

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»
Факультет инфокоммуникационных технологий

Отчет по лабораторной работе №1
по дисциплине «Администрирование компьютерных сетей»

Выполнили: Шурубова Прасковья К3343
Исмагилова Карина К3344
Преподаватель: Самохин Н. Ю.

Санкт-Петербург

2024 г.

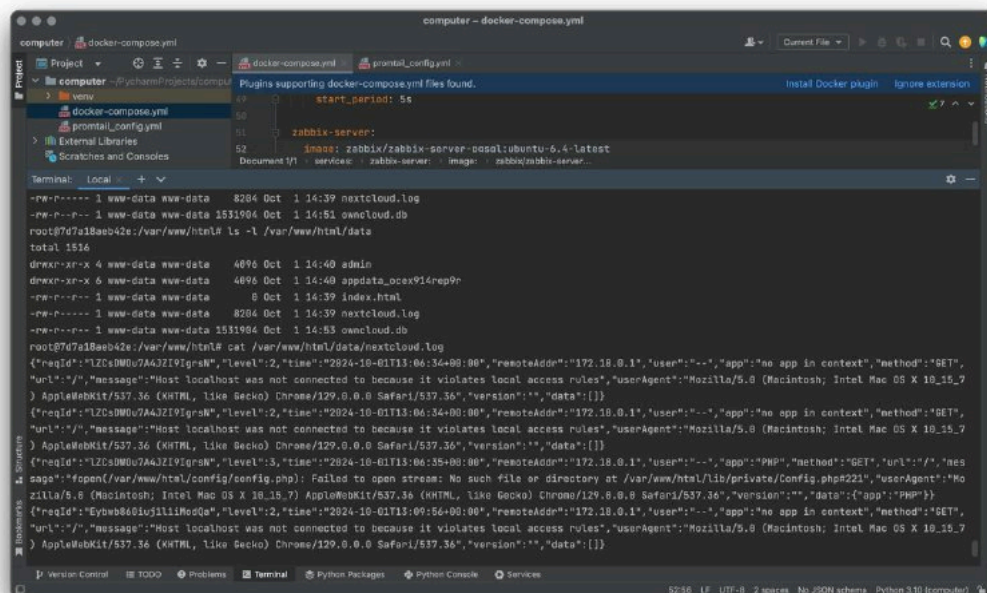
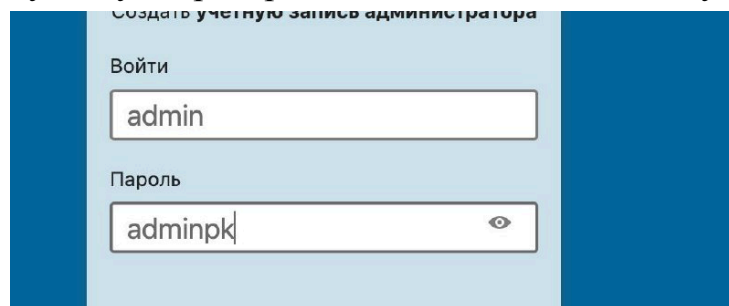
1. Логирование

Создали файл `docker-compose.yml`, который содержит в себе тестовый сервис Nextcloud, Loki, Promtail, Grafana, Zabbix и Postgres. Создали файл `promtail_config.yml`. Поменяли в начале скрипта `http_listen_port` на 3100.

