GS LABS
GS GROUP

Публичный - L0

DRE Advanced Encryption Service

Руководство по установке

Индекс	DREAdvancedEncryptionService-IG
Конфиденциальность	Публичный - L0
Ревизия	1.0
Статус	Согласован

Содержание

1. Аннотация

Документ содержит руководство по установке и первоначальной настройке системы "DRE Advanced Encryption Service" (далее - ADECScrambler или Система).

Документ предназначен для сотрудников отдела мониторинга и инсталляции, а также для других технических специалистов, в обязанности которых входит установка и первоначальная настройка системы.

> ⓘ Данный документ опубликован исключительно с целью изучения системных требований для установки продукта, а также ознакомления с последовательностью и деталями процесса установки. Реальная установка продукта производится с использованием внутренних репозиториев ООО "Цифра", доступ к которым предоставляется заказчику по запросу.

2. Термины и сокращения

Термин	Определение
Транспортный поток (TS)	Набор объединенных элементарных потоков, используемый для передачи аудио, видео и других данных в системах цифрового вещания. Структура транспортного потока определена в стандарте ISO/IEC 13818-1.
Элементарный поток	Поток данных одного типа, передающийся в составе транспортного потока. Примеры: аудиодорожка, видео, телетекст, служебная информация.
Скремблер (Scrambler)	Устройство шифрования транспортного потока, входящее в состав головного оборудования. В терминологии стандарта DVB-Simulcrypt обозначает функциональный логический блок, ответственный за шифрование MPEG2 транспортного потока. Конкретная функциональность зависит от реализации.

Сокращение	Расшифровка
ADEC	(ADvanced EnCryption) - система шифрования транспортного потока, применяемая в дополнение к стандартному алгоритму шифрования (CSA).
EMS	Error Map Server
MPEG	(от Moving Picture Experts Group – Группа Экспертов по Движущемуся Изображению) – название системы кодирования набора сжатых цифровых телевизионных видеосигналов, звуковых сигналов и данных пользователя телевизионной информации в поток цифровых пакетов
IP	(Internet Protocol) – протокол передачи данных по сети Интернет
PID	Идентификатор пакетов, относящихся к одному элементарному потоку. Уникален в пределах транспортного потока.
TS	Transport Stream, Транспортный поток (см. таблицу терминов)

3. Введение

3.1. Требования к квалификации установщика

Для установки системы сотрудник обязан:

- иметь навыки работы с ОС семейства Linux, а именно:
  - установка пакетов;
  - создание и настройка сетевых подключений;
  - запуск служб, настройка автозапуска служб;
  - установка и настройка PostgreSQL;
  - создание и работа с БД под управлением PostgreSQL.
- иметь знания о DNS.
- иметь базовые представления и практические навыки работы с Docker.
- иметь базовые представления и практические навыки работы с Git.

3.2. Схема развертывания

Компоненты системы ADEC Scrambler при развертывании ***условно*** можно разделить на следующие группы:

1. host-машина, на которой развернуты ADEC Manager DB.
2. Кластер, в котором установлены и работают ADEC Manager + ADECScrambler WEB (UI).
3. host-машина, на которой развернута база данных ADEC DB (= TDE DB на структурной схеме).
4. одна или несколько host-машин, на которых развернуты инстансы ADECScrambler.

3.3. Системные требования

Каждый из указанных компонентов Системы (ADEC Manager, базы данных, инстансы ADECScrambler) устанавливается в отдельный Docker контейнер.

Группы компонентов могут быть установлены как на отдельных серверах, так и на одной машине.

Для установки желательно выделить отдельный(-ые) сервер(-а). Сервер(-а) рекомендуется устанавливать в локальной сети, защищенной от доступа извне.

3.3.1. Серверные компоненты

К серверным компонентам относятся все компоненты Системы ADECScrambler, кроме ADECScrambler (instances), а именно: ADEC_DB (= TDE DB на структурной схеме), ADEC_Manager_DB, ADEC Manager, ADECScrambler WEB (UI) и сопутствующие микросервисы (AGS, EMS).

3.3.1.1. Аппаратное Обеспечение

Явные требования отсутствуют.

3.3.1.2. Программное Обеспечение

- ОС Debian 8x64

> ОС должна иметь версию 8.1 (ядро 3.16.0-4).

- Для установки баз данных:
  - PostgreSQL последней версии (рекомендуемая версия - 14.4).

Ограничения:

- Для установки компонентов PostgreSQL требуется **доступ к сети Internet**.
- Для клонирования репозитория (для загрузки образов) требуется **доступ в репозиторий gitlab** (доступ ограничен).
- Порт, на который будет приходить результат работы инстансов ADECScrambler, должен быть открыт.

