DOI:10.16185/j.jxatu.edu.cn.2018.04.015

基于 Hadoop 的分布式入侵检测系统设计与实现*

洪 波，曹子建

（西安工业大学 计算机科学与工程学院，西安 710021）

摘 要： 由于传统入侵检测系统存在单点无法容错和数据处理能力不足，已无法满足日益增长的信息安全问题. 文中利用分布式技术，基于 Hadoop 的集群运算环境和其快速存储的优势，结合 MongoDB 数据库，采用 Java 体系设计了主数据检测器、RabbitMQ 采集器、收发中间件和分析中心等部件，实现了一种开源的分布式入侵检测框架. 通过对 CPU、MEM、TCP 和网络带宽等四个指标进行监控，能较好的发现外部 DDOS 的攻击和入侵并提供报警服务功能.

关键词： 分布式；入侵检测；数据处理；Hadoop；监控

中图号： TN99 **文献标志码：** A **文章编号：** 1673-9965(2018)04-0390-06

Design and Implement of Distributed Intrusion Detection System Based on Hadoop

HONG Bo, *CAO Zijian*

(School of Computer Science and Engineering, Xi'an Technological University, Xi'an 710021, China)

Abstract: Because of the lack of single point fault tolerance and inadequate data processing capability, the traditional intrusion detection system cannot meet the increasing demand for information security. By using the distributed technology and taking advantages of Hadoop's cluster operation environment and rapid storage, an open source intrusion detection framework is designed, with MongoDB adopted as database. Besides, the main data detector, the RabbitMQ collector, the transceiver middleware and the analysis center are designed, respectively. Through monitoring four indicators CPU, MEM, TCP and network bandwidth, the proposed IDS framework can detect external DDOS intrusion attack and intrusion effectively and provide alarm service.

Key words: distributed; intrusion detection; data processing; Hadoop; monitor control

随着 Internet 的广泛应用和网络安全威胁的日益增多，单一基于主机的入侵检测(Intrusion Detection)技术已经越来越不能满足目前形式多样的安全需求. 此外，黑客入侵与攻击的方式变得越来越隐蔽，且趋于多样性、分布化和组织协同化[1-3]. 因此，入侵检测系统也需要满足跨平台、易扩充、协同检测等新的应用需求[4-6]. 利用分布式技术实现分布式入侵检测系统(Distributed Intrusion Detection System, DIDS)变得更加迫切. Hadoop 是由 Apache 基金会(Apache Foundation)开

* 收稿日期：2018-03-06
基金资助：陕西省教育厅专项科研计划项目(17JK0371；17JZ004)；新型网络与检测控制国家地方联合工程实验室基金(GSYSJ2016007)
第一作者简介：洪 波(1970－)，男，西安工业大学讲师，主要研究方向为应用数学与计算机应用. E-mail:1964383053@qq.com.

发的一个开源的分布式系统基础架构[7-9]. 能够为用户在分布式实现过程透明化的情况下，充分利用其集群运算环境和快速存储的优势，来开发适合用户需求的处理大规模数据的分布式系统. Hadoop 是基于面向对象编程语言 Java 开发实现，具有较好的可扩展性和可移植性，其与入侵检测技术进行结合，具有以下几点优势：① Hadoop 集群的容错性和可用性，某一节点的故障只影响系统的部分性能，不会造成整体系统的失效；② Hadoop 集群的分布式计算，把大规模的数据分析任务均衡到集群中的各个节点，能有效提高系统的分析效率. 因此，将 Hadoop 应用于网络数据的入侵检测，可以有效解决服务器单点失效和大规模分布式入侵数据处理能力的瓶颈问题. 基于此，本文拟设计并实现基于 Hadoop 的分布式入侵检测系统.

