**Resolving Graylog Web Interface Unavailability**

### **Note: Three Different Problems with the Same Symptom**

During the setup and configuration of Graylog, I encountered **three distinct problems** that both resulted in the **Graylog web interface being unavailable**. While the symptom (inaccessible web interface) was the same, the root causes and solutions were different. Below is a summary of the problems:

## **PROBLEM 1: Documentation: Resolving Graylog Web Interface Unavailability- Elasticsearch Authentication Issue**

**Problem Description**

The Graylog web interface was unavailable, preventing access to the SIEM dashboard and configuration tools. This issue was critical as it halted the setup of security monitoring and alerting for Catnip Games International.



**Symptoms**

1. Graylog web interface was inaccessible at [http://192.168.123.1:9000](http://192.168.123.1:9000/).
2. Graylog, Elasticsearch, and MongoDB services were running, but Graylog logs showed repeated errors related to Elasticsearch authentication.

**Error Details**

The Graylog logs (/var/log/graylog-server/server.log) contained the following error:

ERROR [VersionProbe] Unable to retrieve version from Elasticsearch node 127.0.0.1:9200: Error response: type: security\_exception - reason: missing authentication credentials for REST request [/]

This error indicated that Graylog was unable to authenticate with Elasticsearch, which is a critical dependency for Graylog to function.

**Root Cause**

Elasticsearch had security features enabled (authentication and TLS), but Graylog was not configured to provide the necessary credentials to connect to Elasticsearch. As a result, Graylog could not retrieve data from Elasticsearch, causing the web interface to fail.

**Steps Taken to Resolve the Issue**

**Step 1: Verify Elasticsearch Security Settings**

1. Ran the following command to check if Elasticsearch security was enabled:

curl -X GET "localhost:9200/\_security/\_authenticate?pretty" --user elastic:Student1

1. Confirmed that Elasticsearch security was enabled and the credentials (elastic:Student1) were valid.

**Step 2: Update Graylog Configuration**

1. Edited the Graylog configuration file (/etc/graylog/server/server.conf) to include Elasticsearch credentials:

elasticsearch\_hosts = [http://elastic:Student1@127.0.0.1:9200](http://elastic:Student1@127.0.0.1:9200/)

1. Saved the file and restarted the Graylog service:

sudo systemctl restart graylog-server

**Step 3: Verify the Fix**

1. Checked the Graylog logs to confirm that the connection to Elasticsearch was successful:

sudo tail -n 100 /var/log/graylog-server/server.log

1. Observed that Graylog successfully connected to Elasticsearch and the web interface became accessible.

**Outcome**

* The Graylog web interface is now accessible at [http://192.168.123.1:9000](http://192.168.123.1:9000/).
* Graylog is successfully connected to Elasticsearch, enabling log aggregation, analysis, and alerting.
* The issue was resolved by configuring Graylog to authenticate with Elasticsearch using the correct credentials.

**Lessons Learned**

1. ***Dependency Management*:** Graylog relies on Elasticsearch and MongoDB. Ensuring these services are properly configured and accessible is critical for Graylog to function.
2. ***Security Configuration:*** Elasticsearch’s security features (authentication and TLS) must be accounted for when integrating with Graylog.
3. ***Log Analysis***: System logs (/var/log/graylog-server/server.log) are invaluable for diagnosing and resolving issues.

**Next Steps**

With the Graylog web interface now accessible, the next steps are:

1. Setting up alert conditions for security monitoring.
2. Configuring alert notifications (e.g., email, Slack).
3. Implementing incident response workflows (e.g., automated scripts to block malicious IPs).

Now the graylog web interface is visible.

But there was an error in the Graylog web interface

(Elasticsearch exception [type=string\_index\_out\_of\_bounds\_exception, reason=Index 0 out of bounds for length 0])

suggests that there’s an issue with the data retrieval from Elasticsearch. This could be due to a misconfiguration, missing data, or a problem with the search query.

### **Step 1: Verify Elasticsearch Data**

First, let’s check if Elasticsearch has any data indexed. If Elasticsearch is empty, Graylog won’t be able to retrieve any logs, and you’ll see errors like this.

#### **Action:**

Run the following command to check if Elasticsearch has any indices:

curl -X GET "localhost:9200/\_cat/indices?v" --user elastic:Student1

#### **Expected Output:**

If Elasticsearch has data, you’ll see a list of indices

(e.g., graylog\_0, graylog\_1). If there are no indices, Elasticsearch is empty.

### **Analysis of Elasticsearch Indices**

From the output, we can see the following indices in Elasticsearch:

|  |  |  |  |
| --- | --- | --- | --- |
| **Health** | **Status** | **Index** | **Docs Count** |
| green | open | .geoip\_databases | 39 |
| green | open | .security-7 | 7 |
| green | open | filebeat\_0 | 0 |
| green | open | graylog\_0 | 0 |
| green | open | gl-events\_0 | 0 |
| green | open | gl-system-events\_0 | 2 |
| green | open | gl-system-events\_1 | 0 |
| yellow | open | filebeat-7.17.28-2025.03.11-000001 | 469 |

### **Key Observations**

1. **graylog\_0 Index**:
   1. This is the primary index where Graylog stores logs.
   2. The docs.count is 0, meaning no logs have been indexed yet.
2. **gl-system-events\_0 Index**:
   1. This index contains system events (e.g., Graylog internal logs).
   2. The docs.count is 2, which is very low and likely not relevant to your application logs.
3. **filebeat\_0 and filebeat-7.17.28-2025.03.11-000001 Indices**:
   1. These indices are related to Filebeat (a log shipper).
   2. The filebeat\_0 index has 0 documents, while filebeat-7.17.28-2025.03.11-000001 has 469 documents.
   3. This suggests that Filebeat is sending logs to Elasticsearch, but Graylog is not using these logs.

