Audit Logs

PoC to evaluate tools available

Why not Loki that we are already familiar with

1. Not Designed for High-Volume Secure Log Storage
    a. Loki is optimized for short-term log storage (e.g., days to weeks).
2. High cardinality issues (too many unique log labels)
3. Data Loss Risk
    a. If Loki crashes before flushing logs to disk or object storage, those logs are lost.
    b. Large log volumes cause delayed flushes, increasing the window of potential data loss.
4. No built-in Immutability

Research

graylog

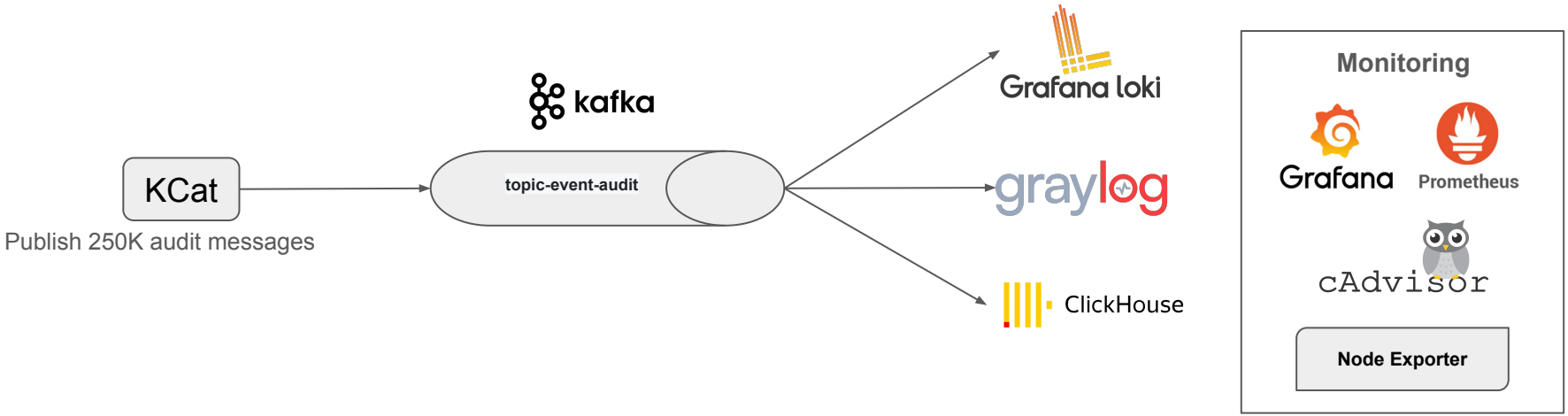
1. E2E Centralized Log Management software
    a. Extensive Input Support
    b. Dashboards and Visualization
    c. Retention and Archiving
    d. Security and Access Control
2. Scalability
3. Rules and Pipelines for data transformations
    a. Flatten Arbitrary JSON data to KV pairs
4. Works with opensearch

ClickHouse

1. High-Performance Columnar Database
2. Efficient Resource Usage
3. Scalability
4. Immutability
5. SQL-Based Queries
6. Data Compression
7. Low Latency for Queries
8. Real-World Adoption Stories
9. Need to have pre-agreed audit log format

