

Теория.

Зачем нужен мониторинг. Yandex Monitoring

Мониторинг — это контроль за характеристиками системы, меняющимися со временем. Это могут быть трафик сайта, утилизация памяти или ресурсов процессора и т.д. Такие характеристики называются **метриками**. Их изменение отслеживают для того, чтобы они не выходили за границы приемлемых значений. Например, если загрузка процессора близка к 100%, то стоит нарастить вычислительные ресурсы.

Задачи мониторинга — хранить и визуализировать метрики в реальном времени, а также оповещать пользователя, если что-то пошло не так.

Нужен ли вам мониторинг

Единственно верный ответ — да, мониторинг нужен всегда. Без мониторинга невозможно построить надёжное приложение или сервис. Даже если ваше приложение небольшое и не критически важное, мониторинг будет полезен. Он поможет вовремя обнаружить, что на сервере заканчивается оперативная память или свободное место на диске, позволит отследить аномально быстрый рост числа аккаунтов и т.д.

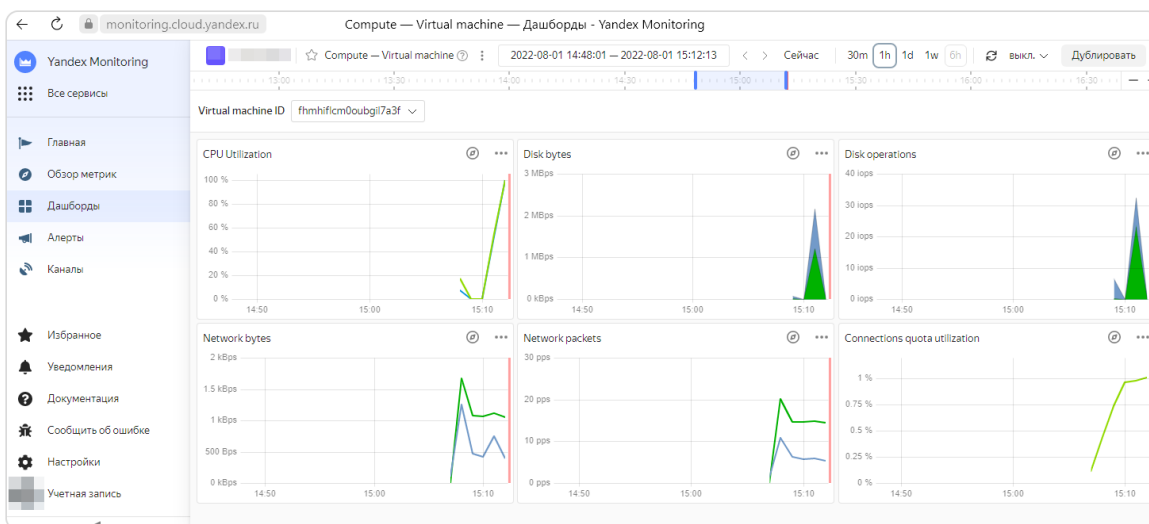
В Yandex Cloud есть собственный инструмент для мониторинга облачной инфраструктуры и размещённых в ней сервисов — [Yandex Monitoring](#).

Обзор Yandex Monitoring

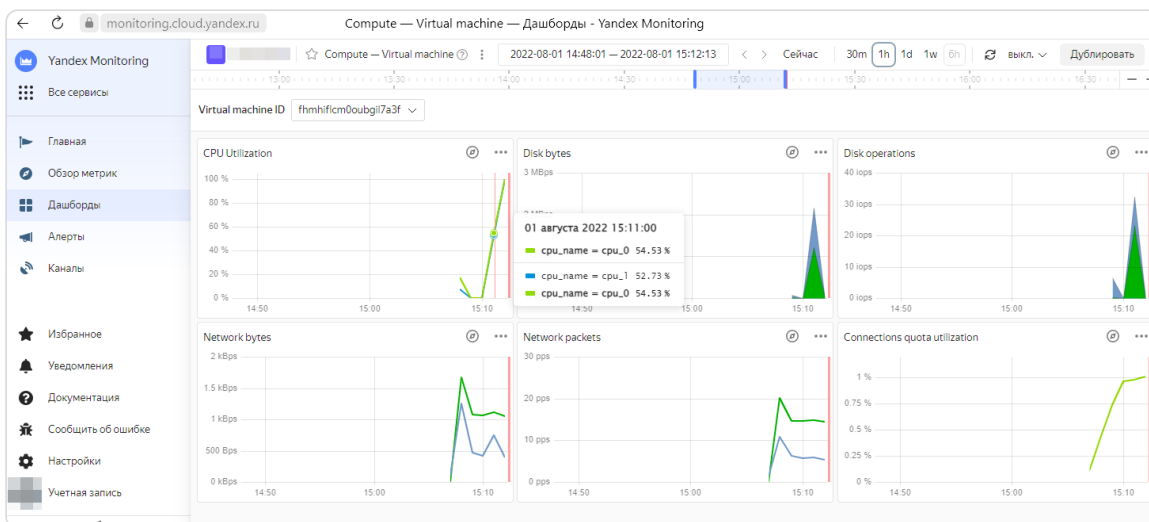
У каждого ресурса в Yandex Cloud есть набор системных метрик. Такие метрики начинают регистрироваться сразу же в момент создания ресурса. Например, для виртуальной машины можно следить за загрузкой процессора и утилизацией оперативной памяти.

Метрики обычно визуализируются с помощью **графиков** (значения метрик можно выводить и в текстовом виде, но это менее наглядно).

Блоки с графиками или текстом называются **виджетами**. Несколько виджетов можно расположить рядом, тогда получается **дашборд**. В Yandex Monitoring есть готовые дашборды для облачных сервисов — виртуальных машин, балансировщиков, кластеров Managed Kubernetes и других. **Сервисные дашборды** не надо настраивать, они работают из коробки.



Когда вы наводите указатель мыши на какой-то график на дашборде, всплывает окно со значением метки.



Если на сервисных дашбордах информации недостаточно, можно настроить собственные, **пользовательские дашборды** и вывести на

них практически любые системные или пользовательские метрики. Вы узнаете, как это сделать, в практической работе.

