

1. Описание поиска логов на базе Grafana и Loki Logs

Loki Grafana — это система сбора и хранения логов, разработанная компанией Grafana Labs. Она оптимизирована для интеграции с Grafana и предназначена для эффективного поиска, просмотра и анализа логов. В отличие от традиционных систем, Loki индексирует только метаданные логов, а не их содержимое, что делает его масштабируемым и экономит ресурсы.

В Cloudlink система Loki интегрирована с логами всех микросервисов а так же логами Kubernetes. В этой статье описан способ поиска и сбора логов микросервисов в системе Loki

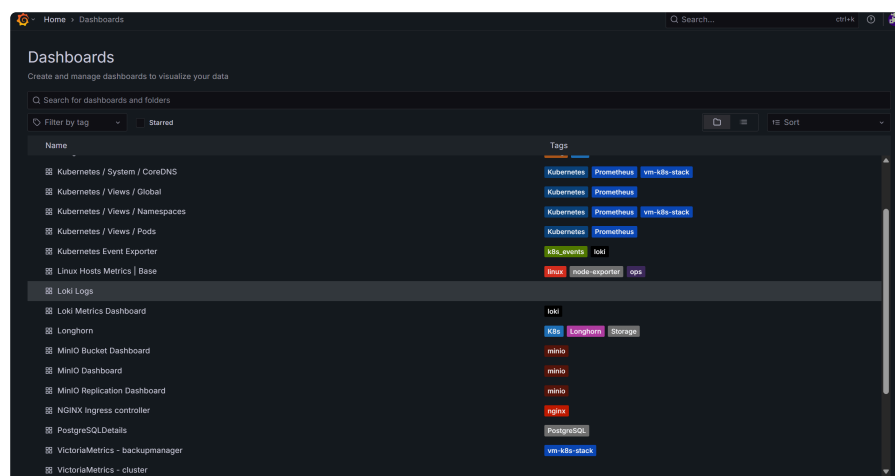
2. Просмотр логов

В инфраструктуре Cloudlink в каждой среде развернута своя Grafana. Чтобы проверить логи в определенной среде, используйте следующий формат запроса:

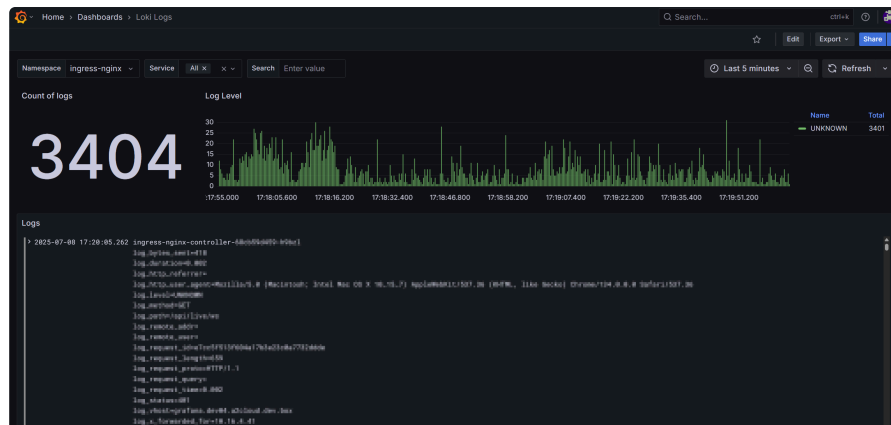
```
https://grafana.{DOMAIN_INT}
```

2.1. Просмотр логов на Loki Dashboard

1. Войдите в веб-интерфейс Grafana под своей учётной записью.
2. Нажмите на логотип Grafana для перехода в меню и выберите **Dashboards**.
3. В списке дашбордов нажмите на **Loki Logs**. Для удобства поиска Loki Logs воспользуйтесь поисковой строкой.

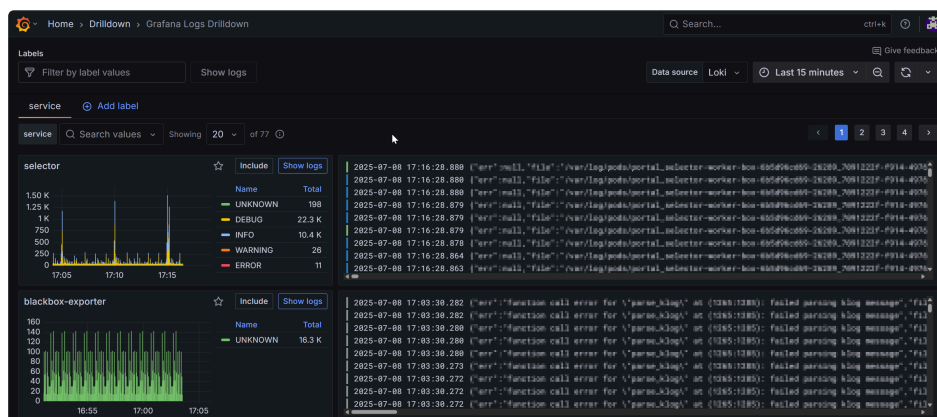


Откроется панель логов. Отфильтруйте логи по **Namespace**, **Service**, и **Search**.



