

Рекомендательная платформа MOODREC

Руководство для администраторов

Документация для администраторов

Данная документация предназначена для персонала, производящего установку и обслуживание сервиса MoodRec на собственном оборудовании. Эта документация содержит информацию по развертыванию сервиса, его конфигурации и дальнейшему обновлению.

Предварительная подготовка

- Перед началом установки настоятельно рекомендуем ознакомиться с техническими требованиями к оборудованию и персоналу. Они подробно описаны в функциональных требованиях к MoodRec.
- Все дальнейшие действия по установке требуют наличия привилегий пользователя `root`.
- Для паролей и секретных комбинаций используется набор из символов – `abcdefghijklmnopqrstuvwxyzABCDEF`. Обязательно сгенерируйте более сложный пароль, состоящий из не менее 32-х символов.

Установка

В данном руководстве показан процесс установки для Debian-подобной ОС (например, **Astra Linux** или **Ubuntu**). Для non-Debian ОС процесс выглядит похожим образом, кроме установки некоторых зависимостей.

Загрузка архива

Всем клиентам предоставляется архив с расширением `.tar.gz`, распакуйте его и переместите содержимое в каталог `/opt`:

```
{
  tar xzf <архив>.tar.gz
  mv moodrec /opt/moodrec
}
```

Проверка времени на сервере

Синхронизация времени между серверами критична: различия могут привести к нарушению работы и ошибочной статистике. Настройте [NTP-сервер](#) и проверьте синхронизацию:

```
timedatectl status
```

Установка системных лимитов

Добавьте следующие лимиты в файл `/etc/security/limits.conf`:

```
*          -          nofile          1048576
root       -          nofile          1048576
root       -          memlock         unlimited
```

Затем переподключитесь и проверьте, что лимиты применились:

```
ulimit -n
```

Установка зависимостей

Установите следующие зависимости:

```
{
  apt update
  apt install --no-install-recommends gcc g++ libc6-dev
}
```

Установка необходимых компонентов

Установка Kafka

Kafka используется как основной брокер сообщений для работы сервиса.

Установка происходит согласно инструкциям:

- [Быстрый старт](#)
- [С чего начать](#)

Установка MongoDB

TIP

MongoDB используется для хранения системных настроек и пользовательских данных. Ниже приведена инструкция по установке из архива.

Дополнительные материалы:

- [Установка MongoDB Community Edition на Ubuntu](#)
- [Описание файла конфигурации](#)
- [Конфигурация базы данных](#)

Перейдите на [страницу загрузки](#), выберите версию 8.0, вашу версию ОС и архив `.tgz`. Скопируйте ссылку и скачайте архив:

```
wget https://fastdl.mongodb.org/linux/mongodb-linux-x86_64-ubuntu2404-8.0.17.tgz
```

Распакуйте архив в `/usr/local` и создайте символическую ссылку:

```
{
  tar xzf mongodb-linux-x86_64-ubuntu2404-8.0.17.tgz -C /usr/local
  ln -s /usr/local/mongodb-linux-x86_64-ubuntu2404-8.0.17 /usr/local/mongodb
}
```

Установите зависимости:

```
{
  apt update
  apt install -y gnupg curl
}
```

Создайте пользователя, группу и рабочие каталоги:

```
{
  groupadd mongodb
  useradd -r -g mongodb -s /bin/false mongodb
  mkdir -p /var/{lib,log}/mongodb
  chown -R mongodb:mongodb /var/{lib,log}/mongodb
}
```

INFO

По умолчанию в MongoDB отключена аутентификация. Для повышения безопасности вы можете включить аутентификацию с использованием механизма SCRAM (аутентификация по паролю).

Создайте файл конфигурации `/etc/mongod.conf`:

```
# mongod.conf

# for documentation of all options, see:
# http://docs.mongodb.org/manual/reference/configuration-options/

storage:
  dbPath: /var/lib/mongodb
  wiredTiger:
    engineConfig:
      directoryForIndexes: true

systemLog:
  destination: file
  logAppend: true
  logRotate: reopen
  path: /var/log/mongodb/mongod.log

net:
  port: 27017
  bindIp: 127.0.0.1
```

INFO

Вы можете включить поддержку TLS для подключения к MongoDB. Для этого добавьте в файл `/etc/mongod.conf` следующие параметры:

```
net:
  tls:
    mode: requireTLS
    certificateKeyFile: <путь до файла с приватным ключом и сертификатом сервера (PEM)>
    CAFile: <путь до сертификата центра сертификации (CA)>
```

После перезапуска сервиса, MongoDB будет принимать подключения только от клиентов, использующих TLS.

Создайте systemd-сервис по пути `/etc/systemd/system/mongod.service` :

```
[Unit]
Description=MongoDB Database Server
Documentation=https://docs.mongodb.org/manual
After=network.target

[Service]
User=mongodb
Group=mongodb
ExecStart=/usr/local/mongodb/bin/mongod --config /etc/mongod.conf
PIDFile=/run/mongod.pid
Restart=always
LimitFSIZE=infinity
LimitCPU=infinity
LimitAS=infinity
LimitNOFILE=256000
LimitNPROC=256000
LimitMEMLOCK=infinity
TasksMax=infinity
TasksAccounting=false

[Install]
WantedBy=multi-user.target
```

Добавьте сервис в автозагрузку, запустите и проверьте статус:

```
{
  systemctl enable mongod.service
  systemctl start mongod.service
  systemctl status mongod.service
}
```

Скачайте и установите [MongoDB Shell](#), подключитесь к базе данных и создайте пользователя в базе `admin` :

```
use admin;
db.createUser({
  "user": "moodrec",
  "pwd": "abcdefghijklmnopqrstuvwxyABCDEF",
  "roles": [{
    "role": "root",
    "db": "moodrec"
  }]
})
```

Обновите параметры подключения в конфигурации `main_config.yaml`:

```
mongo: &mongo_config
database: "moodrec"
username: "moodrec"
password: "abcdefghijklmnopqrstuvwxyABCDEF"
auth_source: "admin"
host: "127.0.0.1"
port: "27017"
```

Установка ClickHouse

TIP

База данных ClickHouse используется для хранения истории, статистики и формирования пользовательских отчетов. Ниже представлена инструкция по установке из DEB-пакетов.

Установите зависимости и добавьте официальный репозиторий:

```
{
  apt update
  apt install -y apt-transport-https ca-certificates dirmngr gnupg
  apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv 8919F6BD2B48D754
  echo "deb https://packages.clickhouse.com/deb stable main" | tee
  /etc/apt/sources.list.d/clickhouse.list
}
```

Установите ClickHouse:

```
{
  apt update
  apt install -y clickhouse-server clickhouse-client
}
```

Добавьте сервис в автозагрузку и запустите:

```
{
  systemctl enable clickhouse-server.service
  systemctl start clickhouse-server.service
  systemctl status clickhouse-server.service
}
```

Обновите параметры подключения в конфигурации `configs/release/main_config.yaml`:

```
clickHouse: &clickHouse_config
host: "127.0.0.1"
port: 9000
user: "moodrec"
password: "moodrec"
db_name: "moodrec"
```

Установка Kvrocks

ТИП

Kvrocks — альтернатива Redis, построенная на RocksDB. В отличие от Redis, он сохраняет данные на диск, что делает его подходящим для хранения больших объемов данных.

Установите зависимости для сборки:

```
{
  apt update
  apt install -y git build-essential cmake libtool python3
}
```

Перейдите на [страницу релизов](#) проекта на GitHub, скачайте последнюю версию и распакуйте архив:

```
{
  wget https://github.com/apache/kvrocks/archive/refs/tags/v2.12.1.tar.gz
  tar xzf v2.12.1.tar.gz
}
```

Выполните сборку и установку:

```
{
  cd kvrocks-2.12.1
  ./x.py build -DPORTABLE=1 -DCMAKE_BUILD_TYPE=Release -j $(nproc)
  mv build/kvrocks /usr/local/bin/
}
```

Создайте пользователя и каталоги:

```
{
  groupadd kvrocks
  useradd -r -g kvrocks -s /bin/false kvrocks
  mkdir -p /var/{lib,log}/kvrocks /etc/kvrocks
  chown -R kvrocks:kvrocks /var/{lib,log}/kvrocks
}
```

Создайте файл конфигурации `/etc/kvrocks/kvrocks.conf`:

```
# kvrocks.conf
bind 0.0.0.0
port 6666
db-name moodrec.db
dir /var/lib/kvrocks
log-dir /var/log/kvrocks
log-retention-days 7
backup-dir /var/lib/kvrocks/backup
requirepass abcdefghijklmnopqrstuvwxyzABCDEF
supervised systemd
workers 8
```