3.3.1.3. Системные требования для развертывания компонентов Системы

Компоненты Системы разворачиваются в кластере Kubernetes. Для данных компонентов должна быть развернута одна нода кластера.

Для установки необходимо предварительно выполнить следующие требования:

- На отдельном сервере подготовлена Ansible node с поддержкой CI/CD. За информацией обращаться к разработчику платформы автоматизации CI/CD ООО "Цифра".
- Установлен и настроен кластер Kubernetes через K3s.
  - Так как развертывание производится в кластере k8s, то необходим config file для доступа к кластеру.
    1. Если пользователь выполнял развертывание Kubernetes самостоятельно, то он сам должен создать config file (см. документацию Kubernetes).
    2. Если Kubernetes был развернут сторонними людьми, то необходимо получить config file у администратора кластера.
- На машине администратора установлен kubectl (https://kubernetes.io/docs/tasks/tools/install-kubectl/).
- На машине администратора установлен helm (https://helm.sh/ru/docs/intro/install/).
- Развернут DNS-сервер, преобразование имен dns зоны настроено на мастера k8s (созданы A записи на зону dns). DNS устанавливается в сетевое окружение DMZ зоны, где будет развернут ADECScrambler.
- Для корректной работы системы ADECScrambler требуется развернуть кластер БД (доступ предоставляется по запросу).
- Для корректной работы системы ADECScrambler необходим доступ к следующим ресурсам:
  - chartmuseum (ссылка и права доступа предоставляются по запросу заказчика)
  - gitlab (ссылка и права доступа предоставляются по запросу заказчика)
- Необходим доступ к репозиторию (ссылка и права доступа предоставляются по запросу заказчика), содержащему helmfile для развертывания Системы. Helm файл содержит инструкции, с помощью которых осуществляются настройки устанавливаемых компонентов. Сами компоненты поставляются в виде образов (images), из которых разворачиваются Docker-контейнеры. Данные берутся из gitlab (ссылка и права доступа предоставляются по запросу заказчика).

3.3.2. ADECScrambler (instances)

3.3.2.1. Аппаратное Обеспечение

Опционально: Для ввода и вывода транспортного потока **может** использоваться плата *Dektec (DTA 160 или DTA 2160 или DTA-2145)*, имеющая три универсальных ASI-порта (вход/выход) и один Gigabit Ethernet порт.

Другие явные требования отсутствуют.

3.3.2.2. Программное Обеспечение

- ОС Debian 8x64

  > ОС должна иметь версию 8.1 (ядро 3.16.0-4).

- Пакеты Docker:
  - Docker-CE версии 17.09.1-ce и выше.
  - Docker-CLI.
  - docker-compose ver.1.24.1 и выше.

    > Для работы компонентов на host-машине дополнительно должен быть запущен docker-daemon (сам docker-daemon автоматически устанавливается вместе с другими пакетами Docker).

- Если ADECScrambler (instance) работает с платой Dektec:
  - на host-машине должны быть установлены драйверы для платы Dektec. Deb-пакет с драйверами входит в комплект поставки.

Ограничения:

- Для установки компонентов docker требуется **доступ к сети Internet**.
- Для загрузки образов требуется **доступ в репозиторий gitlab** (доступ ограничен).
- Порт, на который будет приходить результат работы инстансов ADECScrambler, должен быть открыт.

3.4. Компоненты для установки системы

- Все компоненты системы ADECScrambler, кроме ADECScrambler (instances), устанавливаются из специального репозитория (доступ ограничен)
- Файл ***adec_load_tde.sh*** - скрипт осуществляет загрузку лестницы ключей (только один раз) в целевые структуры ADEC DB
- Файл ***adec_load_keys.sh*** - скрипт осуществляет загрузку ключей в целевые структуры ADEC DB
- Файл ***dektec-3.0.0-Linux.deb*** - установочный пакет драйверов для плат DekTec (*лежит в папке _dist* соответствующего релиза ADECScrambler)

Перейти к Содержанию...

4. Предварительные действия

4.1. Установка PostgreSQL

По умолчанию требуется развернуть кластер БД (доступ к инструкции предоставляется по запросу).

Данный раздел следует использовать только в случае установки БД в режиме Standalone.

ВНИМАНИЕ! Для работы системы требуется PostgreSQL версии 14 или выше.

Установка PostgreSQL на сервер без развертывания и настройки кластера БД (в случае установки БД в режиме Standalone) выполняется стандартным образом.

