文章编号：2096-1472(2021)-03-43-04
DOI:10.19644/j.cnki.issn2096-1472.2021.03.011

基于主动探测的非关系型数据库风险感知系统设计与实现

孙伟明，张华熊

(浙江理工大学，浙江 杭州 310018)
✉swmzstu@qq.com; zhxhz@zstu.edu.cn

摘　要：Elastic数据库是一款主流的非关系型数据库，默认安装时存在潜在的信息泄露风险。本文基于网络主动探测技术，设计实现了一个Elastic数据库风险感知系统。系统首先通过协议构造实现Elastic服务器上各类信息的获取，然后设计了一种基于手机号码、邮箱地址、身份证号、地名地址等多维数据协同分析的敏感信息检测方法，从而评估数据库风险等级并进行预警。本文最后进行了敏感数据检测测试及总体功能测试，实验结果表明了本文敏感信息检测方法及系统设计实现的有效性。

关键词：非关系型数据库；Elastic；信息泄露；主动探测；风险感知
中图分类号：TP309　**文献标识码**：A

Design and Implementation of Non-relational Database Risk Perception System based on Active Detection

SUN Weiming, ZHANG Huaxiong

(*Zhejiang Sci-Tech University*, *Hangzhou* 310018, *China*)
✉swmzstu@qq.com; zhxhz@zstu.edu.cn

Abstract: Elastic database is a mainstream non-relational database with a potential risk of information leakage when installed by default. This paper proposes to design and implement an Elastic database risk perception system based on network active detection technology. The system first realizes acquisition of various types of information on Elastic server through protocol construction, and then designs a sensitive information detection method based on collaborative analysis of multi-dimensional data such as mobile phone numbers, email addresses, ID numbers, and place-name addresses, so to evaluate risk level of the database and issue early warning. At the end of this paper, sensitive data detection test and overall function test are carried out. Experimental results show the effectiveness of the sensitive information detection method and system design and implementation proposed in this paper.

Keywords: non-relational database; Elastic; information leakage; active detection; risk perception

1 引言(Introduction)

当今随着互联网的快速发展，海量非结构化数据对系统存储搜索等实时响应性能要求也相应提高。传统的关系型数据库已不能满足大数据环境下的响应需求，非关系型数据库[1](Not Only SQL，NoSQL)应运而生。与传统的关系型数据库相比，NoSQL数据库不预定义数据模式和表结构，存储类型灵活，并发性能高，可扩展性强。但相较于关系型数据库遵循严格的一致性ACID原则，NoSQL数据库则遵循CAP理论[2]。NoSQL数据库专注于性能和灵活性，其极少内置完整的安全机制[3]。同时NoSQL数据库普遍采用REST的数据接口进行操作，即可通过URL进行数据库的相关操作，这使得NoSQL数据库存在数据安全问题的隐患。Elastic是一个基于Lucene的NoSQL数据库。Elasticsearch是该数据库的搜索引擎，广泛应用于海量数据的搜索功能[4]。本文以网络设备搜索引擎Shodan[5]上可探测的Elastic数据库为研究对象，以主动探测的方式判别是否存在风险，设计实现Elastic数据库风

基金项目：浙江省基础公益研究计划项目(LGG20F020016)．

险感知系统。

2 Elastic数据库存在的安全风险(Security risks of Elastic database)

Elastic数据库是一款主流的NoSQL数据库，其搜索引擎Elasticsearch可提供性能高效的搜索服务，多用于搜索引擎、日志存储、安全防护[6,7]。Elastic的权限、密码设置需要通过扩展插件X-pack来实现。该插件将安全、警告、监视、图形和报告功能捆绑在一个易于安装的软件包中以供用户使用。Elastic可免费使用，但是X-pack收费，免费试用一个月。因此，未安装X-pack的Elastic应用信息容易被泄露。Elastic以文档的形式进行存储，有相应的API可以获取文档中的内容。只要获取Elastic服务器的IP和端口号，即可通过URL进行连接，存在文档泄漏等安全风险。在Shodan官网上搜索Elastic数据库设备在线情况，截至2020年5月30日得到全球分布总量26,763台及在中国的数量9,731台，这些公网上可探测的数据库可能存在一定的安全隐患。

