Proceedings of the 2nd International Conference on Software Engineering and Machine Learning
DOI: 10.54254/2755-2721/77/20240666

Design and implementation of real-time log analysis based on ELK

Bin Lyu

Kanazawa University, Kanazawa, Japan

q873541969@gmail.com

Abstract. With the rapid development of the IT industry, the integrated solution of real-time log analysis has become a member of the new generation of IT operation and maintenance structure. The traditional manual log analysis system is no longer suitable for the rapid development of the operation and maintenance system. The log analysis software commonly used in enterprises includes Elasticsearch, Logstash, and Kibana systems. The advantage is that it is free and open source.

"Design and implementation of real-time log analysis based on ELK" was born for system maintenance and developers to understand server software and hardware information through logs, and check the cumbersome operations in the configuration process. Because frequent analysis of the log can understand the load of the server, performance security, so as to take timely measures to correct errors. If you manage dozens or hundreds of servers, there is no centralized log management system to collect and summarize the logs on all servers, which causes great difficulties for operation and maintenance managers. "Design and implementation of real-time log analysis based on ELK" uses Elasticsearch, Logstash, and Kibana to achieve unified collection and classification of server logs, which can solve the defects of traditional log analysis platforms, improve the efficiency of server application positioning problems, and help enterprises solve data center problems running problems of related applications.

Keywords: Elasticsearch, Logstash, Kibana, Redis.

1. Introduction to Relevant Technologies

With the rapid development of the IT industry, integrated solutions for real-time log analysis have become a part of the new generation of IT operations and maintenance structures. Traditional manual log analysis systems are no longer suitable for the rapidly evolving operations and maintenance system. Commonly used log analysis software in enterprises includes the Elasticsearch, Logstash, and Kibana (ELK) system, which is advantageous for being free and open-source. Utilizing distributed technology can also reduce investment costs for enterprises. Therefore, most domestic and international enterprises choose to deploy real-time log analysis platforms based on the ELK architecture to monitor internal service logs. In summary, the ELK real-time log analysis platform is relatively mature and stable in the global technology ecosystem.

This "Design and Implementation of Real-time Log Analysis Based on ELK" primarily investigates how to implement a high-availability, distributed, real-time log analysis platform based on the ELK for existing enterprise log analysis systems. Accurate trend recognition for small and medium-sized

© 2024 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2755-2721/77/20240666

enterprise data center servers specifically means identifying alarm trends in these data centers. It also involves tracking alarms through trend recognition in small and medium-sized enterprise data center servers. Based on the foundational environment of the enterprise architecture, it is advisable to configure an Elasticsearch, Logstash, and Kibana cluster deployment to reduce performance bottlenecks caused by heavy log reading and writing. The ultimate goal is to establish a centralized log analysis platform that enables enterprise data center operations personnel to understand the load, performance, and security metrics of data center servers through log analysis operations. Moreover, enterprise data center operations personnel can also identify issues within the data center and take timely measures to correct them.

2. System Design

2.1. System Design Technical Solution

The subject mainly studies the application of the real-time log analysis system based on ELK for enterprises. Through the VMware Workstation software, this graduation project's experimental platform is constructed, simulating the continuous integration architecture of a real enterprise data center. The project involves the integration and validation testing of a real-time log analysis system based on ELK with Redis for log analysis recording. The test results determine whether the design meets the project requirements. The overall system design deploys Elasticsearch, Logstash, and Kibana in a distributed ELK format, using the Redis service under Linux to implement data caching for the enterprise-level data ELK log analysis platform.

2.2. System Design

2.2.1. System Topology Diagram

"The Design and Implementation of Real-time Log Analysis Based on ELK" primarily investigates how to implement a high-availability, distributed, real-time log analysis platform based on the existing log analysis system architecture of enterprises. The main content of the study is how to implement a basic business backend log analysis architecture through the ELK architecture, and subsequently, utilize high-availability and distributed technologies to achieve precise real-time log analysis functionality. Ultimately, on this enterprise architecture basis, it is preferable to configure a cluster deployment of Elasticsearch, Logstash, and Kibana to mitigate performance bottlenecks caused by heavy log read and write operations.

