UNLIMITED PRODUCTION

eXpress
Система коммуникаций

Руководство администратора

Обновление

Сборка 211.0
24.04.2023

ОГЛАВЛЕНИЕ

РУЧНОЕ ОБНОВЛЕНИЕ

Внимание! Выполните резервное копирование перед выполнением процедуры обновления!

SINGLE CTS

Обновление сервера VoEx:

1. Перейдите в директорию Express cd /opt/express-voice/.
2. Остановите сервисы из директории Express:

```
DEPLOYKA_SKIP_UPDATE=true dpl --dc stop
```

3. Запустите обновление:

```
dpl -d
```

4. Проверьте логи на наличие ошибок командой:

```
dpl --dc logs --tail=200 -f
```

Для обновления системы:

Примечание:

- 22.07.2022 Начиная с версии 2.4 изменились минимальные требования по версии postgres. Теперь поставляется образ postgresql версии 14.4. Процедура обновления встроенной БД описана на стр. 8.
- 15.12.2022. Перед обновлением сервера на версию 2.6 настройте Kafka. Процедура настройки описана на стр. 17.

1. Перейдите в директорию Express cd /opt/express/.
2. Остановите сервисы из директории Express:

```
DEPLOYKA_SKIP_UPDATE=true dpl --dc stop
```

3. Выполните резервное копирование /var/lib/docker/volumes (или где они лежат в этой системе).

   При выполнении обновления сервера с версии 1.28 измените хозяина файлов для сервисов (делается один раз):

```
docker volume inspect --format '{{ .Mountpoint }}'
cts_ccs_admin_public cts_file_service_uploads
cts_messaging_cache cts_messaging_uploads cts_phonebook_uploads
| xargs sudo chown -R 888:888
```

   Если предыдущая версия nginx меньше, чем 1.20.1, и используются letsencrypt сертификаты:

   - Очистите хранилище letsencrypt (один раз):

```
rm -rf cts/letsencrypt
dpl cadvinstall && dpl nxinstall
```

4. Обновите node exporter и container advisor:

```
dpl cadvinstall && dpl nxinstall
```

5. Запустите обновление:

```
dpl -d
```

   После запуска обновления требуется время на проведение внутренних процедур (ориентировочное время 10-15 минут).

6. Проверьте логи на наличие ошибок командой:

```
dpl --dc logs --tail=200 -f
```

Внимание! С версий 2.2 и 2.3 откатываться назад нельзя!

Для отката обновления поправьте файл **settings**, указав параметр, например:

```
images:
     trusts: ccs/trusts:1.28.0
```

FRONT CTS И BACK CTS

Обновление сервера VoEx:

1. Перейдите в директорию Express cd /opt/express-voice/.
2. Остановите сервисы из директории Express:

```
DEPLOYKA_SKIP_UPDATE=true dpl --dc stop
```

3. Запустите обновление:

```
dpl -d
```

4. Проверьте логи на наличие ошибок командой:

```
dpl --dc logs --tail=200 -f
```

Внимание! Перед началом процедуры обновления проверьте изменения по таблице сетевого взаимодействия!

Примечание:

- 05.03.2022 Для установок с разделением на frontend и backend нужно убедиться, что с frontend сервера доступны tcp порты 2379, 5432, 6379, 9092 на сервере backend. Также желательно закрыть доступ к этим портам отовсюду кроме frontend.
- 22.07.2022 **Начиная с версии 2.4 изменились минимальные требования по версии postgres. Теперь поставляется образ postgresql версии 14.4.** Процедура обновления встроенной БД описана на стр. 8.
- 15.12.2022. Перед обновлением сервера на версию 2.6 настройте Kafka. Процедура настройки описана на стр. 17.

Первым обновляется сервер Front CTS, затем сервер Back CTS.

Для обновления севера Front CTS:

1. Запустите командную строку.
2. Остановите работу приложения командой:

```
DEPLOYKA_SKIP_UPDATE=true dpl --dc stop
```

3. Выполните резервное копирование файлов /var/lib/docker/volumes (после нескольких дней эксплуатации скопированные файлы можно удалить).

