

**Платформа разработки аналитических
приложений
Insight 2.0**

INSIGHT 2.0

**Подготовка сервера и установка приложений
на сервере**

ОГЛАВЛЕНИЕ

1. ОБЩАЯ ИНФОРМАЦИЯ.....	4
2. ПОДГОТОВКА.....	5
3. УСТАНОВКА И ЗАПУСК ANSIBLE	5
3.1. НАСТРОЙКА ХОСТОВ.....	6
4. ОНЛАЙН УСТАНОВКА КОМПОНЕНТОВ INSIGHT 2.0.....	8
4.1. УСТАНОВКА DREMIO	8
4.2. УСТАНОВКА АДМИН-АККАУНТА ПРИ ПЕРВОМ ЗАПУСКЕ.....	9
4.3. УСТАНОВКА NGINX	9
4.4. УСТАНОВКА MYSQL.....	9
4.5. УСТАНОВКА POSTGRESQL	9
4.6. POSTGRESQL - ДОБАВЛЕНИЕ ПОЛЬЗОВАТЕЛЕЙ И БД	9
4.7. УСТАНОВКА JENKINS	9
4.8. УСТАНОВКА KEYCLOAK	9
4.9. УСТАНОВКА APACHE NIFI.....	9
4.10. УСТАНОВКА DOCKER(DOCKER-COMPOSE).....	9
5. ОФФЛАЙН УСТАНОВКА КОМПОНЕНТОВ INSIGHT 2.0.....	10
5.1. УСТАНОВКА DREMIO.....	10
5.2. УСТАНОВКА NGINX	11
5.3. УСТАНОВКА MYSQL.....	11
5.4. УСТАНОВКА POSTGRESQL	11
5.5. УСТАНОВКА JENKINS	11
5.6. УСТАНОВКА KEYCLOAK	11
5.7. УСТАНОВКА APACHE NIFI.....	11
5.8. УСТАНОВКА DOCKER(DOCKER-COMPOSE).....	11
6. УСТАНОВКА DREMIO.....	12
6.1. ПОДГОТОВИТЕЛЬНЫЕ ШАГИ, ОБЩИЕ ДЛЯ ВСЕХ ВАРИАНТОВ РАЗВЕРТЫВАНИЯ	12
6.2. ОДНОНОДОВАЯ КОНФИГУРАЦИЯ DREMIO.	13
6.3. КОНФИГУРАЦИЯ DREMIO КООРДИНАТОР-ИСПОЛНИТЕЛЬ.	14
6.4. УСТАНОВКА АДМИН-АККАУНТА ПРИ ПЕРВОМ ЗАПУСКЕ.....	16

7.	УСТАHOБKA NGINX	16
8.	УСТАHOБKA MYSQL	17
9.	УСТАHOБKA POSTGRESQL	17
10.	УСТАHOБKA JENKINS	18
11.	УСТАHOБKA KEYCLOAK	19
12.	УСТАHOБKA APACHE NIFI	20
13.	УСТАHOБKA DOCKER(DOCKER-COMPOSE).....	20
14.	УСТАHOБKA INSIGHT 2.0	21

1. ОБЩАЯ ИНФОРМАЦИЯ

Программный комплекс Insight 2.0 является сложно организованной системой, для нормального функционирования которой требуется стопроцентное удовлетворение внешних зависимостей, настроек операционной системы и корректное указание значений переменных в конфигурационных файлах.

Ввиду специфики экосистемы Linux и разнообразия сборок даже одноименных дистрибутивов, возможны нюансы развертывания, которые не удалось осветить в данной инструкции.

В случае возникновения сложностей при развертывании, ошибок при запуске и прочих проблемах, необходимо обратиться по адресу электронной почты: devops@goodt.me

Insight 2.0 это продукт, объединяющий в себе **методологию и комплекс программных продуктов** как собственной разработки, так и Opensource решения.

В программный стек необходимый для работы, входит:

- Dremio;
- NGINX;
- MYSQL;
- PostgreSQL;
- Jenkins;
- Keycloak;
- Apache NIFI;
- Docker.

Установка сводится к следующим шагам:

1. Подготовка структуры папок;
2. Установка Ansible;
3. Установка Dremio;
4. Установка плагина goodt-dremio-plugin;
5. Установка SDK rtl-dremio-connector;
6. Установка rtl-goodteditor-player;
7. Запуск приложений;
8. Альтернативная установка без использования Docker:
 1. Подготовка структуры папок;
 2. Установка внешних зависимостей;
 3. Установка Dremio;
 4. Установка приложений.
9. Проверка работоспособности установленных приложений.

2. ПОДГОТОВКА

Плейбуки используемые для развертки хранятся в *git.goodt.me*

Файлы необходимые для установки компонентов хранятся в *art.goodt.me*

Важно!