PoC

1. Docker Compose Stack
2. Data ingestion from Kafka
3. Visualization
4. Performance Comparison

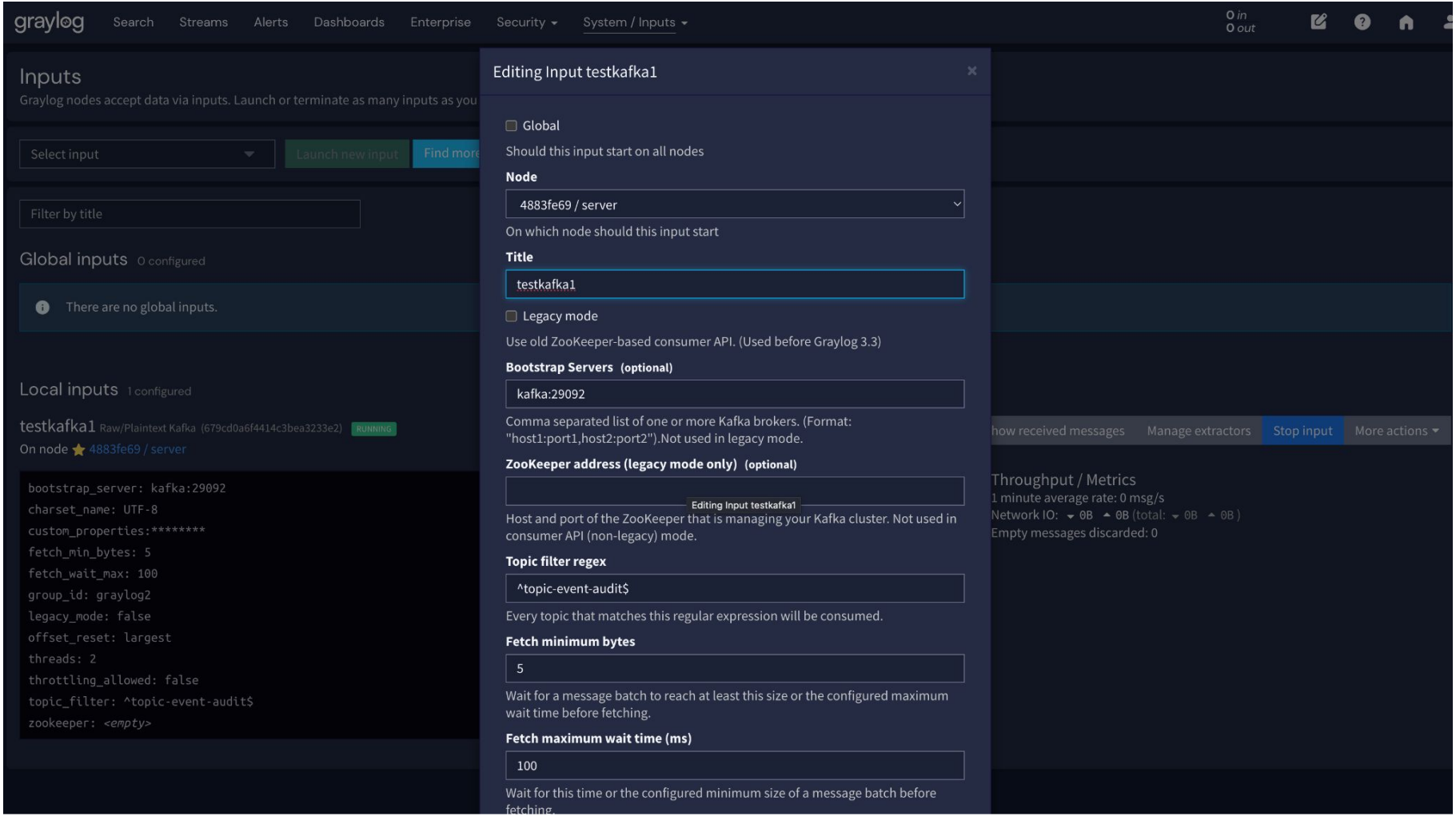


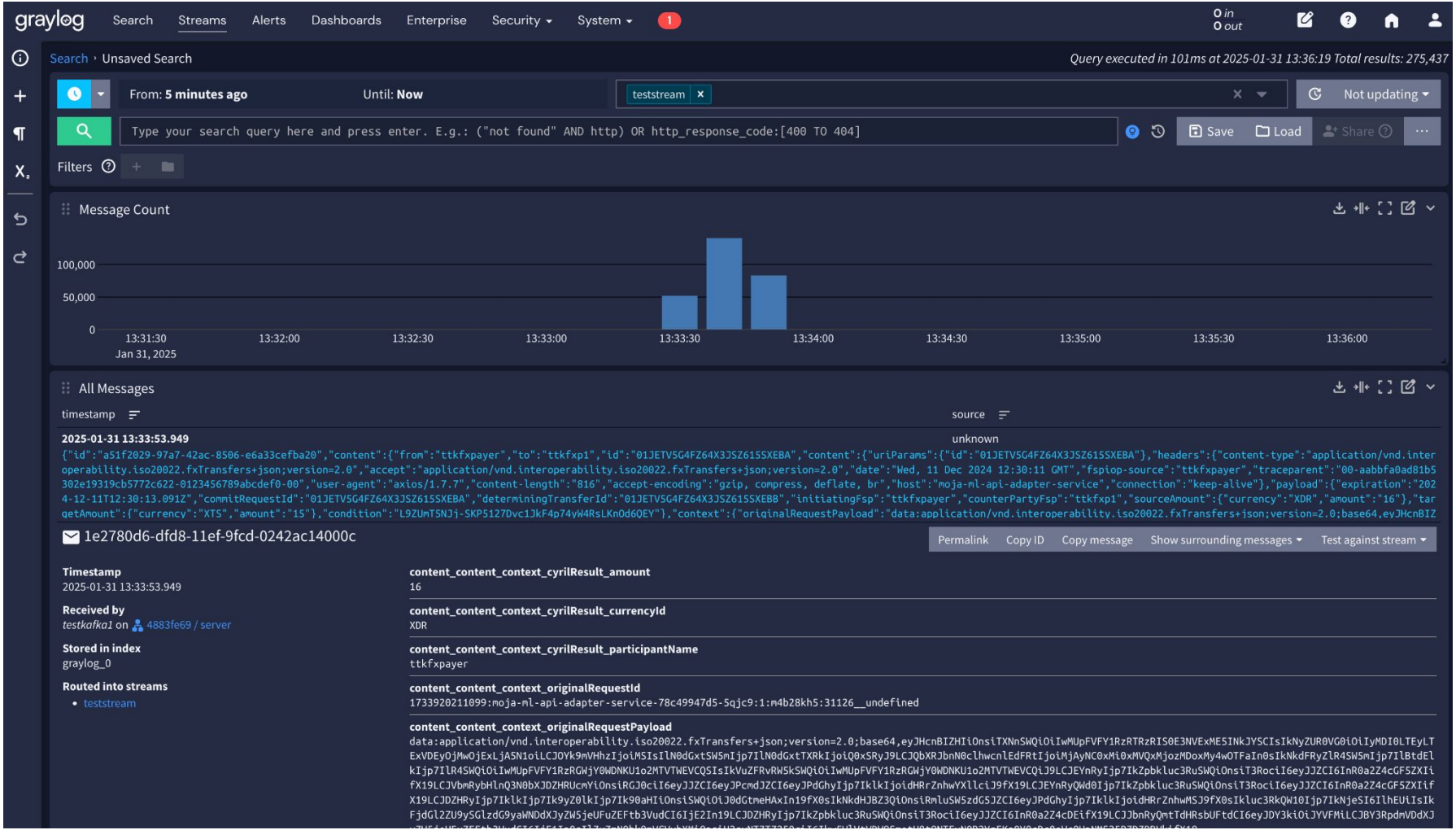
Observations

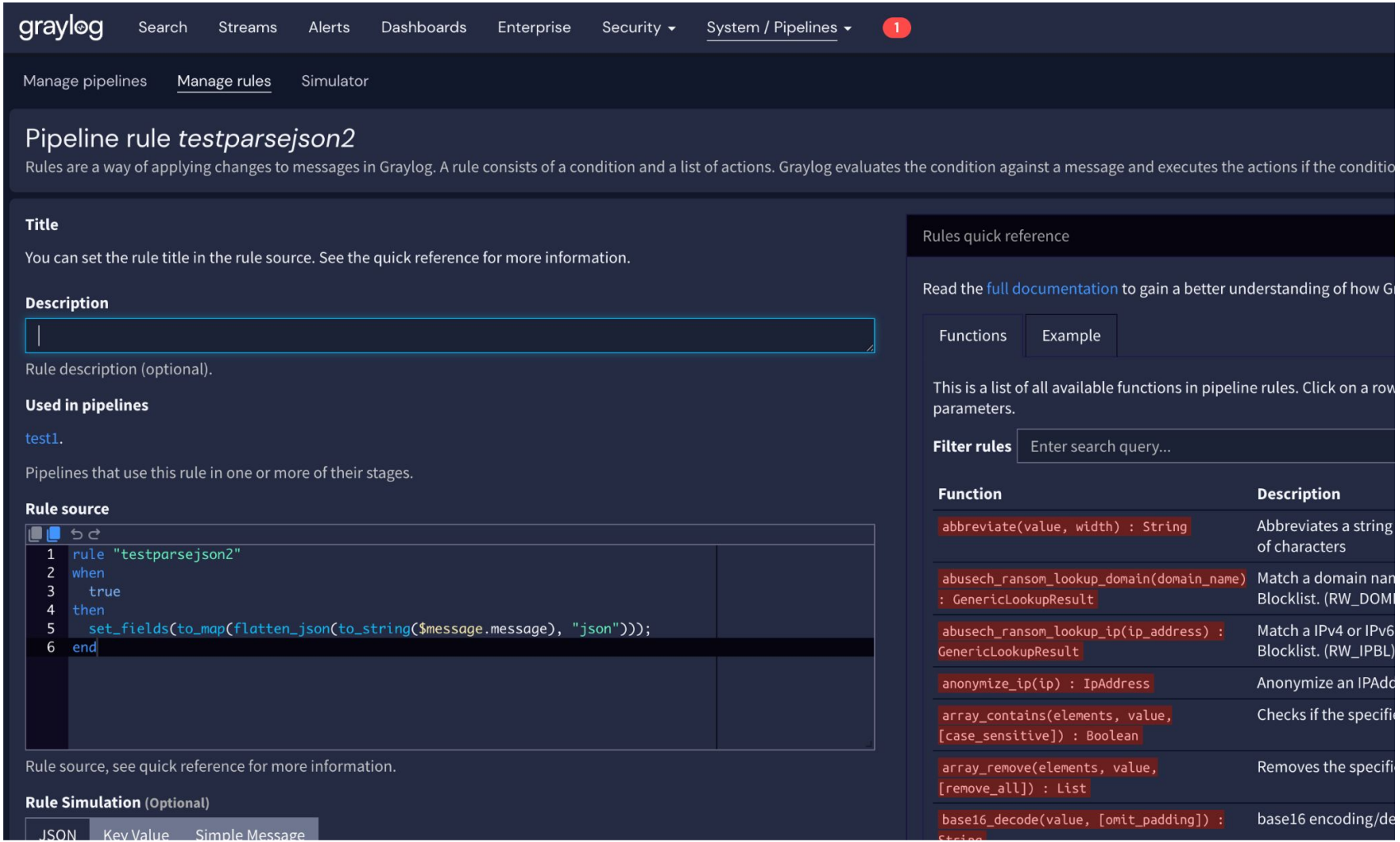
1. Loki seems struggling with high volume and delayed flushing seems not suitable for audit logs
2. Graylog is consuming more resources but looks feature rich and ready made e2e solution
3. Clickhouse is very efficient but needs alignment in the audit logs