   Если предыдущая версия nginx меньше, чем 1.20.1, и используются letsencrypt сертификаты:

   - Очистите хранилище letsencrypt (один раз):

     ```
     rm -rf cts/letsencrypt
     ```

4. Запустите обновление:

```
dpl -d
```

После запуска обновления требуется время на проведение внутренних процедур (ориентировочное время 10-15 минут).

5. Проверьте логи на наличие ошибок командой:

```
dpl --dc logs --tail=200 -f
```

Для обновления севера Back CTS:

1. Запустите командную строку.
2. Остановите работу приложения командой:

```
DEPLOYKA_SKIP_UPDATE=true dpl --dc stop
```

3. Выполните резервное копирование файлов /var/lib/docker/volumes (после нескольких дней эксплуатации скопированные файлы можно удалить).
4. Если версия сервера ниже 1.28, выполните:

```
docker volume inspect --format '{{ .Mountpoint }}'
cts_ccs_admin_public \ cts_file_service_uploads
cts_messaging_cache cts_messaging_uploads \
cts_phonebook_uploads | xargs sudo chown -R 888:888
dpl cadvinstall && dpl nxinstall
```

5. Запустите обновление:

```
dpl -d
```

После запуска обновления требуется время на проведение внутренних процедур (ориентировочное время 10-15 минут).

6. Проверьте логи на наличие ошибок командой:

```
dpl --dc logs --tail=200 -f
```

ОТКАЗОУСТОЙЧИВАЯ КОНФИГУРАЦИЯ

В случае невозможности использования скриптов автоматического обновления выполните обновление в ручном режиме.

Для обновления отказоустойчивой конфигурации:

1. Скопируйте новые версии образов docker и скрипт загрузки образов в каталог /tmp/images и загрузите образы с помощью скрипта load.sh:

```
cp *.tar /tmp/images/
cp /opt/deploy/script/load.sh /tmp/images/
cd /tmp/images/
./load.sh
```

2. Подключитесь к консоли сервера Back и Front кластера с индексом 01 и 02.
3. Выполните команду для остановки антивируса:

```
systemctl stop kesl klnagent64
```

Примечание. В случае зависания сервера при остановке антивируса, перезагрузите оба узла кластера через систему виртуализации.

4. Выполните команду:

```
pcs status
```

5. Убедитесь, что ресурсы кластера запущены согласно списку ниже:
   - dlm-clone [dlm] (back кластер) – запущен на обоих узлах кластера;
   - clvmd-clone [clvmd] (back кластер) – запущен на обоих узлах кластера;

- clusterfs-clone [clusterfs] (back кластер) – запущен на обоих узлах кластера;
- cluster_ip – запущен на одном узле кластера;
- dockerd – запущен на одном узле кластера;
- node_exporter (back кластер) – запущен на одном узле кластера;
- cadvisor (back кластер) – запущен на одном узле кластера;
- vmfence (back кластер) – запущен на одном узле кластера.

Если статус ресурсов кластера не соответствует перечисленным выше, выполните команду, заменив *resource_name* на имя проблемного ресурса:

```
pcs resource cleanup resource_name
```

6. На узлах кластера Back с индексом 01 и 02 выполните команду ниже:

```
ls -la /opt/ex_data/files
```

Примечание. В случае зависание вывода списка директорий необходимо перезагрузить оба узла кластера через систему виртуализации.

7. Подключитесь к консоли сервера Back кластера с индексом 01 или 02 и выполните команду:

```
pcs status | grep dockerd
```

Примечание. Команда выполняется для определения текущего первичного узла, на котором запущены ресурсы кластера.

8. Подключитесь к консоли текущего первичного узла кластера Back и последовательно выполните команды:

```
cd /opt/express
dpl -g
```

9. Подключитесь к консоли сервера Front кластера с индексом 01 или 02 и выполните команду:

```
pcs status | grep dockerd
```

Примечание. Команда выполняется для определения текущего первичного узла, на котором запущены ресурсы кластера.

