2022

ALTUERA

Описание процесса установки программного обеспечения «VideoChat»

Содержание

1. Введение

Материал документа содержит описание процесса установки, настройки и запуска программного обеспечения VideoChat.

Руководство предназначено для системного администратора программного обеспечения Eralink.

1.1. Требования к квалификации персонала

Для успешной установки, настройки и дальнейшего обслуживания системы ответственный сотрудник должен обладать опытом использования и администрирования следующего программного обеспечения:

- OS Linux (RedOS, CentOS, RHEL)
- ansible

2. Назначение и условия применения

2.1. Назначение компонентов VideoChat

2.1.1. Компоненты VideoChat

vc-admin – компонент VideoChat, предоставляющий API для конфигурации ПО
vc-auth – компонент VideoChat, отвечающий за аутентификацию пользователей и валидацию запросов к API
vc-eventserver – компонент VideoChat, агрегирующий статистические метрики WebRTC для системы мониторинга
vc-fileserver – компонент VideoChat, предоставляющий API для работы с файлами, а также осуществляющий контроль занятого файлами места на дисках и выполняющий регулярные работы по удалению старых файлов
vc-geo – компонент VideoChat, отвечающий за преобразования координат в текстовые данные геолокации с использованием внешних GEO сервисов
vc-history – компонент VideoChat, осуществляющий хранение и восстановление истории чатов, а также предоставляющий API для доступа к исторической отчетности
vc-license – компонент VideoChat, осуществляющий контроль лицензионных ограничений решения
vc-mediamerge – компонент VideoChat, отвечающий за конвертации аудио и видео записей взаимодействий в единый файл
vc-msg – компонент VideoChat, осуществляющий доставку ссылок до Клиентов и (опционально) контроль за доставкой
vc-records – компонент VideoChat, управляющий процессом конвертации и загрузки записей взаимодействий

vc-server - основной компонент VideoChat, осуществляющий: контроль жизненного цикла взаимодействий, контроль подключения пользователей к платформе, а также обеспечивающий взаимодействие пользовательских интерфейсов с прочими компонентами
vc-stat – компонент VideoChat, отвечающий за агрегацию статистических данных взаимодействий и предоставляющий API для доступа к исторической отчетности
vc-statchart – вспомогательный компонент VideoChat, предоставляющий возможность отображения информации о качественных показателях WebRTC для взаимодействий
vc-surl – компонент VideoChat, ответственный за укорачивание URL ссылок для отправки Клиентам
frontend admin-panel – административный веб-интерфейс системы, предоставляющий возможность осуществлять настройку, а также предоставляющий доступ к статистике
frontend agent – веб интерфейс Агента
frontend user – веб интерфейс Клиента для браузеров мобильных устройств
frontend desktop_user – веб интерфейс Клиента для браузеров ПК
storage – хранилище файлов, (опционально) подключаемое к серверу, и используемое компонентом **vc-fileserver**

2.1.2. Сторонние компоненты VideoChat

activemq – Брокер шины обмена данными между компонентами VideoChat по протоколу MQTT
coturn – STUN/TURN сервер, помогающий устанавливать WebRTC соединения Клиентов, находящихся за NAT, с сервером VideoChat
janus – медиа-шлюз, обеспечивающий установку и контроль WebRTC соединений и RTP медиа потоков для всех пользователей VideoChat
logstash – вспомогательный компонент VideoChat, осуществляющий сбор логирования с браузеров клиентов VideoChat в целях поиска ошибок
mongod – СУБД решения
nginx – веб сервер решения, являющейся единой точкой доступа ко всем пользовательским веб-интерфейсам, а также предоставляющий защищенный и авторизованный доступ веб-интерфейсов ко всем API решения.
ansible – система автоматического управления конфигурациями, используется для инсталляции, обновления и изменения конфигурации компонент VideoChat.
java – среда выполнения JAVA-приложений
firewalld – фреймворк ОС Linux для управления брендмауэром
prometheus – система хранения статистических метрик в timeseries database, используется как система мониторинга решения
grafana – система визуализации метрик Prometheus, используется как система мониторинга решения
prometheus-node-exporter – коллектор статистических метрик ОС для Prometheus
ffmpeg – набор утилит, кодеков и библиотек для работы с аудио и видео данными как в виде файлов, так в виде потоков в реальном времени

2.2. Отказоустойчивость

Платформа VideoChat предоставляет возможность резервирования «cold-standby» - холодный запуск нового сервера VideoChat. Для поддержки режима резервирования cold-standby должны быть соблюдены следующие требования:

- СУБД должна работать в распределенном отказоустойчивом режиме
- Хранилище файлов должно работать в распределенном отказоустойчивом режиме

2.3. Безопасность

Специфика реализации протокола WebRTC в браузерах ТРЕБУЕТ обязательного использования безопасного контролирующего соединения с обязательной валидацией сертификатов. В связи с этим, все подключения к платформе VideoChat должны осуществляться по протоколу https и использовать поддающиеся публичной верификации сертификаты. Взаимодействие локальных компонент VideoChat между собой осуществляется только на внутреннем (loopback) сетевом интерфейсе сервера и недоступно для подключения как из локальных, так из публичных сетей.

