增强的 Zeek 网络流量采集与监控分析系统设计

沈　萍[1],陈俊丽[1],张汉举[2]
(1. 上海大学 通信与信息工程学院,上海 200444;
2. 上海博弋信息科技有限公司,上海 200030)

摘　要:随着计算机技术和网络攻击手段的不断发展,网络监控需求不断强烈。针对企事业单位网络监控效果与实际需求不匹配、缺少可复用的流量采集与监控分析一体化系统的现状,该文设计了增强的 Zeek 的网络流量采集与监控分析系统,用于企事业单位的流量管理。系统利用 Zeek 的可扩展性,设计了多端口识别与自定义采集时间间隔的功能,实现了对网络汇聚流量的更精准和灵活的采集。接着,将采集数据的本地存储与持久化存储相结合,在 Web 端提供对网络安全数据的全面分析。系统实现了流量数据的定制化采集、持久化存储与 Web 交互展示和控制功能,在保证现有应用系统平稳运行的前提下,降低了信息时延,满足了真实大规模网络环境流量数据的个性化采集和实时监测与溯源分析需求,同时为进一步扩展为其他应用模式提供了可用的架构基础。

关键词:网络流量;Zeek;个性化采集;端口识别;流量监控分析;Web
中图分类号:TP39　　**文献标识码**:A　　**文章编号**:1673-629X(2024)10-0077-07
doi:10.20165/j.cnki.ISSN1673-629X.2024.0196

Design of an Enhanced Zeek Network Traffic Collection and Monitoring Analysis System

SHEN Ping[1], CHEN Jun-li[1], ZHANG Han-ju[2]
(1. School of Communication and Information Engineering, Shanghai University, Shanghai 200444, China;
2. Shanghai Boyi Information Technology Co., Ltd., Shanghai 200030, China)

Abstract: With the development of computer technology and network attack methods, the need for network monitoring continues to be strong. We present a network traffic collection, monitoring and analysis system based on enhanced Zeek. The system is designed to address the discrepancy between enterprises and institutions' current network monitoring capabilities and their actual needs. It also aims to provide a reusable, integrated system for traffic management. The system utilizes Zeek's scalability and incorporates multi-port identification and customized collection intervals to achieve a more accurate and flexible collection of network aggregated traffic. It then combines locally stored collected data with persistent storage to comprehensively analyze network security data on the web. The system enables personalized collection, real-time monitoring, and traceability analysis of traffic data in large-scale network environments and reduces information latency while ensuring the smooth operation of existing application systems. It provides a foundation for further expansion into other application modes.

Key words: network traffic; Zeek; personalized collection; port identification; traffic monitoring analysis; Web

0　引　言

网络是当今社会发展中不可或缺的组成部分,随着信息技术的迅猛发展,人们已从互联网时代步入大数据时代,随之而来的是网络数据的急剧增长和网络环境的日益复杂。网络数据包含了大量的数据交互记录,黑客们不断更新迭代的安全威胁手段引发众多安全事件,如数据库信息泄露、DDoS 攻击、SQL 注入、0Day 攻击等,对网络管理和安全监控提出了新的挑战。为了维护网络的可用性、保障信息的安全,甄别和分析异常流量和安全事件,网络流量的采集[1]、监控与分析变得至关重要。

网络流量采集是研究网络行为、进行网络规划的基础,依靠流量采集设备完成。在流量采集设备上,当前市场上有多种具备网络传输实时监视功能的流量采

收稿日期:2024-02-19　　**修回日期**:2024-06-20
基金项目:国家自然科学基金(12174245)
作者简介:沈　萍(1999-),女,硕士生,研究方向为网络安全和机器学习;通信作者:陈俊丽(1972-),女,副教授,博士,研究方向为网络信息安全、智能信号与信息处理。

集工具和入侵检测系统(IDS)。常用的 Netflow[2] 流量采集技术通过识别端口号对具体协议种类和应用流量组成展开分析,但不能说明网络流量的具体协议组成与占比。Packetbeat 部署轻量化,但对于网络流的处理为累加方式,难以体现当前时刻的流量情况以及流量传输速率。Suricata[3] 作为兼容 Snort[4] 规则的多线程入侵检测系统,支持的协议分析有限,在处理大数据流量时需要有更多的硬件资源,所消耗的内存与流数量成正相关。