При размещении плейбука в каталоге отличном от */ansible/playbooks/* внутри команды необходимо прописывать адрес директории где размещены плейбука

Пример команды с расположением плейбука */ansible/playbooks/*

```
ansible-playbook dremio_standalone.yml --extra-vars "mysecret=***** , ram=16"
```

Пример команды с расположением файла в */abc/playbooks/*

```
ansible-playbook -i /abc/playbooks/dremio_standalone.yml --extra-vars "mysecret=***** , ram=16"
```

Перед разверткой представленные плейбуки необходимо отредактировать, вписав в поле *hosts* хост или группу хостов, где будут произведены действия из плейбука
Значения доступных хостов для ansible, указываются в файле */etc/ansible/hosts*

При размещении файла *hosts* в каталоге отличном от */etc/ansible/hosts* внутри команд необходимо использовать опцию *-i hosts* где *hosts* адрес каталога в котором размещается файл

Пример команды с расположением файла в */etc/ansible/hosts*:

```
ansible-playbook dremio_standalone.yml --extra-vars "mysecret=***** , ram=16"
```

Пример команды с расположением файла в */abc/ansible/hosts*:

```
ansible-playbook -i /abc/ansible/hosts dremio_standalone.yml --extra-vars "mysecret=***** , ram=16"
```

3. УСТАНОВКА И ЗАПУСК ANSIBLE

Первым шагом необходимо подключить репозиторий

```
yum install epel-release
```

Установить ansible

yum install ansible

После базовой установки необходимо выполнить установку дополнительных компонентов и импорт файла *hosts* с помощью плейбука *ansible.yml*

Примеры типовых форматирований и именований:

- *mysecret* - пароль от хранилища артефактов *art.goodt.me*

Пример команды для онлайн установки:

```
ansible-playbook ansible.yml --extra-vars "mysecret=*****"
```

Пример команды для оффлайн установки:

```
ansible-playbook ansible.yml --extra-vars "offline=true"
```

Для запуска оффлайн установки, необходимо чтобы в папке */abc/distr* находились следующие файлы:

- *python-crypto-2.6.1-1.el7.rf.x86_64.rpm*
- *python-httplib2-0.9.2-1.el7.noarch.rpm*
- *python-jinja2-2.7.2-4.el7.noarch.rpm*
- *python-keyczar-0.71c-2.el7.noarch.rpm*
- *sshpas-1.05-1.el7.rf.x86_64.rpm*
- *ansible-2.9.27-1.el7.noarch.rpm*
- *ansible-playbook.tar*

3.1. НАСТРОЙКА ХОСТОВ

Перед началом работы необходимо указать хосты в файле *hosts*

Для того чтобы просмотреть/отредактировать содержание файла */etc/ansible/hosts* используйте команду

```
sudo nano /etc/ansible/hosts
```

Пример содержания импортируемого файла *hosts*, где

[*group-name*] - это наименование группы хоста

ip-адрес - это внешний или внутренний *ip* адрес хоста на которой установлен или будет установлен тот или иной компонент системы (группа может содержать 1 и более *ip* адресов).

hosts

[*dremio-master*]

ip-адрес

[dremio-executor]
ip-адрес VM

[dremio-fs]
ip-адрес

[pg]
ip-адрес

[keycloak]
ip-адрес

[zabbix]
ip-адрес

[nifi]
ip-адрес

[airflow]
ip-адрес

[editor]
ip-адрес

[jenkins]
ip-адрес

[dremio-connector]
ip-адрес

[player]
ip-адрес

[mysql]
ip-адрес

4. ОНЛАЙН УСТАНОВКА КОМПОНЕНТОВ INSIGHT 2.0

4.1. УСТАНОВКА DREMIO

1. Настройка Dremio под https

```
ansible-playbook dremio_html.yml --extra-vars "keystore_password=***** ,  
truststore_password=*** , key_passwd=*****"
```

2. Плейбук по первоначальному конфигурированию настроек dremio для проекта и Insight 2.0

```
ansible-playbook dremio_configure.yml -t insight -t dremio_user_add
```

3. Однонодовая конфигурация Dremio.

```
ansible-playbook dremio_standalone.yml --extra-vars "mysecret=***** , ram=16"
```

4. Конфигурация Dremio координатор-исполнитель.