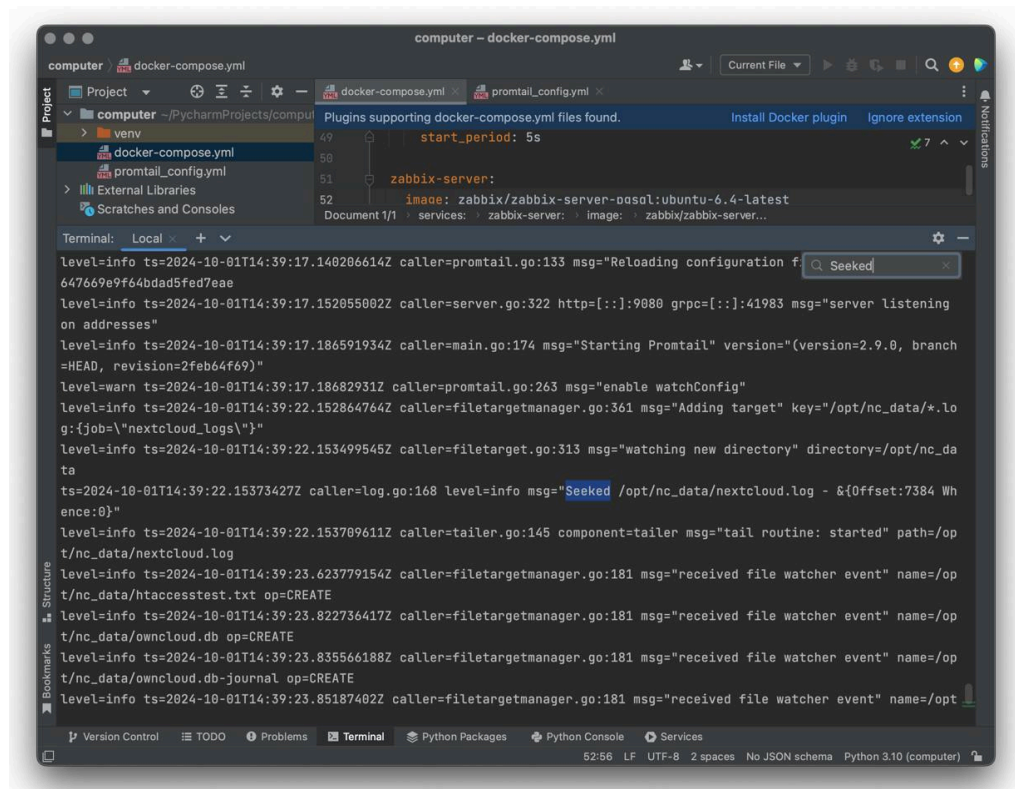
Далее запустили `compose` файл, и у нас всё “взлетело”.

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
6acc35096c36	zabbix/zabbix-server-pgsql:ubuntu-6.4-latest	"/usr/bin/tini -- /u..."	2 minutes ago	Up 2 minutes	0.0.0.0:10051->10051/tcp	zabbix
-back	zabbix/zabbix-web-mysql:ubuntu-6.4-latest	"docker-entrypoint.sh"	2 minutes ago	Up 2 minutes	8443/tcp, 0.0.0.0:8082->8080/tcp	zabbix
-front	postgres:15	"docker-entrypoint.s..."	2 minutes ago	Up 2 minutes (healthy)	5432/tcp	postgr
329e15468fe8	grafana/promtail:2.9.8	"/usr/bin/promtail -..."	2 minutes ago	Up 2 minutes		promta
5cfe22aa6495	grafana/loki:2.9.8	"/usr/bin/loki -conf..."	2 minutes ago	Up 2 minutes	0.0.0.0:3100->3100/tcp	loki
il	grafana/grafana:11.2.8	"/run.sh /run.sh"	2 minutes ago	Up 2 minutes	0.0.0.0:3000->3000/tcp	grafan

Зашли на веб интерфейс через внешний порт, указанный в `compose` файле, создали учётку и проверили, что логи “пошли” в нужный файл.