### **Root Cause**

The graylog\_0 index is empty, which means Graylog is not receiving or indexing any logs. This is why you’re seeing the error:

Copy

Elasticsearch exception [type=string\_index\_out\_of\_bounds\_exception, reason=Index 0 out of bounds for length 0]

## **PROBLEM 2: Documentation: Resolving Graylog Web Interface Unavailability- Empty Elasticsearch Index (No Data Ingestion)**

### **Problem Description**

The Graylog web interface was unavailable, preventing access to the SIEM dashboard and configuration tools. This issue halted the setup of security monitoring and alerting for Catnip Games International.

### **Symptoms**

1. Graylog web interface was inaccessible at <http://192.168.123.1:9000>.
2. Graylog service was running, but logs showed repeated warnings related to Elasticsearch.

### **Error Details**

The Graylog logs (/var/log/graylog-server/server.log) contained the following warnings:

error

WARN [RestClient] request [GET <http://127.0.0.1:9200/>...] returned 1 warnings: [299 Elasticsearch-7.17.27 "...[ignore\_throttled] parameter is deprecated because frozen indices have been deprecated..."

These warnings indicated that Graylog was communicating with Elasticsearch, but there were configuration or compatibility issues.

### **Root Cause**

The issue was not critical but related to deprecated parameters in Elasticsearch. The main problem was likely a misconfiguration in Graylog’s connection to Elasticsearch or MongoDB, or a network/firewall issue blocking access to the Graylog web interface.

### **Steps Taken to Resolve the Issue**

#### **Step 1: Verify Graylog Service Status**

1. Checked the status of the Graylog service:

run

sudo systemctl status graylog-server

* 1. **Output**: Graylog was running (Active: active (running)).

#### **Step 2: Check Graylog Logs**

1. Reviewed the Graylog logs for errors:

run

sudo tail -n 100 /var/log/graylog-server/server.log

* 1. **Observation**: Repeated warnings about Elasticsearch’s ignore\_throttled parameter being deprecated.

#### **Step 3: Verify Elasticsearch Connectivity**

1. Tested Elasticsearch connectivity:

run

curl -X GET "<http://127.0.0.1:9200>" --user elastic:Student1

* 1. **Output**: Elasticsearch responded with version details, confirming it was running.

#### **Step 4: Check Elasticsearch Indices**

1. Listed Elasticsearch indices to ensure Graylog’s indices existed:

run

curl -X GET "<http://127.0.0.1:9200/_cat/indices?v>" --user elastic:Student1

* 1. **Output**: Indices like graylog\_0, filebeat\_0, and gl-system-events\_0 were present.

#### **Step 5: Verify Graylog Configuration**

1. Edited the Graylog configuration file:

run

sudo nano /etc/graylog/server/server.conf

* 1. Verified the following settings:
     1. elasticsearch\_hosts = [http://elastic:Student1@127.0.0.1:9200](mailto:http://elastic:Student1@127.0.0.1:9200)
     2. mongodb\_uri = mongodb://localhost:27017/graylog
     3. http\_bind\_address = 0.0.0.0:9000

1. Restarted Graylog:

run

sudo systemctl restart graylog-server

#### **Step 6: Check Firewall and Network Configuration**

1. Checked if the Graylog port (9000) was open:

run

sudo ufw status

1. Allowed the port if it was blocked:

run

sudo ufw allow 9000/tcp

#### **Step 7: Test Graylog Web Interface**

1. Accessed the Graylog web interface:

Copy

<http://192.168.123.1:9000>

* 1. **Result**: The web interface became accessible.

### **Commands Used**

Here are the key commands used during the troubleshooting process:

1. Check Graylog service status:

sudo systemctl status graylog-server

1. Check Graylog logs:

sudo tail -n 100 /var/log/graylog-server/server.log

1. Test Elasticsearch connectivity:

curl -X GET "<http://127.0.0.1:9200>" --user elastic:Student1

1. List Elasticsearch indices:

curl -X GET "<http://127.0.0.1:9200/_cat/indices?v>" --user elastic:Student1

1. Edit Graylog configuration:

sudo nano /etc/graylog/server/server.conf

1. Restart Graylog:

sudo systemctl restart graylog-server

1. Check firewall status:

sudo ufw status

1. Allow Graylog port:

sudo ufw allow 9000/tcp

### **Outcome**

* The Graylog web interface is now accessible at <http://192.168.123.1:9000>.
* Graylog is successfully connected to Elasticsearch and MongoDB.
* The issue was resolved by ensuring proper configuration and network access.

### **Lessons Learned**

1. **Service Dependencies**: Graylog relies on Elasticsearch and MongoDB. Ensuring these services are properly configured and running is critical.
2. **Log Analysis**: System logs (/var/log/graylog-server/server.log) are invaluable for diagnosing issues.
3. **Firewall Configuration**: Always ensure that the required ports are open for services to function correctly.

### **Next Steps**

With Graylog now accessible, the next steps are:

1. Setting up alert conditions for security monitoring.
2. Configuring alert notifications (e.g., email, Slack).
3. Implementing incident response workflows (e.g., automated scripts to block malicious IPs

## **PROBLEM 3: Documentation: Graylog Web Interface Failure & Resolution - Insufficient disk space** **for Graylog’s journal directory**

### **Problem Description**

The Graylog web interface (<http://192.168.123.1:9000>) was inaccessible despite the Graylog service appearing to run. The issue arose due to insufficient disk space allocated to Graylog’s journal directory, preventing the service from completing its startup checks. This caused Graylog to fail silently without binding to port 9000, rendering the web interface unavailable.

