Arenadata™ Streaming

Версия - v1.4-RUS

Настройки платформы Arenadata Streaming

2019 Arenadata™

Оглавление

1 Настройки сервисов при помощи ADCM **3**
- 1.1 Zookeeper . . . 3
- 1.2 Kafka . . . 4
- 1.3 Nifi . . . 5
- 1.4 Schema-registry . . . 7
- 1.5 Kafka REST Proxy . . . 7
- 1.6 KSQL . . . 7
- 1.7 Kafka-Manager . . . 7
- 1.8 Monitoring Clients . . . 7

2 Настройка брокера **8**

3 Настройка на уровне топика **39**

4 Конфигурирование Producer **45**

5 Конфигурирование Consumer **54**

6 Конфигурирование Connect **64**

7 Конфигурирование Streams **65**

В документации приведены настройки платформы Arenadata Streaming на уровне брокера и топика, а также конфигурирование Producer, Consumer, Connect, Streams и настройки сервисов посредством ADCM.

Инструкция может быть полезна администраторам, программистам, разработчикам и сотрудникам подразделений информационных технологий, осуществляющих внедрение и сопровождение системы.

Important: Контактная информация службы поддержки – e-mail: info@arenadata.io

Платформа **ADS** использует пары ключ-значение в формате файла свойств для конфигурации. Данные значения могут быть поставлены либо из файла, либо программно.

Глава 1

Настройки сервисов при помощи ADCM

В главе приведено описание сервисов **ADS** и их настройка при помощи **ADCM**:

- *Zookeeper*;
- *Kafka*;
- *Nifi*;
- *Schema-registry*;
- *Kafka REST Proxy*;
- *KSQL*;
- *Kafka-Manager*;
- *Monitoring Clients*.

1.1 Zookeeper

Для перехода к настройкам сервиса *Zookeeper* необходимо нажать кнопку с пиктограммой шестеренки в соответствующей строке вкладки “SERVICES” в интерфейсе **ADCM** и перейти в раздел меню “Configuration”. При этом открывается окно конфигурации сервиса *Zookeeper* (Рис.1.1.).

В блоке настроек “Main” задаются основные параметры:

- *connect* – строка подключения к Znode, в которой Zookeeper хранит конфигурацию текущего кластера, используется сервисом Kafka. В текущей реализации данный параметр недоступен для редактирования и генерируется на стороне ADCM автоматически;
- *data_dirs* – каталог для хранения транзакционных логов Zookeeper. Данный параметр указывается как *dataDir* в конфигурационном файле *zoo.cfg*;
- *client_port* – порт, на котором Zookeeper слушает клиентские подключения.

В блоке настроек “Advanced” задаются следующие расширенные параметры:

- *zoo_cfg_content* – содержимое файла *zoo.cfg*, которое в дальнейшем шаблонизируется. Данный параметр может использоваться для внесения дополнительных настроек;
- *zookeeper_env_content* – содержимое файла *zookeeper-env.sh*, которое в дальнейшем шаблонизируется. Данный параметр может использоваться для внесения переменных окружения.

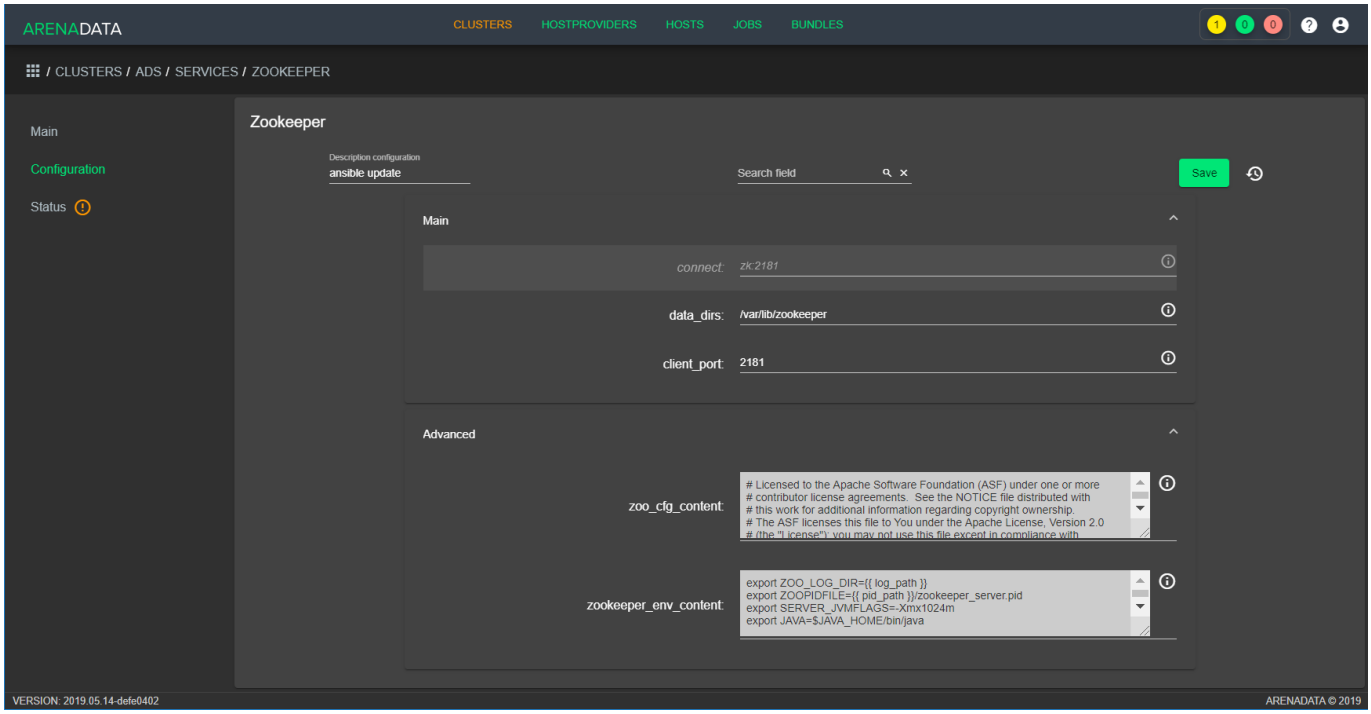


Рис.1.1.: Окно конфигурации сервиса Zookeeper

1.2 Kafka

Для перехода к настройкам сервиса *Kafka* необходимо нажать кнопку с пиктограммой шестеренки в соответствующей строке вкладки “SERVICES” и перейти в раздел меню “Configuration”. При этом открывается окно конфигурации сервиса *Kafka* (Рис.1.2.).

В блоке настроек “Main” задаются основные параметры:

- *data_dirs* – каталог для хранения данных в *Kafka*. Указывается к качестве параметра *log.dirs* в конфигурационном файле server.properties;
- *listeners* – список URI (протокол, хост и порт, на котором поднят брокер), разделенный запятыми. Если используется не *PLAINTEXT* протокол, то необходимо также указать *listener.security.protocol.map*. Для привязки ко всем интерфейсам указать имя хоста как *0.0.0.0*. Оставить имя хоста пустым для привязки к интерфейсу по умолчанию. Указывается в качестве параметра *listeners* в конфигурационном файле server.properties;
- *default_replication_factor* – фактор репликации, с которым по умолчанию создаются и хранятся топики. Указывается в качестве параметра *default.replication.factor* в конфигурационном файле server.properties;
- *delete_topic_enable* – данный параметр позволяет удалять топики. Если параметр выключен, то удаление топика через инструменты администрирования не приводит к фактическому удалению. Указывается в качестве параметра *default.replication.factor* в конфигурационном файле server.properties;
- *log_retention_hours* – количество часов, в течение которых топики хранятся в *Kafka*. Указывается в качестве параметра *log.retention.hours* в конфигурационном файле server.properties;
- *log_roll_hours* – максимальное время, после которого появляется новый журнал сегмента, даже если старый журнал не переполнен. Указывается в качестве параметра *log.roll.hours* в конфигурационном файле server.properties;

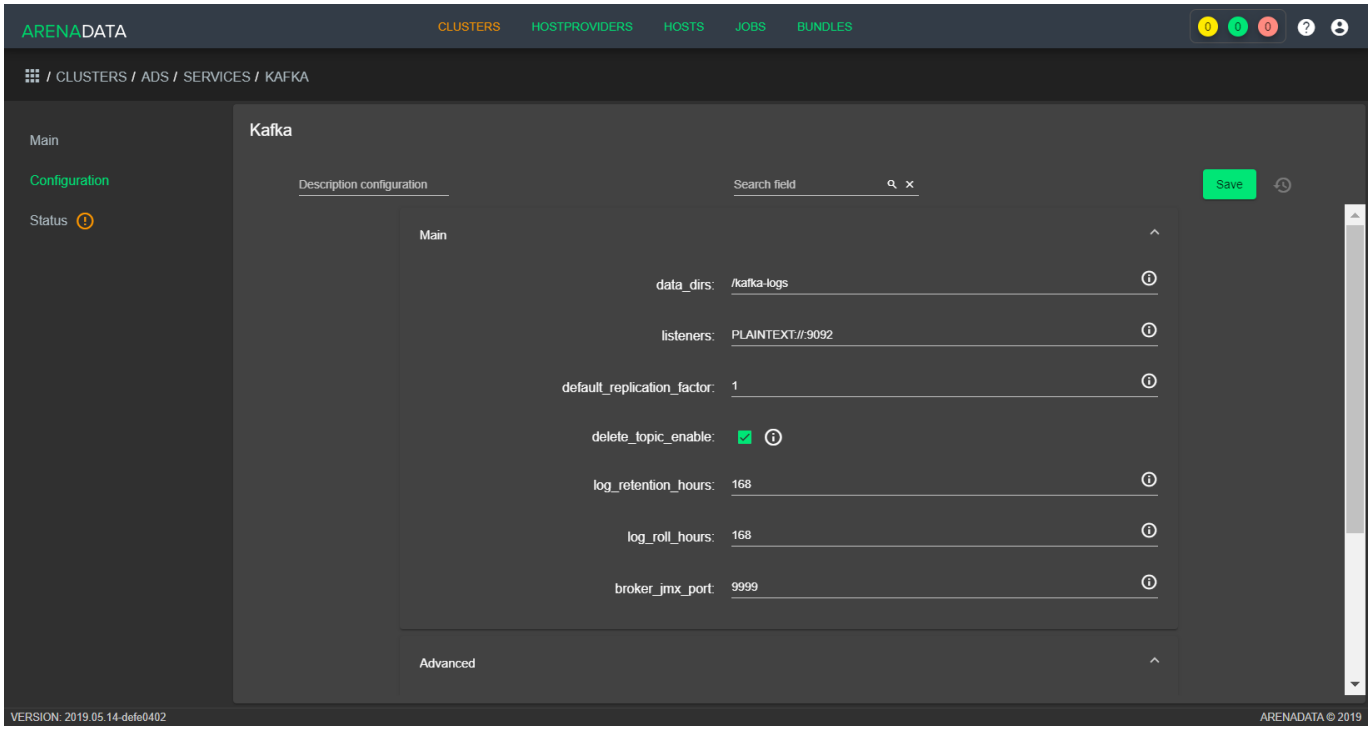


Рис.1.2.: Окно конфигурации сервиса Kafka

- *broker_jmx_port* – порт, по которому *Kafka* брокер отдает jmx-метрики. Указывается в качестве параметра *JMX_PORT* в файле *kafka-env.sh*;
- *manager_port* – порт, на котором поднимается *Kafka-Manager*. Указывается в файле *kafka-manager-env.sh*;
- *schema_registry_heap_opts* – размер кучи, выделяемой процессу *schema-registry*. Указывается в качестве параметра *SCHEMA_REGISTRY_HEAP_OPTS* в *schema-registry-env.sh*;
- *schema_registry_listener_port* – порт, который слушает *schema-registry*. Указывается в качестве параметра *listeners* в конфигурационном файле *schema-registry.properties*;

В блоке настроек “Advanced” задаются следующие расширенные параметры:

- *server_properties_content* – содержимое файла *server.properties*, которое в дальнейшем шаблонизируется. Данный параметр может использоваться для внесения дополнительных настроек;
- *kafka_env_content* – содержимое файла *kafka-env.sh*, которое в дальнейшем шаблонизируется. Данный параметр может использоваться для внесения переменных окружения.

1.3 Nifi

Для перехода к настройкам сервиса *Nifi* необходимо нажать кнопку с пиктограммой шестеренки в соответствующей строке вкладки “SERVICES” и перейти в раздел меню “Configuration”. При этом открывается окно конфигурации сервиса *Nifi* (Рис.1.3.).

В блоке настроек “Main” задаются основные параметры:

- *nifi_ui_port* – http-порт, на котором поднимается веб-интерфейс сервиса *Nifi*. Указывается в качестве параметра *nifi.web.http.port* в конфигурационном файле *nifi.properties*;

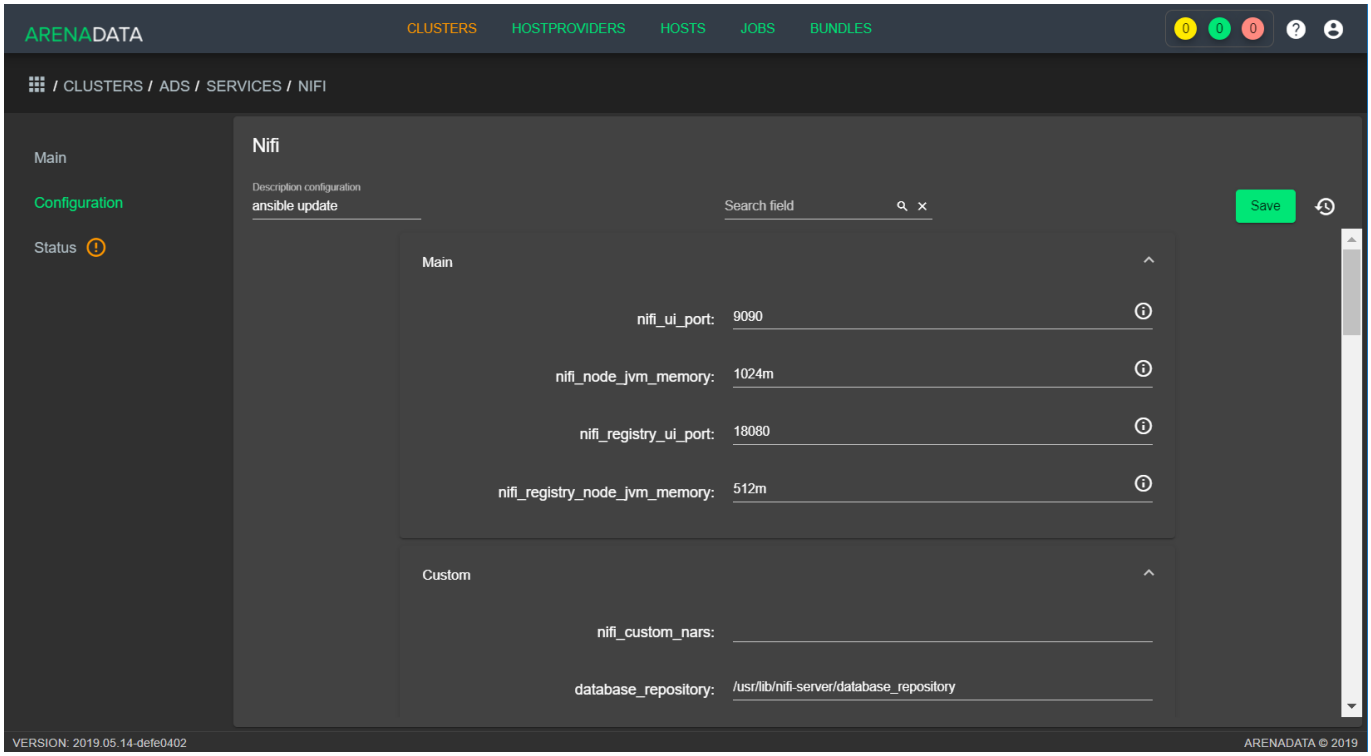


Рис.1.3.: Окно конфигурации сервиса Nifi

- *nifi_node_jvm_memory* – размер кучи, выделяемой процессу сервиса *Nifi*. Указывается в конфигурационном файле *bootstrap.conf*.

