doi:10.6043/j.issn.0438-0479.201604023

基于开源软件的 DNS 查询日志分析系统

郑 海 山

(厦门大学信息与网络中心,福建 厦门 361005)

摘要:域名系统(domain name system,DNS)是互联网的核心基础服务,服务的健壮性和安全性非常重要.针对高等学校的 DNS 配置中存在的问题,提出了一个基于开源软件的 DNS 查询日志分析系统,给出构建 DNS 集群自动化部署的方案,利用开源工具监控 DNS 的配置信息和运行状态,并利用大数据分析工具结合少量的编程生成查询日志的可视化图表.实际运用后表明,该系统通过横向扩展可应对每日上亿条数据的实时分析要求.DNS 服务整体架构清晰,安全性提高,用户的上网日志可实时统计展示,为分析 DNS 服务的运行状态、攻击预警、网络性能调优等方面提供了帮助.

关键词:域名系统;BIND;大数据;日志分析;可视化;部署自动化

中图分类号:TP 393 **文献标志码**:A **文章编号**:0438-0479(2017)02-0252-07

域名系统(domain name system,DNS)作为最古老的互联网服务存在,提供域名和服务器 IP 地址之间的相互映射,使得用户无需记忆物理服务器的 IP 地址而只通过直观的主机名即可访问互联网服务.由于其重要性,所以 DNS 协议设计得非常简单,并且一般客户端会设置至少两个 DNS 服务器地址以免其中某台出现故障.协议的简单和客户端的冗余配置使得 DNS 故障率较低,或者故障被冗余配置掩盖,导致管理员对 DNS 服务器疏于维护.实际上,随着上网客户端越来越多,网络攻击越来越普遍[1],黑客水平越来越高,BIND(Berkeley internet name daemon)等 DNS 软件特性开发导致默认配置参数的持续变更,使得 DNS 服务器需投入更多的维护工作,然而教育网内部分高等学校(简称高校)DNS 系统尚存在着诸多安全隐患.

由于 DNS 日志量较为庞大,而且 DNS 服务器日志多台分散,一般只是利用错误日志进行分析,排查问题.随着技术的进步,存储价格下降,机器性能提高,通过大数据技术基于 DNS 日志数据进行分析变得可行.季成[2]对 CN 节点的 DNS 查询日志使用聚类和统计分布进行了分析.苏政[3]利用信息地理学对 CN 域名日志进行了多个尺度的空间分析,揭示了互联网访问在空间上的规律.章思宇[4]提出一套基于 DNS 流量评估各类恶意软件的方法.查诚吉[5]、董再旺[6]、王帅[7]、BEGLEITER R[8-9]、Jose[9]等应用大数据框架 Hadoop 对日志进行分析.以上文献均对 DNS 的查询日志做了各类有意义的挖掘,分析过程大多利用了 Hadoop 工具.然而,Hadoop 在 DNS 日志分析方面显得过于复杂,需要较多的集群安装、配置和编程工作量,并且挖掘所得的结果不够直观.董再旺[6]对挖掘结果使用 PHP 语言实现的可视化分析监控系统也存在编程工作量稍大的问题.同时,以上文献只被动利用 DNS 日志的分析结果查找安全问题,对于实践中 DNS 集群的自动化部署和安全配置等最佳实践涉及较少.

本研究提出一个新的方法,通过开源 Ansible 软件自动化构建高效安全的 DNS 服务器集群,结合 dig、tcpdump、bindgraph、Nagios 对服务器进行监控,基于开源 Filebeat、Logstash、Elasticsearch 软件,提供比 Hadoop 更简单的安装配置和更实时的分析结果;同时使用 Kibana 软件通过简单的编程和配置即可提供基于各种维度的可视化分析图表.

1 高校 DNS 系统的问题和解决思路

高校 DNS 系统常见问题包括以下几点:

1) 操作系统没有更新到最新,没有安装必要的安

收稿日期:2016-04-14 **录用日期**:2016-07-06

基金项目:福建省中青年教师教育科研项目(JAT160019)

Email:haishion@xmu.edu.cn

引文格式:郑海山.基于开源软件的 DNS 查询日志分析系统[J].厦门大学学报(自然科学版),2017,56(2):252-258.