流量监控与分析系统承载着网络流量和流向分析的功能。当前已有相关系统设计方案。为了满足大数据场景的需求,文献[5]基于 Hadhoop 生态系统提出了一种大数据存储、分析和管理的解决方案,适用于已有 Hadoop 生态系统的企业参考部署。文献[6]将离线批处理与实时流式处理计算相结合,提出了大规模网络异常流量实时监测系统架构。文献[7]采用分布式云服务系统完成生产现场的数据采集和控制,但应用范围在工业环境。文献[8]采用日志解决方案 ELK 与分布式消息队列 Kafka 为数据平台的运维监控工作提供参考,但未在数据形式多样的现实应用场景中部署测试。在实时监控的同时,找到一种有效利用采集数据开展监测与分析的一体化系统解决方案,且能在企事业单位内部通用且可复用,是亟待解决的问题。

基于上述流量采集设备功能与实际需求不完全匹配、已有监控分析系统性能不完善等情况,该文设计了一种增强的 Zeek 网络流量采集与分析系统,对接现实采集需求,降低数据产生到被系统检测的时延,提高网络安全事件发现与追溯能力。在 Zeek 支持深度解析各种协议机制下,该文首先通过代码扩展进行精确的多端口识别和自定义采集时间间隔,缩小流量监控误差时间。然后,采用 Elasticsearch(ES)平台加入对网络流量数据的持久化存储。同时设计一个系统显控平台实现多样化数据分析,使用户在 Web 端进行数据查阅、分析预警与免输条件查询,帮助企事业单位开展日常监控与安全事件溯源工作。最后在企事业单位真实网络环境应用该系统,验证了该系统的可行性与有效性。

1 系统框架设计

增强的 Zeek 网络流量采集与监控分析系统由流量采集、流量存储、流量统计与分析三个阶段组成,系统的总体框架如图 1 所示。

在流量采集阶段,部署在监测服务器上的流量采集工具捕获通过端口镜像[9] 传送的目标流量数据,基于扩展的 Zeek 完成流量采集并生成详细的流量日志记录。在流量存储阶段,流量日志在监测服务器本地存储的同时,通过数据中间件将其转发到 ES 集群实现持久化存储。最后,搭建系统 Web 端统计与分析流量数据,为企事业单位用户提供实时观测业务流量最新态势与历史状态提供有效途径。

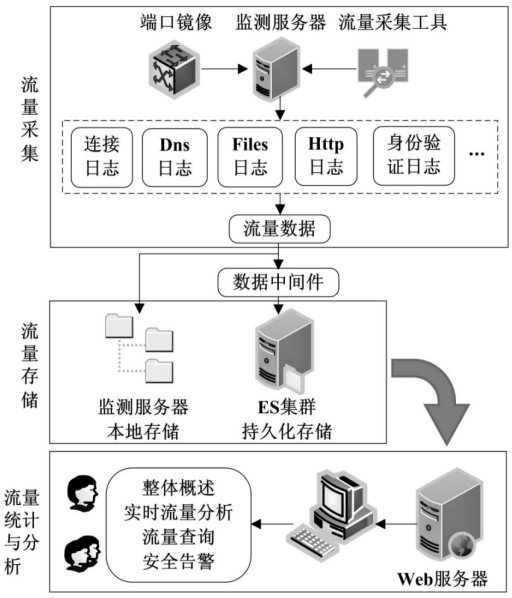


图 1 系统架构

2 系统关键技术实现

2.1 流量采集

2.1.1 系统流量采集需求提出

Zeek(NIDS)[10] 是由 Vern Paxson 开发的网络安全监控工具,采用基于网络流的被动流量采集方式,是业界内基于异常检测的入侵检测方法之一。Zeek 依靠由签名与规则组成的策略收集网络数据,具有广泛的协议支持,如 TCP/IP、HTTP、DNS、SSL/TLS、SMTP、POP3、FTP、SSH、文件传输协议,提供了更为详细和全面的协议感知和分析。Zeek 还允许用户定义和解析自定义协议,以适应特定环境或应用需要。由此,Zeek 生成一组丰富的网络活动日志文件[11],为企事业单位的计算环境提供了原始网络数据。同时,Zeek 支持分布式部署,配置由主节点与多个工作节点组成的 Zeek 集群,允许用户在多个节点上运行 Zeek 实例进行通信与数据共享,以处理大规模的网络流量和提高性能。该工具部署简单,适应高速网络环境,支持分布式部署,支持新的解析器[12-13],为用户根据其具体需求自定义网络流量分析和处理逻辑提供可能。

