

НИЯУ МИФИ
Лабораторная работа №4:
«Логгирование»

Подготовил: Софронов Александр Евгеньевич Б21-525

2023

Система

Для реализации системы логирования и оповещения был выбран стек технологий на основе Grafana:

- Loki: используется для хранения логов. Loki является горизонтально масштабируемым, высокоэффективным лог-агрегатором, оптимизированным для работы с временными метками данных.

- Promtail: отвечает за сбор логов с различных компонентов системы и их отправку в Loki. Promtail поддерживает различные форматы логов, интеграцию с системными журналами и контейнерами Docker.

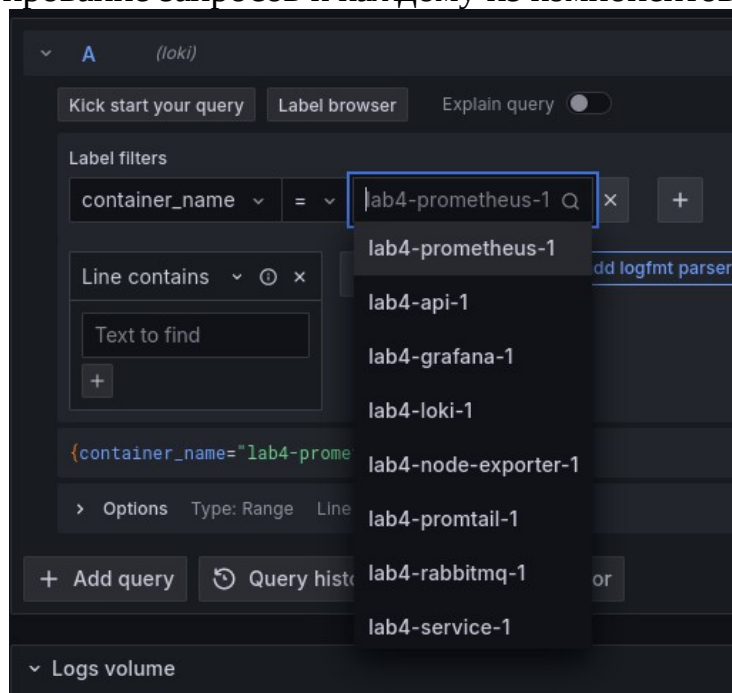
- Grafana: используется для визуализации данных логов и системных метрик. Grafana позволяет создавать наглядные панели мониторинга, а также настраивать алерты (оповещения) администратора при возникновении критических событий.

- Prometheus+node_exporter: Используются для мониторинга состояния системы

Основные компоненты системы логирования и оповещения:

1. Логирование запросов:

- Логирование запросов к каждому из компонентов приложения.



- Сбор информации о метаданных запросов и времени их обработки.