Citation:ZHENG H S.DNS query log analysis system based on open source software[J].J Xiamen Univ Nat Sci,2017,56(2):252-258.(in Chinese)

全补丁.

2) 没有配备或者配置防火墙.

3) DNS 配置错误导致信息泄露.配置错误包括开放 DNS 查询,导致 DNS 不必要地对互联网提供服务,增加被攻击和被利用发起发射攻击的风险,包括没有配置严格的访问控制列表(access control list,ACL),错误配置权威记录传输协议(authoritative transfer,AXFR)导致整个域信息被恶意下载.

4) 缺乏对 DNS 服务器运行状态的有效监控.

5) BIND 软件没有更新到最新,导致存在内存泄露和拒绝服务攻击等问题.

6) DNS 攻击发现手段缺乏,没有充分利用查询日志.

为解决以上问题,高校应使用最新的操作系统,部署最新的 DNS 软件,规划 DNS 服务器集群架构,按角色分开部署.集群内配置适当的防火墙策略,对外只提供必要的服务.对于有些高校,可限制缓存 DNS 只在校内提供服务,可在防火墙端设置或者使用 ACL 限制.

为了使得安装配置自动化,可采用开源运维自动化工具 Ansible 来部署整个 DNS 集群.Ansible 可自动化部署应用软件,自动化管理配置项.配置 DNS 集群只需要定义各个角色,每个角色安装软件,修改配置,加入 IP,这样,一个命令即可高效完成整个集群部署.

对于配置错误引起的问题,应通读 DNS 软件手册和更新记录,并使用监控或者自定义脚本定期运行核查配置.

对于 DNS 查询日志的分析可采用开源软件 Elasticsearch,Elasticsearch 是一个分布式的搜索和分析引擎,基于开源的 Lucene 轻量级全文索引引擎工具包,Lucene 可对所有文档进行全文分词索引,通过自定义的查询语言可以实现对文档高性能的查询.索引的结果使用开源软件 Kibana 展示,既可实时分析日志,也可保存分析语句便于在任意时刻调用结果,甚至可跟监控系统对接实现某些指标超过阈值后自动报警.

2　系统设计方案及其功能

通过以上对高校 DNS 系统问题的研究,结合解决思路设计了如图 1 的系统整体架构.

图 1 中的所有角色均支持横向扩展,其中 Logstash 对 CPU 要求较高,内存和硬盘空间要求较低,而 Elasticsearch 集群内部负责数据存储的节点对硬盘空间要求较高,负责检索的节点对 CPU 和内存要求较高,应差异化配置服务器.

系统实现的源代码保存在版本控制系统内,分为以下 4 个主要模块:

1) Ansible 脚本定义代码.编写 Ansible 配置文件定义图 1 所示的整个 DNS 集群和日志分析集群.Ansible 脚本通过角色定义各个服务器配置,服务器的防火墙配置,服务器运行的软件和软件的配置,开源监控系统的配置.对于服务器的配置变更和增加修改均可通过修改 Ansible 脚本并运行一条命令完成.

2) DNS 记录数据代码.DNS 记录数据为文本格式,属于 DNS 软件的配置,本应在 Ansible 脚本定义代码内定义.然而由于记录的变更较为频繁,如果每次变更一个记录均需运行整个集群的 Ansible 脚本较为繁琐,所以把频繁修改的记录数据和整个集群定义脚本隔离开来,方便记录更新和使用自动化工具生成.

3) 自定义脚本代码.自定义脚本代码包括一些对

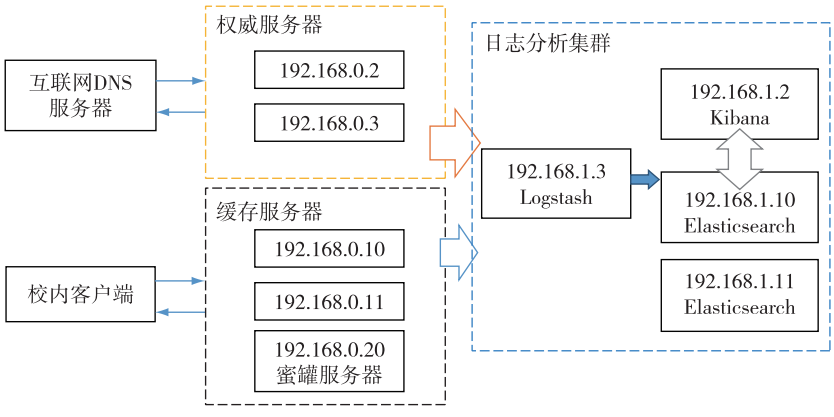

图 1　DNS 查询日志分析系统架构图

Fig. 1　Architecture of DNS log analysis system

简单脚本的组合和保存常用脚本运行命令参数的搭配方式,方便自动化运行脚本检查和通过拷贝替换参数即可针对某些项目进行分析.

