

Exchange Listener

Настроить сервис синхронизации Exchange Listener

Версия 8.0



Эта документация предоставляется с ограничениями на использование и защищена законами об интеллектуальной собственности. За исключением случаев, прямо разрешенных в вашем лицензионном соглашении или разрешенных законом, вы не можете использовать, копировать, воспроизводить, переводить, транслировать, изменять, лицензировать, передавать, распространять, демонстрировать, выполнять, публиковать или отображать любую часть в любой форме или посредством любые значения. Обратный инжиниринг, дизассемблирование или декомпиляция этой документации, если это не требуется по закону для взаимодействия, запрещены.

Информация, содержащаяся в данном документе, может быть изменена без предварительного уведомления и не может гарантировать отсутствие ошибок. Если вы обнаружите какие-либо ошибки, сообщите нам о них в письменной форме.

Содержание

Настроить сервис синхронизации Exchange Listener	4
Способы развертывания сервиса Exchange Listener	4
Настроить сервис Exchange Listener на стороне Creatio	9
Диагностика настроек Exchange Listener	10

Настроить сервис синхронизации Exchange Listener

ПРОДУКТЫ: [ВСЕ ПРОДУКТЫ](#)

Сервис синхронизации Exchange Listener реализует синхронизацию Creatio с почтовыми сервисами [MS Exchange](#) и [IMAP/SMTP](#), используя механизм подписки.

В данной статье описан процесс развертывания сервиса синхронизации для Creatio on-site.

Сервис состоит из двух обязательных компонентов:

- основной модуль Exchange Listener;
- NoSQL СУБД Redis.

Модуль Exchange Listener инициирует исходящее подключение к EWS API. При этом используются учетные данные почтового ящика и создается подписка (subscription) для получения событий при поступлении новых писем. Открытая подписка остается в памяти компонента для обеспечения максимально оперативной реакции на получение нового письма. При получении соответствующего события выполняется загрузка экземпляра письма.

Для развертывания сервиса **достаточным условием** является использование in-мемори хранилища.

СУБД Redis используется для создания масштабируемой и отказоустойчивой системы узлов-обработчиков. Хранилище Redis содержит информацию об обслуживаемых почтовых ящиках. Это позволяет любому контейнеру обработать запросы Creatio на создание новой подписки или проверить статус конкретной подписки, независимо от того, на каком узле открыта подписка.

Обязательные требования к Redis:

- разрешен анонимный доступ;
- выделена отдельная база данных для работы сервиса Exchange Listener.

Способы развертывания сервиса Exchange Listener

Для развертывания сервиса предпочтительным способом является **использование оркестратора Kubernetes и пакетного менеджера Helm**. [Подробнее >>>](#)

Для более быстрого развертывания в среде разработки можно **использовать Docker**. [Подробнее >>>](#)

Развернуть сервис синхронизации с использованием Kubernetes

Для развертывания сервиса выполните следующие шаги:

1. Предварительно настройте целевое окружение:
 - a. Кластер Kubernetes. Подробно о том, как настроить и администрировать кластер, читайте [на сайте документации Kubernetes](#).
 - b. Пакетный менеджер Helm. Установка пакетного менеджера подробно описана [на сайте](#)

[документации Helm](#).

2. Установите Redis. Установка Redis с использованием Helm детально описана [на сайте GitHub](#).

Пример команды для установки Redis:

```
helm install --namespace default --set auth.enabled=false --set=slave.persistence.enabled=fal
```

В этом примере:

default — наименование namespace, куда будет установлен Redis;

redis — произвольное имя для экземпляра Redis.

3. Установите модуль Exchange Listener. Для установки модуля [скачайте helm-пакет](#). Доступные параметры helm-пакета описаны в таблице ниже.

Важно. Для более новых версий Kubernetes укажите версию API, добавив параметр:

```
--set apiVersion=apps/v1
```

Пример команды для установки Exchange Listener с использованием адреса и относительного пути сервиса:

```
helm install --set env.host=<redis_host> --set ApiUrl=<kubernetes_url> --set ingress.enabled=
--set apiVersion=apps/v1 --namespace <namespace name> exchangelistener </path/to/helm/exchang
```

В этом примере:

<redis_host> — адрес Redis-сервера;

<kubernetes_url> — URL или IP-адрес Kubernetes.