4.2. Настройка PostgreSQL и ODBC

Необходимо настроить конфигурационные файлы PostgreSQL. а также ODBC для установления удаленного доступа:

- pg_hba.conf
- postgres.conf
- odbcinst.ini
- odbc.ini

По умолчанию требуется развернуть кластер БД (доступ к инструкции предоставляется по запросу).

Данный раздел следует использовать только в случае установки БД в режиме Standalone.

Следующие действия выполняются на сервере, где будут развернуты базы данных, только после установки пакета postgresql-14.

1. Открыть конфигурационный файл postgresql.conf для редактирования:

```
sudo nano /etc/postgresql/14/main/postgresql.conf
```

2. В файле выполнить следующее:
   a. Изменить значение параметра listen_addresses, как показано ниже, и раскомментировать соответствующую строку:

```
listen_addresses = '*'          # what IP address(es) to listen on;
```

   b. Для настройки автовакуума рекомендуются приведенные ниже значения (использовались при тестировании):

```
autovacuum = on
#log_autovacuum_min_duration = 0
autovacuum_max_workers = 10
autovacuum_naptime = 1s
autovacuum_vacuum_threshold  = 50
autovacuum_analyze_threshold = 50
autovacuum_vacuum_scale_factor = 0.01
autovacuum_analyze_scale_factor = 0.02
```

3. Открыть конфигурационный файл pg_hba.conf для редактирования:

```
sudo nano /etc/postgresql/14/main/pg_hba.conf
```

4. Необходимо, чтобы к postgres могли подсоединиться любые процессы с локальной машины и компьютеры из локальной сети (например, с ip 192.168.x.x). Также необходимо указать настройки IPv6. Таким образом, файл может выглядеть следующим образом (рекомендуется задавать уровень доступа менее открытый, чем trust):

```
# "local" is for Unix domain socket connections only
local        all
all                                                                                 trust
# IPv4 local connections:
host        all                         all                         127.0.0.1
/32                                 md5
host        all                         all                         172.17.0.0
/16                                 md5
host        all                         all                         192.168.0.0
/16                                 md5
# IPv6 local connections:
host        all                         all                         ::1
/128                                                    md5
```

5. После внесения изменений перезапустить PostgreSQL:

```
sudo /etc/init.d/postgresql restart
```

6. Установить unixodbc (если нет), odbc-postgresql:

```
sudo apt-get install unixodbc odbc-postgresql
```

7. Открыть конфигурационный файл ***odbcinst.ini*** для редактирования. Файл находится здесь:

```
/etc/odbcinst.ini
```

8. Проверить, что в файле есть драйвер [PostgreSQL Unicode]:

```
[PostgreSQL ANSI]
Description=PostgreSQL ODBC driver (ANSI version)
Driver=/usr/lib/x86_64-linux-gnu/odbc/psqlodbca.so
Setup=libodbcpsqlS.so
Debug=0
CommLog=0
UsageCount=1

[PostgreSQL Unicode]
Description=PostgreSQL ODBC driver (Unicode version)
Driver=/usr/lib/x86_64-linux-gnu/odbc/psqlodbcw.so
Setup=libodbcpsqlS.so
Debug=0
CommLog=0
UsageCount=1
```

9. При работе ADECScrambler требуются подключения к базам данных, приведенным в таблице ниже. Необходимо настроить к ним доступ:

Название БД	Администратор БД	Примечание
adec_tde	adecadmin	Имеется в виду ADEC_DB (= TDE DB на структурной схеме, с этой базой работают инстансы)
adec_manager	adecadmin	Имеется в виду ADEC_Manager_DB
adecscrambler_errmap	errmapadmin	Имеется в виду EMS_DB

Пример:

a. Открыть конфигурационный файл ***odbc.ini*** для редактирования. Файл находится здесь:

```
/etc/odbc.ini
```

b. Привести файл в соответствии с описанием (убедитесь, что в Servername указан ip сервера, а не 127.0.0.1, иначе контейнеры будут работать со своей внутренней сетью и не смогут получить доступ к базам):

```
[adec_tde]
Description           = PostgreSQL
Driver                = PostgreSQL Unicode
Servername            = <server_ip>
Database          = adec_tde
UserName              = postgres
Password              = postgres
Port                  = 5432
ReadOnly        = No
RowVersioning           = No
ShowSystemTables        = No
ShowOidColumn           = No
FakeOidIndex            = No

[ADEC_MANAGER_DB]
Description  = PostgreSQL
Driver   = PostgreSQL Unicode
Database  = adec_manager
Servername  = <server_ip>
UserName  = adecadmin
Password  = adecadmin
Port   = 5432
ReadOnly  = No
RowVersioning           = No
ShowSystemTables        = No
ShowOidColumn           = No
FakeOidIndex            = No
Trace = 0
Protocol  = 7.4-0
UseServerSidePrepare = 0

[EMS_DB]
Description  = PostgreSQL
Driver   = PostgreSQL Unicode
Database  = adecscrambler_errmap
Servername  = <server_ip>
UserName  = errmapadmin
Password  = errmapadmin
Port   = 5432
ReadOnly  = No
RowVersioning           = No
ShowSystemTables        = No
ShowOidColumn           = No
FakeOidIndex            = No
Trace = 0
Protocol  = 7.4-0
UseServerSidePrepare = 0
```

Перейти к Содержанию...