Создайте systemd-сервис по пути `/etc/systemd/system/kvrocks.service`:

```
[Unit]
Description=kvrocks SSD key-value database
Documentation=https://github.com/apache/kvrocks
Wants=network-online.target
After=network-online.target

[Service]
User=kvrocks
Group=kvrocks
Type=notify
ExecStart=/usr/local/bin/kvrocks -c /etc/kvrocks/kvrocks.conf
WorkingDirectory=/var/lib/kvrocks
Restart=on-failure
ExecStop=/bin/kill -s TERM $MAINPID
RestartSec=10s
LimitNOFILE=100000
LimitNPROC=4096
TimeoutSec=300
NoNewPrivileges=yes

[Install]
WantedBy=multi-user.target
Alias=kvrocks.service
```

Добавьте сервис в автозагрузку и запустите:

```
{
  systemctl enable kvrocks.service
  systemctl start kvrocks.service
  systemctl status kvrocks.service
}
```

Обновите параметры подключения в конфигурации `configs/release/main_config.yaml`:

```
redis: &redis_config
  host: "127.0.0.1"
  port: 6666
  user: "moodrec"
  password: "abcdefghijklmnopqrstuvwxyZABCDEF"
  read_timeout: 10
  write_timeout: 10
  pool_size: 100
```

Запуск сервиса

Перед запуском сервиса убедитесь, что вы заполнили файл конфигурации `main_config.yaml`. Описание всех параметров доступно в разделе **Параметры конфигурации**.

Для управления микросервисами MoodRec используется исполняемый файл `moodrec`. Справка по командам:

Usage:

```
moodrec [flags]
moodrec [command]
```

Available Commands:

```
list          List platform information
start         Start platform services
stop          Stop platform services
restart       Restart platform services
reload-nginx  Reload nginx configuration
help          Help about any command
completion    Generate the autocompletion script for the specified shell
```

Flags:

```
--appname      show application full name
--config string config path (default
"/opt/configs/release/main_config.yaml")
--debug        debug mode (for development purposes)
-h, --help     help for moodrec
-v, --version  version for moodrec
```

Use "moodrec [command] --help" for more information about a command.

На этом установка завершена, сервис готов к запуску и дальнейшей настройке. Дальнейшая настройка происходит в интерфейсе MoodRec.

Параметры конфигурации

Файл конфигурации находится по пути `configs/release/main_config.yaml`.

Общие настройки баз данных

databases.mongo.database `string`

Описание: Имя базы данных MongoDB

По умолчанию: "moodrec"

Обязательный: Да

databases.mongo.username `string`

Описание: Имя пользователя MongoDB

По умолчанию: "root"

Обязательный: Да

databases.mongo.password `string`

Описание: Пароль пользователя MongoDB

По умолчанию: "abcdefghijklmnopqrstuvwxyzABCDEF"

Обязательный: Да

databases.mongo.auth_source `string`

Описание: База данных для аутентификации

По умолчанию: "admin"

Обязательный: Да

databases.mongo.host `string`

Описание: Хост MongoDB

По умолчанию: "127.0.0.1"

Обязательный: Да

databases.mongo.port `string`

Описание: Порт MongoDB

По умолчанию: "27017"

Обязательный: Да

databases.mongo.max_pool_size `int`

Описание: Максимальный размер пула соединений

По умолчанию: 100

Обязательный: Нет

databases.mongo.min_pool_size `int`

Описание: Минимальный размер пула соединений

По умолчанию: 1

Обязательный: Нет

databases.mongo.max_idle_time_ms `int`

Описание: Максимальное время простаивания соединения в миллисекундах

По умолчанию: 60000

Обязательный: Нет

databases.mongo.auth_mechanism `string`

Описание: Механизм аутентификации MongoDB

По умолчанию: "SCRAM-SHA-1"

Обязательный: Нет

databases.mongo.timeout `int`

Описание: Таймаут операций в миллисекундах

По умолчанию: 60000

Обязательный: Нет

databases.mongo.connect_timeout `int`

Описание: Таймаут подключения в миллисекундах

По умолчанию: 60000

Обязательный: Нет

databases.mongo.MONGODB_UPDATER_BATCH_SIZE `null`

Описание: Размер пакета для обновлений (не задан)

По умолчанию: Не указано

Обязательный: Нет

databases.mongo.MONGODB_UPDATER_PERIOD_MS `null`

Описание: Период обновлений в миллисекундах (не задан)

По умолчанию: Не указано

Обязательный: Нет

databases.mongo.MONGODB_SOCKET_TIMEOUT_MIN `null`

Описание: Таймаут сокета в минутах (не задан)

По умолчанию: Не указано

Обязательный: Нет

databases.mongo.MONGODB_SERVER_SELECTION_TIMEOUT `null`

Описание: Таймаут выбора сервера (не задан)

По умолчанию: Не указано

Обязательный: Нет

databases.redis.host `string`

Описание: Хост Redis

По умолчанию: "127.0.0.1"

Обязательный: Да

databases.redis.port `int`

Описание: Порт Redis

По умолчанию: 6666

Обязательный: Да

databases.redis.user `string`

Описание: Имя пользователя Redis

По умолчанию: "moodrec"

Обязательный: Да

databases.redis.password `string`

Описание: Пароль Redis

По умолчанию: "abcdefghijklmnopqrstuvwxyzABCDEF"

Обязательный: Да

databases.redis.read_timeout `int`

Описание: Таймаут чтения в секундах

По умолчанию: 10

Обязательный: Нет

databases.redis.write_timeout `int`

Описание: Таймаут записи в секундах

По умолчанию: 10

Обязательный: Нет

databases.redis.pool_size `int`

Описание: Размер пула соединений

По умолчанию: 100

Обязательный: Нет

databases.clickHouse.host `string`

Описание: Хост ClickHouse

По умолчанию: "127.0.0.1"

Обязательный: Да

databases.clickHouse.port `int`

Описание: Порт ClickHouse

По умолчанию: 9000

Обязательный: Да

databases.clickHouse.user `string`

Описание: Имя пользователя ClickHouse

По умолчанию: "moodrec"

Обязательный: Да

databases.clickHouse.password `string`

Описание: Пароль ClickHouse

По умолчанию: "moodrec"

Обязательный: Да

databases.clickHouse.db_name string

Описание: Имя базы данных ClickHouse

По умолчанию: "moodrec"

Обязательный: Да

Безопасность и лицензирование

security.http_token string

Описание: Токен для HTTP-аутентификации

По умолчанию: "abcdefghijklmnopqrstuvwxyZABCDEF"

Обязательный: Да

security.grpc_token string

Описание: Токен для gRPC-аутентификации

По умолчанию: "abcdefghijklmnopqrstuvwxyZABCDEF"

Обязательный: Да

license.die_when_license_fails bool

Описание: Завершать работу при ошибке лицензии

По умолчанию: true

Обязательный: Нет

license.notification_license_warn_days int

Описание: За сколько дней до истечения лицензии начинать предупреждения

По умолчанию: 10

Обязательный: Нет

license.proxy string

Описание: Прокси-сервер для проверки лицензии

По умолчанию: ""

Обязательный: Нет

license.is_debug bool

Описание: Режим отладки лицензии

По умолчанию: false

Обязательный: Нет

Общие настройки Kafka

kafka.seeds array

Описание: Список брокеров Kafka (хост:порт)

По умолчанию: ["127.0.0.1:29092", "127.0.0.1:29093", "127.0.0.1:29094"]

Обязательный: Да

kafka.default_topic_config.retention.ms string

Описание: Время хранения сообщений в миллисекундах

По умолчанию: "43200000"

Обязательный: Нет

kafka.default_topic_config.retention.bytes string

Описание: Лимит размера хранилища топика (в байтах, -1 без ограничений)

По умолчанию: "-1"

Обязательный: Нет

kafka.default_topic_config.segment.bytes string

Описание: Максимальный размер сегмента лога в байтах

По умолчанию: "1073741824"

Обязательный: Нет

kafka.default_topic_config.cleanup.policy string

Описание: Политика очистки (delete/compact)

По умолчанию: "delete"

Обязательный: Нет

kafka.default_topic_config.max.message.bytes string

Описание: Максимальный размер сообщения в байтах

По умолчанию: "10242880"

Обязательный: Нет

kafka.default_topic_config.min.insync.replicas string

Описание: Минимальное количество синхронных реплик для подтверждения записи

По умолчанию: "3"

Обязательный: Нет

kafka.default_topic_config.compression.type string

Описание: Тип сжатия (uncompressed, gzip, snappy, lz4, zstd)

По умолчанию: "uncompressed"

Обязательный: Нет

kafka.default_topic_config.message.timestamp.type string

Описание: Тип временной метки (CreateTime/LogAppendTime)

По умолчанию: "CreateTime"

Обязательный: Нет

kafka.default_topic_config.file.delete.delay.ms string

Описание: Задержка перед удалением устаревших сегментов (мс)

По умолчанию: "30000"