Пользовательские метрики можно создавать не только для сервисов Yandex Cloud, но и для своих приложений. Так вы сможете собрать все критические показатели вместе и наблюдать за ними, не переключаясь между различными системами мониторинга.

Выгрузка и загрузка метрик

Вы можете использовать метрики Yandex Cloud в другом сервисе или приложении. Для этого их выгружают в файл. И наоборот, в Yandex Cloud Monitoring можно загружать метрики из других приложений и сервисов, чтобы было удобнее контролировать их работу без переключения на разные дашборды.

За загрузку метрик отвечает агент для поставки метрик [Yandex Unified Agent](#).

Хранение метрик и удаление устаревших метрик

Как мы уже говорили, системные метрики автоматически собираются со всех активных ресурсов Yandex Cloud. После удаления ресурса сбор метрик с него прекращается, но уже собранные метрики не удаляются сразу, а хранятся в течение 30 дней.

Например, если вы удалили виртуальную машину 1 августа, то её метрики будут доступны в Yandex Monitoring до 31 августа.

Любые системные метрики, для которых новые значения не поступали в течение 30 дней, считаются устаревшими. Такие метрики автоматически удаляются из Yandex Monitoring раз в сутки.

Например, даже если вы не удаляли виртуальную машину, а просто остановили её и не запускали в течение месяца, то собранные для неё метрики будут удалены.

Автоматическое удаление затрагивает только системные метрики и не распространяется на пользовательские.

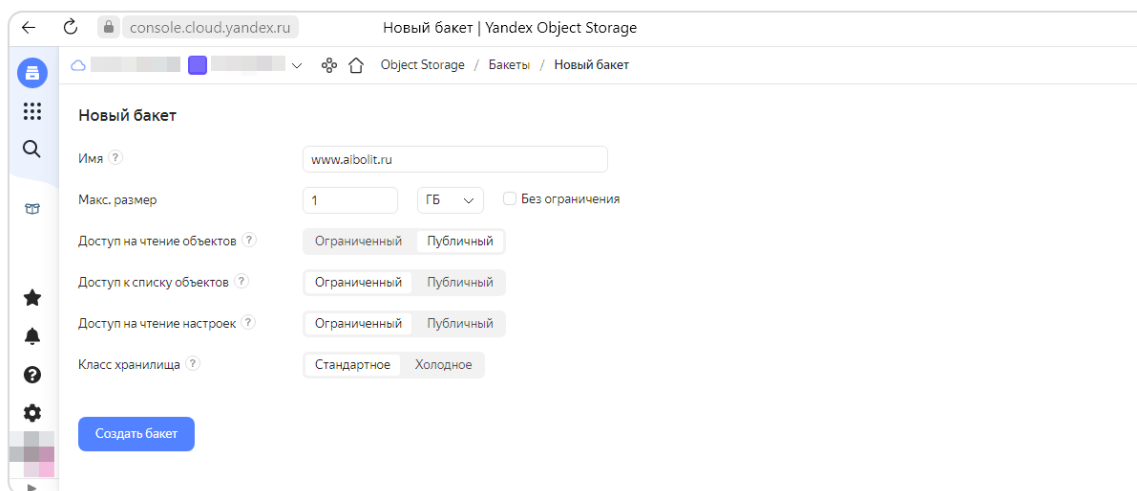
Практическое занятие. Начало работы с Yandex Monitoring

В этой практической работе вы создадите свой дашборд в сервисе Yandex Monitoring.

Подготовка. Что мониторить

Для мониторинга используйте сайт клиники «Доктор Айболит».

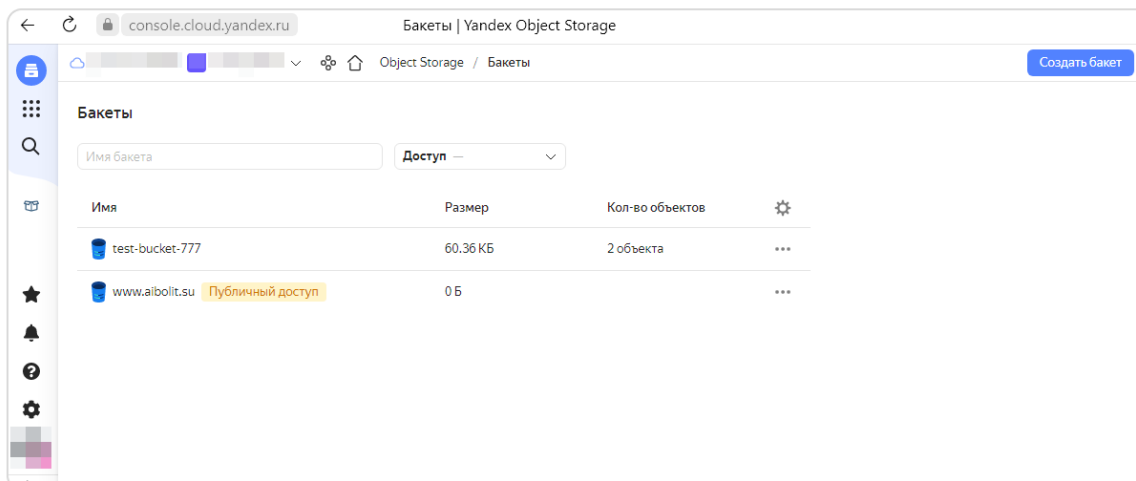
Прежде всего создадим бакет:



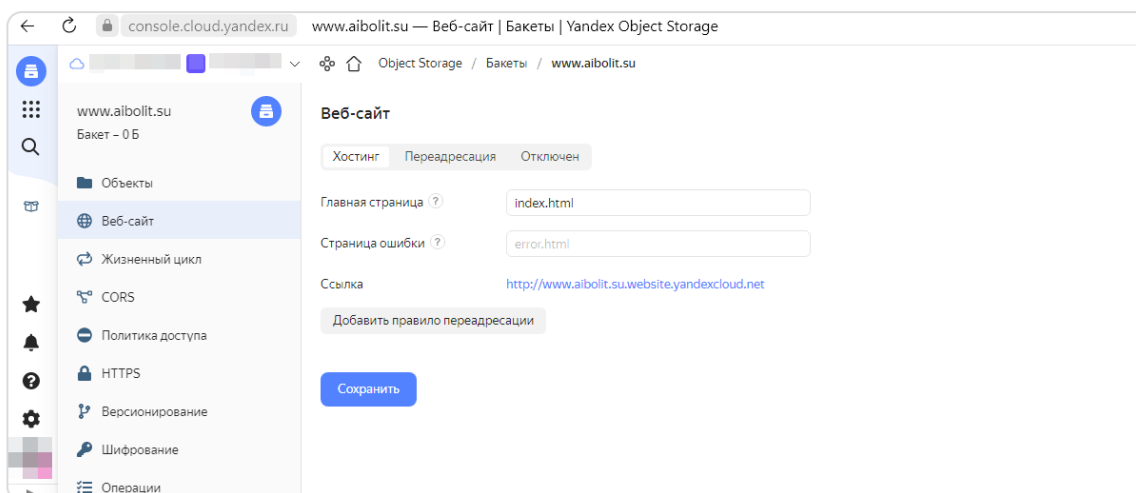
Обратите внимание на несколько особенностей:

- Если вы планируете использовать собственный домен (например `www.example.com`), то присвойте бакету **точно такое же** имя.
 - Откройте публичный доступ на **чтение объектов**. Это позволит пользователям интернета скачивать объекты из бакета и просматривать сайт в браузере.

Если публичный доступ открыт, то рядом с именем бакета в списке появится соответствующая надпись:



Теперь загрузите в бакет файлы сайта (например, [этот](#) и [этот](#)) любым удобным способом. Чтобы настроить хостинг, войдите в бакет в консоли управления. На левой панели выберите вкладку **Веб-сайт** и на открывшейся странице включите **Хостинг**.



Укажите файл с главной страницей сайта (как правило, это index.html), а поле со страницей ошибки можно не заполнять.

Сохраните настройки, и сайт станет доступен по адресам:

- [http\(s\)://<имя_бакета>.website.yandexcloud.net](http(s)://<имя_бакета>.website.yandexcloud.net)
- [http\(s\)://website.yandexcloud.net/<имя_бакета>](http(s)://website.yandexcloud.net/<имя_бакета>)

По умолчанию сайт будет доступен только по протоколу HTTP. Для поддержки HTTPS нужно загрузить в объектное хранилище собственный сертификат безопасности.

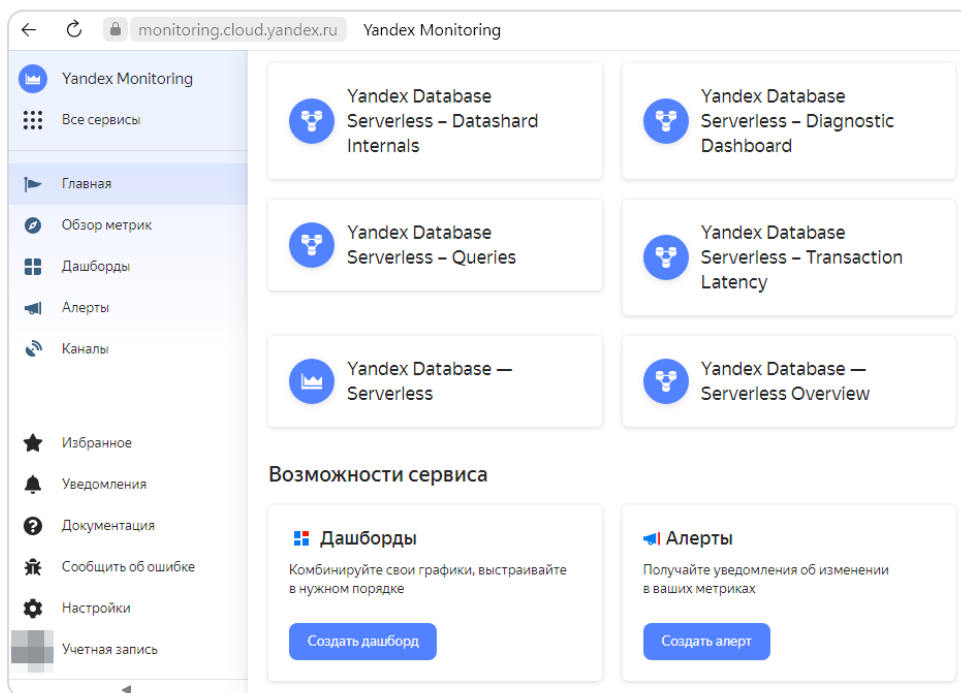
Будем отслеживать объём трафика, поступающий на сайт. Чтобы имитировать нагрузку, мы воспользуемся утилитой `wget`, которую уже использовали в практической работе с кластером Managed Kubernetes. Если у вас не установлена эта утилита, её можно скачать [здесь](#). Адрес же своего сайта можно посмотреть в Object Storage.

Скопировать код

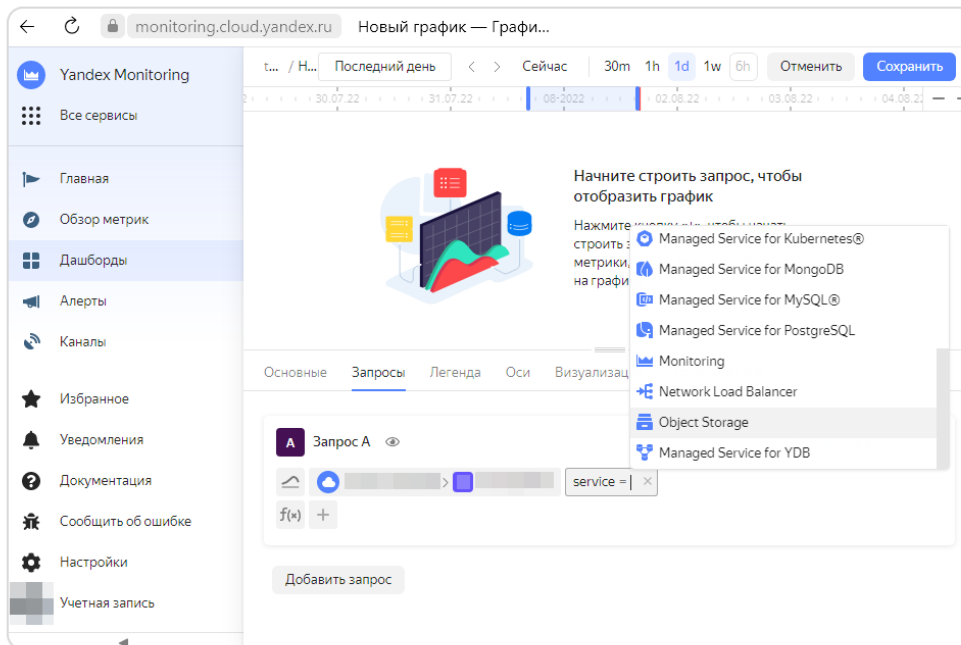
```
while true; do wget -q -O- <адрес_сайта>; done
```

