

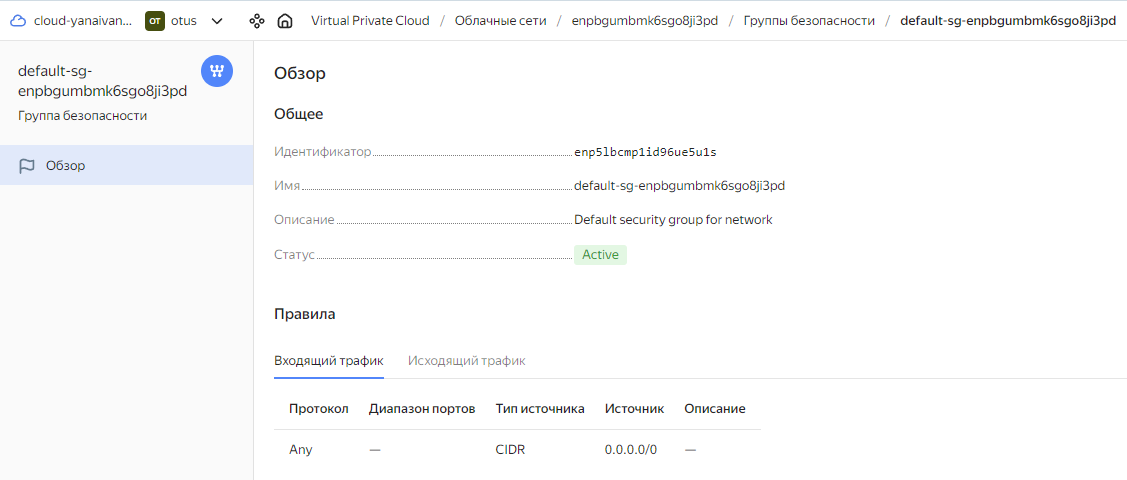
Целью работы являлось создание ETL приложения, позволяющего на постоянной основе получать данные по проходящим в мире праздникам при помощи средств развернутых в Yandex Cloud.

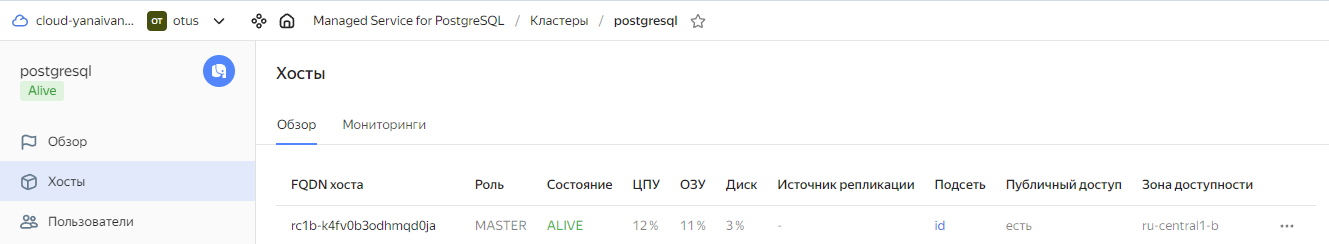
Этапы:

1. Подготовка инфраструктуры
2. Сбор и обработка данных
3. Загрузка данных в S3
4. Загрузка данных в DWH
5. Построение аналитической отчетности на основе датасета