В блоке настроек “Custom” задаются следующие необязательные параметры:

- *nifi_custom_nars* – параметр следует использовать в случае добавления *custom nars*; перечисляются через запятую. Указываются в качестве *nifi.nar.library.directory.lib...* в конфигурационном файле *nifi.properties*;

В блоке настроек “Advanced” задаются следующие расширенные параметры:

- *nifi_properties_content* – содержимое файла *nifi.properties*, которое в дальнейшем шаблонизируется. Данный параметр может использоваться для внесения дополнительных настроек;
- *nifi_env_content* – содержимое файла *nifi-env.sh*, которое в дальнейшем шаблонизируется. Данный параметр может использоваться для внесения переменных окружения;
- *bootstrap_content* – содержимое файла *bootstrap.conf*, которое в дальнейшем шаблонизируется. Данное поле может использоваться для внесения настроек, связанных с запуском сервиса;
- *logback_content* – содержимое файла *logback.xml*, которое в дальнейшем шаблонизируется. Данное поле может использоваться для внесения настроек, связанных с логированием;
- *state_management_content* – содержимое файла *state_management.xml*, которое в дальнейшем шаблонизируется. Данное поле может использоваться для внесения настроек, связанных с хранением состояния сервиса *Nifi*;
- *authorizers_content* – содержимое файла *authorizers.xml*, которое в дальнейшем шаблонизируется. Данное поле может использоваться для внесения настроек авторизации в сервис *Nifi* в том случае, если настроены политики безопасности;
- *login_identity_providers_content* – содержимое файла *login_identity_providers.xml*, которое в дальнейшем шаблонизируется. Данное поле может использоваться для внесения настроек авторизации, используемые

state провайдером, в том случае, если настроены политики безопасности.

1.4 Schema-registry

1.5 Kafka REST Proxy

1.6 KSQL

1.7 Kafka-Manager

1.8 Monitoring Clients

Для перехода к настройкам сервиса *monitoring clients* необходимо нажать кнопку с пиктограммой шестеренки в соответствующей строке вкладки “SERVICES” и перейти в раздел меню “Configuration”. При этом открывается окно конфигурации сервиса *monitoring clients* (Рис.1.4.).

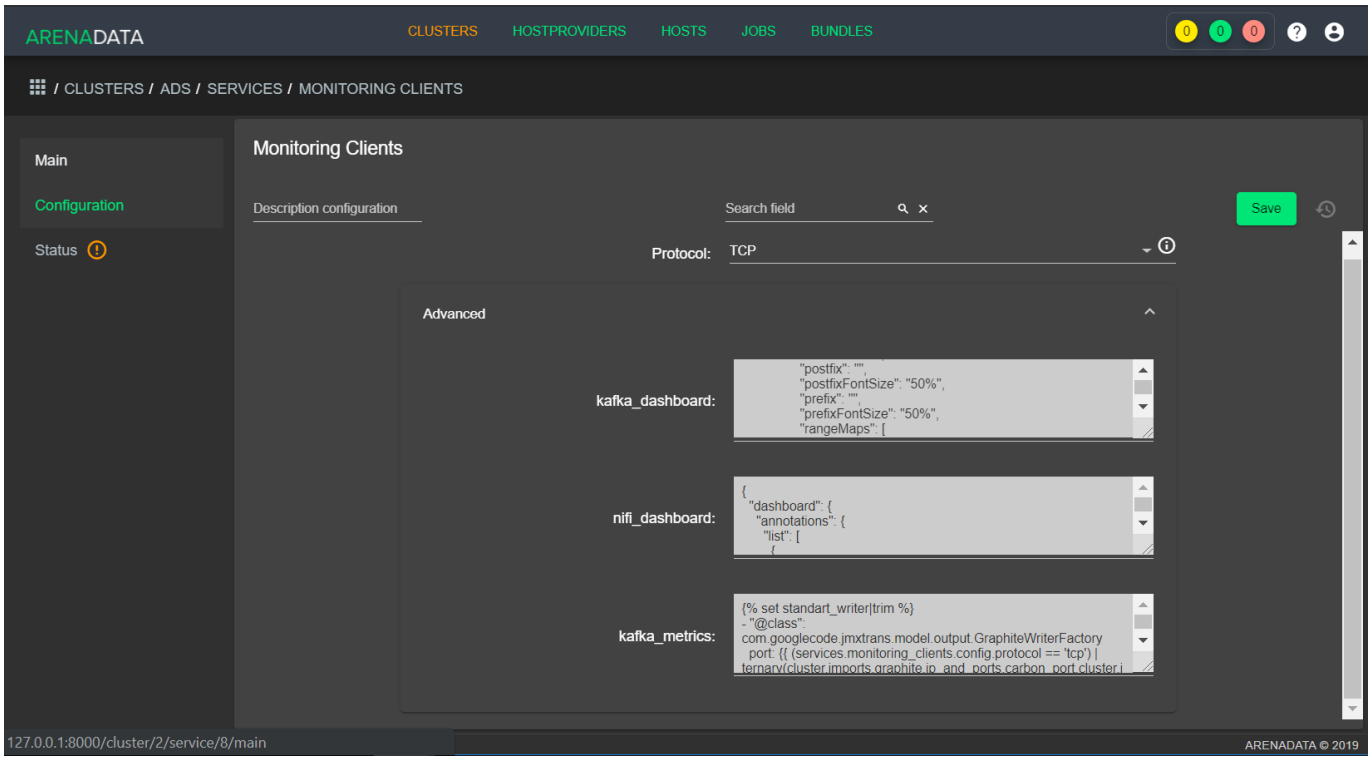


Рис.1.4.: Окно конфигурации сервиса Monitoring Clients

В блоке настроек “Advanced” задаются следующие расширенные параметры:

- *kafka_dashboard* – файл в формате *json*, который в дальнейшем шаблонизируется и отправляется в *Grafana*;
- *kafka_metrics* – файл в формате *yaml*, который в дальнейшем шаблонизируется. Включает в себя *jmx* метрики брокеров *Kafka*.

Глава 2

Настройка брокера

Все настройки брокера хранятся в конфигурационном файле */etc/kafka/conf/server.properties*.

Основные конфигурации брокера:

- *broker.id*
- *log.dirs*
- *zookeeper.connect*

Далее приведен список настроек с описанием и указанием их типа, значений по умолчанию и действительных, их важностью и режимом обновления.

zookeeper.connect – Строка хоста Zookeeper

- TYPE – string
- DEFAULT – high
- DYNAMIC UPDATE MODE – read-only

advertised.host.name – Применяется при неустановленном параметре *advertised.listeners* или *listeners*. Рекомендуется использовать *advertized.listeners*. Обозначает имя хоста для публикации в ZooKeeper для использования клиентами. В средах IaaS может отличаться от интерфейса, к которому привязывается брокер. Если параметр не задан, используется настроенное значение для *host.name*. В противном случае – значение из *java.net.InetAddress.getCanonicalHostName()*

- TYPE – string
- DEFAULT – null
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

advertised.listeners – Слушатели для публикации в ZooKeeper для использования клиентами (если есть отличие от свойства *listeners*). В средах IaaS может отличаться от интерфейса, к которому привязывается брокер. Если параметр не задан, используется значение для *listeners*. В отличие от *listeners* значение “0.0.0.0” недопустимо

- TYPE – string
- DEFAULT – null
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – per-broker

advertised.port – Применяется при неустановленном параметре *advertised.listeners* или *listeners*. Рекомендуется использовать *advertized.listeners*. Обозначает имя хоста для публикации в ZooKeeper для использования клиентами. В средах IaaS может отличаться от интерфейса, к которому привязывается брокер. Если параметр не задан, публикуется тот же порт, к которому привязан брокер

- TYPE – int
- DEFAULT – null
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

auto.create.topics.enable – Включение автоматического создания топика на сервере

- TYPE – boolean
- DEFAULT – true
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

auto.leader.rebalance.enable – Включение автоматической балансировки лидера. Балансировка лидера в фоновом режиме через регулярные промежутки времени

- TYPE – boolean
- DEFAULT – true
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

background.threads – Количество потоков для различных задач фоновой обработки

- TYPE – int
- DEFAULT – 10
- VALID VALUES – [1,...]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – cluster-wide

broker.id – Идентификатор брокера для сервера. Если значение не установлено, создается уникальный идентификатор брокера. Чтобы избежать конфликтов между id брокера, созданными с помощью zookeeper, и id брокера, настроенными пользователем, генерация идентификаторов брокера начинается с *reserved.broker.max.id + 1*

- TYPE – int
- DEFAULT – - 1
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

compression.type – Конечный тип сжатия для топика. Конфигурация принимает стандартные кодеки сжатия (“gzip”, “snappy”, “lz4”). Так же возможно “uncompressed”, что эквивалентно отсутствию сжатия; и “producer”, что означает сохранение исходного кодека сжатия, установленного поставщиком

- TYPE – string
- DEFAULT – producer
- IMPORTANCE – high

- DYNAMIC UPDATE MODE – cluster-wide

delete.topic.enable – Включение возможности удаления топика. При отключенном параметре удаление топика через администратора не имеет результата

- TYPE – boolean
- DEFAULT – true
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

host.name – Применяется только когда параметр *listeners* не установлен. Рекомендуется использовать *listeners*. Обозначает имя хоста брокера. Если параметр задан, то привязка выполняется только к данному адресу. Если параметр не задан, привязка выполняется ко всем интерфейсам

- TYPE – string
- DEFAULT – “”
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

leader.imbalance.check.interval.seconds – Частота, с которой контроллер запускает проверку балансировки партиции

- TYPE – long
- DEFAULT – 300
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

leader.imbalance.per.broker.percentage – Коэффициент дисбаланса лидера, допустимый для каждого брокера. Контроллер запускает балансировку лидера, если он превышает данное значение для брокера. Указывается в процентах

- TYPE – int
- DEFAULT – 10
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

listeners – Listener List – Разделенный запятыми список URI, которые прослушиваются, и имена слушателей сети. Если имя слушателя не является протоколом безопасности, необходимо установить *listener.security.protocol.map*. Для привязки ко всем интерфейсам указать имя хоста “0.0.0.0”. Если имя хоста не указано, привязка осуществляется к интерфейсу по умолчанию. Примеры списков слушателей сети: PLAINTEXT://myhost:9092,SSL://:9091,CLIENT://0.0.0.0:9092,REPLICATION://localhost:9093

- TYPE – string
- DEFAULT – null
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – per-broker

log.dir – Каталог хранения данных журнала (дополнительный для свойства log.dirs)

- TYPE – string
- DEFAULT – /tmp/kafka-logs

- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

log.dirs – Каталоги хранения данных журнала. Если параметр не установлен, используется значение свойства *log.dir*

- TYPE – string
- DEFAULT – null
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – high

log.flush.interval.messages – Количество накопленных в партиции журнала данных перед их сбросом на диск

- TYPE – long
- DEFAULT – 9223372036854775807
- VALID VALUES – [1,...]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – cluster-wide

log.flush.interval.ms – Максимальное время хранения данных в любом топике в памяти до их сброса на диск. Указывается в миллисекундах. Если параметр не установлен, используется значение *log.flush.scheduler.interval.ms*

- TYPE – long
- DEFAULT – null
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – cluster-wide

log.flush.offset.checkpoint.interval.ms – Частота обновления постоянной записи последнего сброса, который действует как точка восстановления журнала

- TYPE – int
- DEFAULT – 60000
- VALID VALUES – [0,...]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

log.flush.scheduler.interval.ms – Частота log flusher проверки на необходимость сброса какого-либо журнала на диск. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 9223372036854775807
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

log.flush.start.offset.checkpoint.interval.ms – Частота обновления постоянной записи смещения начала журнала

- TYPE – int

- DEFAULT – 60000
- VALID VALUES – [0,...]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

log.retention.bytes – Максимальный размер журнала перед его удалением

- TYPE – long
- DEFAULT – - 1
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – cluster-wide

log.retention.hours – Количество часов для хранения файла журнала перед его удалением, третично по отношению к свойству *log.retention.ms*. Указывается в часах

- TYPE – int
- DEFAULT – 168
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

log.retention.minutes – Количество минут для хранения файла журнала перед его удалением, вторично по отношению к свойству *log.retention.hours*. Указывается в минутах

- TYPE – int
- DEFAULT – null
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

log.retention.ms – Количество миллисекунд для хранения файла журнала перед его удалением. Указывается в миллисекундах. Если параметр не установлен, используется значение *log.retention.minutes*

- TYPE – long
- DEFAULT – null
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – cluster-wide

log.roll.hours – Максимальное время до развертывания нового сегмента журнала, вторично по отношению к свойству *log.roll.ms*. Указывается в часах

- TYPE – int
- DEFAULT – 168
- VALID VALUES – [1,...]
- IMPORTANCE – [1,...]
- DYNAMIC UPDATE MODE – read-only

log.roll.jitter.hours – Максимально допустимое значение джиттера для вычитания из *logRollTimeMillis*, вторично по отношению к свойству *log.roll.jitter.ms*. Указывается в часах

- TYPE – int
- DEFAULT – int

- VALID VALUES – [0,...]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

log.roll.jitter.ms – Максимально допустимое значение джиттера для вычитания из *logRollTimeMillis*. Указывается в миллисекундах. Если параметр не установлен, используется значение *log.roll.jitter.hours*

- TYPE – long
- DEFAULT – long
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – cluster-wide

log.roll.ms – Максимальное время до развертывания нового сегмента журнала. Указывается в миллисекундах. Если параметр не установлен, используется значение *log.roll.hours*

- TYPE – long
- DEFAULT – null
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – cluster-wide

log.segment.bytes – Максимальный размер одного файла журнала

- TYPE – int
- DEFAULT – 1073741824
- VALID VALUES – [14,...]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – cluster-wide

log.segment.delete.delay.ms – Время ожидания перед удалением файла из файловой системы

- TYPE – long
- DEFAULT – 60000
- VALID VALUES – [0,...]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – cluster-wide

message.max.bytes – Наибольший размер пакета данных, разрешенный ADS. При увеличении параметра следует также увеличить размер выборки для потребителей с целью обеспечения возможности получения пакета данных установленного размера. Параметр можно настроить для каждого топика с помощью поуровневой конфирурации топика *max.message.bytes*

- TYPE – int
- DEFAULT – 1000012
- VALID VALUES – [0,...]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – cluster-wide

min.insync.replicas – При установленном поставщиком подтверждении acks на “all” или “-1”, *min.insync.replicas* задается на минимальное количество реплик для подтверждения записи. Если этот минимум не может быть удовлетворен, то поставщик задает исключение (либо *NotEnoughReplicas*, либо *NotEnoughReplicasAfterAppend*). Совместное использование *min.insync.replicas* и acks обеспечивает более высокую гарантию к устойчивости. Типичным сценарием является создание топика с коэффициентом репликации *3*, параметром *min.insync.replicas* равным *2* и acks установленным на “all”. Это гарантирует, что поставщик задает исключение, если большинство реплик не принимает запись

- TYPE – int
- DEFAULT – 1
- VALID VALUES – [1,. . .]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – cluster-wide

num.io.threads – Число потоков, используемых сервером для обработки запросов, которые могут включать дисковые операции ввода-вывода