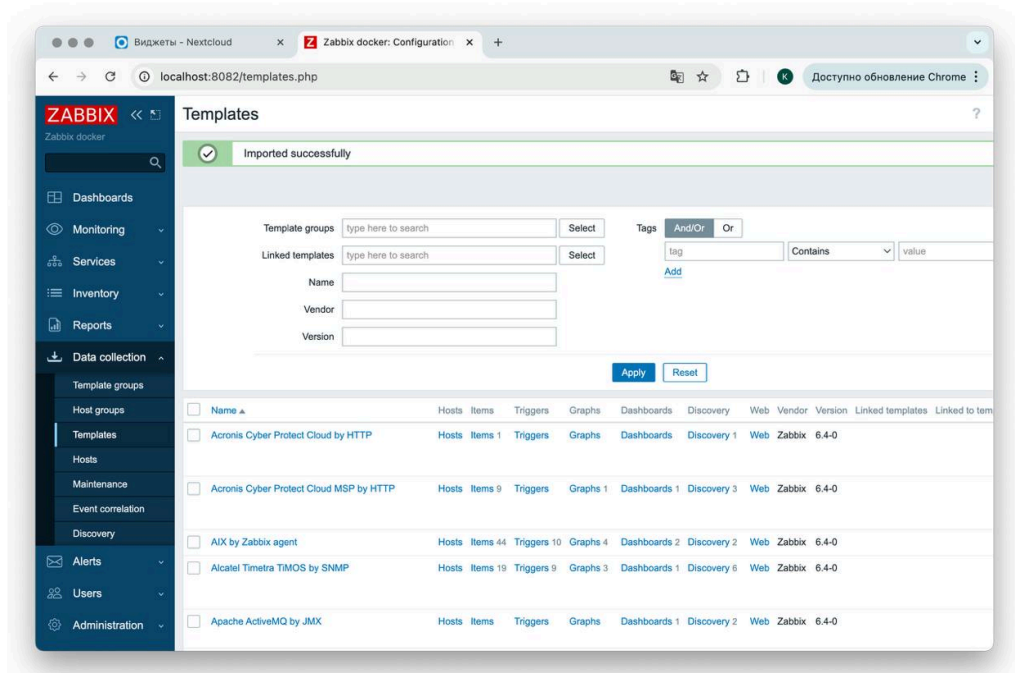
После инициализации Nextcloud проверяем в логах promtail, что он “подцепил” нужный нам `log`-файл.



2. Мониторинг

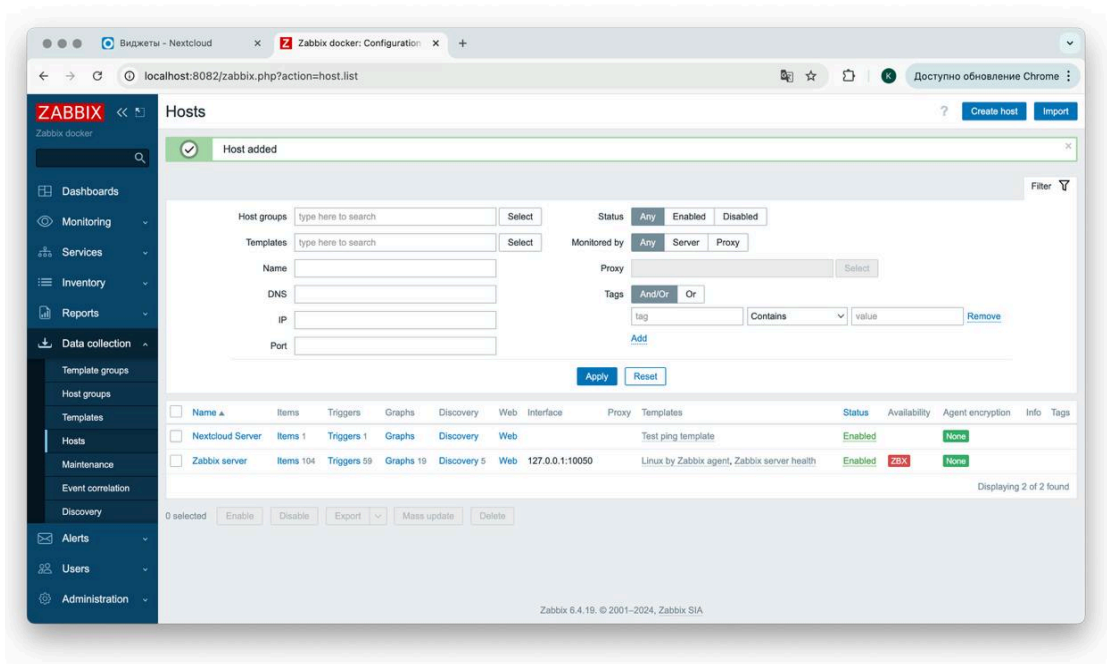
Настроили Zabbix и подключились к веб-интерфейсу по <http://localhost:8082>.