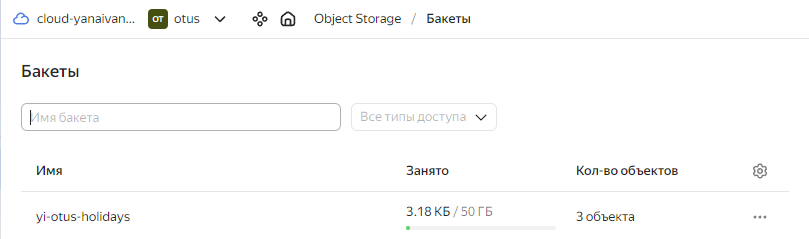
Подготовка инфраструктуры

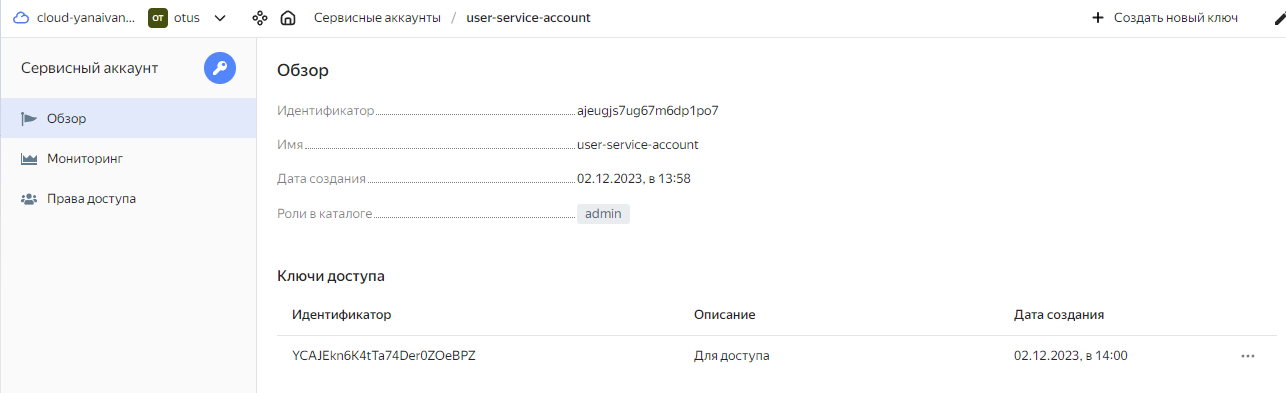
В Yandex Cloud была развернута Виртуальная машина и настроен доступ для работы с Airflow и Postgresql.





Для работы с S3 был создан сервисный аккаунт и статистические ключи для доступа к Object Storage. Ключи были спрятаны в переменные Airflow, а в DAGе прописан путь к ним.



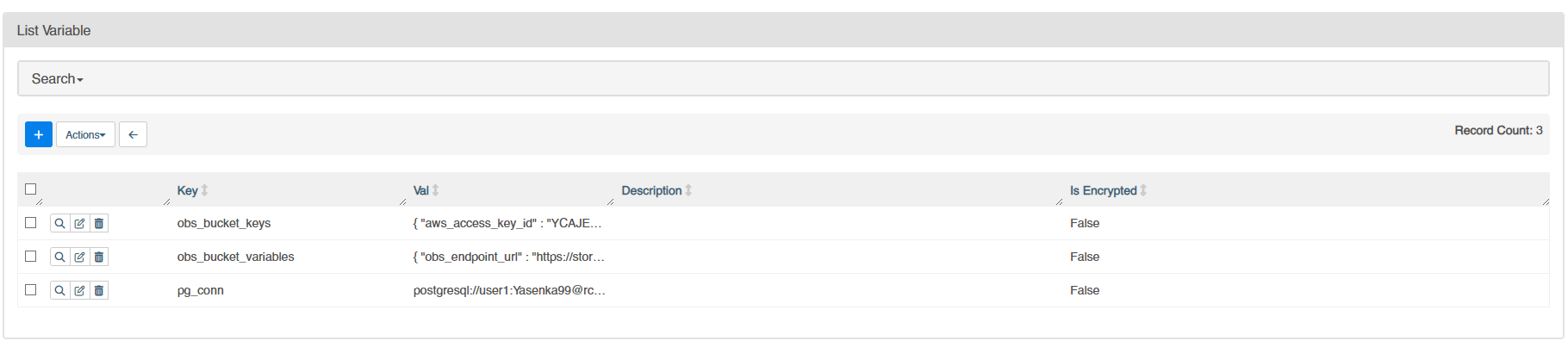


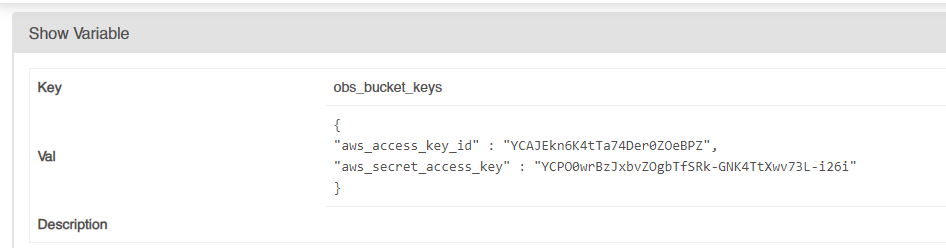
Сбор и обработка данных

В качестве источника данных был выбран веб-сайт с API, предоставляющим данные о проходящих по всему миру праздниках (https://date.nager.at).

