

Установка менеджера рассылок

- [Установка менеджера рассылок](#)
- [Настройка менеджера рассылок](#)
- [Установка и настройка сервиса уведомлений](#)
- [Настройка для уведомлений о сгорании бонусов](#)
- [Настройка отправки уведомлений об изменении баланса](#)
- [Обновление менеджера рассылок](#)
- [Логирование менеджера рассылок](#)



Устанавливается на систему начиная с Ubuntu 18.04.

Для установки менеджера рассылок сначала необходимо получить сертификат.

Команда получения сертификата

```
curl http://downloads.artix.su/static/downloads/docker/cs_env/import_docker_cert.sh -o /tmp/import_docker_cert.sh
chmod u+x /tmp/import_docker_cert.sh
/tmp/import_docker_cert.sh
```

Установка менеджера рассылок

1. Для установки менеджера рассылок необходимо скачать скрипт:

```
curl -L http://downloads.artix.su/static/downloads/docker/delivery/install.sh -o install.sh
```

2. Сделать файл исполняемым:

```
chmod u+x install.sh
```

3. Запустить скрипт установки:

```
./install.sh
```

4. Скрипт установит docker со всем окружением, скачает docker-compose и запустит скрипт развертывания образов для работы с рассылками и запустит их.



Для рассылки сообщений об операциях с бонусами нужно установить [сервис уведомлений](#).

Настройка менеджера рассылок

Для того чтобы сервис мог получать информацию о сгорании бонусов, нужно его сконфигурировать. Всю конфигурацию сервиса можно передать через [сетевые настройки менеджера рассылок](#) и "[Настройки внешних шлюзов](#)".



В Artix Loyalty Management обязательно должны быть указаны корректные настройки доступа до КС и БС.

Установка и настройка сервиса уведомлений

Данный сервис нужен для отправки сообщений о сгорании, начислении и списании бонусов в менеджер рассылок.

Для установки сервиса необходимо выполнить команду:

```
sudo apt-get install artixcs-accounting-event-notifier
```

Настройка работы сервиса осуществляется в конфигурационном файле /opt/artixcs-accounting-event-notifier/application.properties.

Пример настройки

```
server.port=39011
#Уровень логирования
logging.level.su.artix.accounting=info
#####
# Задержка в миллисекундах, в течении которого будут группироваться операции для почечового объединения
su.artix.accounting.event.notifier.group.duration=10000
# Время (в секундах) - сколько будет храниться информация о посланных транзакциях.
# Чем больше значение - тем больше времени может простаивать "принимающий" события сервер.
# С другой стороны, чем больше это значение, тем большая нагрузка будет на СУБД. Параметр нужно подбирать эмпирически
su.artix.accounting.event.notifier.mongo.send.event.expired.seconds=3600
# После этого количества ошибок сервис уйдет в состояние ERROR и перестанет обрабатывать сообщения.
# Состояние можно получить запросом GET http://localhost:${server.port}/v1/state
su.artix.accounting.event.notifier.retry.count=50
# Если в БД не сохранено время последнего прочитанного события - начинать искать записи с текущего момента
su.artix.accounting.event.notifier.begin.from.current=true
### Автоматический перезапуск сервиса
# Перезапускать сервис, после перехода в состояние ERROR
su.artix.accounting.event.notifier.restart.auto.allow=true
# Через сколько секунд перезапускать сервис
su.artix.accounting.event.notifier.restart.after.seconds=10
#####
# Тип сервиса, на который будут слаться события: [ARTIX,TDERA]
su.artix.accounting.event.notifier.type=ARTIX
#####
### Общие параметры для send-ера
# время ожидания, за которое принимающий сервер должен установить соединение
su.artix.accounting.event.notifier.rest.sender.connection.timeout.seconds=1
# Для ARTIX
su.artix.accounting.event.notifier.rest.sender.prefix=http://localhost:8110/rest/sender/
su.artix.accounting.event.notifier.rest.sender.user=admin
su.artix.accounting.event.notifier.rest.sender.password=admin
#####
# TDERA формат сообщения
su.artix.accounting.event.notifier.rest.tdera.url=http://in.gates.wp.tdera.ru/customer/balance/
su.artix.accounting.event.notifier.rest.tdera.auth=Bearer YXJ0aXhfa2Fzc2E6YXJ0aXhfa2Fzc2FfdGVzdA==
su.artix.accounting.event.notifier.rest.tdera.sender=artix
#####
#Версия протокола, согласно которому сервис будет извлекать данные из коллекции с событиями. На текущий момент доступна только VERSION_1
su.artix.accounting.event.notifier.data.protocol.version=VERSION_1
su.artix.accounting.event.notifier.data.mongodb.collection.name=event
spring.data.mongodb.uri=mongodb://localhost/accountingEvent
#####
...
# Используемая БД для сервиса формирования сгоревших бонусов. Возможные значения MYSQL, MSSQL
su.artix.accounting.event.notifier.bonusfetcher.db=MYSQL
# Настройки подключения к БС
su.artix.accounting.sql.driver=com.mysql.jdbc.Driver
su.artix.accounting.sql.url=jdbc:mysql://localhost:3306/db_bonuses_certificates
su.artix.accounting.sql.user=netroot
su.artix.accounting.sql.passwd=netroot
```