- TYPE – int
- DEFAULT – 8
- VALID VALUES – [1,. . .]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – cluster-wide

num.network.threads – Количество потоков, используемых сервером для получения запросов от сети и отправки ответов в сеть

- TYPE – int
- DEFAULT – 3
- VALID VALUES – [1,. . .]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – cluster-wide

num.recovery.threads.per.data.dir – Число потоков в каталоге данных, используемых для восстановления журнала при запуске или при сбросе по прекращению работы

- TYPE – int
- DEFAULT – 1
- VALID VALUES – [1,. . .]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – cluster-wide

num.replica.alter.log.dirs.threads – Число потоков, которые могут перемещать реплики между каталогами журналов, включая дисковые операции ввода-вывода

- TYPE – int
- DEFAULT – null
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

num.replica.fetchers – Количество потоков выборки, используемых для репликации данных от исходного брокера. Увеличение этого значения может увеличить степень параллелизма ввода-вывода в брокере-подписчике

- TYPE – int
- DEFAULT – 1
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – cluster-wide

offset.metadata.max.bytes – Максимальный размер для записи метаданных с учетом фиксации смещения

- TYPE – int
- DEFAULT – 4096
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

offsets.commit.required.acks – Принятие необходимых подтверждений acks перед фиксацией данных. Значение по умолчанию "-1" не следует переопределять

- TYPE – short
- DEFAULT – - 1
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

offsets.commit.timeout.ms – Фиксация смещения откладывается до тех пор, пока все реплики для топика смещения не получат коммит или данный установленный таймаут не будет достигнут. Аналогично времени ожидания запроса поставщика

- TYPE – int
- DEFAULT – 5000
- VALID VALUES – [1,. . .]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

offsets.load.buffer.size – Размер пакета для чтения из сегментов смещений при загрузке смещений в кэш

- TYPE – int
- DEFAULT – 5242880
- VALID VALUES – [1,. . .]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

offsets.retention.check.interval.ms – Частота проверки устаревших смещений

- TYPE – long
- DEFAULT – 600000
- VALID VALUES – [1,. . .]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

offsets.retention.minutes – Сброс смещений старше установленного срока хранения

- TYPE – int
- DEFAULT – 1440
- VALID VALUES – [1,. . .]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

offsets.topic.compression.codec – Кодек сжатия для топика смещения. Сжатие может использоваться для достижения “атомных” коммитов

- TYPE – int
- DEFAULT – 0
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

offsets.topic.num.partitions – Количество партиций для коммита топика смещения (не следует изменять после развертывания)

- TYPE – int
- DEFAULT – 50
- VALID VALUES – [1,. . .]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

offsets.topic.replication.factor – Коэффициент репликации для топика смещения (устанавливается выше с целью обеспечения доступности). Создание внутреннего топика невозможно, пока размер кластера не соответствует данному требованию коэффициента репликации

- TYPE – short
- DEFAULT – 3
- VALID VALUES – [1,. . .]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

offsets.topic.segment.bytes – Размер сегмента топика смещений в байтах. Значение должно быть относительно небольшим с целью ускорения сжатия журнала и загрузку кэша

- TYPE – int
- DEFAULT – 104857600
- VALID VALUES – [1,. . .]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

port – Применяется при неустановленном параметре *listeners*. Рекомендуется использовать *listeners*. Обозначает порт для прослушивания и приема подключений

- TYPE – int
- DEFAULT – 9092

- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

queued.max.requests – Количество запросов в очереди до блокировки сетевых потоков

- TYPE – int
- DEFAULT – 500
- VALID VALUES – [1,...]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

quota.consumer.default – Применяется при неустановленном параметре динамических квот по умолчанию в Zookeeper. Любой потребитель группы customerId/consumer дросселируется при получении большего количества байтов, чем данное установленное значение в секунду

- TYPE – long
- DEFAULT – 9223372036854775807
- VALID VALUES – [1,...]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

quota.producer.default – Применяется при неустановленном параметре динамических квот по умолчанию в Zookeeper. Любой поставщик с известным clientId дросселируется при получении большего количества байтов, чем данное установленное значение в секунду

- TYPE – long
- DEFAULT – 9223372036854775807
- VALID VALUES – [1,...]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

replica.fetch.min.bytes – Минимальное количество байт, ожидаемое для каждого ответа на выборку. При недостаточном объеме срабатывает параметр *replicaMaxWaitTimeMs*

- TYPE – int
- DEFAULT – 1
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

replica.fetch.wait.max.ms – Максимальное время ожидания для каждого запроса на выборку с последующей публикацией реплик. Значение всегда должно быть меньше параметра *replica.lag.time.max.ms* для предотвращения частого сжатия ISR низкопроизводительных топиков

- TYPE – int
- DEFAULT – 500
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

replica.high.watermark.checkpoint.interval.ms – Верхний предел частоты сохранения на диск (Частота сохранения высокого водяного знака на диск)

- TYPE – long
- DEFAULT – 5000
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

replica.lag.time.max.ms – Удаление подписчика лидером из isr в случае, если подписчик не отправил ни одного запроса на выборку или не считал конечное смещение журнала лидеров

- TYPE – long
- DEFAULT – 10000
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

replica.socket.receive.buffer.bytes – Буфер приема сокетов для сетевых запросов

- TYPE – int
- DEFAULT – 65536
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

replica.socket.timeout.ms – Время ожидания сокета для сетевых запросов. Значение должно быть не менее установленного параметра *replica.fetch.wait.max.ms*

- TYPE – int
- DEFAULT – 30000
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

request.timeout.ms – Максимальное время ожидания клиентом ответа на запрос. Если ответ не получен до истечения установленного значения, клиент повторно отправляет запрос при необходимости

- TYPE – int
- DEFAULT – 30000
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

socket.receive.buffer.bytes – Буфер SO_RCVBUF сокета сервера сокетов. При значении параметра "-1" используется ОС по умолчанию

- TYPE – int
- DEFAULT – 102400
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

socket.request.max.bytes – Максимальное количество байт в запросе сокета

- TYPE – int
- DEFAULT – 104857600
- VALID VALUES – [1,. . .]

- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

socket.send.buffer.bytes – Буфер SO_SNDBUF сокета сервера сокетов. При значении параметра "-1" используется ОС по умолчанию

- TYPE – int
- DEFAULT – 102400
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

transaction.max.timeout.ms – Максимально допустимое время ожидания для транзакций. Если запрошенное клиентом время транзакции превышает установленное значение, тогда брокер выдает ошибку в *InitProducerIdRequest*. Это предотвращает чрезмерное превышение времени ожидания для клиента, которое может тормозить чтение данных потребителями из топиков, включенных в транзакцию

- TYPE – int
- DEFAULT – 900000
- VALID VALUES – [1,. . .]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

transaction.state.log.load.buffer.size – Размер пакета для чтения из сегментов журнала транзакций при загрузке в кэш идентификаторов поставщиков и транзакций

- TYPE – int
- DEFAULT – 5242880
- VALID VALUES – [1,. . .]
- IMPORTANCE – [1,. . .]
- DYNAMIC UPDATE MODE – read-only

transaction.state.log.min.isr – Переопределение конфигурации *min.insync.replicas* для топика транзакции

- TYPE – int
- DEFAULT – 2
- VALID VALUES – [1,. . .]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

transaction.state.log.num.partitions – Количество партиций для топика транзакции (после развертывания параметр должен остаться неизменным)

- TYPE – int
- DEFAULT – 50
- VALID VALUES – [1,. . .]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

transaction.state.log.replication.factor – Коэффициент репликации для топика транзакции (задается выше для обеспечения доступности). Создание внутреннего топика завершится ошибкой, пока размер кластера не будет соответствовать данному требованию к фактору репликации

- TYPE – short
- DEFAULT – 3
- VALID VALUES – [1,...]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

transaction.state.log.segment.bytes – Байты сегмента топика транзакции должны быть относительно небольшими для ускорения сжатия журнала и загрузки кэша

- TYPE – int
- DEFAULT – 104857600
- VALID VALUES – [1,...]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

transactional.id.expiration.ms – Максимальное время ожидания для координатора транзакций прежде, чем предварительно истечет срок действия идентификатора транзакции поставщика без получения обновлений состояния транзакции. Указывается в миллисекундах

- TYPE – int
- DEFAULT – 604800000
- VALID VALUES – [1,...]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

unclean.leader.election.enable – Указывает, следует ли включить не входящие в набор ISR реплики и установка последнего средства в качестве лидера, даже если это может привести к потере данных

- TYPE – boolean
- DEFAULT – false
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – cluster-wide

zookeeper.connection.timeout.ms – Максимальное время ожидания клиентом установки соединения с Zookeeper. Если параметр не задан, используется значение для *zookeeper.session.timeout.ms*. Указывается в миллисекундах

- TYPE – int
- DEFAULT – null
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

zookeeper.max.in.flight.requests – Максимальное количество неподтвержденных запросов, отправленных клиентом в Zookeeper, перед блокировкой

- TYPE – int

- DEFAULT – 10
- VALID VALUES – [1,. . .]
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

zookeeper.session.timeout.ms – Тайм-аут сеанса Zookeeper. Указывается в миллисекундах

- TYPE – int
- DEFAULT – int
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

zookeeper.set.acl – Настройка клиента для использования безопасных списков управления доступом ACL

- TYPE – boolean
- DEFAULT – boolean
- IMPORTANCE – high
- DYNAMIC UPDATE MODE – read-only

broker.id.generation.enable – Автоматическое создание идентификатора брокера на сервере. При включенном параметре значение, настроенное для *reserved.broker.max.id*, должно быть пересмотрено

- TYPE – boolean
- DEFAULT – true
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

broker.rack – Стойка брокера. Используется при назначении репликации в стойке для отказоустойчивости. Примеры: “RACK1”, “us-east-1d”

- TYPE – string
- DEFAULT – string
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

connections.max.idle.ms – Время ожидания бездействующих соединений: потоки процессора сокета сервера закрывают соединения, которые простаивают больше установленного значения. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 600000
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

controlled.shutdown.enable – Включение контролируемого завершения работы сервера

- TYPE – boolean
- DEFAULT – true
- IMPORTANCE – medium

- DYNAMIC UPDATE MODE – read-only

controlled.shutdown.max.retries – Контролируемое выключение может завершиться ошибкой по нескольким причинам: параметр определяет количество повторных попыток подключения при возникновении таких сбоев

- TYPE – int
- DEFAULT – 3
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

controlled.shutdown.retry.backoff.ms – Перед каждой повторной попыткой подключения системе требуется время для восстановления состояния, вызвавшего предыдущий сбой (сбой контроллера, задержка реплики и т.д.). Параметр определяет время ожидания перед повторной попыткой. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 5000
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

controller.socket.timeout.ms – Время ожидания сокета для каналов контроллер-брокер. Указывается в миллисекундах

- TYPE – int
- DEFAULT – 30000
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

default.replication.factor – Коэффициенты репликации по умолчанию для автоматически создаваемых топиков

- TYPE – int
- DEFAULT – 1
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

delegation.token.expiry.time.ms – Время действия токена перед его обновлением. Значение по умолчанию 1 день. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 86400000
- VALID VALUES – [1,. . .]
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

delegation.token.master.key – Мастер/секретный ключ для создания и проверки делегированных токенов. Один и тот же ключ должен быть настроен для всех брокеров. Если ключ не установлен или задана пустая строка, брокеры отключают поддержку делегированных токенов

- TYPE – password

- DEFAULT – null
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

delegation.token.max.lifetime.ms – Максимальный срок действия токена, по истечении которого он больше не может быть обновлен. Значение по умолчанию 7 дней. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 604800000
- VALID VALUES – [1,...]
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

delete.records.purgatory.purge.interval.requests – Интервал очистки записей на удаление. Значение указывается в количестве запросов

- TYPE – int
- DEFAULT – 1
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

fetch.purgatory.purge.interval.requests – Интервал очистки запросов выборки. Значение указывается в количестве запросов

- TYPE – int
- DEFAULT – 1000
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

group.initial.rebalance.delay.ms – Время, в течение которого координатор группы ожидает присоединения большего числа потребителей к новой группе перед выполнением первой перебалансировки. Более длительная задержка означает потенциально меньшее количество перебалансировок, но увеличивает время до начала обработки. Указывается в миллисекундах

- TYPE – int
- DEFAULT – 3000
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

group.max.session.timeout.ms – Максимально допустимое время ожидания сессии для зарегистрированных потребителей. Более длительные тайм-ауты дают потребителям больше времени для обработки данных между heartbeat-сообщениями за счет большего времени для выявления сбоев. Указывается в миллисекундах

- TYPE – int
- DEFAULT – 300000
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

group.min.session.timeout.ms – Минимально допустимое время ожидания сессии для зарегистрированных потребителей. Более короткие тайм-ауты приводят к более быстрому обнаружению сбоев за счет более частых heartbeat-сообщений, которые могут перегружать ресурсы брокера. Указывается в миллисекундах

- TYPE – int
- DEFAULT – 6000
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

inter.broker.listener.name – Имя слушателя для связи между брокерами. Если параметр не задан, имя слушателя определяется свойством *security.inter.broker.protocol*. Одновременная установка параметров *inter.broker.listener.name* и *security.inter.broker.protocol* вызывает ошибку

- TYPE – string
- DEFAULT – null
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

inter.broker.protocol.version – Версия межброкерского протокола. Обычно параметр задается после обновления всех брокеров до новой версии. Пример некоторых допустимых значений: “0.8.0”, “0.8.1”, “0.8.1.1”, “0.8.2”, “0.8.2.0”, “0.8.2.1”, “0.9.0.0”, “0.9.0.1”. Необходимо проверить ApiVersion для полного списка

- TYPE – string
- DEFAULT – 1.1-IV0
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

log.cleaner.backoff.ms – Время спящего режима при отсутствии журналов для очистки. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 15000
- VALID VALUES – [0,...]
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – cluster-wide

log.cleaner.dedupe.buffer.size – Общая память, используемая для дедупликации журнала во всех чистых потоках

- TYPE – long
- DEFAULT – 134217728
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – cluster-wide

log.cleaner.delete.retention.ms – Длительность хранения удаленных записей. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 86400000

- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – cluster-wide

log.cleaner.enable – Включение процесса очистки журналов для запуска на сервере. Параметр должен быть включен, если используются какие-либо топики с помощью *cleanup.policy=compact*, включая топик внутренних смещений. Если параметр отключен, данные топики не сжимаются и постоянно растут в объеме

- TYPE – boolean
- DEFAULT – true
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

log.cleaner.io.buffer.load.factor – Коэффициент загрузки буфера дедуплирования журнала очистки – процент заполнения буфера дедуплирования. Более высокое значение позволит очистить больше журнала, но приведет к большему количеству хэш-конфликтов

- TYPE – double
- DEFAULT – 0.9
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – cluster-wide

log.cleaner.io.buffer.size – Общая память, используемая для ввода-вывода буферов журнала очистки через все чистые потоки

- TYPE – int
- DEFAULT – 524288
- VALID VALUES – [0,...]
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – cluster-wide

log.cleaner.io.max.bytes.per.second – Очистка журнала дросселируется таким образом, чтобы сумма операций чтения и записи была меньше установленного значения

- TYPE – double
- DEFAULT – 1.7976931348623157E308
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – cluster-wide

log.cleaner.min.cleanable.ratio – Минимальное отношение грязного журнала к общему журналу для журнала, пригодного для очистки

- TYPE – double
- DEFAULT – 0.5
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – cluster-wide

log.cleaner.min.compaction.lag.ms – Минимальное время, в течение которого сообщение остается несжатым в журнале. Применяется только для журналов с функцией сжатия. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 0

- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – cluster-wide

log.cleaner.threads – Количество фоновых потоков для очистки журнала

- TYPE – int
- DEFAULT – 1
- VALID VALUES – [0,...]
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – cluster-wide

log.cleanup.policy – Политика очистки по умолчанию для сегментов, превышающих период хранения. Допустимые политики: “delete” и “compact”

- TYPE – list
- DEFAULT – delete
- VALID VALUES – [compact, delete]
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – cluster-wide

log.index.interval.bytes – Интервал добавления записи в индекс смещения

- TYPE – int
- DEFAULT – 4096
- VALID VALUES – [0,...]
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – cluster-wide

log.index.size.max.bytes – Максимальный размер индекса смещения. Указывается в байтах

- TYPE – int
- DEFAULT – 10485760
- VALID VALUES – [4,...]
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – cluster-wide

log.message.format.version – Версия формата сообщений, которую брокер использует для добавления данных в журналы. Значение должно быть действительным ApiVersion. Некоторые примеры: “0.8.2”, “0.9.0.0”, “0.10.0”. Необходимо проверить ApiVersion для получения более подробной информации. Установив версию формата сообщений, пользователь подтверждает, что все существующие данные на диске меньше или равны указанной версии. Неправильное задание параметра приводит к тому, что потребители с более старыми версиями получают данные в нечитаемом формате

- TYPE – string
- DEFAULT – 1.1-IV0
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

log.message.timestamp.difference.max.ms – Максимальное допустимое различие между отметкой времени, когда брокер получает сообщение, и отметкой времени, указанной в сообщении. При *log.message.timestamp.type=CreateTime* сообщение отклоняется, если разница в отметке времени превышает указанный порог. Конфигурация игнорируется, если *log.message.timestamp.type=LogAppendTime*. Максимально допустимое различие временных отметок должно быть не больше, чем *log.retention.ms*. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 9223372036854775807
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – cluster-wide

log.message.timestamp.type – Определить, является ли отметка времени в сообщении временем создания сообщения или временем добавления журнала. Параметр может принимать значение “CreateTime” либо “LogAppendTime”

- TYPE – string
- DEFAULT – CreateTime
- VALID VALUES – [CreateTime, LogAppendTime]
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – cluster-wide

log.preallocate – Предварительное выделение файла при создании нового сегмента. При испольвании платформы ADS в Windows рекомендуется установить значение “true”

- TYPE – boolean
- DEFAULT – false
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – cluster-wide

log.retention.check.interval.ms – Частота проверки журналом очистки на наличие какого-либо журнала на удаление. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 300000
- VALID VALUES – [1,. . .]
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

max.connections.per.ip – Максимальное количество подключений с каждого IP-адреса

- TYPE – int
- DEFAULT – 2147483647
- VALID VALUES – [1,. . .]
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

max.connections.per.ip.overrides – Ip или hostname переопределяет максимальное количество подключений по умолчанию

- TYPE – string
- DEFAULT – “”
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

max.incremental.fetch.session.cache.slots – Максимальное количество сессий инкрементной выборки

- TYPE – int
- DEFAULT – 1000
- VALID VALUES – [0,. . .]
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

num.partitions – Число партиций по умолчанию для каждого топика

- TYPE – int
- DEFAULT – 1
- VALID VALUES – [1,. . .]
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

password.encoder.old.secret – Старый секрет для кодирования динамически настроенных паролей. Установка параметра требуется только при обновлении секрета. Если параметр задан, все динамически закодированные пароли декодируются и перекодируются с помощью *password.encoder.secret* при запуске брокера

- TYPE – password
- DEFAULT – null
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

password.encoder.secret – Секрет для кодирования динамически настроенных паролей для данного брокера

- TYPE – password
- DEFAULT – null
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

principal.builder.class – Полное имя класса, реализующего интерфейс ADSPrincipalBuilder, который используется для создания объекта ADSPrincipal во время авторизации. Конфигурация также поддерживает устаревший интерфейс PrincipalBuilder, который ранее использовался для аутентификации клиентов по протоколу SSL. Если параметр не задан, действие по умолчанию зависит от используемого протокола безопасности. Для аутентификации SSL имя принципала отличается от имени из сертификата клиента, если он предоставлен; в противном случае, если аутентификация клиента не требуется, имя принципала задается “ANONYMOUS”. Для аутентификации SASL принципал задается на основании правил, определенных в *sasl.kerberos.principal.to.local.rules* с использованием GSSAPI и идентификатора аутентификации SASL для других механизмов. Для PLAINTEXT имя принципала – “ANONYMOUS”

- TYPE – class
- DEFAULT – null

- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – per-broker

producer.purgatory.purge.interval.requests – Интервал очистки запросов поставщика. Значение указывается в количестве запросов

- TYPE – int
- DEFAULT – 1000
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

queued.max.request.bytes – Разрешенное число байтов в очереди до того, как запросы не будут прочитаны

- TYPE – long
- DEFAULT – - 1
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

replica.fetch.backoff.ms – Длительность спящего режима при возникновении ошибки партиции. Указывается в миллисекундах

- TYPE – int
- DEFAULT – 1000
- VALID VALUES – [0,. . .]
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

replica.fetch.max.bytes – Количество байтов сообщений, получаемых каждой партицией. Параметр не является абсолютным максимумом. Если первый пакет записей в первой непустой партиции выборки больше установленного значения, пакет данных все равно будет возвращен для обеспечения гарантии возможности выполнения. Максимальный размер пакета записей, принятый брокером, определяется через *message.max.bytes* (конфигурация брокера) или *max.message.bytes* (конфигурация топика)

- TYPE – int
- DEFAULT – 1048576
- VALID VALUES – [0,. . .]
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

replica.fetch.response.max.bytes – Максимальное количество байтов, ожидаемое для полного ответа на выборку. Параметр не является абсолютным максимумом. Записи извлекаются пакетами, и если первый пакет записей в первой непустой партиции выборки больше установленного значения, пакет данных все равно будет возвращен для обеспечения гарантии возможности выполнения. Максимальный размер пакета записей, принятый брокером, определяется через *message.max.bytes* (конфигурация брокера) или *max.message.bytes* (конфигурация топика)

- TYPE – int
- DEFAULT – 10485760
- VALID VALUES – [0,. . .]

- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

reserved.broker.max.id – Максимальное число, которое можно использовать для broker.id

- TYPE – int
- DEFAULT – 1000
- VALID VALUES – [0,...]
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

sasl.enabled.mechanisms – Список механизмов SASL, включенных на сервере ADS. Список может содержать любой механизм, для которого обеспечивается безопасность. По умолчанию включен только GSSAPI

- TYPE – list
- DEFAULT – GSSAPI
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – per-broker

sasl.jaas.config – Параметры контекста входа JAAS для соединений SSL в формате, используемом файлами конфигурации JAAS. Формат файла конфигурации JAAS описан по ссылке. Формат значения: "(=)*;"

- TYPE – password
- DEFAULT – null
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – per-broker

sasl.kerberos.kinit.cmd – Путь команд Kerberos kinit

- TYPE – string
- DEFAULT – /usr/bin/kinit
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – per-broker

sasl.kerberos.min.time.before.relogin – Время ожидания авторизации потока между попытками обновления

- TYPE – long
- DEFAULT – 60000
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – per-broker

sasl.kerberos.principal.to.local.rules – Список правил для сопоставления имен принципалов с короткими именами (обычно с именами пользователей операционной системы). Правила оцениваются по порядку, и первое правило, совпадающее с именем принципала, используется для сопоставления его с коротким именем. Все последующие правила в списке игнорируются. По умолчанию имена принципалов формы {username}/{hostname}@{REALM} сопоставляются с именем {username}. Важно обратить внимание, что данная конфигурация игнорируется, если расширение ADSPrincipalBuilder обеспечивается настройкой *main.builder.class*

- TYPE – list
- DEFAULT – DEFAULT

- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – per-broker

sasl.kerberos.service.name – Имя принципала Kerberos, которое запускает ADS. Значение можно определить в конфигурации ADS JAAS либо в конфигурации ADS

- TYPE – string
- DEFAULT – null
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – per-broker

sasl.kerberos.ticket.renew.jitter – Процент случайного джиттера по отношению к времени возобновления

- TYPE – double
- DEFAULT – 0.05
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – per-broker

sasl.kerberos.ticket.renew.window.factor – Время ожидания авторизации потока до тех пор, пока не будет достигнут указанный коэффициент времени от последнего обновления до истечения срока действия тикета, и попытка возобновления тикета за этот период времени

- TYPE – double
- DEFAULT – 0.8
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – per-broker

sasl.mechanism.inter.broker.protocol – Механизм SASL для взаимодействия между брокерами. По умолчанию используется GSSAPI

- TYPE – string
- DEFAULT – GSSAPI
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – per-broker

security.inter.broker.protocol – Протокол безопасности для связи между брокерами. Допустимые значения: “PLAINTEXT”, “SSL”, “SASL_PLAINTEXT”, “SASL_SSL”. Одновременная установка параметров *security.inter.broker.protocol* и *inter.broker.listener.name* вызывает ошибку

- TYPE – string
- DEFAULT – PLAINTEXT
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – read-only

ssl.cipher.suites – Список наборов шифров. Именованная комбинация аутентификации, шифрования, MAC и ключей обмена алгоритма для согласования параметров безопасности для сетевого подключения с использованием протокола TLS или SSL. По умолчанию поддерживаются все доступные варианты шифрования

- TYPE – list
- DEFAULT – “”

- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – per-broker

ssl.client.auth – Конфигурация брокера ADS для запроса аутентификации клиента. Следующие настройки являются общими:

- *ssl.client.auth=required* – требование проверки подлинности клиента;
- *ssl.client.auth=request* – аутентификация клиента является необязательной;
- *ssl.client.auth=none* – аутентификация клиента не требуется
- TYPE – string
- DEFAULT – none
- VALID VALUES – [required, requested, none]
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – per-broker

ssl.enabled.protocols – Список протоколов, включенных для соединений SSL

- TYPE – list
- DEFAULT – TLSv1.2,TLSv1.1,TLSv1
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – per-broker

ssl.key.password – Пароль закрытого ключа в файле хранилища ключей. Необязательный параметр для клиента

- TYPE – password
- DEFAULT – null
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – per-broker

ssl.keymanager.algorithm – Алгоритм службы управления ключами для SSL-соединений. Значением по умолчанию является алгоритм, настроенный для Java Virtual Machine

- TYPE – string
- DEFAULT – SunX509
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – per-broker

ssl.keystore.location – Расположение файла хранилища ключей. Необязательный параметр для клиента, может использоваться для двусторонней аутентификации клиента

- TYPE – string
- DEFAULT – null
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – per-broker

ssl.keystore.password – Пароль хранилища для файла хранения ключей. Необязательный параметр для клиента, требуется только при настройке *ssl.keystore.location*

- TYPE – password

- DEFAULT – null
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – per-broker

ssl.keystore.type – Формат файла хранилища ключей. Необязательный параметр для клиента

- TYPE – string
- DEFAULT – JKS
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – per-broker

ssl.protocol – Протокол SSL для генерации SSLContext. Значение по умолчанию – “TLS”, что подходит для большинства случаев. Допустимыми значениями в последних JVM являются “TLS”, “TLSv1.1” и “TLSv1.2”. Протоколы “SSL”, “SSLv2” и “SSLv3” могут поддерживаться в более старых JVM, но их использование не рекомендуется из-за известных уязвимостей безопасности

- TYPE – string
- DEFAULT – TLS
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – per-broker

ssl.provider – Имя поставщика безопасности для соединений SSL. Значением по умолчанию является поставщик безопасности по умолчанию для JVM

- TYPE – string
- DEFAULT – null
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – per-broker

ssl.trustmanager.algorithm – Алгоритм доверенной службы управления ключами для SSL-соединений. Значением по умолчанию является алгоритм, настроенный для Java Virtual Machine

- TYPE – string
- DEFAULT – PKIX
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – per-broker

ssl.truststore.location – Расположение файла хранилища trust store

- TYPE – string
- DEFAULT – null
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – per-broker

ssl.truststore.password – Пароль для файла хранилища trust store. При неустановленном пароле доступ к хранилищу есть, но осуществляется с отключенной проверкой надежности

- TYPE – password
- DEFAULT – null
- IMPORTANCE – medium

- DYNAMIC UPDATE MODE – per-broker

ssl.truststore.type – Формат файла хранилища trust store

- TYPE – string
- DEFAULT – JKS
- IMPORTANCE – medium
- DYNAMIC UPDATE MODE – per-broker

alter.config.policy.class.name – Класс политики изменяемых конфигураций для их валидации. Класс осуществляет интерфейс *org.apache.kafka.server.policy.AlterConfigPolicy*

- TYPE – class
- DEFAULT – null
- IMPORTANCE – low
- DYNAMIC UPDATE MODE – read-only

alter.log.dirs.replication.quota.window.num – Количество выборок для сохранения в памяти для квот репликации изменяемых журналов

- TYPE – int
- DEFAULT – 11
- VALID VALUES – [1,. . .]
- IMPORTANCE – low
- DYNAMIC UPDATE MODE – read-only

alter.log.dirs.replication.quota.window.size.seconds – Временной интервал каждой выборки для квот репликации изменяемых журналов. Указывается в секундах

- TYPE – int
- DEFAULT – 1
- VALID VALUES – [1,. . .]
- IMPORTANCE – low
- DYNAMIC UPDATE MODE – read-only

authorizer.class.name – Класс используемой авторизации

- TYPE – string
- DEFAULT – “”
- IMPORTANCE – low
- DYNAMIC UPDATE MODE – read-only

create.topic.policy.class.name – Создание класса политики топика для его валидации. Класс осуществляет интерфейс *org.apache.kafka.server.policy.CreateTopicPolicy*

- TYPE – class
- DEFAULT – null
- IMPORTANCE – low
- DYNAMIC UPDATE MODE – read-only

delegation.token.expiry.check.interval.ms – Интервал сканирования для удаления делегированных токенов с истекшим сроком действия. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 3600000
- VALID VALUES – [1,. . .]
- IMPORTANCE – low
- DYNAMIC UPDATE MODE – read-only

listener.security.protocol.map – Сопоставление имен слушателей и протоколов безопасности. Параметр должен быть определен для того, чтобы один и тот же протокол безопасности мог использоваться в нескольких портах или IP-адресах. Например, внутренний и внешний трафик могут быть разделены, даже если для обоих требуется SSL. То есть, пользователь может определить слушателей с именами “INTERNAL” и “EXTERNAL” свойством: “INTERNAL:SSL, EXTERNAL:SSL”, где ключ и значение разделяются двоеточием, а записи карты разделяются запятыми (без пробелов). Каждое имя слушателя должно отображаться на карте только один раз. Различные настройки безопасности (SSL и SASL) могут быть настроены для каждого слушателя путем добавления стандартизированного префикса (имя слушателя в нижнем регистре) к имени конфигурации. Например, чтобы установить другое хранилище ключей для внутреннего слушателя, будет установлена конфигурация с именем *listener.name.internal.ssl.keystore.location*. Если конфигурация для имени слушателя не задана, используется общая конфигурация (то есть *ssl.keystore.location*)

- TYPE – string
- DEFAULT – PLAINTEXT:PLAINTEXT,SSL:SSL,SASL_PLAINTEXT:SASL_PLAINTEXT,SASL_SSL:SASL_SSL
- IMPORTANCE – low
- DYNAMIC UPDATE MODE – per-broker

metric.reporters – Список классов для использования в качестве репортеров метрик. Реализация интерфейса *org.apache.kafka.common.metrics.MetricsReporter* позволяет подключать классы, которые будут уведомлены о создании новой метрики. JmxReporter всегда включен в реестр статистических данных JMX