В разделе *Data collection* → *Templates* сделали *Import* кастомного шаблона (темплейта) для мониторинга nextcloud, предварительно создав yaml-файл, в котором описывается конфигурация для систем мониторинга Zabbix.

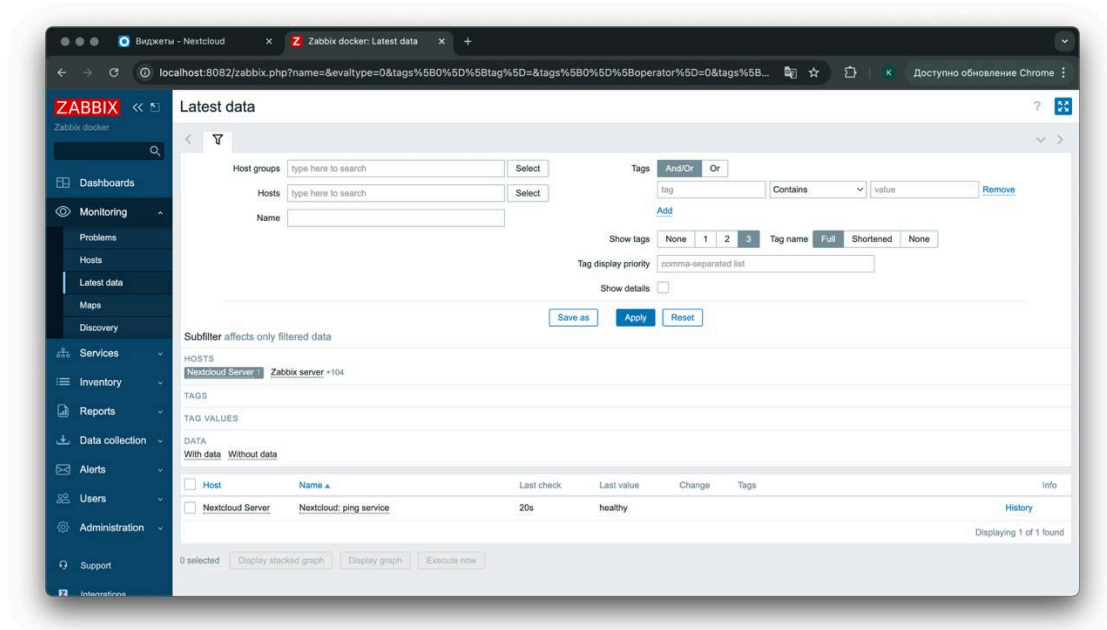


Чтобы Zabbix и Nextcloud могли общаться по своим коротким именам внутри докер-сети, в Nextcloud "разрешили" это имя.

В разделе *Data collection* → *Hosts* сделали (*Create host*). Указали адрес контейнера nextcloud, видимое имя - любое, хост группа - *Applications* (но в целом можно любую другую). Подключили к хосту нужный шаблон мониторинга, выбрав в поле *Templates* добавленный на шаге 2 *Templates/Applications* → *Test ping template*.

