文章编号:1671-4598(2023)04-0272-09 DOI:10.16526/j.cnki.11—4762/tp.2023.04.042 中图分类号:TP399 文献标识码:A

基于流式处理架构的日志采集系统的设计与实现

邵旭东[1]，樊志杰[1,3]，张敬锋[2]，曹志威[1]，周明富[3]，熊已兴[3]，张 林[3]
(1. 公安部第三研究所 信息安全技术部，上海 200031；
2. 安徽省公安厅 科技信息化处，合肥 230061；
3. 上海辰锐信息科技有限公司 研发中心，上海 200031)

摘要：对信息系统运行记录、操作日志、告警信息的采集问题进行了研究，提出了一种面向泛政府行业安全运行管理平台的统一日志采集系统；采用基于消息队列的流式处理架构，实现日志采集、日志处理、日志上报等各个环节的解耦；采用标准化接口和插件技术，实现各种异构日志信息的采集和数据上报；采用消息队列的流量削峰技术，保证日志传输的安全可靠；依据日志流量特征，提出一种支持动态调整消费组的设计模式，达到超过20 000条日志/每秒的高性能采集要求；整个系统由日志采集、数据上报、数据管理、系统管理、策略管理、Agent管理、日志源管理模块和日志采集代理（Agent）子系统组成，可满足对各类安全数据的集中分析、安全威胁感知和智能研判。

关键词：安全运行管理平台；标准化接口；插件技术；消息队列；流式架构

Design and Implementation of Log Collection System Based on Stream Processing Architecture

SHAO Xudong[1], FAN Zhijie[1,2], ZHANG Jingfeng[2], CAO Zhiwei[3], ZHOU Mingfu[3], XIONG Yixing[3], ZHANG Lin[3]
(1. Department of Information Security Technology, The Third Research Institute of the Ministry of Public Security, Shanghai 200031, China; 2. Science and Technology Information Detachment, Anhui Provincial Public Security Department, Hefei 230061, China; 3. Research and Development Center, Shanghai Chenrui Information Technology Company, Shanghai 200031, China)

Abstract: Aimed at the collection problems of operation record, operation log and alarm information of information systems, and a unified log collection system for the safe operation management platform for the pan-government and industry is proposed. The whole system is composed of log collection, data reporting, data management, system management, policy management, Agent management, log source management module and log collection subsystem. The flow processing architecture based on the message queue is adopted to realize the decoupling of log collection, log processing and log reporting. The standardized interfaces and plug-ins are used to collect the heterogeneous log information and data. The traffic peak clipping technology of message queue is adopted to ensure the safety and reliability of the log transmission. According to the characteristics of the log traffic, a supporting dynamic adjustment design modeling of the consumption group is proposed to meet the high collection requirements of over 20 000 logs per second. which can meet the centralized analysis, security threat perception and intelligent analysis of various security data.

Keywords: safe operation management platform; standardized interface; plug-in technology; message queue; streaming architecture

0 引言

随着世界范围内围绕信息的获取、使用、控制的斗争愈演愈烈，全球网络攻击、网络窃密和网络犯罪等问题日渐突出，网络安全问题已经成为与政治安全、经济安全、

收稿日期:2022-12-14； **修回日期**:2022-12-19。
基金项目:上海市人才发展资金资助(2020016)；中国博士后科学基金资助(2020M670998)；上海市自然科学基金资助(21ZR1422000)；公安部科技计划项目资助(2019JZX004)；四川省科技计划项目(重点研发项目)(2021YFS0310)。
作者简介:邵旭东(1976-)，男，浙江余姚人，硕士，副研究员，主要从事信息与网络安全方向的研究。
通讯作者:樊志杰(1982-)，男，山西朔州人，博士，研究员，主要从事信息与网络安全方向的研究。
引用格式:邵旭东，樊志杰，张敬锋，等.基于流式处理架构的日志采集系统的设计与实现[J].计算机测量与控制，2023，31(4)：272-280.

文化安全同等重要，事关国家安全的重大战略要害问题。泛政府行业大数据中心一旦发生安全问题，造成系统崩溃、网络瘫痪、病毒爆发或重要数据丢失、泄漏，势必导致灾难性后果，给泛政务业务系统正常运转、甚至国家信息体系完整性带来重大损失。为此，亟需建立以数据安全为核心的大数据安全保障体系，建立先进的安全运行管理平台、专业的安全技术团队、全面的安全策略以及高效的运行管理机制来保护大数据的安全。

安全运行管理平台汇聚的安全信息包括数据服务、应用、云平台、终端、边界、网络和安全保障设施等系统运行记录、操作日志、告警信息，各类系统运行信息格式不一、支持的传输协议种类繁多，因此需寻求技术手段，对各类异构的数据源进行统一采集和预处理，通过标准协议上报至安全运行管理平台[1-5]。

当前，行业内存在集中监管与审计系统，该系统为了保障泛政府行业信息安全，对接入泛政府行业内网的业务和用户进行监控，一旦发现安全隐患或攻击行为，立刻采取措施，保障泛政府行业内网的安全。该系统支持对以 TBSG 可信边界接入网关为隔离设备的边界接入平台、通过 VPN 网关接入的移动接入平台、通过视频接入专用设备的视频接入平台和以单向光闸为隔离设备的公网接入平台等多种类型平台上的通用网络设备、通用安全设备和专用安全设备的监控，以及设备状态、业务流量和用户访问日志等信息的采集，使得泛政府行业建设单位的管理人员可以方便地对接入平台进行管理，保证接入平台安全稳定地运行。但是该系统也存在一些不足之处，无法满足大数据安全体系下的监管要求，具体来说，存在的主要问题包括：