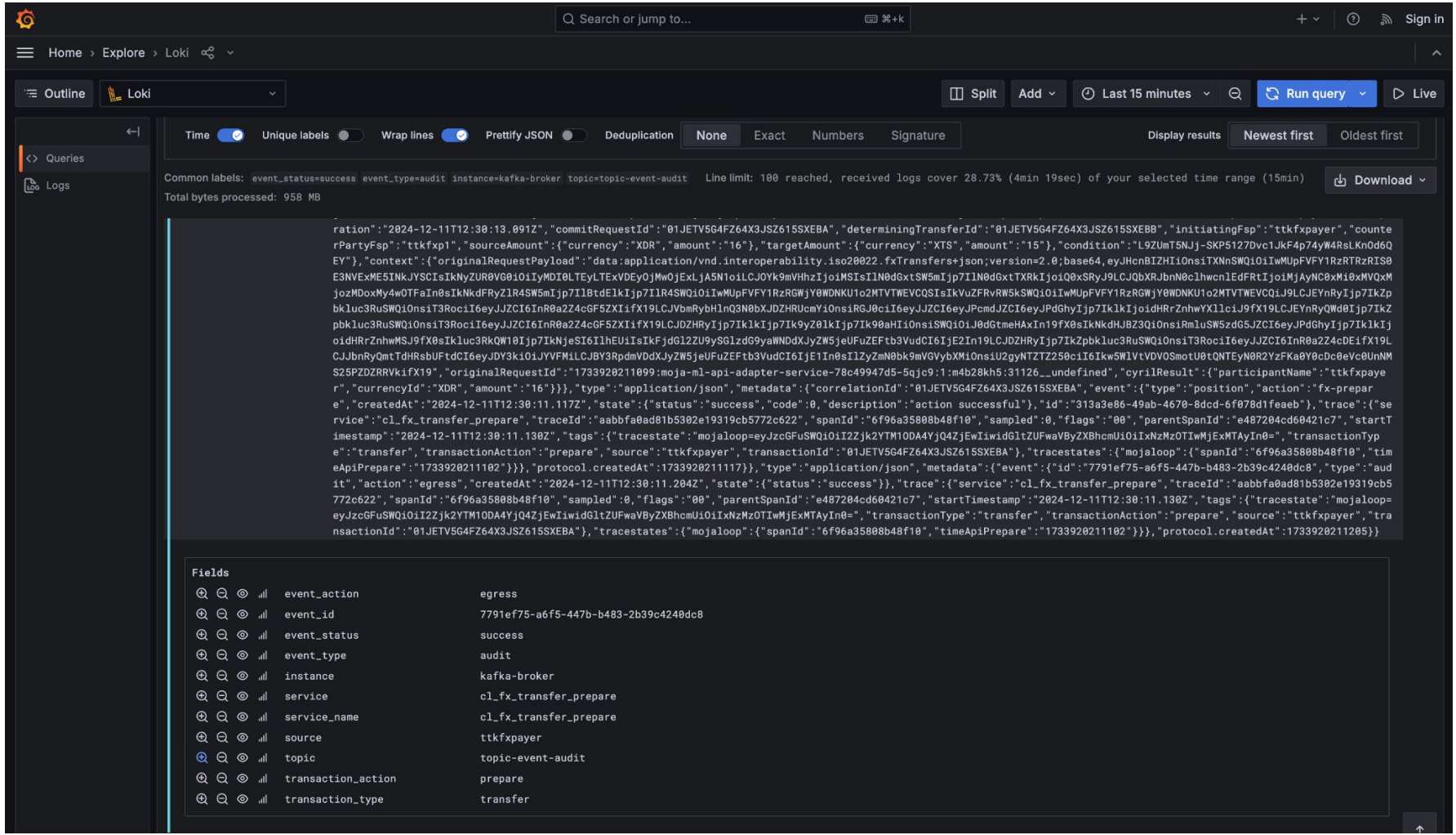
Feedback from team

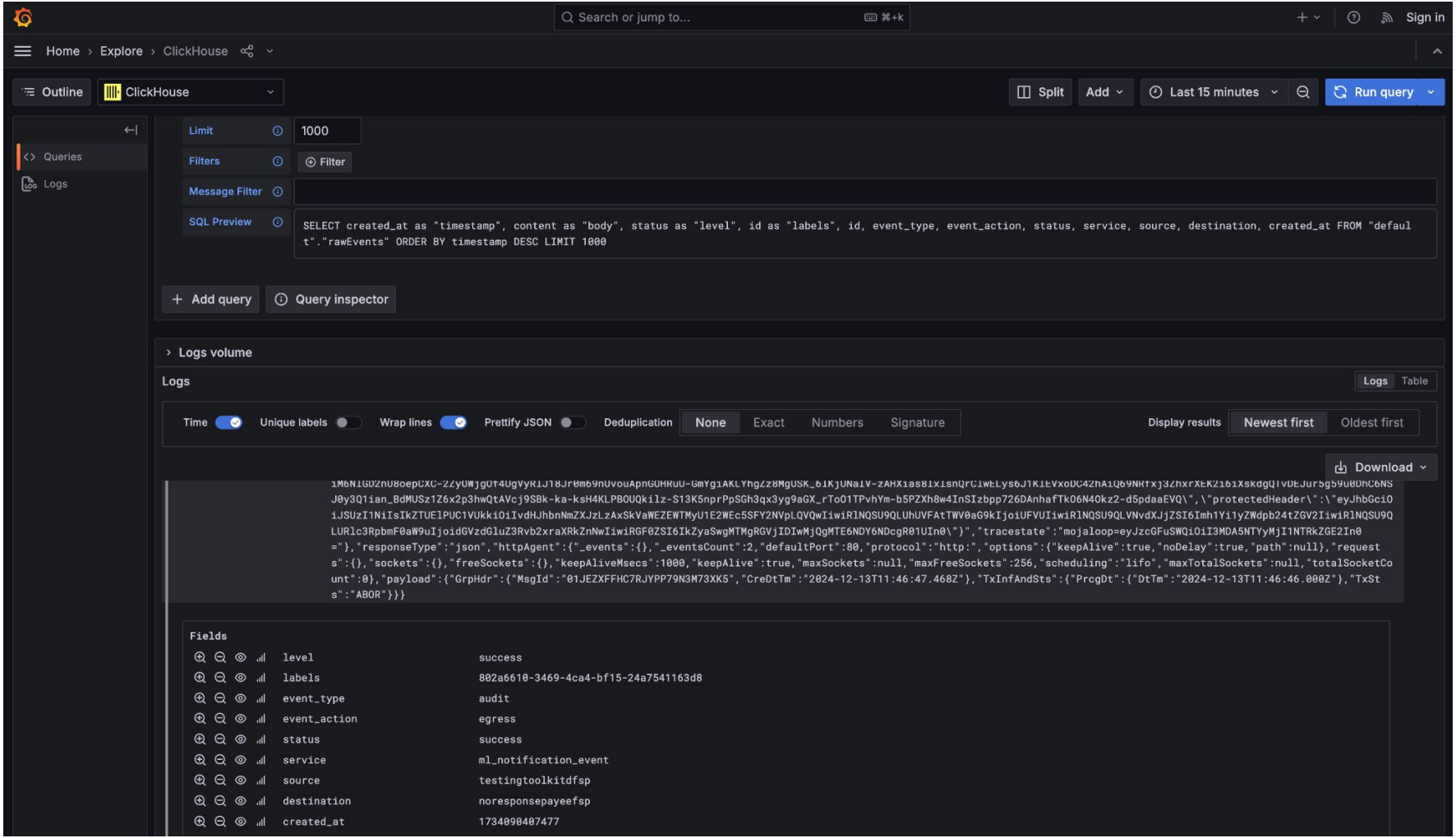
1. Joseph raised a concern regarding HA mentioning he faced some issues with graylog previously. Need to look into this aspect
2. Clickhouse: Muzammil questioned if clickhouse is available in cloud providers as managed services. It seems available in AWS, Azure and Google Cloud. But need to double check.


graylog
Search
Streams
Alerts
Dashboards
Enterprise
Security
System / Inputs
0 in
0 out
Inputs
Graylog nodes accept data via inputs. Launch or terminate as many inputs as you
Select input
Launch new input
Find more
Filter by title
Global inputs 0 configured
There are no global inputs.
Local inputs 1 configured
testkafka1 Raw/Plaintext Kafka (679cd0a6f4414c3bea3233e2) RUNNING
On node 4883fe69 / server
bootstrap_server: kafka:29092
charset_name: UTF-8
custom_properties:********
fetch_min_bytes: 5
fetch_wait_max: 100
group_id: graylog2
legacy_mode: false
offset_reset: largest
threads: 2
throttling_allowed: false
topic_filter: ^topic-event-audit$
zookeeper: <empty>
how received messages
Manage extractors
Stop input
More actions
Throughput / Metrics
1 minute average rate: 0 msg/s
Network IO: 0B 0B (total: 0B 0B)
Empty messages discarded: 0
Editing Input testkafka1
Global
Should this input start on all nodes
Node
4883fe69 / server
On which node should this input start
Title
testkafka1
Legacy mode
Use old ZooKeeper-based consumer API. (Used before Graylog 3.3)
Bootstrap Servers (optional)
kafka:29092
Comma separated list of one or more Kafka brokers. (Format: "host1:port1,host2:port2").Not used in legacy mode.
ZooKeeper address (legacy mode only) (optional)
Editing Input testkafka1
Host and port of the ZooKeeper that is managing your Kafka cluster. Not used in consumer API (non-legacy) mode.
Topic filter regex
^topic-event-audit$
Every topic that matches this regular expression will be consumed.
Fetch minimum bytes
5
Wait for a message batch to reach at least this size or the configured maximum wait time before fetching.
Fetch maximum wait time (ms)
100
Wait for this time or the configured minimum size of a message batch before fetching.