The required project planning is as shown in Table 1. Ela01, Ela02, and Ela03 serve as the log analysis platform's search servers based on Lucene, deployed in a cluster mode for this design. Elasticsearch software and Elasticsearch_head_master plugin are deployed on top. Logstash servers deploy Logstash software, an open-source data collection engine with real-time pipeline capability that can dynamically unify data from different sources and standardize it to the selected target output. Kibana offers numerous plugins that assist operations personnel in parsing, enriching, transforming, and buffering any type of data.

Table 1. Project Planning

Server	IP Address	System Version	Remarks
Ela01	192.168.120.200	CentOS 7 1810	Ela01
Ela02	192.168.120.201	CentOS 7 1810	Ela02
Ela03	192.168.120.202	CentOS 7 1810	Ela03
Logstash	192.168.120.203	CentOS 7 1810	logstash
Kibana	192.168.120.204	CentOS 7 1810	kibana
Redis	192.168.120.205	CentOS 7 1810	redis
MySQL	192.168.120.206	CentOS 7 1810	mysql
Filebeat_nginx	192.168.120.207	CentOS 7 1810	Filebeat_nginx

DOI: 10.54254/2755-2721/77/20240666

Relevant virtual machines are created using VMware Workstation software, which includes virtual machines for Ela01, Ela02, Ela03, Logstash, Kibana, Redis, MySQL, and Filebeat_nginx based on the Jenkins automated continuous integration system. These devices are configured accordingly to support their roles in this project. Memory size is set to 2GB, CPU hardware-assisted virtualization is enabled to minimize resource waste caused by instruction set conversion, and other resources are dynamically allocated based on actual conditions.

2.2.2. System Configuration Setup

In the servers Elasticsearch01, Elasticsearch02, and Elasticsearch03, configure the firewall, SELinux, network, and basic software environment. On this foundational environment, install Elasticsearch, Node plugins, and Elasticsearch Head plugins, and finally configure the Elasticsearch cluster. Implement the log collection function of the Elasticsearch cluster, which can also be linked through distributed technology.

In the Kibana server, configure the firewall, SELinux, network, and basic software environment. On this base environment, perform a source code installation of Kibana and configure it to interface with the Ela cluster. Achieve log collection of the Elasticsearch cluster and interfacing with Kibana, which can also be linked through distributed technology.

In the Logstash server, configure the firewall, SELinux, network, and basic software environment. On this base environment, perform a source code installation of Logstash and configure it to interface with both Ela and the Redis cluster. Implement Logstash's log categorization functionality, which can also be linked through distributed technology.

In the Filebeat & Nginx server, configure the firewall, SELinux, network, and basic software environment. On this base environment, perform a source code installation of Filebeat and Nginx, and configure Filebeat & Nginx to interface with the Redis cluster and Logstash services. Achieve log capture functionality of the monitored servers, which can also be linked through distributed technology.

In the Redis server, configure the firewall, SELinux, network, and basic software environment. On this foundational environment, perform a source code installation of Redis service and configure a Redis distributed cluster that interfaces with MySQL and Logstash services. Achieve log capture functionality of the monitored servers, which can also be linked through distributed technology.

With this, the construction of the real-time log analysis system based on ELK is completed.

3. System Testing

3.1. Test Environment and Content

Hardware Environment: A host computer configured with: Windows 10 Professional, Intel(R) Core (TM) i5-9850H @ 2.61GHz processor, 32GB installed memory, and a 64-bit operating system based on an x64 processor.

Software Environment: VMware® Workstation 15 Pro (15.2.1 Build-17510642).

Test Content: To determine whether the real-time log analysis platform based on ELK can analyze, categorize, and summarize the logs from Nginx, and persistently store the information in a MySQL database.

3.2. System Test Results and Analysis

3.2.1. Elasticsearch Cluster Test

Start the Elasticsearch cluster (Elasticsearch01, Elasticsearch02, Elasticsearch03), and activate the Head component of the Elasticsearch cluster. Artificially shut down the Elasticsearch01 server to simulate a single node failure of the cluster as shown in Figure 1.