1）支持的协议类型较少，只支持 SNMP 和 SYSLOG 等少数日志采集协议，且扩展难度大；

2）系统将采集到的数据首先保存在关系数据库中，上报数据到其他平台时需要再次从数据库中读取、解析，时延大、执行效率低；

3）采集的日志只支持上报到单一的平台上，数据上报协议固定。为了支持多平台、多协议格式上报，扩展难度大；

4）系统为采用模块化的设计，模块之间耦合大，不方便扩展功能。

鉴于此，本文提出统一日志采集技术并研制系统，可解决集中监管与审计系统存在的不足：

1）采用标准化接口和插件技术，可灵活支持各种日志采集协议和数据上报协议，实现系统整体架构的高度稳定性和可扩展性；

2）采用基于消息队列的流式处理架构，实现日志采集、日志处理、日志上报等各个环节的解耦，并支持灵活的功能扩展；

3）通过消息队列的流量削峰保证了日志传输的可靠性；

4）根据日志流量特征，系统支持动态调整消费组规模，满足系统的高性能要求。

1　系统结构及原理

本文研制的统一日志采集系统，软件层面可以分为数据面与控制面两个部分，如图 1 所示。

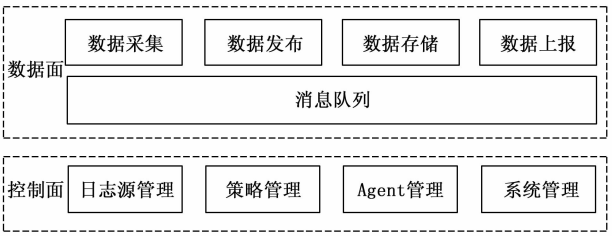

图 1　系统结构图

数据面主要包括数据采集、数据发布、数据存储与数据上报等功能，这些功能之间通过消息队列实现模块功能解耦和数据流式处理，方便功能的扩展。

控制面主要包括日志源管理、策略管理、Agent 管理、系统管理等功能，实现对数据面功能的策略配置与管控。

2　系统软件设计

本文所设计的统一日志采集系统主要由日志采集、数据上报、数据管理、系统管理、策略管理、Agent 管理、日志源管理模块和日志采集代理（Agent）子系统组成[6-10]，如图 2 所示。

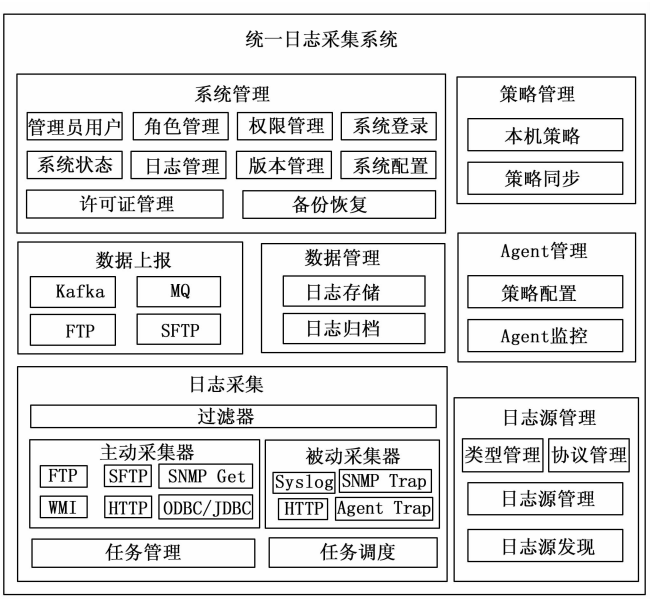

图 2　软件框架图

日志采集模块通过各种采集协议实现对各种数据源的日志采集功能。

数据上报模块支持通过 kafka、消息队列、SFTP、FTP 等协议将日志上报到各类平台。

数据管理模块实现日志数据的本地存储和查询功能。

系统管理模块实现系统登录、用户角色管理、权限管理、版本管理、备份恢复等功能。

策略管理模块实现日志采集策略的增删改查、同步等功能。

Agent 管理模块实现 Agent 策略配置、Agent 状态监控等功能。日志采集代理（Agent）可部署在 Windows 或 Linux 系统中，实现特定日志数据的采集上报，并接受统一日志采集系统的集中管理（如：策略配置、状态管理等）。

日志源管理模块实现日志源类型和采集协议管理以及日志源自动发现等功能。

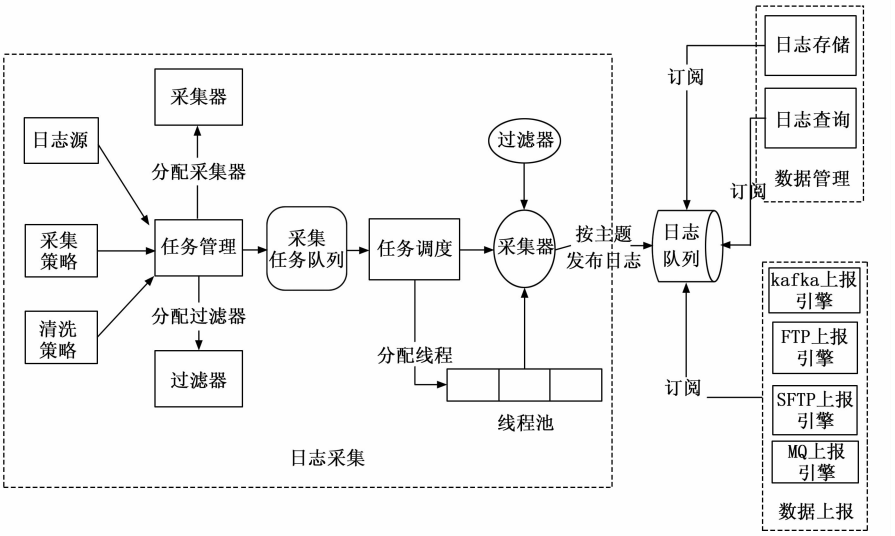

图 3 日志采集模块结构图

2.1 日志采集

日志采集模块由任务管理、任务调度、采集器和过滤器等组成，日志采集流程如图 3 所示。

任务管理模块负责生成采集任务，根据系统配置的日志源、日志采集策略和清洗策略，创建相应的采集器和过滤器，放到任务队列中等待执行。

任务调度模块负责读取采集任务队列，分配工作线程，调用采集器完成相应类型日志的采集，采集器通过调用过滤器完成日志数据的清洗，清洗后的日志数据按照主题发布到日志队列中。

采集器根据采集方式不同可以分为主动采集和被动采集。主动采集主要包括 FTP、SFTP、SNMP Get、WMI、ODBC/JDBC 等协议的日志采集。被动采集主要包括 SYSLOG、HTTP、SNMP Trap 等协议的日志采集。