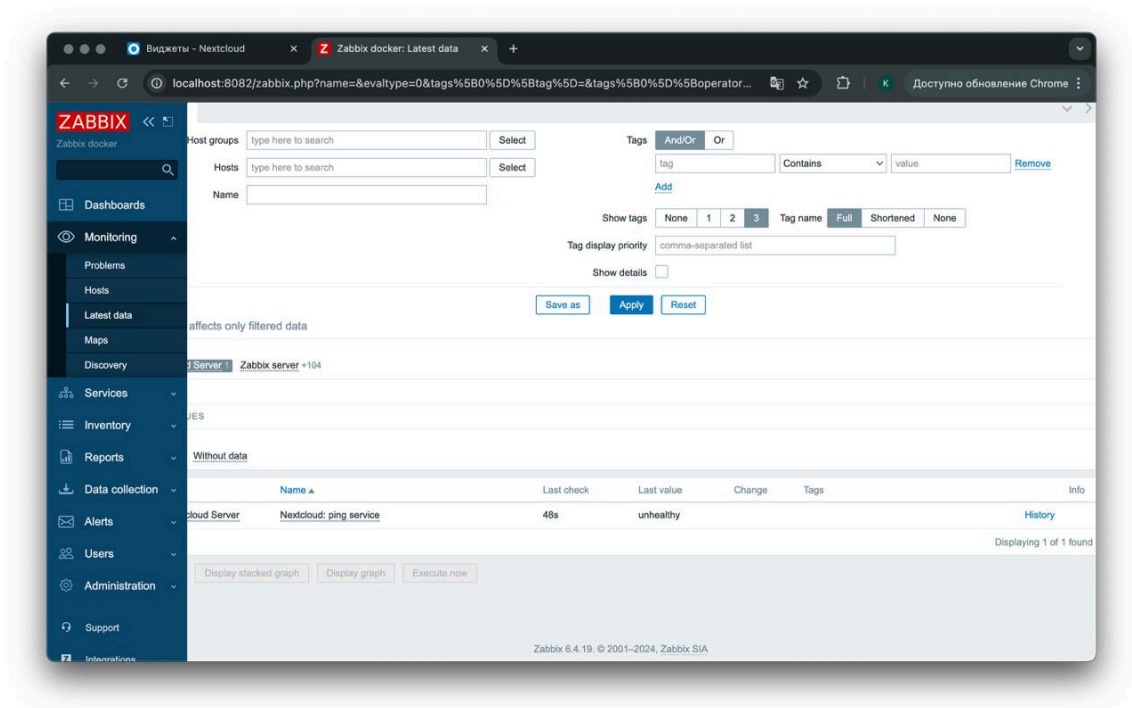


Через какое-то время в разделе Latest data появилось первое значение *healthy*.