- TYPE – list
- DEFAULT – “”
- IMPORTANCE – low
- DYNAMIC UPDATE MODE – cluster-wide

metrics.num.samples – Количество выборок, поддерживаемых для вычисления метрик

- TYPE – int
- DEFAULT – 2
- VALID VALUES – [1,. . .]
- IMPORTANCE – low
- DYNAMIC UPDATE MODE – read-only

metrics.recording.level – Самый высокий уровень записи для метрик

- TYPE – string
- DEFAULT – INFO
- IMPORTANCE – low
- DYNAMIC UPDATE MODE – read-only

metrics.sample.window.ms – Время ожидания вычисления метрик выборки. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 30000
- VALID VALUES – [1,. . .]
- IMPORTANCE – low
- DYNAMIC UPDATE MODE – read-only

password.encoder.cipher.algorithm – Алгоритм шифрования, используемый для кодирования динамически настроенных паролей

- TYPE – string
- DEFAULT – AES/CBC/PKCS5Padding
- IMPORTANCE – low
- DYNAMIC UPDATE MODE – read-only

password.encoder.iterations – Число итераций для кодирования динамически настроенных паролей

- TYPE – int
- DEFAULT – 4096
- VALID VALUES – [1024,. . .]
- IMPORTANCE – low
- DYNAMIC UPDATE MODE – read-only

password.encoder.key.length – Длина ключа, используемая для кодирования динамически настроенных паролей

- TYPE – int
- DEFAULT – 128
- VALID VALUES – [8,. . .]
- IMPORTANCE – low
- DYNAMIC UPDATE MODE – read-only

password.encoder.keyfactory.algorithm – Алгоритм SecretKeyFactory, используемый для кодирования динамически настроенных паролей. По умолчанию используется “PBKDF2WithHmacSHA512”, если имеется, и “PBKDF2WithHmacSHA1” в противном случае

- TYPE – string
- DEFAULT – null
- IMPORTANCE – low
- DYNAMIC UPDATE MODE – read-only

quota.window.num – Количество выборок, сохраняемых в памяти для квот клиента

- TYPE – int
- DEFAULT – 11
- VALID VALUES – [1,. . .]
- IMPORTANCE – low

- DYNAMIC UPDATE MODE – read-only

quota.window.size.seconds – Временной интервал каждой выборки для квот клиента. Указывается в секундах

- TYPE – int
- DEFAULT – 1
- VALID VALUES – [1,. . .]
- IMPORTANCE – low
- DYNAMIC UPDATE MODE – read-only

replication.quota.window.num – Количество выборок, сохраняемых в памяти для квот репликации

- TYPE – int
- DEFAULT – 11
- VALID VALUES – [1,. . .]
- IMPORTANCE – low
- DYNAMIC UPDATE MODE – read-only

replication.quota.window.size.seconds – Временной интервал каждой выборки для квот репликации. Указывается в секундах

- TYPE – int
- DEFAULT – 1
- VALID VALUES – [1,. . .]
- IMPORTANCE – low
- DYNAMIC UPDATE MODE – read-only

ssl.endpoint.identification.algorithm – Алгоритм идентификации конечных точек для валидации имени хоста сервера с использованием сертификата сервера

- TYPE – string
- DEFAULT – null
- IMPORTANCE – low
- DYNAMIC UPDATE MODE – per-broker

ssl.secure.random.implementation – Реализация SecureRandom PRNG, используемая для операций шифрования SSL

- TYPE – string
- DEFAULT – null
- IMPORTANCE – low
- DYNAMIC UPDATE MODE – per-broker

transaction.abort.timed.out.transaction.cleanup.interval.ms – Интервал, в течение которого выполняются отложенные транзакции. Указывается в миллисекундах

- TYPE – int
- DEFAULT – 60000
- VALID VALUES – [1,. . .]

- IMPORTANCE – low
- DYNAMIC UPDATE MODE – read-only

transaction.remove.expired.transaction.cleanup.interval.ms – Интервал удаления транзакций, срок действия которых истекает по установленному параметру *transactional.id.expiration.ms passing.* Указывается в миллисекундах

- TYPE – int
- DEFAULT – 3600000
- VALID VALUES – [1,. . .]
- IMPORTANCE – low
- DYNAMIC UPDATE MODE – read-only

zookeeper.sync.time.ms – Удаленность последователя Zookeeper от лидера Zookeeper. Указывается в миллисекундах

- TYPE – int
- DEFAULT – 2000
- IMPORTANCE – low
- DYNAMIC UPDATE MODE – read-only

Более подробную информацию о конфигурации брокера можно найти в классе scala *kafka.server.KafkaConfig.*

Глава 3

Настройка на уровне топика

Настройки топиков имеют как сервер по умолчанию, так и опциональное переопределение каждого топика. Если топику не задана конфигурация, то используется сервер по умолчанию. Переопределение можно установить во время создания топика, указав один или несколько параметров *–config*.

Далее приведены конфигурации на уровне топика. Настройка сервера по умолчанию указана в Server Default Property. Заданное значение по умолчанию для сервера относится только к топику, если у него нет явного переопределения конфигурации.

cleanup.policy – Определение политики хранения старых сегментов журнала. Политика по умолчанию “delete” отбрасывает старые сегменты при достижении их срока хранения или предельного размера. Значение “compact” включает сжатие журнала по топику

- TYPE – list
- DEFAULT – delete
- VALID VALUES – [compact, delete]
- SERVER DEFAULT PROPERTY –log.cleanup.policy
- IMPORTANCE – medium

compression.type – Окончательный тип сжатия топика. Конфигурация принимает стандартные кодеки сжатия (“gzip”, “snappy”, “lz4”). Также принимает “uncompressed”, что эквивалентно отсутствию сжатия; и “producer”, что означает сохранение исходного кодека сжатия, установленного поставщиком

- TYPE – string
- DEFAULT – producer
- VALID VALUES – [uncompressed, snappy, lz4, gzip, producer]
- SERVER DEFAULT PROPERTY – compression.type
- IMPORTANCE – medium

delete.retention.ms – Время хранения маркированных на удаление данных с целью сжатия топиков журнала. Параметр также дает ограничение на время, в течение которого потребитель должен выполнить чтение, если данные начинаются со смещения *0*, для гарантии валидности снапшота заключительного этапа (в противном случае удаление маркированных данных может произойти до завершения их сканирования). Указывается в миллисекундах

- TYPE – long
- DEFAULT – 86400000

- VALID VALUES – [0,...]
- SERVER DEFAULT PROPERTY – log.cleaner.delete.retention.ms
- IMPORTANCE – medium

file.delete.delay.ms – Время ожидания перед удалением файла из файловой системы. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 60000
- VALID VALUES – [0,...]
- SERVER DEFAULT PROPERTY – log.segment.delete.delay.ms
- IMPORTANCE – medium

flush.messages – Интервал принудительной синхронизации данных, записанных в журнал. Например, если параметр установлен на *1*, синхронизация выполняется после каждого сообщения; если на значение *5* – после каждых пяти сообщений. Установка данного параметра не рекомендуется, эффективней использовать репликацию для обеспечения устойчивости и возможности фоновой очистки операционной системы. Параметр можно переопределить в базовых настройках каждого топика

- TYPE – long
- DEFAULT – 9223372036854775807
- VALID VALUES – [0,...]
- SERVER DEFAULT PROPERTY – log.flush.interval.messages
- IMPORTANCE – medium

flush.ms – Временной интервал принудительной синхронизации данных, записанных в журнал. Например, если параметр установлен на *1000*, синхронизация выполняется по истечении 1000 мс. Установка данного параметра не рекомендуется, эффективней использовать репликацию для обеспечения устойчивости и возможности фоновой очистки операционной системы. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 9223372036854775807
- VALID VALUES – [0,...]
- SERVER DEFAULT PROPERTY – log.flush.interval.ms
- IMPORTANCE – medium

follower.replication.throttled.replicas – Список реплик, для которых репликация журнала должна дросселироваться на стороне подписчика. Список должен описывать набор реплик в формате "[PartitionId]:[BrokerId],[PartitionId]:[BrokerId]:..." или можно использовать специальный символ "*" для дросселирования всех реплик в данном топике

- TYPE – list
- DEFAULT – ""
- VALID VALUES – [partitionId],[brokerId]:[partitionId],[brokerId]:...
- SERVER DEFAULT PROPERTY – follower.replication.throttled.replicas
- IMPORTANCE – medium

index.interval.bytes – Частота добавления индексной записи в индекс смещения. Значение по умолчанию гарантирует индексацию сообщения примерно каждые 4096 байт. Большее индексирование позволяет потребителям приближаться к более точному положению в журнале, но увеличивает сам индекс. Рекомендуется значение не менять

- TYPE – int
- DEFAULT – 4096
- VALID VALUES – [0,...]
- SERVER DEFAULT PROPERTY – log.index.interval.bytes
- IMPORTANCE – medium

leader.replication.throttled.replicas – Список реплик, для которых репликация журнала должна дросселироваться на стороне лидера. Список должен описывать набор реплик в формате "[PartitionId]:[BrokerId],[PartitionId]:[BrokerId]:..." или можно использовать специальный символ "*" для дросселирования всех реплик в данном топике

- TYPE – list
- DEFAULT – ""
- VALID VALUES – [partitionId],[brokerId]:[partitionId],[brokerId]:...
- SERVER DEFAULT PROPERTY – leader.replication.throttled.replicas
- IMPORTANCE – medium

max.message.bytes – Наибольший размер пакета данных, разрешенный ADS. При увеличении параметра следует также увеличить размер выборки для потребителей с целью обеспечения возможности получения пакета данных установленного размера

- TYPE – int
- DEFAULT – 1000012
- VALID VALUES – [0,...]
- SERVER DEFAULT PROPERTY – message.max.bytes
- IMPORTANCE – medium

message.format.version – Версия формата сообщений, которую брокер использует для добавления данных в журналы. Значение должно быть действительным ApiVersion. Некоторые примеры: "0.8.2", "0.9.0.0", "0.10.0". Необходимо проверить ApiVersion для получения более подробной информации. Установив версию формата сообщений, пользователь подтверждает, что все существующие данные на диске меньше или равны указанной версии. Неправильное задание параметра приводит к тому, что потребители с более старыми версиями получают данные в нечитаемом формате

- TYPE – string
- DEFAULT – 1.1-IV0
- SERVER DEFAULT PROPERTY – log.message.format.version
- IMPORTANCE – medium

message.timestamp.difference.max.ms – Максимальное допустимое различие между отметкой времени, когда брокер получает сообщение, и отметкой времени, указанной в сообщении. При *message.timestamp.type=CreateTime* сообщение отклоняется, если разница в отметке времени превышает указанный порог. Конфигурация игнорируется, если *message.timestamp.type=LogAppendTime*. Указывается в миллисекундах

- TYPE – long

- DEFAULT – 9223372036854775807
- VALID VALUES – [0,. . .]
- SERVER DEFAULT PROPERTY – log.message.timestamp.difference.max.ms
- IMPORTANCE – medium

message.timestamp.type – Определить, является ли отметка времени в сообщении временем создания сообщения или временем добавления журнала. Параметр может принимать значение “CreateTime” либо “LogAppendTime”

- TYPE – string
- DEFAULT – CreateTime
- SERVER DEFAULT PROPERTY – log.message.timestamp.type
- IMPORTANCE – medium

min.cleanable.dirty.ratio – Частота очистки журнала (при условии включенного сжатия). По умолчанию избегается очистка, где сжато более 50% журнала. Это ограничивает максимальное пространство, выделенное в журнале на дубликаты (не более 50% журнала могут занимать дубликаты). Более высокое отношение означает меньшее количество дубликатов и более эффективную очистку, но при этом большее количество потерянного пространства в журнале

- TYPE – double
- DEFAULT – 0.5
- VALID VALUES – [0,. . . ,1]
- SERVER DEFAULT PROPERTY – log.cleaner.min.cleanable.ratio
- IMPORTANCE – medium

min.compaction.lag.ms – Минимальное время, в течение которого сообщение остается несжатым в журнале. Применяется только для журналов с функцией сжатия. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 0
- VALID VALUES – [0,. . .]
- SERVER DEFAULT PROPERTY – log.cleaner.min.compaction.lag.ms
- IMPORTANCE – medium

min.insync.replicas – При установленном поставщиком подтверждении acks на “all” или “-1”, *min.insync.replicas* задается на минимальное количество реплик для подтверждения записи. Если этот минимум не может быть удовлетворен, то поставщик задает исключение (*NotEnoughReplicas* или *NotEnoughReplicasAfterAppend*). Совместное использование *min.insync.replicas* и acks обеспечивает более высокую гарантию к устойчивости. Типичным сценарием является создание топика с коэффициентом репликации *3*, параметром *min.insync.replicas* равным *2* и acks установленным на “all”. Это гарантирует, что поставщик задает исключение, если большинство реплик не принимает запись

- TYPE – int
- DEFAULT – 1
- VALID VALUES – [1,. . .]
- SERVER DEFAULT PROPERTY – min.insync.replicas
- IMPORTANCE – medium

preallocate – Предварительное выделение файла на диске при создании нового сегмента журнала

- TYPE – boolean
- DEFAULT – false
- SERVER DEFAULT PROPERTY – log.preallocate
- IMPORTANCE – medium

retention.bytes – Контроль максимального размера партиции (состоящей из сегментов журнала), который может увеличиваться до момента отказа от старых сегментов журнала с целью освобождения места при использовании политики хранения “delete”. По умолчанию ограничения по размеру нет, есть только ограничение по времени. Поскольку данный предел применяется на уровне партиции, необходимо умножить значение лимита по времени на количество партиций, чтобы вычислить объем хранения топика в байтах

- TYPE – long
- DEFAULT – - 1
- SERVER DEFAULT PROPERTY – log.retention.bytes
- IMPORTANCE – medium

retention.ms – Контроль максимального времени, в течение которого хранится журнал, прежде чем отбрасываются старые сегменты журнала с целью освобождения места при использовании политики хранения “delete”. Параметр представляет собой SLA о том, как скоро потребители должны читать свои данные. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 604800000
- SERVER DEFAULT PROPERTY – log.retention.ms
- IMPORTANCE – medium

segment.bytes – Контроль размера файла сегмента для журнала. Сохранение и очистка файла всегда выполняются единовременно, поэтому больший размер сегмента означает меньшее количество файлов, но при этом менее гранулированный контроль над хранением

- TYPE – int
- DEFAULT – 1073741824
- VALID VALUES – [14,. . .]
- SERVER DEFAULT PROPERTY – log.segment.bytes
- IMPORTANCE – medium

segment.index.bytes – Контроль размера индекса, который отображает смещения в позициях файла. Предварительно индексный файл выделяется и сокращается только после сжатия журнала. Обычно параметр не требует изменений

- TYPE – int
- DEFAULT – 10485760
- VALID VALUES – [0,. . .]
- SERVER DEFAULT PROPERTY – log.index.size.max.bytes
- IMPORTANCE – medium

segment.jitter.ms – Максимальный рандомный джиттер. Вычитается из запланированного времени сжатия сегмента во избежание проблемы сегментации thundering herd (огромное количество процессов, ждущих события, в то время как требуется только один процесс). Указывается в миллисекундах

- TYPE – long
- DEFAULT – 0
- VALID VALUES – [0,...]
- SERVER DEFAULT PROPERTY – log.roll.jitter.ms
- IMPORTANCE – medium

segment.ms – Период времени, после которого ADS выполняет сжатие журнала, даже если файл сегмента не заполнен, с целью обеспечения сохранения или сжатия устаревших данных. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 604800000
- VALID VALUES – [0,...]
- SERVER DEFAULT PROPERTY – log.roll.ms
- IMPORTANCE – medium

unclean.leader.election.enable – Указывает, следует ли включить не входящие в набор ISR реплики и установка последнего средства в качестве лидера, даже если это может привести к потере данных