2.2. Просмотр логов в DrillDown

Для более подробного просмотра логов используйте **Drilldown**. Для этого перейдите **Grafana** → **Menu** → **Drilldown** и нажмите **Logs**.



3. Сценарии логирования

3.1. Сценарий 1: Поиск ошибок с кодом 500 или Error в Loki Grafana

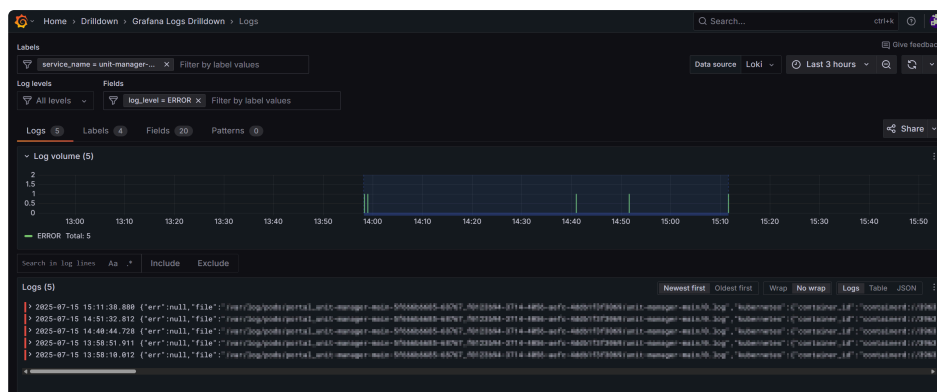
В этом сценарии описан способ, как определить сервис, в котором произошёл сбой, в логах Loki Grafana и получить стек-трейс.

Шаги:

1. Перейдите в **Grafana** → **Menu** и выберите **DrillDown** → **Logs**.
2. В правой части экрана выберите:
 - **Data source** — **Loki**.
 - **Time range** — выберите время, за которое нужно просмотреть логи.
3. В левой части экрана выберите сервис или сервисы для поиска логов и критерий поиска:

- **Labels** — выберите `service_name =` и выберите название вашего сервиса.
- **Fields** — выберите `log-level = ERROR` или `log-level = 500`

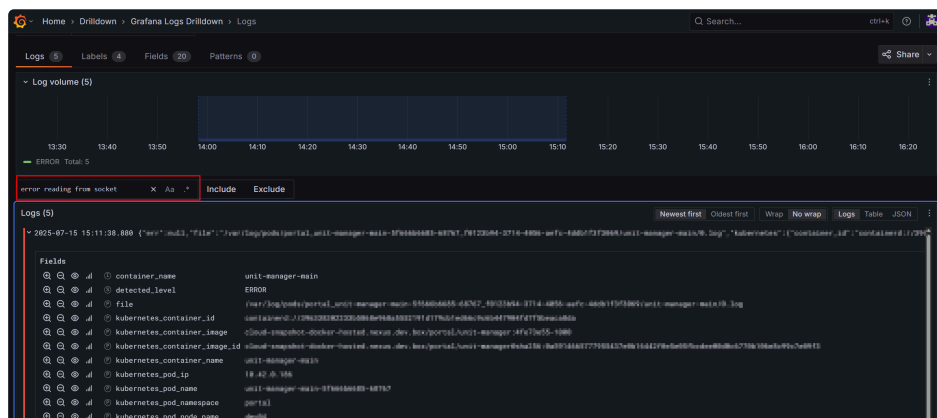
В результате отобразятся все ошибки, которые были выявлены в работе сервиса за указанное время.



Для поиска по специфическим ошибкам:

1. Раскройте запись лога и скопируйте нужную строку или слово.
2. Вставьте эту информацию в строку **Search in log lines**.

Так во всех логах будет найдена искомая строка или слово.



3.2. Сценарий 2: Поиск информации по заказу на портале в логах микросервисов

В этом сценарии описан способ просмотра информации по процессу развертывания заказа в логах микросервисов с помощью `x-request-id` заказа. Этот сценарий подходит для случаев, когда есть проблема с созданием заказа на портале и появилась ошибка.