10. Подключитесь к консоли текущего первичного узла кластера Front и последовательно выполните команды:

```
cd /opt/express
dpl -g
cd /opt/express-voice
dpl -g
```

11. Подключитесь к консоли вторичного узла кластера Back и последовательно выполните команды, заменив full_fqdn_slave_server на полное доменное имя вторичного узла кластера:

```
pcs resource move cluster_ip full_fqdn_slave_server
pcs resource move dockerd full_fqdn_slave_server
```

12. Дождитесь переключения ресурсов кластера dockerd и cluster_ip на вторичный узел кластера Back. Для мониторинга состояния ресурсов периодически выполняйте команду:

```
pcs status
```

13. После переключения ресурсов на вторичный узел кластера Back последовательно выполните команды:

```
cd /opt/express
dpl -g
```

14. Подключитесь к консоли вторичного узла кластера Front и последовательно выполните команды, заменив full_fqdn_slave_server на полное доменное имя вторичного узла кластера:

```
pcs resource move cluster_ip full_fqdn_slave_server
pcs resource move dockerd full_fqdn_slave_server
```

15. Дождитесь переключения ресурсов кластера dockerd и cluster_ip на вторичный узел кластера Front. Для мониторинга состояния ресурсов периодически выполняйте следующую команду:

```
pcs status
```

16. После переключения ресурсов на вторичный узел кластера Front последовательно выполните команды:

```
cd /opt/express
dpl -g
cd /opt/express-voice
dpl -g
```

17. Подключитесь к консоли текущего первичного узла кластера Back и последовательно выполните команды:

```
cd /opt/express
dpl --dc stop
dpl nxinstall && dpl cadvinstall
dpl -d
```

18. После завершения обновления сервера откройте вывод логов работы контейнеров:

```
dpl --dc logs --tail=100 -f
```

19. Дождитесь остановки вывода логов контейнеров кроме контейнера nginx.
20. Подключитесь к консоли сервера Front кластера с индексом 01 или 02 и выполните команду:

```
pcs status | grep dockerd
```

Примечание. Команда выполняется для определения текущего первичного узла, на котором запущены ресурсы кластера.

21. Подключитесь к консоли текущего первичного узла кластера Front и последовательно выполните команды:

```
cd /opt/express
dpl --dc stop
dpl -d
cd /opt/express-voice
dpl --dc stop
dpl -d
```

22. После обновления первичных узлов кластеров Front и Back проверьте функционирование системы, выполните проверку логов на ошибки и функции отправки сообщений.
23. Подключитесь к консоли вторичного узла кластера Back и последовательно выполните команды, заменив full_fqdn_slave_server на полное доменное имя вторичного узла кластера:

```
pcs resource move cluster_ip full_fqdn_slave_server
pcs resource move dockerd full_fqdn_slave_server
```

24. Дождитесь переключения ресурсов кластера dockerd и cluster_ip на вторичный узел кластера Back. Для мониторинга состояния ресурсов периодически выполняйте команду:

```
pcs status
```

25. После переключения ресурсов на вторичный узел кластера Back последовательно выполните команды:

```
cd /opt/express
dpl --dc stop
dpl nxinstall && dpl cadvinstall
dpl -d
```

26. После завершения обновления сервера, откройте вывод лога работы контейнеров и дождитесь остановки вывода логов контейнеров кроме контейнера nginx:

```
dpl --dc logs --tail=100 -f
```

27. Подключитесь к консоли вторичного узла кластера Front и последовательно выполните команды, заменив full_fqdn_slave_server на полное доменное имя вторичного узла кластера:

```
pcs resource move cluster_ip full_fqdn_slave_server
pcs resource move dockerd full_fqdn_slave_server
```

28. Дождитесь переключения ресурсов кластера dockerd и cluster_ip на вторичный узел кластера Front. Для мониторинга состояния ресурсов периодически выполняйте команду:

```
pcs status
```

29. После переключения ресурсов на вторичный узел кластера Front последовательно выполните команды:

```
cd /opt/express
dpl --dc stop
dpl -d
cd /opt/express-voice
dpl --dc stop
dpl -d
```

30. Подключитесь к консоли сервера Back и Front кластера с индексом 01 и 02, выполните команду для запуска антивируса:

```
systemctl start kesl klnagent64
```

ОБНОВЛЕНИЕ POSTGRESQL

Важно! Существует риск потери данных в момент обновления БД. Процедуру обновления рекомендуется выполнять во время минимальной пользовательской активности или остановки всех сервисов. Настоятельно рекомендуется сделать бэкап базы данных.