Настройка файлового хранилища

```
ansible-playbook dremio_fs.yml --extra-vars "ip_coor=192.168.0.1 , ip_exec=192.168.0.2"
```

Установка координатора

```
ansible-playbook dremio_coor.yml --extra-vars "ip_fs=192.168.0.3 , mysecret=***** ,  
ram=16"
```

Установка исполнителя

```
ansible-playbook dremio_exec.yml --extra-vars "ip_fs=192.168.0.3 , ip_coor=192.168.0.1  
, mysecret=***** , ram=16"
```

Установка админ-аккаунта при первом запуске

```
ansible-playbook dremio_admin.yml --extra-vars "user=admin , password=*****"
```

5. Установка админ-аккаунта при первом запуске

```
ansible-playbook dremio_admin.yml --extra-vars "user=admin , password=*****"
```


4.2. УСТАНОВКА АДМИН-АККАУНТА ПРИ ПЕРВОМ ЗАПУСКЕ

```
ansible-playbook dremio_admin.yml --extra-vars "user=admin , password=*****"
```

4.3. УСТАНОВКА NGINX

```
ansible-playbook balanser_nginx.yml
```

4.4. УСТАНОВКА MYSQL

```
ansible-playbook mysql.yml --extra-vars " mysecret=***** , root_password=*****"
```

4.5. УСТАНОВКА POSTGRESQL

```
ansible-playbook -i hosts pgsql.yml --extra-vars "mysecret=***** , pg_pass=*****"
```

4.6. POSTGRESQL - ДОБАВЛЕНИЕ ПОЛЬЗОВАТЕЛЕЙ И БД

```
ansible-playbook pgsql_rtl_db.yml --extra-vars "pg_user_pass=*****"
```

4.7. УСТАНОВКА JENKINS

```
ansible-playbook jenkins.yml --extra-vars "mysecret=*****"
```

4.8. УСТАНОВКА KEYCLOAK

Параметры запуска плейбука находятся в vars/main.yml

Для ssl необходимо положить в директорию file/ актуальный сертификат (fullchain.crt) в pem формате, а также приватный ключ (private.key) к сертификату

```
ansible-playbook keycloak.yml
```

После запуска keycloak будет доступен по следующему пути:

<https://hostname:9443/auth/>

Создание realm через импорт.

Параметры запуска плейбука находятся в vars/main.yml

```
ansible-playbook keycloak-add-realm.yml
```

4.9. УСТАНОВКА APACHE NIFI

```
ansible-playbook nifi.yml
```

4.10. УСТАНОВКА DOCKER(DOCKER-COMPOSE)

```
ansible-playbook docker.yaml --extra-vars "mysecret=*****"
```

5. ОФФЛАЙН УСТАНОВКА КОМПОНЕНТОВ INSIGHT 2.0

5.1. УСТАНОВКА DREMIO.

Настройка Dremio под https

```
ansible-playbook dremio_html.yml --extra-vars "keystore_password=***** ,  
truststore_password=***** , key_passwd=*****"
```

Плейбук по первоначальному конфигурированию настроек dremio для проекта Insight 2.0

```
ansible-playbook dremio_configure.yml --extra-vars "offline=true"
```

Однонодовая конфигурация Dremio.

Создайте директорию */abc/distr*

```
mkdir -p /abc/distr
```

Поместите в данную директорию заранее подготовленные файлы:

- *dremio.tar.gz*
- *java.tar.gz*
- *dremio-telemetry.yaml*
- *dremio-telemetry-prometheus.jar*
- *goodt-dremio-plugins.jar*

Далее выполнить команду:

```
ansible-playbook dremio_standalone.yml --extra-vars "ram=16 , offline=true"
```

Конфигурация Dremio координатор-исполнитель.

Настройка файлового хранилища

```
ansible-playbook dremio_fs.yml --extra-vars "ram=16 , offline=true"
```

Установка координатора

```
ansible-playbook dremio_coor.yml --extra-vars "ip_fs=192.168.0.3 , ram=16 , offline=true"
```

Установка исполнителя

```
ansible-playbook dremio_exec.yml --extra-vars "ip_fs=192.168.0.3 , ip_coor=192.168.0.1 , ram=16 , offline=true"
```

Установка админ-аккаунта при первом запуске

```
ansible-playbook dremio_admin.yml --extra-vars "user=admin , password=*****"
```

5.2. УСТАНОВКА NGINX

```
ansible-playbook balancer_nginx.yml --extra-vars "offline=true"
```

5.3. УСТАНОВКА MYSQL

```
ansible-playbook mysql.yml --extra-vars " offline=true , root_password=*****"
```

5.4. УСТАНОВКА POSTGRESQL

```
ansible-playbook pgsql.yml --extra-vars "pg_pass=***** , offline=true"
```

PostgreSQL - добавление пользователей и БД

```
ansible-playbook pgsql_rtl_db.yml --extra-vars "pg_user_pass=*****"
```

5.5. УСТАНОВКА JENKINS

```
ansible-playbook jenkins.yml --extra-vars "offline=true"
```

5.6. УСТАНОВКА KEYCLOAK

Параметры запуска плейбука находятся в vars/main.yml

Для ssl необходимо положить в директорию file/ актуальный сертификат (fullchain.crt) в pem формате, а также приватный ключ (private.key) к сертификату

```
ansible-playbook keycloak.yml --extra-vars "offline=true, mysecret=*****"
```

После запуска keycloak будет доступен по следующему пути:

<https://hostname:9443/auth/>

Создание realm через импорт.