5. Установка и настройка компонентов Системы

5.1. Процедура установки

Необходимо выполнить установку системы, как описано ниже (в соответствии с документом Описание схемы CD, тэг 4.0).

5.2. Как создать новую среду

1. Создать отдельный проект в Gitlab
2. Настроить данный проект как подмодуль на основе инструкции (ссылка и права доступа предоставляются по запросу заказчика).
3. В проекте среды создать helmfile.yaml с содержимым:

```
---
helmfiles:
  - path: <путь до подмодуля>/helmfile.yaml
    values:
      - <путь до подмодуля>/default.yaml # Загружаем зачения по-умолчанию
      - production.yaml                  # Применяем собственную конфигурацию
      - versions.yaml                    # (опционально) Переопределяем версии некоторых компонентов
```

5.3. Пример .gitlab-ci.yml

```
# здесь перечисляются необходимые шаги(stage) пайплайна
# в случае, если часть вышеописанного функционала
# не требуется, ненужные шаги можно удалить
# (например, оставить только init)
stages:
  - init
  - compose
  - grade

variables:
  # GIT_* переменные необходимы для правильной работы
  # репозитория с сабмодулем
  GIT_SUBMODULE_STRATEGY: recursive
  GIT_STRATEGY: clone
  # если namespace релиза не задаётся через values/шаблоны/helmfile,
  # то его можно задать через переменную NAMESPACE
  NAMESPACE: cas-stand
  STAGED_PIPELINE: "true"

include:
  - project: 'automation/cd-templates'
    ref: "4.0"
    file: pipeline.yml
```

5.4. Двухступенчатый деплой

Для выполнения двухступенчатого деплоя, в случае если часть релизов, описанных в helmfile, следует установить прежде остальных, следует выполнить три условия:

- задать в файле .gitlab-ci.yml переменную *STAGED_PIPELINE* в значение *true*;
- в helmfile.yaml задать переменные *wait* и *waitForJobs*;
- указать для каждого релиза этап его установки посредством меток *stage: first* или *stage: second*.

При этом возможно так же установить допустимый период ожидания выполнения установки релизов/джобов посредством переменной timeout (по умолчанию - 300).

Версия шаблонов CI должна быть не менее 4.0.

5.5. Настройка CD для продукта, опубликованного в Releases

Процедура описана в отдельном репозитории (доступ предоставляется по запросу).

5.6. Настройка и развертывание ADECScrambler

5.6.1. CD для артефактов БД

При развертывании ADECScrambler происходит установка SCH и API для БД через механизм Kubernetes Jobs. В процессе установки сохраняется лог в контейнере.

```
adec_manager_db_sch:
  enabled: true
  # You can optionally override database address and port here:
  #db:
  #  address: 127.0.0.1
  #  port: 5432

adec_manager_db_api:
  enabled: true
```

Этот режим поддерживают: ADEC_Manager_db_***, ADEC_db_***, ErrMap_db_***.

5.6.2. Настройка переменных окружения

В системе развертывания ADEC Manager требуется указывать переменные окружения, которые используются непосредственно в самом процессе деплоя ADEC Manager в кластер.

Настройка переменных осуществляется в gitlab.

В боковом меню выбрать **Settings** (на панели слева) -> **CI/CD** -> **Environment variables**. Отредактировать переменные.

Таблица с описанием используемых переменных Gitlab

Переменная	Описание

ADECMANAGERDB_LOGIN	Имя пользователя для ADEC Manager DB
ADECMANAGERDB_PASSWORD	Пароль пользователя для ADEC Manager DB
ERRMAPDB_LOGIN	Имя пользователя для Errmap DB (базы даных EMS)
ERRMAPDB_PASSWORD	Пароль пользователя для Errmap DB (базы даных EMS)
POSTGRES_LOGIN	Имя администратора PostgreSQL БД
POSTGRES_PASSWORD	Пароль администратора PostgreSQL БД
ROUTINGDB_LOGIN	Имя пользователя для Routing DB
ROUTINGDB_PASSWORD	Пароль пользователя для Routing DB
TDEDB_LOGIN	Имя пользователя для ADEC DB (= TDE DB на структурной схеме)
TDEDB_PASSWORD	Пароль пользователя для ADEC DB

ВАЖНО! Environment variables имеют более высокий приоритет, чем переменные, заданные в файлах.