4) 查询日志分析子系统代码.查询日志分析子系统的配置首先通过 Kibana 自带的 GUI 配置,该配置属于手工操作,本应无法自动化.然而由于配置信息以 JSON 文本形式存在于 Elasticsearch 内,所以配置变更后通过脚本自动下载配置信息保存到源代码,方便系统重建时快速建立整个查询日志分析子系统和对子系统变更历史的追朔.

查询日志分析子系统保存 60 d 内的 DNS 查询记录,可视化揭示用户上网行为、攻击行为的模式和趋势.通过分析厦门大学的需求,本子系统已实现以下功能:

1) 结合 Lucene 查询语句可实现各类自定义查询统计,各个字段均可查询,可组合查询,可保存常用查询.

2) 可按时间钻取(drill down)查询,提供各种时间粒度的查询结果,可合并集群内多个服务器的日志.

3) 查询可视化,在单一仪表盘内一次性展示多个可视化图表.

4) 按饼状图显示服务器域名查询量.

5) 折线图显示各服务器域名查询量随时间分布情况.

6) 显示查询量最大的前几个客户机.

7) 显示校内域名被查次数最多的前几个站点列表.

8) 显示 IPv4 和 IPv6 DNS 查询数量.

9) DNS 查询请求类型分析.

10) 旭日图显示校内用户对互联网内容提供商(internet content provider,ICP)访问的排名情况[10]

11) 查询错误日志,展示十几类错误信息的数量.

3 DNS 的选择和安装配置

3.1 DNS 软件选型

DNS 软件可选较多,常见的有 BIND、PowerDNS、Dnsmasq、Microsoft DNS、互联网 DNS 提供商提供的云 DNS 服务、硬件 DNS 产品等[11].Windows 下的域名系统服务器可基于图形界面管理或者采用脚本管理,自动化程度较低,配置文件不如文本配置直观.云 DNS 服务使用较为简单,成本较低,但是对于 DNS 记录的变更控制力较弱.硬件 DNS 产品完整性较高,功能较多,使用也较为方便.然而硬件存在价格成本,产品化较高导致自定义功能较弱,同时软件的安全更新频率也较低.所以本文中选择当前互联网最为广泛使用的 BIND9 DNS 软件.

3.2 DNS 整体架构

首先将权威服务器和缓存服务器分开部署:权威服务器只负责面对互联网的缓存服务器查询厦门大学域名使用,又分为主服务器和从服务器.缓存服务器提供给本校师生使用,可安装任意多个.分开部署使得其中任何一个服务器即使遭受攻击也不会影响其他服务器,也有利于日志收集时对查询日志进行分类.

服务器可采用虚拟机和实体机混合,操作系统采用 Windows 和 Linux 混合、多个操作系统版本混合,并使用多个 DNS 软件,分布在不同的物理位置以提高多样性.

图 1 中的权威服务器为主从 2 台服务器,为了更为安全,可横向扩展为 3 台服务器:1 台主服务器,2 台从服务器,DNS 记录在主服务器更新,主服务器在互联网隐藏起来,只允许管理员登陆,2 台从服务器对外提供授权域名查询.

缓存服务器一般安装 2 台,并公布 IP 地址给师生使用.同时也可增加多台特殊用途的缓存服务器或转发服务器,比如图 1 中充当简单蜜罐角色的服务器.由于互联网的攻击存在无目的性,攻击前一般会扫描整个网段的 DNS 服务器,所以蜜罐缓存服务器对外不公布 IP 地址,如果蜜罐缓存服务器有任何查询则将查询方 IP 记录下来,作为可能的攻击源分析.

