse 攻击模块。根据 eclipse 攻击的流程，将其分为重启前、重启、重启后 3 个阶段。其中，受害

节点为 V，攻击节点为 A。若 V 重启后，检测到其传入连接与传出连接全部被 A 垄断，说明攻击成功，则收集 A 在重启前向受害节点发送的攻击数据——A 向 V 发起 TCP 连接所使用的攻击 IP 和向 V 发送的 ADDR 消息的相关特征，它们构成了典型的安全威胁基因。

(2)图 4 右上部分为检测器的初次生成模块，实线箭头表示检测器的运行路线。该模块包括了未成熟检测器的生成和成熟检测器的进化过程。用于生成未成熟检测器的样本由两部分组成：一部分是(1)中的典型安全威胁基因本身，另一部分是根据(1)的格式随机生成的样本，这是为了在保证检测器的有效性的同时增加多样性。首先，将以上两部分数据向量化，使其变为一维向量 IP_i 和三维向量 $(fer_i, count_i, pro_i)$，将两种向量统称为 c；然后，为 c 的三维向量的每个向量值设置扩展值 r，即每个向量值加减其对应的扩展值后的取值范围都属于检测器；最后，每个检测器都有其生存年龄，且生存年龄小于生命周期的阈值，生成格式为 $\langle c, r, age\rangle$ 的未成熟检测器。将生成的未成熟检测器作为自体耐受实验(自体数据来自于受害节点的正常网络通信数据)的输入，使用 NSA 算法对每一个未成熟检测器进行训练。若某未成熟检测器不与自体数据中的任何自体匹配，则训练成功，其进化为成熟检测器；若与任一自体匹配，则训练失败，该检测器死亡。

(3)图 4 右下部分为免疫应答模块，虚线箭头表示抗原的运行路线。该模块包括对抗原类别的检测、记忆检测器的进化、抗体及疫苗的生成过程。首先，将包含自体和非自体两种类别的抗原数据作为记忆检测器的输入，若该抗原与记忆检测器集合内的某检测器匹配，则该抗原被分类为非自体数据；若无匹配，则使用成熟检测器集合继续对该抗原进行检测；若有匹配，则其同样分类为非自体数据；若不匹配，则将其作为自体数据填充到耐受训练的输入中。然后，若成熟检测器在其生命周期内匹配非自体的次数超过阈值，则其进化为记忆检测器，用于发现已知威胁。最后，记忆检测器发现未知威胁后，将其注入抗体基因库和疫苗库。