Параметры _LOGIN и _PASSWORD задаются пользователем и используются при подключении к соответствующим базам данных.

5.6.3. Настройка additional

Папка ***additional*** содержит файлы, с помощью которых настраиваются dns, ingress, probes, statsd. Указанные параметры применяются ко всем сервисам и службам в данном репозитории. **Рекомендуется не менять эти настройки**.

5.6.4. Состав репозитория

Репозиторий имеет следующий состав:

- helmfile.yaml - главный конфигурационный файл утилиты helmfile.
- default.yaml - файл с values окружения утилиты helmfile.
- values - папка с values для каждого чарта; они являются шаблонными и забирают значения из values окружения (файла default.yaml).
- versions.yaml - файл с версиями компонентов; если в версии установлена пустая строка, то берется последняя версия (в соответствии с semver2).
- limitation - папка с values ресурсов подов. С помощью этих файлов настраиваются компоненты системы ADECScrambler, в том числе базы данных.

5.6.5. Выбор компонентов для установки

По умолчанию разворачиваются все компоненты ADECScrambler, однако при необходимости можно отключать ненужные: для этого в production.yaml, в корне секции соответствующего компонента нужно выставить *enabled: false*.

5.6.6. Динамические параметры в конфигурационных файлах

В конфигурационных файлах *_server.cfg параметры разделены на две группы:

1. Все параметры, лежащие вне секции "system". Эти параметры можно менять динамически, т.е. без перезапуска соответствующей службы. При изменении значений этих параметров в конфигурационном файле, по прошествии некоторого времени, новые значения будут автоматически применены к службе.

   > ⚠ **Обратите внимание!** Параметры, изменяемые динамически, нельзя задать через переменные окружения (см. выше), они меняются только в конфигурационном файле.

2. Параметры в секции "system". Эти параметры нельзя изменить динамически: чтобы изменения этих параметров вступили в силу, соответствующая служба должна быть перезапущена.

Некоторые из динамически изменяемых параметров (например, xxx.host в *.cfg) нельзя применять со значениями "по умолчанию", они должны быть настроены на production.

5.7. Развертывание системы (основные этапы)

Этапы развёртывания:

1. Предварительные действия:
   a. Создать новую среду.
   b. Настроить двухступенчатый деплой.
   c. Настроить environment variables (см. Настройка переменных окружения).
   d. Настроить yml-файлы, которые определяют состав и настройки разворачиваемых сервисов и баз данных, см. "Настройка additional", "Состав репозитория".
   e. В конфигурационных файлах настроить параметры, которые нельзя оставлять "по умолчанию" и /или нельзя изменить динамически, см. "Динамические параметры в конфигурационных файлах ".
2. Установка компонентов, входящих в состав Системы (кроме instances), в git (с помощью CI/CD). См. " Установка и настройка компонентов Системы".
   Особенности:
   a. Развертывание осуществляется в Gitlab CI/CD.
      i. В боковом меню выбрать **Settings** (на панели слева) -> **CI/CD** -> **Сборочные линии**.
      ii. В правом верхнем углу нажать кнопку **Запустить сборочную линию**.
      iii. Дождаться окончания операции.
   b. Развертывание системы (с помощью CI/CD) выполняется в ДВА этапа (эти два этапа могут быть разнесены по времени):
      i. Установка Баз Данных, входящих в состав ADECScrambler.
      ii. Установка сервисов и служб, входящих в состав ADECScrambler.
   c. Перед установкой требуется создать и соответствующим образом настроить production.yaml (см. CD для артефактов БД).

3. **(Обязательно) удалить jobs**, созданные при развертывании баз данных, иначе в дальнейшем нельзя будет накатить новые DB_API и DB_SCH.

   ⚠ **ВНИМАНИЕ!** При установке в production базы (XXX DB) её старые схемы XXX_DB_API, соответствующие более ранним релизам, автоматически не удаляются. Т.е. старые схемы XXX_DB_API нужно удалять вручную.

4. Наполнение баз данных (с помощью скриптов). См. "Начальное наполнение баз данных".
5. Настроить пользователя ADECScrambler WEB (UI) в DRE Account Manager. См. "Настройка взаимодействия ADECScrambler и DRE Account Manager".
6. Развернуть ADECScrambler (instances). См. "Установка и настройка ADEC Scrambler (Instances)".