- TYPE – boolean
- DEFAULT – false
- SERVER DEFAULT PROPERTY – unclean.leader.election.enable
- IMPORTANCE – medium

Глава 4

Конфигурирование Producer

Далее приведены конфигурации Java-поставщика.

key.serializer – Класс Serializer для ключа, реализующего интерфейс *org.apache.kafka.common.serialization.Serializer*

- TYPE – class
- IMPORTANCE – high

value.serializer – Класс Serializer для значения, реализующего интерфейс *org.apache.kafka.common.serialization.Serializer*

- TYPE – class
- IMPORTANCE – high

acks – Количество подтверждений, которые поставщик требует от лидера перед рассмотрением запроса. Параметр контролирует устойчивость отправляемых записей. Возможны следующие настройки:

- *acks=0* – поставщик не ждет подтверждения с сервера. Запись немедленно добавляется в буфер сокета и считается отправленной. Данная настройка не гарантирует получение сервером записи, и конфигурация повторных попыток не вступает в силу (так как клиент обычно не знает о каких-либо сбоях). Смещение, возвращаемое для каждой записи, всегда равно "-1".
- *acks=1* – лидер фиксирует запись в свой локальный журнал, но отвечает, не дожидаясь полного подтверждения от всех подписчиков. В этом случае лидер может выйти из строя сразу после подтверждения записи и до того, как подписчики реплицируют ее, тогда запись теряется
- *acks=all* – лидер ожидает полного набора синхронизированных реплик для подтверждения записи. Данная настройка гарантирует, что запись не будет потеряна, пока хотя бы одна синхронизированная реплика остается в строе. Это наивысшая гарантия. Эквивалентно настройке *acks=-1*
- TYPE – string
- DEFAULT – 1
- VALID VALUES – [all, - 1, 0, 1]
- IMPORTANCE – high

bootstrap.servers – Список пар хост/порт, используемых для установления первоначального подключения к платформе ADS. В дальнейшем клиент будет использовать все сервера, независимо от того, какие указаны в данном параметре – этот список влияет только на начальные хосты, используемые для обнаружения полного набора серверов. Параметр должен быть задан в формате "host1:port1, host2:port2,..." (через запятую и без пробелов). Поскольку данные сервера используются только для первоначального подключения с целью

обнаружения полного набора в кластере (который может динамически меняться), списку необязательно содержать полный набор серверов (можно указать более одного, на случай отказа первого)

- TYPE – list
- DEFAULT – “”
- VALID VALUES – org.apache.kafka.common.config.ConfigDef$NonNullValidator@685cb137
- IMPORTANCE – high

buffer.memory – Общий объем памяти в байтах, которую поставщик может использовать для буферизации записей, ожидающих отправки на сервер. Если записи отправляются быстрее, чем они могут быть доставлены на сервер, поставщик блокирует параметр *max.block.ms*, после чего будет сделано исключение. Параметр должен соответствовать примерно общему объему памяти, которая используется поставщиком, но не полному объему, так как не вся память используется для буферизации. Некоторый дополнительный объем используется для сжатия (если оно включено), а также для поддержания запросов на лету

- TYPE – long
- DEFAULT – 33554432
- VALID VALUES – [0,. . .]
- IMPORTANCE – high

compression.type – Тип сжатия для всех данных, созданных поставщиком. По умолчанию используется значение “none” (без сжатия). Допустимые значения: “none”, “gzip”, “snappy” и “lz4”. Сжатие выполняется над полной партией данных, поэтому эффективность дозирования влияет на коэффициент сжатия (более многочисленное порционирование означает лучшее сжатие)

- TYPE – string
- DEFAULT – none
- IMPORTANCE – high

retries – Установка значения больше нуля приводит к тому, что клиент переотправляет любую запись, передача которой завершается с временной ошибкой. Повторная попытка ничем не отличается от повторной отправки записи клиентом при получении ошибки. Повторная отправка данных без установки параметра *max.in.flight.requests.per.connection* в значение “1” потенциально может изменить порядок записей, так как если две партии данных отправляются в одну партицию, при этом первая партия не выполняется и повторно отправляется, а вторая выполняется успешно, то данные второго пакета появляются в партиции первыми

- TYPE – int
- DEFAULT – 0
- VALID VALUES – [0,. . . ,2147483647]
- IMPORTANCE – high

ssl.key.password – Пароль закрытого ключа в файле хранилища ключей. Необязательный параметр для клиента

- TYPE – password
- DEFAULT – null
- IMPORTANCE – high

ssl.keystore.location – Расположение файла хранилища ключей. Необязательный параметр для клиента, может использоваться для двусторонней аутентификации клиента

- TYPE – string

- DEFAULT – null
- IMPORTANCE – high

ssl.keystore.password – Пароль хранилища для файла хранения ключей. Необязательный параметр для клиента, требуется только при настройке *ssl.keystore.location*

- TYPE – password
- DEFAULT – null
- IMPORTANCE – high

ssl.truststore.location – Расположение файла хранилища trust store

- TYPE – string
- DEFAULT – null
- IMPORTANCE – high

ssl.truststore.password – Пароль для файла хранилища trust store. При неустановленном пароле доступ к хранилищу есть, но осуществляется с отключенной проверкой надежности

- TYPE – password
- DEFAULT – null
- IMPORTANCE – high

batch.size – При отправке нескольких записей в одну и ту же партицию поставщик пытается объединить их. Это помогает производительности как на клиенте, так и на сервере. Конфигурация управляет размером пакета в байтах. Пакетирование большего размера, чем задан в параметре, не осуществляется. В таком случае отправленные брокерам запросы содержат несколько пакетов (по одному для каждой партиции) с доступными для отправки данными. Небольшой размер пакета делает его менее востребованным и может снизить пропускную способность (нулевой размер пакета полностью отключает пакетирование). Очень большой размер пакета может использовать память расточительно, так как всегда выделяется буфер указанного размера пакета в ожидании дополнительных записей

- TYPE – int
- DEFAULT – 16384
- VALID VALUES – [0,...]
- IMPORTANCE – medium

client.id – Строка id для передачи на сервер при выполнении запросов. Целью является возможность отслеживания источника запросов за пределами ip/port, позволяя включать логическое имя приложения в журнал запросов на стороне сервера

- TYPE – string
- DEFAULT – “”
- IMPORTANCE – medium

connections.max.idle.ms – Закрытие бездействующих соединений по истечению заданного периода. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 540000
- IMPORTANCE – medium

linger.ms – Поставщик объединяет в один пакет все записи, поступающие между трансмиссиями запросов. Обычно это происходит, когда данные поступают быстрее, чем могут быть отправлены. Однако клиент может уменьшить количество запросов даже при умеренной загрузке. Это реализуется путем добавления небольшого промежутка времени искусственной задержки, то есть вместо немедленной отправки данных поставщик ждет до указанной отметки с целью пакетирования данных. Это можно рассматривать как аналог алгоритма Nagle в TCP. Параметр дает верхнюю границу задержки по времени для пакетной обработки. Но как только достигается установленный размер пакета данных *batch.size* для партиции, пакет немедленно отправляется (независимо от заданного параметра *linger.ms*). Однако, имея меньший объем байт пакета, чем в указаном параметре *batch.size*, осуществляется задержка в течение времени, заданного *linger.ms*, с целью ожидания появления новых данных. По умолчанию параметр *linger.ms* равен "0" (то есть без задержки). Например, установка "linger.ms=5" приведет к уменьшению количества отправленных запросов, но добавит до *5 мс* задержки для данных, отправленных при отсутствии нагрузки. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 0
- VALID VALUES – [0,...]
- IMPORTANCE – medium

max.block.ms – Время блокировки *ADSProducer.send()* и *ADSProducer.partitionsFor()*. Данные методы могут быть заблокированы либо по причине заполненного буфера, либо из-за недоступности метаданных. Блокировка в предоставленных пользователем сериализаторах или разделителе не учитывается по таймауту данного параметра. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 60000
- VALID VALUES – [0,...]
- IMPORTANCE – medium

max.request.size – Максимальный размер запроса в байтах. Параметр ограничивает количество пакетов данных, которые поставщик отправляет в одном запросе во избежание отправки огромных запросов. Параметр также эффективно ограничивает максимальный размер пакета данных. При этом сервер имеет свой собственный предел размера пакета данных, который может отличаться от указанного

- TYPE – int
- DEFAULT – 1048576
- VALID VALUES – [0,...]
- IMPORTANCE – medium

partitioner.class – Класс Partitioner, реализующий интерфейс *org.apache.kafka.clients.producer.Partitioner*

- TYPE – class
- DEFAULT – org.apache.kafka.clients.producer.internals.DefaultPartitioner
- IMPORTANCE – medium

receive.buffer.bytes – Размер буфера приема TCP (SO_RCVBUF) при чтении данных. Если значение равно "-1", используется ОС по умолчанию

- TYPE – int
- DEFAULT – 32768
- VALID VALUES – [-1,...]
- IMPORTANCE – medium

request.timeout.ms – Максимальное время ожидания клиентом ответа на запрос. Если ответ не получен до истечения установленного значения, клиент повторно отправляет запрос при необходимости. Значение параметра должно быть больше, чем *replica.lag.time.max.ms* (конфигурация брокера), с целью сокращения возможного дублирования данных по причине излишних попыток поставщика. Указывается в миллисекундах

- TYPE – int
- DEFAULT – 30000
- VALID VALUES – [0,. . .]
- IMPORTANCE – medium

sasl.jaas.config – Параметры контекста входа JAAS для соединений SSL в формате, используемом файлами конфигурации JAAS. Формат файла конфигурации JAAS описан по ссылке. Формат значения: “(=)*;”

- TYPE – password
- DEFAULT – null
- IMPORTANCE – medium

sasl.kerberos.service.name – Имя принципала Kerberos, которое запускает ADS. Значение можно определить в конфигурации ADS JAAS либо в конфигурации ADS

- TYPE – string
- DEFAULT – null
- IMPORTANCE – medium

sasl.mechanism – Механизм SASL для клиентских подключений. Может быть любой механизм, для которого обеспечивается безопасность. По умолчанию используется GSSAPI

- TYPE – string
- DEFAULT – GSSAPI
- IMPORTANCE – medium

security.protocol – Протокол безопасности для связи между брокерами. Допустимые значения: “PLAINTEXT”, “SSL”, “SASL_PLAINTEXT”, “SASL_SSL”

- TYPE – string
- DEFAULT – PLAINTEXT
- IMPORTANCE – medium

send.buffer.bytes – Размер буфера отправки TCP (SO_SNDBUF) при отправке данных. Если значение равно “-1”, используется ОС по умолчанию

- TYPE – int
- DEFAULT – 131072
- VALID VALUES – [-1,. . .]
- IMPORTANCE – medium

ssl.enabled.protocols – Список протоколов, включенных для соединений SSL

- TYPE – list
- DEFAULT – TLSv1.2,TLSv1.1,TLSv1
- IMPORTANCE – medium

ssl.keystore.type – Формат файла хранилища ключей. Необязательный параметр для клиента

- TYPE – string
- DEFAULT – JKS
- IMPORTANCE – medium

ssl.protocol – Протокол SSL для генерации SSLContext. Значение по умолчанию – “TLS”, что подходит для большинства случаев. Допустимыми значениями в последних JVM являются “TLS”, “TLSv1.1” и “TLSv1.2”. Протоколы “SSL”, “SSLv2” и “SSLv3” могут поддерживаться в более старых JVM, но их использование не рекомендуется из-за известных уязвимостей безопасности

- TYPE – string
- DEFAULT – TLS
- IMPORTANCE – medium

ssl.provider – Имя поставщика безопасности для соединений SSL. Значением по умолчанию является поставщик безопасности по умолчанию для JVM

- TYPE – string
- DEFAULT – null
- IMPORTANCE – medium

ssl.truststore.type – Формат файла хранилища trust store

- TYPE – string
- DEFAULT – JKS
- IMPORTANCE – medium

enable.idempotence – При установленном значении “true” поставщик гарантирует, что ровно одна копия каждого сообщения записывается в поток. При значении “false” в поток могут быть записаны дубликаты сообщений при повторных попытках отправки данных поставщиком из-за сбоев брокера или по другим причинам. Данный параметр требует, чтобы свойство *max.in.flight.requests.per.connection* было меньше или равно “5”, повторные попытки более “0”, и acks установлены на “all”. Если перечисленные настройки явно не заданы пользователем, выбираются подходящие значения. При установке несовместимых значений, выдается ConfigException

- TYPE – boolean
- DEFAULT – false
- IMPORTANCE – low

interceptor.classes – Список классов для использования в качестве интерсепторов. Реализация интерфейса *org.apache.kafka.clients.producer.ProducerInterceptor* позволяет перехватывать (и, возможно, видоизменять) записи, полученные поставщиком до их публикации в кластере ADS. По умолчанию интерсепторы не установлены

- TYPE – list
- DEFAULT – “”
- VALID VALUES – org.apache.kafka.common.config.ConfigDef$NonNullValidator@6a41eaa2
- IMPORTANCE – low

max.in.flight.requests.per.connection – Максимальное количество неподтвержденных запросов, отправляемых клиентом по одному соединению перед блокировкой. Если параметр имеет значение больше *1*, то в случае сбоев существует риск переупорядочения данных из-за повторных попыток (если они включены)

- TYPE – int

- DEFAULT – 5
- VALID VALUES – [1,...]
- IMPORTANCE – low

metadata.max.age.ms – Период времени, после которого принудительно обновляются метаданные даже при отсутствии видимых изменений в лидере партиции с целью предварительного обнаружения новых брокеров или партиций. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 300000
- VALID VALUES – [0,...]
- IMPORTANCE – low

metric.reporters – Список классов для использования в качестве репортеров метрик. Реализация интерфейса *org.apache.kafka.common.metrics.MetricsReporter* позволяет подключать классы, которые будут уведомлены о создании новой метрики. JmxReporter всегда включен в реестр статистических данных JMX

- TYPE – list
- DEFAULT – “”
- VALID VALUES – org.apache.kafka.common.config.ConfigDef$NonNullValidator@7cd62f43
- IMPORTANCE – low

metrics.num.samples – Количество выборок, поддерживаемых для вычисления метрик

- TYPE – int
- DEFAULT – 2
- VALID VALUES – [1,...]
- IMPORTANCE – low

metrics.recording.level – Самый высокий уровень записи для метрик

- TYPE – string
- DEFAULT – INFO
- VALID VALUES – [INFO, DEBUG]
- IMPORTANCE – low

metrics.sample.window.ms – Время ожидания вычисления метрик выборки. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 30000
- VALID VALUES – [0,...]
- IMPORTANCE – low

reconnect.backoff.max.ms – Максимальный период времени ожидания повторного подключения к брокеру при неоднократных сбоях соединения. Отсрочка на хост увеличивается экспоненциально для каждого последующего сбоя соединения, вплоть до установленного максимума. После расчета увеличения отсрочки к значению добавляется *20%* случайного джиттера во избежание помех связи. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 1000

- VALID VALUES – [0,...]
- IMPORTANCE – low

reconnect.backoff.ms – Базовый период времени ожидания повторного подключения к хосту. Позволяет избегать многократного подключения к узлу в узком цикле. Данная отсрочка применяется ко всем попыткам подключения клиента к брокеру. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 50
- VALID VALUES – [0,...]
- IMPORTANCE – low

retry.backoff.ms – Время ожидания перед повторной попыткой отправки неудавшегося запроса в партицию топика. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 100
- VALID VALUES – [0,...]
- IMPORTANCE – low

sasl.kerberos.kinit.cmd – Путь команд Kerberos kinit

- TYPE – string
- DEFAULT – /usr/bin/kinit
- IMPORTANCE – low

sasl.kerberos.min.time.before.relogin – Время ожидания авторизации потока между попытками обновления