Параметры запуска плейбука находятся в vars/main.yml

```
ansible-playbook keycloak-add-realm.yml
```

5.7. УСТАНОВКА APACHE NIFI

```
ansible-playbook nifi.yml --extra-vars "offline=true"
```

5.8. УСТАНОВКА DOCKER(DOCKER-COMPOSE)

```
ansible-playbook docker.yaml --extra-vars "mysecret=*****, offline=true"
```

6. УСТАНОВКА DREMIO.

6.1. ПОДГОТОВИТЕЛЬНЫЕ ШАГИ, ОБЩИЕ ДЛЯ ВСЕХ ВАРИАНТОВ РАЗВЕРТЫВАНИЯ

Плейбук по конфигурированию настроек Dremio под https

Примеры типовых форматирований и именований:

- `offline` - ключ указывающий на тип установки (`true` - оффлайн установка)
- `artifactory` - путь для скачивания дистрибутивов
- `user` - логин от хранилища артефактов `art.goodt.me`
- `mysecret` - пароль от `art.goodt.me`
- `keystore_password` - пароль от хранилища `keystore.jks`
- `key_passwd` - пароль сертификата в хранилище ключей. Если не установлено, будет использоваться значение `keystore_password`.
- `truststore_password` - пароль от хранилища `truststore.jks`

Необходимо положить в директорию `file/` данного плейбука актуальные хранилища сертификатов (`keystore.jks`, `truststore.jks`)

Пример запуска установки:

```
ansible-playbook dremio_html.yml --extra-vars "keystore_password=*** ,  
truststore_password=*** , key_passwd=****"
```

Плейбук по первоначальному конфигурированию настроек dremio для проекта Insight 2.0

Параметры запуска плейбука находятся в `vars/main.yml`:

- `offline` - ключ указывающий на тип установки (`true` - оффлайн установка)
- `artifactory` - путь для скачивания дистрибутивов
- `user` - логин от хранилища артефактов `art.goodt.me`
- `mysecret` - пароль от `art.goodt.me`
- `dremio_url` - веб адрес, где установлен Dremio
- `dremio_login` - Логин админ пользователя Dremio, устанавливается при разворачивании Dremio
- `dremio_password` - пароль админ пользователя Dremio
- `user_login` - имя создаваемого пользователя в Dremio
- `user_password` - пароль создаваемого пользователя. Должен содержать не менее 8 символов, 1 цифры, 1 заглавной буквы
- `realm_name` - имя рилма, берется из Keycloak
- `postgres_host` - адрес базы данных, где находятся базы проекта
- `postgres_port` - порт подключения к базе данных

- postgres_user - пользователь в postgres имеющий доступ к базам проекта
- postgres_password - пароль пользователя в postgres

Следующие переменные заполняются только при расположении базы Keycloak на отдельном сервере БД

- postgres_kk_host - адрес базы данных, где находится база Keycloak
- postgres_kk_port - порт подключения к базе данных Keycloak
- postgres_kk_user - пользователь, имеющий доступ к бд Keycloak
- postgres_kk_password - пароль пользователя, имеющего доступ к бд Keycloak

Плейбук универсален, содержит в себе настройки проекта Insight 2.0, а также создание пользователя в Dremio

Имеется два тэга

- insight - производит настройку дремио для проекта Insight
- dremio_user_add - создание пользователя в Dremio

Пример запуска онлайн-установки и конфигурирования для проекта Insight 2.0 и создание пользователя:

```
ansible-playbook dremio_configure.yml -t insight -t dremio_user_add
```

Для запуска оффлайн установки, необходимо чтобы в папке /abc/distr/scripts/ находились следующие файлы:

- goodt-dremio-scripts.tar.gz

Пример запуска оффлайн-установки:

```
ansible-playbook dremio_configure.yml --extra-vars "offline=true"
```

6.2. ОДНОНОДОВАЯ КОНФИГУРАЦИЯ DREMIO.