Обязательный: Нет

kafka.default_topic_config.compression.lz4.level string

Описание: Уровень сжатия LZ4

По умолчанию: "9"

Обязательный: Нет

kafka.default_topic_config.compression.gzip.level string

Описание: Уровень сжатия GZIP

По умолчанию: "1"

Обязательный: Нет

kafka.default_topic_config.compression.zstd.level string

Описание: Уровень сжатия ZSTD

По умолчанию: "3"

Обязательный: Нет

kafka.default_topic_config.leader.replication.throttled.replicas string

Описание: Ограничение пропускной способности репликации лидера

По умолчанию: ""

Обязательный: Нет

kafka.default_topic_config.follower.replication.throttled.replicas string

Описание: Ограничение пропускной способности репликации фолловера

По умолчанию: ""

Обязательный: Нет

kafka.default_topic_config.message.downconversion.enable string

Описание: Включить понижающую конверсию сообщений для старых клиентов

По умолчанию: "false"

Обязательный: Нет

kafka.default_topic_config.message.format.version string

Описание: Версия формата сообщений

По умолчанию: "3.0-IV1"

Обязательный: Нет

kafka.default_topic_config.flush.ms string

Описание: Максимальное время перед сбросом на диск (мс)

По умолчанию: "10000"

Обязательный: Нет

kafka.default_topic_config.flush.messages string

Описание: Максимальное количество сообщений перед сбросом

По умолчанию: "10000"

Обязательный: Нет

kafka.default_topic_config.min.compaction.lag.ms string

Описание: Минимальная задержка перед компакцией (мс)

По умолчанию: "60000"

Обязательный: Нет

kafka.default_topic_config.preallocate string

Описание: Предварительно выделять место для сегментов

По умолчанию: "true"

Обязательный: Нет

kafka.default_topic_config.min.cleanable.dirty.ratio string

Описание: Доля «грязных» данных для запуска компакции

По умолчанию: "0.4"

Обязательный: Нет

kafka.default_topic_config.index.interval.bytes string

Описание: Интервал байтов между индексируемыми сообщениями

По умолчанию: "8192"

Обязательный: Нет

kafka.default_topic_config.unclean.leader.election.enable string

Описание: Разрешить небезопасное лидерство при отказах

По умолчанию: "false"

Обязательный: Нет

kafka.default_topic_config.delete.retention.ms string

Описание: Время хранения удалённых сообщений при log compaction (мс)

По умолчанию: "43200000"

Обязательный: Нет

kafka.default_topic_config.message.timestamp.after.max.ms string

Описание: Максимальное отклонение временной метки в будущее (мс)

По умолчанию: "86400000"

Обязательный: Нет

kafka.default_topic_config.message.timestamp.before.max.ms string

Описание: Максимальное отклонение временной метки в прошлое (мс)

По умолчанию: "86400000"

Обязательный: Нет

kafka.default_topic_config.segment.ms string

Описание: Максимальное время жизни сегмента перед созданием нового (мс)

По умолчанию: "259200000"

Обязательный: Нет

kafka.default_topic_config.segment.jitter.ms string

Описание: Случайное смещение к segment.ms для предотвращения массовой ротации

По умолчанию: "1000"

Обязательный: Нет

kafka.default_topic_config.message.timestamp.difference.max.ms `string`

Описание: Максимальная разница между временем сообщения и текущим временем (мс)

По умолчанию: "86400000"

Обязательный: Нет

kafka.default_topic_config.segment.index.bytes `string`

Описание: Максимальный размер индекса сегмента (байт)

По умолчанию: "10242880"

Обязательный: Нет

kafka.default_topic_config.max.compaction.lag.ms `string`

Описание: Максимальное время до компакции сообщения (мс)

По умолчанию: "86400000"

Обязательный: Нет

Сервис offers

offers.license `object`

Описание: Настройки лицензии (все поля из секции license)

По умолчанию: см. license

Обязательный: Да

offers.gracefully_shutdown_timeout `int`

Описание: Таймаут на корректное завершение работы (секунд)

По умолчанию: 60

Обязательный: Нет

offers.batch_size_offers `int`

Описание: Размер батча при импорте продуктов

По умолчанию: 100

Обязательный: Нет

offers.max_size_limit_feed `int`

Описание: Максимальный размер фида в Гб

По умолчанию: 2

Обязательный: Нет

offers.http_request_timeout `int`

Описание: Таймаут HTTP-запроса при загрузке фида (минуты)

По умолчанию: 5

Обязательный: Нет

offers.servers_settings.grpc_server.host `string`

Описание: Хост для gRPC-сервера

По умолчанию: "localhost"

Обязательный: Да

offers.servers_settings.grpc_server.port string

Описание: Порт для gRPC-сервера

По умолчанию: "1111"

Обязательный: Да

offers.servers_settings.grpc_clients.tasks array

Описание: Адреса gRPC-клиентов для сервиса tasks

По умолчанию: ["127.0.0.1:2222"]

Обязательный: Да

offers.servers_settings.grpc_clients.vectors array

Описание: Адреса gRPC-клиентов для сервиса vectors

По умолчанию: ["127.0.0.1:4444"]

Обязательный: Да

offers.servers_settings.grpc_timeouts.tasks int

Описание: Таймаут запроса к сервису tasks (секунд)

По умолчанию: 60

Обязательный: Нет

offers.servers_settings.grpc_timeouts.vectors int

Описание: Таймаут запроса к сервису vectors (секунд)

По умолчанию: 60

Обязательный: Нет

offers.servers_settings.grpc_pool_size.tasks int

Описание: Количество соединений в пуле для tasks

По умолчанию: 10

Обязательный: Нет

offers.servers_settings.grpc_pool_size.vectors int

Описание: Количество соединений в пуле для vectors

По умолчанию: 60

Обязательный: Нет

offers.logger.level string

Описание: Уровень логирования (trace, debug, info, warn, error)

По умолчанию: "trace"

Обязательный: Да

offers.security object

Описание: Настройки безопасности (http_token, grpc_token)

По умолчанию: см. security

Обязательный: Да

offers.databases.mongo `object`

Описание: Параметры MongoDB для сервиса offers (все поля из databases.mongo)

По умолчанию: см. databases.mongo

Обязательный: Да

offers.databases.redis `object`

Описание: Параметры Redis для сервиса offers (все поля из databases.redis)

По умолчанию: см. databases.redis

Обязательный: Да

offers.brokers.kafka.seeds `array`

Описание: Список брокеров Kafka

По умолчанию: см. kafka.seeds

Обязательный: Да

offers.brokers.kafka.retry_timeout_consumer `int`

Описание: Таймаут повторных попыток consumer (секунд)

По умолчанию: 5

Обязательный: Нет

offers.brokers.kafka.retry_timeout_producer `int`

Описание: Таймаут повторных попыток producer (секунд)

По умолчанию: 5

Обязательный: Нет

offers.brokers.kafka.retry_max_count_producer `int`

Описание: Максимальное количество повторных попыток producer

По умолчанию: 5

Обязательный: Нет

offers.brokers.kafka.topics.update_offers_in_index.replication_factor `int`

Описание: Фактор репликации для топика update_offers_in_index

По умолчанию: 3

Обязательный: Нет

offers.brokers.kafka.topics.update_offers_in_index.partitions `int`

Описание: Количество партиций для топика update_offers_in_index

По умолчанию: 3

Обязательный: Нет

offers.brokers.kafka.topics.update_offers_in_index.configs `object`

Описание: Конфигурация топика (все поля из default_topic_config)

По умолчанию: см. kafka.default_topic_config

Обязательный: Нет

Сервис tasks

tasks.gracefully_shutdown_timeout `int`

Описание: Таймаут на корректное завершение работы (секунд)

По умолчанию: 60

Обязательный: Нет

tasks.servers_settings.grpc_server.host `string`

Описание: Хост gRPC-сервера

По умолчанию: "localhost"

Обязательный: Да

tasks.servers_settings.grpc_server.port `string`

Описание: Порт gRPC-сервера

По умолчанию: "2222"

Обязательный: Да

tasks.servers_settings.grpc_clients.audience `array`

Описание: Адреса gRPC-клиентов для сервиса audience

По умолчанию: ["127.0.0.1:7777"]

Обязательный: Да

tasks.servers_settings.grpc_clients.offers `array`

Описание: Адреса gRPC-клиентов для сервиса offers

По умолчанию: ["127.0.0.1:1111"]

Обязательный: Да

tasks.servers_settings.grpc_clients.catboss `array`

Описание: Адреса gRPC-клиентов для сервиса catboss

По умолчанию: ["127.0.0.1:5555"]

Обязательный: Да

tasks.servers_settings.grpc_timeouts.audience `int`

Описание: Таймаут запроса к audience (секунд)

