

Содержание

1. Введение	3
2. Список используемых определений, сокращений и терминов	4
3. Начальные условия	5
4. Требования к ПО	6
5. Требования к оборудованию	7
6. Предварительная настройка	8
7. Установка	9
7.1. Вызов справки	9
7.2. Описание режимов установки	9
7.3. Критерии успешности установки	9
7.4. Проверка списка работающих контейнеров	10
7.5. Завершение установки с ошибкой	10
7.6. Базовые действия с продуктом после завершения установки	11
7.7. Рекомендуемые действия по настройке сторонних компонентов	12
8. Проверка работоспособности	13
8.1. Аутентификация пользователя на Портале	13
9. Удаление программы	15

2. Список используемых определений, сокращений и терминов

Представленная ниже таблица (Таблица 1) содержит список используемых сокращений и терминов.

Таблица 1. Список используемых сокращений и терминов

Термин	Описание
bash (Bourne Again SHell)	Командная оболочка (shell) для ОС Linux, которая предоставляет интерфейс для взаимодействия с ОС через команды и скрипты
billing	Сервис, отвечающий за расчет и выставление счетов
CPU	Вычислительное ядро процессора
Docker	Открытая платформа для автоматизации развёртывания, доставки и управления приложениями в контейнерах
Docker-compose	Инструмент для определения и управления многоконтейнерными приложениями в Docker
IaaS	Сервис, предоставляющий виртуальную инфраструктуру для развёртывания и управления приложениями
Orchestrator	Сервис, координирующий работу различных компонентов системы
Orchestrator-cron	Сервис, отвечающий за запуск задач по расписанию
Orchestrator-daemon	Сервис, выполняющий фоновые операции и управляющий другими сервисами
portal	Веб-интерфейс для управления и мониторинга системы
portal-amqp_daemon	Сервис, который работает с AMQP (Advanced Message Queuing Protocol) и отвечает за обработку сообщений, связанных с порталом. Может принимать, обрабатывать и отправлять сообщения через AMQP
portal-cron	Сервис, отвечающий за выполнение периодических задач в портале. Может быть настроен на запуск определенных задач по расписанию
portal_sidekiq	Сервис, использующий Sidekiq — фреймворк для обработки фоновых задач в Ruby. Отвечает за выполнение асинхронных задач, связанных с порталом, таких как обработка очередей задач, отправка уведомлений и т.д.
RAM	Оперативная память
SaaS	Модель обслуживания, при которой предоставляется пользователю готовое прикладное программное обеспечение, полностью обслуживаемое провайдером
ОС	Операционная система
ПО	Программное обеспечение
ЦОД	Центр обработки данных

3. Начальные условия



Онлайн версия требует доступа к сети Интернет и репозиторию образов. Инсталлятор загружает необходимые компоненты и файлы из репозитория после запуска установки.

Для доступа к репозиторию нужны учётные данные. Онлайн версия обеспечивает удобную установку и обновление программы.



Оффлайн версия не требует доступа к сети Интернет и репозиторию образов. Все компоненты и файлы уже находятся внутри установщика. Оффлайн версия полезна для установки на компьютер пользователя без подключения к сети Интернет.

Целью инсталлятора в обоих случаях является удобная и надёжная установка программы в дата-центре на виртуальную машину или физический сервер.

4. Требования к ПО

Минимальные системные требования для установки:

- Docker версии больше или равно 18.06.3.
- Docker-compose версии не ниже 1.27.0-rc1 и не выше 2.17.0-rc.1.
- Версия bash не ниже 4. Для проверки версии введите команду:

```
bash --version
```

- Пакет gettext. Для Ubuntu рекомендуется установить пакет gettext-base, а для CentOS 7 — пакет gettext.
- Наличие утилит: echo, md5sum, base64 и других. Утилиты используются в процессе установки и проверки целостности файлов.

Перед началом инсталляции для онлайн установки обеспечьте доступ к следующим ресурсам:

- Репозиторию для получения образов приложения: pkgs.basistech.ru/bcc (docker login). Для доступа к этому репозиторию потребуется учётная запись.
- Публичному реестру Docker (hub.docker.com) для получения образов инфраструктурных сервисов.

5. Требования к оборудованию

Минимальные системные требования:

- 8 Гбайт CPU;
- 16 Гбайт оперативной памяти (RAM);
- 100 Гбайт свободного места на жёстком диске.

6. Предварительная настройка



Инсталлятор устанавливает сторонние сервисы maildev, postgres, rabbitmq, redis, nginx в базовой минимальной конфигурации. Минимальная конфигурация и их дополнительная настройка не описаны в документации.

Перед началом установки выполните следующие шаги:

1. Выберите место для хранения файла (директорию).
2. Распакуйте архив с инсталлятором, используя следующую команду:

```
tar -xzf <имя_файла_дистрибутива>.tgz
```

Например, имя <имя_файла_дистрибутива> может быть следующим:

```
bcc_cmp-1.0.5.rc9_installer_v6_offline.tgz
```

3. Найдите в файле `install.conf` переменные с предварительно настроенными параметрами в таблице (Таблица 2).