因此,该文结合 Zeek 抓包机制,同时考量现实流量采集现状困境产生的需求,提出了一种扩展的 Zeek 数据采集处理机制。具体如下:

(1)多端口识别。引入端口函数进行多端口识别,同时在连接日志中新增端口字段 interface,以实现轻量型区分多端口场景下的网络流量;

(2)自定义采集时间间隔。在长连接处理过程中,应用 HOOK 定义长连接处理策略,计算上一时间点与当前时间点内各字段的差值,并将其更新到连接日志中,为进一步提高时效性、缩短事件溯源误差时间提供支持。

在执行流量采集分析前,将流量采集点(通常为监测服务器)的网卡设置为“混杂模式”,接收到达本网卡的所有镜像流量。经 Zeek 接收处理后转发到存储集群中。所有增强功能配置于 *.zeek 文件中,于 local.zeek 文件中进行引用,重新加载脚本并重启生效。

系统流量采集需求阶段的流程如图 2 所示。

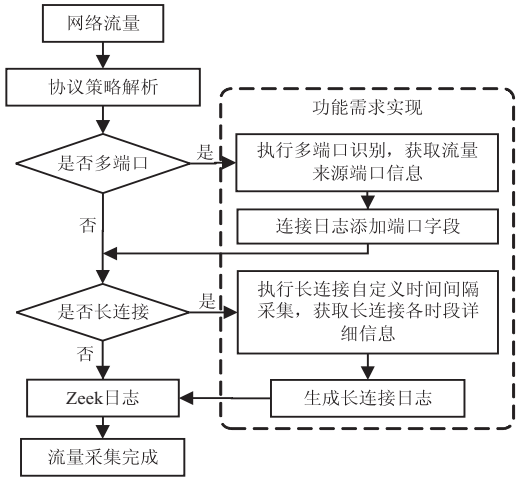


图 2　系统流量采集需求实现

2.1.2　多端口识别

在网络环境中,多端口服务器的管理很重要,对多端口进行准确识别和区分是开展业务活动的基础。Zeek 抓包形成的日志文件中未包含端口字段,面对多端口场景时,存在一个程序无法有效分辨流量来源的问题。为解决这一不足,可以选择同时启动多个 Zeek 程序,每个程序负责抓取特定的单一端口流量,但会伴随着大量内存资源的消耗。因此,为满足在单个程序下进行端口识别的需求,提高检索和展示便捷性,扩展多端口识别功能。具体实现如下:

```
module AddInterfaces;
export {const include_logs: set[Log::ID] = { Conn::LOG } &redef;};
type Addfields: recode { interface:string &log; }
function interface_ext_func(path: string): AddedFields{
if ( Cluster::nodes [ Cluster::node ] ? $interface )
return AddedFields ( $interface = Cluster::nodes[Cluster::node]$interface );
else
return AddedFields ( $interface = fmt("%s:unknown-interface", Cluster::node) );
}
```

上述代码的目的是为 Zeek 添加一个名为 AddInterfaces 的模块,其中包含一个常量和一个函数。定义的常量 include_logs 是一个包含 Log::ID 类型元素的集合,初始值为 Conn::LOG,即连接信息,用于将后续添加会话流接口的函数作用于 conn.log 连接日志。type 用于设置 interface 字段类型。定义的函数 interface_ext_func,用于获取集群节点的端口信息。Cluster::node 是一个返回端口标识的函数调用,$interface 为赋值的端口字段。如果端口信息存在,将该端口信息包装成 AddedFields 类型返回,否则构建一个包含节点名称和"unknown-interface"字符串的格式化字符串,并将其作为端口信息返回。

2.1.3　自定义采集时间间隔

在 Zeek 数据包抓取机制中,当数据包获取完成后写进日志,记录起始时间和时间间隔。在处理长连接时,直到长连接结束才进行日志写入。日常网络会话一定时间内结束,但实际网络环境中存在的特殊情况长连接,包含某些攻击手段、大文件上传下载或视频流的传输等,与日常网络会话耗时差别较大,不容易在同一时间维度内进行评判。为提高时效性,缩短采集与信息追踪上的误差时间,避免由于一些事件引起的长连接占用网络带宽甚至造成网络堵塞,但日志上溯源不到的情形,产生了自定义采集时间间隔进行流量采集的需求。具体实现如下。

```
Option LongConnection::default_durations = LongConnection::Durations(5sec);
Redef record Conn::Info += {
prev_orig_bytes: count &default=0;
}
Hook LongConnection::long_policy(rec: Conn::Info, if: Log::ID, filter: Log::Filter){
local temp = rec$orig_bytes;
rec$orig_bytes = rec$orig_bytes - rec$prev_orig_bytes;
rec$prev_orig_bytes=temp;
}
```