对于主动日志采集是通过周期轮询的方式实现日志的采集，任务管理模块负责周期生成日志采集任务，并写入采集任务队列，等待任务调度模块调度执行。

对于被动日志采集，采集器采用监听服务，实现日志数据的持续接收，处理流程如图 4 所示。

过滤器负责根据系统配置的数据清洗策略对采集器获取到的日志数据进行过滤处理。根据不同的日志源类型和数据清洗策略，系统可实现多种类型的过滤器，多个过滤器可以通过组合模式形成新的过滤器。

本系统通过制定标准化的采集器接口和过滤器接口，对于各种数据源和日志类型，按照标准接口实现各类采集器和过滤器，并以插件的方式注入系统中，实现日志采集系统整体架构的稳定性和可扩展性，灵活支持项目的各种实际需求，如图 5 所示。

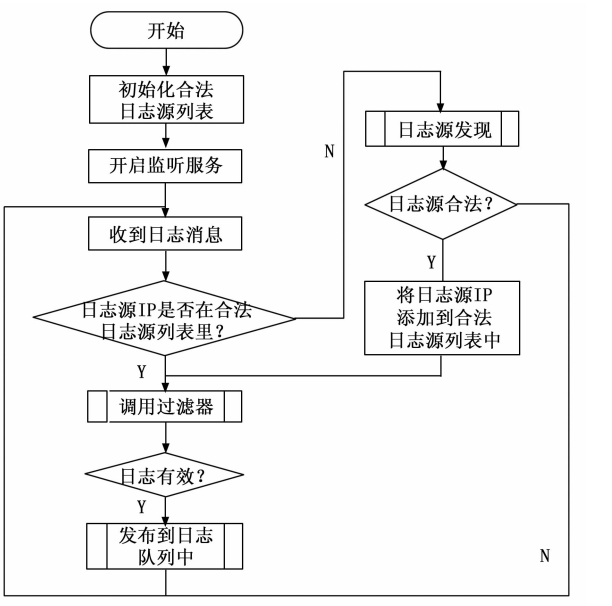

图 4 被动采集流程图

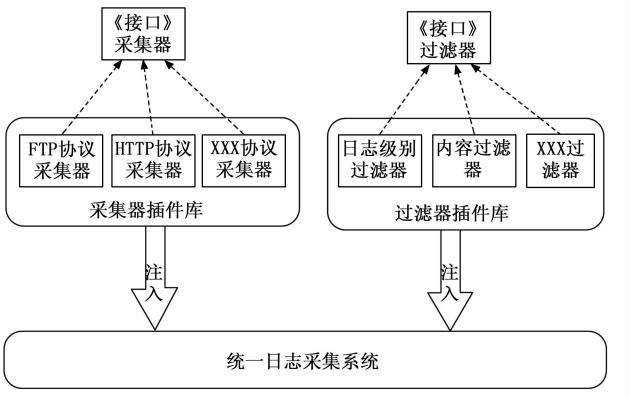

图 5 采集器和过滤器插件工作原理

2.2 日志数据发布

采集器通过各种采集协议采集到原始日志后，经过清洗、转换等处理后按照主题发布到日志队列中。

主题可采用如下格式：＜日志源类型＞.＜协议类型＞.＜日志级别＞.＜日志源 IP＞，方便数据存储、数据上报等模块按需进行订阅。

2.3　数据上报

数据上报模块支持通过 kafka、MQ、SFTP、FTP 等协议将日志上报到安全运行管理平台。

本系统通过制定标准化的数据上报引擎接口，对于各种上报协议方式，按照标准接口实现各类数据上报引擎，并以插件的方式注入系统中，实现日志采集系统整体架构的稳定性和可扩展性，如图 6 所示。

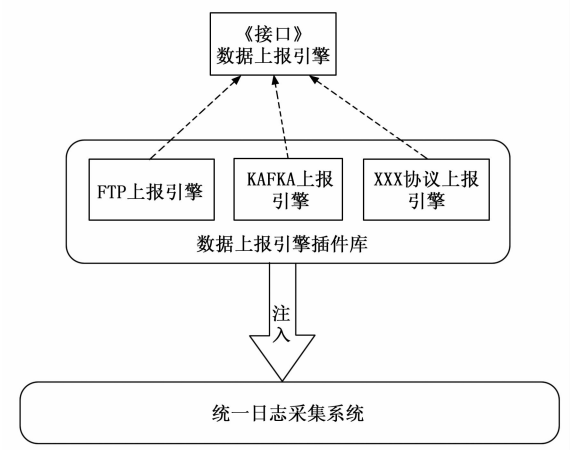

图 6　数据上报引擎工作原理

数据上报模块根据系统配置策略，加载指定的数据上报引擎，并创建独立的工作线程，按照主题从日志队列中订阅日志消息，将日志数据上报到安全运行管理平台。

系统通过监测队列中的积压消息数，动态调整工作线程数，实现日志消息传输的低延迟和可靠性。

对于 MQ、kafka 等协议上报方式，当接收端异常无法上报时，系统将产生告警日志，如果故障未恢复且 RabbitMQ 中的积压消息达到配置的最大门限值，为了防止内存耗尽，影响系统的整体性能和运行稳定性，系统将启动本地缓存机制，将日志接收并保存到本地文件中。

在上报故障期间，上报线程定期尝试将收到的新数据发送到 MQ 或 kafka，以检测其是否恢复故障，如果未恢复，则继续保存到本地文件中。

如果检测到接收端故障恢复，数据上报正常，则原上报线程将接收的新数据直接上报给 MQ、kafka。同时启动独立的线程负责读取保存在本地缓存中的日志信息并上报，上报成功后删除本地缓存文件。

对于 FTP、SFTP 等协议的处理方式，采集日志数据后先保存到本地文件中，然后通过另外的上传线程负责将本地文件上传到 FTP 服务器上。

2.4　数据管理

数据管理模块负责实现日志消息的本地存储、归档和检索等功能。

数据管理模块按照主题订阅日志，从日志队列中获取日志数据进行本地存储或者保存到全文搜索引擎中。

2.4.1　日志存储

对于日志存储，日志文件可按照年、月、日分级存储，即目录结构为：＜根目录＞\＜年＞\＜月＞\＜日＞\＜日志文件＞，这样既方便查找日志文件，又避免同一级目录下文件数过多，影响查询速度。