1　需求分析

入侵检测技术通过对黑客入侵过程以及行为特征的分析，使得其对入侵过程与入侵事件做出实时的判断以及响应. 从其检测方法上来分，大致有异常入侵检测(Anomaly Intrusion Detection)和误用入侵检测(Misuse Intrusion Detection)两种[10-12]. 在异常入侵检测中，假定条件为所有入侵行为都与正常用户操作行为不同. 建立用户正常行为的模型，理论上能把所有与正常行为不同的行为状态看做为可疑的行为. 例如，通过对网络数据的流量进行统计分析，可以把异常的网络流量视为入侵. 其最大的缺点是并不是所有的黑客入侵都呈现出异常，更为重要的是，正常系统的行为通常难于计算和更新. 而误用入侵检测则假定所有入侵行为和规则能用一种模式或一个特征来进行抽象表达. 因而所有已知的入侵行为都能用规则匹配的方法进行识别. 误用入侵检测的重点是怎样描述入侵模式来区分黑客的入侵与正常用户的行为. 其缺陷是只能识别已知的攻击，而对未知的攻击模式则无能为力.

对比这两种检测方法可以发现，异常检测难于定量分析，具有固有的不确定性. 而误用检测通过定义好的入侵模式，对审计记录信息或网络实时数据流进行模式匹配的检测，但仅限于已知的入侵方式. 由此看来，对于检测方法来说，没有一种方法是完美无缺的，均不能解决所有的入侵问题[13-16]. 因而对入侵检测方法的研究仍然是网络入侵检测领域中的重点和难点之一.

本文结合 Hadoop 集群的结构特点设计了一种分布式入侵检测系统，其结构如图 1 所示. 从图 1 中可知，系统由数据检测器、数据采集器、数据收发中间件、数据分析中心、系统监控以及报警服务等几个部分构成. 系统参考了 check-list 系统的设计思路，在 check-list 系统的基础上添加了数据检测器、系统监控页面、报警服务等功能. 此外，系统的一些其他模块使用开源软件实现，原因在于这些开源的软件大企业应用中大量使用，经受了大量的考验，有利于系统的稳定和长期运行，而自己独立开发的软件系统在稳定性和安全性方面存在问题.

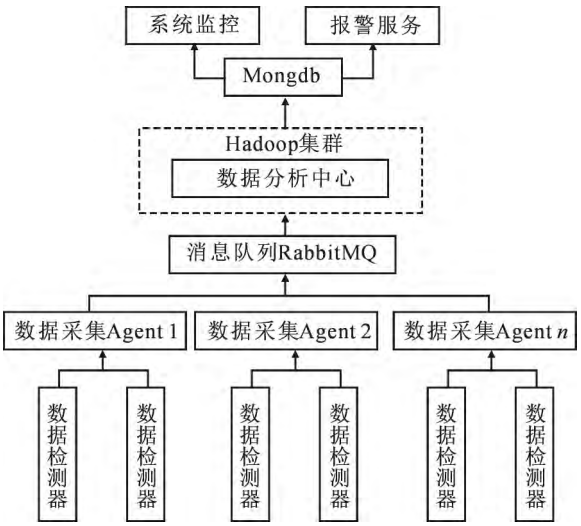

图 1　分布式入侵检测系统结构图

Fig. 1　Structure diagram of distributed intrusion detection system

2　系统设计

2.1　数据检测器

数据监测器是系统数据采集和事件分析单元，分布于系统的底层. 检测器是独立运行于主机上的检测主题，系统对检测器没有任何限制，检测器以 Root 权限运行. 按照数据源分类，数据检测器分为基于主机的检测器和基于网络的检测器. 对于系统而言，使用基于主机的检测器，主要检测主机的 CPU 使用率、MEM 使用率、TCP 连接数、网络带宽、WEB 服务器运行日志和用户行为日志这六个指标. 其中 CPU、MEM、TCP 和网络带宽这四个指标每台主机通用，通过抓取服务抓取这四个指标，以 min 为单位抓取数据，将数据直接写入数据收

发中间件中，再反馈到数据中心；对于用户行为日志和 WEB 服务器运行日志信息通过日志监控服务定时抓取最新的日志信息和定期清理信息. 网络数据检测器使用主机自带的防火墙等保护程序.

2.2 数据采集器

数据采集器在每台监控主机上是唯一的，所有检测器并不直接发送数据到收发器. 数据采集代理控制本地主机上所有检测器. 当检测器要传送数据时，数据采集代理连接到收发器传送. 系统使用 Flume（一款收集海量日志的开源软件）作为数据采集器. Flume 监听数据检测器写入数据的文件，如果文件有新的内容，就将新的内容按行发送到 Flume 的 Collector. Collector 将收集的数据汇总并写入到 RabbitMQ 中.