5.8. Редактирование production.yaml (для сервисов ADECScrambler)

Файл production.yaml создается на основе default.yaml, содержащего основные настройки Системы. Настройки, заданные в default.yaml, кроме (опционально) параметров подключения, являются достаточными для эксплуатации Системы.

Особенности:

- Значения параметров, заданные в production.yaml, имеют более высокий приоритет, чем значения, заданные в default.yaml.
- Если параметр не задан в production.yaml, то будет использовано значение, заданное в default.yaml.
- Если параметр не задан ни в production.yaml, ни в default.yaml, то будет использовано значение, заданное в конфигурационном файле данного компонента.

Перейти к Содержанию...

6. Начальное наполнение баз данных

Информация по начальному наполнению БД предоставляется по запросу Заказчика.

Перейти к Содержанию...

7. Настройка взаимодействия ADECScrambler и DRE Account Manager

7.1. Добавление пользователя в DRE Account Manager

Для эксплуатации ADECScrambler WEB (UI) и, как следствие, добавления ADECScrambler (instances), необходимо, чтобы в DRE Account Manager (далее по тексту - Account Manager или ACM) был(и) создан(ы) пользователь(-ли) ADECScrambler WEB (UI), без ограничений доступа.

Подробное описание работы в DRE Account Manager приведено в документе "DRE Account Manager. Руководство пользователя".

7.2. Пробный запуск ADECScrambler WEB (UI)

1. Открыть Internet web browser.
2. Ввести IP-адрес web-сервера (ADECScrambler WEB (UI)) и нажать Enter, например:

```
http://127.0.0.1
```

3. На экране должна отобразиться стартовая страница:

4. Ввести login и password пользователя, полученные у администратора Account Manager, и нажмите "Войти".
    a. В случае успеха будет отображена начальная страница (вкладка Серверы).

Перейти к Содержанию...

8. Установка и настройка ADEC Scrambler (Instances)

8.1. Установка пакетов Docker

Установка пакетов docker-ce выполняется при наличии доступа в Интернет. Способ выполнения этой операции остается **на усмотрение заказчика**.

⚠ Помимо других компонентов docker также должен быть установлен docker-ce-cli

Инструкция по установке docker-ce описана на официальном сайте:

- на ОС Debian: https://docs.docker.com/engine/installation/linux/docker-ce/debian/#install-using-the-repository
- на другие платформы: https://docs.docker.com/install/#supported-platforms

8.2. Добавление пользователя в группу docker

При работе с Docker необходимо все команды с ним выполнять под *sudo* .

1. Чтобы этого избежать, рекомендуется добавить своего пользователя в группу docker. Для этого зайти в систему под требуемым пользователем (если это **не root**) и выполнить следующую команду:

   ```
   usermod -a -G docker <current_user>
   ```

2. Перелогиньтесь либо выполните перезагрузку, с тем чтобы новые права вошли в силу.

8.3. Запуск docker daemon

⚠ **Обратите внимание!** Запуск docker daemon необходим только для работы с ADEC Manager.

Для работы компонентов на host-машине дополнительно должен быть запущен docker-daemon (для прослушивания запросов по http REST API). Сам docker-daemon автоматически устанавливается вместе с другими пакетами Docker.

Работа с docker daemon описана здесь: https://docs.docker.com/config/daemon/

В общем случае необходимо выполнить следующее:

1. Запустите Docker daemon:

   ```
   sudo dockerd -H unix:// -H 0.0.0.0:2375 &
   ```

   где:

   a. 2375 - порт, по которому можно будет потом обращаться к Docker daemon. Можно задать другое значение, но обычно используется 2375. Порт впоследствии используется при создании хоста через WEB-интерфейс.

8.4. Открытие портов

Как сказано выше, для работы ADECScrambler (Instances) на всех машинах (такой вариант возможен, например, если выходные данные выводятся на другой ЭВМ) должны быть **открыты** все **порты**, которые будут использоваться для обмена входными/выходными данными.

8.5. Предварительные действия

Перед развертыванием ADECScrambler (instance) необходимо выполнить следующее:

1. Настроить odbc.ini (если это не было сделано ранее), указав в нем в качестве ресурса используемую базу ADEC DB (TDE_DB). Наименование DSN ресурса должно быть – **adec_tde**. См. Настройка PostgreSQL и ODBC.
   Пример содержимого файла:

```
[adec_tde]
Description           = PostgreSQL
Driver                = PostgreSQL Unicode
Servername            = <server_ip>
Database          = adec_tde
UserName              = postgres
Password              = postgres
Port                  = 5432
ReadOnly        = No
RowVersioning           = No
ShowSystemTables        = No
ShowOidColumn           = No
FakeOidIndex            = No
```

2. Ключи HWRK (keyfile11.dat + keyfile12.dat) и BBMK (bbmk.dat) (см. "Начальное наполнение баз данных") положить на host-машину, где будет развернут ADECScrambler (instance), в папку */etc*.
3. Предоставить пользователям права на чтение, изменение и запуск содержимого этой папки:

```
sudo chmod -R 777 /etc
```

4. Скачать образ adec_scrambler_go (инстанса), который будет развернут на этой машине:

```
docker pull <link_to_adec_scrambler>:<version>
```

> **ВНИМАНИЕ!** Этот шаг необходим для корректного запуска инстансов через WEB-интерфейс.