日志存储模块执行流程如下所示：

1）查询数据表获取系统配置的所有已使能的日志存储策略；

2）对于每条存储策略，创建一个线程，线程函数的执行逻辑如下：

（1）根据配置的订阅主题信息，依次订阅这些主题的日志消息；

（2）接收日志消息并保存日志到系统配置的文件存储路径下；

（3）根据系统配置的文件切割策略，完成日志文件的切割。

2.4.2　日志归档

日志归档模块根据系统配置的策略，将历史日志文件压缩后归档。归档日志可根据系统配置的保存时间定期清理。

日志归档模块执行流程如下：

1）启动周期定时器，每天在指定时间（如：凌晨 3 点）触发归档操作；

2）归档步骤如下：

（1）查询数据表，获取系统配置的所有已使用的日志存储策略；

（2）遍历每条日志存储策略，执行如下流程：

2.4.3　日志查询

日志查询模块通过订阅机制，获取日志数据保存到全文搜索引擎中，用于支持本地数据查询功能。

日志查询模块可根据系统配置的策略，定期清除过期数据。对于已过期的日志，系统提供日志文件下载功能。

日志查询模块执行流程如下：

1）查询数据表获取系统配置的所有已使能的日志查询策略；

2）对于每条策略，创建一个线程，线程执行逻辑如下：

（1）根据配置的订阅主题信息，依次订阅这些主题的日志消息；

（2）接收日志消息并保存日志到全文搜索引擎中，并制定相应参数，实现过期日志的自动删除功能。

2.5　策略管理

提供日志源、数据上报策略、日志存储策略和日志归档策略等管理功能，将配置信息保存到 MySQL 数据库，然后发布配置更新消息到 RabbitMQ 消息队列中，日志采集、数据上报和数据管理等模块通过订阅相应的配置更新消息，并按照新策略执行，流程如图 8 所示。

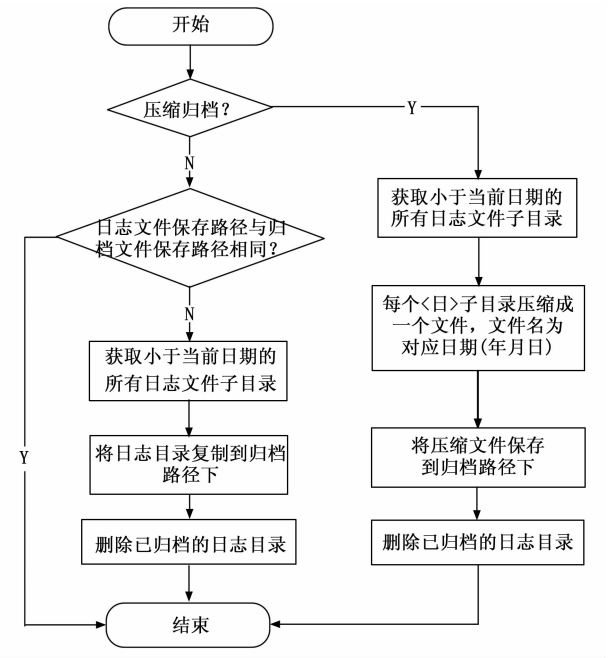

图 7 日志归档流程图

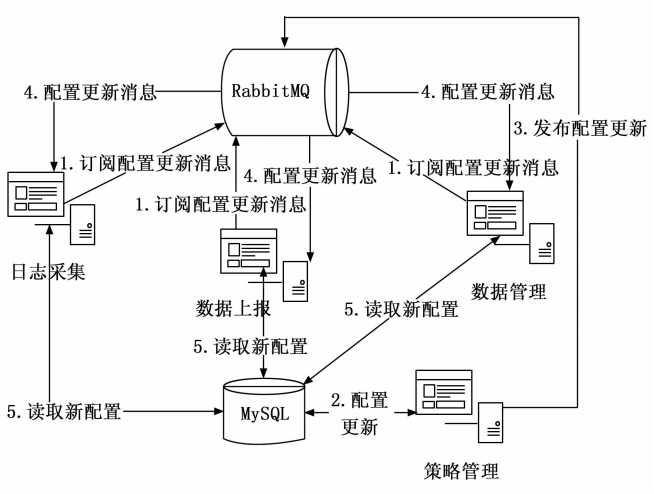

图 8 策略管理模块结构图

2.6 系统管理

系统管理模块提供用户管理、角色管理、系统登录、版本管理、备份恢复、系统状态监控等功能。

2.6.1 用户管理

用户管理主要完成用户的查询、新增、修改、删除、导入、登录 IP 限制和授权等操作。

2.6.2 角色管理

角色管理主要完成角色的查询、新增、修改、删除和授权等操作。

2.6.3 系统登录

平台提供登录页面，支持使用数字证书或用户名密码方式登录系统，平台提供配置方法，可以选择支持其中的一种或两种方式登录系统。为了提高系统的安全性，使用用户名密码登录时，支持字符验证码或手机短信动态验证码，使用哪种方式支持可配置。

用户登录流程如下：

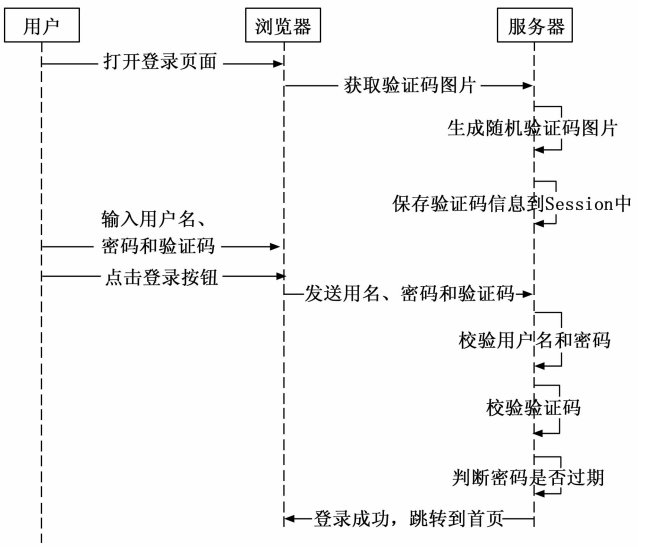

图 9 用户登录流程图

1）用户打开登录页面，向服务器请求一个验证码图片；

2）服务器生成一个随机验证码图片返回，并将验证码保存到用户 Session 中；

3）用户在浏览器中输入用户名、密码和验证码信息，并点击登录按钮；

4）服务器验证收到的用户名、密码和验证码是否正确，如果不正确提示登录失败并返回登录页面；等保测试下如果用户连续登录失败达到一定次数后，应锁定账户，记录安全审计事件；