ПРОЦЕДУРА ОБНОВЛЕНИЯ

Перед обновлением встроенной БД:

1. Убедитесь в наличии свободного места на диске.

Для обновления требуется дисковое пространство, равное по размеру существующей базе. Текущий размер базы можно узнать с помощью команды:

```
docker system df -v | grep postgres_data
```

2. Уточните текущую версию PostgreSQL.

   Если после обновления возникнут ошибки, может потребоваться откат версии PostgreSQL. Поэтому желательно знать, на какой версии БД работала до обновления. Важно выполнить команду вывода текущей версии БД до обновления образа PostgreSQL:

   ```
   DEPLOYKA_SKIP_UPDATE=true dpl --dc exec postgres cat
   /var/lib/postgresql/data/PG_VERSION
   ```

3. Убедитесь, что имеется доступ к apt.postgresql.org. Доступ требуется для загрузки исполняемых файлов старой версии. При отсутствии доступа см. раздел «Обновление без доступа к apt.postgresql.org».

   Чтобы откатить версию базы, нужно перенести файлы из директории с резервной копией в директорию уровнем выше.

4. В конец файла settings добавьте настройку, запускающую обновление контейнера БД при старте:

   ```
   postgres_upgrade: true
   ```

   Примечание. Если данной настройки не будет, новые версии образа PostgreSQL при запуске будут выдавать ошибку «postgres_upgrade disabled, exiting».

5. Выполните обновление образа PostgreSQL и его запуск с помощью команды:

   ```
   dpl -d postgres
   ```

6. Для отслеживания процесса обновления в логах используйте команду:

   ```
   dpl --dc logs -f --tail=1 postgres
   ```

   Когда обновление будет закончено в логах появится сообщение:

   ```
   DB upgrade is done, please disable postgres_upgrade in the
   settings
   ```

7. Отключите настройку postgres_upgrade, иначе контейнер с PostgreSQL не запустится.
8. Уберите настройку postgres_upgrade и загрузите стандартный образ с помощью команды:

   ```
   DEPLOYKA_SKIP_UPDATE=true dpl -d postgres
   ```

Для автоматического обновления воспользуйтесь настройкой postgres_auto_upgrade, с ней база будет автоматически обновляться при выпуске образа с новой версией.

Если версию БД по каким-то причинам нужно оставить без изменений, остаются доступны образы postgres без скрипта обновления. Их можно включить через параметр images, например:

```
images:

  postgres: postgres:9.5.24
```

РЕЗЕРВНАЯ КОПИЯ

После обновления базы старая версия сохраняется для возможности отката версии. Копия расположена в том же volume, который использует контейнер postgres. В директории с именем upgrade_backup_<timestamp>.

Если по итогам обновления все сервисы работают нормально, и откат не требуется, удалите резервную копию командой:

```
DEPLOYKA_SKIP_UPDATE=true dpl --dc exec postgres find
/var/lib/postgresql/data -type d -name upgrade_backup_* -exec rm -r
{} \;
```

ОТКАТ ВЕРСИИ

Чтобы откатить версию базы, нужно перенести файлы из директории с резервной копией в директорию уровнем выше.

Для работы с файловой системой volume запустите временный контейнер:

```
DEPLOYKA_SKIP_UPDATE=true dpl --dc run --rm --entrypoint=/bin/bash
postgres
```

Или выполните операцию непосредственно с хоста. Docker volumes обычно хранятся в /var/lib/docker/volumes.

После переноса файлов выставите образ postgres предыдущей версии в файле в settings, например:

```
images:

  postgres: postgres:9.5.24
```

ОБНОВЛЕНИЕ БЕЗ ДОСТУПА К APT.POSTGRESQL.ORG

Для обновления без доступа к apt.postgresql.org собран специальный образ 14.4-from-9.5.

Примечание. Поддерживается только обновление с версии 9.5.