После установки корректных настроек, необходимо перезапустить сервис командой:

```
sudo service artixcs-accounting-event-notifier restart
```

При настройке и запуске сервиса на внешний сервис будут отправляться сообщения вида:

- в режиме ARTIX (в качестве примера указано формирование запроса, если его посылать с помощью curl):

```
curl -uadmin:admin -XPOST -H "Content-Type: application/json; charset=UTF-8" "http://localhost:8083/rest/sender/send" \
--data '{"number":"3000","accountNumber":"3000","amountAdd":92597,"amountSub": -322,"date":"2018-01-10T08:57:21.234+0000","cashId":"1",
shiftNum":4,"checkNum":5,"sessionId":"1$3000-7c39e6b2-750a-400e-9dd1-76cee3039eac$5$4"}'
```

- в режиме TDERA:

```
POST {"params":{"sender":"artix"},"doc":{"number":"3000","accountNumber":"3000","amountAdd":92597,"amountSub": -322,"date":"2018-01-10T08:
57:21.234+0000","cashId":"1","shiftNum":4,"checkNum":5,"sessionId":"1$3000-7c39e6b2-750a-400e-9dd1-76cee3039eac$5$4"}}
```

Логирование работы сервиса выполняется в файле /var/log/artix/artixcs-accounting-event-notifier/artixcs-accounting-event-notifier.log.

Настройка для уведомлений о сгорании бонусов

В artixcs-accounting-event-notifier добавлен сервис, формирующий сгоревшие бонусы и добавляющий event в БД notifier.

Для того чтобы запустить сервис по отправке сгораний нужно включить настройку командой:

```
su.artix.accounting.event.notifier.bonusfetcher.enable=true
```

и перезапустить сервис:

```
sudo service artixcs-accounting-event-notifier restart
```

В БД accountingEvent в коллекцию event будут поступать сообщения о сгорании. Для этих видов сообщений (которые будут отдаваться клиенту) добавлено два типа CHECK (это сообщения ADD, SUB, BACK, PAY) и FIRED. Сообщения о сгорании отличаются от сообщений для остальных операций (ADD, SUB, BACK, PAY).

Сообщения о сгорании бонусов будут формироваться из БД, которая будет указана в настройках.

Проводники (asset) группируются по accountNumber (транзакции собираются в одну по номеру счета) с lastStatus=COMMITTED и status=ADD,BACK за время с момента последнего запуска. Настройка su.artix.accounting.event.notifier.bonusfetcher.scheduler.cron в конфигурационном файле /opt/artixcs-accounting-bonuses/application.properties отвечает за периодичность запуска сервиса формирования сгораний.

Настройка сервиса

```
#####
# Включение сервиса получения и отправки сгоревших бонусов
su.artix.accounting.event.notifier.bonusfetcher.enable=true
# Настройка запуска сервиса по крону. По-умолчанию раз в сутки
# Сгоревшие бонусы формируются за период с момента последнего запуска
# Если по каким-то причинам эта информация отсутствует(нештатная ситуация),
# то сгоревшие бонусы будут сформированы за последние 24 часа
# Формирование: по счету суммируются все транзакции со статусами ADD и BACK
# На основе этой информации формируется уникальный дескриптор, чтобы не отправлять
# одни и те же сгорания дважды
su.artix.accounting.event.notifier.bonusfetcher.scheduler.cron=0 * * * * ?
# Формирование сгоревших бонусов - длительная операция
# Количество попыток сформировать сгоревшие бонусы. По умолчанию 1 попытка
# Пример: настройка = 2
# Сервис формирует сгоревшие бонусы раз в сутки, если произошла штатная ситуация,
# то сервис попробует снова. Итого 2 попытки
su.artix.accounting.event.notifier.bonusfetcher.trycount=1
# Используемая БД для сервиса формирования сгоревших бонусов. Возможные значения MYSQL, MSSQL
su.artix.accounting.event.notifier.bonusfetcher.db=MYSQL
# Настройки подключения к БС
su.artix.accounting.sql.driver=com.mysql.jdbc.Driver
su.artix.accounting.sql.url=jdbc:mysql://localhost:3306/db_bonuses_certificates
su.artix.accounting.sql.user=netroot
su.artix.accounting.sql.passwd=netroot
```