5）如果当前为等保测试状态且设置了密码的最长使用期限，则判断密码是否过期，如果密码过期则跳转到用户密码修改页面；否则表示登录成功，跳转到首页。

2.6.4 版本管理

提供系统版本升级功能，为了简化升级流程，最大程度支持向下兼容，系统对软件部署和升级方式做如下约束：

1）只能从低版本升级到更高版本，支持跨版本升级，任何高版本升级包都能够升级低版本；

2）除了用到的一些系统级的配置文件，软件安装部署后要求其所有的程序文件、数据文件和配置文件等都在同一个根目录下，根目录下可以包含子目录。

系统升级流程如图 10 所示。

2.6.5 备份恢复

系统备份主要包括配置文件、数据文件和数据库的备份，其中对于数据库备份目前先实现全量备份。

备份时可以选择备份的内容：配置文件、数据文件和/或数据库，如果输入的备份密码为空，则表示备份包不做加密。

系统备份的执行流程如图 11 所示。

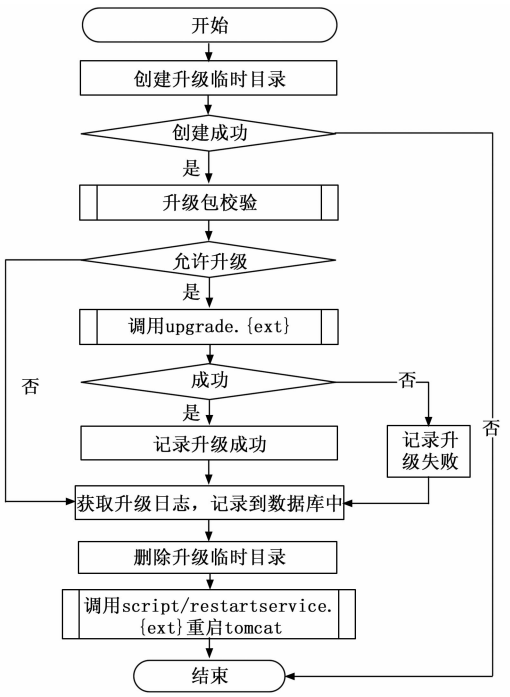

图 10　系统升级流程图

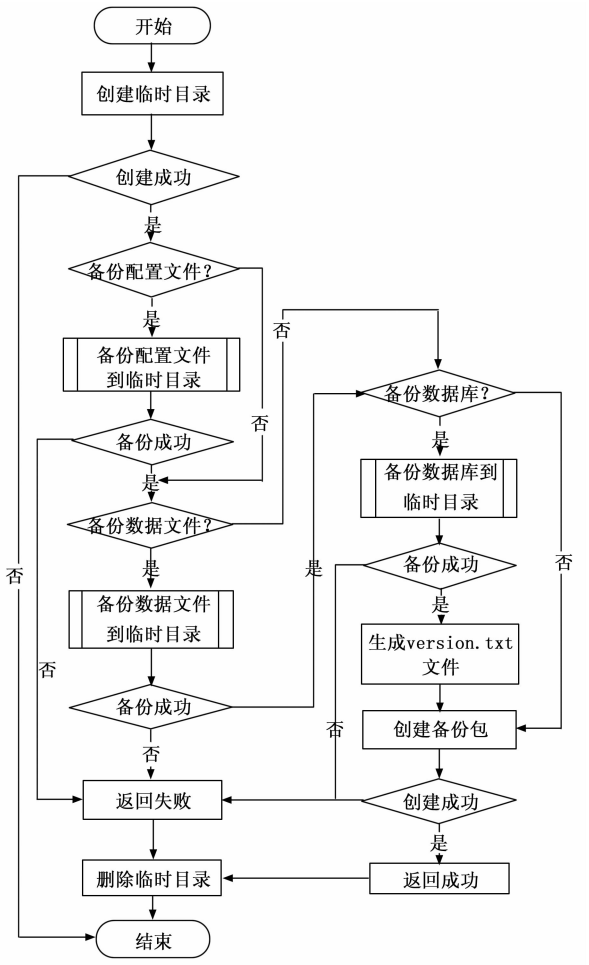

图 11　系统备份流程图

2.6.6　系统状态监控

系统状态监控模块通过向设备状态获取线程订阅的方式，获取服务器的 CPU、内存、磁盘利用率以及各个网卡接口的数据流量等信息，经过处理后保存到数据库中。

以图表的形式展现服务器的内存、磁盘、CPU 占用率以及网络接口流量信息。在首页和性能监控页面中都可以查看设备的状态，首页由于展示区域有限应以紧凑的形式进行展示，对于性能监控页面可以详细地分开展示设备的各种状态。

2.7　日志源管理

日志源管理模块负责管理维护各类日志源属性信息（如：系统名称、IP 地址、采集协议、认证凭据、日志源类型等），并实现日志源自动发现功能。

系统支持日志源自动发现功能。一旦被监控设备或系统有信息发送到统一日志采集系统，系统会根据信息类型和发送端 IP 地址等进行匹配，一旦发现新的 IP 地址在监控范围内，且监控设备数量不超过授权许可数量时，自动产生不同类型的新设备。新发现的 IP 如果发送的是 Agent 代理发送的事件信息，系统会自动生成一个主机类型日志源；新发现的 IP 如果发送的是其他类型信息，系统会自动生成一个未分类新日志源，需要管理员定义日志源信息。