Для обновления БД до версии 14.4:

1. В файл settings добавьте секцию images с соответствующим тэгом:

   ```
   images:
     postgres: postgres:14.4-from-9.5
   ```

2. В конец файла settings добавьте настройку, запускающую обновление контейнера БД при старте:

   ```
   postgres_upgrade: true
   ```

   Примечание. Если данной настройки не будет, новые версии образа PostgreSQL при запуске будут выдавать ошибку «postgres_upgrade disabled, exiting».

3. Выполните обновление образа PostgreSQL и его запуск с помощью команды:

   ```
   dpl -d postgres
   ```

4. Чтобы отследить процесс обновления в логах, используйте команду:

   ```
   dpl --dc logs -f --tail=1 postgres
   ```

   Когда обновление будет закончено, в логах появится сообщение:

```
DB upgrade is done, please disable postgres_upgrade in the
settings
```

5. Уберите добавленную секцию images и настройку postgres_upgrade и загрузите стандартный образ с помощью команды:

```
DEPLOYKA_SKIP_UPDATE=true dpl -d postgres
```

ОБНОВЛЕНИЕ С ИСПОЛЬЗОВАНИЕМ ANSIBLE-СЦЕНАРИЕВ

SINGLE CTS, BACK CTS И FRONT CTS

Для выполнения обновления всех контейнеров выполните следующие шаги:

1. Запустите ansible playbook:

```
ansible-playbook --ask-pass -v 05-update_cts.yaml
```

2. Введите пароль учетной записи root после ввода команды.

Для обновления ПО, установленного в контейнерах docker, на сервере Registry выполните:

Внимание! При выполнении скриптов обновления нельзя пропускать паузы, заложенные в него, т. к. их пропуск может привести к ошибкам обновления.

1. Скопируйте новые версии образов docker и скрипт загрузки образов в каталог /tmp/images и загрузите образы с помощью скрипта download.sh (скрипт доступен по ссылке):

```
cp *.tar /tmp/images/
cp /opt/deploy/script/load.sh /tmp/images/
cd /tmp/images/
./ download.sh
```

2. Подключитесь к консоли сервера Back и Front кластера с индексом 01 и 02, выполните команду для остановки антивируса:

```
systemctl stop kesl klnagent64
```

Примечание. В случае зависания сервера при остановке антивируса перезагрузите оба узла кластера через систему виртуализации.

3. Выполните команду:

```
pcs status
```

4. Убедитесь, что ресурсы кластера запущены согласно списку ниже:
   - dlm-clone [dlm] (back кластер) – запущен на обоих узлах кластера;
   - clvmd-clone [clvmd] (back кластер) – запущен на обоих узлах кластера;
   - clusterfs-clone [clusterfs] (back кластер) – запущен на обоих узлах кластера;
   - cluster_ip – запущен на одном узле кластера;
   - dockerd – запущен на одном узле кластера;
   - node_exporter (back кластер) – запущен на одном узле кластера;
   - cadvisor (back кластер) – запущен на одном узле кластера;
   - vmfence (back кластер) – запущен на одном узле кластера;
5. В случае если статус ресурсов кластера не соответствует перечисленным выше, выполните следующую команду, заменив *resource_name* на имя проблемного ресурса:

```
pcs resource cleanup resource_name
```

6. На узлах кластера Back с индексом 01 и 02 выполните команду ниже:

```
ls -la /opt/ex_data/files
```

Примечание. В случае зависания вывода списка директорий перезагрузите оба узла кластера через систему виртуализации.

7. Подключитесь к консоли сервера Back и Front кластера с индексом 01 либо 02 и выполните команду для определения текущего первичного узла, на котором запущены ресурсы кластера:

```
pcs status | grep dockerd
```

8. Подключитесь к консоли текущего первичного узла кластера Back и Front и последовательно выполните команду:

```
docker ps -a > current_version.txt
```

9. Сохраните полученный файл. Он потребуется для процедуры отката на предыдущую версию.
10. Запустите скрипт автоматического обновления всех контейнеров первичных серверов и введите пароль учетной записи root после ввода команды:

```
ansible-playbook --ask-pass -v 05-master_update_cts.yaml
```