Далее с помощью подключения к серверу по SSH в Visual Studio Code был написан пайплайн загрузки данных (DAGs) в S3 бакет и прогрузки данных в PostgreSQL с настроенным расписанием на ежедневное обновление.

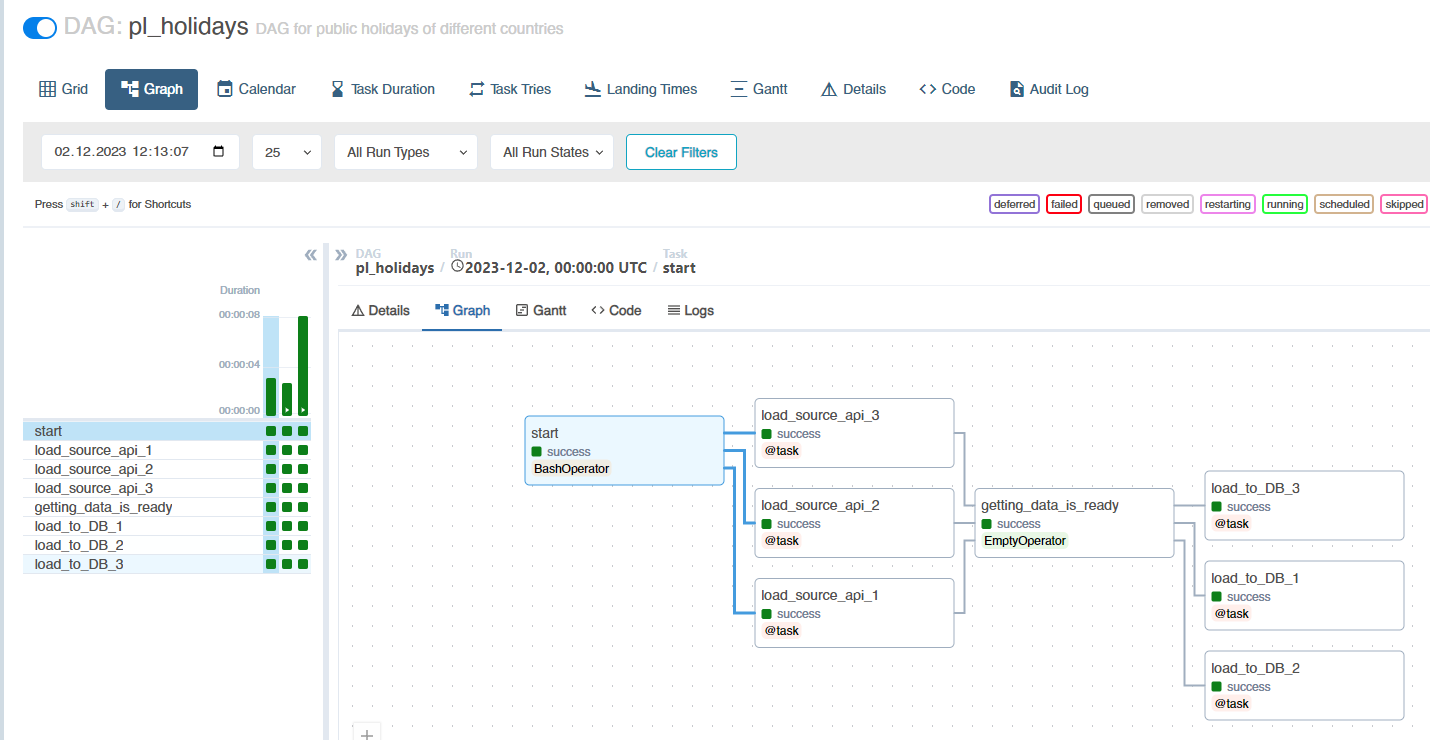
Для ускорения работы пайплайна скачивание и прогрузка данных осуществляются параллельно и независимо друг от друга.