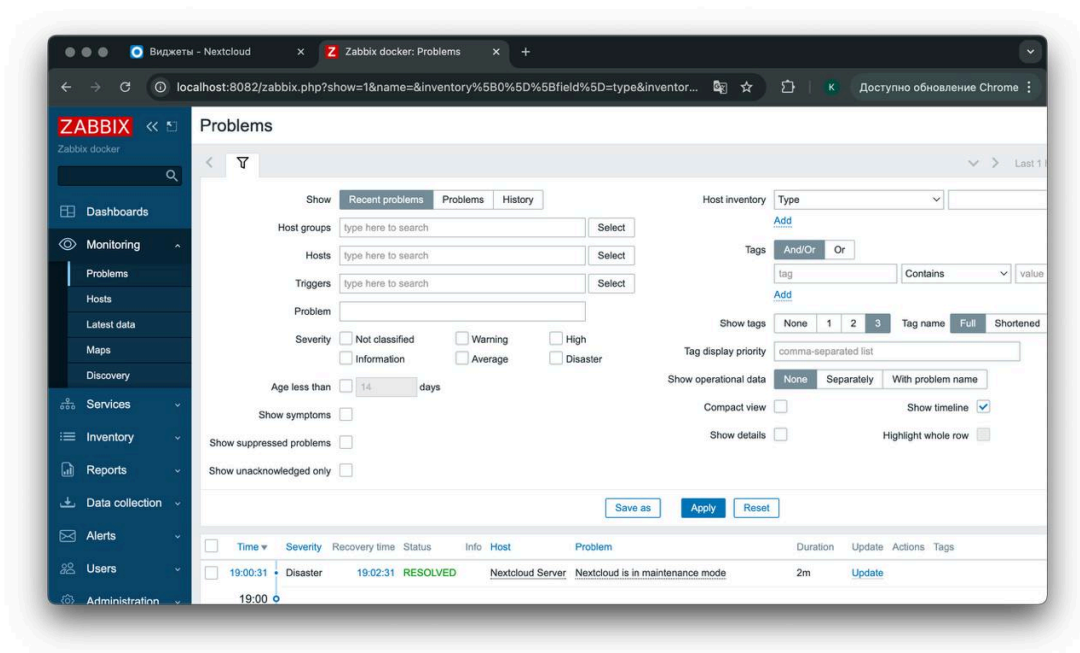


Дополнительно проверили работу триггера, включив в некстклауде *maintenance mode* (**php occ maintenance:mode --on** в контейнере). Затем включили режим обратно. Убедились, что проблема помечена как “решённая”.

```
(venv) ildarismagilov@MacBook-Air-Karina-2 computer % docker exec -u www-data -it nextcloud bash
www-data@7d7a18aeb42e:~/html$ php occ maintenance:mode --on
Maintenance mode enabled
www-data@7d7a18aeb42e:~/html$ exit
exit
(venv) ildarismagilov@MacBook-Air-Karina-2 computer % curl http://localhost:8080/status.php
{"installed":true,"maintenance":true,"needsDbUpgrade":false,"version":"29.0.6.1","versionstring":"29.0.6","edition":"","productname":"Nextcloud","extendedSupport":false}
(venv) ildarismagilov@MacBook-Air-Karina-2 computer %
```



```
(venv) ildarismagilov@MacBook-Air-Karina-2 computer % docker exec -u www-data -it nextcloud bash
www-data@7d7a18aeb42e:~/html$ php occ maintenance:mode --off
Maintenance mode disabled
www-data@7d7a18aeb42e:~/html$ exit
exit
(venv) ildarismagilov@MacBook-Air-Karina-2 computer % curl http://localhost:8080/status.php
{"installed":true,"maintenance":false,"needsDbUpgrade":false,"version":"29.0.6.1","versionstring":"29.0.6","edition":"","productname":"Nextcloud","extendedSupport":false}
(venv) ildarismagilov@MacBook-Air-Karina-2 computer %
```



3. Визуализация

В терминале выполнили команду `docker exec -it grafana bash -c "grafana cli plugins install alexanderzobnin-zabbix-app"`, затем `docker restart grafana`.

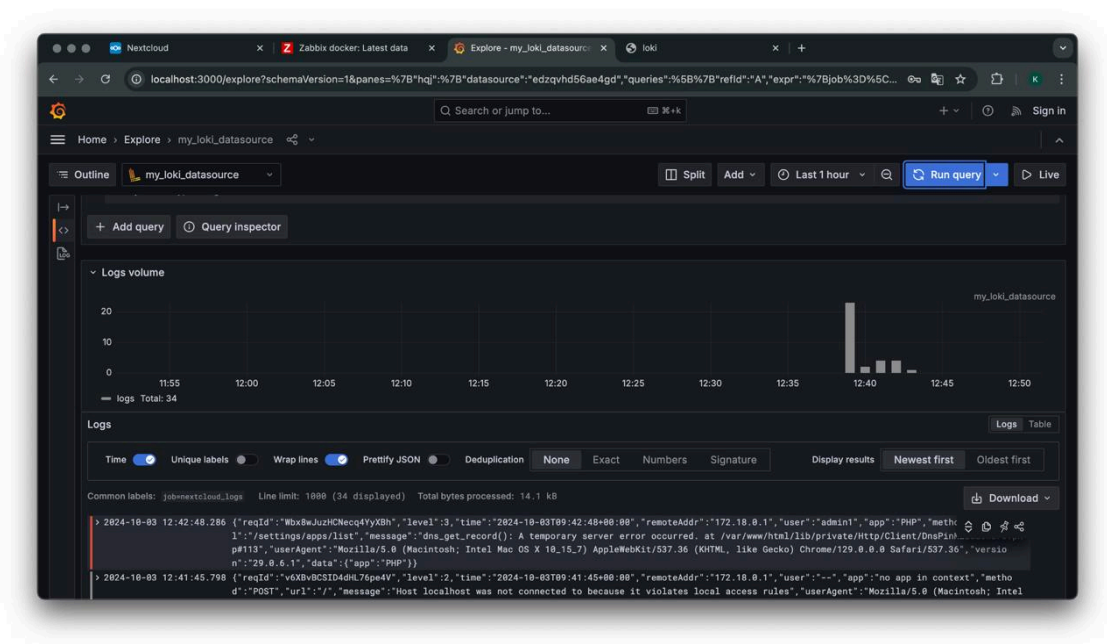
Зашли в графану (по умолчанию <http://localhost:3000/>), раздел Administration → Plugins. Нашли там Zabbix, активировать (*Enable*)