图 1 中多台 DNS 的更新机制为:在主权威服务器变更 DNS 记录后重新加载配置文件后,通过 DNS notify 机制通知从权威服务器立即更新主服务器变更的记录,并立即使用命令 rndc flushtree 远程通知缓存服务器刷新主权威服务器的 Zone 域.为了使得 DNS 记录变更能尽快传递到整个互联网,应根据需求动态调整不同 DNS 记录的存留时间(time to live,TTL).

对安全和高可用性要求更高的,可使用开源软件 LVS(Linux virtual server)、Keepalived 或者商业硬件产品在前端做负载均衡.

3.3 BIND 软件安装和配置

服务器需要最小化安装,剔除任何不需要的功能.BIND 软件安装应当从包管理器安装,不建议自行编译源代码以防止安全更新较为困难.服务器应当配置防火墙,DNS 集群内各个角色服务器的防火墙的配置可参考表 1.

表 1　服务器的防火墙配置策略

Tab.1　Firewall Configuration Strategy

服务器角色	防火墙开放端口	备注
主权威服务器	互联网 IP 的 TCP/UDP 53 端口	若主权威服务器对互联网隐藏,可只允许从权威服务器访问 53 端口
从权威服务器	互联网 IP 的 TCP/UDP 53 端口	
缓存服务器	主权威服务器的 TCP 953 端口 校内 IP 地址的 TCP/UDP 53 端口	953 端口为 rndc 远程控制端口
缓存服务器(蜜罐)	互联网 IP 的 TCP/UDP 53 端口	
Logstash 服务器	所有 DNS 服务器的 TCP 5044 端口	
Elasticsearch 服务器	所有 DNS 服务器、Logstash、Kibana 的 TCP 9200 端口	9200 端口为 Elasticsearch 对外提供服务的端口
Kibana 服务器	无对外端口开放	只允许管理员访问

为了使得配置文件可读性更高,可使用 $INCLUDE 关键字拆封 DNS 配置文件,对于有相同功能的 DNS 记录,应放入同一个文件,与其他业务整合而自动生成的 DNS 记录也应放入同一个文件,通过 $INCLUDE 进入主配置文件,隔离各个不同功能的 DNS 记录,也为 DNS 记录版本控制带来好处.

BIND 较为重要的几个配置项解析如下:

1) recursion:配置是否接受递归查询,权威服务器应当设置为否.

2) allow-recursion:接受哪些客户端的递归查询,一般设置为校内 IP.

3) allow-transfer:AXFR 重要配置,限制哪些 IP 可下载整个域信息.

4) max-cache-size:控制 DNS 服务器使用缓存内存的大小.

5) rate-limit:防止发射放大攻击.

6) recursive-clients:控制 DNS 服务器对外并发查询的数量,默认限制1 000.观察本数值可判断 DNS 查询的繁忙程度和是否正在被攻击.

如果需要部署智能 DNS,即对于同一个域名,根据查询客户端的源地址提供离它最近的服务器 IP,可参考多 view 的配置方法[12].

3.4　DNS 检测工具和自定义脚本

DNS 软件安装完成后,为检测配置是否正确,可使用以下开源工具检查:

1) named-checkconf:用于在每次修改 DNS 配置后检查语法错误.

2) rndc:named 的命令行管理工具,可用于 DNS 配置变更后刷新配置,收集 BIND 服务器状态,导出缓存数据分析等.

3) dig:DNS 客户端工具,可用来检查本机和上下游 DNS 服务器的配置.DNS 整体体系结构类似树形,对于高校来说,校内权威服务器的认定是由 edu.cn DNS 服务器授权的,所以可使用 dig 检查 edu.cn 对于本校 DNS 服务器配置的正确性.具体操作为可运行命令 dig edu.cn 得到 edu.cn 的所有权威服务器,再运行命令 dig @dns.edu.cn xmu.edu.cn 对每个权威服务器检索即可检查正确性.

4) tcpdump:实时截取 DNS 服务器网络中传输的数据包分析,并提供逻辑语句过滤出系统管理员想要分析的数据包,可在攻击时协助定位攻击源.配合 dig,可检查服务器解析 DNS 记录是否正常.

5) Nagios:开源监控软件,可监控 DNS 服务器是否存活和服务器状态是否正常.

6) Fail2ban:Fail2ban 可分析 DNS 查询日志,通过自定义规则,限制某些 IP 对 DNS 服务器的请求;可以控制 DNS 客户端对服务器查询的频次,可部分抵挡恶意机器的攻击扫描.

