|  |
| --- |
| Инструкция по скачиванию и установке |
| ПО КИС «Теплоэнергоучет»Количество листов: 9  |

Содержание

[1 Общие сведения 3](#_Toc126059443)

[1.1 Сокращения и условные обозначения 3](#_Toc126059444)

[2 Скачивание и установка дистрибутива для развертывания портала 4](#_Toc126059445)

[2.1 Зависимости развертывания модулей 5](#_Toc126059446)

[3 Процесс развертывания модулей 6](#_Toc126059447)

[3.1 Базовая настройка ОС 6](#_Toc126059448)

[3.2 Настройка файлового сервера: 6](#_Toc126059449)

[3.3 Настройка сервера очередей: 6](#_Toc126059450)

[3.4 Настройка сервера базы данных 7](#_Toc126059451)

[3.5 Настройка web сервера с web приложениями и прокси 8](#_Toc126059452)

[3.6 Настройка backend модулей системы 8](#_Toc126059453)

# Общие сведения

Документ описывает процесс по скачиванию и установке программного продукта ПО КИС «Теплоэнергоучет».

## Сокращения и условные обозначения

|  |  |
| --- | --- |
| **Сокращение** | **Расшифровка** |
| ОС | Операционная система |
| ПО | Программное обеспечение |
| БД | База данных |
| СУБД | Средство управления базами данных |

# Скачивание и установка дистрибутива для развертывания портала

Скачивание дистрибутива производится по ссылке, предоставленной администратором вендора.

Список модулей системы представлен в таблице:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Модуль** | **name** | **Расположение** | **Port** | **Server** |
| Веб сервер с двумя приложениями (Frontend-react и Management-ui) Реверс прокси сервер | Nginx | Docker digitexp/nginxSettings + configs /etc/nginx + папки приложений | 80 | server-api-fe-01 |
| Фронт-энд портал | Frontend-react | /var/www/html/app | 80 | server-api-fe-01 |
| Фронт-энд админка | Management-ui | /var/www/html/admin | 80 | server-api-fe-01 |
| Свободная объектно-реляционная система управления базами данных (СУБД) | PostgreSQL | Docker digitexp/postgresql:12Configs /opt/POSTGRESQL\_HOME | 5432 | server-db-01 |
| Программный брокер сообщений на основе стандарта AMQP | RabbitMQ | Docker rabbitmq:latest | 5672 | server-mq-01 |
| Объединенные модули1. Конструктор отчетов
2. Справочники
3. Согласования
4. Администрирование
5. Администрирование по ОГ
6. Регламентная отчетность
7. Уведомления
8. Конструктор запросов
 | Dynamicdq | /opt/DYNAMICDQ\_HOME | 8804 | server-api-be-01 |
| Интеграция с СУ | Middle level integration | /opt/MIDDLE\_LEVEL\_INTEGRATION\_HOME | 8811 | server-api-be-01 |
| Шина данных | ESB | /opt/ESB\_HOME | 9001 | server-api-be-01 |
| Метрологический модуль | Сommercial | /opt/COMMERCIAL\_HOME | 8821 | server-api-be-01 |

## Зависимости развертывания модулей

|  |  |
| --- | --- |
| **Модуль** | **Зависимости** |
| Nginx | Нет |
| PostgreSQL | Нет |
| RabbitMQ | Нет |
| Dynamicdq | Nginx, PostgreSQL, RabbitMQ |
| Middle level integration | Nginx, PostgreSQL |
| ESB | Nginx, PostgreSQL |
| Commercial | Nginx, PostgreSQL, ESB |
| Frontend-react | Nginx, Dynamicdq, ESB, Commercial |
| Management-ui | Nginx, Dynamicdq, ESB |

# Процесс развертывания модулей

Ниже приведен пример развертывания демонстрационного стенда, для создания отказоустойчивого кластера требуется большее количество серверов и программных компонентов:

1. Подготовить виртуальные или физические серверы со следующими характеристиками:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Сервер** | **ОС** | **Процессор (ядра)** | **Память** | **Дисковое пространство** | **Предполагаемый IP адрес** |
| server-api-fe-01 | РЕД ОС | 16 | 24 | 200 ГБ | 192.168.0.2 |
| server-api-be-01 | РЕД ОС | 16 | 24 | 200 ГБ | 192.168.0.3 |
| server-db-01 | РЕД ОС | 24 | 24 | 250 ГБ | 192.168.0.4 |
| server-mq-01 | РЕД ОС | 16 | 16 | 100 ГБ | 192.168.0.5 |
| server-fs-01 | РЕД ОС | 16 | 16 | 100 ГБ | 192.168.0.6 |