Таблица 2. Список параметров и их значений

Параметр	Описание	Значение по умолчанию
COP_INSTALLER_APP_NAME	Переменная, содержащая наименование проекта. Допускается использовать буквы латинского алфавита, подчеркивание и тире.	bcc
COP_INSTALLER_APP_DIR	Каталог, содержащий место для установки файлов проекта.	\${PWD}/srv/\${COP_INSTALLER_APP_NAME}

4. Отредактируйте значения:
 - для переменной «COP_INSTALLER_APP_NAME» укажите новое наименование проекта;
 - для переменной «COP_INSTALLER_APP_DIR» укажите каталог (директорию), в который будут установлены файлы, например COP_INSTALLER_APP_DIR=/opt/bcc.
5. После внесения изменений в «install.conf», сохраните файл и перейдите к установке продукта.

Учётные данные для первого входа:

- в ходе установки продукта учётные данные пользователя (имя и пароль) автоматически создаются для входа в систему:
 - INITIAL_USERNAME — имя пользователя;
 - INITIAL_USER_PASSWORD — пароль.

Файл с секретами:

- в процессе установки будет создан файл с секретами (**secrets.txt**), который содержит важные конфиденциальные данные. Файл **secrets.txt** расположен на уровень выше распакованного архива.

Успешный логин:

- смените первоначальный пароль после первого входа. Подробнее процесс смены пароля описан в Руководстве Пользователя. Администратор.

7.4. Проверка списка работающих контейнеров

Для проверки списка работающих контейнеров и их состояния, выполните следующую команду в рабочей директории установщика:

```
"docker-compose ps"
```

Данная команда отобразит список всех контейнеров, запущенных с использованием файла **docker-compose.yml** в текущей директории. Каждый контейнер может находиться в одном из следующих состояний:

- «up» — контейнер работает и доступен;
- «healthy» — контейнер работает и доступен, успешно пройдена внутренняя проверка;
- «unhealthy» — контейнер помечен как неработоспособный.

Убедитесь, что ни один из контейнеров не помечен как «unhealthy». Если имеются контейнеры со статусом «unhealthy», это может указывать на проблемы с работоспособностью. Чтобы получить полный список контейнеров, запущенных после установки, выполните команду:

```
"docker ps -a"
```

Команда отобразит все контейнеры, включая остановленные и завершённые.

7.5. Завершение установки с ошибкой

Критерии ошибки:

- если в процессе установки возникли ошибки, отобразятся сообщения об ошибках в выводе скрипта инсталляции.

Поиск сообщений об ошибках:

- если установка завершилась с ошибкой появляется надпись: «Ошибка установки». Проверьте журналы и лог-файлы для поиска сообщений об ошибках. Имя файла с логом имеет следующий формат: «<Временная метка>_install.log». Например, имя файла с логом может выглядеть следующим образом «20240212117403926643_install.log».

Перезапуск и очистка после ошибки:

- если установка завершилась неудачно, после решения проблемы, указанной в логах неудачной установки, очистите все временные файлы и настройки, созданные при неудачной попытке командой:

```
bash install sh -u
```

Перезапустите процесс установки.

7.6. Базовые действия с продуктом после завершения установки

1. Поиск Лог-файлов (log files).

Лог-файлы расположены по пути <директория установки>/<компонент>/log, имеют расширение .log, где <директория установки> соответствует значению COP_INSTALLER_APP_DIR, см. раздел [Предварительная настройка](#).

Проверьте соответствующие файлы для получения информации о работе системы и возможных проблемах.

2. Перезапуск контейнера.

Для перезапуска конкретного контейнера, выполните команду:

```
"docker-compose restart <service_name>"
```

Примеры возможных service_name:

- billing
- iaas;
- orchestrator-cron;
- orchestrator-daemon;
- orchestrator;

- portal;
- portal-cron;
- portal-amqp_daemon;
- portal_sidekiq.

3. Файлы настроек.

Файлы настроек, такие как конфигурационные файлы (например, `products.yml`) и файлы окружения (`env`), находятся в директории установки в поддиректориях `<service_name>/configs`.

Откройте соответствующий файл с помощью текстового редактора и внесите необходимые изменения.

7.7. Рекомендуемые действия по настройке сторонних компонентов

Настройка SSL в nginx:

- для обеспечения безопасного соединения с сервером, рекомендуется настроить SSL веб-сервера nginx.

Следуйте документации по настройке [SSL](#).

Настройка отказоустойчивой конфигурации для postgresql, redis, rabbitmq:

- для обеспечения отказоустойчивости и надёжности системы настройте отказоустойчивую конфигурацию для базы данных:
 - postgresql <https://www.postgresql.org/docs/current/high-availability.html>;
 - кэш (redis):
 - <https://redis.io/docs/management/sentinel/>;
 - <https://redis.io/docs/management/scaling/>;
 - брокер сообщений (rabbitmq) <https://www.rabbitmq.com/clustering.html>.

Следуйте документации по настройке отказоустойчивости для соответствующих компонентов.