7) grep,awk,sort,cut 等命令行工具:可组合这些工具编写自定义脚本对 DNS 配置文件的序列号进行自动化增长,也可对 DNS Log 进行简单分析,为大规模日志分析做准备.

8) bindgraph:简单的开源可视化 DNS 日志分析工具.适合单服务器,可作为集群 DNS 日志分析的一种补充.

对于 DNS 管理员,应当熟练掌握以上工具的使用方法.DNS 的攻击形式多样,新方法层出不穷,应当可通过组合以上工具分析 DNS 运行状态和排错,有识别攻击源或类型的能力.

3.5 DNS 监控

DNS 单服务器的监控使用开源 Nagios 监控,通过插件 check_ping 监控服务器是否存活,通过 check_tcp 监控 TCP53 端口是否开放,通过 check_udp 监控 UDP53 端口是否开放,通过 check_dns 监控 DNS 请求是否正常.通过安装代理上报服务器的 CPU、Load、内存占用、硬盘空间等数据.在系统状态出现异常情况下第一时间发送邮件报警到管理员邮箱.

通过开源软件 Topbeat 收集 DNS 服务器的 CPU、内存、硬盘、进程数,进入 Elasticsearch,可在事后追朔服务器的运行状态.

4 BIND 查询日志分析子系统的实现

DNS 日志存于文本文件,日志量庞大,集群内多个服务器日志分散,无法进行快速查询,所以需要使用采集工具中心化所有日志,并进行全文索引,方便分析.DNS 日志的大小跟 TTL 有相当大的关系,统计出来某个站点的查询量大不一定代表这个站点的访问量就较大,有可能只是因为站点的 TTL 设置较小而已.黄振[13]提出了一种对 DNS 查询日志间接测量域名访问量的方法,并考虑了 TTL 的影响因素,进行加权调整.所以对 DNS 日志的分析应当参考 TTL 值,然而如果某个客户端的查询量非常大,则可直接以这个数据判断客户端是否中病毒.

由于 DNS 日志会记录访问者的上网记录,所以日志应当在一段时间后删除.如要提供给第三方分析,应当对 IP 地址进行不可逆哈希,防止用户的上网隐私泄露.

4.1 日志的采集

查询日志分析子系统采用在 DNS 服务器安装开源软件 Filebeat 收集日志,对于采集工具的选择,需要对被采集服务器影响较小,Filebeat 只需要一条命令即可安装,通过配置 DNS 日志保存目录,即可自动收集.收集完的日志通过网络传送给 Logstash 分析.

由于 DNS 日志较大,从 Filebeat 到 Logstash 到 Elasticsearch,可能存在吞吐量问题,如果下一个步骤来不及处理数据会被简单丢弃.解决的方法是根据各个应用自身的日志信息,收集吞吐量和出错信息,横向扩展多台 Logstash 或者 Elasticsearch,或者在中间加入队列服务器异步化处理.

4.2 日志的索引、保存

Logstash 的主要作用是对日志进行分析,预处理,分拆成不同字段,根据后期查询的要求提供数据源.DNS 查询日志格式为:

```
09-Feb-2016 03:38:25.786 queries: client 127.0.0.1#9527 (dog.xmu.edu.cn): query: dog.xmu.edu.cn IN A+(192.168.0.10)
```

如果整条记录不加处理直接存入 Elasticsearch,Elasticsearch 会对记录进行分词索引,查询速度虽然较快,但不利于分字段分析.所以我们需要把查询日志拆封,把各个字段分析出来,分别存入 Elasticsearch.以上的查询日志,通过正则表达式,可以拆封得到查询时间、客户端 IP 地址、所查询的域名、查询的类型等信息.

为了更加精细化分析,对于 IP 地址,通过开源 IP 地理信息数据库,获得 IP 对应的国家、省份、城市和具体位置.而对于校内的 IP 地址,则不使用开源 IP 地理信息数据库,而是替换成校内自定义的片区楼宇数据库,在查询时即可根据片区楼宇来统计查询量.

对于查询的域名,可以根据域名使用点号进行分级的特点,通过正则表达式,拆封域名,得到二级、三级域名和主机名.