graylog
Search
Streams
Alerts
Dashboards
Enterprise
Security
System
1
0 in
0 out
Search › Unsaved Search
Query executed in 101ms at 2025-01-31 13:36:19 Total results: 275,437
From: 5 minutes ago
Until: Now
teststream
Not updating
Type your search query here and press enter. E.g.: ("not found" AND http) OR http_response_code:[400 TO 404]
Save
Load
Share
Filters
Message Count
100,000
50,000
0
13:31:30
Jan 31, 2025
13:32:00
13:32:30
13:33:00
13:33:30
13:34:00
13:34:30
13:35:00
13:35:30
13:36:00
All Messages
timestamp
source
2025-01-31 13:33:53.949
unknown
{"id":"a51f2029-97a7-42ac-8506-e6a33cefba20","content":{"from":"ttkfxpayer","to":"ttkfxp1","id":"01JETV5G4FZ64X3JSZ615SXEBA","content":{"uriParams":{"id":"01JETV5G4FZ64X3JSZ615SXEBA"},"headers":{"content-type":"application/vnd.interoperability.iso20022.fxTransfers+json;version=2.0","accept":"application/vnd.interoperability.iso20022.fxTransfers+json;version=2.0","date":"Wed, 11 Dec 2024 12:30:11 GMT","fspiop-source":"ttkfxpayer","traceparent":"00-aabbfa0ad81b5302e19319cb5772c622-0123456789abcdef0-00","user-agent":"axios/1.7.7","content-length":"816","accept-encoding":"gzip, compress, deflate, br","host":"moja-ml-api-adapter-service","connection":"keep-alive"},"payload":{"expiration":"2024-12-11T12:30:13.091Z","commitRequestId":"01JETV5G4FZ64X3JSZ615SXEBA","determiningTransferId":"01JETV5G4FZ64X3JSZ615SXEBB","initiatingFsp":"ttkfxpayer","counterPartyFsp":"ttkfxp1","sourceAmount":{"currency":"XDR","amount":"16"},"targetAmount":{"currency":"XTS","amount":"15"},"condition":"L9ZUmT5NJj-SKP5127Dvc1JkF4p74yW4RsLKnOd6QEY"},"context":{"originalRequestPayload":"data:application/vnd.interoperability.iso20022.fxTransfers+json;version=2.0;base64,eyJHcnBIZ
1e2780d6-dfd8-11ef-9fcd-0242ac14000c
Permalink
Copy ID
Copy message
Show surrounding messages
Test against stream
Timestamp
2025-01-31 13:33:53.949
Received by
testkafka1 on 4883fe69 / server
Stored in index
graylog_0
Routed into streams
teststream
content_content_context_cyrilResult_amount
16
content_content_context_cyrilResult_currencyId
XDR
content_content_context_cyrilResult_participantName
ttkfxpayer
content_content_context_originalRequestId
1733920211099:moja-ml-api-adapter-service-78c49947d5-5qjc9:1:m4b28kh5:31126__undefined
content_content_context_originalRequestPayload
data:application/vnd.interoperability.iso20022.fxTransfers+json;version=2.0;base64,eyJHcnBIZHIiOnsiTXNnSWQiOiIwMUpFVFY1RzRTRzRIS0E3NVExME5INkJYSCIsIkNyZUR0VG0iOiIyMDI0LTEyLTExVDEyOjMwOjExLjA5N1oiLCJOYk9mVHhzIjoiMSIsIlN0dGxtSW5mIjp7IlN0dGxtTXRkIjoiQ0xSRyJ9LCJQbXRJbnN0clhwcnlEdFRtIjoiMjAyNC0xMi0xMVQxMjozMDoxMy4wOTFaIn0sIkNkdFRyZlR4SW5mIjp7IlBtdElkIjp7IlR4SWQiOiIwMUpFVFY1RzRGWjY0WDNKU1o2MTVTWEVCQSIsIkVuZFRvRW5kSWQiOiIwMUpFVFY1RzRGWjY0WDNKU1o2MTVTWEVCQiJ9LCJEYnRyIjp7IkZpbkluc3RuSWQiOnsiT3RociI6eyJJZCI6InR0a2Z4cGF5ZXIifX19LCJDbmRybHlnQ3N0bXJDZHRUcmYiOnsiRGJ0ciI6eyJJZCI6eyJPcmdJZCI6eyJPdGhyIjp7IklkIjoidHRrZnhwYXllciJ9fX19LCJEYnRyQWd0Ijp7IkZpbkluc3RuSWQiOnsiT3RociI6eyJJZCI6InR0a2Z4cGF5ZXIifX19LCJDZHRyIjp7IklkIjp7Ik90aHIiOnsiSWQiOiJ0dGtmeHAxIn19fX0sIkNkdHJBZ3QiOnsiRmluSW5zdG5JZCI6eyJPdGhyIjp7IklkIjoidHRrZnhwMSJ9fX0sIkluc3RkQW10Ijp7IkNjeSI6IlhUIiwiFjdGl2ZU9ySGlzdG9yaWNDdXJyZW5jeUFuZEFtb3VudCI6IjE2In19LCJDZHRyIjp7IkZpbkluc3RuSWQiOnsiT3RociI6eyJJZCI6InR0a2Z4cDEifX19LCJBbnRyQmtTdHRsbUFtdCI6eyJDY3kiOiJYVFMiLCJBY3RpdmVDdXJ
Permalink
Copy ID
Copy message
Show surrounding messages
Test against stream


graylog
Search
Streams
Alerts
Dashboards
Enterprise
Security
System / Pipelines
1
Manage pipelines
Manage rules
Simulator
Pipeline rule testparsejson2
Rules are a way of applying changes to messages in Graylog. A rule consists of a condition and a list of actions. Graylog evaluates the condition against a message and executes the actions if the conditio
Title
You can set the rule title in the rule source. See the quick reference for more information.
Description
Rule description (optional).
Used in pipelines
test1.
Pipelines that use this rule in one or more of their stages.
Rule source
rule "testparsejson2"
when
true
then
set_fields(to_map(flatten_json(to_string($message.message), "json")));
end
Rule source, see quick reference for more information.
Rule Simulation (Optional)
JSON
Key Value
Simple Message
Rules quick reference
Read the full documentation to gain a better understanding of how G
Functions
Example
This is a list of all available functions in pipeline rules. Click on a row
parameters.
Filter rules
Enter search query...
Function
Description
abbreviate(value, width) : String
Abbreviates a string
of characters
abusech_ransom_lookup_domain(domain_name)
: GenericLookupResult
Match a domain nan
Blocklist. (RW_DOM
abusech_ransom_lookup_ip(ip_address) :
GenericLookupResult
Match a IPv4 or IPv6
Blocklist. (RW_IPBL)
anonymize_ip(ip) : IpAddress
Anonymize an IPAdd
array_contains(elements, value,
[case_sensitive]) : Boolean
Checks if the specifi
array_remove(elements, value,
[remove_all]) : List
Removes the specifi
base16_decode(value, [omit_padding]) :
base16 encoding/de