3 系统需求分析(System requirements analysis)

本文设计实现Elastic数据库风险感知系统，目的是判断IP主机上是否存在没有安全机制的Elastic数据库。如若可以不经过验证机制连接到Elastic数据库，则记录下该IP地址，并从该数据库中获取数据进行敏感性检测。敏感数据包含诸如姓名、身份证号、邮编、住址等敏感信息，由敏感数据检测结果判断该Elastic数据库存在的风险大小。

4 系统设计与实现(System design and implementation)

4.1 感知Elastic数据库风险的方法

Elastic数据库使用URL进行交互，其默认端口是9200。本文通过以下命令进行查询：

$$IP:9200/\_cat/indices?v\&pretty \quad (1)$$

如果该IP端没有安装Elastic数据库，则无法访问；若Elastic数据库设置了密码，则会报错，报错信息如图1所示；若能正常访问，则说明存在安全风险。判断Elastic数据库是否存在风险的流程如图2所示。正常访问返回Elastic数据库相关信息，具体信息如表1所示。表1中health和status描述的是数据库的状态；index是索引，可以理解为关系数据库中的数据库名；pri、rep分别表示主集群数量、备份集群数量；其后两个字段表示当前数据库所存的文档数和已删除的文档数；store.size和pri.store.size分别表示主从数据库总的大小、主数据库的大小。在Elastic中，每一个index对应关系数据库中的一个数据库，每个index下的文档类型相似。

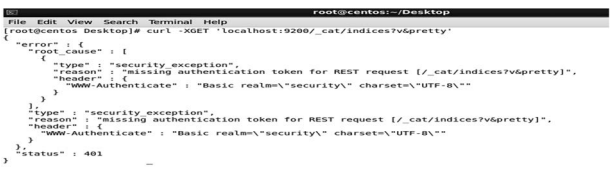

图1 Elastic设置密码报错示意图

Fig.1 Schematic diagram of Elastic password setting error

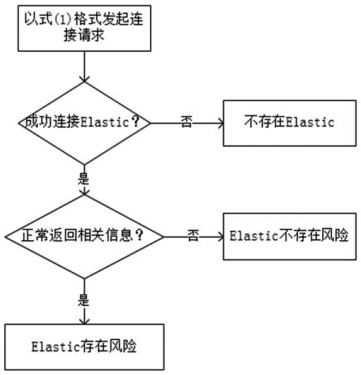

图2 Elastic数据库风险判断流程图

Fig.2 Elastic database risk judgment flowchart

表1 Elastic 索引信息表

Tab.1 Elastic index information table

字段	含义	示例
health	Elastic集群健康状态	green
status	当前状态	open
index	索引	chat
uuid	索引id	hwezy
pri	主集群数量	1
rep	备份集群数量	1
docs.count	当前文件数量	25
docs.deleted	删除文件数量	0
store.size	总数据库大小	70.9 kB
pri.store.size	主数据库大小	35.4 kB

4.2 整体架构设计

整个系统分为日志模块、系统配置模块、IP探测模块、数据检测模块，系统整体流程如图3所示。

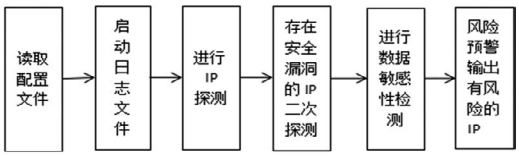

图3 系统整体流程图

Fig.3 System flow chart

系统采用多线程[8]的设计模式，通过Qt平台[9]进行系统实现。多线程的并发使用提高了CPU的利用率，处理网络请求更快，特别有利于网络应答模式的实现。多线程要处理好共享资源的读写，避免产生死锁。Qt具有优良的跨平台特性，支持多种操作系统。此外，Qt面向对象，具有良好的封装性，可重用性好。Qt的信号槽机制(Signals/Slots)代替回调函数，使各个模块之间的协同工作更加简单高效。Qt丰富的API为编程提供了极大便利，其中的正则表达式在后续的内容检测上发挥了很大的作用。