拆封的 Logstash 关键代码如下:

```
grok {
match=> {"message"=> "%{DATA:querydate} queries:client %{IP:clientip}#%{INT} \(%{DATA:queryhost}\): query:%{DATA} IN %{WORD:rrname} [+-]%{WORD} * \(%{IP}\)"}
}
#对 DNS 客户端进行地理信息定位,对权威服务器的查询使用开源 IP 地理信息数据库,而对于缓存服务器的查询应使用校内自定义的片区楼宇数据库
geoip {
source=> ["clientip"]
}
#由于已经将 message 全部拆封成字段保存到 Elasticsearch,所以原始 message 已无必要保存,此举可节省近一半硬盘空间.
mutate {
remove_field=> ["message"]
}
if ([queryhost]=~"^[^.]+\.[^.]+$") {
grok {
match=> {"queryhost"=> "^(?<domainname2>[^.]+)\.(?<domainname1>[^.]+)$"}
```

```
    }
    }
    if ([queryhost]=~"^[^.]+\.[^.]+\.[^.]+$") {
    grok {
    match=> {"queryhost"=> "^(?<domainname3>[^.]+)\.(?<domainname2>[^.]+)\.(?<domainname1>[^.]+)$"}
    }
    }
    if ([queryhost]=~"^.*?[^.]+\.[^.]+\.[^.]+\.[^.]+$") {
    grok {
    match=> {"queryhost"=> "^.*?(?<domainname4>[^.]+)\.(?<domainname3>[^.]+)\.(?<domainname2>[^.]+)\.(?<domainname1>[^.]+)$"}
    }
    }
```

Logstash 预处理后的字段提交到 Elasticsearch 保存,由于日志量较大,而且由于查询日志保存了用户的上网记录,所以需要定期清理 Elasticsearch 数据库空间,本子系统的规则是自动清理 60 d 前的记录,清理使用 curator 脚本,放入 crontab 每日定期运行,运行命令为

```
/usr/local/bin/curator --host 127.0.0.1 delete indices --older-than 60 --time-unit days --timestring'%Y.%m.%d' --prefix dnslog
```

由于查询日志大小的不确定性,有可能某日攻击会导致生成非常大量的查询日志,所以管理员必须经常性观察 Elasticsearch 集群占用磁盘的情况.

4.3　日志可视化展现

日志存储到 Elasticsearch 后,通过 Kibana 实现对日志的可视化查询,建立基于 Lucene 查询命令,对于需要经常性分析的,可保存以待随时调用查询.而对于需要可视化展示的,则构建相应的图表.

以创建校内用户对互联网 ICP 访问的排名情况的图 2 旭日图为例,具体步骤为:新建一个可视化图表,选择饼状图,使用新的搜索,饼状图的聚合度量 metrics 使用 Count 函数,buckets 桶字段分别采用 domainname1、domainname2、domainname3 字段依次切片,为顶级域名、二级域名和三级域名.为了使结果更精确,通过搜索排除查询量较大的疑似攻击日志.

图 2 旭日图中最内层的圆表示校内用户对互联网诸如.com、.cn、.net、.org 等顶级域名的总访问量对比,内层圆外的第 1 个圆环体现对应的顶级域名下的二级域名诸如 qq.com、baidu.com、360.cn 等的访问量对比.由于国内高校的域名一般为.edu.cn 结尾,所以校内用户对国内其他高校的访问量对比体现在第 2 个圆环内.

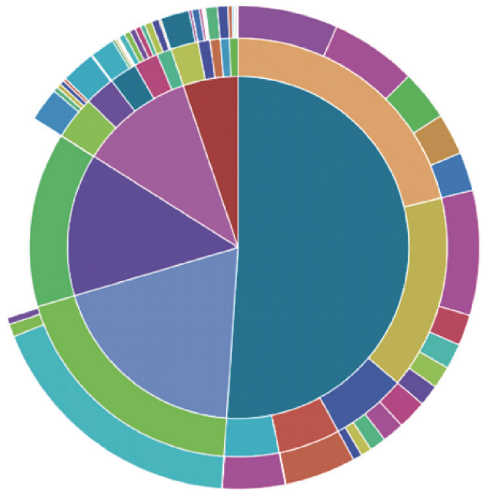

图 2　对互联网 ICP 访问的排名情况的旭日图

Fig. 2　Sunburst chart of ICP access count ranking

通过以上方法,建立多个旭日图、折线图、柱形图、表格、地图等可视化图表,并放入 Kibana 的 Dashboard,即可实现在一个页面查看服务器总的查询量、分服务器查询量、校内域名查询量排名、查询量最大的客户端排名、DNS 查询请求类型、校内用户对互联网访问情况、查询客户端地理信息分布情况等各类统计分析信息.