2.3 数据收发中间件

使用数据收发中间件的原因为：一个监控系统可能需要监控多个区域或者网络，不通区处理的任务类型不同，所产生的数据用途不同，使用数据收发中间件可以将不用的数据分类，便于数据分析中心分析数据. 系统使用 RabbitMQ 作为数据收发中间件，RabbitMQ 是一种消息队列，将不同的日志信息写入到不同的队列中，数据采集器作为数据的生产者只将不同类型的数据写入对应的队列中，数据分析中心作为数据消费端将队列中的数据读出并分析计算.

2.4 数据分析中心

数据分析中心作为核心模块，采用分布式检测、集中存储和集中分析的模式. 通过分析检测器发送回来的数据，根据不同的需要定制不同的数据分析流程，充分的发挥 Hadoop 集群的计算能力，分析收集上来的数据从而发现底层检测不能发现的入侵行为，将入侵检测结果及时反馈到系统管理员，从而调整每台主机的防火墙策略. 数据分析中的分析流程针对不同日志，采用不同的分析流程.

2.5 数据库

采用开源的 NoSQL 数据库 MongoDB. 他是一个基于分布式文件存储的数据库，具有海量数据处理的能力[17]. 其特点为：面向集合，数据存储在数据集中；模式自由，意味着存储在数据库中的文件用户无需知道数据的存储格式；存储在集合中的文档以键—值对的形式存在，便于查找. 主要处理数据采集器采集的日志等数据，有数据量大、格式不固定等特点，传统的关系型数据库不善于处理海量数据，因此使用 MongoDB 作为数据库存储数据分析中心分析完毕的数据，也便于系统监控从 MongoDB 中读取数据及时显示. 系统监控的 CPU、MEM、TCP 连接数、网络带宽在 Mongodb 中的数据格式如下：

id：由 MongoDB 产生，是每一个 document 的唯一标识.

serverName：服务器名称.

quotaName：指标名称，四个指标分别为 CPU、MEM、TCP_CONN 和 NetWork_Width，每个指标对应的集合分别为：quota_cpu，quota_mem，quota_tcp，quota_network_width.

quotaValue：标识指标当前的值.

currentTime：标识数据写入 MongoDB 的时间，便于按时间维度查询 MongoDB.

2.6 系统监控

系统监控模块是用于显示被监控系统每台主机的运行状态，以及 Hadoop 集群的运行状态，通过监控每台主机的运行状态，可以及时的发现外部入侵的痕迹，并及时处理. 监控系统主要是监控每台主机的 CPU、MEM、TCP 连接和网路带宽等基础信息，这些信息是每台机器必须有信息，这些信息可以清晰的反映出每一个主机的运行状况.

2.7 报警服务

报警服务位于整个系统的最顶层. 一方面数据分析中心分析收集到的数据，判断系统是否遭到攻击及运行是否正常，当数据分析中心分析出某些信息异常时，通过报警服务给相关的管理员发送报警信息，包括短信报警和邮件报警. 另一方面当监控系统监控到某台主机的运行状态出现异常时及时报警，如 CPU 使用率居高不下等问题，通过报警服务管理员可以及时发现并处理系统的问题，保证系统的长久运行.

3 系统实现

3.1 数据采集器实现

数据采集器是系统的数据采集最基本的单元，可以采集多种数据，如系统运行的日志等信息以及系统运行时的数据. 本文主要实现被监控主机运行时的 CPU 使用率、内存、TCP 连接数和网络带宽等信息的采集. 以 CPU 使用率和 TCP 连接数为例

介绍实现，其他的信息原理相同. 数据检测器使用 Java 语言实现，使用的协议是 SNMP 协议. SNMP 是基于 TCP/IP 协议族的网络管理标准，它的前身是简单网关监控协议(SGMP)，用来对通信线路进行管理. 数据检测器通过 Java 的 SNMP 驱动包与 SNMP 服务建立连接，连接到 161 端口. 通过给服务端发送相应的 SNMP OID，获取相应的信息，CPU 使用率的 OID 为：1. 3. 6. 1. 2. 1. 25. 3. 3. 1. 2. SNMP 获取数据的方法是 GETNEXT 方法，GETNEXT 使用递归查询的方式查询当前 OID 树下所有 OID 对应的值.