11. После обновления первичных серверов проверьте функционирование системы, выполнив проверку логов на наличие ошибок и функции отправки сообщений.
12. Запустите скрипт автоматического обновления всех контейнеров вторичных серверов:

```
ansible-playbook --ask-pass -v 06-slave_update_cts.yaml
```

13. Введите пароль учетной записи root.
14. Подключитесь к консоли сервера Back и Front кластера с индексом 01 и 02, выполните команду для запуска антивируса:

```
systemctl start kesl klnagent64
```

Для отката на предыдущую версию ПО, установленного в контейнерах docker:

1. Подключитесь к консоли узлов кластера Back с индексами 01 и 02.
2. Добавьте в файл /opt/express/settings следующие строки, заменив в них версию ПО на значения из файла current_version.txt (файл получен на шаге 3 описания автоматического обновления с помощью скриптов ansible):

```
images:
  messaging: messaging:1.39.6
  settings: settings:1.39.0
  audit: audit:1.39.0
  admin: admin:1.39.1
  file_service: file_service:1.39.0
  voex: voex:1.39.0
  ad_phonebook: ad_phonebook:1.39.1
  email_notifications: email_notifications:1.39.0
  botx: botx:1.39.1
  ad_integration: ad_integration:1.39.0
  kdc: kdc:1.39.0
  routing_schema_service: routing_schema_service:1.39.0
```

3. Подключитесь к консоли узлов кластера Front с индексами 01 и 02.
4. Добавьте в файл /opt/express/settings следующие строки, заменив в них версию ПО на значения из файла current_version.txt (получен на шаге 3 описания автоматического обновления с помощью скриптов ansible):

```
images:
  trusts: trusts:1.39.0
```

5. Запустите скрипт автоматического обновления всех контейнеров первичных серверов и введите пароль учетной записи root после ввода команды:

```
ansible-playbook --ask-pass -v 05-master_update_cts.yaml
```

6. После обновления первичных серверов проверьте нормальное функционирование системы, выполнив проверку логов и функцию отправки сообщений.
7. Запустите скрипт автоматического обновления всех контейнеров вторичных серверов:

```
ansible-playbook --ask-pass -v 06-slave_update_cts.yaml
```

8. Введите пароль учетной записи root после ввода команды.

Для обновления ПО, установленного в контейнерах docker Web client кластера, на сервере Registry выполните:

Важно! При выполнении скриптов обновления нельзя пропускать паузы, заложенные в него, т.к. их пропуск может привести к ошибкам обновления.

1. Скопируйте новые версии образов docker и скрипт загрузки образов в каталог /tmp/images сервера Registry.
2. Загрузите образы с помощью скрипта load.sh:

```
cp *.tar /tmp/images/
cp /opt/deploy/script/load.sh /tmp/images/
cd /tmp/images/
./load.sh
```

3. С помощью команды ниже уточните новую версию образа контейнера web client:

```
docker images | grep web_client
```

4. Измените параметр web_client_image на актуальную версию, полученную на предыдущем шаге (параметр локализован в файле настроек group_vars/all.yaml в каталоге сценариев ANSIBLE web client (/opt/deploy/playbook-webclient)).
5. Запустите скрипт автоматического обновления всех контейнеров первичного узла кластера Web Client:

```
ansible-playbook --ask-pass -v 05-master_update_web.yaml
```

6. Введите пароль учетной записи root.
7. После обновления первичного узла кластера Web Client проверьте нормальное функционирование системы, выполнив проверку логов и функции отправки сообщений.
8. Запустите скрипт автоматического обновления всех контейнеров вторичного узла кластера Web Client аналогично пп. 5-6.