Создание дашборда и виджета

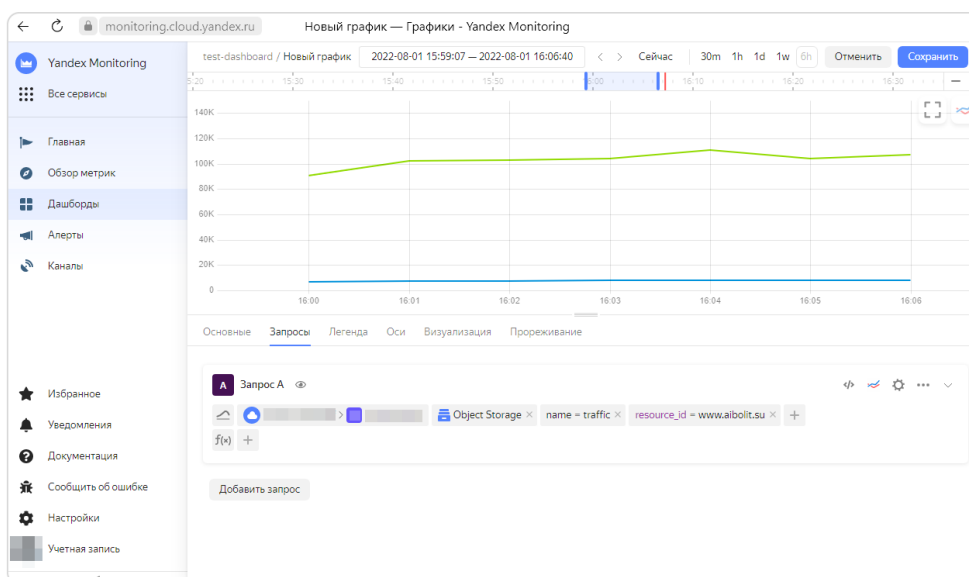
1. Откройте начальную страницу [Yandex Monitoring](https://monitoring.cloud.yandex.ru). На вкладке **Главная** нажмите кнопку **Создать дашборд**.



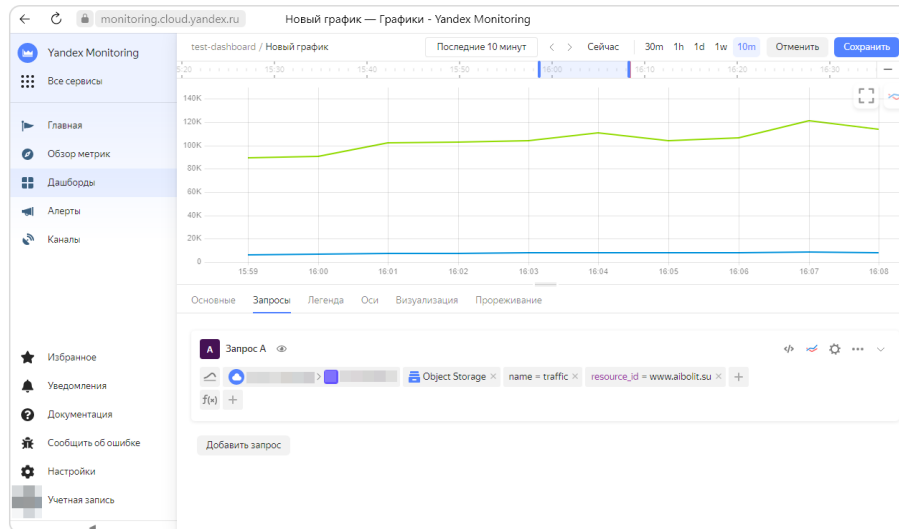
2. Открылось окно создания дашборда. В правом верхнем углу нажмите кнопку **Сохранить**, введите название нового дашборда и сохраните его.
3. Новый дашборд пустой, он не содержит виджетов. Давайте добавим виджет с графиком. Нажмите кнопку **Редактировать** и в блоке **Добавить виджет** выберите **График**.
4. Метрики добавляются в виджет с помощью запросов. Но это совсем не сложно — запросы создаются с помощью удобного конструктора, размещённого в нижней части страницы.
Конструктор сам предлагает, какие параметры нужно заполнить, и показывает доступные значения.



5. Выберите `service = Object Storage`, затем `name = traffic` и `resource_id = <имя_бакета_с_сайтом>`. Больше ничего выбирать не нужно. Этот запрос означает, что вы хотите мониторить входящий и исходящий трафик своего сайта. Сразу после сохранения запроса вверху отобразятся графики, которые соответствуют указанным параметрам.



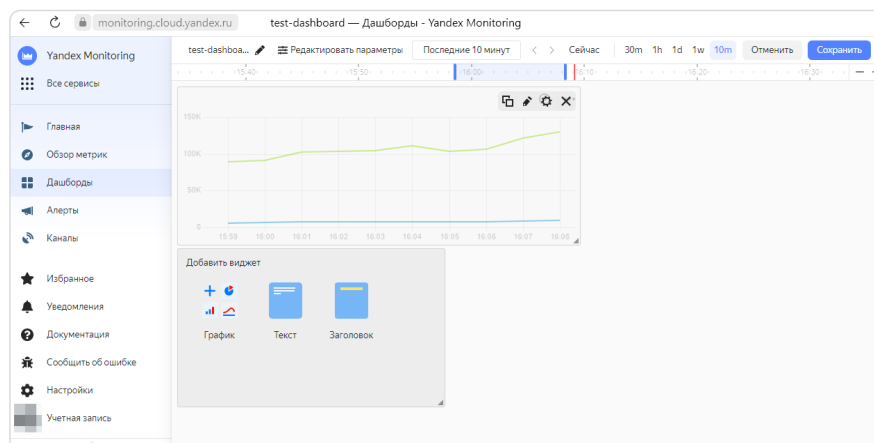
6. Вы можете настроить период времени, который хотите видеть на графике. В нашем случае будет более наглядно, если вы укажете небольшой период, например, 10 или 15 минут.