以开发机为例，开发机使用的是 4 核 CPU，因此该 OID 树下应该有 4 个子树，使用 GETNEXT 方法递归 4 次获取所有核的使用率求平均值. TCP 连接数的 OID 为：1. 3. 6. 1. 2. 1. 6. 9，由于系统的 TCP 连接数只有一个值，因此该 OID 树只有一个，只需要递归一次就可以获取到系统当前的 TCP 连接数. 数据检测器的工作流程为：

1) 启动数据检测器. 数据监测器安装在每一台被监控机器上；

2) 初始化数据检测器，链接到被监控机器的 SNMP 服务. 链接 SNMP 服务需要 IP 地址和端口两个值，SNMP 服务监听 161 端口；

3) 开始执行数据抓取，链接到 SNMP 服务后，数据检测器会根据开发人员设置的 OID 遍历 SNMP 服务树上的每个节点，直到找到与设置的 OID 相同的节点，返回相应的值，如果该节点有值则返回，如果没有值则返回 NULL；

4) 在数据检测器程序中开启 4 个线程，线程 1 负责抓取 CPU 使用率信息，线程 2 负责抓取内存 MEM 使用率信息，线程 3 负责抓取当前机器的 TCP 连接数，线程 4 负责抓取网络带宽信息. 这 4 个线程按照分钟级别抓取数据，抓取到数据后写入到指定的 4 个不同的文件中；

5) 数据抓取程序如果没有收到外部强制停止的指令，程序默认不会停止的一直进行循环数据抓取.

3.2　系统监控实现

系统监控基于 B/S 结构开发，采用 Java 体系下的 Spring MVC 框架和 Velocity 模板技术实现[18]. 系统监控的模块功能主要有以下 3 个模块.

1) 用户管理模块：登陆页面以 form 表单形式提交，使用 HTTP 协议的 post 方式提交，在后台收到登陆请求后，现在 Spring MVC 框架的拦截器中进行拦截，验证 cookie. 如果是已登陆用户直接进入主页，如果是未登录用户，查询用户信息的 Collections 验证登陆信息. 如果验证成功. 跳转到主页，如果为成功跳转到错误提示页面.

2) 监控模块：监控模块有指标查看和指标定义两个功能，指标显示用户显示数据检测器抓取到的数据，如 CPU 使用率、TCP 连接数. 指标显示使用 Velocity 和 JavaScript 技术实现. Velocity 使得页面的设计和代码分开便于管理人员维护系统. 指标显示使用 JavaScript 的 HightChart 绘图函数库绘图，每次刷新页面是按照传到后台的参数查询数据库，将查询数据传到绘图函数中显示出结果.

3) MongDB 管理模块：通过 MongoDB 管理模块管理线上的 MongoDB，可以减少管理复杂度. 正常情况下开发人员的开发机无法访问线上的数据库，通过 MongoDB 管理模块，使用 HTTP 协议访问就可以解决无法访问的问题.

3.3　其他模块实现

系统的数据检测器和系统监控以及报警服务需自己实现，其他模块使用了开源软件实现. 数据采集器使用了开源的海量日志收集工具 Flume 收集数据. 数据收发中间件使用 RabbitMQ，用户将不同的数据流写入不同的队列. RabbitMQ 的主要配置为

```
#RABBITMQ_NODE_PORT=8088//端口号
#HOSTNAME=www.ddsnort.com
RABBITMQ_NODENAME=mq
RABBITMQ_CONFIG_FILE=/etc/rab.conf//配置文件路径
RABBITMQ_MNESIA_BASE=/rabbitmq/data//MNESIA 数据库的路径
RABBITMQ_LOG_BASE=/rabbitmq/log //log 路径
RABBITMQ_PLUGINS_DIR=/rabbitmq/plugins//插件路径
```

更详细的配置可以参考 RabbitMQ 的官网. 数据分析中使用了 Hadoop 集群的 MapReduce 计算框架来处理收集到的海量数据，从这些数据中发现黑客入侵的蛛丝马迹. Hadoop 是一个计算框架，在 Hadoop 上可以定义自己的数据处理流程，将定义好的数据处理流程放至 Hadoop 集群运行.