Инсталлятор также устанавливает и настраивает брендмауэр со следующими параметрами:

Зона **public** – источник: все публичные ip адреса вне зависимости от сетевого интерфейса. Разрешены только соединения в соответствии с разделом 3.3 Матрица внешних соединений. Прочие соединения блокируются.

Зона **trusted** – источник: все приватные ip адреса (10/8, 172.16/12, 192.168/16) вне зависимости от сетевого интерфейса. Разрешены любые соединения.

3. Планирование установки

Данная инструкция предназначена для установки всех компонентов решения VideoChat на один сервер. Разворачивание решения выполняется с использованием системы автоматического управления конфигурациями ansible

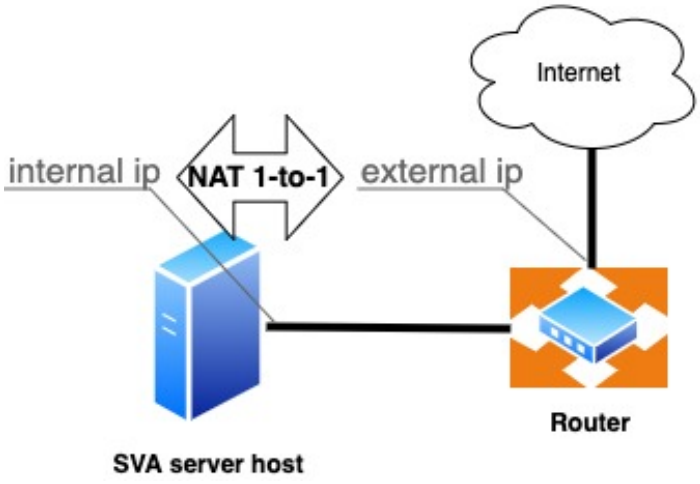
3.1. Перечень задач для установки

1. Убедиться в соответствии предварительным требованиям (раздел 4.1 Необходимые предварительные условия для разворачивания решения VideoChat)
2. Выделить (виртуальный) сервер
3. Установить ansible (раздел 3.4 Стороннее ПО)
4. Настроить параметры установки (раздел 4.2 Настройка параметров установки)
5. Запустить установку решения (раздел 4.3 Установка VideoChat)
6. Проверить инсталляцию и состояние системы после установки

3.2. Сетевая конфигурация для установки решения

Решение VideoChat поддерживает установку в следующей сетевой конфигурации:

- Один локальный сетевой интерфейс с ip адресом из приватного диапазона (internal_ip)
- Публичный ip адрес (external_ip) на маршрутизаторе с трансляцией NAT 1-к-1
- Все операционные взаимодействия с решением VideoChat осуществляются ТОЛЬКО через публичный ip адрес
- Допустимо подключение других локальных сетевых интерфейсов с целью доступа для администрирования сервера, но не получения доступа к сервисам VideoChat



3.3. Матрица внешних соединений

Вектор взаимодействия		Протокол	Адрес источника	Адрес назначения	Порт назначения
Администраторы	Сервер VideoChat	TCP (ssh)	На усмотрение администраторов сервера	Internal_ip, external_ip (опционально)	22
Агенты, Клиенты, Администраторы	Admin-panel, frontend agent, frontend user, frontend desktop_user	TCP (https)	Любой интернет адрес	external_ip	443
Агенты, клиенты	frontend agent, frontend user, frontend desktop_user	TCP,UDP (STUN, TURNs сигнализация)	Любой интернет адрес	external_ip	5349
Агенты, Клиенты	frontend agent, frontend user, frontend desktop_user	UDP (TURN-server RTP, DTLS+STRP, DTLS+SCTP)	Любой интернет адрес	external_ip	20000-29999 50000-59999
Сервер VideoChat	Шлюзы доставки сообщений	TCP (https)	internal_ip	Telegram Bot API (https://api.telegram.org) SMSC.ru API (https://smsc.ru)	443
Telegram Bot API	Сервер VideoChat	TCP (https)	https://api.telegram.org	external_ip	443
Сервер VideoChat	Сервис геокодирования	TCP (https)	internal_ip	https://nominatim.openstreetmap.org/reverse	443
Сервер VideoChat	Letsencrypt API	TCP (https)	Internal_ip	acme-v02.api.letsencrypt.org	443

Letsencrypt API	Сервер VideoChat	TCP (http)	acme-v02.api.letsencrypt.org	external_ip	80

3.4. Стороннее ПО

Для разворачивания решения VideoChat необходима установка ПО ansible версии не ниже 2.10.4. Установка прочих сторонних компонент, как и настройки сервера, осуществляется автоматически, при инсталляции.