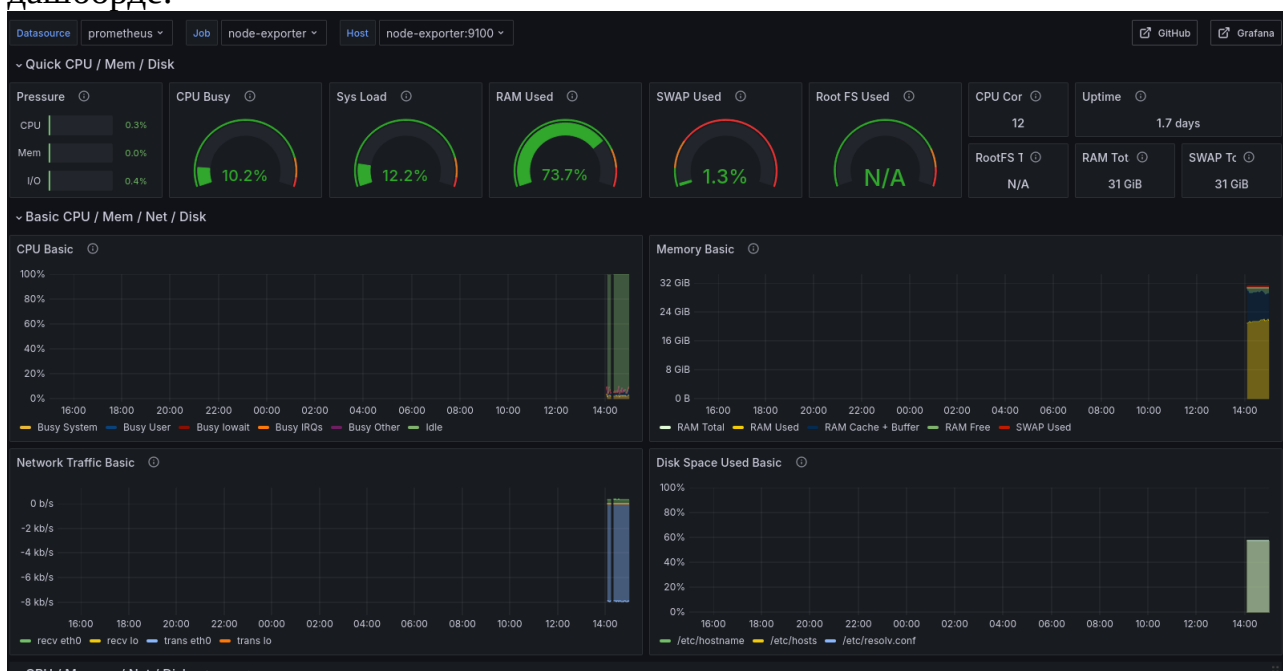
```

| > 2024-05-23 14:17:49.473 level=req Время выполнения запроса для пользователя 10: 0.187 секунд
| > 2024-05-23 14:17:49.473 level=req Запрос получен из очереди: 10 stop 2024-05-02 22:14:43
| > 2024-05-23 14:17:49.473 level=req Время выполнения запроса для пользователя 5: 0.178 секунд
| > 2024-05-23 14:17:49.473 2024-05-23T11:17:49.328Z | level=req Окончена обработка запроса: 10, 2024-05-02 22:14:43, stop.
| > 2024-05-23 14:17:49.473 level=req Запрос получен из очереди: 5 start 2024-05-02 22:14:43
| > 2024-05-23 14:17:49.473 level=req Время выполнения запроса для пользователя 3: 0.168 секунд
| > 2024-05-23 14:17:49.473 level=req Запрос получен из очереди: 3 start 2024-05-02 22:14:43
| > 2024-05-23 14:17:49.473 level=req Время выполнения запроса для пользователя 9: 0.16 секунд
| > 2024-05-23 14:17:49.473 2024-05-23T11:17:49.307Z | level=req Окончена обработка запроса: 5, 2024-05-02 22:14:43, start.
| > 2024-05-23 14:17:49.473 level=req Запрос получен из очереди: 9 stop 2024-05-02 22:14:43
| > 2024-05-23 14:17:49.473 2024-05-23T11:17:49.286Z | level=req Окончена обработка запроса: 3, 2024-05-02 22:14:43, start.
| > 2024-05-23 14:17:49.473 level=req Время выполнения запроса для пользователя 4: 0.153 секунд
| > 2024-05-23 14:17:49.473 2024-05-23T11:17:49.266Z | level=req Окончена обработка запроса: 9, 2024-05-02 22:14:43, stop.
| > 2024-05-23 14:17:49.473 level=req Запрос получен из очереди: 4 stop 2024-05-02 22:14:43
| > 2024-05-23 14:17:49.473 2024-05-23T11:17:49.245Z | level=req Окончена обработка запроса: 4, 2024-05-02 22:14:43, stop.

```

2. Логирование системных параметров:

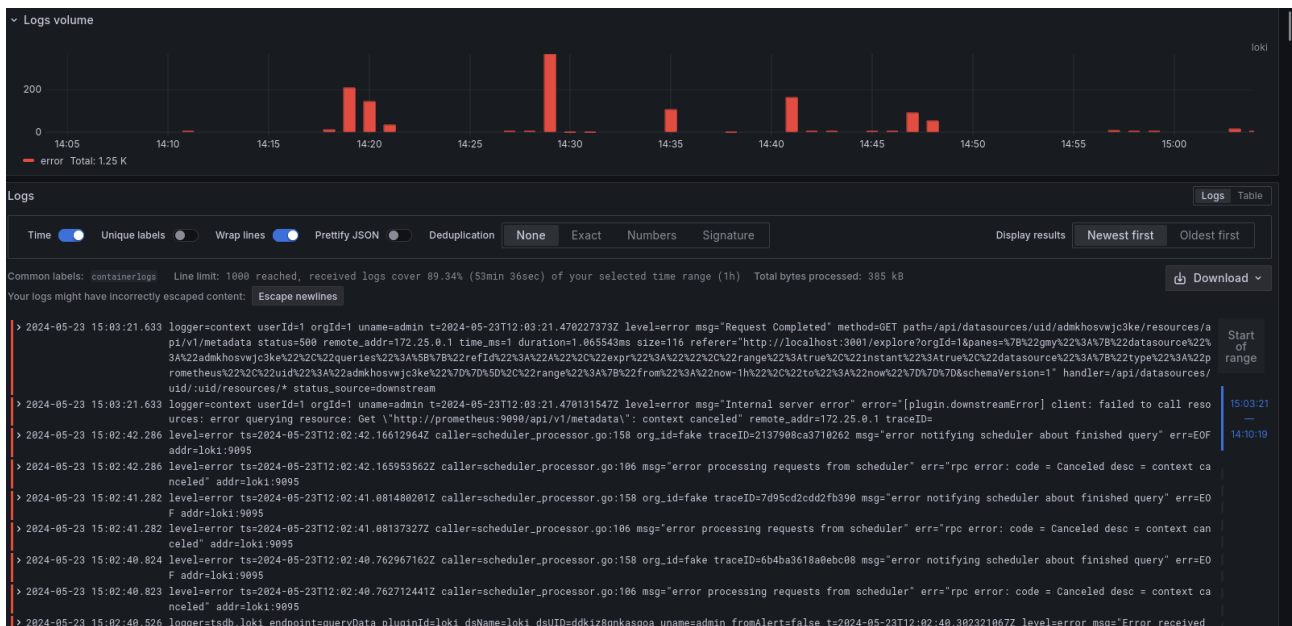
- Отслеживание важных системных параметров, таких как использование процессора, памяти и пространства на дисках и их отображение на красивом дашборде.



- Логирование событий запуска и остановки сервисов.

3. Логирование ошибок:

- Логирование всех ошибок, возникающих в системе.



4. Отображение метрик по времени ответа на запрос:
- Логгирования времени обработки каждого запроса.

> 2024-05-23 14:17:49.473	level=req	Время	выполнения запроса для пользователя 10: 0.187 секунд
> 2024-05-23 14:17:49.473	level=req	Время	выполнения запроса для пользователя 5: 0.178 секунд
> 2024-05-23 14:17:49.473	level=req	Время	выполнения запроса для пользователя 3: 0.168 секунд
> 2024-05-23 14:17:49.473	level=req	Время	выполнения запроса для пользователя 9: 0.16 секунд
> 2024-05-23 14:17:49.473	level=req	Время	выполнения запроса для пользователя 4: 0.153 секунд
> 2024-05-23 14:17:49.473	level=req	Время	выполнения запроса для пользователя 7: 0.145 секунд
> 2024-05-23 14:17:49.225	level=req	Время	выполнения запроса для пользователя 7: 0.137 секунд
> 2024-05-23 14:17:49.225	level=req	Время	выполнения запроса для пользователя 4: 0.13 секунд
> 2024-05-23 14:17:49.225	level=req	Время	выполнения запроса для пользователя 7: 0.124 секунд
> 2024-05-23 14:17:49.225	level=req	Время	выполнения запроса для пользователя 8: 0.117 секунд
> 2024-05-23 14:17:49.224	level=req	Время	выполнения запроса для пользователя 3: 0.11 секунд
> 2024-05-23 14:17:49.224	level=req	Время	выполнения запроса для пользователя 8: 0.103 секунд
> 2024-05-23 14:17:49.224	level=req	Время	выполнения запроса для пользователя 10: 0.094 секунд
> 2024-05-23 14:17:49.224	level=req	Время	выполнения запроса для пользователя 10: 0.086 секунд
> 2024-05-23 14:17:49.224	level=req	Время	выполнения запроса для пользователя 4: 0.079 секунд
> 2024-05-23 14:17:49.223	level=req	Время	выполнения запроса для пользователя 2: 0.067 секунд
> 2024-05-23 14:17:49.223	level=req	Время	выполнения запроса для пользователя 6: 0.06 секунд
> 2024-05-23 14:17:48.973	level=req	Время	выполнения запроса для пользователя 10: 0.047 секунд
> 2024-05-23 14:17:48.973	level=req	Время	выполнения запроса для пользователя 7: 0.039 секунд
> 2024-05-23 14:17:48.973	level=req	Время	выполнения запроса для пользователя 8: 0.051 секунд