DOI: 10.54254/2755-2721/77/20240666

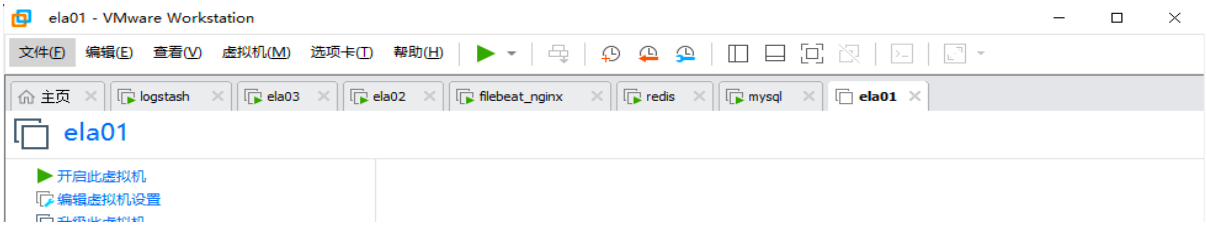

Figure 1. Elasticsearch01 Fault

Following the simulated fault of Elasticsearch01, the remaining nodes in the cluster, Elasticsearch02 and Elasticsearch03, seamlessly continue to provide the log collection functionality of the Elasticsearch cluster, without affecting the normal operation of other backend servers, as shown in Figure 2.

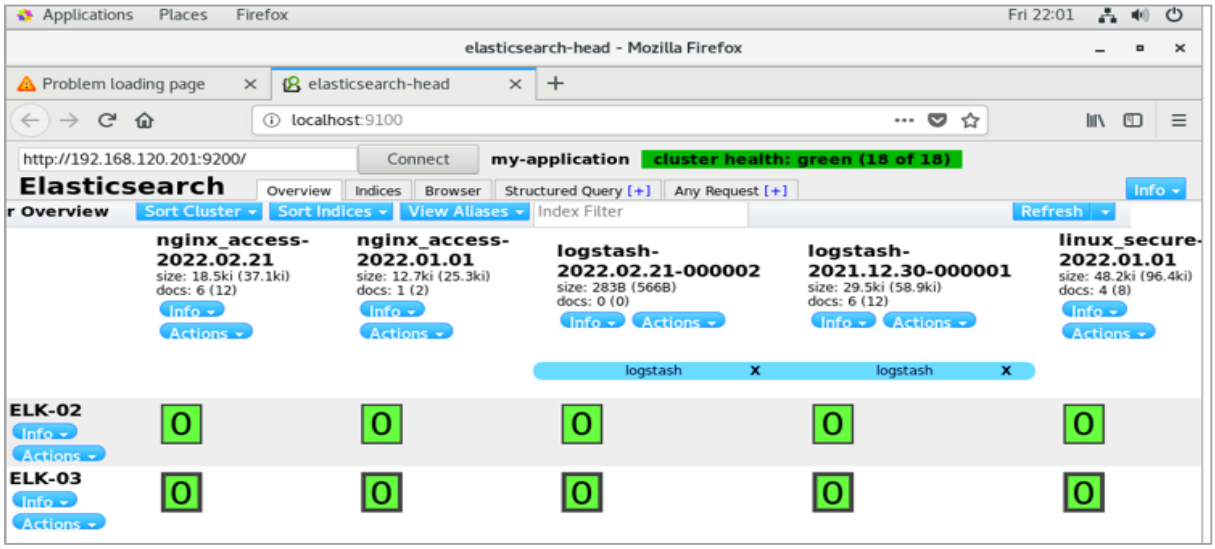

Figure 2. Elasticsearch Cluster Operating Normally

3.2.2. Comprehensive Project Test

Start the Elasticsearch cluster (Elasticsearch01, Elasticsearch02, Elasticsearch03), and access its frontend Head interface at 192.168.120.200:9100, 192.168.120.201:9100, 192.168.120.202:9100; this access is local as the server is Localhost:9100, as shown in Figure 3.