5　结　论

应用本系统后,DNS 集群内服务器职责清晰,服务器安全性提高,自动化程度提高,服务器的运行状况透明.本系统性能方面可承受厦门大学每日近 40 GB 的 DNS 查询日志总量和每日近 3 亿次的查询总数的日志分析.根据查询日志的统计,可以揭示用户对各类网站访问量的对比,揭示工作日和假期用户的访问行为习惯变化趋势.定期针对查询量最大的多台客户端和蜜罐服务器收集到的客户端,交由网络部通知用户检查是否中毒,为防病毒和识别网络攻击提供了新的渠道.针对内部模拟的反射攻击、大批量查询请求攻击,均可通过可视化图表快速定位,取得了良好的效果.不足之处在于由于没有排除 TTL 对 DNS 查询量的影响,所以无法得出域名准确的访问量.对上网日

志的精确分析,应当直接获取交换机的上网日志,而本系统使用的开源大数据分析架构也同样适用于海量上网日志的实时分析,今后可继续在这方面进行研究.

参考文献:

[1] 任立军.域名系统 DNS 安全增强的研究与设计[D].成都:电子科技大学,2013:1.

[2] 季成.基于 DNS 查询日志的互联网访问模式分析[D].北京:清华大学,2009:56.

[3] 苏政.基于日志数据的域名访问源多尺度分析[D].南京:南京师范大学,2013:9.

[4] 章思宇.基于 DNS 流量的恶意软件域名挖掘[D].上海:上海交通大学,2014:14.

[5] 查诚吉.基于 DNS 日志的移动互联网分析[D].北京:北京邮电大学,2014:14.

[6] 董再旺.国家域名日志可视化分析监控系统设计与实现[D].北京:中国科学院大学,2014:3.

[7] 王帅,汪来富,金华敏,等.网络安全分析中的大数据技术应用[J].电信科学,2015(7):145-150.

[8] BEGLEITER R,ELOVICI Y,HOLLANDER Y,et al.A fast and scalable method for threat detection in large-scale DNS logs[C]// 2013 IEEE International Conference on Big Data.[S.l.]:IEEE,2013:738-741.

[9] JOSE A S,B A.Automatic detection and rectification of DNS reflection amplification attacks with hadoop mapreduce and chukwa[C]//2014 Fourth International Conference on Advances in Computing and Communications (ICACC).Cochin:IEEE,2014:195-198.

[10] 季成,李晓东,袁坚,等.基于 k-means 算法的 DNS 查询模式分析[J].清华大学学报(自然科学版),2010(4):601-604,608.

[11] WIKIPEDIA.Comparison of DNS server software[EB/OL].[2016-02-12].https://en.wikipedia.org/wiki/Comparison_of_DNS_server_software.

[12] 张焕杰.多 view 的 DNS 服务器配置文件组织[J].中国教育网络,2009(5):75.

[13] 黄振.基于 DNS 缓存的域名访问量估测[D].哈尔滨:哈尔滨工业大学,2013:21.

DNS Query Log Analysis System Based on Open Source Software

ZHENG Haishan*

(Information & Network Center,Xiamen University,Xiamen 361005,China)

Abstract:Domain name system is one of the most important parts of the Internet.Robustness and security of the service are extremely important. However, numerous problems exist in the University's DNS configuration. This paper, through the setup experience of Xiamen University,proposes a DNS query log analysis system based on open source software.This system gives the best practice of how to automatically build DNS cluster,the method of monitoring and examining the DNS configuration and running status by using open source tools.Additionally,the system offers the query log visualizations generated by using big data analysis tools combined with a small amount of programming.Furthermore,the system can deal with real-time analysis of more than one hundred million bits of data daily through horizontal expansion.After using the system,DNS service exhibits a clear structure and security.The query log statistics shows in real time.All these features offer great help for analyzing the running status of the DNS server, showing attack warning,and optimizing network performance.

Key words:domain name system;bind;big data;log analysis;visualization;automation deployment