4 系统测试与分析

指标定义及指标查看功能是整个监控系统的核心，数据检测器以分钟级别抓取数据，在展示页面上以点·min^{-1}显示. 通过展示页面可以展示从当天的凌晨到查看时刻系统的 CPU 指标的运行状况，如图 2 所示.

在上述分布式入侵检测系统实现的基础上，使用 DDOS (Distributed Denial of Service)攻击工具对系统进行测试，对被监控系统发起 DDOS 攻击，机器系统存在大量的 TCP 链接，这时可以看到系统监控中 TCP 连接数飙升. 如图 3 所示，正常情况系统的 TCP 连接数趋于一个平稳的状态，由图可知保持在 0～50 之间，当系统遭受 DDOS 攻击时，TCP 连接数急剧增加到 400 左右. 对于外部的攻击监控系统可以及时的发现外部攻击以及入侵.

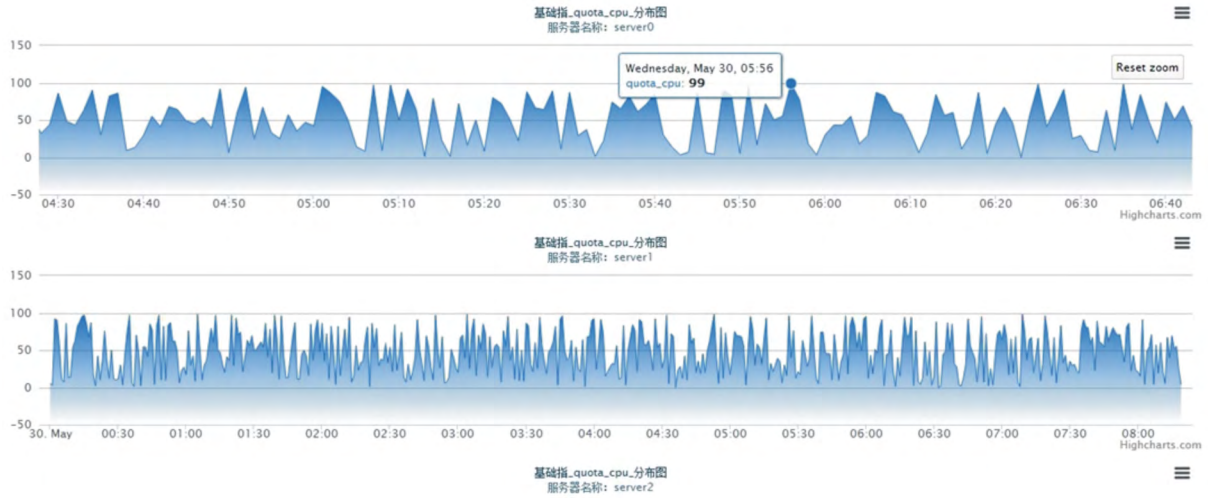

图 2 CPU 指标趋势图

Fig. 2 Trend diagram of CPU indicator

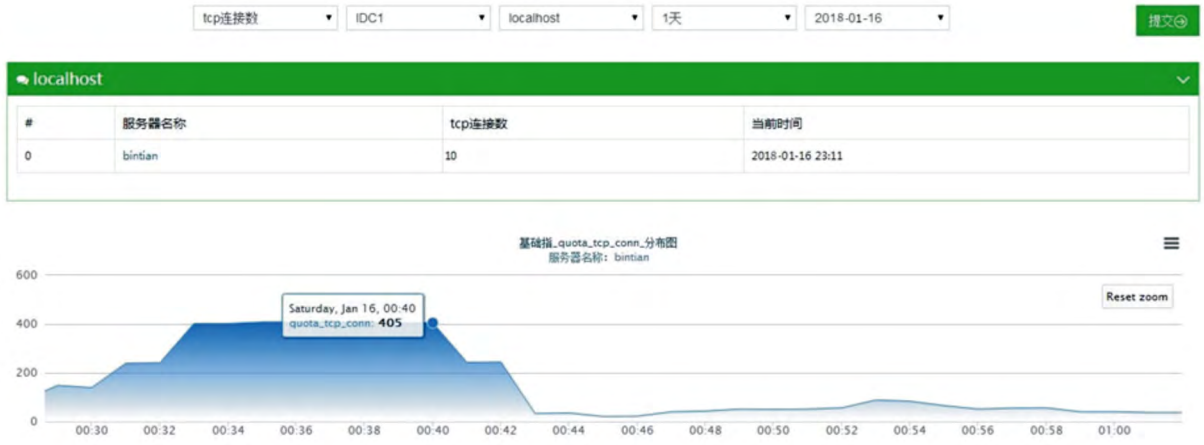

图 3 系统遭受攻击时 TCP 网络连接数示意图

Fig. 3 Schematic diagram of the TCP network connection number when the system is attacked

5 结 论

本文采用分布式技术、开源软件开发思想，基于 Java 体系设计并实现了一种基于 Hadoop 的分布式入侵检测系统，实现了数据采集方式的分布化、数据处理与分析的分布化，通过对 CPU、MEM、TCP 和网络带宽等四个指标进行监控，在主数据检测器、RabbitMQ 采集器、收发中间件和分析中心等四个部件的协同工作下，能较好地发现外部 DDOS 的攻击和入侵并提供报警服务功能，

解决了传统入侵检测系统单点故障和处理能力的瓶颈问题. 该分布式入侵检测系统可以作为商业入侵检测系统的有力补充.

参 考 文 献：

[1] 江颉，王卓芳，陈铁明，等. 自适应 AP 聚类算法及其在入侵检测中的应用[J]. 通信学报，2015，36(11)：118.
JIANG Jie，WANG Zhuofang，CHEN Tieming，et al. Adaptive AP Clustering Algorithm and Its Application to Intrusion Detection[J]. Journal on Communications，2015，36(11)：118. (in Chinese)

[2] 康松林，刘乐，刘楚楚，等. 多层极限学习机在入侵检测中的应用[J]. 计算机应用，2015，35(9)：2513.
KANG Songlin，LIU Le，LIU Chuchu，et al. Intrusion Detection Based on Multiple Layer Extreme Learning Machine [J]. Journal of Computer Applications，2015，35(9)：2513. (in Chinese)