По умолчанию: 60

Обязательный: Нет

tasks.servers_settings.grpc_timeouts.offers `int`

Описание: Таймаут запроса к offers (секунд)

По умолчанию: 60

Обязательный: Нет

tasks.servers_settings.grpc_timeouts.catboss `int`

Описание: Таймаут запроса к catboss (секунд)

По умолчанию: 60

Обязательный: Нет

tasks.servers_settings.grpc_pool_size.audience `int`

Описание: Размер пула соединений для audience

По умолчанию: 10

Обязательный: Нет

tasks.servers_settings.grpc_pool_size.offers `int`

Описание: Размер пула соединений для offers

По умолчанию: 10

Обязательный: Нет

tasks.servers_settings.grpc_pool_size.catboss `int`

Описание: Размер пула соединений для catboss

По умолчанию: 10

Обязательный: Нет

tasks.security `object`

Описание: Настройки безопасности (http_token, grpc_token)

По умолчанию: см. security

Обязательный: Да

tasks.logger.level `string`

Описание: Уровень логирования

По умолчанию: "trace"

Обязательный: Да

tasks.databases.mongo `object`

Описание: Параметры MongoDB для сервиса tasks

По умолчанию: см. databases.mongo

Обязательный: Да

Сервис trans

trans.lib_dir_path `string`

Описание: Директория с библиотеками для векторизации

По умолчанию: "/opt/moodrec/lib/"

Обязательный: Да

trans.model_path `string`

Описание: Путь к директории с моделью

По умолчанию: "/opt/moodrec/models/model-example"

Обязательный: Да

trans.model_file_name `string`

Описание: Имя файла модели ONNX

По умолчанию: "model.onnx"

Обязательный: Да

trans.full_clean_text `bool`

Описание: Выполнять полную очистку текста перед векторизацией (для тестирования)

По умолчанию: false

Обязательный: Нет

trans.gracefully_shutdown_timeout `int`

Описание: Таймаут на корректное завершение работы (секунд)

По умолчанию: 60

Обязательный: Нет

trans.count_goroutines_for_vectorization `int`

Описание: Количество горутин для параллельной векторизации

По умолчанию: 4

Обязательный: Нет

trans.servers_settings.grpc_server.host `string`

Описание: Хост gRPC-сервера

По умолчанию: "localhost"

Обязательный: Да

trans.servers_settings.grpc_server.port `string`

Описание: Порт gRPC-сервера

По умолчанию: "1234"

Обязательный: Да

trans.security `object`

Описание: Настройки безопасности

По умолчанию: см. security

Обязательный: Да

trans.logger.level `string`

Описание: Уровень логирования

По умолчанию: "trace"

Обязательный: Да

trans.databases.mongo `object`

Описание: Параметры MongoDB

По умолчанию: см. databases.mongo

Обязательный: Да

trans.databases.redis `object`

Описание: Параметры Redis

По умолчанию: см. databases.redis

Обязательный: Да

trans.brokers.kafka.subscribe_count_goroutines.update_offers_in_index `int`

Описание: Количество горутин для чтения топика update_offers_in_index

По умолчанию: 3

Обязательный: Нет

trans.brokers.kafka.seeds `array`

Описание: Список брокеров Kafka

По умолчанию: см. kafka.seeds

Обязательный: Да

trans.brokers.kafka.retry_timeout_consumer `int`

Описание: Таймаут повторных попыток consumer (секунд)

По умолчанию: 5

Обязательный: Нет

trans.brokers.kafka.retry_timeout_producer `int`

Описание: Таймаут повторных попыток producer (секунд)

По умолчанию: 5

Обязательный: Нет

trans.brokers.kafka.retry_max_count_producer `int`

Описание: Максимальное количество повторных попыток producer

По умолчанию: 5

Обязательный: Нет

trans.brokers.kafka.topics.save_offers_to_index.replication_factor `int`

Описание: Фактор репликации для топика save_offers_to_index

По умолчанию: 3

Обязательный: Нет

trans.brokers.kafka.topics.save_offers_to_index.partitions `int`

Описание: Количество партиций для топика save_offers_to_index

По умолчанию: 3

Обязательный: Нет

trans.brokers.kafka.topics.save_offers_to_index.configs `object`

Описание: Конфигурация топика

По умолчанию: см. kafka.default_topic_config

Обязательный: Нет

Сервис vectors

vectors.count_workers_index `int`

Описание: Количество воркеров для работы с индексом

По умолчанию: 120

Обязательный: Нет

vectors.vectors_update_interval `int`

Описание: Интервал обновления векторной базы (минуты)

По умолчанию: 1

Обязательный: Нет

vectors.gracefully_shutdown_timeout `int`

Описание: Таймаут на корректное завершение работы (секунд)

По умолчанию: 60

Обязательный: Нет

vectors.vector_dimension `int`

Описание: Размерность вектора

По умолчанию: 384

Обязательный: Да

vectors.index_file_dir `string`

Описание: Директория для хранения файла индекса

По умолчанию: "/opt/moodrec/index"

Обязательный: Да

vectors.databases.redis `object`

Описание: Параметры Redis

По умолчанию: см. databases.redis

Обязательный: Да

vectors.servers_settings.grpc_server.host `string`

Описание: Хост gRPC-сервера

По умолчанию: "localhost"

Обязательный: Да

vectors.servers_settings.grpc_server.port `string`

Описание: Порт gRPC-сервера

По умолчанию: "4444"

Обязательный: Да

vectors.servers_settings.grpc_clients.offers `array`

Описание: Адреса gRPC-клиентов для offers

По умолчанию: ["127.0.0.1:1111"]

Обязательный: Да

vectors.servers_settings.grpc_clients.catboss `array`

Описание: Адреса gRPC-клиентов для catboss

По умолчанию: ["127.0.0.1:5555"]

Обязательный: Да

vectors.servers_settings.grpc_timeouts.offers **int**

Описание: Таймаут запроса к offers (секунд)

По умолчанию: 60

Обязательный: Нет

vectors.servers_settings.grpc_timeouts.catboss **int**

Описание: Таймаут запроса к catboss (секунд)

По умолчанию: 60

Обязательный: Нет

vectors.servers_settings.grpc_pool_size.offers **int**

Описание: Размер пула соединений для offers

По умолчанию: 10

Обязательный: Нет

vectors.servers_settings.grpc_pool_size.catboss **int**

Описание: Размер пула соединений для catboss

По умолчанию: 10

Обязательный: Нет

vectors.security **object**

Описание: Настройки безопасности

По умолчанию: см. security

Обязательный: Да

vectors.logger.level **string**

Описание: Уровень логирования

По умолчанию: "trace"

Обязательный: Да

vectors.brokers.kafka.subscribe_count_goroutines.save_offers_to_index **int**

Описание: Количество горутин для чтения топика save_offers_to_index

По умолчанию: 3

Обязательный: Нет

vectors.brokers.kafka.seeds **array**

Описание: Список брокеров Kafka

По умолчанию: см. kafka.seeds

Обязательный: Да

vectors.brokers.kafka.retry_timeout_consumer **int**

Описание: Таймаут повторных попыток consumer (секунд)

По умолчанию: 5

Обязательный: Нет

vectors.brokers.kafka.retry_timeout_producer `int`

Описание: Таймаут повторных попыток producer (секунд)

По умолчанию: 5

Обязательный: Нет

vectors.brokers.kafka.retry_max_count_producer `int`

Описание: Максимальное количество повторных попыток producer

По умолчанию: 5

Обязательный: Нет

Сервис events

events.security `object`

Описание: Настройки безопасности

По умолчанию: см. security

Обязательный: Да

events.withSortByCatName `bool`

Описание: Использовать сортировку по названию категории при выборе топ категорий

По умолчанию: true

Обязательный: Нет

events.custom_kafka_events_topics.json_events_topic.schema `string`

Описание: Формат схемы для json_events_topic (json/proto)

По умолчанию: "json"

Обязательный: Нет

events.custom_kafka_events_topics.json_events_topic.subscribe_count_goroutines `int`

Описание: Количество горутин для чтения json_events_topic

По умолчанию: 3

Обязательный: Нет

events.custom_kafka_events_topics.proto_events_topic.schema `string`

Описание: Формат схемы для proto_events_topic (proto)

По умолчанию: "proto"

Обязательный: Нет

events.custom_kafka_events_topics.proto_events_topic.subscribe_count_goroutines `int`

Описание: Количество горутин для чтения proto_events_topic

По умолчанию: 3

Обязательный: Нет

events.custom_kafka_events_topics.proto_events_topic.message_name `string`

Описание: Имя сообщения в proto-файле

По умолчанию: "Event"

Обязательный: Да (если schema=proto)

events.gracefully_shutdown_timeout `int`

Описание: Таймаут на корректное завершение работы (секунд)