## Базовая настройка ОС

Произвести базовую настройку ОС на всех созданных серверах согласно инструкции производителя (<https://redos.red-soft.ru/base/>):

1. Настроить сетевое подключение
2. Выставить точное время, указать внешний источник точного времени
3. Задать доменное имя

## Настройка файлового сервера:

1. На сервере server-fs-01 создать сетевую директорию
2. Подключить сетевой диск к серверу server-api-be-01 в директорию
/opt/FILES\_HOME

## Настройка сервера очередей:

Согласно ниже приведенной инструкции выполнить первичную установку и настройку ПО RabbitMQ на сервере server-mq-01

Установить сервер RabbitMQ

|  |
| --- |
| yum install rabbitmq-serversystemctl enable rabbitmq-server.service systemctl start rabbitmq-server.service |

Произвести следующую настройку

|  |
| --- |
| rabbitmqctl add\_user 'RabbitMQ' 'Password' rabbitmqctl set\_user\_tags RabbitMQ administrator rabbitmqctl delete\_user guestrabbitmqctl set\_permissions -p / RabbitMQ "." "." ".\*" rabbitmqctl list\_users rabbitmq-plugins enable rabbitmq\_management systemctl restart rabbitmq-server |

Скопировать заранее подготовленные конфигурации сервера очередей с внешнего ресурса (внешняя ссылка будет предоставлена по требованию)

На Web интерфейсе ПО RabbitMQ (по умолчанию http://192.168.0.5:15672) выполнить следующие действия

1. вход с следующими учетными данными - login: RabbitMQ password: Password
2. На вкладке Overview в раздел Import / export definitions выполнить импорт конфигурации из файла

## Настройка сервера базы данных

Согласно ниже приведенной инструкции выполнить первичную установку и настройку сервера БД на сервере server-db-01

Установить СУБД Postgres при помощи следующей команды:

|  |
| --- |
| dnf install postgresql-server postgresql-contribsystemctl start postgresql.servicesystemctl enable postgresql.service |

Скопировать заранее подготовленную резервную копию структуры и демо данных базы данных с внешнего ресурса.

Перенести скопированные файлы в директорию и запустить из этого каталога следующую команду:

|  |
| --- |
| zcat backup\_name\_full.sql.gz | su postgres -c psql |

## Настройка web сервера с web приложениями и прокси

Список разворачиваемых компонентов:

1. Nginx
2. Frontend-react
3. Management-ui

**Инструкция по развертыванию**

1. Установить службу контейнеризации на сервере server-api-fe-01 при помощи следующих команд:

|  |
| --- |
| systemctl start docker systemctl enable dockerdnf install docker-ce |

1. Скопировать заранее подготовленные файлы **docker image** с внешнего ресурса (внешняя ссылка будет предоставлена по требованию)
2. Запустить **docker container** на основе полученного **docker image**

|  |
| --- |
| docker load < nginx.tardocker run -d -p 80:80 --name nginx nginx |

## Настройка backend модулей системы

Список разворачиваемых компонентов:

|  |  |  |
| --- | --- | --- |
| **app\_name** | **container\_name / image\_name** | **container\_port** |
| DYNAMICDQ | dynamicdq | 8804 |
| MIDDLE\_LEVEL\_INTEGRATION | middle\_level\_integration | 8811 |
| ESB | esb | 9001 |
| COMMERCIAL | commercial | 8821 |

**Инструкция по развертыванию**

1. Установить службу контейнеризации на сервере server-api-be-01 при помощи следующих команд:

|  |
| --- |
| systemctl start docker systemctl enable docker dnf install docker-ce |

1. Скопировать заранее подготовленные файлы **docker image** с внешнего ресурса
2. Запустить **docker container** на основе полученного **docker image**

|  |
| --- |
| docker run -d -p <container\_port>:<container\_port> \-v /etc/hosts:/etc/hosts:ro,Z \-v /opt/<app\_name>\_HOME/logs:/opt/<app\_name>\_HOME/logs:rw,Z \-v /opt/FILES\_HOME:/opt/FILES\_HOME:rw,Z \--name <container\_name> <image\_name> |

1. Проверить запущенность контейнеров командами
	1. docker ps - проверка, что докер запустился
	2. tailf /opt/<app\_name>\_HOME/logs/<app\_name>.log - проверка, что приложения запустилось
	3. curl http://<host>:<port>/<app\_name>/actuator/health - дополнительное подтверждение, что приложение запустилось
	Ожидаемый ответ: {"status":"UP"}
2. Запустить веб-портал по адресу <http://192.168.0.2>

Альтернативным способом установки демо стенда будет являться импорт существующего виртуального сервера в существующую инфраструктуру (файлы виртуальной машины могут быть предоставлены по запросу).