上述代码的目的是配置长连接解析策略。首先设置默认长连接持续时间,如果一个连接的持续时间超过预先设定的时长如 5 s,它将被认为是一个长连接。接着,基于长连接的定时写入期望实现的数据结果溯源主要体现在字节数、数据包数、时间间隔等字段上,因此,通过 redef 定位,实现增强结果体现在连接日志 conn.log 中。以字段源字节数为例,重新定义了 Conn::Info 记录,添加了一个名为 prev_orig_bytes 的字段,其类型为 count,默认值设置为 0。接着是核心模块,通过 HOOK 定义长连接策略。通过临时变量 temp 保存当前连接的源字节数(rec$orig_bytes),然

后更新 rec $ orig_bytes 为当前源字节数减去上一个时间点的源字节数(rec $ prev_orig_bytes)的差值。最后,将 rec $ prev_orig_bytes 更新为 temp,以备下一次计算使用。其他字段如目的字节、时间间隔等处理方法一致。同一个长连接的所有记录写在一个文件下,直到长连接结束,该长连接下的所有记录添加 flag 标识,用于指示当前长连接是否结束,未结束标识为 False,已结束标识为 True。

通过这种方式,可以观察到在当前流量采集持续时段内,长连接各个重要字段的变化情况。

2.2 流量存储

在流量监控系统中,高效存储获取到的数据直接关系到整个监控运行效率。在面对大流量背景下,采用传统的数据库存储方法,频繁的数据检索、写入和读取操作会引起的系统整体性能的下降。为优化系统存储效率,该文提出一种综合实现方法,该方法具体描述为:在本地使用轮询机制进行一周内数据的存储;在 Elasticsearch(ES)分布式存储上进行持久化存储,借助数据中间件完成数据传输,用于流量信息数据统计与显示。其中,配置 ES 集群所在机器的可承载空间时,确保至少满足 6 个月的流量数据,以符合《网络安全法》中相关规定要求。

2.2.1 本地轮询存储

本地存储的管理采用系统中的 logrotate 技术完成。Logrotate 技术可用于实现轮询检测,具体过程是采用 crontab 每隔一定时间检查特定的日志文件,如果文件满足约定条件,执行日志文件的旋转操作,按约定条件生成新的日志文件。在这个过程中,logrotate 还会记录可能发生问题的时间。该操作是轮转节省空间的实质,将节省设备日志占用空间和用户系统维护时间,提高系统的可用性与稳定性。本系统选择设置本地保留时间为一周,轮转周期设置为每天凌晨两点执行一次。

2.2.2 ES 分布式存储

所有经数据中间件解析后的流量数据汇聚形成完整的消息队列,将其存储到分布式实时搜索和分析引擎 Elasticsearch(ES)[14-15] 中。解析过程中可以充分利用企事业单位的资产识别与管理系统如 CMDB[16] 中的资产信息进行 IP 映射等操作,还可以构建由原/目的 IP 组成的数据流。ES 通过将数据索引和分片存储在多个节点上,每个节点都可以保存数据的一部分,这种分布式的设计提高了系统的可扩展性,还增强了数据的冗余性和容错性,为构建大规模、高性能的搜索和分析系统奠定了坚实的基础。ES 数据存储结构如图 3 所示。