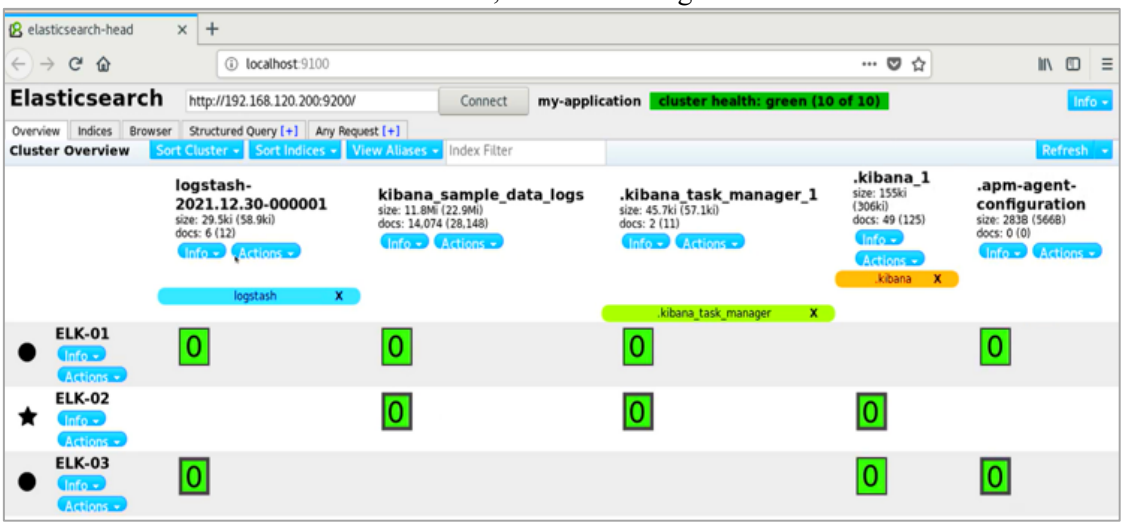

Figure 3. Elasticsearch Head

DOI: 10.54254/2755-2721/77/20240666

Sequentially start Redis, Kibana, Logstash, MySQL, and the Filebeat service on the target end Nginx server. Access the Kibana frontend, and by clicking on the Nginx option, it outputs information pulled from the Nginx log files, as shown in Figure 4. Kibana has successfully connected to the Elasticsearch cluster and completed the startup steps.



Figure 4. Kibana Started

In the Logstash server, Logstash has successfully connected to the Redis cluster and has already collected data sent from the Filebeat on the Nginx server, as shown in Figure 5.