Search or jump to... ⌘+k
Sign in
Home › Explore › Loki
Outline
Loki
Split
Add
Last 15 minutes
Run query
Live
Queries
Logs
Time
Unique labels
Wrap lines
Prettify JSON
Deduplication
None
Exact
Numbers
Signature
Display results
Newest first
Oldest first
Common labels: event_status=success event_type=audit instance=kafka-broker topic=topic-event-audit
Line limit: 100 reached, received logs cover 28.73% (4min 19sec) of your selected time range (15min)
Download
Total bytes processed: 958 MB
ration":"2024-12-11T12:30:13.091Z","commitRequestId":"01JETV5G4FZ64X3JSZ615SXEBA","determiningTransferId":"01JETV5G4FZ64X3JSZ615SXEBB","initiatingFsp":"ttkfxpayer","counterPartyFsp":"ttkfxp1","sourceAmount":{"currency":"XDR","amount":"16"},"targetAmount":{"currency":"XTS","amount":"15"},"condition":"L9ZUmT5NJj-SKP5127Dvc1JkF4p74yW4RsLKnOd6QEY"},"context":{"originalRequestPayload":"data:application/vnd.interoperability.iso20022.fxTransfers+json;version=2.0;base64,eyJHcnBIZHIiOnsiTXNnSWQiOiIwMUpFVFY1RzRTRzRIS0E3NVExME5INkJYSCIsIkNyZUR0VG0iOiIyMDI0LTEyLTExVDEyOjMwOjExLjA5N1oiLCJOYk9mVHhzIjoiMSIsIlN0dGxtSW5mIjp7IlN0dGxtXRkIjoiQ0xSRyJ9LCJQbXRJbnN0clhwcnlEdFRtIjoiMjAyNC0xMi0xMVQxMjozMDoxMy4wOTFaIn0sIkNkdFRyZlR4SW5mIjp7IlBtdElkIjp7IlR4SWQiOiIwMUpFVFY1RzRGWjY0WDNKU1o2MTVTWEVCQSIsIkVuZFRvRW5kSWQiOiIwMUpFVFY1RzRGWjY0WDNKU1o2MTVTWEVCQiJ9LCJEYnRyIjp7IkZpbkluc3RuSWQiOnsiT3RociI6eyJJZCI6InR0a2Z4cGF5ZXIifX19LCJVbmRybHlnQ3N0bXJDZHRUcmYiOnsiRGJ0ciI6eyJJZCI6eyJPcmdJZCI6eyJPdGhyIjp7IklkIjoidHRrZnhwYXllciJ9fX19LCJEYnRyQWd0Ijp7IkZpbkluc3RuSWQiOnsiT3RociI6eyJJZCI6InR0a2Z4cGF5ZXIifX19LCJVbmRybHlnQ3N0bXJDZHRUcmYiOnsiRGJ0ciI6eyJJZCI6eyJPcmdJZCI6eyJPdGhyIjp7IklkIjoidHRrZnhwMSJ9fX0sIkluc3RkQW10Ijp7IkNjeSI6IlhEUiIsIkFjdGl2ZU9ySGlzdG9yaWNDdXJyZW5jeUFuZEFtb3VudCI6IjE2In19LCJDZHRyIjp7IkZpbkluc3RuSWQiOnsiT3RociI6eyJJZCI6InR0a2Z4cDEifX19LCJJbnRyQmtTdHRsbUFtdCI6eyJDY3kiOiJYVFMiLCJBY3RpdmVDdXJyZW5jeUFuZEFtb3VudCI6IjE1In0sIlZyZmN0bk9mVGVybXMiOnsiU2gyNTZTZ250ciI6Ikw5WlVtVDVOSmotU0tQNTEyN0R2YzFKa0Y0cDc0eVc0UnNMS25PZDZRRVkifX19","originalRequestId":"1733920211099:moja-ml-api-adapter-service-78c49947d5-5qjc9:1:m4b28kh5:31126__undefined","cyrilResult":{"participantName":"ttkfxpayer","currencyId":"XDR","amount":"16"}}},"type":"application/json","metadata":{"correlationId":"01JETV5G4FZ64X3JSZ615SXEBA","event":{"type":"position","action":"fx-prepare","createdAt":"2024-12-11T12:30:11.117Z","state":{"status":"success","code":0,"description":"action successful"},"id":"313a3e86-49ab-4670-8dcd-6f078d1feaeb"},"trace":{"service":"cl_fx_transfer_prepare","traceId":"aabbfa0ad81b5302e19319cb5772c622","spanId":"6f96a35808b48f10","sampled":0,"flags":"00","parentSpanId":"e487204cd60421c7","startTimestamp":"2024-12-11T12:30:11.130Z","tags":{"tracestate":"mojaloop=eyJzcGFuSWQiOiI2Zjk2YTM1ODA4YjQ4ZjEwIiwidGltZUFwaVByZXBhcmUiOiIxNzMzOTIwMjExMTAyIn0=","transactionType":"transfer","transactionAction":"prepare","source":"ttkfxpayer","transactionId":"01JETV5G4FZ64X3JSZ615SXEBA"},"tracestates":{"mojaloop":{"spanId":"6f96a35808b48f10","timeApiPrepare":"1733920211102"}}},"protocol.createdAt":1733920211117}},"type":"application/json","metadata":{"event":{"id":"7791ef75-a6f5-447b-b483-2b39c4240dc8","type":"audit","action":"egress","createdAt":"2024-12-11T12:30:11.204Z","state":{"status":"success"}},"trace":{"service":"cl_fx_transfer_prepare","traceId":"aabbfa0ad81b5302e19319cb5772c622","spanId":"6f96a35808b48f10","sampled":0,"flags":"00","parentSpanId":"e487204cd60421c7","startTimestamp":"2024-12-11T12:30:11.130Z","tags":{"tracestate":"mojaloop=eyJzcGFuSWQiOiI2Zjk2YTM1ODA4YjQ4ZjEwIiwidGltZUFwaVByZXBhcmUiOiIxNzMzOTIwMjExMTAyIn0=","transactionType":"transfer","transactionAction":"prepare","source":"ttkfxpayer","transactionId":"01JETV5G4FZ64X3JSZ615SXEBA"},"tracestates":{"mojaloop":{"spanId":"6f96a35808b48f10","timeApiPrepare":"1733920211102"}}},"protocol.createdAt":1733920211205}}
Fields
event_action egress
event_id 7791ef75-a6f5-447b-b483-2b39c4240dc8
event_status success
event_type audit
instance kafka-broker
service cl_fx_transfer_prepare
service_name cl_fx_transfer_prepare
source ttkfxpayer
topic topic-event-audit
transaction_action prepare
transaction_type transfer


Search or jump to...
Sign in
Home › Explore › ClickHouse
Outline
ClickHouse
Split
Add
Last 15 minutes
Run query
Queries
Logs
Limit
1000
Filters
Filter
Message Filter
SQL Preview
SELECT created_at as "timestamp", content as "body", status as "level", id as "labels", id, event_type, event_action, status, service, source, destination, created_at FROM "defaul t"."rawEvents" ORDER BY timestamp DESC LIMIT 1000
Add query
Query inspector
Logs volume
Logs
Logs
Table
Time
Unique labels
Wrap lines
Prettify JSON
Deduplication
None
Exact
Numbers
Signature
Display results
Newest first
Oldest first
Download
J0y3Q1ian_BdMUSz1Z6x2p3hwQtAVcj9SBk-ka-ksH4KLPBOUQki1z-S13K5nprPpSGh3qx3yg9aGX_rToO1TPvhYm-b5PZXh8w4InSIzbpp726DAnhafTkO6N4Okz2-d5pdaaEVQ\",\"protectedHeader\":\"eyJhbGciO
iJSUzI1NiIsIkZTUElPUC1VUkkiOiIvdHJhbnNmZXJzLzAxSkVaWEZEWTMyU1E2WEc5SFY2NVpLQVQwIiwiRlNQSU9QLUhUVFAtTWV0aG9kIjoiUFVUIiwiRlNQSU9QLVNvdXJjZSI6Imh1Yi1yZWdpb24tZGV2IiwiRlNQSU9Q
LURlc3RpbmF0aW9uIjoidGVzdGluZ3Rvb2xraXRkZnNwIiwiRGF0ZSI6IkZyaSwgMTMgRGVjIDIwMjQgMTE6NDY6NDcgR01UIn0\"}","tracestate":"mojaloop=eyJzcGFuSWQiOiI3MDA5NTYyMjI1NTRkZGE2In0
="},"responseType":"json","httpAgent":{"_events":{},"_eventsCount":2,"defaultPort":80,"protocol":"http:","options":{"keepAlive":true,"noDelay":true,"path":null},"request
s":{},"sockets":{},"freeSockets":{},"keepAliveMsecs":1000,"keepAlive":true,"maxSockets":null,"maxFreeSockets":256,"scheduling":"lifo","maxTotalSockets":null,"totalSocketCo
unt":0},"payload":{"GrpHdr":{"MsgId":"01JEZXFFHC7RJYPP79N3M73XK5","CreDtTm":"2024-12-13T11:46:47.468Z"},"TxInfAndSts":{"PrcgDt":{"DtTm":"2024-12-13T11:46:46.000Z"},"TxSt
s":"ABOR"}}}
Fields
level
success
labels
802a6610-3469-4ca4-bf15-24a7541163d8
event_type
audit
event_action
egress
status
success
service
ml_notification_event
source
testingtoolkitdfsp
destination
noresponsepayeefsp
created_at
1734090407477