4.3 IP探测模块设计与实现

Elastic数据库探测分两步进行：(1)通过式(1)判断该IP是否存在Elastic数据库。如果该IP端没有安装Elastic数据库，结束探测，否则进入第二步；(2)抓取index下文档的内容进行分析。第一步通过式(1)获得数据库相关的index及对应的大小；第二步通过第一步所获得的index值，通过以下命令查询数据库内容：

$$IP:9200/index/\_search?size=MaxReadTablesNum \quad (2)$$

size是设定查询一次返回的记录数。通过构造URL把满足主数据库特定大小或是包含敏感词的索引获取到相应的记录，并把这些记录保存在输出目录的文件中。

IP探测过程如图4所示。首先使用首次探测队列、二次探测队列来存储上述两步分别构造的URL。在探测开始前，需要从外部输入文件中读取要探测的IP，构造好URL后压入首次探测队列中。探测开始时，基于Qt网络请求的特性，首先设置连接网络请求处理的信号槽机制。在处理网络数据时，为加快请求速度，除了采用多线程技术，还采用了批处理的方式。网络请求正常返回后，调用多线程处理回复的报文。主进程以最大请求量MaxRequestNum发起探测请求，当返回后的数据流中含有index记录，且pri.store.size满足探测最小存储值时，构造式(1)所示的URL加入二次探测队列中，以备二次探测。当所有IP均完成第一步探测后，进行第二步探测。二次IP探测请求时，数据库返回的记录保存在输出文件中。二次IP探测时，通常某个IP下存在多个满足条件的索引index，为了避免Elastic在大并发读取数据库时引起的拒绝访问，本文引入了延时处理机制。一批请求发出后，会延时固定时间后再进行下一批次的访问请求，延时时间的设定在保证Elastic服务端不被拒绝的同时要保证程序的高效性。综合这两种因素本文设定默认延时为2 秒。当二次探测队列为空时，为保证最后一批请求的数据能顺利保存，设定固定延时等待，程序默认延时等待5 分钟，随后访问请求阶段结束。

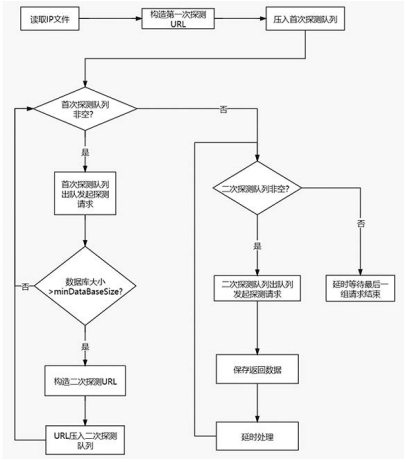

图4 IP探测流程图

Fig.4 IP detection flowchart

4.4 敏感数据检测模块设计

从Elastic数据库中取出数据后，进行相应的敏感检测，以进一步评估安全风险。检测数据中是否包含电话号码、身份证号、邮箱地址、地名地址等相关信息。包含这些信息，证明该数据库有着较高的安全隐患。前期探测得来的数据以字符串的形式保存在输出文件中，检测模块要在字符串中快速比对出上述四种相关信息。电话号码、身份证号、邮箱地址均有鲜明可识别的特征，故采用正则表达式的方式进行识别；地名地址识别较前者更为复杂，需单独设计算法检测。检测得出的敏感信息文档及对应的IP、数据库索引index均输出保存在风险数据库文件夹下。

(1)识别电话号码。电话号码在这里主要考虑手机号码的识别。手机号码一共有11位数字，前两位数字只能是13/14/15/16/17/18/19中的一个，后9位为0—9的数字。由此得到手机号码的正则表达式：

$$\wedge(1[3-9])\backslash d\{9\}\$ \quad (3)$$

(2)识别身份证号。现在我国的居民身份证号是18位，由17位数字体本体码和1位校验码组成。从左往右排列顺序依次是：6位数字地址码、8位数字出生日期码、3位数字顺序码和1位数字校验码。1—6位为地址码，首位可取1—8中任意一个，其余5位可取0—9。7—14位为日期码，前4位是年，后4位分别是月、日。年前两位只能取18/19/20之一，后两位为0—9；月是01—12；日期要考虑到2月没有30，只到28/29。15—17位为顺序码，取0—9；18位为校验码，取0—9或X/x。由此得到身份证号码的正则表达式：

$$[1-8]\backslash d\{5\}(18|19|20)\backslash d\{2\}((0[1-9])|(1[0-2]))(([0-2][1-9])|10|20|30|31)\backslash d\{3\}[0-9Xx] \quad (4)$$

