

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

«Система поиска и установления информационных связей между данными "Ведемотик"»

Руководство по установке и эксплуатации ПО

Содержание

| | |
|---|----------|
| 1 ОБЩИЕ СВЕДЕНИЯ | 3 |
| 2 ПРОГРАММНО-АППАРАТНЫЕ СРЕДЫ ФУНКЦИОНИРОВАНИЯ ПО | 3 |
| 3 ОБЩИЕ ПРИНЦИПЫ ФУНКЦИОНИРОВАНИЯ ПО | 3 |
| 4 ОБЯЗАННОСТИ И ФУНКЦИИ АДМИНИСТРАТОРА ЗАКАЗЧИКА..... | 4 |
| 5 УСТАНОВКА ПО | 4 |
| 5.1 Подготовка к установке | 4 |
| 6 УСТАНОВКА КОМПОНЕНТОВ..... | 5 |
| 6.1 Предварительный запуск | 5 |
| 6.2 Настройка модуля безопасности | 5 |
| 6.3 Основной запуск | 7 |
| 7 НАСТРОЙКА ПО | 7 |
| 7.1 Проверка корректности установки и настройки модулей ПО..... | 7 |
| 8 ПОДДЕРЖАНИЕ ФУНКЦИОНИРОВАНИЯ ПО | 8 |

1 ОБЩИЕ СВЕДЕНИЯ

Настоящий документ содержит руководство по установке и эксплуатации программного обеспечения «Система поиска и установления информационных связей между данными "Begemotik"» (далее — ПО, Система, Begemotik).

2 ПРОГРАММНО-АППАРАТНЫЕ СРЕДЫ ФУНКЦИОНИРОВАНИЯ ПО

Требования к клиенту

ПО работает через «тонкий клиент». Для доступа к системе необходимо использовать веб-браузер на базе Chromium (желательно использовать браузер Google Chrome последней версии).

Требования к серверу

ПО устанавливается на серверах под управлением любой Unix- совместимой ОС с использованием системы кластеризации Docker Swarm.

На данный момент Разработчиком протестирована и гарантируется работа ПО под управлением OS семейства Debian версий 9 и 10 (либо Ubuntu Server 18-20 версии) в связке с Docker-CE 20.10.

Также для доступа к ПО требуется стабильное интернет-соединение (не менее 10 Мбит/с) и доступ к ресурсу update.begemotik.facct.ru

Требования к оборудованию

| Характеристики | Минимальные | Рекомендуемые |
|----------------|------------------|---------------------------|
| Процессор | Intel i7 3.2 GHz | Intel Xeon E5-2670 и выше |
| RAM | 32 GB | 64 GB |
| Жесткий диск | 2x240 GB SSD | 2x240 GB NVMe |

3 ОБЩИЕ ПРИНЦИПЫ ФУНКЦИОНИРОВАНИЯ ПО

ПО представляет из себя клиент-серверное приложение, реализованное в виде "тонкого клиента" (SPA-приложение в веб-браузере) и сервера, состоящего из набора модулей, оформленных в виде образов Docker для удобства инсталляции и использования. ПО поделено на функциональные модули, отвечающие за интерфейс пользователя, подключение к источникам и анализ данных. Коммуникация между модулями ПО осуществляется через gRPC (по протоколу HTTP/2) и аутентификацией компонентов с использованием JWT.

Детальная информация о реализации ПО представлена в документе «Функциональные характеристики ПО».

4 ОБЯЗАННОСТИ И ФУНКЦИИ АДМИНИСТРАТОРА ЗАКАЗЧИКА

В обязанности администратора входит следующее:

- Произвести установку ПО.
- Поддерживать функционирование ПО.

5 УСТАНОВКА ПО

5.1 Подготовка к установке

В рамках подготовки к установке необходимо выполнить следующие шаги:

1. Обновить Операционную Систему и входящие в неё компоненты до последних версий.

Пример для ОС **Debian/Ubuntu**:

```
$ apt update && apt upgrade -y
```

2. Инициализировать Docker Swarm:

```
$ docker swarm init
```

3. Подключить Docker Registry:

```
$ docker login https://update.begemotik.facc.ru
```

Логин и пароль находятся в Приложении к документы, предоставляемому по запросу.

4. Создать необходимые конфигурационные файлы (docker configs):

```
$ docker config create <config_name> <config_file.conf>
```

Список необходимых конфигурационных файлов и их содержимое находится в Приложении к настоящему Руководству, предоставляемое по запросу Заказчика. Данную процедуру необходимо повторить для каждого конфигурационного файла.

5. Создать необходимые файлы секретов (docker secrets):

```
$ docker secret create <secret_name> <secret_file.txt>
```

Список необходимых файлов секретов и их содержимое находится в Приложении к настоящему Руководству, предоставляемое по запросу Заказчика. Данную процедуру необходимо повторить для каждого файла секретов.

6. Создать необходимые сети Docker:

```
$ docker network create -d overlay -o encrypted --subnet 10.200.0.0/16 begemotik  
$ docker network create -d overlay -o dmz --subnet 10.100.0.0/16 dmz
```

6 УСТАНОВКА КОМПОНЕНТОВ

Внимание! При выполнении всех действий в данном разделе необходимо, чтобы машина, на которой происходит установка ПО, имела доступ в сеть Интернет

Установка и запуск ПО происходит в несколько этапов.