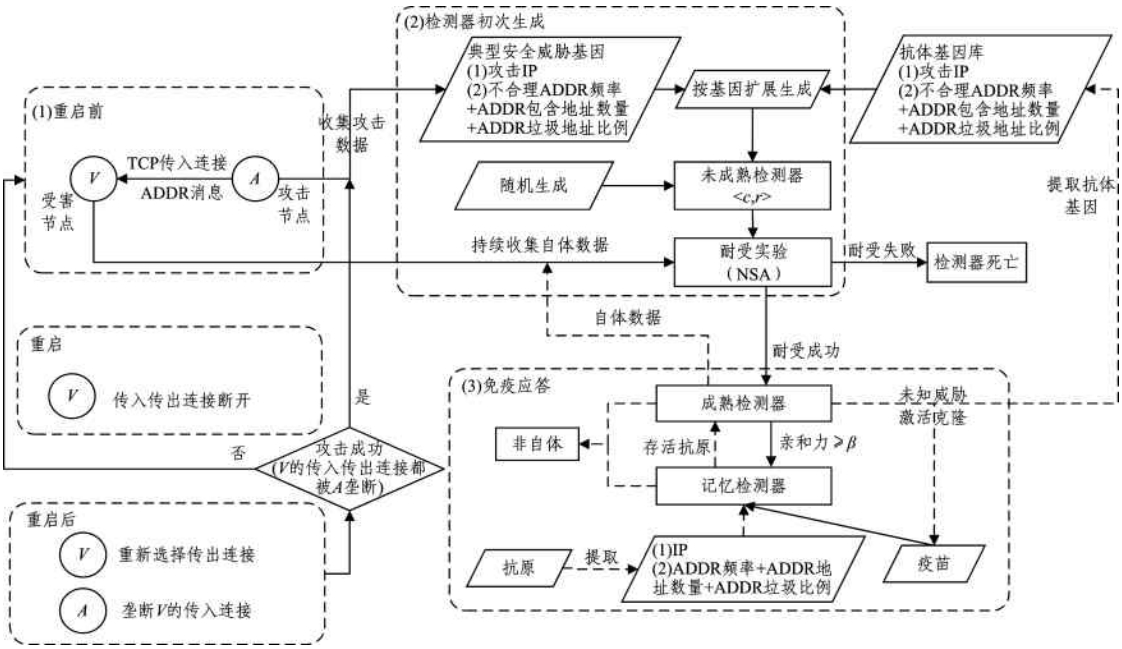

图 4　基于免疫的 eclipse 攻击检测模型架构

Fig. 4　Eclipse attack detection model architecture based on immune

4　模型定义

4.1　模型要素的定义

从图 4 中可以看出，该免疫模型包含了抗原、抗体、检测器和疫苗。其中，抗原包括自体、非自体数据；检测器包括未成熟检测器、成熟检测器和记忆检测器。在描述该模型如何工作前，首先对以上各概念进行形式化定义。

设字符串集合 $E=\bigcup_{i=1}^{\infty}\{0,1\}^i$，$R=\{\langle a,b\rangle \mid a\in B\wedge |a|=l, b\in E\}$，其中 $B=\{0,1\}^l$，l 为自然数，$|a|$ 表示字符串 a 的长度。定义 Ag 为抗原集合，$Ag\subset R$。对于任意抗原 $x\in Ag$，x 是包含其他节点与受害节点建立的 TCP 连接、ADDR 消息在内的数据以及 x 的类别(自体或非自体)。自体集合 $Self\subset Ag$，非自体集合 $Nonself\subset Ag$。$Self\cup Nonself=Ag$，$Self\cap Nonself=\varnothing$。$Self$ 为受害节点的正常网络通信数据集合，$Nonself$ 为来自攻击节点的网络数据集合。那么，$Self$ 即为自体耐受实验的输入。

定义 $I=\{\langle c,r,age\rangle \mid c,r\in B, age\in \mathrm{N}\}$ 为未成熟检测器集合。其中，c 为被向量化后的攻击特征，包括一维向量 IP_i 和三维向量 $(fre_i, count_i, pro_i)$ 的两组向量，r 为 c 的三维向量的扩展值；age 为抗体年龄；N 为自然数集合。定义 $D=\{\langle c, r, age, count\rangle \mid c,r\in B, age, count\in \mathrm{N}\}$ 为检测器集合，其中，$count$ 为匹配数。检测器集合 D 由记忆检测器 M 和成熟检测器 T 两部分组成。因此，$D=M\cup T$，$M\cap T=\varnothing$，$M=\{x \mid x\in D, \forall y\in Self(\langle x,y\rangle\notin Match\wedge x.count\geqslant\beta)\}$ 为记忆检测器集合，β 为匹配非自体次数的阈值；$T=\{x \mid x\in D, \forall y\in Self(\langle x,y\rangle\notin Match\wedge x.count<\beta)\}$ 为成熟检测器集合。其中，集合 $Match=\{\langle x,y\rangle \mid x\in D, y\in Ag, f_{\mathrm{match}}(x.c,y)=1\}$ 为 $B\times Ag$ 的匹配关系，$f_{\mathrm{match}}(x.c,y)$ 的取值取决于 $x.d$ 与 y 之间的

匹配度(亲和力):若其大于给定的阈值,f_{match} 值为 1,否则为 0。那么,若某检测器 x 与属于自体 $Self$ 的 y 不匹配,并且 x 在其生命周期内检测出非自体的次数超过了阈值 β,则该检测器属于记忆检测器。若某检测器 x 与属于自体 $Self$ 的 y 不匹配,并且 x 在其生命周期内检测出非自体的次数小于阈值 β,则该检测器属于成熟检测器。