8.6. Установка и настройка драйверов для Dektec

На host-машину, где будет разворачиваться ADECScrambler (instance), должны быть установлены драйверы для платы Dektec.

8.6.1. Установка драйверов для плат DekTec

Драйверы для плат DekTec, необходимые для работы ADEC Scrambler, поставляются в виде deb-пакета.

Порядок действий для установки:

1. Убедиться, что текущий пользователь - администратор Системы.
2. Перейти в папку с deb-пакетом ***dektec-3.0.0-Linux.deb***
3. Установить пакет с помощью команды:

```
sudo dpkg -i [package filename]
```

4. Выполнить команду обновления ссылок на библиотеки:

```
sudo ldconfig
```

5. После установки пакета необходимо проверить корректность установки драйверов Dektec. Для этого:
   a. Выполнить команду:

```
lsmod | grep Dta
```

   b. В ответе должна содержаться информация о модулях *DtaNw* и *Dta*. Пример:

```
DtaNw                  35275  0
Dta                   927540  1 DtaNw
```

8.6.2. Настройка сетевого интерфейса платы DekTec

⚠ Данная процедура выполняется администратором сервера.

1. С помощью команды

```
sudo ifconfig -a
```

   выяснить, какой индекс был присвоен интерфейсу DekTec. Допустим, был присвоен индекс *eth1* (может отличаться, в зависимости от конфигурации оборудования).
2. Открыть файл */etc/network/interfaces* для редактирования:

```
sudo nano /etc/network/interfaces
```

3. Для получения ip адреса через dhcp, добавить в файл строки:

```
auto eth1
iface eth1 inet dhcp
```

4. Для задания статического ip добавить строки:

```
auto eth1
 iface eth1 inet static
 address [ip]
 netmask [netmask]
```

, где [ip] - адрес в формате X.X.X.X, [netmask] - маска сети в формате Y.Y.Y.Y.

8.7. Настройка sysctl

Для работы **с multicast потоками (по UDP)** на каждой host-машине, где установлен ADECScrambler (instance), требуется добавить настройки сети (*net_core_wmem_default*, *net_core_rmem_max*, *net_core_wmem_max*) для ОС (Debian), а также изменить значения других параметров.

Процедура выполняется после развертывания ADEC Manager, но до развертывания инстансов ADEC Scrambler.

В случае обновления ADECScrambler достаточно убедиться, что эти настройки уже выставлены в системе. Если нет, то выполнить описанные действия.

Последовательность действий (на каждой host-машине, где установлен ADECScrambler (instance)):

1. Открыть на редактирование файл */etc/sysctl.conf*.
2. В файле */etc/sysctl.conf* настроить параметры следующим образом (если параметры отсутствуют, то их необходимо добавить в файл):

```
net.core.rmem_default = 33554432
net.ipv4.tcp_mem = 8388608        12582912        16777216
net.ipv4.udp_mem = 8388608        12582912        16777216
net.ipv4.udp_rmem_min = 16384
net.ipv4.udp_wmem_min = 16384
net.netfilter.nf_conntrack_count = 810
vm.dirty_ratio = 50
vm.swappiness = 30
net.core.wmem_default=31457280
net.core.rmem_max=33554432
net.core.wmem_max=3355443
```

3. Выполнить команду для применения новых настроек:

```
sudo sysctl -p
```

После перезагрузки ОС указанные настройки будут сброшены до стандартных.

8.8. Развертывание ADECScrambler (instance)

Развертывание осуществляется с помощью ADECScrambler WEB (UI).

ADECScrambler WEB (UI) описан в документе "DRE Advanced Encryption Service. Руководство пользователя".

В общем случае нужно выполнить следующее:

1. Подключиться к ADECScrambler WEB (UI).
2. Создать описание сервера (host-машины), на котором развернут ADECScrambler (instance) - вкладка "Серверы", кнопка "+ Добавить".
3. Создать описание ADECScrambler (instance) - двойной клик на сервер, вкладка "Запущенные скремблеры", кнопка "+ Добавить".
4. Создать PIDMap для ADECScrambler (instance) - двойной клик на скремблер, вкладка "Каналы", таблица: PIDMap, кнопка "+ Добавить".