[3] 刘珊珊，谢晓尧，徐洋，等. 基于 PCA 的 PSO-BP 入侵检测研究[J]. 计算机应用研究，2016，33(9)：2795.
LIU Shanshan，XIE Xiaoyao，XU Yang，et al. Research on Network Intrusion Detection Based on PCA PSO-BP [J]. Application Research of Computers，2016，33(9)：2795. (in Chinese)

[4] 曹元大. 入侵检测技术[M]. 北京：人民邮电出版社 2007.
CAO Yuanda. Intrusion Detection Technology[M]. Beijing：People's Post and Telecommunications Press，2007. (in Chinese)

[5] 雅各布森 D. 网络安全基础：网络攻防、协议与安全[M]. 仰礼友，赵红宇，译. 北京：电子工业出版社，2011.
JACOBSON D. Network Security Foundation：Network Attack and Defense，Protocol and Security[M]. YANG Liyou，ZHAO Hongyu，Translated. Beijing：Electronic Industry Press，2011. (in Chinese)

[6] 卿斯汉，蒋建春，马恒太，等. 入侵检测技术研究综述[J]. 通信学报，2004，25(7)：19.
QIN Sihan，JIANG Jianchun，MA Hengtai，et al. Research on Intrusion Detection Techniques：A Suvvey [J]. Journal on Communications，2004，25(7)：19. (in Chinese)

[7] WHITE T. Hadoop 权威指南[M]. 3 版. 华东师范大学 数据科学与工程学院，译. 北京：清华大学出版社，2015.
WHITE T. Hadoop Authoritative Guide[M]. 3rd ed. School of Data Science & Engineering，East China Normal University，Translated. Beijing：Tsinghua University Press，2015. (in Chinese)

[8] 董西成. Hadoop 技术内幕：深入解析 MapReduce 架构设计与实现原理[M]. 北京：机械工业出版社，2013.
DONG Xicheng. Hadoop Technology Insider：In-depth Analysis of MapReduce Architecture Design and Implementation Principles [M]. Beijing：Machinery Industry Press，2013. (in Chinese)