在一般的免疫模型中,亲和力 f_{match} 可以通过欧几里得距离、曼哈顿距离、海明距离、r 连续位匹配等方式计算。本模型使用如下的匹配方式来计算亲和力:

$$f_{match}(x,y)=\begin{cases}1, & \exists i\neq j,0<i,j\leqslant l,x_i.IP=y_j.c.IP\vee\\ & x_i.fre\in(y_i.fre-r_{fre},y_i.fre+r_{fre})\vee\\ & x_i.count\in(y_i.count-r_{count},y_i.count+r_{count})\vee\\ & x_i.pro\in(y_i.pro-r_{pro},y_i.pro+r_{pro})\\ 0, & \text{others}\end{cases} \tag{1}$$

4.2 构成模块的定义

4.2.1 自体与非自体的演化

$$Self(t)=\begin{cases}\{x_1,x_2,\cdots,x_n\}, & t=0\\ Ag_{Self}(t-1), & t\geqslant 1\end{cases} \tag{2}$$

$$Ag(t)=\begin{cases}Self(0), & t=0\\ Ag_{new}(t), & t\geqslant 1\end{cases} \tag{3}$$

$$Ag_{Self}(t)=Ag(t-1)-Ag_{Nonself}(t) \tag{4}$$

$$Ag_{Nonself}(t)=\begin{cases}x|x\in TH, & t=0\\ \{x|x\in Ag(t-1)\wedge\exists y\in B(t-1)\wedge\langle y,x\rangle\in Match\}, & t>0\end{cases} \tag{5}$$

$$D(t)=M(t)\cup T(t),t\geqslant 0 \tag{6}$$

式(2)和式(3)分别模拟了自体与非自体的演化过程。其中,初始时刻即 $t=0$ 时的自体集合为 eclipse 攻击模块中受害节点的正常网络通信数据集合,即 $x_i\in R(i\geqslant 1,i\in N)$;$t>0$ 时刻的自体集合 $Self(t)$ 为 $(t-1)$ 时刻被免疫应答模块分类为自体的抗原集合 $Ag_{Self}(t-1)$。初始($t=0$)时刻的非自体集合 $Ag_{Nonself}(0)$ 为 eclipse 攻击模块中从重启前阶段收集的攻击数据中提取出的典型安全威胁基因 TH,$TH=\{g|g\in(IP\wedge ADDR)\wedge\exists x\in Nonself(\langle g,x\rangle\in Match)\}$。$IP$ 为攻击 IP 地址的集合,$ADDR$ 为 ADDR 消息发送相关特征的集合。式(4)刻画了自体抗原的演化过程,t 时刻的自体抗原由上一时刻的自体抗原除去该时刻的非自体抗原构成。式(5)表示的是 t 时刻的非自体抗原集合,$t=0$ 时刻时其由典型安全威胁基因构成,$t>0$ 时刻时其为被检测为与典型安全威胁基因 TH 中元素匹配的非自体的抗原集合。$Ag_{new}(t)$ 为 t 时刻新收集的抗原数据。

4.2.2 检测器的初次生成

$$I(t)=\begin{cases}\{x|x\in(TH\cup I_{new}(0))\}, & t=0\\ I_{tolerance}(t)\cup I_{new}(t)-I_{maturation}(t), & t\geqslant 1\end{cases} \tag{7}$$

$$I_{tolerance}(t)=\left\{y\left|\begin{array}{l}y\in I(t)\wedge(y.d=x.d,y.age=x.age+1)\wedge\\ x\in I(t-1),\exists z\in Self(t-1)\langle x,y\rangle\notin Match\end{array}\right.\right\} \tag{8}$$