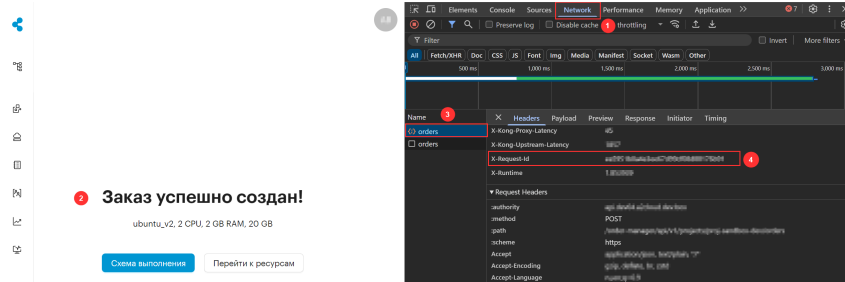
1. Начните создание заказа на портале Cloudlink: заполните необходимые поля и перед нажатием кнопки "Заказать" откройте панель разработчика — клавиша `F12` или правая кнопка мыши → **Исследовать элемент**:
 - a. В панели разработчика перейдите на вкладку **Network**.
 - b. Не закрывая панель разработчика, нажмите в вашем заказе кнопку **Заказать**.

с. В панели разработчика в колонке **Name** нажмите на появившуюся строку **orders**.

d. В колонке напротив — в **Headers** пролистайте до идентификатора **X-Request-ID** и скопируйте его значение.



X-Request-ID — уникальный идентификатор запроса, который используется для трассировки всех операций, связанных с этим запросом. Он необходим для поиска логов в Grafana Loki по заказу, находящемуся в процессе разворачивания на портале.

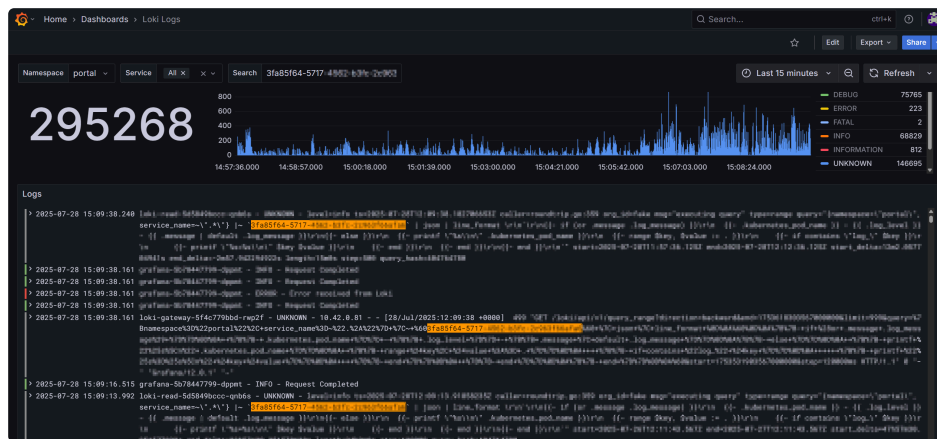


2. Откройте Grafana и перейдите в **Dashboards** → **Loki Logs**.

3. Укажите следующие параметры фильтрации:

- **Namespace** — выберите сервис. Например, "portal".
- **Service** — укажите "all".
- **Value** — введите скопированное ранее значение **X-Request-ID**.

4. Нажмите **Search**.



В логах по этому запросу доступна следующая информация:

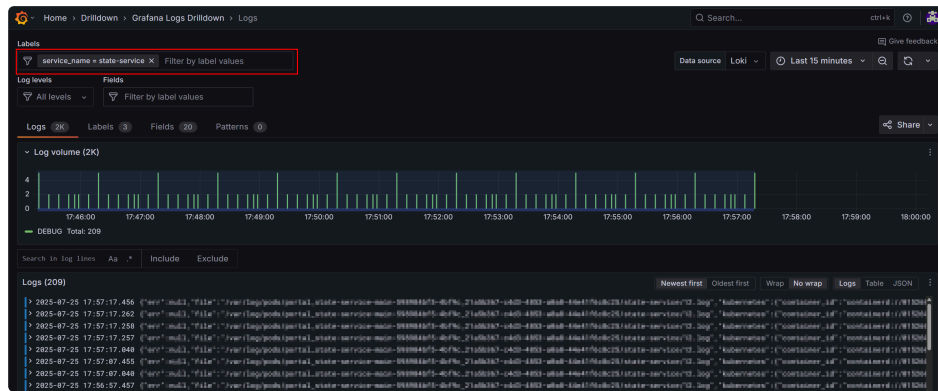
- зарезервированные GPU;
- выделенные ресурсы;
- отправленные сообщения в оркестратор;
- отклики других сервисов и прочее.