АВАРИЙНЫЕ СИТУАЦИИ ПРИ ОБНОВЛЕНИИ ИЗ ЛОКАЛЬНОГО РЕПОЗИТОРИЯ REGISTERY

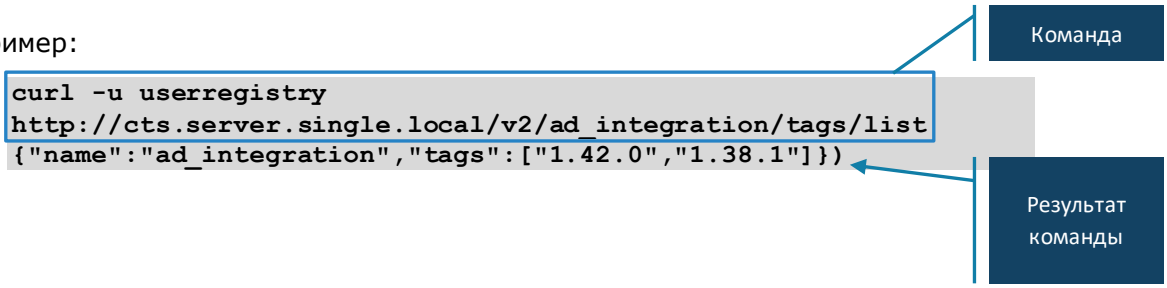
Аварийные ситуации, перечисленные ниже, могут произойти в том случае, если имеется локально развернутый сервер Registery.

Ситуация 1. Отсутствия доступа к сети интернет с узла с репозиторием.

1. С узла, имеющего доступ в интернет, скачайте актуальные контейнеры с помощью скрипта по ссылке (вложение download.sh).
2. Запустите второй скрипт по ссылке (вложение upload.sh) и дождитесь окончания загрузки.
3. Сделайте тестовый запрос из консоли при помощи обращения к URL и получите версии, находящиеся в репозитории.

(Пример:

```
curl -u userregistry
http://cts.server.single.local/v2/ad_integration/tags/list
{"name":"ad_integration","tags":["1.42.0","1.38.1"]})
```



Ситуация 2. Если в п.3 предыдущей операции результат команды – “no basic auth credentials”.

1. Удалите файл .docker/config.json.
2. Пройдите повторную авторизацию в Docker registry.

ПРОЦЕДУРА ОБНОВЛЕНИЯ СЕРТИФИКАТА

Для работы изделия требуется оформить сертификат на внешнее имя сервиса Express (FQDN или wildcard), выпущенный публичным доверенным центром сертификации и удовлетворяющий следующим требованиям:

- версия 3 и не ниже TLS 1.2;
- длина ключа не меньше 2048 бит;
- алгоритм подписи SHA 256;
- версия синтаксиса X.509 3;
- незашифрованный закрытый ключ.

Файл должен содержать в себе сертификат сервера, сертификаты промежуточного центра сертификации и корневого центра сертификации. Формат сертификатов должен соответствовать кодировке Base64. Файл закрытого ключа должен содержать нешифрованный закрытый ключ кодировки Base64.

Примерная структура файла сертификата изображена на рисунке ниже (Рисунок 1).

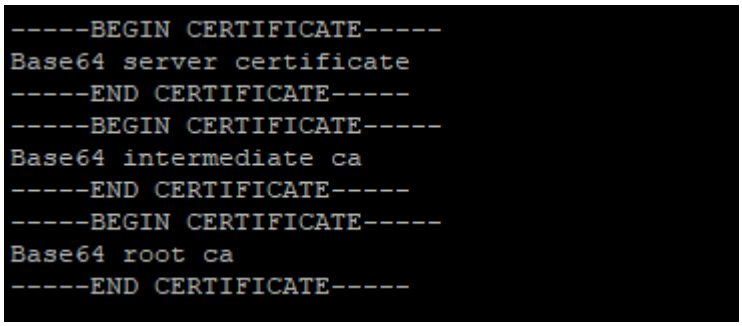


Рисунок 1

Поддерживается использование бесплатного сертификата от компании Let`s Encrypt.

Для обновления сертификата:

1. Подготовьте сертификат согласно требованиям выше.
2. Обновите файлы сертификатов, расположенные в папке /opt/express/nginx/certs/. Для разделенного сервера файлы обновляются на Front CTS.
3. Выполните команду в консоли:

```
cd /opt/express && dpl -d && dpl --dc restart nginx
```

ОБНОВЛЕНИЕ KAFKA

Для настройки Kafka перед обновлением сервера на версию 2.6:

1. Ограничьте подключения к серверу снаружи.