Примеры типовых форматирований и именований:

- user - логин от хранилища артефактов art.goodt.me
- mysecret - пароль от хранилища артефактов art.goodt.me
- ram - количество оперативной памяти на ВМ (8,16,32,64)
- offline - ключ указывающий на тип установки (true - оффлайн установка)

Пример команды для онлайн установки конфигурации:

```
ansible-playbook dremio_standalone.yml --extra-vars "mysecret=*****, ram=16"
```

Требования для оффлайн установки

Для запуска оффлайн установки, необходимо чтобы в папке `/abc/distr` находились следующие файлы:

- `dremio.tar.gz`
- `jdk-11.0.9_linux-x64_bin.tar.gz`
- `dremio-telemetry.yaml`
- `dremio-telemetry-prometheus.jar`
- `goodt-dremio-plugins.jar`

Далее выполнить команду установки оффлайн:

```
ansible-playbook dremio_standalone.yml --extra-vars "ram=16 , offline=true"
```

6.3. КОНФИГУРАЦИЯ DREMIO КООРДИНАТОР-ИСПОЛНИТЕЛЬ.

Данная конфигурация состоит минимум из трех узлов:

1. Файловое хранилище
2. Узел-координатор
3. Узел-исполнитель

Откройте для редактирования файл переменных `/etc/environment`

```
sudo nano /etc/environment
```

Внесите значения переменных для хостов используемых при развертке согласно примеру:

```
ip_coor=192.168.0.1  
ip_exec=192.168.0.2  
ip_fs=192.168.0.3
```

Сохраните используя комбинацию клавиш `Ctrl+X`

Настройка файлового хранилища

Примеры типовых форматирований и именовании:

- `ip_coor` - IP адрес координатора
- `ip_exec` - IP адрес исполнителя

Пример команды для онлайн установки:

```
ansible-playbook dremio_fs.yml --extra-vars "ip_coor=192.168.0.1 , ip_exec==192.168.0.2"
```

Для запуска оффлайн установки, необходимо чтобы в папке /abc/distr находились следующие файлы

- *nfs-utils-1.3.0-0.68.el7.x86_64.rpm*

Пример команды для оффлайн установки:

```
ansible-playbook dremio_fs.yml --extra-vars "ram=16 , offline=true"
```

Установка координатора

Примеры типовых форматирований и именований:

- `ip_fs` - IP адрес файлового хранилища

Пример команды для онлайн установки:

```
ansible-playbook dremio_coor.yml --extra-vars "ip_fs==192.168.0.3 , mysecret= *****, ram=16"
```

Для запуска оффлайн установки, необходимо чтобы в папке /abc/distr находились следующие файлы:

- *dremio.tar.gz*
- *jdk-11.0.9_linux-x64_bin.tar.gz*
- *dremio-telemetry.yaml*
- *dremio-telemetry-prometheus.jar*
- *goodt-dremio-plugins.jar*
- *nfs-utils-1.3.0-0.68.el7.x86_64.rpm*

Пример команды для оффлайн установки:

```
ansible-playbook dremio_coor.yml --extra-vars "ip_fs==192.168.0.3 , ram=16 , offline=true"
```

Установка исполнителя

Примеры типовых форматирований и именований:

- `ip_fs` - IP адрес файлового хранилища
- `ip_coor` - IP адрес координатора

Пример команды для онлайн установки:

```
ansible-playbook dremio_exec.yml --extra-vars "ip_fs==192.168.0.3 , ip_coor==192.168.0.1 , mysecret=***** , ram=16"
```

Для запуска оффлайн установки, необходимо чтобы в папке /abc/distr находились следующие файлы:

- *dremio.tar.gz*
- *jdk-11.0.9_linux-x64_bin.tar.gz*
- *dremio-telemetry.yaml*
- *dremio-telemetry-prometheus.jar*
- *goodt-dremio-plugins.jar*
- *nfs-utils-1.3.0-0.68.el7.x86_64.rpm*

Пример команды для оффлайн установки:

```
ansible-playbook dremio_exec.yml --extra-vars "ip_fs==192.168.0.3 , ip_coor==192.168.0.1 , ram=16 , offline=true"
```

6.4. УСТАНОВКА АДМИН-АККАУНТА ПРИ ПЕРВОМ ЗАПУСКЕ

Примеры типовых форматирований и именовании:

- `dremio_user` - имя учетной записи
- `password` - пароль

Пример команды для установки:

```
ansible-playbook dremio_admin.yml --extra-vars "user=admin , password=*****"
```

7. УСТАНОВКА NGINX

Примеры типовых форматирований и именовании:

- `offline` - ключ указывающий на тип установки (`true` - оффлайн установка)
- `DREMIO_CONNECTOR_URL` - `proxy_pass` в конфиге `dremio_connector.vonf nginx`
- `FILEUPLOAD_URL` - `proxy_pass` в конфиге `fileupload.conf nginx`
- `KEYCLOAK_URL` - `proxy_pass` в конфиге `keycloak.conf nginx`
- `LEARNING_URL` - `proxy_pass` в конфиге `learning.conf nginx`
- `MONITOR_URL` - `proxy_pass` в конфиге `monitor.conf nginx`
- `ORGSTRUCTURE_URL` - `proxy_pass` в конфиге `orgstructure.conf nginx`
- `QUIZ_URL` - `proxy_pass` в конфиге `quiz.conf nginx`
- `ROSTALENT_URL` - `proxy_pass` в конфиге `rostalent.conf nginx`
- `TASKSETTING_URL` - `proxy_pass` в конфиге `tasksetting.conf nginx`
- `SERVER_NAME`: `server_name` в конфиге `ssl.conf nginx`

Пример команды для онлайн установки:

```
ansible-playbook balanser_nginx.yml
```

Для запуска оффлайн установки, необходимо чтобы в папке */abc/distr* находились следующие файлы:

- *nginx-1.20.1-1.el7.ngx.x86_64.rpm*

Пример команды для оффлайн установки:

```
ansible-playbook balancer_nginx.yml --extra-vars "offline=true"
```

8. УСТАНОВКА MYSQL

Примеры типовых форматирований и именований:

- *user* - логин от хранилища артефактов *art.goodt.me*
- *mysecret* - пароль от хранилища артефактов *art.goodt.me*
- *root_password* - задать пароль *root* пользователя *MySQL* (8 символов, минимум 1 цифра, 1 заглавная буква, 1 спец.символ)

Пример команды для онлайн установки:

```
ansible-playbook mysql.yml --extra-vars " mysecret=***** , root_password=*****"
```

Для запуска оффлайн установки, необходимо чтобы в папке */abc/distr* находились следующие файлы:

- *mysql-5.7.38-1.el7.x86_64.rpm-bundle.tar*
- *MySQL-python-1.2.5-1.el7.x86_64.rpm*

Пример команды для оффлайн установки:

```
ansible-playbook mysql.yml --extra-vars " offline=true , root_password=*****"
```

9. УСТАНОВКА POSTGRESQL

Примеры типовых форматирований и именований:

- *user* - логин от хранилища артефактов *art.goodt.me*
- *mysecret* - пароль от хранилища артефактов *art.goodt.me*
- *pg_pass* - пароль для пользователя *postgres*
- *offline* - ключ указывающий на тип установки (*true* - оффлайн установка)

Пример команды для онлайн установки:

```
ansible-playbook -i hosts pgsql.yml --extra-vars "mysecret=***** , pg_pass=*****"
```

Для запуска оффлайн установки, необходимо чтобы в папке `/abc/distr` находились следующие файлы:

- `postgresql11-11.15-1PGDG.rhel7.x86_64.tar.gz`

Пример команды для оффлайн установки:

```
ansible-playbook pgsql.yml --extra-vars "pg_pass=***** , offline=true"
```

PostgreSQL - добавление пользователей и БД

Примеры типовых форматирований и именований:

- `pg_user_pass` - переменная часть пароля пользователя

Пример запуска заполнения PostgreSQL:

```
ansible-playbook pgsql_rtl_db.yml --extra-vars "pg_user_pass=*****"
```

10. УСТАНОВКА JENKINS

Примеры типовых форматирований и именований:

- `user` - логин от хранилища артефактов `art.goodt.me`
- `mysecret` - пароль от хранилища артефактов `art.goodt.me`

Пример команды для онлайн установки:

```
ansible-playbook jenkins.yml --extra-vars "mysecret=*****"
```

Для запуска оффлайн установки, необходимо чтобы в папке `/abc/distr` находились следующие файлы:

- `jenkins.tar`

Пример команды для оффлайн установки:

```
ansible-playbook jenkins.yml --extra-vars "offline=true"
```

11. УСТАНОВКА KEYCLOAK

Параметры запуска плейбука находятся в `vars/main.yml`

Примеры типовых форматирований и именований:

- *offline* - ключ указывающий на тип установки (true - оффлайн установка)
- *artifactory* - путь для скачивания дистрибутивов
- *mysecret* - пароль от art.goodt.me
- *keycloak_host* - имя сервера, где будет установлен Keycloak
- *db_host* - имя сервера баз данных
- *db_name* - имя базы для Keycloak
- *db_user* - имя пользователя, под которым будет выполнено подключение к БД
- *db_pass* - пароль пользователя, под которым будет выполнено подключение к БД
- *keycloak_user* - имя пользователя, под которым будет выполнено подключение к Keycloak
- *keycloak_pass* - пароль пользователя, под которым будет выполнено подключение к Keycloak

Для ssl необходимо положить в директорию `file/` актуальный сертификат (`fullchain.crt`) в pem формате, а также приватный ключ (`private.key`) к сертификату

Пример команды для онлайн установки:

```
ansible-playbook keycloak.yml
```

Для запуска оффлайн установки, необходимо чтобы в папке `/abc/distr` находились следующие файлы:

- *keycloak-18.0.2.tar.gz*
- *jdk-11.0.9_linux-x64_bin.tar*

Пример команды для оффлайн установки:

```
ansible-playbook keycloak.yml --extra-vars "offline=true, mysecret=*****"
```

После запуска keycloak будет доступен по следующему пути:

<https://hostname:9443/auth/>

Создание realm через импорт.