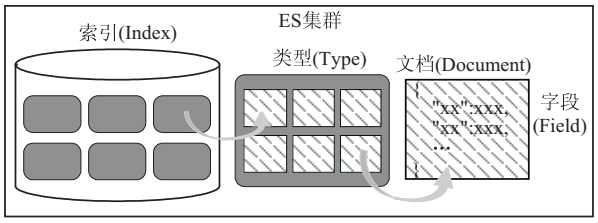


图 3 ES 数据存储结构

传输过来的数据通过按月标识的 ES 索引检索,该索引的记录格式如表 1 所示,其中各列记录了开展流量分析所需要的重要信息。可以选择设置保留周期为半年。

表 1 ES 中流量记录格式

列名	类型	说明
ts	date	时间戳
uid	keyword	连接的唯一标识符
interface	keyword	端口
id. orig_h	ip	源 ip
id. orig_p	short	源端口
id. resp_h	ip	目的 ip
id. resp_p	short	目的端口
proto	keyword	连接的传输层协议
conn_state	keyword	连接状态
flag	boolean	长连接状态标志
local_orig	boolean	连接是否来自本地
local_resp	boolean	响应是否来自本地
missed_bytes	interger	内容间隙中丢失的字节数
history	keyword	连接的状态历史
orig_pkts	interger	发起方发送的数据包数
orig_bytes	interger	发起方发送的字节数
resp_pkts	interger	响应者发送的数据包数
resp_bytes	interger	响应者发送的字节数

2.3 流量统计与分析

本系统设计 Web 端显控平台,接收来自 ES 数据源的流量数据进行个性化展示,为网络管理人员等用户提供全面了解网络通信情况、网络使用与访问情况、带宽被何种类型应用占用等的途径。同时还承担故障诊断的重要角色,用于当网络安全事故发生时,通过流量历史记录追踪与溯源情况[17-18],实现在最短时间内发现潜在安全威胁并迅速响应。

依据数据结构与需求分析,该系统设置的显控平台包含 3 类面板模块——总览管理、实时统计与 IP 查询,如图 4 所示。其中各端口分别对应一组实时统计与 IP 查询面板,总览管理面板为各端口的关键情况汇总。假设端口数为 n,则显控平台总面板数为 $1+2*n$。

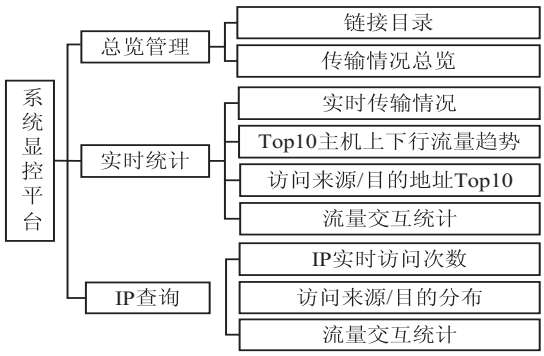


图 4　系统显控平台设计

(1)总览管理面板侧重整体网络环境的概述,支持通过在访问目录上添加其他面板访问链接,实现由总览管理面板到子面板的快速总分导航。

(2)实时统计用于管理流量数据实时变化趋势。在所有上下行流量传输趋势上,根据专线带宽上限设置显控平台告警与邮件告警,当流量达到带宽 80% ~ 90% 时设置二级告警提示,当流量达到带宽 90% 以上时设置一级告警提示。

(3)IP 查询和实时统计面板设置关联。实时统计面板发送自定义变量,由 IP 查询面板接收并展示传递过来的变量。在实时统计面板中,将流量交互模块的源/目的地址列设置成可以跳转的链接,通过{ $ Column}获取值并引入超链接 URL 中进行发送。接着,在 IP 查询面板中创建变量接收由实时统计面板传递过来的变量值,限定该面板仅展示专线下该 IP 作为变量的流量情况。在显示上,可跳转的变量字段较其他字段有一条下划线标识,以便用户明确识别。IP 查询功能实现了免输条件查询,为用户检索指定条件数据带来便捷,提高了平台的用户交互性。

2.4　系统评估

在系统功能实现过程中,使用监控覆盖率、时延与稳定性评估系统性能。本系统监测服务端接收镜像流量几乎不会出现丢包现象,监控覆盖率全面。增强 Zeek 的扩展功能自定义采集时间间隔提高了信息时效性,缩短采集与信息追踪上的误差时间,因此,从网络流量产生到被系统检测到并做出响应的时间间隔可控,表现为低时延。稳定性考量的是系统持续可靠地工作的能力,需要综合一段时间内系统运行效果再得出结论。