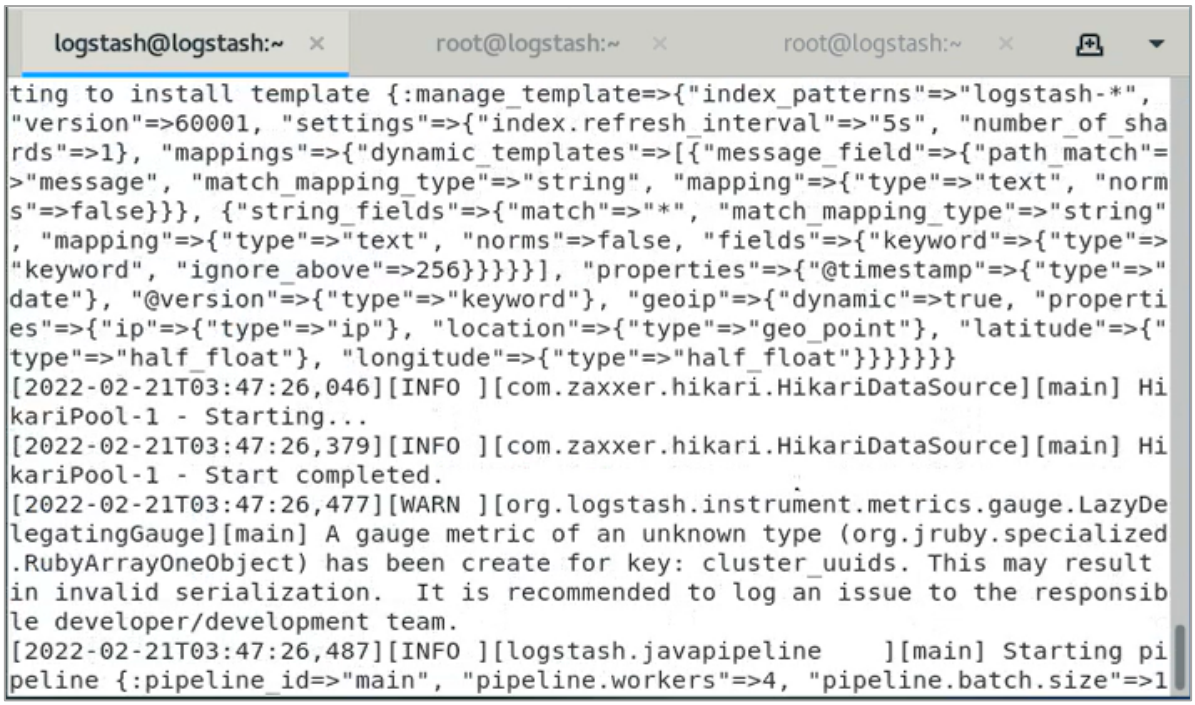

Figure 5. Logstash Started

DOI: 10.54254/2755-2721/77/20240666

On the Nginx_Filebeat server, Filebeat has collected log information from the Nginx server and transmitted it to the Redis server for data caching; while caching, the data is also written to backend servers, as shown in Figure 6.

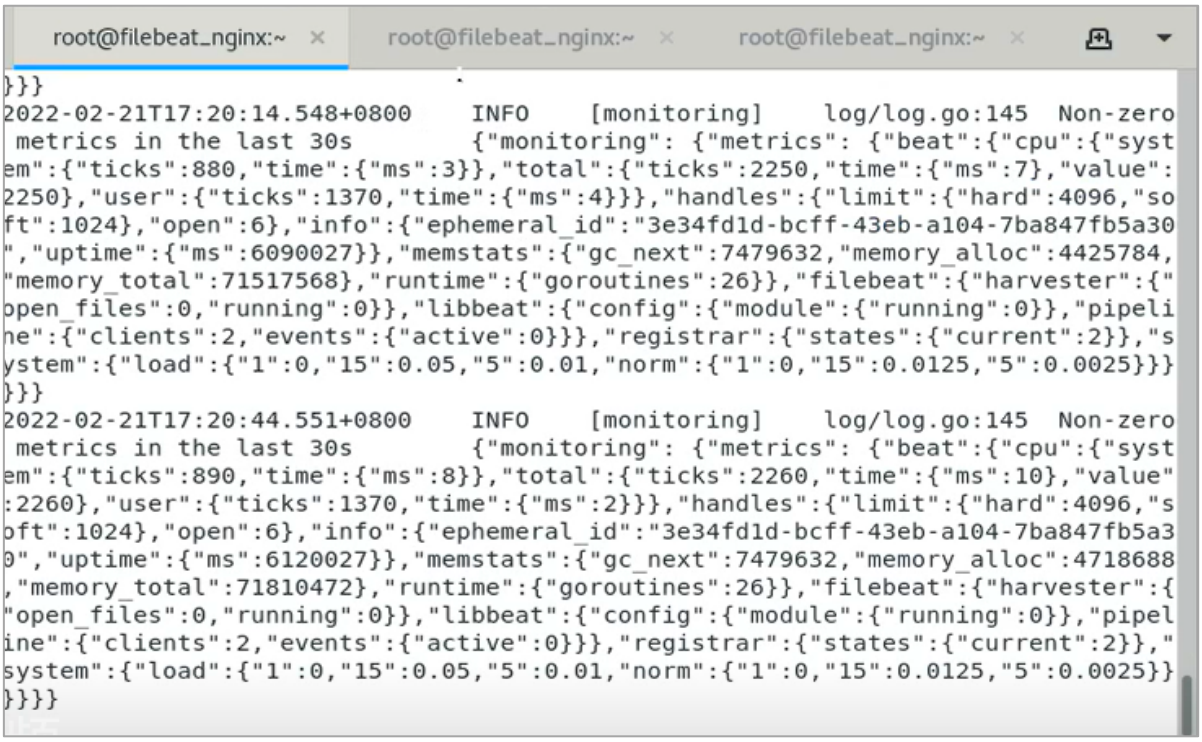

Figure 6. Filebeat Started

In the Redis server, it is possible to query the data marked "Filebeat" that has been collected from the Nginx server and cached, as shown in Figure 7.



Figure 7. Redis Started

Access the Kibana frontend address, click on the Nginx option, and it outputs information pulled from the Nginx log files. Clearly and precisely output some information about the Nginx server, such as: the client IP addresses accessing the Nginx server, the status codes returned by the backend Nginx server, the response times, and the client browser information.