(3)识别邮箱地址。邮箱地址的基本格式为“名称@域名”。名称部分，为了检测的全面性，这部分允许有汉字、字母、数字。汉字用转义字符表示，正则表达式为：

$$[\backslash u4e00-\backslash u9fa5] \quad (5)$$

域名部分允许由字母、数字、英文句号、下划线、中划线组成。由此可得出识别邮箱地址的正则表达式：

$$[A-Za-z0-9\backslash u4e00-\backslash u9fa5]+@[a-zA-Z0-9\_-]+(\backslash.[a-zA-Z0-9\_-]+)+ \quad (6)$$

(4)识别地名地址。由于地名地址的复杂和多样性，无法通过简单的正则表达式来完成。为此本文提出了一种简单高效的地名地址识别方法。该方法在进行地址检测时，首先对文档内容以标点符号进行分句，对每个句子进行识别。其次，建立省、市、县三级行政名称数据库，分别存放在省、市、县三个文件夹下，以备算法识别时匹配使用。判断是否是地址的标准：包含省、市、县(区)、乡镇(街道)、村组、号室院路等六级地址中的三个以上，至少到乡镇一级且顺序从大到小不乱序则判定是地址信息。

具体的方法(图5)思路如下：

(1)查找字符串中是否包含省级名称，定位其在字符串中的位置，用indexofProvince表示，没匹配到记为-1，匹配到则记录其在字符串中的位置，同时匹配成功计数count+1；市县与此类似，在字符串中的位置分别记为indexofCity、indexofCounty。

(2)镇或是乡一级的地址直接按照关键字镇、乡来匹配，其在字符串中的位置记为indexofCountryside，匹配成功计数count+1；村组或院楼号室等匹配同理。

(3)判断是否包含地址信息。count大于等于3、最后一级地址不是县区级、六级地址中存在的几级地址从大到小在分句中位置依次递增，这三个条件同时满足，才判定包含地址信息；否则不包含地址信息。

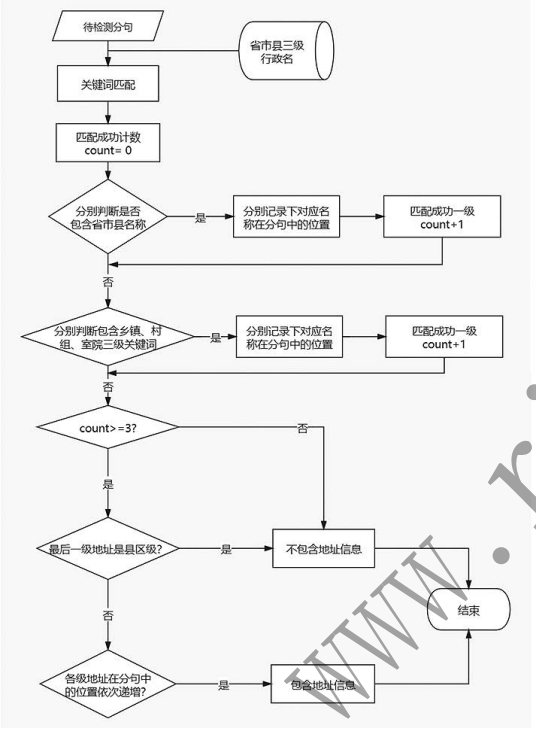

图5 地址识别算法流程图

Fig.5 Address recognition algorithm flowchart

在识别出敏感信息后，系统会把该IP地址和相应的index、敏感信息的记录一起输出到敏感数据文件中，而正常的数据会被输出到正常的文件中进行保存。

5 系统测试(System test)