Подключили Loki к Grafana, раздел *Connections* → *Data sources* → *Loki*. В настройках подключения указать любое имя и адрес <http://loki:3100>, все остальное оставили по дефолту.

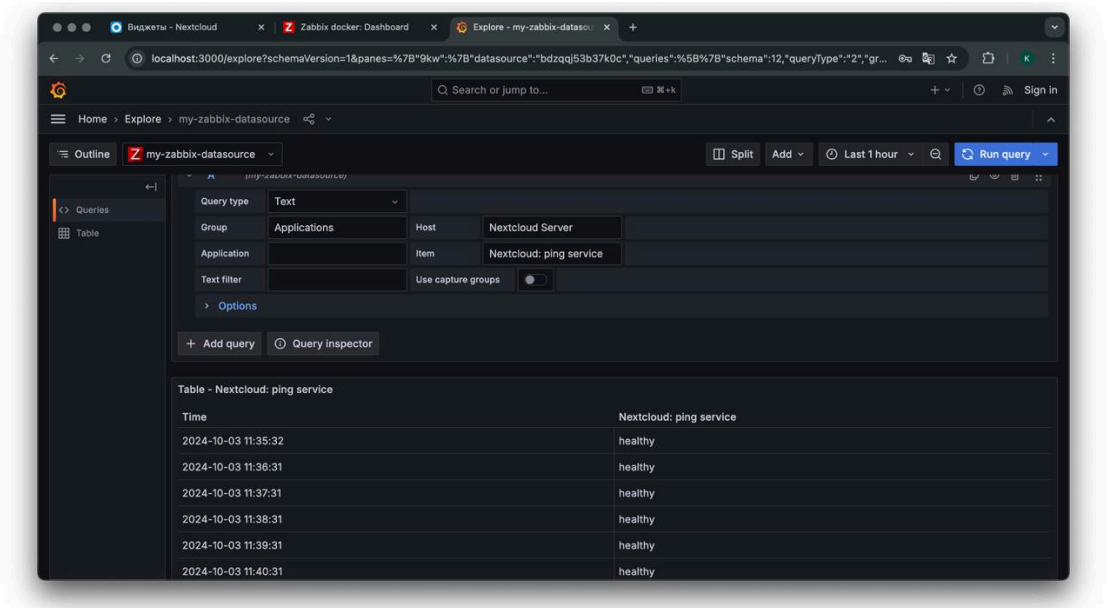
Сохранить подключение, нажав *Save & Test*. Если нет ошибок и сервис предлагает перейти к визуализации и/или просмотру данных, значит в Части 1 все настроено правильно.

Точно так же с Zabbix: снова подключили новый датасурс, в этот раз Zabbix. В качестве *URL* указали http://zabbix-front:8080/api_jsonrpc.php, заполнили *Username* и *Password*, через *Save & test* проверили, что подключение успешно.