7. Нажмите кнопку **Сохранить**, чтобы созданный виджет появился на вашем новом дашборде.

Вы можете растянуть или сжать виджет до нужного размера, потянув мышью за правый нижний угол с треугольником, и перетащить виджет на нужное место на дашборде.

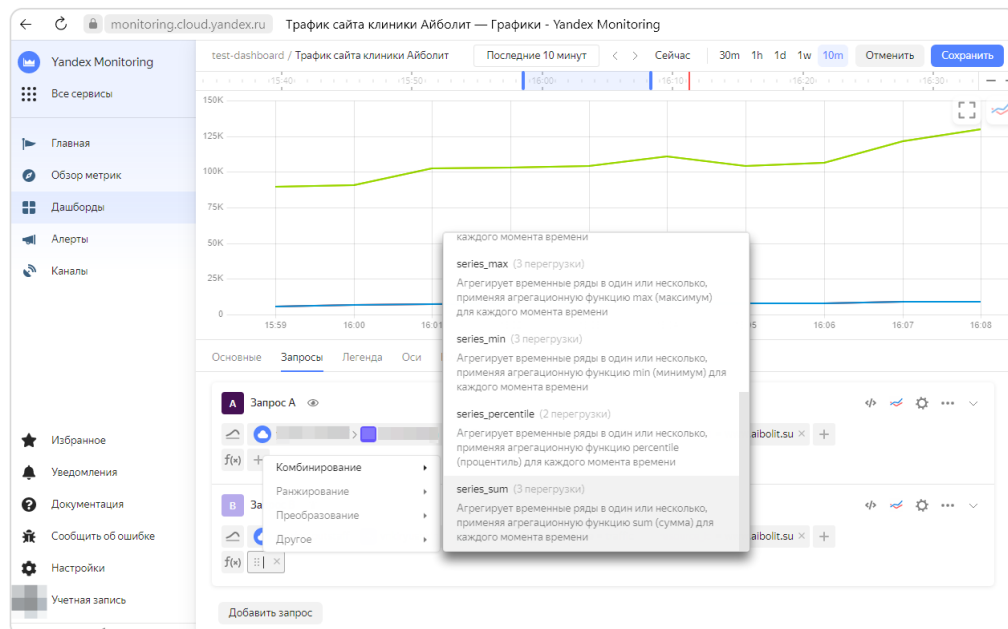
В правом верхнем углу виджета отображаются значки для редактирования («карандаш») и настройки («шестеренка»). Вы можете в любой момент отредактировать запрос, по которому строится график, а в настройках можете изменить название виджета или включить отображение легенды. Также вы можете удалить виджет, если он больше не нужен.



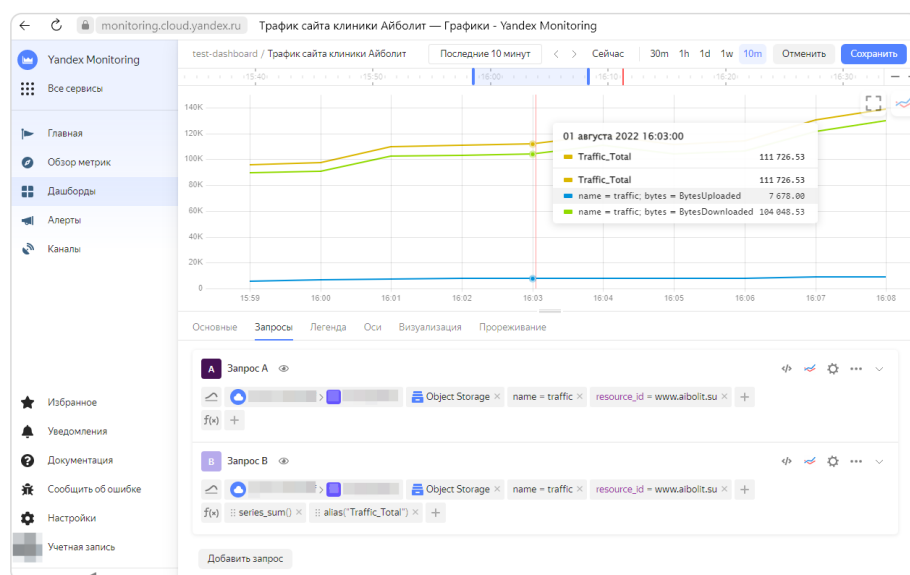
8. Задайте название для вашего виджета, например: «Трафик сайта клиники Айболит».
9. Нажмите кнопку **Сохранить**, чтобы сохранить новый виджет на дашборде.

Использование функций в виджетах

10. На виджете вы видите два отдельных графика для входящего и исходящего трафика. Допустим, вы хотите также посмотреть и суммарный трафик. Вернёмся в созданный ранее виджет и добавим к нему еще один график. Нажмите на виджете значок редактирования («карандаш»).
11. Внизу нажмите кнопку **Добавить запрос**. Повторите все настройки предыдущего запроса. Затем в строке ниже добавьте функцию — нажмите значок «+», выберите раздел **Комбинирование** и функцию `series_sum()`. В скобках в данном случае ничего указывать не нужно.