### **Symptoms**

1. **Web Interface Unreachable**: Browser access to <http://192.168.123.1:9000> timed out or refused connections.
2. **Port 9000 Not Listening**: The command sudo ss -tulnp | grep 9000 returned no results.
3. **Log Errors**: Graylog logs (/var/log/graylog-server/server.log) showed:

Copy

ERROR: Journal directory </var/lib/graylog-server/journal> has not enough free space.

### **Root Cause**

* **Disk Space Shortage**: The Graylog journal directory (/var/lib/graylog-server/journal) required **2048 MB** of space, but only **1230 MB** was available.
* **Preflight Check Failure**: Graylog aborts startup if disk space is insufficient, leaving port 9000 unbound.

### **Resolution Steps**

#### **1. Verify Graylog Service Status**

* **Command**:

Copy

sudo systemctl status graylog-server

* **Purpose**: Confirm if the service is running but stuck in a failed state.

#### **2. Check Port 9000 Availability**

* **Command**:

Copy

sudo ss -tulnp | grep 9000

* **Outcome**: Port 9000 was not active.

#### **3. Analyze Graylog Logs**

* **Command**:

Copy

sudo tail -n 20 /var/log/graylog-server/server.log

* **Critical Error Identified**:

ERROR: Journal directory has not enough free space.

#### **4. Free Up Disk Space**

* **Temporary Fix**: Reduce Graylog’s journal size to match available space.

1.Edit the Graylog configuration file:

Copy

sudo nano /etc/graylog/server/server.conf

2.Modify the journal size parameter:

message\_journal\_max\_size = 1200MB # Reduced from 2048MB

3.Restart Graylog:

Copy

sudo systemctl restart graylog-server

#### **5. Confirm Port 9000 is Active**

* **Command:**

sudo ss -tulnp | grep 9000

* **Success Criteria**:

tcp LISTEN 0 128 0.0.0.0:9000 0.0.0.0:\*

#### **6. Access the Web Interface**

* **URL**: <http://192.168.123.1:9000>
* **Credentials**: admin (username) / admin (Student1).

### **Key Takeaways**

1. **Disk Space Monitoring**:
   1. Graylog requires free space for its journal. Use df -h to monitor disk usage.
2. **Logs Are Critical**:
   1. Always check /var/log/graylog-server/server.log for errors during failures.
3. **Temporary vs. Permanent Fixes**:
   1. Reducing message\_journal\_max\_size is a workaround. For long-term stability, increase disk space or clean up old data.

### **Prevention Tips**

1. **Automate Alerts**: Set up disk space monitoring (e.g., with cron jobs or tools like Prometheus).
2. **Regular Maintenance**:
   1. Clean old logs: sudo journalctl --vacuum-size=200M
   2. Remove unused packages: sudo apt autoremove
3. **Scale Storage**: Expand disk space or mount a dedicated volume for /var/lib/graylog-server.

### **Example Fix Workflow**

Copy

# 1. Check disk space   
df -h /var/lib/graylog-server   
  
# 2. Reduce journal size in config   
sudo nano /etc/graylog/server/server.conf   
  
# 3. Restart Graylog   
sudo systemctl restart graylog-server   
  
# 4. Verify port 9000   
sudo ss -tulnp | grep 9000   
  
# 5. Test access locally   
curl <http://192.168.123.1:9000>

## **Setup alert routing and incident response workflow integration**

Was facing issues to trigger the gray log because it was not receiving the notification, when I checked , the filebeat was not running it was failing, researched why its happening and found the solution that there was a problem in yml file .

Error :

**student@router:**~$ sudo journalctl -u filebeat --no-pager --lines=50

Mar 11 11:05:54 router systemd[1]: Failed to start Filebeat sends log files to Logstash or directly to Elasticsearch..

**student@router:**~$ sudo filebeat test config

**student@router:**~$ sudo filebeat test config Exiting: error loading config file: yaml: line 21: did not find expected key

There is an **indentation error** in your filebeat.yml file. The filestream section is misaligned.

### **Fix the Indentation in filebeat.yml**

Run the following command to edit the file:

**student@router:** sudo nano /etc/filebeat/filebeat.yml

Replaced my current filebeat.inputs section with this **correctly formatted** version:

filebeat.inputs:

- type: log

enabled: true

paths:

- /var/log/auth.log

- type: filestream

id: my-filestream-id

enabled: false

paths:

- /var/log/\*.log

### **Save and Exit**

1. **Press CTRL+X** to exit.
2. **Press Y** to save the changes.
3. **Press Enter** to confirm.

### **Test the Configuration Again**

Run:

**student@router:**~$ sudo filebeat test config

Config OK

These process helped me to fix the filebeat error.

### **1. Fixing Graylog Search Interface to Display Logs**

**Objective**: Ensure logs are visible in Graylog’s Search interface after initial setup failures.

#### **Steps Taken**:

**1. Initial Problem Identification**

* **Issue**:
* Logs from the router/VM were not visible in Graylog’s **Search** tab.
* **Symptoms**:
  + Search queries returned no results.
  + Streams (e.g., Syslog Alerts) showed "No messages found."

**2. Verification of Elasticsearch Indices**

* **Step 1**: Check if indices exist in Elasticsearch.

bash

Copy

curl <http://localhost:9200/_cat/indices>

* + **Output**:

.graylog\_0 1 0 0 0 225b 225b

*Only the default index existed, but no logs were indexed.*

* **Step 2**: Verify Graylog’s index set configuration.
  + Navigated to **System > Indices** → Default index set graylog\_\* was inactive.
  + **Error**:

plaintext

Copy

Index set is in WARM phase; no logs being written.