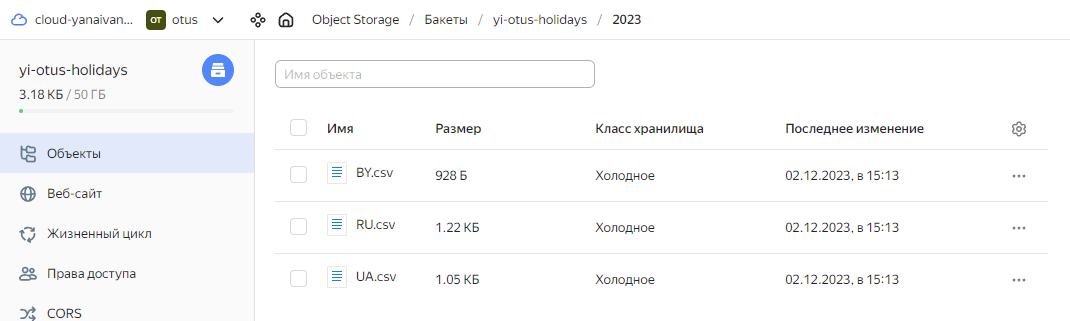




Код находится в репозитории [pl\_holidays.py](https://github.com/Yananisan/Public-Holidays/blob/main/pl_holidays.py)

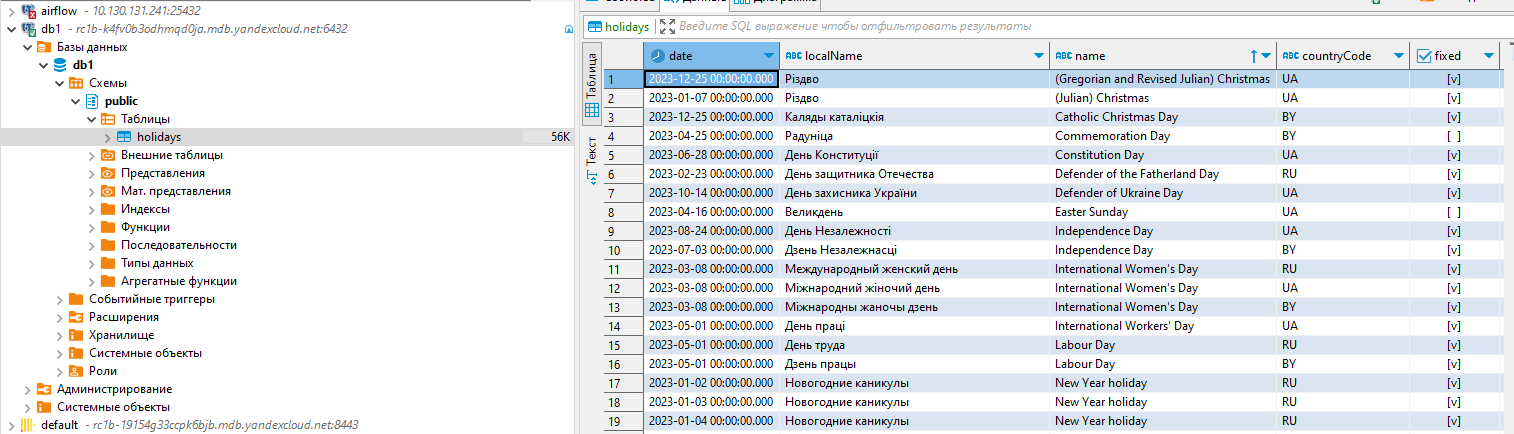
Загрузка данных в S3

Полученные от API данные в формате byte-json были преобразованы в csv и загружены в созданный бакет Yandex Object Storage для хранения сырых данных.



Загрузка данных в DWH

Функция toDB() забирает файлы из S3, подключается к PostgreSQL и загружает данные в таблицу.



Построение аналитической отчетности на основе витрины данных

В создано подключение к БД PostgreSQL, подготовлен датасет и созданы чарты с расчетами необходимых метрик.  
На основе созданных чартов собран дашборд.