5.1 敏感数据检测测试

本文通过实验对敏感检测方法进行有效性测试。对于以正则表达式方式进行判别的手机号码、身份证号、电子邮件这三类，检测的数据充分考虑检测可能遇到的各种情况(具体的情况已在前文描述)，分别以50 条真实数据进行测试，检测结果是100%识别出了对应的敏感信息。地名地址的种类比较多，本文为了随机抽取全国各地的标准地址，测试地址由100 个211高校地址、50 个各地医院地址、50 个酒店饭店地址组成，基本涵盖了各种类型的标准地址。检测结果为成功检测192 个，未成功检测8 个。四类检测算法的准确率如表2所示。

表2 敏感检测算法准确率

Tab.2 Sensitive detection algorithm accuracy rate

项目名称	检测数/条	检出数/条	准确率
手机号码	50	50	100%
电子邮箱	50	50	100%
身份证号	50	50	100%
地名地址	200	192	96%

地名地址中检测未成功的地址分为以下四类：(1)市—县—路—学校名(学校名字中有省名)导致不能识别。例如：郑州市郑东新区金水路与博学路交叉口龙子湖高校园区河南中医学院。(2)省—市—县—路—号(市名中含有镇、乡、村、院、路等关键字)，导致不能识别。例如：河南省新乡市牧野区建设路东段46号。例中新乡市含“乡”字。(3)未达到三个地址级别的判定条件。例如：郑州市中原路与大学路交叉口西100米路南黄河饭店。例中只有市和路两级，算法中未涉及交叉路口的情况。(4)由于中文断句分词的误识别导致把地址方位词识别成省名。例如：青岛市黄岛区长江西路66号。算法在长江西路中识别出“江西”这个省名，导致未能准确识别出地址信息。

在误检测方面，手机号码、电子邮件、身份证号检测分别对容易误识别的数字串、字符串等各50 条进行识别。手机号码因为如12015700230034112一段数字串里包含类似手机号的数字15700230034，被误识别。身份证号的误检测也存在类似的问题。因为手机号码和身份证号的长度是固定的，分别是11位和18位，本文在检测前对每个数字串进行长度判断，提前判别出不符合要求的数据，避免误报的情况发生。电子邮件未发生误检测的情况。对于地址地名的误检测，本文采用随机真实的数据，分别是：(1)含省、市公司名的句子100 条，如河南新野纺织股份有限公司；(2)只含省-市-县三级的地址名50 条，如浙江杭州市江干区；(3)易误识别的文段50 条，如北京中关村软件园等。检测结果均能准确识别为非地名地址，未发生误报的情况。

总体来说，本文设计的敏感检测方法简便快捷，准确率在96%以上。

5.2 总体功能测试

使用搜索网络空间在线设备的专用搜索引擎Shodan搜索关键字Elastic，得到全球在线Elastic设备共26,763 个，在中国的有9,731 个。统计记录下在中国的IP后，作为输入数据进行探测，未输入账号密码就可以获取到相关的数据即存在安全风险。在数据检测部分检测出敏感信息，则设置为高风险预警；若没有检测出敏感信息，则设置为低风险预警。

经过上述过程探测得到数据，并对数据进行分析，得到

存在Elastic漏洞地区分布统计如表3所示。由探测的结果可知，广东、北京、浙江、上海等地区存在风险的Elastic数据库数量最多。未知地区的Elastic数据库大多是云服务提供商所有的数据库。测试用的9,731 个Elastic数据库中存在风险的共有9,665 个，全国各个省(自治区、直辖市)均有分布。测试结果说明系统各功能已实现。

表3 存在漏洞风险的Elastic全国分布表

Tab.3 Elastic national distribution table at risk of vulnerabilities

分布区域	存在风险的Elastic数	分布区域	存在风险的Elastic数
未知区域	2,644	陕西省	27
广东省	2,169	湖南省	26
北京市	1,915	海南省	26
浙江省	1,018	重庆市	23
上海市	528	宁夏回族自治区	20
香港特别行政区	288	内蒙古自治区	12
山东省	249	江西省	12
江苏省	140	广西壮族自治区	11
天津市	120	云南省	10
四川省	74	黑龙江省	10
台湾地区	57	贵州省	10
河南省	55	山西省	7
河北省	50	吉林省	7
湖北省	39	新疆维吾尔自治区	4
福建省	39	西藏自治区	4
辽宁省	34	甘肃省	4
安徽省	33	总计	9,665