По умолчанию: 60

Обязательный: Нет

events.session_limit_top_categories `int`

Описание: Количество топ категорий, хранимых для сессии

По умолчанию: 3

Обязательный: Нет

events.session_events_ttl `int`

Описание: Время жизни событий сессии (минуты)

По умолчанию: 900

Обязательный: Нет

events.uid_limit_top_categories `int`

Описание: Количество топ категорий, хранимых для пользователя

По умолчанию: 3

Обязательный: Нет

events.uid_events_ttl `int`

Описание: Время жизни событий пользователя (дни)

По умолчанию: 90

Обязательный: Нет

events.servers_settings.grpc_clients.audience `array`

Описание: Адреса gRPC-клиентов для audience

По умолчанию: ["127.0.0.1:7777"]

Обязательный: Да

events.servers_settings.grpc_clients.offers `array`

Описание: Адреса gRPC-клиентов для offers

По умолчанию: ["127.0.0.1:1111"]

Обязательный: Да

events.servers_settings.grpc_pool_size.audience `int`

Описание: Размер пула соединений для audience

По умолчанию: 100

Обязательный: Нет

events.servers_settings.grpc_pool_size.offers `int`

Описание: Размер пула соединений для offers

По умолчанию: 100

Обязательный: Нет

events.servers_settings.grpc_timeouts.audience **int**

Описание: Таймаут запроса к audience (секунд)

По умолчанию: 60

Обязательный: Нет

events.servers_settings.grpc_timeouts.offers **int**

Описание: Таймаут запроса к offers (секунд)

По умолчанию: 60

Обязательный: Нет

events.servers_settings.grpc_server.host **string**

Описание: Хост gRPC-сервера

По умолчанию: "localhost"

Обязательный: Да

events.servers_settings.grpc_server.port **string**

Описание: Порт gRPC-сервера

По умолчанию: "9999"

Обязательный: Да

events.logger.level **string**

Описание: Уровень логирования

По умолчанию: "trace"

Обязательный: Да

events.databases.redis **object**

Описание: Параметры Redis

По умолчанию: см. databases.redis

Обязательный: Да

events.brokers.kafka.seeds **array**

Описание: Список брокеров Kafka

По умолчанию: см. kafka.seeds

Обязательный: Да

events.brokers.kafka.subscribe_count_goroutines.events **int**

Описание: Количество горутин для чтения топика events

По умолчанию: 3

Обязательный: Нет

events.brokers.kafka.subscribe_count_goroutines.uid_reassign **int**

Описание: Количество горутин для чтения топика uid_reassign

По умолчанию: 3

Обязательный: Нет

events.brokers.kafka.retry_timeout_consumer `int`

Описание: Таймаут повторных попыток consumer (секунд)

По умолчанию: 1

Обязательный: Нет

events.brokers.kafka.retry_timeout_producer `int`

Описание: Таймаут повторных попыток producer (секунд)

По умолчанию: 1

Обязательный: Нет

events.brokers.kafka.retry_max_count_producer `int`

Описание: Максимальное количество повторных попыток producer

По умолчанию: 1

Обязательный: Нет

events.brokers.kafka.async_producer_worker_count `int`

Описание: Количество воркеров для асинхронной отправки

По умолчанию: 50

Обязательный: Нет

events.brokers.kafka.topics.events.replication_factor `int`

Описание: Фактор репликации для топика events

По умолчанию: 3

Обязательный: Нет

events.brokers.kafka.topics.events.partitions `int`

Описание: Количество партиций для топика events

По умолчанию: 3

Обязательный: Нет

events.brokers.kafka.topics.events.configs `object`

Описание: Конфигурация топика events

По умолчанию: см. kafka.default_topic_config

Обязательный: Нет

Сервис catboss

catboss.security `object`

Описание: Настройки безопасности

По умолчанию: см. security

Обязательный: Да

catboss.host_url `string`

Описание: URL хоста для внешнего доступа

По умолчанию: "<https://example.com>"

Обязательный: Да

catboss.sdk_url string

Описание: URL для загрузки SDK (трекера)

По умолчанию: "<https://example.com/static/moodrec-tracker.js>"

Обязательный: Да

catboss.gracefully_shutdown_timeout int

Описание: Таймаут на корректное завершение работы (секунд)

По умолчанию: 60

Обязательный: Нет

catboss.servers_settings.grpc_server.host string

Описание: Хост gRPC-сервера

По умолчанию: "localhost"

Обязательный: Да

catboss.servers_settings.grpc_server.port string

Описание: Порт gRPC-сервера

По умолчанию: "5555"

Обязательный: Да

catboss.servers_settings.grpc_clients.tasks array

Описание: Адреса gRPC-клиентов для tasks

По умолчанию: ["127.0.0.1:2222"]

Обязательный: Да

catboss.servers_settings.grpc_clients.vectors array

Описание: Адреса gRPC-клиентов для vectors

По умолчанию: ["127.0.0.1:4444"]

Обязательный: Да

catboss.servers_settings.grpc_clients.events array

Описание: Адреса gRPC-клиентов для events

По умолчанию: ["127.0.0.1:9999"]

Обязательный: Да

catboss.servers_settings.grpc_clients.offers array

Описание: Адреса gRPC-клиентов для offers

По умолчанию: ["127.0.0.1:1111"]

Обязательный: Да

catboss.servers_settings.grpc_clients.audience array

Описание: Адреса gRPC-клиентов для audience

По умолчанию: ["127.0.0.1:7777"]

Обязательный: Да

catboss.servers_settings.grpc_clients.auth **array**

Описание: Адреса gRPC-клиентов для auth

По умолчанию: ["127.0.0.1:8888"]

Обязательный: Да

catboss.servers_settings.grpc_timeouts.tasks **int**

Описание: Таймаут запроса к tasks (секунд)

По умолчанию: 60

Обязательный: Нет

catboss.servers_settings.grpc_timeouts.events **int**

Описание: Таймаут запроса к events (секунд)

По умолчанию: 60

Обязательный: Нет

catboss.servers_settings.grpc_timeouts.vectors **int**

Описание: Таймаут запроса к vectors (секунд)

По умолчанию: 60

Обязательный: Нет

catboss.servers_settings.grpc_timeouts.offers **int**

Описание: Таймаут запроса к offers (секунд)

По умолчанию: 60

Обязательный: Нет

catboss.servers_settings.grpc_timeouts.audience **int**

Описание: Таймаут запроса к audience (секунд)

По умолчанию: 60

Обязательный: Нет

catboss.servers_settings.grpc_timeouts.auth **int**

Описание: Таймаут запроса к auth (секунд)

По умолчанию: 60

Обязательный: Нет

catboss.servers_settings.grpc_pool_size.tasks **int**

Описание: Размер пула соединений для tasks

По умолчанию: 10

Обязательный: Нет

catboss.servers_settings.grpc_pool_size.events **int**

Описание: Размер пула соединений для events

По умолчанию: 10

Обязательный: Нет

catboss.servers_settings.grpc_pool_size.vectors **int**

Описание: Размер пула соединений для vectors

По умолчанию: 10

Обязательный: Нет

catboss.servers_settings.grpc_pool_size.offers **int**

Описание: Размер пула соединений для offers

По умолчанию: 10

Обязательный: Нет

catboss.servers_settings.grpc_pool_size.audience **int**

Описание: Размер пула соединений для audience

По умолчанию: 10

Обязательный: Нет

catboss.servers_settings.grpc_pool_size.auth **int**

Описание: Размер пула соединений для auth

По умолчанию: 10

Обязательный: Нет

catboss.logger.level **string**

Описание: Уровень логирования

По умолчанию: "trace"

Обязательный: Да

catboss.brokers.kafka.subscribe_count_goroutines.uid_reassign **int**

Описание: Количество горутин для чтения топика uid_reassign

По умолчанию: 3

Обязательный: Нет

catboss.brokers.kafka.seeds **array**

Описание: Список брокеров Kafka

По умолчанию: см. kafka.seeds

Обязательный: Да

catboss.brokers.kafka.retry_timeout_consumer **int**

Описание: Таймаут повторных попыток consumer (секунд)

По умолчанию: 5

Обязательный: Нет

catboss.brokers.kafka.retry_timeout_producer **int**

Описание: Таймаут повторных попыток producer (секунд)