Search or jump to... ⌘+k

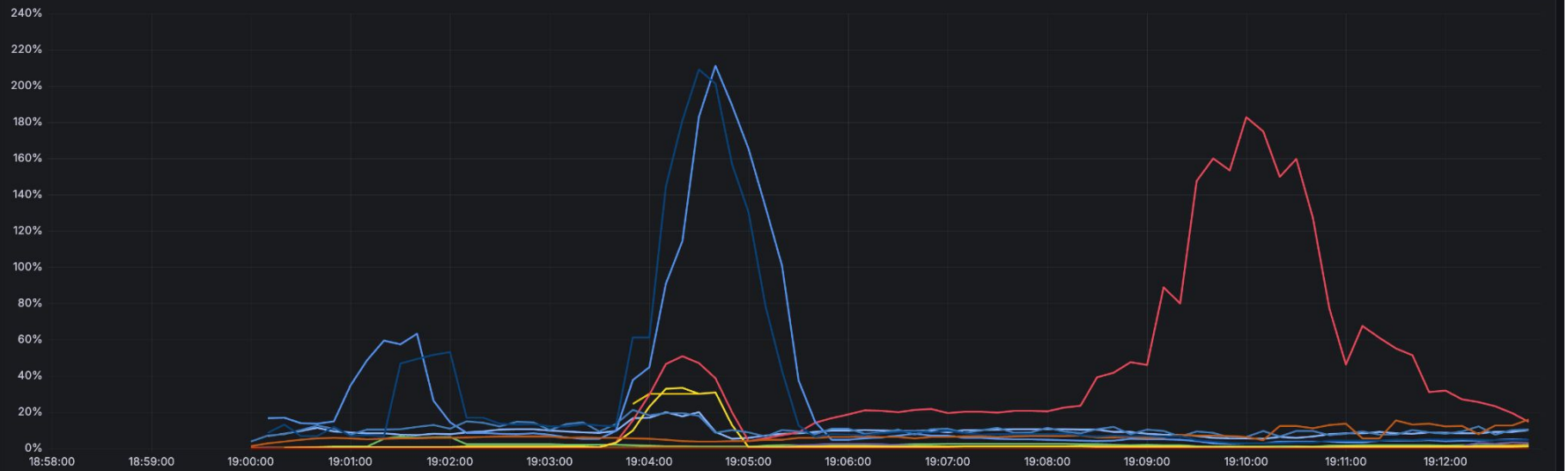
Sign in

Home › Dashboards › Docker Prometheus Monitoring › View panel

Back to dashboard | Share

interval 1m | IP All | Env All | CPU Name All | Image render | Last 15 minutes | Refresh 10s

Container CPU usage



Name	Mean	Last *
loki-loki-1	36.4%	14.8%
upbeat_wiles	29.1%	30.2%
graylog-datanode-1	25.7%	4.79%
graylog-graylog-1	25.3%	4.82%
kafka	10.4%	10.3%
clickhouse	9.30%	10.0%
monitoring-cadvisor-1	6.98%	16.0%
loki-promtail-1	3.15%	1.14%
graylog-mongodb-1	2.02%	1.97%
monitoring-grafana-1	0.712%	2.95%

Search or jump to... ⌘+k

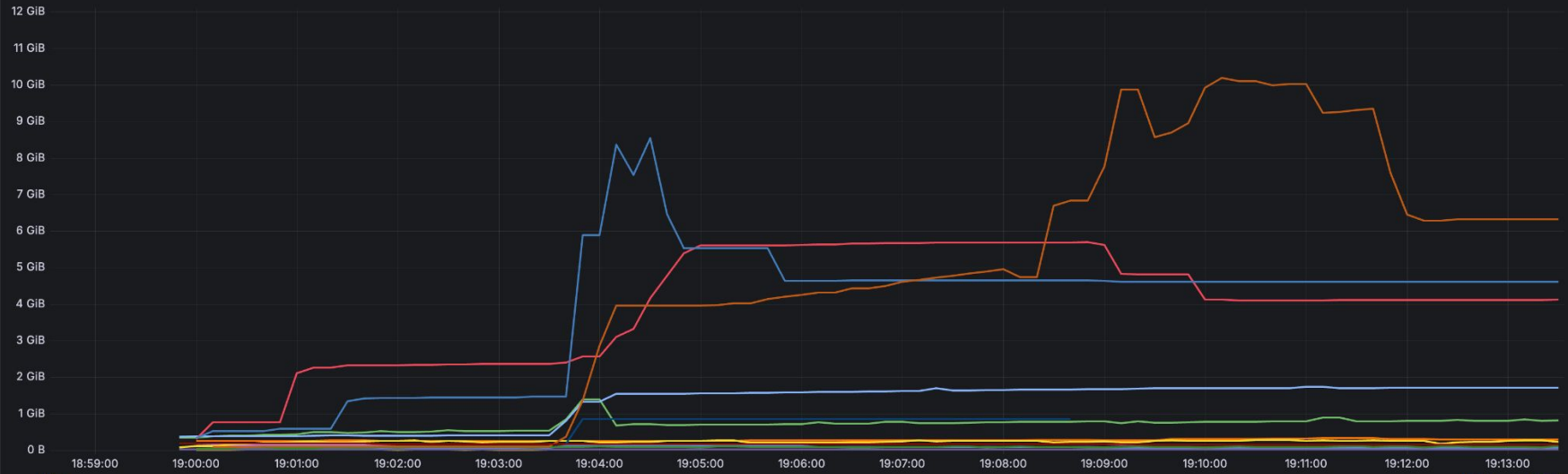
Sign in

Home › Dashboards › Docker Prometheus Monitoring › View panel

Back to dashboard | Share

interval 1m | IP All | Env All | CPU Name All | Image render | Last 15 minutes | Refresh 10s

Container Memory Usage



Name	Mean	Last *
loki-loki-1	4.60 GiB	6.32 GiB
graylog-datanode-1	3.97 GiB	4.12 GiB
graylog-graylog-1	3.88 GiB	4.61 GiB
kafka	1.29 GiB	1.71 GiB
upbeat_wiles	855 MiB	877 MiB
clickhouse	719 MiB	835 MiB
monitoring-grafana-1	285 MiB	303 MiB
monitoring-prometheus-1	240 MiB	233 MiB
kafka_audits-kafka-mon-ui-1	171 MiB	160 MiB
graylog-mongodb-1	162 MiB	160 MiB

Appendix

Problem Statement:

The current system publishes audit logs to a Kafka topic, but there is no mechanism to store or visualize these logs. To address this gap, we need a solution tailored to the following characteristics:

1. **Scalability**: Ability to handle high write speeds for large datasets.
2. **Reliability**: Ensure no data loss, with support for archiving and retention policies.
3. **Immutability**: Stored audit data must remain unaltered.
4. **Query Simplicity**: Simple queries on predefined fields with index-based optimization. No complex querying required.
5. **Integration Support**: Compatibility with visualization, transformation, and backup tools.
6. **Efficiency**: Minimal resource consumption during ingestion by avoiding full-text search. Full-text search is generally unnecessary, so configuring indexes for predefined fields would be more efficient. This approach conserves resources during ingestion and improves write throughput. In the past, the ELK stack was observed to consume significant resources when handling large datasets.
7. **Open Licensing**: Must have an Apache License or equivalent.
8. **Kubernetes Compatibility**: Easy deployment and management in Kubernetes environments.