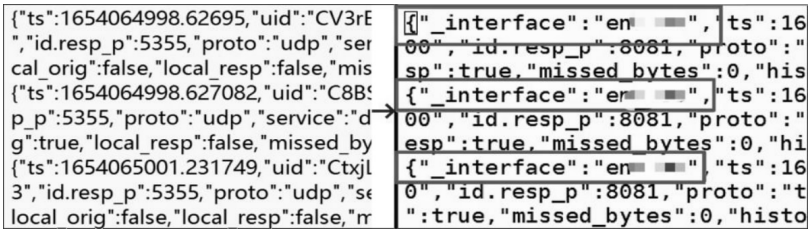
3　结果分析与展示

3.1　应用环境

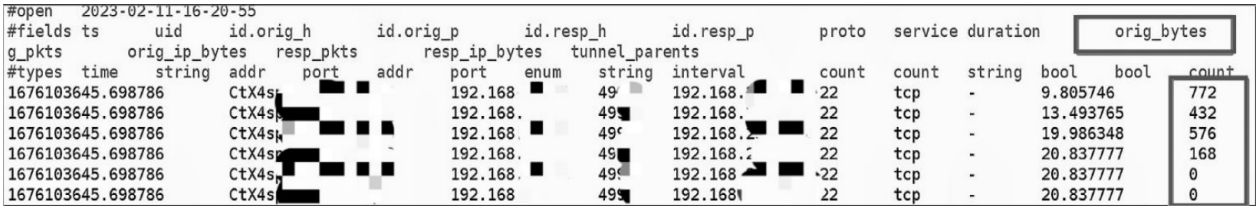
本系统部署旨在对某大型制造业企业 2023 年以来两地厂区间的专线流量交互情况进行个性化流量采集,依据采集数据做监测分析,为专线流量中发生应用服务访问卡滞等网络事件的流量溯源分析提供支持。流量监控设备采用 2.1 小节中设计的 Zeek 增强版本,部署在流量监测服务器上,获取流量镜像后传来的全部流量。流量分析展示系统部署在应用服务器上,服务器操作系统为 Centos7.9。

3.2　日志记录

图 5(a)、(b)分别展示了真实专线场景下使用标准 Zeek 和增强 Zeek 形成的日志数据。可以观察到后者日志中新增了 interface 字段;在长连接日志记录上,以 orig_bytes 为例,表现为标识相同、时间间隔等量递增、orig_bytes 为当前时段值的效果。



(a)多端口识别效果

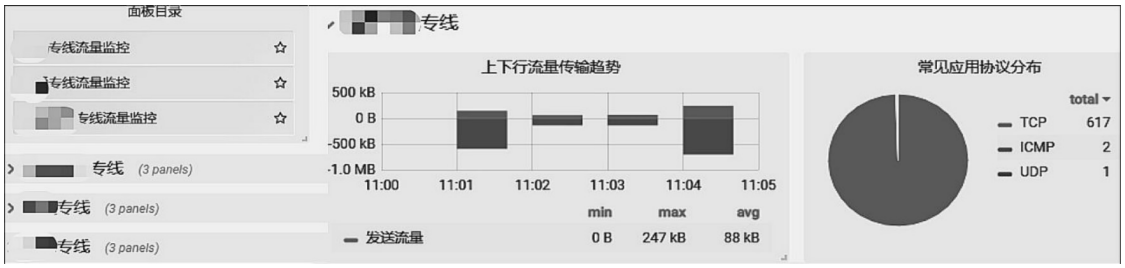


(b)长连接定时采集效果

图 5　增强的 Zeek 日志记录

3.3　总览管理展示

使用本系统开展某企业专线流量整体状况展示工作。在总览管理面板中,包含已添加的面板快速访问标签,同时汇总各流量专线的关键信息面板。用户可以实时查看各端口上下行流量的传输趋势与协议分布,同时支持用户通过目录跳转至详情页,无需返回上一级目录。其中各流量专线分流基于企业 CMDB 系统中业务与 IP、端口的对应关系完成(如图 6 所示)。