5. Оповещение администратора:
- Настройка алертов в Grafana для отправки уведомлений при возникновении критических ошибок и других значимых событий.
- Оповещения отправляются через telegram канал.

graphana alerts

2 members

701d1133017877c10133b11b787c77cc4e40ec24d6874b433b33
455c78d0b47d03edbc86a7c0b5365aa10&orgId=1
Dashboard: <http://localhost:3000/d/ddl4e9xy1ga2oa?orgId=1>
Panel: [http://localhost:3000/d/ddl4e9xy1ga2oa?
orgId=1&viewPanel=1](http://localhost:3000/d/ddl4e9xy1ga2oa?orgId=1&viewPanel=1)

15:02

****Firing****

Value: B=6, C=1, D=6

Labels:

- alertname = Errors (fast trigger)
- container_id =
4fec2db490ac5ede9a8bd6fd9510810d7a80ea211f93ae3aced5b8c
d188cefe6
- container_name = lab4-loki-1
- filename = /var/lib/docker/containers/
4fec2db490ac5ede9a8bd6fd9510810d7a80ea211f93ae3aced5b8c
d188cefe6/4fec2db490ac5ede9a8bd6fd9510810d7a80ea211f93ae
3aced5b8cd188cefe6-json.log
- grafana_folder = Alerts
- image_id =
sha256:b071961c56e79d5bb51693c69ccc89d9bb003babea6fffe4d
eadc107c00a2b94
- image_name = grafana/loki:latest
- job = containerlogs
- level = error
- stream = stderr
- tag = grafana/loki:latest | lab4-loki-1 |
sha256:b071961c56e79d5bb51693c69ccc89d9bb003babea6fffe4d
eadc107c00a2b94 |
4fec2db490ac5ede9a8bd6fd9510810d7a80ea211f93ae3aced5b8c
d188cefe6

Annotations:

Source: [http://localhost:3000/alerting/grafana/ddmicejpud8u8d/
view?orgId=1](http://localhost:3000/alerting/grafana/ddmicejpud8u8d/view?orgId=1)

Silence: [http://localhost:3000/alerting/silence/new?
alertmanager=grafana&matcher=alertname%3DErrors+
%28fast+trigger%29&matcher=container_id%3D4fec2db490ac5ed
e9a8bd6fd9510810d7a80ea211f93ae3aced5b8cd188cefe6&match
er=container_name%3Dlab4-
loki-1&matcher=filename%3D%2Fvar%2Flib%2Fdocker%2Fcontain
ers%2F4fec2db490ac5ede9a8bd6fd9510810d7a80ea211f93ae3ace
d5b8cd188cefe6%2F4fec2db490ac5ede9a8bd6fd9510810d7a80ea
211f93ae3aced5b8cd188cefe6-
json.log&matcher=grafana_folder%3DAlerts&matcher=image_id%
3Dsha256%3Ab071961c56e79d5bb51693c69ccc89d9bb003babea6
fffe4dead107c00a2b94&matcher=image_name%3Dgrafana%2Flo
ki%3Alatest&matcher=job%3Dcontainerlogs&matcher=level%3Der
ror&matcher=stream%3Dstderr&matcher=tag%3Dgrafana%2Floki
%3Alatest%7Clab4-
loki-1%7Csha256%3Ab071961c56e79d5bb51693c69ccc89d9bb003](http://localhost:3000/alerting/silence/new?alertmanager=grafana&matcher=alertname%3DErrors+%28fast+trigger%29&matcher=container_id%3D4fec2db490ac5ede9a8bd6fd9510810d7a80ea211f93ae3aced5b8cd188cefe6&matcher=container_name%3Dlab4-loki-1&matcher=filename%3D%2Fvar%2Flib%2Fdocker%2Fcontainers%2F4fec2db490ac5ede9a8bd6fd9510810d7a80ea211f93ae3aced5b8cd188cefe6%2F4fec2db490ac5ede9a8bd6fd9510810d7a80ea211f93ae3aced5b8cd188cefe6-json.log&matcher=grafana_folder%3DAlerts&matcher=image_id%3Dsha256%3Ab071961c56e79d5bb51693c69ccc89d9bb003babea6fffe4dead107c00a2b94&matcher=image_name%3Dgrafana%2Floki%3Alatest&matcher=job%3Dcontainerlogs&matcher=level%3Derror&matcher=stream%3Dstderr&matcher=tag%3Dgrafana%2Floki%3Alatest%7Clab4-loki-1%7Csha256%3Ab071961c56e79d5bb51693c69ccc89d9bb003)