$$I_{maturation}(t)=\{x|x\in I_{tolerance}(t)\wedge x.age>\alpha\} \tag{9}$$

$$I_{new}(t)=\{y_1,y_2,\cdots,y_n\} \tag{10}$$

图 5 是检测器初次生成模块的示意图。结合方程来看,式(7)模拟免疫细胞在骨髓中的生长过程。$t=0$ 时,初始的未成熟检测器由典型的安全威胁基因 TH 和按 TH 扩展生成;$t>0$ 时,由经历一次自体耐受的检测器和新生成的未成熟检测器除去新生成的成熟检测器而构成。如式(8)所示,$I_{tolerance}$ 为经历一次自体耐受后剩下的检测器。如式(9)所示,$I_{maturation}(t)$ 为 t 时刻经历 α 个耐受期后成熟的检测器。$\alpha\geqslant 1$ 模拟耐受期——未成熟检测器必须在骨髓模型中通过 NSA 否定选择删除那些识别到自体的未成熟检测器,并经历一个周期为 α 的耐受期后方可成熟。如式(10)所示,$I_{new}(t)$ 为在 $t(t>0)$ 时刻根据 TH 和 AB 扩展生成的未成熟检测器(AB 为上一次免疫检测生成的记忆检测器所提取的抗体基因),以达到不断为未成熟检测器集合增加新检测器的目的。

由于该模型中的自体数量相对稳定且较少,因此该模型将以高效的方式产生成熟的检测器。同时,由于 I_{new} 具有随机性,因此新生成的成熟检测器具有较好的多样性,在 I_{new} 中,$y_i=\langle d,0\rangle(d\in B,1\leqslant i\leqslant\varepsilon)$;并且,在耐受过程中,由于自体抗原随时间动态变化,因此对应的耐受也是动态变化的。

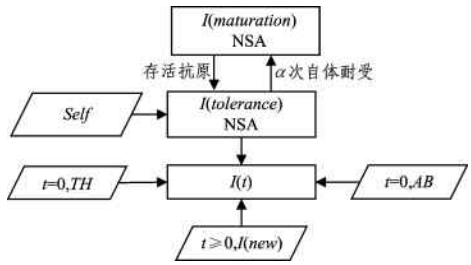

图 5 自体耐受模型

Fig. 5 Self-tolerance model

4.2.3 免疫应答

$$T(t)=\begin{cases}\varnothing, & t=0\\ T'(t)\cup T_{new}(t)-M_{new}(t)-T_{dead}(t), & t\geqslant 1\end{cases} \tag{11}$$

$$T'(t)=T''(t)-P(t)\cup P'(t) \tag{12}$$

$$T''(t)=\{y|y\in T_b\wedge(y.d=x.d,y.age=x.age+1,y.count=x.count,x\in T_b(t-1))\} \tag{13}$$

$$P(t)=\{x|x\in T''(t)\wedge\exists y\in Nonself(t-1)\langle x,y\rangle\in Match\} \tag{14}$$

$$P'(t)=\{y|y\in T_b\wedge(y.d=x.d,y.age=x.age,y.count=x.count+1,x\in P(t))\} \tag{15}$$

$$T_{new}(t)=\{y|y\in T_b\wedge(y.d=x.d,y.age=0,y.count=0,x\in I_{maturation}(t))\} \tag{16}$$

$$M_{new}(t)=\{x|x\in T'(t)\wedge(x.count\geqslant\beta)\} \tag{17}$$

$$T_{dead}(t)=\{x|x\in T'(t)\wedge(x.age>\lambda\wedge x.count<\beta)\} \tag{18}$$

$$M(t)=\begin{cases}\varnothing, & t=0\\ M(t-1)-M_{dead}(t)\cup M_{new}(t), & t\geqslant 1\end{cases} \tag{19}$$

$$M_{dead}(t)=\{x|x\in M(t-1)\wedge\exists y\in Self(t-1)\langle x,y\rangle\in Match\} \tag{20}$$