Адрес сервиса Exchange Listener: **<kubernetes_url>/<listener_path>**.

Чтобы проверить доступность, сделайте запрос по адресу:

<kubernetes_url>/<listener_path>/api/listeners/status (Рис. 1).

Пример команды для установки Exchange Listener с использованием Node IP и адреса порта:

```
helm install --set env.host=<redis_host> --set service.type=<node_IP> --set service.nodePort=
```

Адрес сервиса Exchange Listener — **<node_IP:node_port>**.

Чтобы проверить доступность, сделайте запрос по адресу: **<node_IP:node_port>/api/listeners/status** (Рис. 1).

Рис. 1 — Пример ответа сервиса Exchange Listener

```
{
  "ServiceStatus": "Started",
  "version": "0.5.0",
  "connections": {
    "657b3ea8-477f-419c-a07a-4d4cc2158fc5": {
      "SenderEmailAddress": " ",
      "BpmUser": " ",
      "BpmEndpoint": "https:// /0/ServiceModel/ExchangeListenerService.svc/NewEmail",
      "State": "exists",
      "Id": "657b3ea8-477f-419c-a07a-4d4cc2158fc5",
      "RedisKey": "Subscription_657b3ea8-477f-419c-a07a-4d4cc2158fc5_exchangelistener-api-2",
      "UseFullEmail": true
    },
    "332bcae2-0530-4636-bc09-50a594389f53": {
      "SenderEmailAddress": " ",
      "BpmUser": " ",
      "BpmEndpoint": "https:// /0/ServiceModel/ExchangeListenerService.svc/NewEmail",
      "State": "exists",
      "Id": "332bcae2-0530-4636-bc09-50a594389f53",
      "RedisKey": "Subscription_332bcae2-0530-4636-bc09-50a594389f53_exchangelistener-api-1",
      "UseFullEmail": true
    }
  }
}
```

На заметку. Если вам нужно отключить маркировку загружаемых писем категорией “Creatio”, то при установке приложения в команде `helm install` нужно указать параметр `--set FeaturesMarkEmailsAsSynchronized=false`.

Доступные параметры helm-пакета Exchange Listener

Параметр	Описание параметра	Значение по умолчанию
replicaCount	Количество StatefulSet-обработчиков.	2
service.type	Тип сервиса. Детально типы сервисов Kubernetes описаны в документации Kubernetes .	ClusterIP
service.nodePort	Если параметр service.type равен NodePort, то в этом параметре указывается внешний порт сервиса. Детально тип NodePort описан в документации Kubernetes .	
env.host	Адрес хоста Redis.	
env.port	Порт хоста Redis.	6379
env.base	Номер базы данных Redis.	0
ingress.enabled	Использование переопределения адресов при помощи ingress.	false
ApiUrl	Адрес сервиса при ingress.enabled=true.	
ingress.path	Относительный путь сервиса.	
log4Net.level	Уровень логирования по умолчанию.	Info

Для расчета требований к серверам воспользуйтесь [калькулятором системных требований](#).

Развернуть сервис синхронизации в Docker

Для настройки сервиса необходим сервер (физический или виртуальный компьютер) с установленной ОС Linux или Windows.

Важно. Развертывание сервиса синхронизации в Docker рекомендуется к установке только для dev-среды. Данный способ отличается высокой скоростью развертывания, но он не обеспечивает выполнение требований, актуальных для продуктивной среды: отказоустойчивости функции, масштабирования для обработки большого количества запросов и единого подхода к управлению компонентами с использованием систем оркестрации контейнеров. Для продуктивных сред настоятельно рекомендуется установка с использованием оркестратора Kubernetes и пакетного менеджера Helm.