По умолчанию: 5

Обязательный: Нет

catboss.brokers.kafka.retry_max_count_producer `int`

Описание: Максимальное количество повторных попыток producer

По умолчанию: 5

Обязательный: Нет

catboss.brokers.kafka.async_producer_worker_count `int`

Описание: Количество воркеров для асинхронной отправки

По умолчанию: 50

Обязательный: Нет

catboss.brokers.kafka.topics.impressions.replication_factor `int`

Описание: Фактор репликации для топика impressions

По умолчанию: 3

Обязательный: Нет

catboss.brokers.kafka.topics.impressions.partitions `int`

Описание: Количество партиций для топика impressions

По умолчанию: 3

Обязательный: Нет

catboss.brokers.kafka.topics.impressions.configs `object`

Описание: Конфигурация топика impressions

По умолчанию: см. kafka.default_topic_config

Обязательный: Нет

catboss.databases.mongo `object`

Описание: Параметры MongoDB

По умолчанию: см. databases.mongo

Обязательный: Да

catboss.databases.redis `object`

Описание: Параметры Redis

По умолчанию: см. databases.redis

Обязательный: Да

catboss.databases.clickHouse `object`

Описание: Параметры ClickHouse

По умолчанию: см. databases.clickHouse

Обязательный: Да

Сервис audience

audience.security `object`

Описание: Настройки безопасности

По умолчанию: см. security

Обязательный: Да

audience.gracefully_shutdown_timeout `int`

Описание: Таймаут на корректное завершение работы (секунд)

По умолчанию: 60

Обязательный: Нет

audience.servers_settings.grpc_server.host `string`

Описание: Хост gRPC-сервера

По умолчанию: "localhost"

Обязательный: Да

audience.servers_settings.grpc_server.port `string`

Описание: Порт gRPC-сервера

По умолчанию: "7777"

Обязательный: Да

audience.servers_settings.grpc_clients.tasks `array`

Описание: Адреса gRPC-клиентов для tasks

По умолчанию: ["127.0.0.1:2222"]

Обязательный: Да

audience.servers_settings.grpc_clients.catboss `array`

Описание: Адреса gRPC-клиентов для catboss

По умолчанию: ["127.0.0.1:5555"]

Обязательный: Да

audience.servers_settings.grpc_timeouts.tasks `int`

Описание: Таймаут запроса к tasks (секунд)

По умолчанию: 60

Обязательный: Нет

audience.servers_settings.grpc_timeouts.catboss `int`

Описание: Таймаут запроса к catboss (секунд)

По умолчанию: 60

Обязательный: Нет

audience.servers_settings.grpc_pool_size.tasks `int`

Описание: Размер пула соединений для tasks

По умолчанию: 10

Обязательный: Нет

audience.servers_settings.grpc_pool_size.catboss `int`

Описание: Размер пула соединений для catboss

По умолчанию: 10

Обязательный: Нет

audience.logger.level `string`

Описание: Уровень логирования

По умолчанию: "trace"

Обязательный: Да

audience.databases.redis `object`

Описание: Параметры Redis

По умолчанию: см. databases.redis

Обязательный: Да

audience.databases.mongo `object`

Описание: Параметры MongoDB

По умолчанию: см. databases.mongo

Обязательный: Да

audience.brokers.kafka.seeds `array`

Описание: Список брокеров Kafka

По умолчанию: см. kafka.seeds

Обязательный: Да

audience.brokers.kafka.subscribe_count_goroutines.events `int`

Описание: Количество горутин для чтения топика events

По умолчанию: 3

Обязательный: Нет

audience.brokers.kafka.retry_timeout_consumer `int`

Описание: Таймаут повторных попыток consumer (секунд)

По умолчанию: 1

Обязательный: Нет

audience.brokers.kafka.retry_timeout_producer `int`

Описание: Таймаут повторных попыток producer (секунд)

По умолчанию: 1

Обязательный: Нет

audience.brokers.kafka.retry_max_count_producer `int`

Описание: Максимальное количество повторных попыток producer

По умолчанию: 1

Обязательный: Нет

audience.brokers.kafka.async_producer_worker_count `int`

Описание: Количество воркеров для асинхронной отправки

По умолчанию: 50

Обязательный: Нет

audience.brokers.kafka.topics.audience_mapping.replication_factor `int`

Описание: Фактор репликации для топика audience_mapping

По умолчанию: 3

Обязательный: Нет

audience.brokers.kafka.topics.audience_mapping.partitions `int`

Описание: Количество партиций для топика audience_mapping

По умолчанию: 3

Обязательный: Нет

audience.brokers.kafka.topics.audience_mapping.configs `object`

Описание: Конфигурация топика audience_mapping

По умолчанию: см. kafka.default_topic_config

Обязательный: Нет

audience.brokers.kafka.topics.uid_reassign.replication_factor `int`

Описание: Фактор репликации для топика uid_reassign

По умолчанию: 3

Обязательный: Нет

audience.brokers.kafka.topics.uid_reassign.partitions `int`

Описание: Количество партиций для топика uid_reassign

По умолчанию: 3

Обязательный: Нет

audience.brokers.kafka.topics.uid_reassign.configs `object`

Описание: Конфигурация топика uid_reassign

По умолчанию: см. kafka.default_topic_config

Обязательный: Нет

audience.brokers.kafka.topics.hard_ids_mapping.replication_factor `int`

Описание: Фактор репликации для топика hard_ids_mapping

По умолчанию: 3

Обязательный: Нет

audience.brokers.kafka.topics.hard_ids_mapping.partitions `int`

Описание: Количество партиций для топика hard_ids_mapping

По умолчанию: 3

Обязательный: Нет

audience.brokers.kafka.topics.hard_ids_mapping.configs `object`

Описание: Конфигурация топика `hard_ids_mapping`

По умолчанию: см. `kafka.default_topic_config`

Обязательный: Нет

Сервис `gateway`

`gateway.gracefully_shutdown_timeout` `int`

Описание: Таймаут на корректное завершение работы (секунд)

По умолчанию: 60

Обязательный: Нет

`gateway.request_timeout` `int`

Описание: Максимальное время жизни HTTP-запроса (секунды)

По умолчанию: 2

Обязательный: Нет

`gateway.servers_settings.http_server.host` `string`

Описание: Хост HTTP-сервера

По умолчанию: "0.0.0.0"

Обязательный: Да

`gateway.servers_settings.http_server.port` `string`

Описание: Порт HTTP-сервера

По умолчанию: "80"

Обязательный: Да

`gateway.servers_settings.grpc_clients.events` `array`

Описание: Адреса gRPC-клиентов для `events`

По умолчанию: ["127.0.0.1:9999"]

Обязательный: Да

`gateway.servers_settings.grpc_clients.offers` `array`

Описание: Адреса gRPC-клиентов для `offers`

По умолчанию: ["127.0.0.1:1111"]

Обязательный: Да

`gateway.servers_settings.grpc_clients.catboss` `array`

Описание: Адреса gRPC-клиентов для `catboss`

По умолчанию: ["127.0.0.1:5555"]

Обязательный: Да

`gateway.servers_settings.grpc_clients.audience` `array`

Описание: Адреса gRPC-клиентов для `audience`

По умолчанию: ["127.0.0.1:7777"]

Обязательный: Да

`gateway.servers_settings.grpc_clients.auth` `array`

Описание: Адреса gRPC-клиентов для auth

По умолчанию: ["127.0.0.1:8888"]

Обязательный: Да

gateway.servers_settings.grpc_pool_size.events `int`

Описание: Размер пула соединений для events

По умолчанию: 100

Обязательный: Нет

gateway.servers_settings.grpc_pool_size.offers `int`

Описание: Размер пула соединений для offers

По умолчанию: 100

Обязательный: Нет

gateway.servers_settings.grpc_pool_size.catboss `int`

Описание: Размер пула соединений для catboss

По умолчанию: 100

Обязательный: Нет

gateway.servers_settings.grpc_pool_size.audience `int`

Описание: Размер пула соединений для audience

По умолчанию: 100

Обязательный: Нет

gateway.servers_settings.grpc_pool_size.auth `int`

Описание: Размер пула соединений для auth

По умолчанию: 10

Обязательный: Нет

gateway.servers_settings.grpc_timeouts.catboss `int`

Описание: Таймаут запроса к catboss (секунд)

По умолчанию: 60

Обязательный: Нет

gateway.servers_settings.grpc_timeouts.events `int`

Описание: Таймаут запроса к events (секунд)

По умолчанию: 60

Обязательный: Нет

gateway.servers_settings.grpc_timeouts.offers `int`

Описание: Таймаут запроса к offers (секунд)

По умолчанию: 60

Обязательный: Нет

gateway.servers_settings.grpc_timeouts.audience `int`

Описание: Таймаут запроса к audience (секунд)

По умолчанию: 60

Обязательный: Нет

gateway.servers_settings.grpc_timeouts.auth **int**

Описание: Таймаут запроса к auth (секунд)

По умолчанию: 60

Обязательный: Нет

gateway.security **object**

Описание: Настройки безопасности

По умолчанию: см. security

Обязательный: Да

gateway.logger.level **string**

Описание: Уровень логирования

По умолчанию: "trace"

Обязательный: Да

gateway.databases.mongo **object**

Описание: Параметры MongoDB

По умолчанию: см. databases.mongo

Обязательный: Да

Сервис auth

auth.invite_url **string**

Описание: Базовый URL для приглашений (добавляется токен)