图 6 是动态免疫记忆模型的示意图。结合公式来看,式

(11)模拟成熟检测器的演化情况。其中，$T(t)$为 t 时刻的成熟检测器集合，当 $t=0$ 时，成熟检测器集合为空；当 $t\geqslant 1$ 时，成熟检测器集合由该时刻已有的成熟检测器新产生的成熟检测器 $T_{\text{new}}(t)$并且除去 t 时刻刚升级为记忆检测器的 $M_{\text{new}}(t)$和已经死亡的成熟检测器 $T_{\text{dead}}(t)$而构成。由未成熟检测器生成的成熟检测器 $I_{\text{maturation}}$若能在生命周期 λ 内与抗原结合且累计足够的亲和力 β，那么就能进化为新的记忆检测器 M_{new}；否则将走向死亡 T_{dead}，并被新的成熟的检测器 T_{new}所代替。如式(12)所示，T'模拟成熟检测器的一代演化，由已有且未死亡的成熟检测器除去与非自体抗原结合过的、累积了一定亲和力的成熟检测器构成。如式(13)所示，T''模拟检测器年龄的自然增长，检测器的年龄 age 增加 1，其中 T_b 表示 t 时刻的成熟检测器集合。如式(14)所示，P 模拟与非自体抗原结合的检测器。如式(15)所示，P'模拟成熟检测器累积亲和力的过程，匹配次数 $count$ 增加 1。如式(16)所示，T_{new}模拟产生新成熟检测器的过程，由未成熟检测器 $I_{\text{maturation}}$进化而来，年龄 age 和匹配次数都为 0。式(17)、式(18)在成熟检测器的生命周期中，通过克隆选择淘汰那些对抗原分类没有作用或作用不大的检测器，保留优势检测器——对抗原具有良好分类作用的检测器使之进化为记忆检测器，以便当类似抗原二次入侵时能进行更高效的应答。式(19)刻画了记忆检测器的动态变迁情况。其中，M_{dead}模拟记忆检测器的死亡调节机制——记忆检测器若匹配自体，将会发生错误识别(将自体中的字符串分类为异常)，该记忆检测器将被淘汰。记忆检测器的淘汰机制能够进一步降低错误率，增强模型的自适应能力。

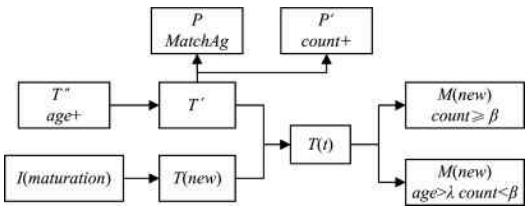

图 6　动态免疫记忆模型

Fig. 6　Dynamic immune memory model

定义由成熟检测器发现未知威胁而激活的抗体为 $AB=\{x\mid \exists y\in M(x=y.d),\forall y\in Self(\langle x,y\rangle\notin Match\wedge x.count\geqslant\beta)\}$。定义由成熟检测器发现未知威胁而灭活的疫苗为 $VC=\{x\mid x\in Ag_{\text{dead}},\forall y\in Self(\langle x,y\rangle\notin Match\wedge x.count=\beta\}$。

5　实验仿真与结果分析

本次实验仿真本地区块链受害节点的网络连接、被攻击的过程，并且分别收集受害节点正常通信的自体数据及攻击节点的攻击数据。

5.1　仿真参数的设置

(1)区块链受害节点的状态：数量为 1，年龄为刚启动 1 天，路由表状态为最坏状态——表中全是新鲜的合法地址。

(2)攻击节点的资源：从 240.0.0.0/8－249.0.0.0/8 中选择攻击 IP，共有 10×2^{24} 个地址。攻击 IP 的数量很大；ADDR 消息包含的“垃圾”地址可从 252.0.0.0/8 中选择。

5.2　仿真实验的步骤

(1)初始化工作：一方面，模仿比特币源码[1)]在本地构建区块链节点，模拟其路由表和网络连接；另一方面，按照 eclipse攻击流程编写攻击程序。

(2)初始化完成后，将该节点的作为受害节点，并将攻击 IP 加入受害节点的传入连接集合中；在此基础上，向受害节点，发送包含“垃圾”IP 的 ADDR 消息；一段时间后，不断地重启受害节点，并查看其传出连接集合中是否全部为攻击 IP。

(3)若全部为攻击 IP，则攻击成功。收集攻击节点用于与受害节点建立传入连接的攻击 IP；发送 ADDR 消息的频率、ADDR 消息所包含的地址数量及其中垃圾地址所占比例等数据将作为典型安全威胁基因。

(4)收集受害节点正常传入及传出连接的 IP 地址、ADDR 消息的频率、ADDR 消息所包含的地址数量及其中垃圾地址所占比例作为自体数据。

(5)将典型安全威胁基因和随机生成的检测器作为未成熟检测器初次生成的输入。

(6)在自体耐受阶段，将步骤(4)输出的自体数据作为 NSA 算法的训练数据；将步骤(5)输出的未成熟检测器集合作为 NSA 算法的输入，生成成熟检测器。

(7)在免疫应答阶段，输入抗原集合，首先使用记忆检测器检测某抗原，若匹配成功，则将其归入非自体类。若失败，则继续使用成熟检测器检测该抗原，若匹配成功，则将其归入非自体类，并将其注入抗体基因库和疫苗库，分别作为下一次检测生成未成熟检测器和记忆检测器的输入；若该抗原与记忆检测器和成熟检测器都不匹配，则将其归入自体类。