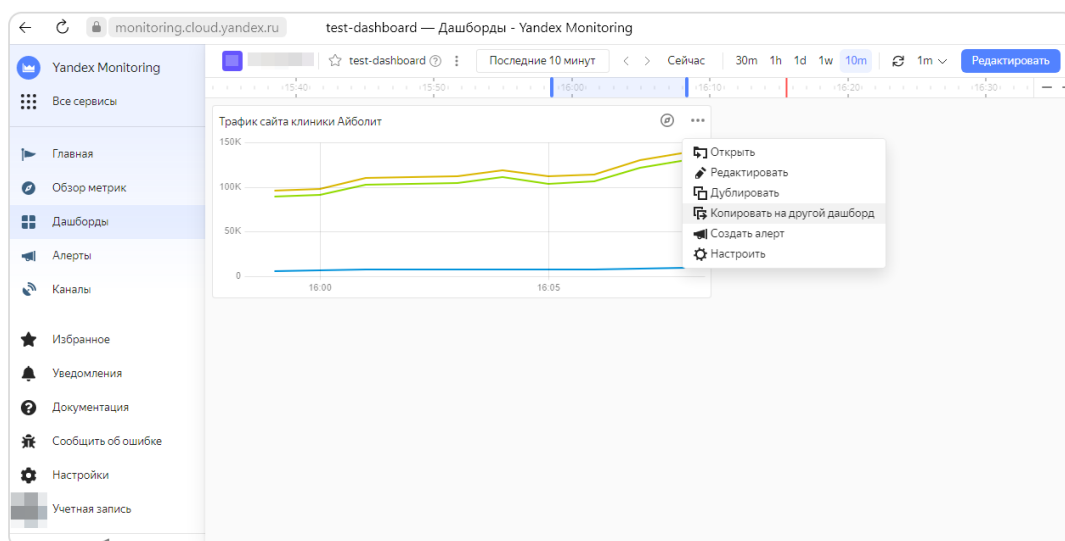


12. На графике появилась ещё одна линия, которая показывает значения суммарного трафика. Если вы наведёте на неё указатель мыши, то во всплывающем окне увидите, что у этой метрики нет имени. Чтобы добавить имя, в той же строке, где вы описывали функцию, снова нажмите на значок «+», выберите раздел **Другое** и функцию **alias**. В скобках введите имя метрики, например, `Traffic_Total`. Теперь при наведении указателя мыши на график будет отображаться это имя.



13. Нажмите кнопку **Сохранить**. На дашборде вы увидите обновленный виджет с тремя графиками. Снова нажмите кнопку **Сохранить**, чтобы сохранить изменения на дашборде.

14. Виджеты можно копировать с одного дашборда на другой с помощью меню:



Практическая работа. Выгрузка метрик в формате Prometheus

Как мы уже говорили раньше, метрики можно выгружать из Yandex Cloud Monitoring в сторонние приложения и сервисы. Пожалуй, чаще всего их выгружают для сервера [Prometheus](#).

На сегодняшний день Prometheus — один из самых популярных инструментов для мониторинга приложений и сервисов. В основе его лежит специализированная СУБД для анализа временных рядов, которая обеспечивает высокое быстродействие. В отличие от большинства систем мониторинга, Prometheus не ждёт, пока сторонние приложения передадут ему свои метрики, а сам опрашивает подключенные к нему приложения и собирает нужные данные.

Prometheus и Yandex Cloud Monitoring решают схожие задачи — хранят значения разных метрик. Prometheus фактически является стандартом для обмена метриками. Поэтому даже используя сервисы Yandex Cloud, IT-администраторы часто хотят отслеживать их работу с помощью Prometheus. Чтобы не лишать специалистов привычных инструментов, Yandex Cloud Monitoring поддерживает выгрузку данных в формате Prometheus. Для этого используется метод [prometheusMetrics](#).

Для визуализации данных, собираемых Prometheus, можно использовать сервис [Grafana](#) (в нем можно зарегистрироваться бесплатно на тестовый период). Вы можете установить Grafana на свой компьютер, а можете работать в облачной версии.

Посмотрим, как происходит выгрузка метрик в Prometheus и работа с ними в Grafana. Вы снова будете мониторить сайт клиники «Доктор Айболит».

Подготовка

1. [Создайте API-ключ](#) через консоль управления Yandex Cloud или CLI.

Если вы создаете ключ в консоли управления, то перейдите в каталог, из которого будете выгружать метрики (например `default`). Затем перейдите на вкладку **Сервисные аккаунты** и выберите существующий аккаунт. Нажмите кнопку **Создать новый ключ** и выберите **Создать API-ключ**. В описании ключа можно указать, например, «для доступа к Prometheus». Сохраните секретную часть ключа в отдельный файл, например, `prometheus-key.txt`.

2. Назначьте сервисному аккаунту роль `monitoring.viewer` на выбранный каталог.
3. Создайте [файл спецификации](#) `prometheus.yml` (см. пример ниже, замените в нем значение параметра `folderId` на идентификатор каталога, а значение для `bearer_token` — на ключ доступа из файла `prometheus-key.txt`):

Скопировать код

```
global:

  scrape_interval:     15s # Set the scrape interval to every 15
seconds. Default is every 1 minute.

  evaluation_interval: 15s # Evaluate rules every 15 seconds. The
default is every 1 minute.

  # scrape_timeout is set to the global default (10s).


rule_files:

scrape_configs:

  - job_name: 'prometheus'

    static_configs:

      - targets: ['localhost:9090']


  - job_name: 'yc-monitoring-export'
```

```
metrics_path: '/monitoring/v2/prometheusMetrics'

params:

  folderId:

    - '<идентификатор_каталога>'

  service:

    - 'storage'

bearer_token: '<секретная_часть_API-ключа>'

static_configs:

- targets: ['monitoring.api.cloud.yandex.net']

  labels:

    folderId: '<идентификатор_каталога>'
    service: 'storage'
```

Запуск сервера Prometheus

4. Если вы уже работаете с Prometheus, пропустите все шаги по установке — просто добавьте секцию `scrape_configs` из примера выше в спецификацию вашего сервера Prometheus и перезапустите сервер, а затем переходите к настройке Grafana.
5. Для запуска сервера Prometheus используйте официальный Docker-образ [prom/prometheus](https://prometheus.io/docs/operating/docker/).
Сначала загрузите образ. Для этого запустите Docker Desktop (в терминале выполните команду):

Скопировать код

```
docker pull prom/prometheus
```

6. Чтобы на сервере сразу был ваш файл спецификации, создайте свой образ на основе `prom/prometheus`. Подготовьте Dockerfile с двумя командами:

Скопировать код

```
FROM prom/prometheus
```

```
ADD prometheus.yml /etc/prometheus/
```

7. Сохраните этот файл в тот же каталог, где находится `prometheus.yml`. Назовите его именем по умолчанию: `Dockerfile`.
8. В терминале перейдите в каталог с `Dockerfile`. Создайте образ с вашей конфигурацией (используйте ваш идентификатор в Yandex Container Registry):