6 结论(Conclusion)

Elastic数据库作为当前主流NoSQL数据库之一，提供的Elasticsearch搜索引擎是流行的企业搜索引擎，广泛应用于分布式文档存储和搜索服务。Elastic需要收费软件X-pack进行安全配置。本文设计实现的Elastic数据库风险感知系统，从主动探测的角度出发，运用简单的URL请求即可批量判断目的IP主机上是否存在未进行安全配置的Elastic数据库，同时通过对抓取数据的敏感性检测，实现了数据库风险高低的评估和预警以及对Elastic数据库的风险感知能力。

参考文献(References)

[1] 陈忠菊.NoSQL数据库的研究和应用[J].电脑编程技巧与维护,2020(09):81-83.

[2] 唐婷.大数据环境下NoSQL数据库技术[J].信息与电脑(理论版),2019(15):142-144.

[3] 郎云海.大数据下的NoSQL数据库安全策略的改进[J].通讯世界,2019,26(08):29-30.

[4] 杨文杰,倪平波,宋卫平,等.基于Elasticsearch服务化的探究[J].科技资讯,2020,18(24):64-65;68.

[5] Shodan. Shodan官网[EB/OL].[2020-05-12].https://www.shodan.io/.

[6] Ruoyu Wang, Daniel Sun, Guoqiang Li, et al. Pipeline provenance for cloud-based big data analytics[J]. Special Issue: Software Tools and Techniques for Fog and Edge Computing, 2020,50(5):658-674.

[7] Neel Shah, Darryl Willick, Vijay Mago. A framework for social media data analytics using Elasticsearch and Kibana[EB/OL]. [2018-12-11]. https://link.springer.com/article/10.1007/s11276-018-01896-2.

[8] 唐曦文.多线程在仪器控制软件设计中的研究与应用[J].航空精密制造技术,2020,56(04):23-25;32.

[9] 夏梦迎,武樱楠,侯家成,等.基于Linux Qt的动力集中动车组显示屏设计及实现[J].铁道机车与动车,2020(01):18-20;35;5.

作者简介：

孙伟明(1993-)，男，硕士生.研究领域：软件设计，网络安全.

张华熊(1971-)，男，博士，教授.研究领域：软件设计，网络安全.

(上接第62页)

[3] 谢修芳.软件实训教学资源服务系统设计与实现[D].长沙:湖南大学,2015.

[4] 张立臣.实训管理系统的设计与实现[D].沈阳:东北大学,2015.

[5] 谷春英,姚青山.物联网物理空间实体的关联关系建模研究[J].电子元器件与信息技术,2019,3(12):16-17.

[6] 张月红.高等院校网络靶场建设的需求分析及架构功能设计[J].软件工程,2020,23(6):42-44.

[7] 韩燕丽,杨慧炯.工程应用导向的面向对象系列课程体系重构[J].软件工程,2019,22(3):60-62.

[8] 马恬煜.UML面向对象分析与设计[M].北京:清华大学出版社,2018.

[9] 施莹.Ajax技术在物联网信息系统中的应用[J].无线互联科技,2020,17(08):149-150.

[10] 邢彤彤,覃蕊,高峰.基于PHP+MySQL技术的农家乐推广网络系统开发与实现[J].计算机产品与流通,2020(5):52.

[11] Eyada M., Saber, W., EI Genidy, et al. Performance Evaluation of IoT Data Management Using MongoDB Versus MySQL Databases in Different Cloud Environments[J]. IEEE Access, 2020(8):110656-110668.

作者简介：

刘青龙(1996-)，男，本科生.研究领域：电子信息工程技术.

王法胜(1983-)，男，博士，教授.研究领域：软件设计.本文通讯作者.