DOI: 10.54254/2755-2721/77/20240666

Upon accessing the Kibana frontend address and clicking on the Nginx option, the backend Nginx server's access time, client IP address, result, status, and accessing agent are analyzed and summarized, as shown in Figure 8. The data collected by the Filebeat service includes specific entries, such as on March 20, 2022, at 10:55:16, when the Kibana server accessed the backend Nginx server and obtained relevant data; another example is on March 20, 2022, at 10:53:31, when the Windows 10 host accessed the backend Nginx server and obtained relevant data.

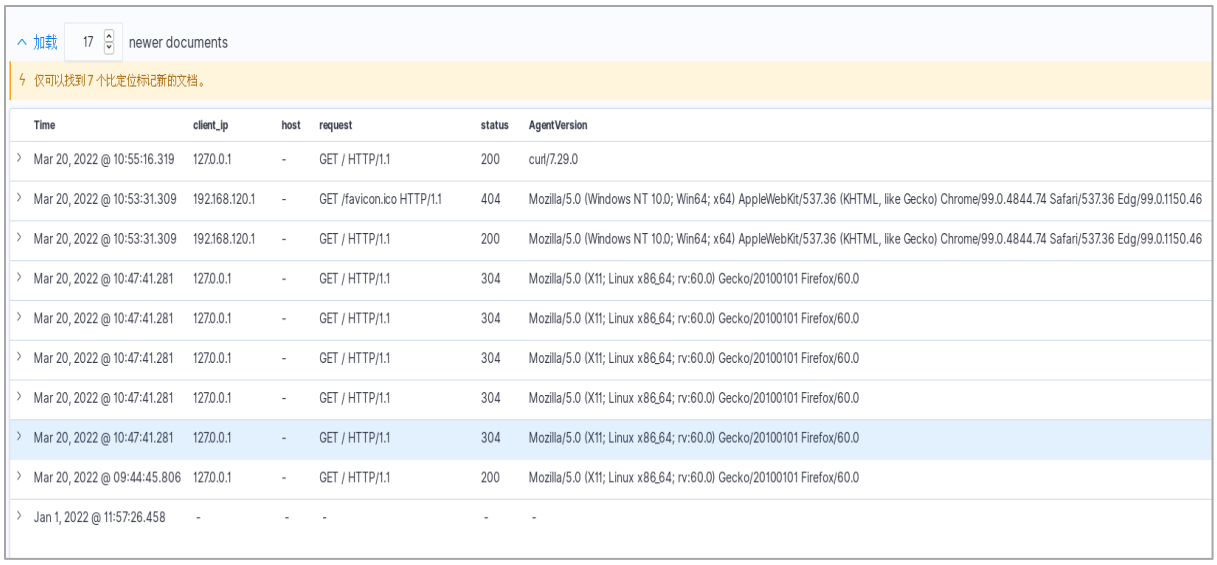

Figure 8. Kibana Outputs Analysis of Nginx Server Logs

After accessing the Kibana frontend address, further detailed categorization and analysis of the filtered data from the backend Nginx server concerning access time, client IP address, result, status, and accessing agent are performed. The data accessed using the Curl command to the backend Nginx server is filtered and statistically analyzed, as shown in Figure 9.