Скопировать код

```
docker build . -t  
cr.yandex/<идентификатор_реестра>/my-prometheus:latest -f Dockerfile
```

9. Аутентифицируйтесь в Yandex Container Registry с помощью Docker Credential helper (чтобы Docker мог от вашего имени отправить образ в ваш реестр):

Скопировать код

```
yc container registry configure-docker
```

10. Теперь отправьте образ в ваше хранилище в облаке:

Скопировать код

```
docker push cr.yandex/<идентификатор_реестра>/my-prometheus:latest
```

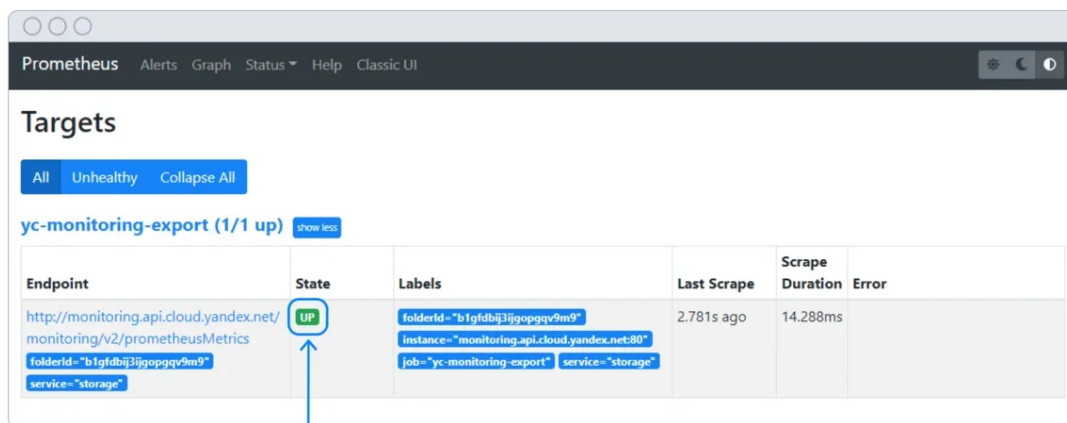
11. Создайте виртуальную машину с помощью Container Optimized Image, вы уже делали это раньше в практической работе (в разделе **Выбор образа загрузочного диска** переключитесь на

вкладку **Container Solution** и нажмите **Настроить**. Выберите из реестра созданный вами образ, остальные настройки оставьте по умолчанию и нажмите **Применить**).

При создании виртуальной машины используйте ваш сервисный аккаунт. Задайте логин (например `prom`) и `ssh`-ключ.

Назовите машину, например, `for-prometheus`.

12. Проверьте статус сервера по адресу `http://<публичный IP-адрес VM с Prometheus>:9090/targets`. Через несколько минут после запуска статус процессов `prometheus` и `yc-monitoring-export` должен стать **UP**.



13. Подайте нагрузку на ваш сайт:

Скопировать код

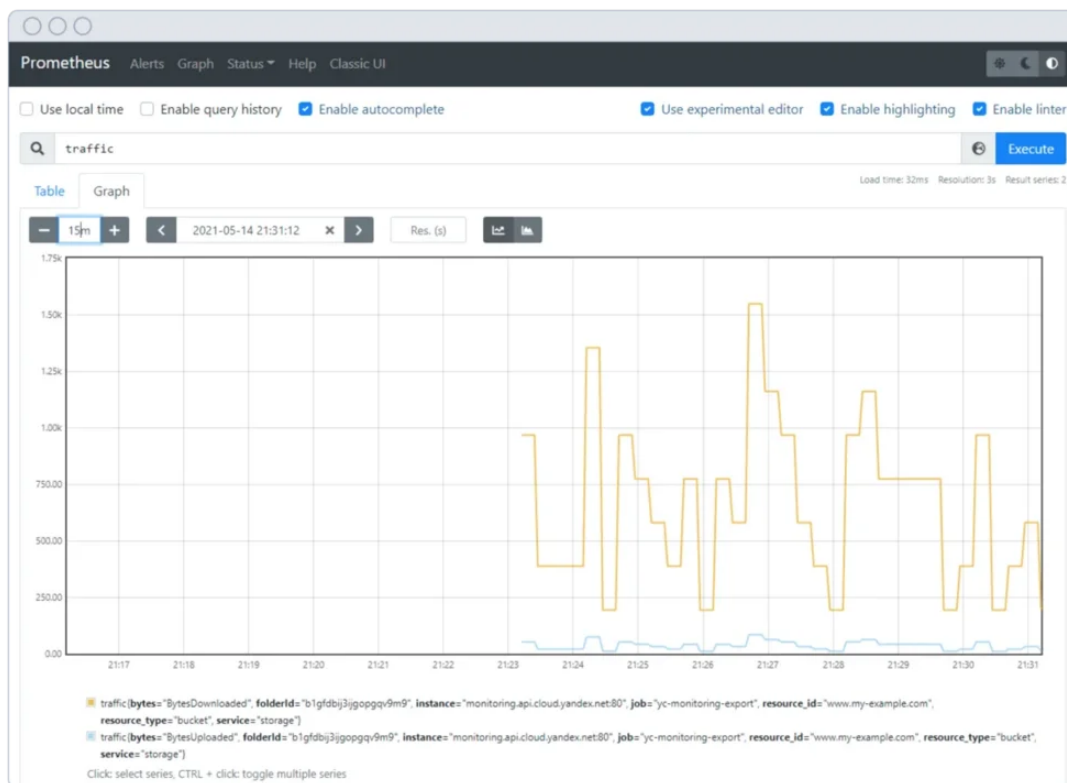
```
while true; do wget -q -O- <адрес_сайта>; done
```

14. Подождите несколько минут и проверьте, как поставляются метрики в Prometheus.

В верхнем меню выберите пункт **Graph**. Нажмите на значок «Земли». Откроется меню с доступными метриками. Выберите метрику, которую вы хотите проверить, например, `traffic` и нажмите кнопку **Execute**.

Переключитесь на вкладку **Graph**. Выберите текущее время, для наглядности уменьшите интервал запроса данных (например до

15 минут). Вскоре вы увидите график изменения выбранной метрики.