Для развертывания сервиса выполните следующие шаги:

1. Предварительно настройте целевое окружение:

- a. Платформа контейнеризации Docker. Подробно о том, как установить и настроить платформу, описано [на сайте документации Docker](#).
 - b. СУБД Redis. Redis Server устанавливается при помощи [установочного файла для ОС Windows](#) или при помощи [инструкции для ОС Linux](#). Необходимо разворачивать анонимный СУБД Redis. Подробнее о разворачивании СУБД Redis в Docker читайте на сайте сообщества [Docker Hub](#).
2. Установите и запустите модуль Exchange Listener. Для этого необходимо скачать и развернуть образ Docker-контейнера.

Рассмотрим пример команды для скачивания и запуска образа с использованием командной строки и установленного Docker.

```
docker run \
-d \
# Проброс портов
-p <localhost_port>:80 \
--restart unless-stopped \
# Подключение к Redis
--env ExchangeListenerRedisHost=<redis_host>:<redis_port> \
--env ExchangeListenerRedisDatabase=<redis_database_number> \
--env PodName=ExchangeListener \
--name ExchangeListener \
# Актуальный образ ExchangeListener в Docker Hub
bpmonline/exchangelistener:<listener_version>
```

В этом примере:

<localhost_port> — порт локального сервера;

<redis_host> — адрес redis-сервера;

<redis_database_number> — номер БД redis-сервера;

<service_name> — название сервиса (задается вручную).

<listener_version> — версия микросервиса Listener.

На заметку. Актуальную версию Exchange Listener можно узнать в сообществе Docker Hub по [ссылке](#).

3. Чтобы проверить доступность развернутого Docker-контейнера, выполните команду:

```
docker ps -a --filter "name=<service_name>"
```

Адрес сервиса Exchange Listener — **localhost:<localhost_port>**.

Чтобы проверить доступность, сделайте запрос по адресу: <Localhost:<localhost_port>/api/listeners/status (Рис. 2).

Настроить сервис Exchange Listener на стороне Creatio

1. Убедитесь, что анонимный сервис ExchangeListenerService доступен по адресу [Адрес приложения Creatio]/0/ServiceModel/ExchangeListenerService.svc (Рис. 2).

Рис. 2 — Пример ответа сервиса ExchangeListenerService

Service

This is a Windows® Communication Foundation service.

Metadata publishing for this service is currently disabled.

If you have access to the service, you can enable metadata publishing by completing the following steps to modify your web or application configuration file:

1. Create the following service behavior configuration, or add the <serviceMetadata> element to an existing service behavior configuration:

```
<behaviors>
  <serviceBehaviors>
    <behavior name="MyServiceTypeBehaviors" >
      <serviceMetadata httpGetEnabled="true" />
    </behavior>
  </serviceBehaviors>
</behaviors>
```

2. Add the behavior configuration to the service:

```
<service name="MyNamespace.MyServiceType" behaviorConfiguration="MyServiceTypeBehaviors" >
```

Note: the service name must match the configuration name for the service implementation.

3. Add the following endpoint to your service configuration:

```
<endpoint contract="IMetadataExchange" binding="mexHttpBinding" address="mex" />
```

Note: your service must have an http base address to add this endpoint.

The following is an example service configuration file with metadata publishing enabled:


```
<configuration>
  <system.serviceModel>

    <services>
      <!-- Note: the service name must match the configuration name for the service implementation. -->
      <service name="MyNamespace.MyServiceType" behaviorConfiguration="MyServiceTypeBehaviors" >
        <!-- Add the following endpoint. -->
        <!-- Note: your service must have an http base address to add this endpoint. -->
        <endpoint contract="IMetadataExchange" binding="mexHttpBinding" address="mex" />
      </service>
    </services>

    <behaviors>
      <serviceBehaviors>
        <behavior name="MyServiceTypeBehaviors" >
          <!-- Add the following element to your service behavior configuration. -->
          <serviceMetadata httpGetEnabled="true" />
        </behavior>
      </serviceBehaviors>
    </behaviors>

  </system.serviceModel>
</configuration>
```

For more information on publishing metadata please see the following documentation: <http://go.microsoft.com/fwlink/?LinkId=65455>.

2. Установите нужные значения системных настроек. Для этого:
 - a. Перейдите в дизайнер системы, например, по кнопке .
 - b. В блоке “Настройка системы” перейдите по ссылке “Системные настройки”.
 - c. Укажите значения системных настроек:

“**ExchangeListenerServiceUri**”(код “ExchangeListenerServiceUri”). Формат значения настройки:

[*адрес сервиса, используемый при установке*]/api/listeners.