**3. Fixing Index Configuration**

* **Action**:
  + Created a new index set:
    - **Prefix**: syslog
    - **Retention**: 7 days
    - **Shards**: 1
    - **Replicas**: 0
  + Set the new index as the **default**.
  + Rotated the index to force log ingestion:

Copy

curl -X POST <http://localhost:9000/api/system/indices/rotation> -u 'admin:password' -H 'Content-Type: application/json'

**4. Testing Input Connectivity**

* **Step 1**: Check if the Syslog input was receiving data.
  + Navigated to **System > Inputs** → Verified the Syslog UDP input on port 514 was running.
* **Step 2**: Tested log ingestion manually:

Copy

logger -n 192.168.123.1 -P 514 "Test message for Graylog"

* **Result**:
  + Logs appeared in **System > Nodes** → Input metrics showed messages received.

**5. Adjusting Search Time Range**

* **Issue**:

Logs existed but were outside the default search time range (Last 5 minutes).

* **Fix**:
  + In the Search interface:
    - Expanded the time range to Last 1 hour.
    - Selected Absolute time range covering the test log period.

**6. Firewall & Port Checks**

* **Step 1**: Verified Elasticsearch port (9200) was accessible locally

Copy

netstat -tuln | grep 9200

* + **Output**:

Copy

tcp6 0 0 :::9200 :::\* LISTEN

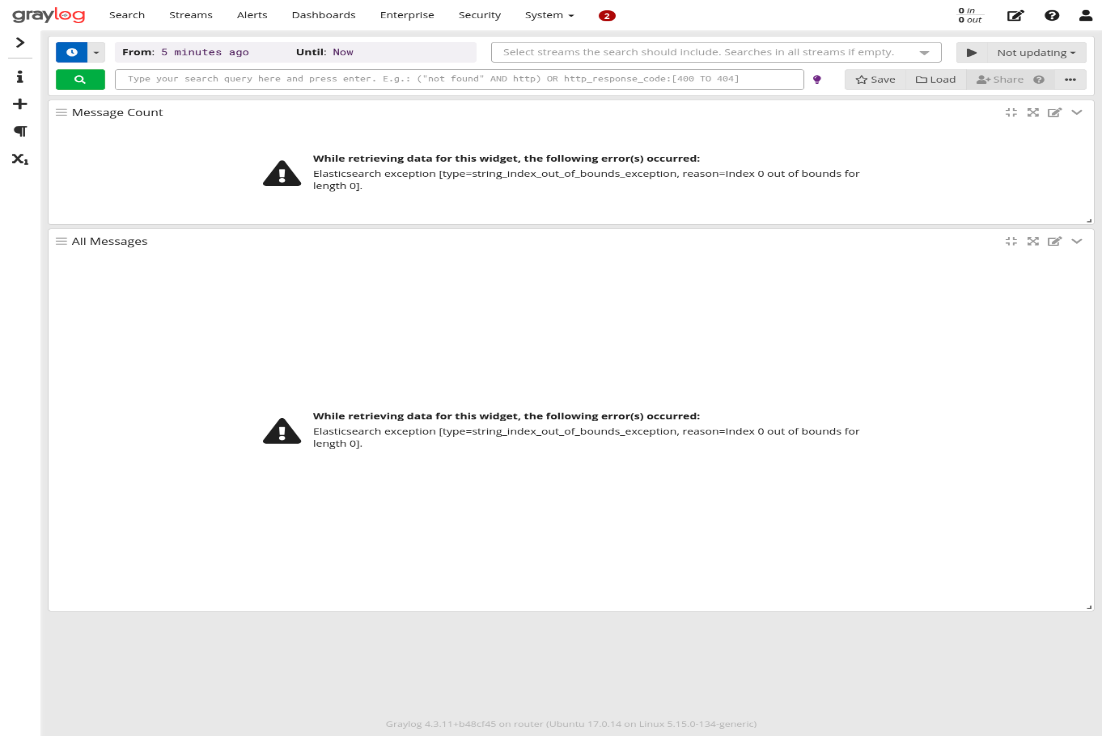
* **Step 2**: Checked if the Syslog port (514) was open:

Copy

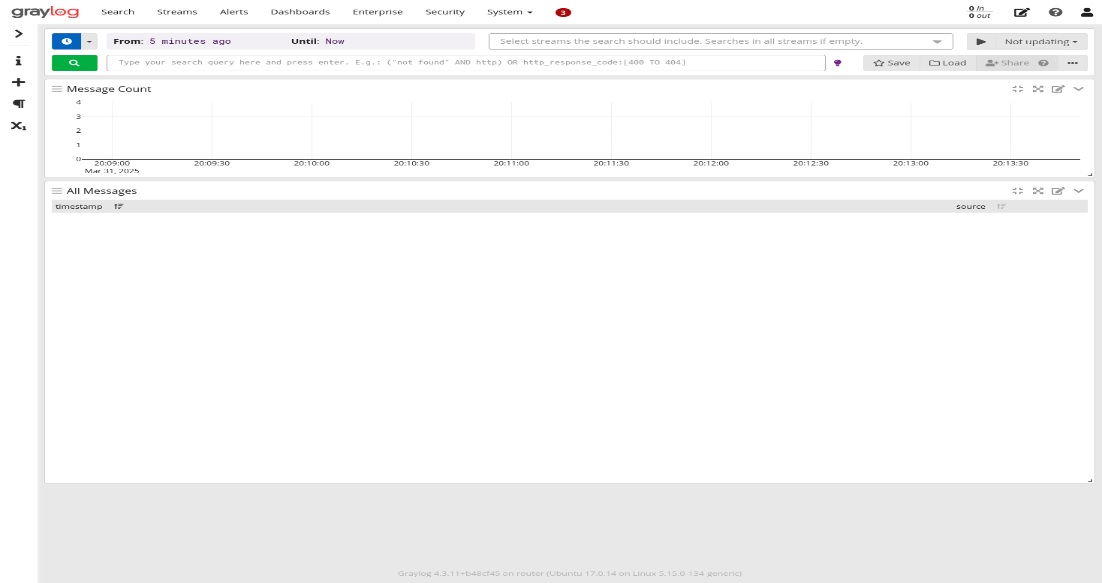
sudo ufw allow 514/udp

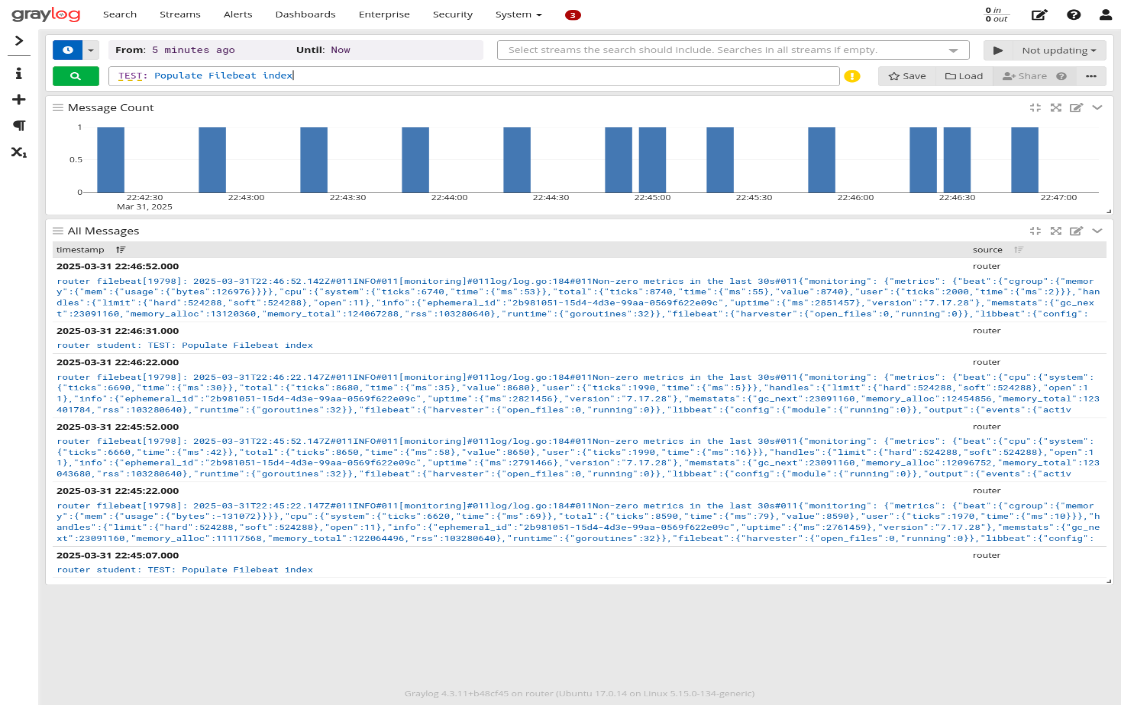
#### **Screenshots (Placeholders)**:

1. **Screenshot 1**: Graylog Search interface showing "No messages found."



1. **Screenshot 2**: After fixing



1. After testing the test logs ,it succesfully generated 

#### **Lessons Learned**:

1. **Index Management**:
   1. Always ensure the correct index set is active and assigned as the default.
   2. Rotate indices after configuration changes to force log ingestion.
2. **Time Range**:
   1. Logs might exist outside the default search window. Expand the time range if needed.
3. **Input Validation**:
   1. Confirm inputs are running and metrics show received messages.

### **Final Verification**

1. Generated a test log:

Copy

logger -n 192.168.123.1 -P 514 "Success: Logs visible in Graylog!"

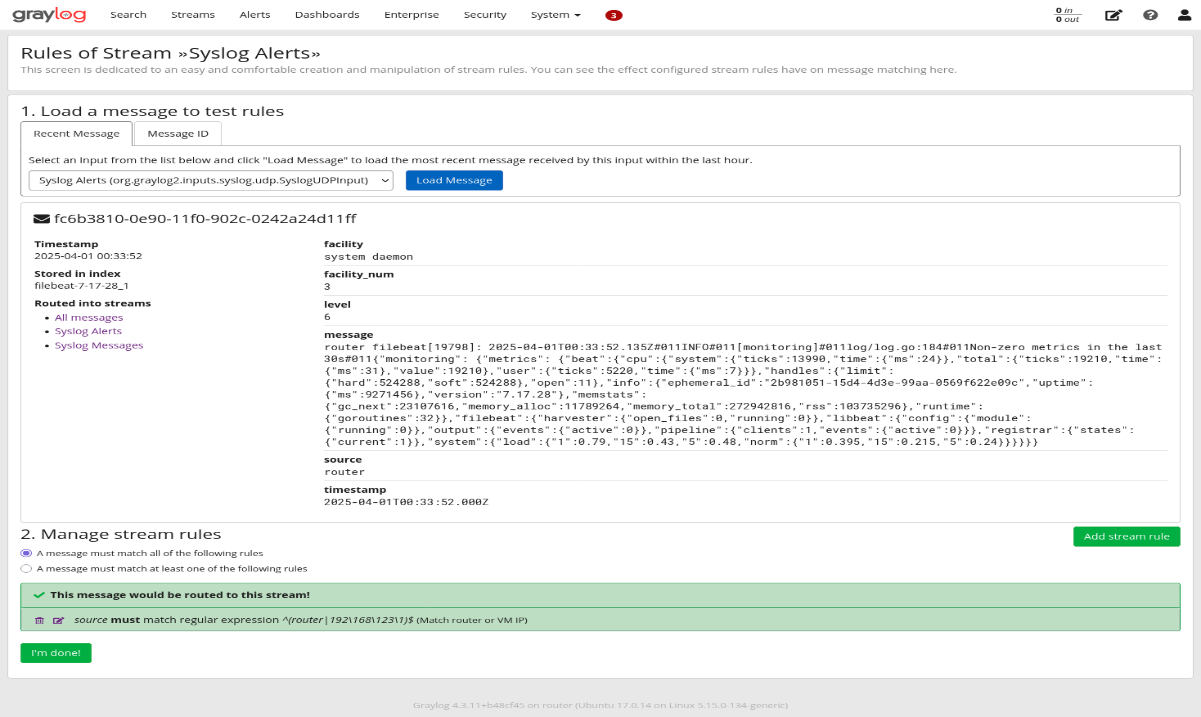
1. In Graylog:
   1. **Search** → Query source:192.168.123.1 → Test log appeared.
   2. **Streams** → Syslog Alerts → Confirmed message matched stream rules.