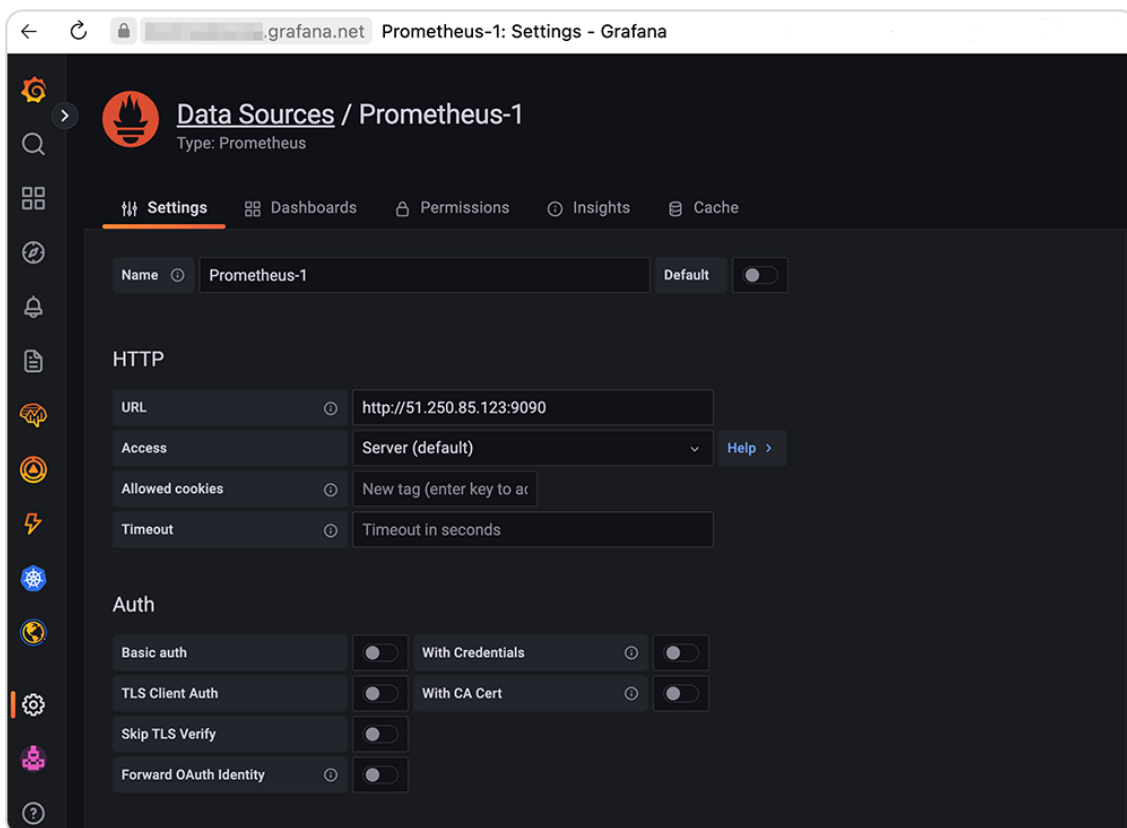
Настройка Grafana

13. Теперь посмотрим, как метрики визуализируются в системе Grafana.

Если у вас еще нет аккаунта в Grafana, [создайте его](#) с помощью нескольких простых шагов, это бесплатно. Вам откроется интерфейс по адресу https://<ваш_логин>.grafana.net/.

Добавление источника данных

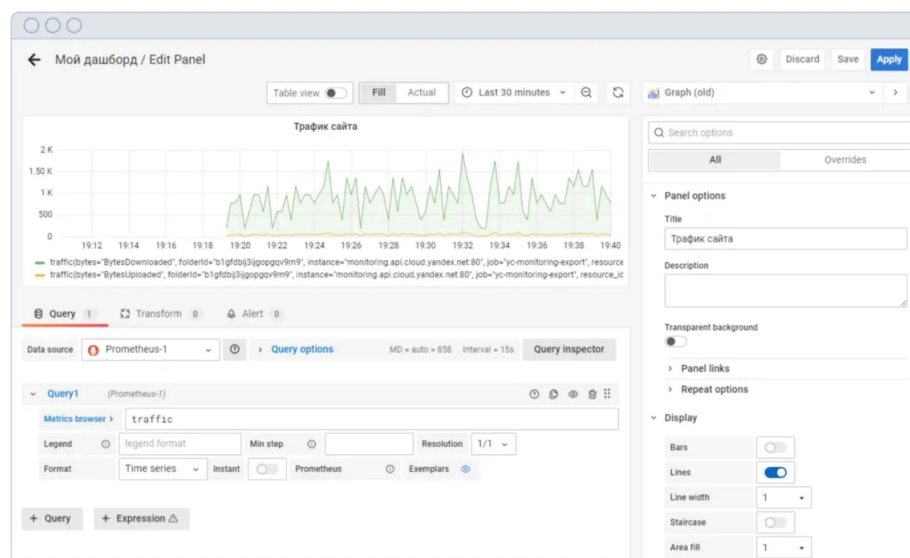
14. Настройте Prometheus в качестве источника данных. На главной странице нажмите кнопку **Connect data**. Из предложенного списка выберите источник `Prometheus data source` и нажмите кнопку **Create Prometheus data source**.



15. В следующем окне в поле URL введите endpoint сервера Prometheus `http://<публичный IP-адрес ВМ с Prometheus>:9090`. Больше никакие настройки менять не нужно. Внизу нажмите кнопку **Save & Test**. Должна отобразиться надпись **Data source is working**.

Добавление дашборда

16. Вернитесь на главную страницу (нажав на логотип в левом верхнем углу) и нажмите **Create your first dashboard**. Откроется окно настройки дашборда.
17. В нижней части экрана на вкладке **Query** выберите источник данных — **Prometheus**.
18. Выберите метрику, которую вы хотите отслеживать. Нажмите на поле **Metrics**, в открывшемся списке выберите метрику `traffic`. Сверху отобразится график выбранной метрики.



19. Вверху справа в поле **Panel Title** укажите название графика (например, «Трафик сайта»).
20. Теперь сохраните настройки — в правом верхнем углу нажмите кнопку **Save** и укажите название дашборда (например, «Мой дашборд»).