Параметры запуска плейбука находятся в `vars/main.yml`

- *realm_name* - название realm

Пример команды для импорта:

```
ansible-playbook keycloak-add-realm.yml
```

12. УСТАНОВКА APACHE NIFI

Параметры запуска плейбука находятся в vars/main.yml:

- *offline* - ключ указывающий на тип установки (true - оффлайн установка)
- *artifactory* - путь для скачивания дистрибутивов
- *user* - логин от хранилища артефактов art.goodt.me
- *mysecret* - пароль от art.goodt.me
- *fqdn_nifi* - адрес или полное доменное имя куда разворачиваем Apache NIFI
- *nifi_port* - порт на котором будет разворачиваться Apache NIFI
- *JVM_memory_max* - Объем максимальной памяти выделяемый для java
- *nifi_user* - имя создаваемого пользователя в Apache NIFI
- *nifi_password* - пароль создаваемого пользователя. Должен содержать не менее 12 символов
- *keystore_password* - пароль от хранилища keystore.jks
- *key_passwd* - пароль сертификата в хранилище ключей. Если не установлено, будет использоваться значение keystore_password.
- *truststore_password* - пароль от хранилища truststore.jks

Пример команды для онлайн установки:

```
ansible-playbook nifi.yml
```

Для запуска оффлайн установки, необходимо чтобы в папке /abc/distr/ находились следующие файлы:

- jdk-8u301-linux-x64.tar.gz
- nifi-1.15.3-bin.tar.gz

Пример команды для оффлайн установки:

```
ansible-playbook nifi.yml --extra-vars "offline=true"
```

Для ssl необходимо положить в директорию file/ актуальные хранилища сертификатов (keystore.jks, truststore.jks)

13. УСТАНОВКА DOCKER(DOCKER-COMPOSE)

Параметры запуска плейбука находятся в vars/main.yml:

- *offline* - ключ указывающий на тип установки (true - оффлайн установка)
- *mysecret* - ключ указывает пароль для доступа на скачивание архива с пакетами
- *DREMIO_CONNECTOR_VER* - версия дремио коннектора
- *FILEUPLOAD_VER* - версия апп fileupload
- *INS_EDITOR_VER* - версия эдатора

Пример команды для онлайн установки:

```
ansible-playbook docker.yaml --extra-vars "mysecret=*****"
```

Для запуска оффлайн установки, необходимо чтобы в папке */abc/distr/docker* находился архив с пакетами:

- *docker-pack.tar*

Пример команды для оффлайн установки:

```
ansible-playbook docker.yaml --extra-vars "mysecret=*****, offline=true"
```

При оффлайн установки приложений через Docker необходимо разместить образы для запуска приложений в каталог */abc/distr/docker/images*

14. УСТАНОВКА INSIGHT 2.0

Перед началом установки компонентов Insight 2.0 просмотрите доступные версии релизов размещенных в art.goodt.me и убедитесь в наличии необходимой версии для установки.

Для запуска чарта необходимо выполнить bash скрипт со следующим содержанием

```
helm install insight goodt/insight \  
--namespace=test-insight \  
--set version=release_ins_1.0b2 \  
--set host.name=$app_host \  
--set imageCredentials.username= \  
--set imageCredentials.password= \  
--set mysql.enabled=false \  
--set env.RTL_DREMIO_HTTP_URL=https://$dremio_host:9047/ \  
--set env.RTL_DREMIOCONNECTOR_URL=https://$app_host/rtl-api/dremio-connector \  
--set dremioconnector.values.RTL_DREMIO_USER=dremio_connector \  
--set dremioconnector.values.RTL_DREMIO_PASS= \  
--set dremioconnector.values.RTL_DREMIO_JDBC_URL= \  
--set rtlgoodteditorplayer.values.UEP_API_URL=https://$app_host/api/ \  
--set rtlgoodteditorplayer.values.UEP_PLAYER_URL=https://$app_host/player/ \  
--set rtlgoodteditorplayer.values.UEP_SUP_DB_HOST=$ip_mysql_db \  