Настройка отправки уведомлений об изменении баланса

Один из способов настроить отработку уведомлений покупателям об изменении баланса:

- установить Accounting-сервер;
- указать подключение БС к БД Mongo (здесь имеется ввиду не хранилище данных accounting-сервера, а БД, в которую будут помещаться события об изменении баланса на счетах);
- настроить и запустить сервис отправки уведомлений artixcs-accounting-event-notifier (см. выше).

Настройка формирования сообщений об изменении баланса на счетах находится в конфигурационном файле БС, например /opt/artixcs-accounting-bonuses-certificates/application.properties.

Состав данных, которые будут передаваться на внешние сервисы, определяется выбранными настройками. При включении настройки accounting.changeBalance.event.enabled сервис будет отправлять сообщения, содержащие информацию в формате описанном выше. Для отправки в сообщении не только данных об изменении баланса, но и значение баланса, который установлен на карте после начисления/списания, необходимо для параметра accounting.changeBalance.event.sendBalance установить значение true (активный параметр влияет на производительность).

Наименование	Тип данных	Возможные значения	Описание	Примечания
accounting.changeBalance.event.enabled	логический	<ul style="list-style-type: none">• true• false	Генерировать события по изменению баланса	По умолчанию false
accounting.changeBalance.event.protocol.version	строковый	VERSION_1	Внутренняя версия протокола между accounting-сервером и данным сервисом уведомлений. Не влияет на формат сообщения, посылаемый внешнему сервису	В настоящее время доступна только VERSION_1
accounting.changeBalance.event.mongodb.uri	строковый		Параметры подключения к БД Mongo, в которую производится запись событий об изменении баланса	
accounting.changeBalance.event.mongodb.database	строковый		Название БД, в которую записываются события об изменении баланса	
accounting.changeBalance.event.sendBalance	логический	<ul style="list-style-type: none">• true• false	Указывать в событии баланс по счету после применения операций начисления/списания	По умолчанию false При значении true производительность accounting-сервера может сильно просесть

Пример настройки

Отправка сигналов об изменении баланса на счетах ###
#включить генерацию событий по изменению баланса. События генерируются на основе команды комита asset, т.е. операции сгорания бонусов в данных событиях НЕ БУДЕТ
accounting.changeBalance.event.enabled = true
#Версия протокола, согласно которому сервис будет извлекать данные из коллекции с событиями. На текущий момент доступна только VERSION_1
accounting.changeBalance.event.protocol.version=VERSION_1
#Параметры подключения к БД mongodb, в которую будут писаться события об изменении баланса
accounting.changeBalance.event.mongodb.uri = mongodb://localhost
#Название БД mongodb, в которую будут писаться события об изменении баланса
accounting.changeBalance.event.mongodb.database = accountingEvent
#Указывать в событии баланс по счету ПОСЛЕ применения операции. Внимание! Включение опции может серьезно просадить производительность!
accounting.changeBalance.event.sendBalance = false

Обновление менеджера рассылок

Для обновления менеджера рассылок необходимо:

1. Перейти в директорию, где находится docker-compose.yml.
2. Выполнить команду для выключения контейнеров:

docker-compose down

3. Выполнить команду для обновления образов:

```
docker-compose pull
```

4. Выполнить команду для обновления контейнеров:

```
docker-compose up -d
```

Логирование менеджера рассылок

Логи рассылок хранятся в директории `/var/lib/docker/volumes/ubuntu_delivery_logs/_data/logs`. Запись логов ведется в несколько файлов:

Файл	Описание
delivery-gateway.log	Лог сервиса по отправке сообщений.
delivery-service.log	Лог сервиса рассылок. Сервис, в котором формируются сообщения для отправки.