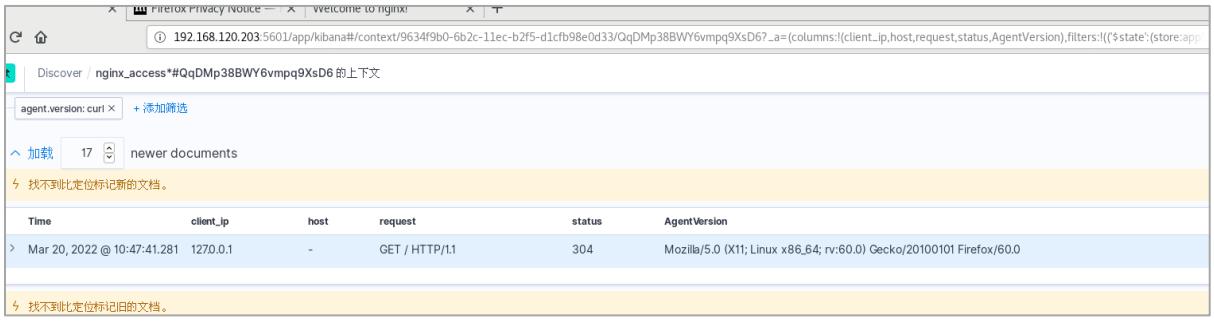

Figure 9. Output of Records Accessed Using Curl Mode

Test Results: The real-time log analysis platform based on ELK has achieved the functionality of analyzing Nginx log files, and it can cache firsthand information from the ELA cluster to the Redis server, while the backend MySQL server can record these details, meeting the requirements of the project task.

3.2.3. Test Analysis

In conclusion, the real-time log analysis platform based on ELK has successfully implemented the analysis of Nginx log files, and can cache firsthand information from the ELA cluster to the Redis server,

DOI: 10.54254/2755-2721/77/20240666

with the backend MySQL server capable of recording these details. This preliminarily fulfills the project's requirement for real-time log analysis.

4. Conclusion

From the perspective of traditional log analysis system architectures, the "Real-time Log Analysis System Based on ELK" utilizes distributed technology and high availability to avoid the flaws of single-point failures and the difficulty of dynamic expansion. In the technical architecture of enterprise log analysis, configuring appropriate Kibana templates can ensure the efficiency of quick problem localization. Cluster deployment in critical areas of the Elasticsearch, Logstash, Kibana architecture reduces performance bottlenecks caused by extensive log reading and writing.

The integration and development of computer technology and communication technology have led to the rapid development of the application and scale of computer information technology. As the internet continues to spawn innovative applications, the demand for the internet's scale, functionality, and performance is increasing. The design of real-time log analysis based on ELK can ensure the operational and maintenance benefits for enterprises. However, it is not perfect; technology will continue to evolve and update, and currently, this architecture is capable of meeting existing needs.

The fusion of computer technology and IT development has spurred the rapid development of internet applications and scales. As innovative internet applications continually emerge, people's demands for applications' functionality and performance are increasing. Traditional integration technology can no longer perfectly solve these issues. Therefore, a new generation of automated continuous integration platforms is set to take its place within the existing integration architecture. Enterprise applications are constantly changing in response to customer demands, and the rise of apps presents many new challenges in this field. IT industry automation has become a trend of the era, and the direction of building automated integration development has been established. It is foreseeable that in the near future, this project design could be applied in more enterprise environments, and this design will go further on the path of network architecture application.

References

[1] Yang, D. Q., & Li, L. (2020, January 31). Discussion on constructing simulation training environments based on virtualization technology. Engineering and Technology Research.

[2] Li, L. T. (2021, December 1). Design and implementation of a data collection and analysis platform based on ELK. Electronics Technology and Software Engineering.

[3] Wang, J. S., & Hong, S. (2021, October 15). Research on network security log visualization based on ELK Stack. Software.

[4] Li, S. D., Liu, Z. R., & Zhu, Q. (2022, March 15). Design and implementation of an operation and maintenance support system based on ELK. Journal of Qingdao University (Engineering & Technology Edition).

[5] Xu, H. J. (2020, October 5). Hierarchical optimization strategy for ELK clusters. Network Security and Informatization.

[6] Ruan, X. L., & Dong, K. L. (2019, December 15). Research and exploration of Nginx reverse proxy log analysis and business service quality evaluation system based on ELK. Network Security Technology and Application.

[7] Li, W., & Shao, Z. (2019, December 5). Research on server monitoring for Linux-like systems. Network Security and Informatization.

[8] Zhu, Q. S., Li, J. W., Wang, J., Li, D. F., & Gao, S. (2021, February 15). Development of a remote monitoring system for intelligent heavy tractors based on embedded Linux. Internal Combustion Engines and Power Plants.

[9] Fan, B. Z. (2020, July 23). Optimization of power monitoring systems based on embedded Linux systems. Ship Science and Technology.

[10] Schoenecker Kathryn A; Zeigenfuss Linda C.; Augustine David J. Can grazing by elk and bison stimulate herbaceous plant productivity in semiarid ecosystems? [J]. Ecosphere,2022-04-08.