```

```
--set rtlgoodteditorplayer.values.UEP_SUP_DB_PASS= \
--set rtlgoodteditorplayer.values.UEP_SUP_DB_NAME=sup_editor
```

Не забывайте поставить обратный слэш на предыдущей строке

Задать версии компонентов можно с помощью дополнения команды выше: `-f components_version.yaml` Где `omponents_version.yaml` подготовленный файл вида:

```
dremioconnector:
  version: release_ins_1.0b4
rtlfileupload:
  version: release_ins_1.0b4
rtlgoodteditorplayer:
  version: release_ins_1.1b0
```

Так же файл можно сформировать скриптом `set_components_vers_from_art_repo_file.sh` расположенным в репозитории `ansible-scripts`

Таблица 1: Описание параметров скрипта запуска чарта Insight 2.0:

Name	Description	Value
env.RTL_DREMIO_HTTP_URL	URL для подключения к Dremio веб-интерфейсу	https://\$dremio_host:9047/
env.RTL_DREMIOCONNECTOR_URL	URL для подключения к Dremio Connector компоненту	https://\$app_host/rtl-api/dremio-connector
dremioconnector.values.RTL_DREMIO_USER	Пользователь для подключения к БД Dremio	-
dremioconnector.values.RTL_DREMIO_PASS	Пароль пользователя для подключения к БД Dremio	-
dremioconnector.values.RTL_DREMIO_JDBC_URL	URL для подключения к Dremio	dremio_host:port
dremioconnector.values.RTL_DREMIO_PASS	Номер порта компонента Dremio Connector	4000

Name	Description	Value
dremioconnector.values.RTL_DREMIO_JDBC_URL	RSA ключ для подключения Dremio Connector	rostalent.rsa
dremioconnector.values.RTL_APP_PORT	Enable Disable параметр для авторизации по ключу	false
dremioconnector.values.RTL_KEY_RS	URL для подключения к WFM	http://localhost
dremioconnector.values.RTL_DREMIO_AUTH	Enable Disable параметр для авторизации по ключу	http://localhost
mysql.enabled	Включить развертывание mysql БД в виде контейнера(бета)	false
mysql.auth.password	Задать пароль для пользователя sup_editor	-
rtlgoodteditorplayer.values.UEP_SUP_DB_HOST	IP адрес для подключения к БД Editor компонента	mysql_db_host
rtlgoodteditorplayer.values.UEP_SUP_DB_NAME	Имя БД Editor компонента	sup_editor
rtlgoodteditorplayer.values.UEP_SUP_DB_PASS	Пароль пользователя sup_editor для БД Editor компонента	-

Внимание!

UEP_SUP_DB_HOST на данный момент, ввиду тех. особенностей миграции, работает только в виде IP адреса.

Для установки параметров Keycloak необходимо дополнительно указать в bash скрипт следующие строки:

```
--set env.RTL_PROFILE=prod \  
--set env.RTL_KCLK_URL= \  
--set env.RTL_KCLK_REALM= \  
--set env.RTL_KCLK_USER= \
```

```

--set env.RTL_KCLK_PASS= \
--set env.RTL_KCLK_CLID= \
--set env.RTL_KCLK_CLSECRET= \
--set env.RTL_USE_MINIO=false \
--set env.RTL_USE_FILESYSTEM=true \
--set env.RTL_USE_WEBDAV=false

```

Не забывайте поставить обратный слэш на предыдущей строке

Таблица 2: Описание параметров keycloak:

Name	Description	Value
env.RTL_PROFILE	Профиль конфигурации	prod
env.RTL_KCLK_URL	URL ссылка для использования Keycloak	https://keycloak_url
env.RTL_KCLK_REALM	-	-
env.RTL_KCLK_USER	Имя пользователя, под которым будет выполнено подключение к Keycloak	-
env.RTL_KCLK_PASS	Пароль пользователя, под которым будет выполнено подключение к Keycloak	-
env.RTL_KCLK_CLID	-	-
env.RTL_KCLK_CLSECRET	-	-
env.RTL_USE_MINIO	-	-
env.RTL_USE_FILESYSTEM	-	-
env.RTL_USE_WEBDAV	-	-

Текущая версия helm chart предназначена для для использования DNS имен вида *.goodt.me и использует SSL сертификат для данных имен.

Текущая версия helm chart по умолчанию использует SSO авторизацию через сервер <https://goodt-auth.goodt.me:9443>

По умолчанию, для Insight 2.0, включены компоненты *editor-player*, *dremio-connector*, *fileupload*. Для отключения данных компонентов можно использовать модификаторы

```

--set dremioconnector.enabled=false
--set rtlfileupload.enabled=false

```

При необходимости удаления Insight 2.0 выполните:

```
helm uninstall insight --namespace=test-insight
```