- TYPE – long
- DEFAULT – 60000
- IMPORTANCE – low

sasl.kerberos.ticket.renew.jitter – Процент случайного джиттера по отношению к времени возобновления

- TYPE – double
- DEFAULT – 0.05
- IMPORTANCE – low

sasl.kerberos.ticket.renew.window.factor – Время ожидания авторизации потока до тех пор, пока не будет достигнут указанный коэффициент времени от последнего обновления до истечения срока действия тикета, и попытка возобновления тикета за этот период времени

- TYPE – double
- DEFAULT – 0.8
- IMPORTANCE – low

ssl.cipher.suites – Список наборов шифров. Именованная комбинация аутентификации, шифрования, MAC и ключей обмена алгоритма для согласования параметров безопасности для сетевого подключения с использованием протокола TLS или SSL. По умолчанию поддерживаются все доступные варианты шифрования

- TYPE – list

- DEFAULT – null
- IMPORTANCE – low

ssl.endpoint.identification.algorithm – Алгоритм идентификации конечных точек для валидации имени хоста сервера с использованием сертификата сервера

- TYPE – string
- DEFAULT – null
- IMPORTANCE – low

ssl.keymanager.algorithm – Алгоритм службы управления ключами для SSL-соединений. Значением по умолчанию является алгоритм, настроенный для Java Virtual Machine

- TYPE – string
- DEFAULT – SunX509
- IMPORTANCE – low

ssl.secure.random.implementation – Реализация SecureRandom PRNG, используемая для операций шифрования SSL

- TYPE – string
- DEFAULT – null
- IMPORTANCE – low

ssl.trustmanager.algorithm – Алгоритм доверенной службы управления ключами для SSL-соединений. Значением по умолчанию является алгоритм, настроенный для Java Virtual Machine

- TYPE – string
- DEFAULT – PKIX
- IMPORTANCE – low

transaction.timeout.ms – Максимальный интервал времени, который координатор транзакции ожидает для обновления статуса транзакции от поставщика перед тем, как будет прервана текущая транзакция. Если установленное значение больше, чем значение *transaction.max.timeout.ms* в настройках брокера, запрос завершается ошибкой *InvalidTransactionTimeout*

- TYPE – int
- DEFAULT – 60000
- IMPORTANCE – low

transactional.id – Идентификатор транзакции. Параметр позволяет использовать семантику достоверности, которая охватывает несколько сессий поставщика, и позволяет гарантировать клиенту, что транзакции, использующие тот же TransactionalId, завершены до начала любых новых транзакций. Если TransactionalId не указан, то поставщик ограничивается идемпотентной доставкой. Важно, что параметр *enable.idempotence* должен быть включен при сконфигурированном *TransactionalId*. Значение по умолчанию “null”, что означает невозможность использования транзакций. Для транзакций требуется по меньшей мере три брокера по умолчанию, что является рекомендуемым параметром для продуктивной системы; для разработки можно изменить настройки в параметре брокера *transaction.state.log.replication.factor*

- TYPE – string
- DEFAULT – null
- VALID VALUES – non-empty string
- IMPORTANCE – low

Глава 5

Конфигурирование Consumer

Далее приведены конфигурации для нового потребителя.

key.deserializer – Класс десериализатора для ключа, реализующего интерфейс *org.apache.kafka.common.serialization.Deserializer*

- TYPE – class
- IMPORTANCE – high

value.deserializer – Класс десериализатора для значения, реализующего интерфейс *org.apache.kafka.common.serialization.Deserializer*

- TYPE – class
- IMPORTANCE – high

bootstrap.servers – Список пар хост/порт, используемых для установления первоначального подключения к платформе ADS. В дальнейшем клиент будет использовать все сервера, независимо от того, какие указаны в данном параметре – этот список влияет только на начальные хосты, используемые для обнаружения полного набора серверов. Параметр должен быть задан в формате “host1:port1, host2:port2,...” (через запятую и без пробелов). Поскольку данные сервера используются только для первоначального подключения с целью обнаружения полного набора в кластере (который может динамически меняться), списку необязательно содержать полный набор серверов (можно указать более одного, на случай отказа первого)

- TYPE – list
- DEFAULT – “”
- VALID VALUES – org.apache.kafka.common.config.ConfigDef$NonNullValidator@7cd62f43
- IMPORTANCE – high

fetch.min.bytes – Минимальный объем данных, которые сервер должен вернуть по запросу на выборку. При недостаточном объеме данных запрос ожидает их накопления до установленного значения. Значение по умолчанию *1 байт* означает, что запросы на выборку отвечают, как только доступен *1 байт* данных, или по истечению времени ожидания запроса. Установка большего значения заставляет сервер ожидать больших объемов данных для накопления, что может немного повысить пропускную способность сервера за счет некоторой дополнительной задержки

- TYPE – int
- DEFAULT – 1
- VALID VALUES – [0,...]
- IMPORTANCE – high

group.id – Уникальная строка, идентифицирующая группу потребителей. Свойство требуется, если потребитель использует функциональность группового управления с помощью подписки (топика) или стратегии управления смещением на основе ADS

- TYPE – string
- DEFAULT – “”
- IMPORTANCE – high

heartbeat.interval.ms – Время ожидания для координатора потребителя между heartbeat-сообщениями при использовании средств группового управления ADS. Heartbeat-сообщения используются для обеспечения активности сессии потребителя и перебалансировки, когда пользователи присоединяются или покидают группу. Значение должно быть установлено ниже, чем параметр *session.timeout.ms* (обычно оно не более *1/3* от этого значения). Его можно настроить еще ниже с целью контроля ожидаемого времени для нормальных перебалансировок. Указывается в миллисекундах

- TYPE – int
- DEFAULT – 3000
- IMPORTANCE – high

max.partition.fetch.bytes – Максимальный объем данных для каждой партиции, который будет возвращен серверу. Из пакетов записи извлекаются потребителями. Если первый пакет записей в первой непустой партиции выборки больше установленного значения, пакет данных все равно будет возвращен для обеспечения гарантии возможности выполнения. Максимальный размер пакета записей, принятый брокером, определяется через *message.max.bytes* (конфигурация брокера) или *max.message.bytes* (конфигурация топика). Для ограничения размера запроса потребителя используется параметр *fetch.max.bytes*

- TYPE – int
- DEFAULT – 1048576
- VALID VALUES – [0,. . .]
- IMPORTANCE – high

session.timeout.ms – Время ожидания для выявления сбоев потребителей при использовании группового управления ADS. Потребитель посылает периодические heartbeat-сообщения брокеру с целью подтверждения своей активности. Если брокер не получает ни одного бита до истечения установленного времени сеанса, то удаляет данного потребителя из группы и начинает балансировку. Значение параметра должно быть в допустимом диапазоне конфигурации брокера *group.min.session.timeout.ms* и *group.max.session.timeout.ms*. Указывается в миллисекундах

- TYPE – int
- DEFAULT – 10000
- IMPORTANCE – high

ssl.key.password – Пароль закрытого ключа в файле хранилища ключей. Необязательный параметр для клиента

- TYPE – password
- DEFAULT – null
- IMPORTANCE – high

ssl.keystore.location – Расположение файла хранилища ключей. Необязательный параметр для клиента, может использоваться для двусторонней аутентификации клиента

- TYPE – string

- DEFAULT – null
- IMPORTANCE – high

ssl.keystore.password – Пароль хранилища для файла хранения ключей. Необязательный параметр для клиента, требуется только при настройке *ssl.keystore.location*

- TYPE – password
- DEFAULT – null
- IMPORTANCE – high

ssl.truststore.location – Расположение файла хранилища trust store

- TYPE – string
- DEFAULT – null
- IMPORTANCE – high

ssl.truststore.password – Пароль для файла хранилища trust store. При неустановленном пароле доступ к хранилищу есть, но осуществляется с отключенной проверкой надежности

- TYPE – password
- DEFAULT – null
- IMPORTANCE – null

auto.offset.reset – В случае если в ADS нет начального смещения или текущее смещение больше не существует на сервере (например, так как данные удалены):

- *earliest*: автоматически сбросить смещение до самого раннего смещения;
- *latest*: автоматически сбросить смещение до последнего смещения;
- *none*: исключение для потребителя, если предыдущее смещение для группы потребителей не найдено;
- *anything else*: исключение для потребителя.
- TYPE – string
- DEFAULT – latest
- VALID VALUES – [latest, earliest, none]
- IMPORTANCE – medium

connections.max.idle.ms – Закрытие бездействующих соединений по истечению заданного периода. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 540000
- IMPORTANCE – medium

enable.auto.commit – При значении “true” смещение потребителя периодически фиксируется в фоновом режиме

- TYPE – boolean
- DEFAULT – true
- IMPORTANCE – medium

exclude.internal.topics – Предоставление потребителю записей из внутренних топиков (например, смещения). При значении “true” единственным способом получения записей из внутреннего топика является подписка на него

- TYPE – boolean
- DEFAULT – true
- IMPORTANCE – medium

fetch.max.bytes – Максимальный объем данных, который сервер должен вернуть для запроса на выборку. Параметр не является абсолютным максимумом. Из пакетов записи извлекаются потребителями. Если первый пакет записей в первой непустой партиции выборки больше установленного значения, пакет данных все равно будет возвращен для обеспечения гарантии возможности выполнения. Максимальный размер пакета записей, принятый брокером, определяется через *message.max.bytes* (конфигурация брокера) или *max.message.bytes* (конфигурация топика). При этом потребитель параллельно выполняет несколько выборок

- TYPE – int
- DEFAULT – 52428800
- VALID VALUES – [0,. . .]
- IMPORTANCE – medium

isolation.level – Контроль транзакционно записанных данных. Если установлено значение “read_committed”, *consumer.poll()* возвращает только совершенные транзакционные сообщения. При значении “read_uncommitted” (по умолчанию), *consumer.poll()* возвращает все сообщения, даже прерванные. Нетранзакционные сообщения возвращаются в любом режиме. Данные всегда возвращаются в порядке смещения. Следовательно, в режиме “read_committed” *consumer.poll()* возвращает сообщения только до последнего стабильного смещения (LSO), которое меньше, чем смещение первой открытой транзакции. То есть любые данные, появляющиеся после текущей транзакциии, удерживаются до завершения соответствующей транзакции. Потребители “read_committed” не могут считывать высокий водяной знак в процессе транзакции. Так же в режиме “read_committed” метод *seekToEnd* возвращает LSO

- TYPE – string
- DEFAULT – read_uncommitted
- VALID VALUES – [read_committed, read_uncommitted]
- IMPORTANCE – medium

max.poll.interval.ms – Максимальная задержка времени между вызовами *poll()* при управлении группами потребителей. Параметр устанавливает верхнюю границу времени, в течение которого потребитель может бездействовать. Если *poll()* не вызывается до истечения установленного таймаута, потребитель считается неисправным, и группа перебалансируется с целью переназначения партиций. Указывается в миллисекундах

- TYPE – int
- DEFAULT – 300000
- VALID VALUES – [1,. . .]
- IMPORTANCE – medium

max.poll.records – Максимальное число записей, возвращаемых за один вызов *poll()*

- TYPE – int
- DEFAULT – 500
- VALID VALUES – [1,. . .]
- IMPORTANCE – medium

partition.assignment.strategy – Класс стратегии назначения партиций, которую клиент использует для распределения принадлежности партиции инстансам потребителя при групповом управлении

- TYPE – list
- DEFAULT – class org.apache.kafka.clients.consumer.RangeAssignor
- VALID VALUES – org.apache.kafka.common.config.ConfigDef$NonNullValidator@6d4b1c02
- IMPORTANCE – medium

receive.buffer.bytes – Размер буфера приема TCP (SO_RCVBUF) при чтении данных. Если значение равно "-1", используется ОС по умолчанию

- TYPE – int
- DEFAULT – 65536
- VALID VALUES – [-1,...]
- IMPORTANCE – medium

request.timeout.ms – Максимальное время ожидания клиентом ответа на запрос. Если ответ не получен до истечения установленного значения, клиент повторно отправляет запрос при необходимости. Указывается в миллисекундах

- TYPE – int
- DEFAULT – 305000
- VALID VALUES – [0,...]
- IMPORTANCE – medium

sasl.jaas.config – Параметры контекста входа JAAS для соединений SSL в формате, используемом файлами конфигурации JAAS. Формат файла конфигурации JAAS описан по ссылке. Формат значения: "(=)*;"

- TYPE – password
- DEFAULT – null
- IMPORTANCE – medium

sasl.kerberos.service.name – Имя принципала Kerberos, которое запускает ADS. Значение можно определить в конфигурации ADS JAAS либо в конфигурации ADS

- TYPE – string
- DEFAULT – null
- IMPORTANCE – medium

sasl.mechanism – Механизм SASL для клиентских подключений. Может быть любой механизм, для которого обеспечивается безопасность. По умолчанию используется GSSAPI

- TYPE – string
- DEFAULT – GSSAPI
- IMPORTANCE – medium

security.protocol – Протокол безопасности для связи между брокерами. Допустимые значения: "PLAINTEXT", "SSL", "SASL_PLAINTEXT", "SASL_SSL"

- TYPE – string
- DEFAULT – PLAINTEXT
- IMPORTANCE – medium

send.buffer.bytes – Буфер SO_SNDBUF сокета сервера сокетов. При значении параметра "-1" используется ОС по умолчанию

- TYPE – int
- DEFAULT – 131072
- VALID VALUES – [-1,...]
- IMPORTANCE – medium

ssl.enabled.protocols – Список протоколов, включенных для соединений SSL

- TYPE – list
- DEFAULT – TLSv1.2,TLSv1.1,TLSv1
- IMPORTANCE – medium

ssl.keystore.type – Формат файла хранилища ключей. Необязательный параметр для клиента

- TYPE – string
- DEFAULT – JKS
- IMPORTANCE – medium

ssl.protocol – Протокол SSL для генерации SSLContext. Значение по умолчанию – "TLS", что подходит для большинства случаев. Допустимыми значениями в последних JVM являются "TLS", "TLSv1.1" и "TLSv1.2". Протоколы "SSL", "SSLv2" и "SSLv3" могут поддерживаться в более старых JVM, но их использование не рекомендуется из-за известных уязвимостей безопасности

- TYPE – string
- DEFAULT – TLS
- IMPORTANCE – medium

ssl.provider – Имя поставщика безопасности для соединений SSL. Значением по умолчанию является поставщик безопасности по умолчанию для JVM

- TYPE – string
- DEFAULT – null
- IMPORTANCE – medium

ssl.truststore.type – Формат файла хранилища trust store

- TYPE – string
- DEFAULT – JKS
- IMPORTANCE – medium

auto.commit.interval.ms – Частота автофиксации потребительских смещений при включенном параметре *enable.auto.commit* (значение "true"). Указывается в миллисекундах

- TYPE – int
- DEFAULT – 5000
- VALID VALUES – [0,...]
- IMPORTANCE – low

check.crcs – Автоматическая проверка CRC32 считываемых записей. Проверка добавляет некоторые накладные расходы, поэтому она может быть отключена в случаях, требующих высокой производительности

- TYPE – boolean
- DEFAULT – true
- IMPORTANCE – low

client.id – Строка id для передачи на сервер при выполнении запросов. Целью является возможность отслеживания источника запросов за пределами ip/port, позволяя включать логическое имя приложения в журнал запросов на стороне сервера