3　系统主要功能

本文所研究统一日志采集系统的主要功能模块包括：日志信息采集、数据管理、数据上报、Agent 管理、日志源管理等功能[11-15]。

3.1　日志采集功能

支持各种数据源的运行信息和日志数据采集，按数据请求方向分为主动采集和被动采集两类。

主动采集：由统一日志采集系统主动向日志数据源请求获取日志数据，日志源设备被动响应请求回应数据，采集协议包括 SFTP、SNMP Get、WMI、HTTPS 等。

被动采集：统一日志采集系统支持特定协议监听，被动等待日志数据源传入数据，支持的采集协议有 SYSLOG、HTTPS、SNMP Trap 等。

系统支持的主要采集协议参见表 1。

3.2　数据管理功能

支持采集的数据本地分类存储，提供全文查询和文件检索。提供数据清洗、压缩、归档等数据管理维护功能。

3.3　数据上报功能

支持通过 kafaka、消息队列、SFTP、FTP 等协议将数据上报到安全数据系统、安全审计中心等外部日志分析处理平台。

3.3.1　本地缓存

在上报故障期间，上报线程应定期尝试将收到的新数据发送到 MQ、kafka，以检测其是否恢复故障，如果未恢复，则继续保存到本地文件中。

表 1 采集协议

采集方式	采集协议	说明
主动采集	FTP	通过 FTP 定时获取远程设备机器日志。
	SFTP	通过 SFTP 定时获取远程设备机器日志。
	SNMP Get	支持系统定时通过 SNMP GET 获取设备 CPU 占用率、内存占率、磁盘占用率、接口状态、当前连接数等性能信息，支持被采集设备对应 OID 配置。
	HTTP	支持定时通过 HTTP 协议从远程设备上下载日志文件。
	WMI	支持通过 WMI 协议获取 WINDOWS 信息。
	ODBC/JDBC	支持通过连接远端数据库获取数据事务日志。
被动采集	HTTP	支持 RESTAPI 方式接收日志。
	SYSLOG	接收日志源发送的 SYSLOG 日志。
	SNMP Trap	接收设备发送过来的 SNMP TRAP 日志。
	Agent Trap	接供 Agent 日志接收功能，包括 Windows、Linux 系统日志、进程、文件操作以及系统上运行的应用软件日志。

如果检测到接收端故障恢复，数据上报正常，则原上报线程将接收的新数据直接上报给 MQ、kafka。同时启动独立的线程负责读取保存在本地缓存中的日志信息并上报，上报成功后删除本地缓存文件。

3.3.2 跨边界级联

在跨各类边界进行日志信息采集时，边界外侧部署一套统一日志采集系统 1，采集各类设备运行日志信息，通过 FTP 协议将采集的数据通过边界设备传到边界内测，该功能同属数据上报模块，只需要新增一个 FTP 级联上报协议；边界内侧部署另一套统一日志采集系统 2，通过 FTP 协议采集外侧传入的文件并解析处理，该功能同属日志采集模块，只需要新增一个 FTP 级联数据采集协议。

3.4 策略管理功能

支持数据采集、数据处理、数据上报等策略的配置管理。

3.5 日志源管理

支持对各类日志源信息（日志源类型、系统名称、系统品牌、系统型号、软件版本、IPv4 地址、IPv6 地址、MAC 地址、采集协议、协议版本、认证凭证）的增加、删除、修改、查询、导入、导出等功能，支持日志源自动发现。

3.6 Agent 管理

设置 Agent 配置策略，包括系统的基本参数，如：日志采集间隔、本地日志保存时间、文件大小设置、系统登录密码、卸载密码等参数，服务设置了相关参数后，主机 Agent 自动同步该参数。

监控各主机 Agent 信息及运行状态，提供 Agent 管理操作日志、各模块的启停等运行信息查询、导出、归档等操作。

4 实施方式与应用案例

4.1 实施方式

日志采集、数据管理、数据上报和策略管理作为独立的模块进行部署。此外，系统还需要部署 RabbitMQ、ElasticSearch 和 MySQL[16-19]。统一日志采集系统部署框架如图 12 所示。

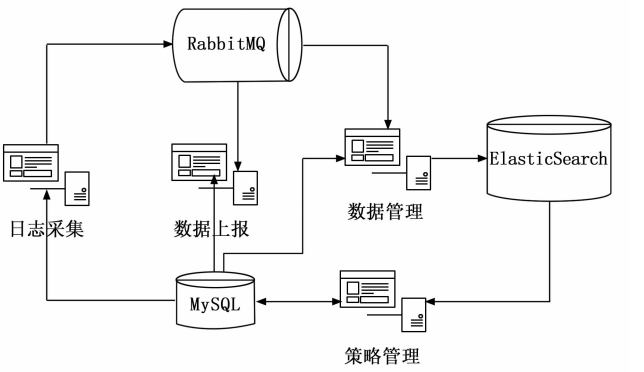

图 12 统一日志采集系统部署图

图 12 中的箭头表示它们之间的数据流向。日志采集模块从 MySQL 数据库中读取日志源、采集策略、清洗策略等配置信息，完成日志的采集后发布到 RabbitMQ 消息队列中。

数据上报模块从 MySQL 数据库中读取上报策略信息，然后从 RabbitMQ 消息队列中订阅指定主题的日志消息，按照相应协议完成日志上报。

数据管理模块从 MySQL 数据库中读取日志查询、日志存储和日志归档等策略信息，从 RabbitMQ 消息队列中订阅指定主题的日志消息，将日志数据保存到 ElasticSearch 全文搜索引擎或本地文件中，并对本地存储的日志文件进行归档处理。

策略管理模块负责完成日志源管理、数据上报策略配置、日志存储和日志归档策略配置等，将配置信息保存到 MySQL 数据库。系统管理模块通过查询 ElasticSearch 全文搜索引擎提供日志数据的查询功能。

4.2 应用案例

本文所设计系统已在多地泛政府行业部门进行实际验证，用于泛政府行业信息内网和新一代移动警务平台中应用、云平台、终端、边界、网络、安全基础设施的全面日志采集，具体方案如下：

4.2.1 新一代泛政府行业信息网部署

如图 13 所示，某省泛政府行业信息网按业务功能划分为用户接入和数据接入两大子网，以及子网和外部网络之间的用户访问和数据交换通道，为实现全面日志采集，需要在三个不同网域分别部署一套统一日志采集系统。其中数据中心采用云化部署模式，采集数据中心内主机、网络、