1. Предварительный запуск, при котором происходит получение образов необходимых компонентов с сервера обновлений Разработчика, инициализация модулей.
2. Настройка модуля безопасности (генерация ключей компонентов, добавление ключей в настройки соответствующих компонентов)
3. Опциональная настройка модуля подключения данных (настройка подключения к необходимым источникам данных)
4. Основной запуск (на данном этапе приложение уже готово к финальной настройке и работе с пользователями)

В поставку компонентов могут входить следующие решения (не являющиеся частью ПО):

- Веб-сервер nginx (BSD);
- Application proxy Traefik (MIT);
- СУБД PostgreSQL (PostgreSQL – Open Source license, similar to the BSD or MIT licenses.);
- СУБД Redis (BSD);
- СУБД MongoDB (SSPL+Apache) – как источник демо-данных.

6.1 Предварительный запуск

Для установки и запуска всех компонентов необходимо создать Docker Stack:

```
$ docker stack deploy --with-registry-auth --compose-file begemotik.yml  
begemotik
```

6.2 Настройка модуля безопасности

*Внимание! На данном этапе необходимо убедиться, что в конфигурационном файле модуля безопасности по пути **server:auth:token:secret** установлен надёжный секретный ключ.*

Для взаимодействия с модулем Безопасности приложение должно иметь действительный AuthToken приложения, предоставляющий полномочия для выполнения определённого набора запросов к модулю Безопасности.

Для генерации токена приложения исполняемый файл Security поддерживает команду tokengen:

```
$ /bin/start tokengen
--recipient - название компонента, для которого генерируется токен. Пример:
begemotik_backend
--scope - название запроса, право на выполнение которого предоставляется в формате
servicename_rpcname
Пример:
AuthService.AuthUser - auth_authuser AccountService.ModifyAccountProfile-
account_modifyaccountprofile AccountService.GetAccount - account_getaccount
ManagementService.RegisterCompany - management_registercompany
--expiration - срок действия токена в днях.
Пример: Для действия токена в течение года - 365
Флаг --scope (-s) может дублироваться.
```

Для генерации токенов понадобится доступ к модулю Безопасности:

```
$ SECURITY_CONTAINER_ID=$(docker ps | grep begemotik | grep security | awk
'{print $1};')
$ docker exec -ti $SECURITY_CONTAINER_ID bash
```

В данный момент мы находимся в контейнере security и все команды по генерации токенов выполняются именно в нём.

Список компонентов и необходимые для них разрешения для запросов описаны в Приложении к настоящему Руководству, предоставляемое по запросу Заказчика.

Полученные токены необходимо добавить в конфигурационные файлы соответствующих компонентов. Обычно токены находятся в конфигурационных файлах по пути **security:token**. Также есть возможность переопределить значения конфигурационного файла через переменные окружения (**SECURITY_TOKEN**), что для токенов безопасности довольно удобно.

Переменные окружения можно как записать в stack-файл *begemotik.yml* в соответствующие блоки *environment* (рекомендуется), так и задать напрямую для сервисов средствами docker (**но в данном случае после удаления стека они пропадут**):

```
$ docker service --env-add SECURITY_TOKEN=<token> <service_name>
```

6.3 Основной запуск

После всех шагов, описанных в предыдущих пунктах, необходимо перезагрузить begemotik stack. Для этого надо выполнить следующие шаги:

```
$ docker stack rm begemotik  
$ docker stack deploy --with-registry-auth --compose-file begemotik.yml  
begemotik
```

7 НАСТРОЙКА ПО

После успешной установки ПО необходимо добавить первого пользователя с правами администратора. Сделать это можно только прямой записью в БД с пользователями.

```
$ DB_CONTAINER_ID=$(docker ps | grep begemotik | grep db | awk '{print  
$1};')  
$ docker cp add_user.sql $DB_CONTAINER_ID:/tmp  
$ docker exec -ti $DB_CONTAINER_ID bash
```

После выполнения данных шагов мы попали в контейнер БД, далее пример для СУБД PostgreSQL:

```
$ su postgres  
$ psql -d <begemotik_auth> -a -f /tmp/add_user.sql  
$ rm /tmp/add_user.sql
```

<begemotik_auth> - необходимо заменить на название БД, которая указана в соответствующем конфигурационном файле на этапе подготовки к установке ПО.

add_user.sql - содержимое данного файла SQL указано в Приложении. *Рекомендуем сменить пароль администратора на собственный перед началом установки!*

7.1 Проверка корректности установки и настройки модулей ПО

Проверка корректности установки происходит в 2 этапа:

1. Все сервисы, указанные в stack-файле begemotik.yaml, запущены:

```
$ docker stack ps begemotik | grep Running
```

Если каких-то сервисов нет в списке, необходимо просмотреть записи в журнальном файле соответствующего контейнера и исправить ошибки в конфигурации.

2. Веб-интерфейс ПО доступен и в нём отображается страница входа с предложением ввести логин и пароль для доступа к Системе.

8 ПОДДЕРЖАНИЕ ФУНКЦИОНИРОВАНИЯ ПО

Поддержание функционирования ПО состоит в контроле настроек, произведенных в рамках установки ПО и функционировании Docker Swarm. Иных регламентных мероприятий со стороны администратора заказчика ПО не требует.