Перейти к Содержанию...

8.9. Настройка взаимодействия ADECScrambler и DRE Account Manager (загрузка прав для ADECScrambler в ACM)

Для корректного взаимодействия систем ADECScrambler и DRE Account Manager (ACM) должны быть настроены и использоваться permissions по работе с разными разделами web-интерфейса ADECScrambler.

Процедура выполняется в следующих случаях:

- при установке системы ADECScrambler "с нуля";
- в случае обновления/добавления/удаления прав (permissions).

Данную процедуру можно выполнять только после установки обновленных баз ADECScrambler.

Последовательность действий:

1. В ACM добавить сервис "adecscrambler":

   > ⓘ Данная процедура выполняется однократно, до установки/обновления permissions в Account Manager.
   >
   > Имейте в виду, что в случае сбоя/переустановки Account Manager "с нуля" этот сервис может быть удалён - в этом случае его придётся создавать заново.

   a. Войдите в web-интерфейс Account Manager.
   b. В левой части окна выберите Administration -> Services.

      > ⚠ Подробное описание работы с сервисами Account Manager приведено в документе "DRE Account Manager. Руководство пользователя", в разделе "Сервисы".

   c. Проверьте, что в списке сервисов присутствует сервис 'adecscrambler'. Если его нет, добавьте его.
2. В ACM добавить в роль "MDS" (или любую другую роль для пользователей ADECScrambler) права из сервиса "adecscrambler":

> ⚠ Ниже описаны общие действия. Подробное описание работы в Account Manager приведено в документе "DRE Account Manager. Руководство пользователя", в разделе "Роли".
>
> Администратору Account Manager также может понадобиться выдать права (permissions) подсистемам, подключащимся к ADECScrambler (например, Billing system). Таким подсистемам Account Manager обычно выдает системные токены. Перечень таких подсистем не относится к компетенции пользователей и разработчиков ADECScrambler, необходимые действия выполняет администратор Account Manager.

a. Войдите в web-интерфейс Account Manager.
b. В левой части окна выберите Administration -> Roles.
c. В появившейся форме, в поле Name введите имя административной роли (например, MDS). Особенности:
    i. Есть общая роль, которая отвечает за синхронизацию ACM и ADECScrambler. В примере выше она (по умолчанию) называется MDS.
    ii. В теории, на разных стендах, у неё может быть разное название. Так же, могут быть дополнительные роли для синхронизации прав.
    iii. Имя администратора может отличаться от *MDS*, поэтому фактически наличие permissions нужно проверить у администратора для синхронизации прав. Если такого администратора нет, то нужно создать администратора *MDS*.
d. На экране отобразится список разрешений, доступных для этой роли. Необходимо, чтобы в правый список были добавлены все разрешения, которые начинаются с smscasaccess. Если это не так, добавьте их к административной роли.

3. Обновить конфигурационные файлы репозитория ADECScrambler:
    a. В production.yaml, в секции **account_manager**, выставить параметр service_name: "adecscrambler".
    b. В production.yaml, в секциях **routing_db_sch** и **routing_db_api**, выставить параметр enabled: true.
    c. В production.yaml включить накат с помощью **adec_routing_init** - в секции adec_routing_init выставить параметр enabled: true.

> ⚠ При включении adec_routing_init наполнение роутов для ADECScrambler'а (выполнение скриптов routings_load.sh и create_acm_permissions.sh в ACM) осуществляется автоматически. Т.е. никаких дополнительных действий в ADECScrambler не требуется.

> ⚠ **Обратите внимание!** Если ags_web уже запущен и подключен к routing-server на момент выполнения скрипта, то его (ags_web) потребуется перезапустить.

    d. Проверить, что в production.yaml, в секции **routing_server**, секция включена (параметр enabled: true) и в ней (в секции routing_server) имеются блок *configEnabled: true* и *config*.
    e. В production.yaml настроить секцию **pg_db**. Секция отвечает за настройку соединения с БД, в которой будут развернуты базы для компонентов, связанных с AGS (routing_db и ems_db). Пример pg_db:

```
pg_db:
  enable: true
  address: 127.0.0.1
  port: 5432
  master_port: 5432
  async_port: 5432
  max_conn_lifetime: 60
  max_conn_idle_time: 60
```

4. Если web-интерфейс ADECScrambler уже запущен, то сбросьте кеш страницы (в браузере) и выполните переавторизацию в ADECScrambler Web.

Перейти к Содержанию...