По умолчанию: "<https://example.com/invite?token=>"

Обязательный: Да

auth.gracefully_shutdown_timeout **int**

Описание: Таймаут на корректное завершение работы (секунд)

По умолчанию: 60

Обязательный: Нет

auth.servers_settings.grpc_clients.catboss **array**

Описание: Адреса gRPC-клиентов для catboss

По умолчанию: ["127.0.0.1:5555"]

Обязательный: Да

auth.servers_settings.grpc_pool_size.catboss **int**

Описание: Размер пула соединений для catboss

По умолчанию: 100

Обязательный: Нет

auth.servers_settings.grpc_timeouts.catboss **int**

Описание: Таймаут запроса к catboss (секунд)

По умолчанию: 60

Обязательный: Нет

auth.servers_settings.grpc_server.host `string`

Описание: Хост gRPC-сервера

По умолчанию: "localhost"

Обязательный: Да

auth.servers_settings.grpc_server.port `string`

Описание: Порт gRPC-сервера

По умолчанию: "8888"

Обязательный: Да

auth.security `object`

Описание: Настройки безопасности

По умолчанию: см. security

Обязательный: Да

auth.logger.level `string`

Описание: Уровень логирования

По умолчанию: "trace"

Обязательный: Да

auth.databases.mongo `object`

Описание: Параметры MongoDB

По умолчанию: см. databases.mongo

Обязательный: Да

Обновление

Для обновления сервиса необходимо заменить **бинарные файлы** из архива с новым релизом. Дополнительные действий от пользователя не требуется, сервис готов к запуску.

Мониторинг работы сервиса и сбор логов

В зависимости от используемой в организации системы мониторинга, специалист должен выбрать способ отслеживания состояния процессов сервиса.

На странице описаны все процессы, которые необходимо поддерживать в рабочем состоянии для полноценной работы сервиса.

Рекомендуется использовать проверку на существование процесса по имени встроенными средствами системы мониторинга либо с помощью `pidof`.

В обязательном порядке по хосту (или виртуальной машине) должна собираться такая информация:

- потребление и количество свободного ОЗУ;
- потребление и количество свободного времени CPU;
- количество свободного места и информация по утилизации дисков (скорость, задержки, SMART);
- количество открытых файлов и соединений.

Базы данных и сервисы

Сервис	IP адрес по умолчанию	Порт	Важность	Доступ	Уведомление
Интерфейс платформы MoodRec	0.0.0.0	443 (HTTPS)	Чрезвычайная	Публичный или через VPN / allowlist	Недоступен веб-интерфейс
Публичный API (события, рекомендации, импорт)	0.0.0.0	443 (HTTPS)	Чрезвычайная	Публичный или в сетевом периметре клиента	Недоступны API-методы
MongoDB	127.0.0.1	27017 (TCP)	Чрезвычайная	Только из внутренней сети	Невозможно установить TCP-соединение
ClickHouse	127.0.0.1	9000 (TCP)	Чрезвычайная	Только из внутренней сети	Невозможно установить TCP-соединение
Apache Kafka	127.0.0.1	9092, 9093 (TCP)	Чрезвычайная	Только из внутренней сети	Невозможно установить TCP-соединение
KV Rocks	127.0.0.1	6666 (TCP)	Чрезвычайная	Только из внутренней сети	Невозможно установить TCP-соединение
Сервис работы с продуктами	127.0.0.1	1111 (TCP)	Высокая	Только из внутренней сети	Сервис недоступен по TCP
Сервис управления задачами	127.0.0.1	2222 (TCP)	Высокая	Только из внутренней сети	Сервис недоступен по TCP
Сервис расчёта продуктовых представлений	127.0.0.1	3333 (TCP)	Высокая	Только из внутренней сети	Сервис недоступен по TCP

Сервис	IP адрес по умолчанию	Порт	Важность	Доступ	Уведомление
Сервис обращений к векторному индексу	127.0.0.1	4444 (TCP)	Высокая	Только из внутренней сети	Сервис недоступен по TCP
Сервис выдачи рекомендаций	127.0.0.1	5555 (TCP)	Чрезвычайная	Только из внутренней сети	Сервис недоступен по TCP
Сервис аудиторий	127.0.0.1	7777 (TCP)	Высокая	Только из внутренней сети	Сервис недоступен по TCP
Сервис работы с пользователями	127.0.0.1	8888 (TCP)	Высокая	Только из внутренней сети	Сервис недоступен по TCP
Сервис обработки событий	127.0.0.1	9999 (TCP)	Чрезвычайная	Только из внутренней сети	Сервис недоступен по TCP
SMTP / SMTPS (оповещения)	внешний SMTP-сервер	25 или 465	Средняя	Исходящее соединение к SMTP-провайдеру	Невозможно установить соединение с SMTP

Дополнительно по каждому процессу

Можно собирать дополнительные данные: потребление памяти, количество открытых файлов и соединений, использование CPU и т.д. Пример — в разделе «Диагностика проблем», блок «Сбор информации для запроса в поддержку».

Процессорное время

CPU time: https://en.wikipedia.org/wiki/CPU_time

Тип метрики	Описание	Важность	Уведомление
CPU system time	Использование CPU процессом в процентах (system)	Информация	-
CPU iowait time	Использование CPU процессом в процентах (iowait)	Высокая	> 15% * количество ядер CPU
CPU user time	Использование CPU процессом в процентах (user)	Информация	-
CPU utilization	Использование CPU процессом в процентах (total)	Высокая	> 50% * количество ядер CPU

Память

RSS: https://en.wikipedia.org/wiki/Resident_set_size SWAP: https://en.wikipedia.org/wiki/Paging#Unix_and_Unix-like_systems

Тип метрики	Описание	Важность	Уведомление
RSS (resident set size)	Память процесса (RSS)	Высокая	> 20% от общего количества памяти
SWAP	Использование SWAP	Высокая	> 5% от общего объема SWAP

Сеть

Рекомендуется мониторить количество соединений по каждому состоянию:

https://en.wikipedia.org/wiki/Transmission_Control_Protocol#Protocol_operation

Тип метрики	Описание	Важность	Уведомление
CLOSE	Закрыт. Сокет не используется.	Информация	-
CLOSE_WAIT	Удаленная сторона отключилась; ожидание закрытия сокета.	Средняя	Количество соединений > 2500
CLOSING	Сокет закрыт, затем удаленная сторона отключилась; ожидание подтверждения.	Информация	-
ESTABLISHED	Соединение установлено.	Средняя	Количество соединений > 5000
FIN_WAIT1	Сокет закрыт; отключение соединения.	Информация	-
FIN_WAIT2	Сокет закрыт; ожидание отключения удаленной стороны.	Информация	-
LAST_ACK	Удаленная сторона отключилась, затем сокет закрыт; ожидание подтверждения.	Информация	-
LISTEN	Ожидает входящих соединений.	Информация	-
SYN_RECV	Идет начальная синхронизация соединения.	Информация	-
SYN_SENT	Активно пытается установить соединение.	Высокая	Количество соединений > 5000
TIME_WAIT	Сокет закрыт, но ожидает пакеты в сети для обработки.	Высокая	Количество соединений > 5000

Сбор и хранение логов приложения

Сбор логов сервиса может быть организован разными способами, в зависимости от инфраструктуры. Ниже описана типовая схема, в которой используется связка `systemd – journald – Filebeat – Logstash – OpenSearch`.

Архитектура хранения и обработки

Логи приложения хранятся в **OpenSearch** — это основное хранилище для долгосрочного хранения и индексации.

Для поиска, анализа и построения дашбордов используется **OpenSearch Dashboards**.

Сбор логов с серверов, контейнеров или подов выполняется агентом **Filebeat**.