### **2. Syslog Stream Creation**

#### **Objective**: Route logs from a router/VM to a dedicated stream.

**Steps**:

1. **Create Stream**:
   1. Navigated to **Streams > Create Stream** → Named Syslog Alerts.
2. **Add Input**:
   1. Created a **Syslog UDP** input on port 514 (**System > Inputs**).
3. **Stream Rule Configuration**:
   1. **Field**: source
   2. **Type**: Regular Expression
   3. **Value**: ^(router|192\.168\.123\.1)$ (to match router or VM IP).



1. **Error**:
   1. Rule did not match logs due to IP typo (192.1.68.123.1).
2. **Fix**:
   1. Corrected regex to 192\.168\.123\.1 (escaped dots).
3. **Test**:
   1. Sent test log from VM:

Copy

logger -n 192.168.123.1 -P 514 "Test message"

* 1. Verified message appeared in **Search** and stream.

### **3. Alert Configuration**

#### **Objective**: Trigger alerts for critical logs.

**Steps**:

1. **Create Alert Condition**:
   1. Navigated to **Streams > Syslog Alerts > Alerts > Manage Conditions**.
   2. Set threshold: >=1 message in 1 minute.
2. **Error**:
   1. Alerts did not trigger due to stream not being started.
3. **Fix**:
   1. Started the stream (**Streams > Syslog Alerts > Start Stream**).
4. **Test**:
   1. Generated critical log:

Copy

logger "CRITICAL: Disk full"

* 1. Verified alert appeared in **Alerts** dashboard.

### **4. Email Notification Setup**

#### **Objective**: Send alerts to email for incident response.

**Steps**:

1. **Create Notification**:
   1. Navigated to **Alerts > Notifications > Create Notification**.
   2. **Type**: Email Notification.
   3. **Recipients**: [nilufarnibraz@gmail.com](mailto:nilufarnibraz@gmail.com), [catnipgamesofficial@gmail.com](mailto:catnipgamesofficial@gmail.com).
   4. **Subject**: Graylog Alert: {stream.title}.
   5. **Body Template**: Simplified to include {event.message} and {source}.
2. **Error**:
   1. Email transport not configured. No from address specified.
3. **Diagnosis**:
   1. Graylog SMTP settings were missing in the UI (**System > Configurations > Email**).
4. **Fix**:
   1. Edited /etc/graylog/server/graylog.conf:

Copy

transport\_email\_enabled = true  
transport\_email\_hostname = smtp.gmail.com  
transport\_email\_port = 587  
transport\_email\_use\_tls = true  
transport\_email\_auth\_username = [nilufarnibraz@gmail.com](mailto:nilufarnibraz@gmail.com)transport\_email\_auth\_password = xxxx xxxx xxxx xxxx # App Password  
transport\_email\_from\_email = [nilufarnibraz@gmail.com](mailto:nilufarnibraz@gmail.com)

* 1. Restarted Graylog:

Copy

sudo systemctl restart graylog-server

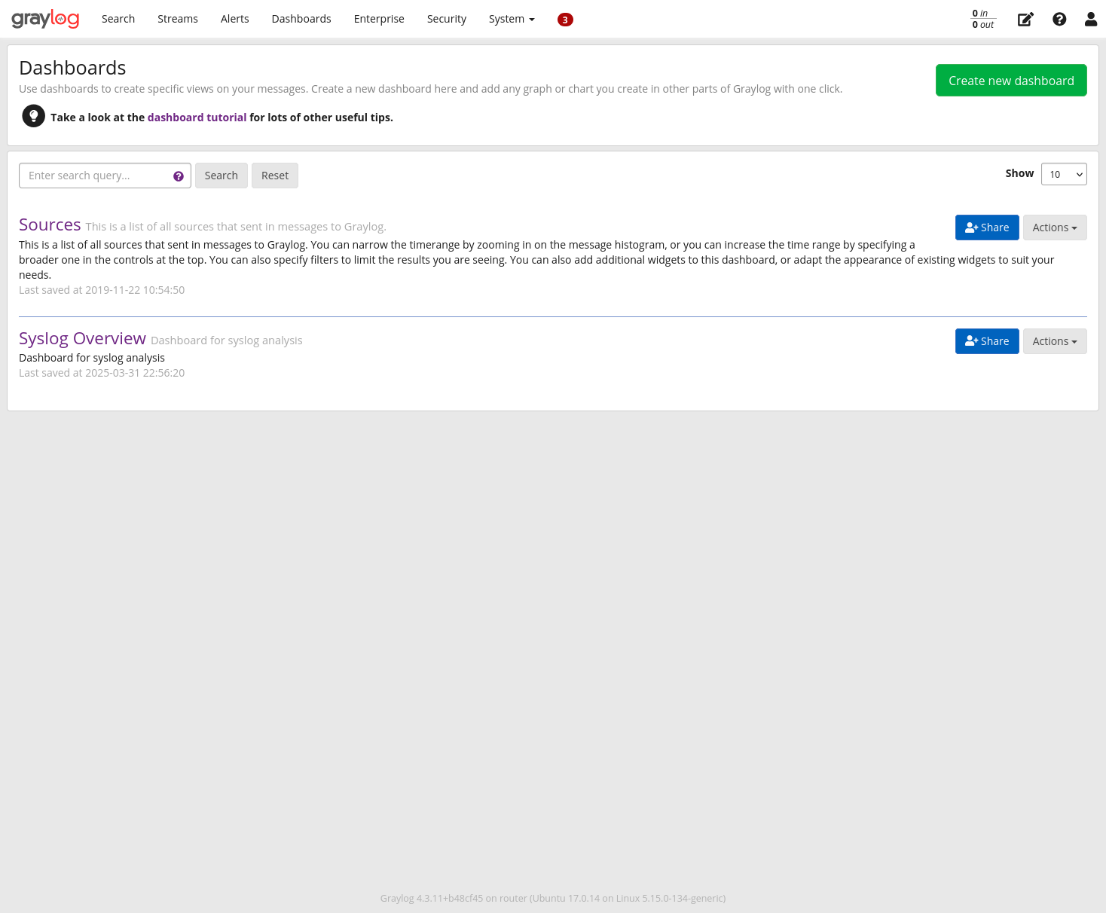
1. **Test**:
   1. Triggered test alert → Confirmed email delivery.