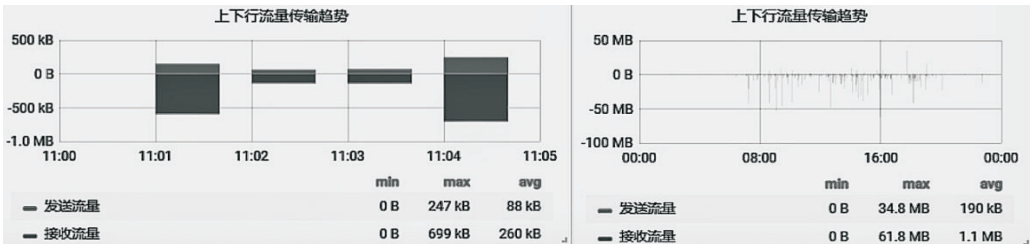


(a)面板汇总 (b)指定条线面板详情

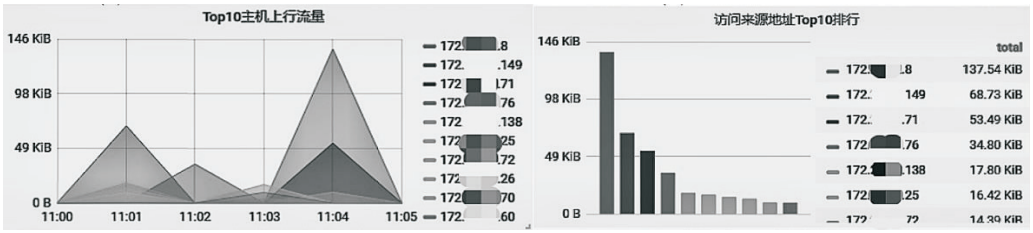
图 6 总览管理展示

3.4 实时统计展示

使用本系统开展某企业专线流量动态演变监控工作。实时统计展示设置为呈现用户关注的最近 15 分钟内的流量状况。用户可以查阅当前端口的上下行流量传输趋势、流量与地址 Top10 统计和累计流量交互。用户还通过灵活调节时间范围跨度,了解其他时间维度的流量变化。



(a)当前时间上下行流量传输趋势 (b)全天上下行流量传输趋势



(c)Top10 主机上行流量 (d)Top10 访问来源地址

上下行流量交互统计

源IP地址	目的IP地址	目的IP地址功能	上行流量	下行流量	上行流量平均	下行流量平均
172 .72	172. 53	邮件服务器	14.39 KiB	2.46 kB	7.19 KiB	1.20 KiB
172 .26	172 .62	邮件服务器	9.17 KiB	48.88 kB	9.17 KiB	47.74 KiB

(e)流量交互统计

图 7 流量实时统计展示

图 7(a)、(b)分别呈现了过去 15 分钟和全天时间内的实时流量上下行情况。图 7(c)、(d)分别展示了 Top10 主机上流量、源地址 Top10,图 7(e)展示了累计流量交互,表格由时间、数据流、源/目的端口、上/下行流量列构成。由图 7 可以得出:

(1)该厂区业务流量与一天内工作时间相关联,上午 8 点至下午 8 点工作时段内,厂区业务交互频繁,流量传输密切,较其他时刻流量数据明显增多;

(2)上下行流量传输保持在带宽范围内的稳定状态;

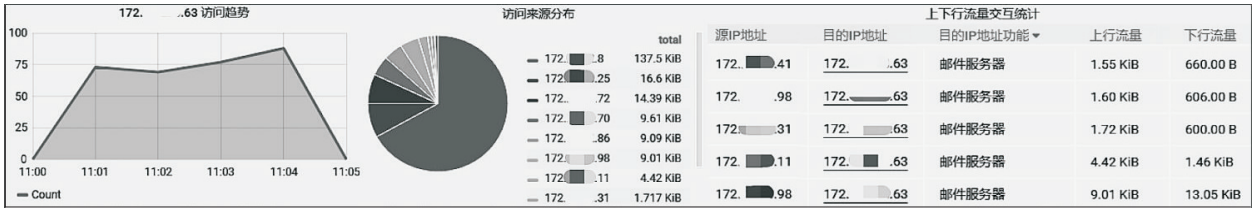
(3)流量目的地址集中在邮件服务器等已知的稳定资产范围内。

3.5 IP 查询展示

使用本系统开展某企业专线流量 IP 查询工作。IP 查询用于展示特定 IP 地址下的流量交互情况。图 8 展示了特定目的 IP 的流量交互情况,其中图 8(a)为不同时间点该目的 IP 被访问次数,图 8(b)展示了访问来源的分布情况,可以看出当前在线流量的访问来源为 172. x. x. 8,占据了 75% 以上分布,图 8(c)则提供了所有访问该目的 IP 的数据流详细情况,为用户提供了深入了解流量特征的视角。

本系统通过代码扩展多端口识别与自定义采集时间间隔功能,明确流量来源并降低流量信息时延。通过总览管理、实时展示和 IP 查询演示和追踪目标网络环境流量,感知企业网络环境安全态势。从信息安全管理体系安全策略来看,本系统提供了一种全面的安全防护机制,具备对网络安全事件的跟踪与防范能力,能够保护关键数据和业务不受损失。