3.3. Сценарий 3: Просмотр логов сервиса

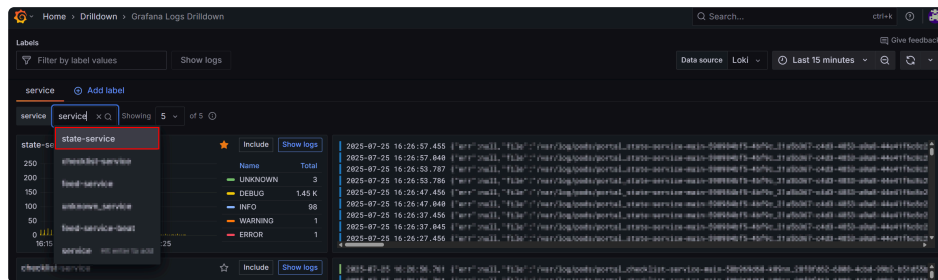
В этом сценарии описан пример поиска логов конкретного сервиса в Loki Grafana на примере сервиса State Service.

Шаги:

1. Перейдите в **Grafana** → **Menu** и выберите **DrillDown** → **Logs**.
2. В левой части экрана введите в поле **Service** название сервиса и нажмите на него. После этого откроется дашборд и логи этого сервиса за выбранный период.



Для просмотра подробного отчета, раскройте логи.



3.4. Сценарий 4. Анализ времени отклика микросервисов в Loki Grafana

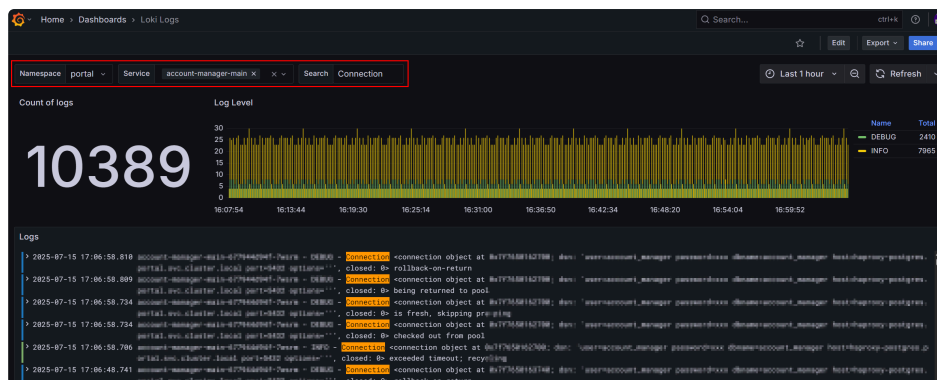
В этом сценарии рассматривается способ выявления в логах Loki Grafana снижения производительности микросервисов.

Шаги:

1. Перейдите в **Grafana** → **Menu** и выберите **Dashboards** → **Loki Logs**.
2. В правой части экрана выберите:
 - **Time range** — выберите время, за которое нужно просмотреть логи.
3. В левой части экрана выберите сервис или сервисы для поиска логов и критерий поиска:
 - **Namespace** — выберите сервис. Например, "portal".
 - **Service** — выберите микросервис, в котором нужно просмотреть логи. Например, "account-manager".

- **Value** — введите значение для поиска. В этом примере поиск происходит по слову "Connection", чтобы найти все логи, связанные с временем отклика микросервиса.

4. Нажмите **Search**.



4. Архитектура логирования

Для записи логов в Loki Grafana используется агент сбора логов — **Vector** со следующими компонентами:

- **vector** — сборщик логов на сервере.
- **vector-aggregator** — агрегатор логов, полученных от **vector**.

Запись логов:



1. Vector (на каждом узле):

- Собирает логи из файлов и логи Kubernetes.
- Добавляет метки (labels), которые используются в запросах.
- Отправляет логи на vector-aggregator.

2. Vector-aggregator

- Получает логи от всех серверов Vector.
- Отправляет логи через HTTP-запрос в Gateway.

3. Gateway (reverse proxy)

- Получает запрос и направляет его внутрь системы Loki по запросу write.

4. Loki (write)

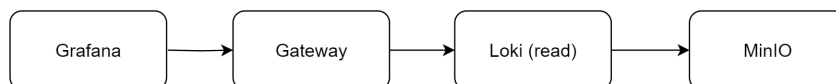
- Принимает поток логов.

- Отдаёт их в объектное хранилище MinIO.

5. MinIO

- Сохраняет логи.

Чтение логов:



1. Grafana

- В интерфейсе Grafana пользователь делает запрос данных к Loki.
- Запрос уходит в gateway.

2. Gateway

- Получает LogQL-запрос от Grafana.
- Перенаправляет его в компонент Loki — read.

3. Loki (read)

- Анализирует логи.
- Ищет их в MinIO.

4. MinIO

- Передает логи по запросу из хранилища в Grafana.