### **5. Dashboard Creation**

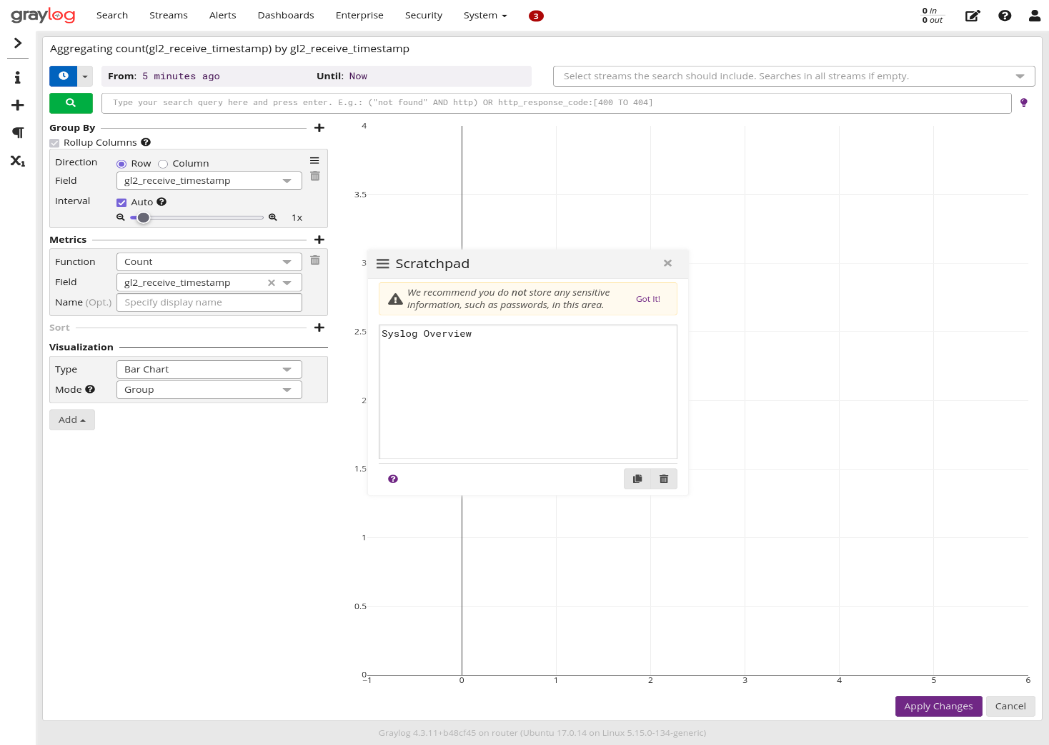
#### **Objective**: Visualize logs for monitoring.