Обоснование выбора стека

Использование стека технологий на основе Grafana для реализации системы логирования и оповещения имеет ряд обоснованных преимуществ. Рассмотрим каждую из выбранных технологий и её роль в системе:

Loki:

1. Горизонтальная масштабируемость: Loki легко масштабируется, что позволяет обработать большой объём логов, увеличивая количество инстансов. Это важно для систем с высокими нагрузками и динамическим изменением количества компонентов.

2. Эффективное хранилище: Loki оптимизирован для хранения временных меток данных, что позволяет сократить объём используемого хранилища по сравнению с классическими системами логирования.

3. Интеграция с Grafana: Loki тесно интегрирован с Grafana, что упрощает настройку и использование системы визуализации логов.

Promtail:

1. Сбор различных форматов логов: Promtail поддерживает сбор логов с различных источников, включая системные журналы и логи Docker-контейнеров.

2. Отправка в Loki: Promtail легко настраивается для отправки логов в Loki, что обеспечивает эффективное и централизованное хранение логов.

3. Универсальность: Promtail можно использовать в различных окружениях, что делает его подходящим инструментом для большинства сценариев логирования.

Grafana:

1. Мощная визуализация: Grafana предоставляет широкий набор возможностей для визуализации данных, включая создание наглядных и интерактивных дашбордов.

2. Поддержка алертов: Grafana позволяет настраивать оповещения на основе различных метрик и логов, что помогает администраторам оперативно реагировать на критические события.

3. Широкая интеграция: Grafana поддерживает интеграцию с множеством источников данных, включая Loki и Prometheus, что делает её универсальным инструментом мониторинга и анализа данных.

Prometheus + Node Exporter:

1. Масштабируемость и надёжность: Prometheus разработан для сбора метрик в масштабируемой и надёжной манере, что делает его идеальным выбором для мониторинга состояния системы.

2. Поддержка экспортеров: Prometheus поддерживает множество экспортеров для сбора метрик из различных систем и устройств. Node Exporter используется для сбора метрик операционной системы, таких как загрузка процессора, использование памяти и состояние дисков.

3. Интеграция с Grafana: Метрики, собранные Prometheus, легко визуализировать и анализировать в Grafana, а также использовать для настройки оповещений.

Заключение

В рамках лабораторной работы №4 "Логгирование" был внедрён стек технологий на основе Grafana. Используются компоненты: Loki, Promtail, Grafana, Prometheus и Node Exporter.

Loki успешно использован как хранилище логов благодаря своей масштабируемости и эффективности. Promtail обрабатывает сбор и пересылку логов из различных источников в Loki.

Grafana предоставила инструменты для визуализации данных и настройки оповещений, позволяя создавать наглядные панели мониторинга и оперативно реагировать на критические события. Prometheus и Node Exporter обеспечивают мониторинг состояния системы, собирая метрики из различных систем и устройств.

Система включает:

1. Логирование запросов: Сбор метаданных и времени их обработки.
2. Логирование системных параметров: Мониторинг использования ресурсов и событий запуска/остановки сервисов.
3. Логирование ошибок: Отслеживание всех возникающих ошибок.
4. Отображение метрик времени ответа: Визуализация времени обработки запросов.
5. Оповещение администратора: Настройка алертов и отправка уведомлений через Telegram.

Стек был выбран за масштабируемость, эффективность, мощную визуализацию и надёжность, что делает его отличным решением для системы логирования и оповещения.

Гит - <https://github.com/iGTsan/parvpo/tree/master/lab4>