   Добавьте правила в IPTABLES для цепочки DOCKER-USER, выполнив следующие команды:

```
sudo iptables -A DOCKER-USER  -p tcp --dport 80 -j DROP
sudo iptables -A DOCKER-USER  -p tcp --dport 443 -j DROP
sudo iptables -A DOCKER-USER  -p tcp --dport 5001 -j DROP
```

   Если цепочки DOCKER-USER нет, то уберите параметр «'bridge': none» из файла /etc/docker/daemon.json и перезапустить службу docker.

2. Выполните проверку Kafka Lag и дождитесь ответа всех offset.

   Выведите список consumer-groups:

```
docker exec -t cts_kafka_1 /opt/kafka/bin/kafka-consumer-
groups.sh  --list --bootstrap-server localhost:9092
```

   Вывод:

```
trusts
audit
file_service
ad_integration
conference_bot
admin
ad_phonebook
events
botx
messaging
```

   По всем consumer-groups выполните:

```
docker exec -t cts_kafka_1 /opt/kafka/bin/kafka-consumer-
groups.sh --describe --group admin --bootstrap-server
localhost:9092
```

   Вывод:

```
GROUP  TOPIC           PARTITION  CURRENT-OFFSET  LOG-END-OFFSET
LAG  CONSUMER-ID
HOST          CLIENT-ID
admin  retry-30m       0          -               0
-    'admin@172.21.0.27'/<0.3836.0>-a0399c9b-a8a7-46c3-a7ab-
85fb1dcd  /172.21.0.27  'admin@172.21.0.27'/<0.3836.0>
admin  retry-5m        0          -               0
-    'admin@172.21.0.27'/<0.3815.0>-eb08d50c-4207-4c7b-870f-
0080d344  /172.21.0.27  'admin@172.21.0.27'/<0.3815.0>
admin  system-events   0          18              18
0    'admin@172.21.0.27'/<0.3880.0>-58d2f6fe-840b-412e-ae41-
39f999ba  /172.21.0.27  'admin@172.21.0.27'/<0.3880.0>
admin  retry-15m       0          -               0
-    'admin@172.21.0.27'/<0.3807.0>-0326ec57-bb39-4636-8639-
eb942ee5  /172.21.0.27  'admin@172.21.0.27'/<0.3807.0>
```

   Параметр LAG должен = 0.

   Если LAG ≠ 0, подождите 10 минут. Если значение параметра не меняется на 0, перезапустите сервис с LAG.

3. Удалите Kafka volume.

   Внимание! Предварительно создайте резервную копию /var/lib/docker/volumes/cts_kafka_logs.

   Выполните:

   - остановку и удаление контейнера Kafka:

     ```
     docker stop cts_kafka_1 && docker rm $_
     ```

   - удаление volume:

     ```
     docker volume rm cts_kafka_logs
     ```

4. Обновите версии и запустите сервисы.

   Из директории СК «Express» запустите сервисы:

   ```
   dpl -d
   ```

5. Выполните проверку контейнера Kafka:

   ```
   docker ps -a | grep kafka
   ```

   Вывод:

   ```
   CONTAINER ID   IMAGE                                COMMAND
   CREATED       STATUS          NAMES
   70c2fd5b3922   registry_name/kafka:2.13-3.3.1   "start-kafka"
   4 days ago   Up 3 minutes   cts_kafka_1
   ```

   Просмотр логов Kafka:

   ```
   docker logs cts_kafka_1
   ```

6. Удалите правила IPTABLES:

   ```
   sudo iptables -D DOCKER-USER -p tcp -m tcp --dport 80 -j DROP
   sudo iptables -D DOCKER-USER -p tcp -m tcp --dport 443 -j DROP
   sudo iptables -D DOCKER-USER -p tcp -m tcp --dport 5001 -j DROP
   ```

ИСТОРИЯ ИЗМЕНЕНИЙ

Раздел «История изменений» содержит список изменений в документе, связанных с изменениями/доработками СК «Express».

Сборка 2.8.0

№	Раздел	Изменение	Сервер	Ссылка
1.	Обновление без доступа к apt.postgresql.org	Актуализирован порядок обновления	CTS	Стр. 10