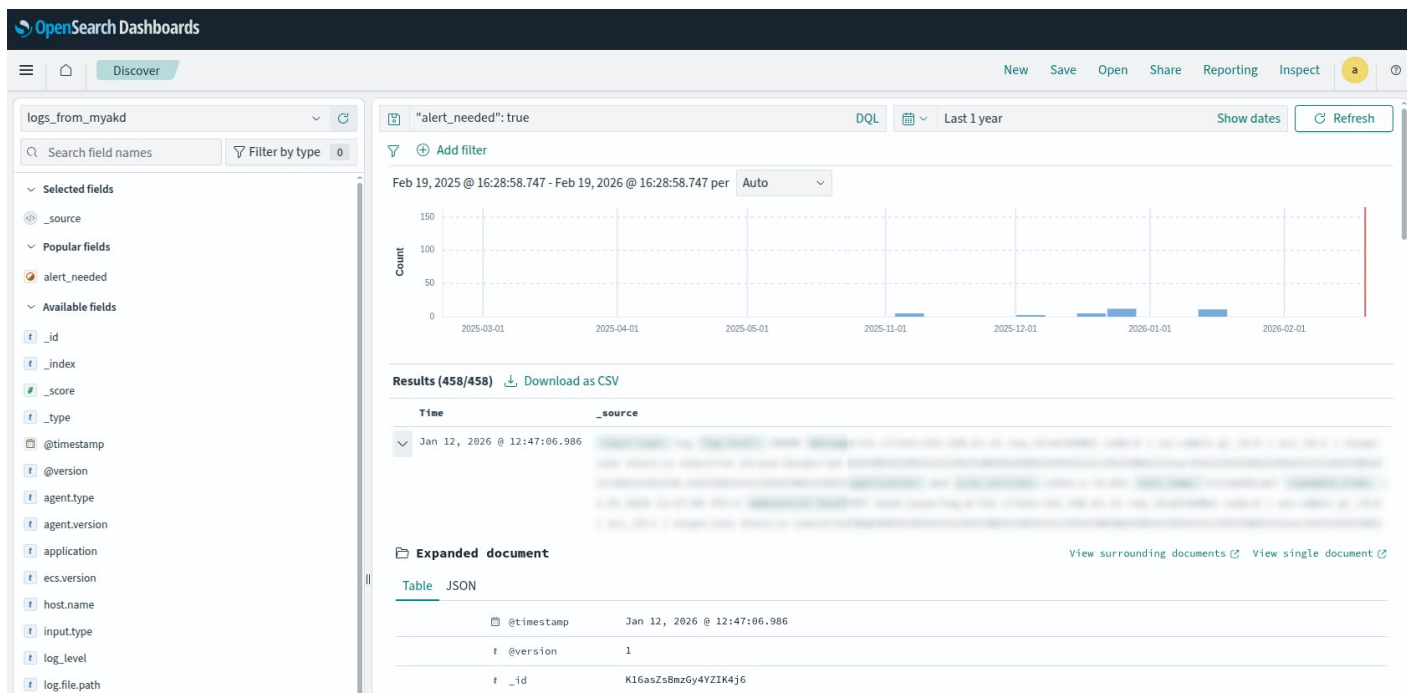
Обработка, парсинг, обогащение и нормализация логов выполняется в **Logstash**.

Типовая цепочка движения логов:

Приложение – journald – Filebeat – Logstash – OpenSearch – OpenSearch Dashboards

Все компоненты рекомендуется использовать в совместимых версиях (желательно в рамках одной ветки OpenSearch или OSS-совместимых релизов).

Итоговый результат может выглядеть так:



Логирование сервиса через systemd

Для каждого бинарного файла микросервиса скрипт-установщик создаёт unit-файл `systemd`. Пример:

```
[Unit]
Description=moodrec-offers
Wants=network-online.target
After=network-online.target

[Service]
Type=simple
ExecStart=/opt/moodrec/bin/offers
StandardOutput=journal
StandardError=journal
SyslogIdentifier=moodrec-offers
Restart=always
RestartSec=2
KillSignal=SIGTERM
KillMode=control-group
TimeoutStopSec=20

[Install]
WantedBy=multi-user.target
```

Сервис пишет логи в `journald` через `stdout` и `stderr`.

Чтение логов из `journald` через `Filebeat`

`Filebeat` может напрямую читать события из `journald` и отправлять их в `Logstash`.

Пример конфигурации:

```
filebeat.inputs:
  - type: journald
    id: moodrec-journald
    seek: cursor
    include_matches:
      - "_SYSTEMD_UNIT=moodrec-offers.service"

processors:
  - add_host_metadata: {}
  - rename:
      fields:
        - from: "systemd.unit"
          to: "service.name"
      ignore_missing: true
      fail_on_error: false

output.logstash:
  hosts: ["logstash:5044"]

logging.level: info
logging.to_files: true
logging.files:
  path: /var/log/filebeat
  name: filebeat
  keepfiles: 7
```

В этой схеме:

- Filebeat считывает записи из journald в реальном времени.
- Logstash принимает события, разбирает строки логов (grok, dissect и другие фильтры), добавляет метаданные (application, environment, pod_name и др.), нормализует формат и отправляет данные в OpenSearch.
- OpenSearch индексирует события и хранит их.
- OpenSearch Dashboards используется для поиска и визуализации логов.

Работу MoodRec обеспечивают следующие микропроцессы.

Сервисы

Сервис	Назначение
vectors	Работа с векторным хранилищем и поиск
trans	Преобразование описаний товаров в векторные представления
tasks	Выполнение отложенных заданий
offers	Управление продуктами
gateway	Проксирование HTTP (JSON) → gRPC (Proto)
events	Обработка событий
catboss	Работа с рекомендациями
auth	Авторизация
audience	Управление аудиториями

Проверка статуса всех сервисов:

```
moodrec status
```

Проверка статуса конкретного сервиса:

```
moodrec status vectors  
moodrec status trans  
moodrec status tasks  
moodrec status offers  
moodrec status gateway  
moodrec status events  
moodrec status catboss  
moodrec status auth  
moodrec status audience
```

Если сервис запущен корректно, вывод содержит `active (running)`. Если сервис остановлен — `inactive` или `failed`.

Управление конкретным сервисом выполняется отдельно для каждого микропроцесса.

Запуск:

```
moodrec start vectors
```

Остановка:

```
moodrec stop vectors
```

Перезапуск:

```
moodrec restart vectors
```

Аналогично для остальных сервисов:

```
moodrec start <service>  
moodrec stop <service>  
moodrec restart <service>
```

где `<service>` — один из: vectors, trans, tasks, offers, gateway, events, catboss, auth, audience.

После выполнения операций рекомендуется проверить состояние:

```
moodrec status <service>
```

Создание дашборда мониторинга в OpenSearch Dashboards

Мониторинг строится на данных, которые уже попадают в OpenSearch из цепочки `journald → Filebeat → Logstash`. Ниже — практический порядок действий.

1. Проверка поступления логов

В консоли сервера выполните:

```
moodrec status
```

Убедитесь, что сервисы активны.

В OpenSearch Dashboards откройте раздел Discover и проверьте, что создаются индексы (например, `moodrec-*` или `filebeat-*`) и в них появляются события.

Если индексов нет — проверьте подключение Filebeat к Logstash и Logstash к OpenSearch.

2. Создание Index Pattern (Data View)

Перейдите в: Stack Management → Data Views → Create data view

Укажите: Имя: `moodrec-logs` Index pattern: `moodrec-*` (или `filebeat-*`) Time field: `@timestamp`

После сохранения данные станут доступны для визуализаций.

3. Базовые поля для мониторинга

Для сервисов, логирующихся через systemd, обычно используются:

- `@timestamp`
- `service.name`
- `host.name`
- `log.level`
- `message`
- `application` (если добавляется в Logstash)
- `environment`

Если используется rename из `systemd.unit` → `service.name`, фильтрация будет идти по `service.name = moodrec-offers`.

4. Создание визуализаций

Перейдите в Visualize → Create visualization.

4.1 График количества логов во времени

Тип: Line chart Метрика: Count Ось X: Date histogram по `@timestamp` Интервал: auto

Этот график показывает всплески активности или ошибок.

4.2 Количество ошибок

Создайте фильтр: `log.level: error` или `message: *ERROR*`

Тип: Metric Метрика: Count

Так формируется индикатор текущего числа ошибок.

4.3 Ошибки по сервисам

Тип: Bar chart Метрика: Count Buckets → Split series → Terms → `service.name`

Добавьте фильтр по `log.level:error`.

Это покажет, какой микросервис генерирует больше ошибок.

4.4 Топ хостов

Тип: Data table Метрика: Count Buckets → Terms → `host.name`

Используется для выявления проблемных узлов.

5. Создание дашборда

Перейдите в Dashboard → Create dashboard → Add panels.

Добавьте:

- График логов во времени
- Индикатор ошибок
- Ошибки по сервисам
- Таблицу по хостам

Сохраните дашборд, например: MoodRec Monitoring

6. Настройка фильтров верхнего уровня

Добавьте глобальные фильтры:

`service.name: moodrec-offers ``environment: production` Или создайте несколько дашбордов для разных сред.

7. Практическая схема мониторинга

Операционный мониторинг:

- Резкий рост общего числа логов — возможен цикл ошибок
- Рост error/warn — деградация
- Отсутствие логов от сервиса — сервис не работает

Проверка состояния сервиса выполняется локально:

```
moodrec status
```

Если сервис активен, но логов нет — проверяется Filebeat и Logstash.

8. Расширенный мониторинг

Дополнительно можно:

Создать Alert в OpenSearch Alerting: условие — если `count(log.level:error) > N` за 5 минут.

Создать отдельный индекс для error-логов через Logstash pipeline.

Добавить парсинг времени выполнения запросов и строить latency-графики.

В итоге дашборд становится центральной точкой контроля: состояние сервисов, ошибки, нагрузка, распределение по хостам и средам.