自 2023 年投入运行以来,系统未出现崩溃失效情况,功能完好,系统稳定性良好,在企业安全运营中起到了关键作用,并拟计划向其他厂区拓展部署。



(a)指定 IP 被访问次数　　(b)指定 IP 访问来源统计　　(c)指定 IP 上下行交互统计

图 8　IP 查询展示

4　结束语

该文设计的系统实现了流量数据的统一管理。系统充分利用 Zeek 扩展性,通过实现多端口识别与自定义采集时间间隔,有效区分汇聚流量与降低监控时延,保证了安全事件溯源的高效性。同时,使用 ES 实现可持续存储,进行完善的显控平台设计,在 Web 端动态更新与分析数据,展示了所监控环境的网络安全态势。本系统已在企事业单位部署与验证,运行良好,流量数据管理有效,且能复用到其他系统上,是一个可靠的网络流量管理解决方案。

进一步的工作是结合机器学习手段,使用异常检测算法同步检测所捕获的流量,并通报异常流量。

参考文献:

[1] 徐　建,吴烨虹,程晶晶. 移动僵尸网络中数据流的采集与分析[J]. 计算机技术与发展,2016,26(11):101-105.

[2] Cisco. Cisco IOS NetFlow configuration guide[EB/OL]. (2011-07-20)[2024-01-26]. https://www.cisco.com/c/en/us/td/docs/ios/netflow/configuration/guide/12_2sr/nf_12_2sr_book/ios_netflow_roadmap.html.

[3] WALEED A, JAMALI A F, MASOOD A. Which open-source ids? snort, suricata or zeek[J]. Computer Networks, 2022,213:109116.

[4] NIKNAMI N, INKROTT E, WU J. Towards analysis of the performance of IDSs in software-defined networks[C]// 19th international conference on mobile ad hoc and smart systems (MASS). Denver:IEEE,2022:787-793.

[5] 陈吉荣,乐嘉锦. 基于 Hadoop 生态系统的大数据解决方案综述[J]. 计算机工程与科学,2013,35(10):25-35.

[6] 李天枫,姚　欣,王劲松. 大规模网络异常流量实时云监测平台研究[J]. 信息网络安全,2014(9):1-5.

[7] 彭　勰. 基于现场采集与云服务的流量积算管理系统研究[J]. 现代电子技术,2017,40(1):104-107.

[8] 李祥池. 基于 ELK 和 Spark Streaming 的日志分析系统设计与实现[J]. 电子科学技术,2015,2(6):674-678.

[9] 潘竹虹,许卓斌. 信息采集网络支撑系统的设计与实现[J]. 厦门大学学报:自然科学版,2016,55(3):426-433.

[10] Zeek. An open source network security monitoring tool[EB/OL]. (2024-02-18)[2024-02-18]. https://zeek.org/.

[11] CHOHRA A, SHIRANI P, KARBAB E B, et al. Chameleon: optimized feature selection using particle swarm optimization and ensemble methods for network anomaly detection[J]. Computers & Security,2022,117:102684.

[12] CHROMIK J, REMKE A, HAVERKORT B R, et al. A parser for deep packet inspection of IEC-104: a practical solution for industrial applications[C]//49th annual IEEE/IFIP international conference on dependable systems and networks - industry track. Portland:IEEE,2019:5-8.

[13] FARRAH D, DACIER M. Zero conf protocols and their numerous man in the middle (MITM) attacks[C]//IEEE security and privacy workshops (SPW). San Francisco:IEEE, 2021:410-421.

[14] 姚　攀,马玉鹏,徐春香. 基于 ELK 的日志分析系统研究及应用[J]. 计算机工程与设计,2018,39(7):2090-2095.

[15] 周　映,韩晓霞. ELK 日志分析平台在电子商务系统监控服务中的应用[J]. 信息技术与标准化,2016(7):67-70.

[16] 蔚周鹏,陈俊丽,张汉举. 基于 CMDB 的资产识别与管理系统设计与实现[J]. 计算机技术与发展,2023,33(12):106-112.

[17] 杨　彬,高俊涛,王志宝,等. 基于词嵌入的元组级数据溯源方法[J]. 计算机技术与发展,2023,33(12):49-57.

[18] 杨　波,徐胜超. 云网安全服务可视化监测方法[J]. 计算机技术与发展,2023,33(10):80-85.