Analysis:

Relational Database Systems

Relational database systems like MySQL or PostgreSQL provide robust querying capabilities and ACID compliance, but they are **not suitable** for this use case:

1. **Complexity**: Designed for complex relational queries, which are unnecessary for simple, predefined field-based queries.
2. **Scaling Limitations**: High throughput requirements for large datasets would necessitate extensive sharding and tuning. Even tools like Vitess addresses some scalability challenges of MySQL, but it introduces significant complexity and fails to address core requirements such as high write throughput, schema flexibility, and efficient ingestion.
3. **Resource Utilization**: Traditional RDBMS are not optimized for high write speeds on large datasets like audit logs, leading to resource inefficiencies.

AWS services like S3, Athena, Quicksight

For projects like COMESA that aim to leverage cloud-managed services, AWS offers a suite of tools to streamline workflows without the need to build custom solutions. By following established market practices, we can design a robust system tailored to our needs.

The proposed approach involves:

- Data Storage: Use Amazon S3 as the primary storage layer, ingesting data from Kafka using Kafka Connect with an S3 Sink Connector.
- Data Querying: Utilize Amazon Athena for SQL-based queries directly on data stored in S3.
- Data Visualization: Leverage Amazon QuickSight to create visual dashboards and reports based on the queried data.

This combination provides scalability, efficiency, and a managed solution, aligning with project goals and industry standards.

Cassandra + PrestoDB + Apache Spark

This combination was considered due to its distributed, high-throughput capabilities:

1. **Strengths**:
    - Cassandra offers excellent scalability for write-heavy workloads.
    - PrestoDB enables fast querying across distributed datasets.
    - Apache Spark provides visualisation.
2. **Concerns**:
    - **Schema Rigidity**: Cassandra requires a predefined schema, which is challenging given the variety of Kafka messages in the audit topic. While the payload (such as FSPIOP/ISO20022) may follow a fixed format, many other fields within the Kafka messages can vary across different services. Schema flexibility is essential to accommodate the evolving log formats and ensure the system can handle a range of message structures.
    - **Overengineering**: The combination introduces additional complexity and maintenance overhead compared to simpler alternatives.

ELK Stack

Previously, we used the **ELK stack** (Elasticsearch, Logstash, Kibana) for log aggregation and analysis. While it provided powerful search and visualization capabilities, we encountered **high resource consumption**, especially during the ingestion of large datasets. However, Elasticsearch can be **tuned** to improve performance by disabling full-text search and configuring field-level indexing for only relevant fields.

Despite these potential optimizations, **licensing** remains a concern with the ELK stack. While the existing **Elastic License** will not directly impact our usage since we do not plan to offer a commercial SaaS solution with it, it's important to note that the **Elastic License** is **not Apache 2.0**. This license change raises the possibility of restrictions on certain use cases. As an alternative, we considered **OpenSearch**, which is a **fully open-source** fork of Elasticsearch and licensed under the **Apache 2.0 license**, providing greater flexibility and fewer legal concerns.

vNext AuditingBC

Another consideration is the **vNext AuditingBC**, which is developed and integrated with the **ELK stack**. While it provides a more standardized and structured way to manage audit logs, there are significant implications to adopting this approach:

- **Increased Implementation Effort**:
  Transitioning to **vNext AuditingBC** would require considerable effort, including updating existing services to use the new library, testing the new audit event format, and ensuring compatibility across the system.
- **Dependency on ELK Stack**:
  The solution relies on the **ELK stack**, which raises the same concerns around **resource consumption** and **licensing** as outlined earlier.
- **Impact on AuditingBC**:
  If we decide to move away from the ELK stack entirely, further changes to **vNext AuditingBC** would be required to align with the new backend for audit log management, adding to the overall complexity and effort.

OpenSearch + Graylog (Recommended Solution)

OpenSearch and Graylog present a balanced, efficient solution for audit log management.

1. **OpenSearch**:
    - **Advantages**:
        - OpenSearch supports indexing on specific fields, avoiding resource-intensive full-text search.
        - Provides high ingestion throughput and customizable query capabilities.
        - Includes visualization tools like OpenSearch Dashboards.
    - **Optimization**: Resource consumption during ingestion can be significantly reduced by disabling full-text search and indexing only required fields. This directly addresses the issues previously observed with ELK stack.
    - **Ingestion:** We can use Kafka Connect with the OpenSearch Sink Connector to ingest data from Kafka
2. **Graylog**:
    - **Advantages**:
        - Purpose-built for log aggregation and analysis.
        - Seamlessly integrates with OpenSearch for storage and querying.
        - Provides a user-friendly interface for visualization and log management.
    - **Feature Compatibility**:
      Graylog supports essential features such as **retention policies**, **archiving**, and **Kubernetes deployment**, providing a flexible and scalable solution for managing logs in cloud-native environments.
    - **Commercial Offerings**:
      For organizations that need additional functionality, **Graylog's commercial offerings** for some advanced features. However, for basic audit log management, the **open-source version** is sufficient. Organizations can opt for commercial packages as their needs grow.

Follow-Up

After my initial investigation where I suggested **Graylog** (based on OpenSearch) as a potential tool for log management, I conducted further research and identified **ClickHouse** as another interesting option. While it requires compromising on **JSON flexibility**, it excels in **resource efficiency** and **high write throughput**, making it a strong alternative to traditional solutions like the ELK stack.

Why ClickHouse?

1. **Efficient Resource Usage**: Handles large-scale data without excessive resource consumption.
2. **High Write Throughput**: Optimized for fast ingestion of high-volume logs.
3. **Immutability**: Ensures data integrity with its append-only architecture.
4. **Retention and Archival**: Supports Time-To-Live (TTL) policies for data lifecycle management.
5. **Visualization Integration**: Works with tools like **Grafana** and **Apache Superset**.

Real-World Adoption Stories

Organizations have migrated from ELK to ClickHouse for better performance and cost efficiency:

1. **Didi's Migration**:
    - Transitioned to ClickHouse for a next-generation log storage system with reduced resource usage and improved performance. Learn more.
2. **Trip.com's 50PB Logging Solution**:
    - Built a massive logging system with ClickHouse, reducing costs and addressing scaling challenges. Learn more.

Considerations

- **ClickHouse** is ideal for systems where resource efficiency, immutability, and high write throughput are priorities. However, its limited JSON handling might require additional preprocessing.
- **Graylog with OpenSearch** offers greater flexibility for structured and unstructured log data, making it better suited for environments with diverse query need