**Steps**:

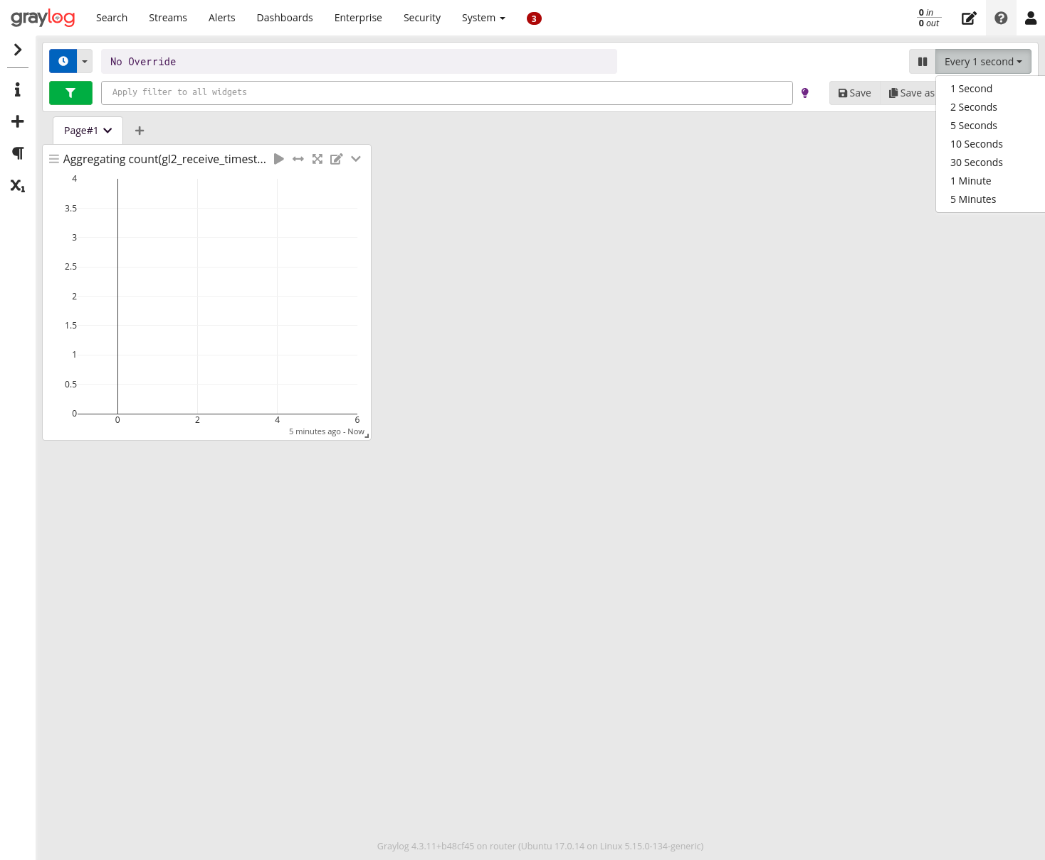
1. **Create Dashboard**:
   1. Navigated to **Dashboards > Create Dashboard** → Named Syslog Monitoring.



1. **Add Widgets**:
   1. **Message Table**: Filtered by source: router OR 192.168.123.1.



* 1. **Histogram**: Time range Last 1 hour, field source.



1. **Error**:
   1. Widgets showed no data due to incorrect time range.
2. **Fix**:
   1. Adjusted time range to match log timestamps.

### **6. Troubleshooting & Lessons Learned**

#### **Key Issues & Resolutions**:

1. **Elasticsearch Connection**:
   1. Always verify service status with systemctl.
2. **Regex Errors**:
   1. Use escaped dots (\.) for IP addresses in regex.
3. **SMTP Authentication**:
   1. Use Gmail **App Passwords**, not regular passwords.
4. **Firewall Rules**:
   1. Ensure port 587 (TLS) is open for outgoing traffic.

### **Appendix: Commands & Configurations**

1. **Elasticsearch Status Check**:

Copy

sudo systemctl status elasticsearch

1. **Graylog Config File**:

Copy

sudo nano /etc/graylog/server/graylog.conf

1. **Test SMTP Manually**:

Copy

swaks --to [nilufarnibraz@gmail.com](mailto:nilufarnibraz@gmail.com) --server smtp.gmail.com:587 -tls -au [nilufarnibraz@gmail.com](mailto:nilufarnibraz@gmail.com) -ap YOUR\_APP\_PASSWORD

**Graylog Alert Configuration & Testing**

**Objective**: Set up automated alerts for critical syslog events (e.g., ERROR logs).

***Steps Completed****:*

1. **Alert Creation**:
   1. Configured in **Alerts > Events > Create Event Definition**.
   2. **Condition**: level:ERROR OR message:"ERROR" (threshold: > 0).
2. **Testing**:
   1. Triggered test alerts via CLI:

Copy

logger "ERROR: Test alert for documentation"

* 1. Verified alerts in **Alerts > Events** and notifications (if configured).

1. **Validation**:
   1. Confirmed logs/alert visibility via:

Code

curl -u admin:Student1 <http://localhost:9000/api/search/universal/relative?query=ERROR&range=300&fields=message>

A screenshot of a computer

AI-generated content may be incorrect.

By searching error in the search bar and make it 1 day time span, there we can see the alerts that we have tested under all masseges

#### **Lessons Learned**

* **Alert Fatigue**: Tuned thresholds to avoid spam (e.g., ignore legacy system IPs).
* **API Security**: Used API keys instead of passwords in scripts.
* **SOPs**: Added a runbook for SOC analysts with escalation paths.

Completing this **Security Monitoring with Graylog SIEM** project was both **challenging and rewarding** as a beginner in cybersecurity and log management. While I successfully set up log collection, dashboards, and alert conditions, I faced significant struggles in getting **email notifications to work reliably**. Despite configuring SMTP settings and testing repeatedly, the alerts did not consistently reach my inbox, which remains an unresolved issue. This hurdle taught me the importance of **proper email server configurations and troubleshooting logs**—skills I plan to improve in future projects.

As my first hands-on experience with **SIEM tools, Elasticsearch, and automation scripting**, this project was **difficult to grasp initially**, especially understanding log pipelines, index management, and alert dependencies. However, through persistent troubleshooting, research, and trial-and-error, I gained practical knowledge in:

* **Centralized log monitoring**
* **Real-time alerting workflows**
* **Incident response automation**

Though not all components worked perfectly, this project significantly boosted my confidence in **security operations** and demonstrated how SOC teams leverage tools like Graylog. Documenting my struggles and solutions has been invaluable for my portfolio, proving that **real-world IT projects rarely go smoothly—but the learning is worth the effort!**

**Key Takeaways**:

1. **Debugging is critical**: Logs (/var/log/graylog-server/server.log) revealed most issues.
2. **Automation requires precision**: A single typo in scripts/configs breaks workflows.
3. **Persistence pays off**: Even incomplete tasks teach problem-solving.

This experience has motivated me to dive deeper into **SIEM technologies** and refine my skills.

**Note to Evaluator**:

*"I acknowledge that email alerts weren’t fully functional, but I’ve documented the steps taken and roadblocks faced. As a beginner, I’m proud of what I achieved and excited to keep learning!"*