(8)在免疫应答阶段，若在生命周期内某成熟检测器检测出某个非自体的次数超过阈值 β，则其进化为记忆检测器。

(9)从步骤(5)开始重复执行以上步骤(这里设置为执行 5 次)，同时不断收集步骤(3)输出的典型安全威胁基因和步骤(4)输出的自体数据。

5.3　对比实验及数据

基于 5.1 节的仿真参数设置，按照 5.2 节的实验步骤执行实验。对于检测攻击的模型，我们更关注模型检测的准确率与效率。因此，将两个指标定义为：准确率＝标记与分类结果相同的抗原数量/抗原总量；效率＝抗原总量/检测需要的时间。结果数据为每秒能检测的抗原数量。本节进行了两大组对比实验。

(1)本文的免疫检测模型相对于传统免疫检测模型的创新点在于，初次生成未成熟检测器的方式——典型安全威胁基因及随机生成。因此，按照未成熟检测器生成方式的不同，第一大组共进行了 3 组检测准确率和效率的对比实验：第一组单纯由随机生成的方式生成未成熟检测器，数量为 3000 个；第二组由典型安全威胁基因扩展生成未成熟检测器，数量为 3000 个；第三小组由典型安全威胁基因、随机生成两种方式各生成未成熟检测器 1500 个，共 3000 个。

在第一组实验中，设置模型参数为：成熟检测器进化为记忆检测器的阈值 $\beta=4$(由于数据量等因素的限制，$\beta>8$ 时实

1) https://github.com/bitcoin/bitcoin.git

验效果较差，因此设置 $\beta=4$)；检测器的扩展值 $r=\langle 0.02,75,0.04\rangle$，$r$ 中的 3 个值分别为所有攻击特征数据中 ADDR 消息发送频率均值的 10%、地址数量均值的 10%和垃圾地址占比均值的 10%。设置数据量为：自体耐受实验中的训练数据为 2000 条；抗原数据共 1000 条，包括自体数量 500 条和非自体数量 500 条。

表 1 在不同未成熟检测器生成方式下的对比实验结果

Table1 Comparison of experimental results with different immature detector generation methods

实验组别	未成熟检测器生成方式	准确率/%	效率(每秒的检测数量)
第一组	完全随机生成	48.60	260
第二组	典型安全威胁基因	72.50	381
第三组	典型安全威胁基因+随机生成	80.06	275

从表 1 中可以看出，第三组实验的准确率最高，但效率相对前两组实验处于中等水平。这是由于：第一组实验中完全随机生成的方式的随机性太大，生成的未成熟检测器有很大一部分是无效的；第二组实验中仅典型安全威胁基因的生成方式完全依赖于“经验”，即从之前受到的攻击中收集的攻击数据，在此种方式下检测的准确率相对较高，但未收集过的攻击特征不会被检测到；而第三组实验结合了前两组实验的生成方式，因此生成的未成熟检测器结合了经验和随机性，具有较好的检测效果，但是相对复杂的生成方式也牺牲了一部分效率，因此效率中等。这说明，典型安全威胁基因和随机生成两种方式相结合而生成的未成熟检测器进化为成熟检测器甚至是记忆检测器的概率更大；但同时也会消耗更多的时间。在本文提出的免疫检测模型中，未成熟检测器使用的生成方式即是第三组方式。因此，第一大组实验证明本文模型提出的创新点是有效的。

(2)在第二组实验中，调整了自体耐受实验中的训练数据量大小，即自体数据量；以下参数和数据保持不变：成熟检测器进化为记忆检测器的阈值 $\beta=4$；检测器的扩展值 $r=\langle 0.02,10,0.05\rangle$，分别为 ADDR 消息发送频率、地址数量和垃圾地址占比的扩展值；抗原数据共 1000 条，包括自体数量 500 条和非自体数量 500 条。