[9] 黄书杭. 基于 Hadoop 入侵检测与防御系统的优化设计与实现[D]. 杭州：浙江工业大学，2015.
HUANG Shuhang. Optimization Design and Implementation of Intrusion Detection and Defense System Based on Hadoop[D]. Hangzhou：Zhejiang University of Technology，2015. (in Chinese)

[10] 钱铁云，王毅，张明明，等. 基于深度神经网络的入侵检测方法[J]. 华中科技大学学报(自然科学版)，2018，46(1)：6.
QIAN Tieyun，WANG Yi，ZHANG Mingming，et al. Intrusion Detection Method Based on Deep Neural Network [J]. Journal of Huazhong University of Science and Technology (Natural Science Edition)，2018，46(1)：6. (in Chinese)

[11] 刁振军，张琦，曹子建. 融合 Snort 和代理的网络异常检测与防御系统研究[J]. 电子设计工程，2018，26(1)：43.
DIAO Zhenjun，ZHANG Qi，CAO Zijian. Research on the Network Anomaly Detection and Defense System Based on Snort and Agent [J]. Electronic Design Engineering，2018，26(1)：43. (in Chinese)

[12] 栾玉飞，白雅楠，魏鹏. 大数据环境下网络非法入侵检测系统设计[J]. 计算机测量与控制，2018，26(1)：194.
LUAN Yufei，BAI Yanan，WEI Peng. Design of Network Illegal Intrusion Detection System in Large Data Environment[J]. Computer Measurement & Control，2018，26(1)：194. (in Chinese)

[13] 徐鑫. 入侵防御系统攻击特征库的建立方法研究[D]. 成都：电子科技大学，2011.
XU Xin. Research on the Method for Building Intrusion Detection System Attack Feature Library [D]. Chengdu：University of Electronic Science and Technology，2011. (in Chinese)

[14] 马里克. 网络安全原理与实践[M]. 李晓楠，译. 北京：人民邮电出版社，2013.
MALIK S. Network Security Principles and Practices [M]. LI Xiaonan，Translated. Beijing：People's Post and Telecommunications Press，2013. (in Chinese)

(下转第 407 页)

[9] 王起全，宋天宝. 基于 BN-LOPA 方法的重油催化工艺火灾爆炸风险分析[J]. 消防科学与技术，2017，36(2)：262.
WANG Qiquan，SONG Tianbao. Heavy Oil FCCU Fire and Explosion Risk Analysis Based on BN-LOPA Method[J]. Fire Science and Technology，2017，36(2)：262.(in Chinese)

[10] TANG A P，OU J P，LU Q N. Lifeline System Network Reliability Calculation Based on GIS and FTA[J]. Journal of Harbin Institute of Technology，2006，13(6)：398.

（编辑、校对　肖　晨）

（上接第 395 页）

[15] 刘建军. WEB 入侵检测技术研究[D]. 北京：北京邮电大学，2015.
LIU Jianjun. Research on WEB Intrusion Detection Technology [D]. Beijing：Beijing University of Posts and Telecommunications，2015.(in Chinese)

[16] 吴丽云，李生林，甘旭升，等. 基于 PLS 特征提取的网络异常入侵检测 CVM 模型[J]. 控制与决策，2017，32(4)：755.
WU Liyun，LI Shenglin，GAN Xusheng，et al. Network Anomaly Intrusion Detection CVM Model based on PLS Feature Extraction[J]. Control and Decision，2017，32(4)：755.(in Chinese)

[17] 霍多罗夫. MongoDB 权威指南[M]. 2 版. 邓明，王明辉，译. 北京：人民邮电出版社，2014.
CHODOROW K. Mongo D B：The Authoritative Guide [M]. 2nd ed. DENG Ming，WANG Minghui，Translated. Beijing：People's Post and Telecommunications Press，2014.(in Chinese)

[18] 埃史尔. Java 编程思想[M]. 陈吴鹏，译. 北京：机械工业出版社，2007.
ECKEL B. Thinking in Java[M]. CHEN Wupeng，Translated. Beijing：Machinery Industry Press，2007.(in Chinese)

（编辑、校对　肖　晨）