- TYPE – string
- DEFAULT – “”
- IMPORTANCE – low

fetch.max.wait.ms – Максимальный период времени, в течение которого сервер блокируется, прежде чем ответить на запрос выборки (в случае недостаточного объема данных для незамедлительного ответа, заданного функцией *fetch.min.bytes*). Указывается в миллисекундах

- TYPE – int
- DEFAULT – 500
- VALID VALUES – [0,...]
- IMPORTANCE – low

interceptor.classes – Список классов для использования в качестве интерсепторов. Реализация интерфейса *org.apache.kafka.clients.consumer.ConsumerInterceptor* позволяет перехватывать (и, возможно, видоизменять) полученные потребителем записи. По умолчанию интерсепторы не установлены

- TYPE – list
- DEFAULT – “”
- VALID VALUES – org.apache.kafka.common.config.ConfigDef$NonNullValidator@6093dd95
- IMPORTANCE – low

metadata.max.age.ms – Период времени, после которого принудительно обновляются метаданные даже при отсутствии видимых изменений в лидере партиции с целью предварительного обнаружения новых брокеров или партиций. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 300000
- VALID VALUES – [0,...]
- IMPORTANCE – low

metric.reporters – Список классов для использования в качестве репортеров метрик. Реализация интерфейса *org.apache.kafka.common.metrics.MetricsReporter* позволяет подключать классы, которые будут уведомлены о создании новой метрики. JmxReporter всегда включен в реестр статистических данных JMX

- TYPE – list
- DEFAULT – “”
- VALID VALUES – org.apache.kafka.common.config.ConfigDef$NonNullValidator@5622fdf
- IMPORTANCE – low

metrics.num.samples – Количество выборок, поддерживаемых для вычисления метрик

- TYPE – int

- DEFAULT – 2
- VALID VALUES – [1,...]
- IMPORTANCE – low

metrics.recording.level – Самый высокий уровень записи для метрик

- TYPE – string
- DEFAULT – INFO
- VALID VALUES – [INFO, DEBUG]
- IMPORTANCE – low

metrics.sample.window.ms – Время ожидания вычисления метрик выборки. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 30000
- VALID VALUES – [0,...]
- IMPORTANCE – low

reconnect.backoff.max.ms – Максимальный период времени ожидания повторного подключения к брокеру при неоднократных сбоях соединения. Отсрочка на хост увеличивается экспоненциально для каждого последующего сбоя соединения, вплоть до установленного максимума. После расчета увеличения отсрочки к значению добавляется *20%* случайного джиттера во избежание помех связи. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 1000
- VALID VALUES – [0,...]
- IMPORTANCE – low

reconnect.backoff.ms – Базовый период времени ожидания повторного подключения к хосту. Позволяет избегать многократного подключения к узлу в узком цикле. Данная отсрочка применяется ко всем попыткам подключения клиента к брокеру. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 50
- VALID VALUES – [0,...]
- IMPORTANCE – low

retry.backoff.ms – Время ожидания перед повторной попыткой отправки неудавшегося запроса в партицию топика. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 100
- VALID VALUES – [0,...]
- IMPORTANCE – low

sasl.kerberos.kinit.cmd – Путь команд Kerberos kinit

- TYPE – string
- DEFAULT – /usr/bin/kinit

- IMPORTANCE – low

sasl.kerberos.min.time.before.relogin – Время ожидания авторизации потока между попытками обновления

- TYPE – long
- DEFAULT – 60000
- IMPORTANCE – low

sasl.kerberos.ticket.renew.jitter – Процент случайного джиттера по отношению к времени возобновления

- TYPE – double
- DEFAULT – 0.05
- IMPORTANCE – low

sasl.kerberos.ticket.renew.window.factor – Время ожидания авторизации потока до тех пор, пока не будет достигнут указанный коэффициент времени от последнего обновления до истечения срока действия тикета, и попытка возобновления тикета за этот период времени

- TYPE – double
- DEFAULT – 0.8
- IMPORTANCE – low

ssl.cipher.suites – Список наборов шифров. Именованная комбинация аутентификации, шифрования, MAC и ключей обмена алгоритма для согласования параметров безопасности для сетевого подключения с использованием протокола TLS или SSL. По умолчанию поддерживаются все доступные варианты шифрования

- TYPE – list
- DEFAULT – null
- IMPORTANCE – low

ssl.endpoint.identification.algorithm – Алгоритм идентификации конечных точек для валидации имени хоста сервера с использованием сертификата сервера

- TYPE – string
- DEFAULT – null
- IMPORTANCE – low

ssl.keymanager.algorithm – Алгоритм службы управления ключами для SSL-соединений. Значением по умолчанию является алгоритм, настроенный для Java Virtual Machine

- TYPE – string
- DEFAULT – SunX509
- IMPORTANCE – low

ssl.secure.random.implementation – Реализация SecureRandom PRNG, используемая для операций шифрования SSL

- TYPE – string
- DEFAULT – null
- IMPORTANCE – low

ssl.trustmanager.algorithm – Алгоритм доверенной службы управления ключами для SSL-соединений. Значением по умолчанию является алгоритм, настроенный для Java Virtual Machine

- TYPE – string
- DEFAULT – PKIX
- IMPORTANCE – low

Глава 6

Конфигурирование Connect

Глава 7

Конфигурирование Streams

Далее приведены конфигурации для клиентской библиотеки ADS Streams.

application.id – Идентификатор приложения потоковой обработки. Должен быть уникальным в пределах платформы ADS. Используется как: 1) префикс идентификатора клиента по умолчанию, 2) идентификатор группы для управления членством, 3) префикс топика изменений в журнале

- TYPE – string
- IMPORTANCE – high

bootstrap.servers – Список пар хост/порт, используемых для установления первоначального подключения к платформе ADS. В дальнейшем клиент будет использовать все сервера, независимо от того, какие указаны в данном параметре – этот список влияет только на начальные хосты, используемые для обнаружения полного набора серверов. Параметр должен быть задан в формате "host1:port1, host2:port2,..." (через запятую и без пробелов). Поскольку данные сервера используются только для первоначального подключения с целью обнаружения полного набора в кластере (который может динамически меняться), списку необязательно содержать полный набор серверов (можно указать более одного, на случай отказа первого)

- TYPE – list
- IMPORTANCE – high

replication.factor – Коэффициент репликации для топиков журнала изменений и репартицирования топиков, созданных приложением потоковой обработки

- TYPE – int
- DEFAULT – 1
- IMPORTANCE – high

state.dir – Местоположение каталога хранилища state store

- TYPE – string
- DEFAULT – /tmp/kafka-streams
- IMPORTANCE – high

cache.max.bytes.buffering – Максимальный объем памяти в байтах, используемый для буферизации всех потоков

- TYPE – long
- DEFAULT – 10485760
- VALID VALUES – [0,...]

- IMPORTANCE – medium

client.id – Строка префикса ID, используемая для идентификаторов клиента внутреннего потребителя и поставщика (восстановления потребителя); шаблон "-StreamThread–"

- TYPE – string
- DEFAULT – ""
- IMPORTANCE – medium

default.deserialization.exception.handler – Класс обработки исключений, реализующий интерфейс *org.apache.kafka.streams.errors.DeserializationExceptionHandler*

- TYPE – class
- DEFAULT – org.apache.kafka.streams.errors.LogAndFailExceptionHandler
- IMPORTANCE – medium

default.key.serde – Класс сериализатора/десериализатора по умолчанию для ключа, реализующего интерфейс *org.apache.kafka.common.serialization.Serde*

- TYPE – class
- DEFAULT – org.apache.kafka.common.serialization.Serdes$ByteArraySerde
- IMPORTANCE – medium

default.production.exception.handler – Класс обработки исключений, реализующий интерфейс *org.apache.kafka.streams.errors.ProductionExceptionHandler*

- TYPE – class
- DEFAULT – org.apache.kafka.streams.errors.DefaultProductionExceptionHandler
- IMPORTANCE – medium

default.timestamp.extractor – Класс выделения временных меток по умолчанию, реализующий интерфейс *org.apache.kafka.streams.processor.TimestampExtractor*

- TYPE – class
- DEFAULT – org.apache.kafka.streams.processor.FailOnInvalidTimestamp
- IMPORTANCE – medium

default.value.serde – Класс сериализатора/десериализатора по умолчанию для значения, реализующего интерфейс *org.apache.kafka.common.serialization.Serde*

- TYPE – class
- DEFAULT – org.apache.kafka.common.serialization.Serdes$ByteArraySerde
- IMPORTANCE – medium

num.standby.replicas – Число резервных реплик для каждой задачи

- TYPE – int
- DEFAULT – 0
- IMPORTANCE – medium

num.stream.threads – Количество потоков для выполнения потоковой обработки

- TYPE – int
- DEFAULT – 1

- IMPORTANCE – medium

processing.guarantee – Гарантия на обработку. Возможные значения: “at_least_once” (по умолчанию) и “exact_once”. Для обработки “exact_once” требуется по меньшей мере три брокера по умолчанию, что является рекомендуемой настройкой для продуктивной среды; для разработки можно изменить параметр, при этом переустановив настройку брокера *transaction.state.log.replication.factor*

- TYPE – string
- DEFAULT – at_least_once
- VALID VALUES – [at_least_once, exactly_once]
- IMPORTANCE – medium

security.protocol – Протокол безопасности для связи между брокерами. Допустимые значения: “PLAINTEXT”, “SSL”, “SASL_PLAINTEXT”, “SASL_SSL”

- TYPE – string
- DEFAULT – PLAINTEXT
- IMPORTANCE – medium

application.server – Пара “host:port”, указывающая на встроенную конечную точку пользователя, которая может использоваться для обнаружения местоположений хранилищ state stores в рамках одного приложения

- TYPE – string
- DEFAULT – “”
- IMPORTANCE – low

buffered.records.per.partition – Максимальное количество записей в буфере для каждой партиции

- TYPE – int
- DEFAULT – 1000
- IMPORTANCE – low

commit.interval.ms – Частота сохранения положения процессора. Если параметр *processing.guarantee* установлен на “exactly_once”, значение по умолчанию равно “100”, иначе (при “at_least_once”) значение по умолчанию равно “30000”. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 30000
- IMPORTANCE – low

connections.max.idle.ms – Закрытие бездействующих соединений по истечению заданного периода. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 540000
- IMPORTANCE – low

key.serde – Сериализатор/десериализатор для ключа, реализующего интерфейс *org.apache.kafka.common.serialization.Serde*. Данная конфигурация устарела, вместо нее используется *default.key.serde*

- TYPE – class
- DEFAULT – null

- IMPORTANCE – low

metadata.max.age.ms – Период времени, после которого принудительно обновляются метаданные даже при отсутствии видимых изменений в лидере партиции с целью предварительного обнаружения новых брокеров или партиций. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 300000
- VALID VALUES – [0,. . .]
- IMPORTANCE – low

metric.reporters – Список классов для использования в качестве репортеров метрик. Реализация интерфейса *org.apache.kafka.common.metrics.MetricsReporter* позволяет подключать классы, которые будут уведомлены о создании новой метрики. JmxReporter всегда включен в реестр статистических данных JMX

- TYPE – list
- DEFAULT – “”
- IMPORTANCE – low

metrics.num.samples – Количество выборок, поддерживаемых для вычисления метрик

- TYPE – int
- DEFAULT – 2
- VALID VALUES – [1,. . .]
- IMPORTANCE – low

metrics.recording.level – Самый высокий уровень записи для метрик

- TYPE – string
- DEFAULT – INFO
- VALID VALUES – [INFO, DEBUG]
- IMPORTANCE – low

metrics.sample.window.ms – Время ожидания вычисления метрик выборки. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 30000
- VALID VALUES – [0,. . .]
- IMPORTANCE – low

partition.grouper – Класс Partition grouper, реализующий интерфейс *org.apache.kafka.streams.processor.PartitionGrouper*

- TYPE – class
- DEFAULT – org.apache.kafka.streams.processor.DefaultPartitionGrouper
- IMPORTANCE – low

poll.ms – Время блокировки ожидания ввода. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 100

- IMPORTANCE – low

receive.buffer.bytes – Размер буфера приема TCP (SO_RCVBUF) при чтении данных. Если значение равно "-1", используется ОС по умолчанию

- TYPE – int
- DEFAULT – 32768
- VALID VALUES – [0,...]
- IMPORTANCE – low

reconnect.backoff.max.ms – Максимальный период времени ожидания повторного подключения к брокеру при неоднократных сбоях соединения. Отсрочка на хост увеличивается экспоненциально для каждого последующего сбоя соединения, вплоть до установленного максимума. После расчета увеличения отсрочки к значению добавляется *20%* случайного джиттера во избежание помех связи. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 1000
- VALID VALUES – [0,...]
- IMPORTANCE – low

reconnect.backoff.ms – Базовый период времени ожидания повторного подключения к хосту. Позволяет избегать многократного подключения к узлу в узком цикле. Данная отсрочка применяется ко всем попыткам подключения клиента к брокеру. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 50
- VALID VALUES – [0,...]
- IMPORTANCE – low

request.timeout.ms – Максимальное время ожидания клиентом ответа на запрос. Если ответ не получен до истечения установленного значения, клиент повторно отправляет запрос при необходимости. Указывается в миллисекундах

- TYPE – int
- DEFAULT – 40000
- VALID VALUES – [0,...]
- IMPORTANCE – low

retries – Установка значения больше нуля приводит к тому, что клиент переотправляет любую запись, передача которой завершается с временной ошибкой

- TYPE – int
- DEFAULT – 0
- VALID VALUES – [0,...,2147483647]
- IMPORTANCE – low

retry.backoff.ms – Время ожидания перед повторной попыткой отправки неудавшегося запроса в партицию топика. Позволяет избежать неоднократной отправки запросов в сжатом цикле. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 100

- VALID VALUES – [0,...]
- IMPORTANCE – low

rocksdb.config.setter – Класс или имя класса установщика конфигурации базы данных Rocks, реализующий интерфейс *org.apache.kafka.streams.state.RocksDBConfigSetter*

- TYPE – class
- DEFAULT – null
- IMPORTANCE – low

send.buffer.bytes – Размер буфера отправки TCP (SO_SNDBUF) при отправке данных. Если значение равно "-1", используется ОС по умолчанию

- TYPE – int
- DEFAULT – 131072
- VALID VALUES – [0,...]
- IMPORTANCE – low

state.cleanup.delay.ms – Время ожидания перед удалением state каталогов после перемещения партиции. Удаляются только state каталоги, которые не были изменены. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 600000
- IMPORTANCE – low

timestamp.extractor – Класс выделения временных меток, реализующий интерфейс *org.apache.kafka.streams.processor.TimestampExtractor*. Данная конфигурация устарела, вместо нее используется *default.timestamp.extractor*

- TYPE – class
- DEFAULT – null
- IMPORTANCE – low

value.serde – Класс сериализатора/десериализатора для значения, реализующего интерфейс *org.apache.kafka.common.serialization.Serde*. Данная конфигурация устарела, вместо нее используется *ddefault.value.serde*

- TYPE – class
- DEFAULT – null
- IMPORTANCE – low

windowstore.changelog.additional.retention.ms – Добавление *maintainMs* с целью исключения риска преждевременного удаления данных из журнала. Позволяет осуществлять отставание часов. По умолчанию устанавливается 1 день. Указывается в миллисекундах

- TYPE – long
- DEFAULT – 86400000
- IMPORTANCE – low

zookeeper.connect – Соединение строки для управления топиками ADS через Zookeeper. Конфигурация устарела и игнорируется, поскольку Streams API больше не использует Zookeeper

- TYPE – string

- DEFAULT – “”
- IMPORTANCE – low