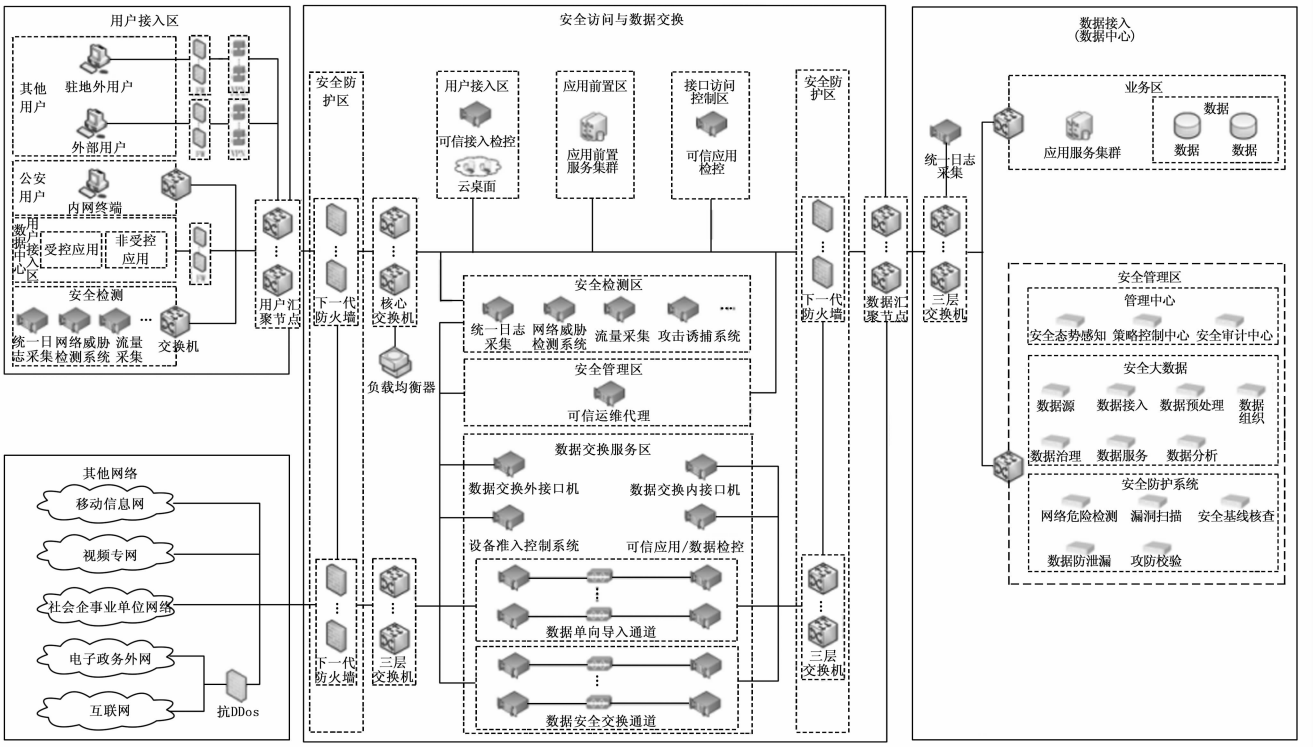

图 13　某省泛政府行业信息网统一日志采集系统部署拓扑

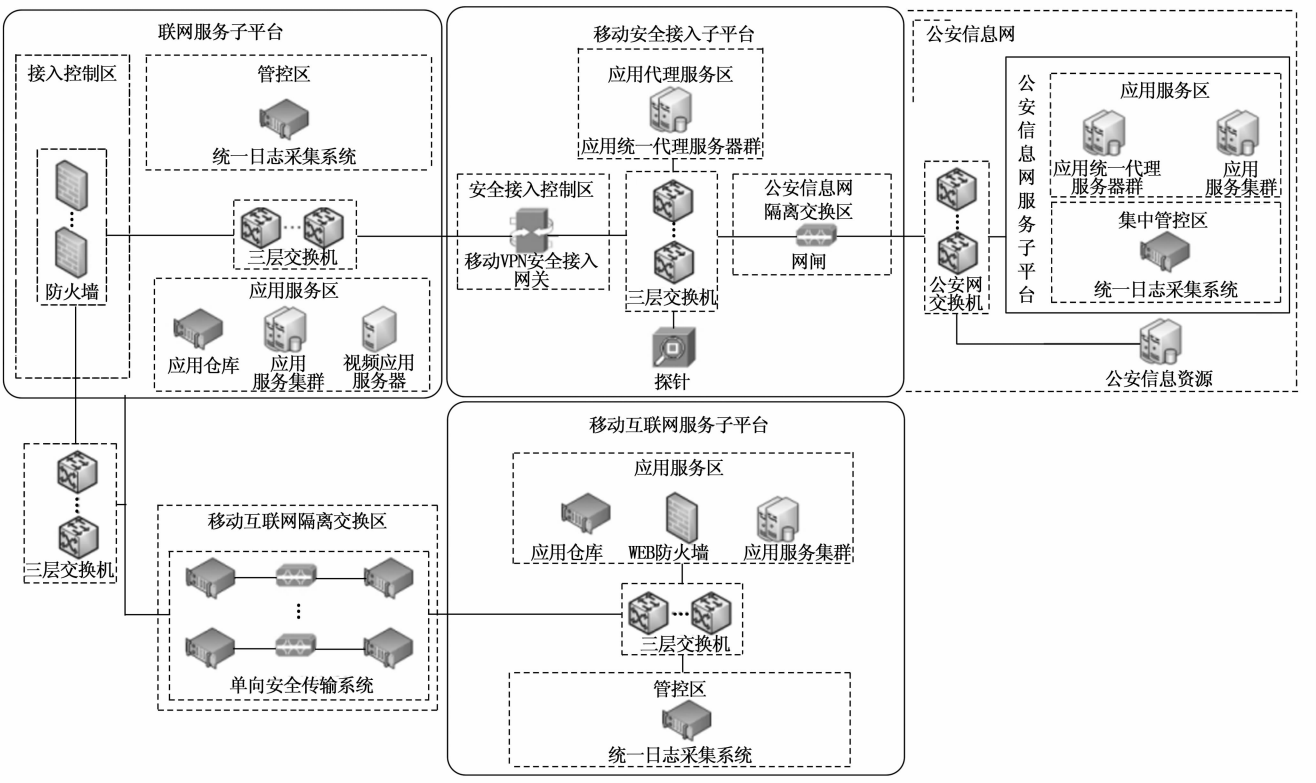

图 14　某省移动警务平台统一日志采集系统部署拓扑

安全设备和应用系统日志；用户接入区采用软硬一体部署模式，采集用户接入区内终端、网络、安全设备和应用系统日志；安全访问与数据交换通道采用软硬一体部署模式，采集网络、安全设备和应用系统日志[20-22]。