Перешли в *Explore*, выбрали в качестве селектора (индекса) **job** либо **filename** — эти значения были в выпадающем списке. Затем нажали *Run query* и увидели свои логи.

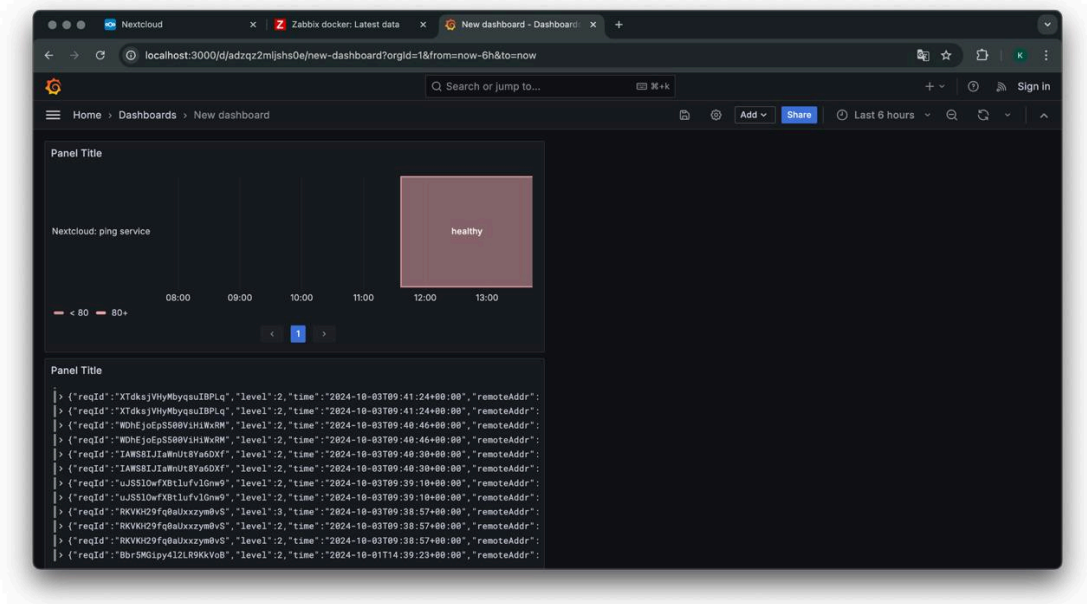


То же самое с Zabbix, при выставлении всех соответствующих фильтров.



Задание

Создали два дашборда в Grafana с использованием датасурсов Zabbix (цветная плашка) и Loki (таблица с логами).



Ответы на вопросы

1. Чем SLO (Service Level Objective) отличается от SLA (Service Level Agreement)?

Цель: SLO — это внутренние цели, SLA — это обязательства перед клиентами.

Формальность: SLO не является юридически обязывающим, SLA — юридически обязывающий документ.

Содержание: SLO описывает ожидаемый уровень сервиса, SLA включает SLO и последствия за их невыполнение.

2. Чем отличается инкрементальный бэкап от дифференциального?

Объем данных: Инкрементальный бэкап сохраняет изменения с последнего бэкапа (полного или инкрементального), дифференциальный — с последнего полного бэкапа.

Скорость создания: Инкрементальный бэкап быстрее, дифференциальный — медленнее, так как он сохраняет больше данных.

Восстановление: Восстановление из инкрементального бэкапа требует последнего полного и всех инкрементальных, дифференциальный — только последнего полного и последнего дифференциального.

3. В чем разница между мониторингом и observability?

Фокус: Мониторинг сосредоточен на сборе метрик и логов, observability — на понимании состояния системы.

Вопросы: Мониторинг отвечает на "Что произошло?", observability — на "Почему это произошло?" и "Как это исправить?".

Глубина анализа: Мониторинг предоставляет поверхностные данные, observability позволяет глубже анализировать и выявлять причины проблем.