表 2 自体耐受实验中不同自体量的对比实验结果

Table 2 Comparison of experimental results about different amount of self-data in self-tolerance experiment

自体耐受数据量	准确率/%	效率(每秒的检测数量)
1600	77.63	243
1800	80.20	187
2000	82.07	153

从表 2 可以看出，自体耐受数据量越大时模型的准确率越高，其中自体耐受数据量为 2000 条时准确率最高，但是相应的效率较低。准确率高是由于自体耐受实验的训练数据范围决定了未成熟检测器进化到成熟检测器的准确率；但同时，自体耐受实验需要的时间更长，因此效率相对低。

结束语 本文实现了基于免疫的 eclipse 攻击检测模型，首先，总结了区块链受到的 eclipse 攻击的流程及其特征；其次，将基于免疫的检测方式与 eclipse 攻击的特征结合起来，以期在区块链受害节点重启前就预测出可能的攻击源；最后，引入并应用了典型安全威胁基因和抗体基因库，使得该模型检测的准确率和效率相对更高。

参 考 文 献

[1] NAKAMOTO S. Bitcoin: A peer-to-peer electronic cash system [R]. 2008.

[2] TSAI W T, YU L, WANG R, et al. Blockchain Application Development Techniques[J]. Journal of Software, 2017, 28(6): 1474-1487. (in Chinese)
蔡维德，郁莲，王荣，等. 基于区块链的应用系统开发方法研究[J]. 软件学报，2017，28(6)：1474-1487.

[3] HE P, YU G, ZHANG Y F, et al. Survey on Blockchain Technology and Its Application Prospect[J]. Computer Science, 2017, 44(4): 1-7. (in Chinese)
何蒲，于戈，张岩峰，等. 区块链技术与应用前瞻综述[J]. 计算机科学，2017，44(4)：1-7.

[4] BASTIAAN M. Preventing the 51%-attack: a stochastic analysis of two phase proof of work in bitcoin[OL]. http://referaat.cs.utwente.nl/conference/22/paper/7473/preventingthe-51-attack-a-stochasticanalysis-of-two-phase-proof-of-work-in-bitcoin.pdf. 2015.

[5] MALHOTRA Y. Bitcoin Protocol : Model of Cryptographic Proof Based Global Crypto-Currency & Electronic Payments System [OL]. http://www.yogeshmalhotra.com/BitcoinProtocol.html.

[6] KARAME G O, ANDROULAKI E, CAPKUN S. Double-spending fast payments in bitcoin[C]// Proceedings of the 2012 ACM Conference on Computer and Communications Security. ACM, 2012: 906-917.

[7] JOHNSON B, LASZKA A, GROSSKLAGS J, et al. Game-theoretic analysis of DDoS attacks against Bitcoin mining pools[C]// International Conference on Financial Cryptography and Data Security. Springer, 2014: 72-86.

[8] HEILMAN E, KENDLER A, ZOHAR A, et al. Eclipse Attacks on Bitcoin's Peer-to-Peer Network[C]// USENIX Security Symposium. 2015: 129-144.

[9] HARMER P K, WILLIAMS P D, GUNSCH G H, et al. An artificial immune system architecture for computer security applications[J]. IEEE Transactions on Evolutionary Computation, 2002, 6(3): 252-280.

[10] MA H T, JIANG J C. Distributed Model of Intrusion Detection System Based on Agent[J]. Journal of Software, 2000, 11(10): 1312-1319. (in Chinese)
马恒太，蒋建春. 基于 Agent 的分布式入侵检测系统模型[J]. 软件学报，2000，11(10)：1312-1319.

[12] JONES J D G, DANGL J L. The plant immune system[J]. Nature, 2006, 444(7117): 323-329.

[13] JERNE N K. Towards a network theory of the immune system [J]. Annales Dimmunologie, 1974, 125: 373-389.

[14] LI T. An immune based model for network monitoring[J]. Chinese Journal of Computers, 2006, 29(9): 1515-1522. (in Chinese)
李涛. 基于免疫的网络监控模型[J]. 计算机学报，2006，29(9)：1515-1522.