4.2.2 新一代移动警务平台部署

如图 14 所示，某省新一移动警务平台按业务功能划分移动互联网服务子平台（I 区）、联网服务子平台（II 区）、泛政府行业信息网服务子平台（III 区），以及移动安全接入子平台、移动互联网隔离交换区。为实现全过程日志采集，需要在三个不同子区域分别部署一套统一日志采集系统，均采用软硬一体化部署模式。其中 I 区统一采集系统负责，采集 I 区内主机、网络、安全设备和应用系统日志；II 区统一日志采集系统负责采集 II 区内终端、网络、安全设备和应用系统日志，同时采集两个隔离交换区内安全设备日志；III 区统一日志采集 III 区内网络、安全设备和应用系统日志[23-25]。

5 结束语

当前，全国泛政府行业均以习近平总书记“以安全保发展、以发展促安全”新时代网络安全思想为指引，牢固确立以人民为中心的发展思想，全面贯彻总体国家安全观，全面实施泛政府行业大数据战略，坚持大数据建设应用与安全防护同步规划实施，加快构建大数据安全保障体系。本文提出的统一日志采集系统，采用标准化接口和插件技术，全面提升泛政府行业大数据安全运行管理平台中日志采集的稳定性和可扩展性。同时，采用基于消息队列的流式处理架构，实现日志采集、日志处理、日志上报等各个环节的解耦，并支持灵活的功能扩展；通过消息队列的流量削峰，保证日志传输的可靠性，有效解决不同网域间海量日志数据的统一采集，进而为安全运行管理平台提供数据支撑、安全分析和智能决策。

参考文献：

[1] 沈昌祥，张大伟，刘吉强，等. 可信 3.0 战略：可信计算的革命性演变 [J]. 中国工程科学，2016，18 (6)：53 - 57.

[2] 张玉清，王晓菲，刘雪峰，等. 云计算环境安全综述 [J]. 软件学报，2016，27 (6)：1328 - 1348.

[3] 王晓妮，段 群. 基于云计算的数据安全风险及防御策略研究 [J]. 计算机测量与控制，2019，27 (5)：199 - 202.

[4] 樊志杰，郑长松，曹志威. 基于动态策略的移动警务终端安全管控系统的设计与实现 [J]. 计算机测量与控制，2021，29 (6)：219 - 223.

[5] 刘 凌. 公安专网下专用实验室视频监控系统设计与实现 [J]. 云南警官学院学报，2013，1 (1)：88 - 91.

[6] 刘敬浩，孙晓伟，金 杰. 基于主成分分析和循环神经网络的入侵检测模型 [J]. 中文信息学报，2020，34 (10)：105 - 112.

[7] 黄 蓉. 计算机网络安全与数据完整性技术探究 [J]. 网络安全技术与应用，2021 (4)：57 - 58.

[8] 王坤峰，苟 超，段艳杰，等. 生成式对抗网络 GAN 的研究进展与展望 [J]. 自动化学报，2017，43 (3)：321 - 332.

[9] LUO X，ZHOU M，LI S，et al. Incorporation of efficient second－order solvers into latent factor models for accurate prediction of missing QoS data [J]. IEEE Transactions on Cybernetics，2018，48 (4)：1216 - 1228.

[10] HAH H G，ZHANG H J，QIAO J F. Robust deep neural network using fuzzy denoising autoencoder [J]. International Journal of Fuzzy Systems，2020，22 (4)：1356 - 1375.

[11] 罗军舟，杨 明，凌 振，等. 网络空间安全体系与关键技术 [J]. 中国科学：信息科学，2016，46 (8)：939 - 968.

[12] 王 娜，杜学绘，王文娟，等. 边界网关协议安全研究综述 [J]. 计算机学报，2017，40 (7)：1626 - 1648.

[13] 周敏菲. 基于 Kafka 和 Storm 的实时日志流处理系统的设计与实现 [D]. 贵州：贵州大学，2017.

[14] 郭鹏程，李迎春，付春燕，等. 海量日志数据采集系统的设计与优化 [J]. 电子测量技术，2018，41 (1)：12 - 17.

[15] 逯 衍. 大数据平台数据采集系统的设计与实现 [D]. 北京：北京交通大学，2018.

[16] 辜家伟. 面向日志大数据分析的业务服务系统的设计与实现 [D]. 广州：华南理工大学，2018.

[17] 魏江涛，冯建峰，姜美雷，等. 船载系统日志自动分析软件的设计与实现 [J]. 计算机测量与控制，2021，29 (6)：142 - 146.

[18] 张智慧，吴 珏，杨福军. 基于模型结构和事件日志的流程相似度计算 [J]. 计算机测量与控制，2020，28 (3)：235 - 241.

[19] 王国林，介阳阳，叶君好，等. 基于日志挖掘的装备健康管理系统设计与实现 [J]. 计算机测量与控制，2018，26 (4)：112 - 115.

[20] WANG Y D，XUAN Z. Cross-device hand vein recognition based on improved SIFT [J]. International Journal of Wavelets Multiresolution & Information Processing，2018，16 (7)：1 - 17.

[21] 李锦川，钱秀槟，方 星. 基于国产操作系统的网络日志管理系统构建 [J]. 计算机安全，2010 (10)：59 - 61.

[22] CAO Q，SHEN H，GAO J，et al. Popularity prediction on social platforms with coupled graph neural networks [C] //Proceedings of the 13th International Conference on Web Search and Data Mining，Houston，Texas，UAS，2020：70 - 78.

[23] AKBANOV M，VASSILSKOS V G，LOGGOTHETIS M D. Ransomware detection and mitigation using software-defined networking：the case of Wanna cry [J]. Computers & Electrical Engineering，2019，76：111 - 121.

[24] ZHAO P，FAN Z，CAO Z，et al. Intrusion detection model using temporal convolutional network blend into attention mechanism [J]. International Journal of Information Security and Privacy，2022，16 (1)：1 - 20.

[25] FAN Z，CAO Z. Method of network intrusion discovery based on convolutional long-short term memory network and implementation in VSS [J]. IEEE Access，2021，9：122744 - 122753.