“URL сервиса обработки событий Exchange в Creatio” (код “BpmonlineExchangeEventsEndpointUrl”). Формат значения настройки: [*адрес анонимного сервиса ExchangeListenerService*]/NewEmail. Например, <https://mycreatio.com/0/ServiceModel/ExchangeListenerService.svc/NewEmail>.

Диагностика настроек Exchange Listener

На странице диагностики настроек сервиса Exchange Listener представлены инструменты контроля и проверки работы сервиса.

Здесь вы можете:

- проверить подключение необходимой функциональности;
- убедиться в доступности сервиса;
- получить информацию о подписках;
- проверить правильность заполнения системной настройки “ExchangeListenerServiceUri”;
- проверить работоспособность почтового ящика.

Чтобы перейти на страницу диагностики настроек Exchange Listener, в адресной строке браузера добавьте “/0/ClientApp/#/IntegrationDiagnostics/ExchangeListener” к URL-адресу вашего сайта Creatio и нажмите Enter. Например:

```
http://mycreatio.com/0/ClientApp/#/IntegrationDiagnostics/ExchangeListener
```

Страница диагностики состоит из нескольких блоков с параметрами синхронизации, по которым можно запустить диагностику (Рис. 3). Чтобы отобразилась информация о параметрах подключения, необходимо в блоке с интересующим вас параметром нажать [*Запустить диагностику*].

Рис. 3 — Диагностика настроек Exchange Listener

Диагностика сервиса Exchange Listener

[ЗАПУСТИТЬ ПОЛНУЮ ДИАГНОСТИКУ](#)

Состояние доступности функциональности

- ✓ Синхронизация почты Exchange (ExchangeListenerEnabled)
- ✓ Синхронизация почты Exchange и IMAP (EmailIntegrationV2)
- ✗ Отправка писем (SendEmailsV2)

Проверка доступности сервиса

[ЗАПУСТИТЬ ДИАГНОСТИКУ](#)

Ошибка

Сервис ExchangeListener в данный момент не доступен для текущего сайта
Unable to connect to the remote server

Получение информации по подпискам

[ЗАПУСТИТЬ ДИАГНОСТИКУ](#)

Ошибка

Http failure response for
https://.../V0/rest/MailDiagnosticToolsService/GetSubscribersInfo: 500 OK

Проверка правильности заполнения системной настройки ExchangeListenerServiceUri

[ЗАПУСТИТЬ ДИАГНОСТИКУ](#)

Успешная операция

URL сервиса обработки событий Exchange в Creatio
https://.../V0/ServiceModel/ExchangeListenerService.svc/ProcessFullEmail валиден

Проверка работоспособности почтового ящика

Почтовый ящик



Отправить тестовое письмо

[ЗАПУСТИТЬ ДИАГНОСТИКУ](#)

Ошибка

Unable to connect to the remote server

Состояние доступности функциональности	<p>Для данного блока диагностика запускается автоматически при открытии страницы.</p> <p>Выполняется проверка подключения в Creatio функциональности, необходимой для работы Exchange Listener:</p> <ul style="list-style-type: none"> • ExchangeListenerEnabled; • EmailIntegrationV2; • SendEmailsV2. <p>Подробнее о подключении функциональности читайте в статье Механизм отключения функциональности Feature Toggle документации по разработке.</p>
Проверка доступности сервиса	Выполняется проверка доступности сервиса Exchange Listener из вашего приложения Creatio.
Получение информации по подпискам	Выполняется проверка соединения с удаленным сервером.
Проверка правильности заполнения системной настройки ExchangeListenerServiceUri	Выполняется проверка валидности адреса, указанного в системной настройке.
Проверка работоспособности почтового ящика	<p>Выполняется проверка работы почтовых ящиков MS Exchange.</p> <p>Установите признак [<i>Отправить тестовое письмо</i>], чтобы при запуске диагностики на указанный адрес было отправлено тестовое email-сообщение.</p>