Для возможности установки стороннего ПО Сервер должен иметь доступ к стандартным репозиториями (или их зеркалам):
Для RedOS:

- RedOS-Base

3.4.1. Установка ansible на RedOS

Под пользователем, который будет устанавливать VideoChat выполнить команды

```
pip3 install --user --upgrade pip
pip3 install --user ansible
```

4. Установка VideoChat

4.1. Необходимые предварительные условия для разворачивания решения VideoChat

Ниже перечислены необходимые требования, которые должны быть выполнены ДО установки решения VideoChat

Сущность	Требования	Примечание
Сервер	CPU: 2 cores @2.0GHz RAM: 8GB HDD: 30GB	Минимальные требования для запуска и проверки функциональности
Операционная система	RedOS 7.3.1 MUROM	Прочие ОС не протестированы и не поддерживаются. Рекомендуется «чистая» установка системы с минимально необходимыми настройками

Хранилище	Опционально (не требуется для тестов) подключить к серверу в точку монтирования /opt/sva/storage	Реальные размеры хранилища должны рассчитываются исходя из планируемой нагрузки
external_ip	обязательно	Внешний ip адрес, на который настроена NAT трансляция на internal_ip
internal_ip	Требуется указание, если сервер имеет более одного сетевого интерфейса	Указывается именно тот внутренний ip адрес, на который настроена трансляция NAT с external_ip
FQDN	Обязательно. Выполнить заблаговременно, чтобы DNS записи могли обновиться на всех DNS серверах.	Публично разрешаемая в external_ip A-запись в DNS. Это URL, по которому будет осуществляться доступ к разворачиваемому решению
Файл лицензий	Опционально (запрашивается в Altuera)	По-умолчанию система поставляется с файлом лицензий на одного агента
TLS сертификат	Обязательно, если НЕ используется система автоматического выпуска сертификатов letsencript	Имя файла: FQDN_fullchain.cer где FQDN соответствует указанному выше формат: PEM содержимое: сертификат + полная цепочка intermediate CA
Ключ к TLS сертификату	Обязательно, если НЕ используется система автоматического выпуска сертификатов letsencript	Имя файла: FQDN.key где FQDN соответствует указанному выше Формат: PEM без пароля
Адрес e-mail	Опционально (если используется система автоматизированного	Требуется, если используется система автоматической

	выпуска сертификатов letsencrypt)	генерации сертификатов letsencrypt
Логин и пароль к УЗ SMSC.ru, имя отправителя для сервиса SMSC.ru	Опционально, если настраивается смс-шлюз SMSC.ru	Для получения требуемых данных необходимо зарегистрироваться на сервисе https://smsc.ru и следовать его документации
Telegram bot name	Опционально, если используется шлюз доставки сообщений телеграм	Для создания бота телеграм см раздел 4.2.3.1 Настройка шлюза Telegram
Telegram bot username	Опционально, если используется шлюз доставки сообщений телеграм	
Telegram bot link	Опционально, если используется шлюз доставки сообщений телеграм	
Telegram bot token	Опционально, если используется шлюз доставки сообщений телеграм	

Примечание: Установка без требований к FQDN и сертификатов возможна, но функциональность WebRTC работать не будет в связи со спецификой требований к её реализации в браузерах.

4.2. Настройка параметров установки

Перед установкой необходимо скопировать дистрибутив на сервер и распаковать архив командой `tar -xzf VideoChat_v4.2.0.tgz` и зайти в директорию `cd VideoChat_v4.2.0` Все дальнейшие действия по настройке параметров установки и пути к файлам указаны относительно данной директории.

4.2.1. Настройка параметров сервера

Для настройки параметров сервера необходимо отредактировать первую строку файла `./local/hosts`: (занчение параметра указывается после знака = без пробела)

- `fqdn=` указать полное доменное имя сервера
- `external_ip=` указать внешний ip адрес
- `internal_ip=` указать внутренний ip адрес (можно удалить, если на сервере всего один сетевой интерфейс)

4.2.2. Настройка TLS сертификатов

При использовании предварительно выпущенных сертификатов:

- Создать папку `mkdir -p ./local/.data/certificates`
- скопировать файлы сертификата и ключа в папку `./local/.data/certificates:`
  - `fqdn_fullchain.cer`
  - `fqdn.key`

где `fqdn` – полное доменное имя сервера

установить параметр `acme=false` в первой строке файла `./local/hosts`

При использовании автоматизированной системы генерации сертификатов первой строке файла `./local/hosts` задать следующие параметры:

- `acme=true`
- `acme_email=` e-mail адрес администратора (на него будет зарегистрирована УЗ на acme сервере)

4.2.3. Настройка шлюза отправки сообщений

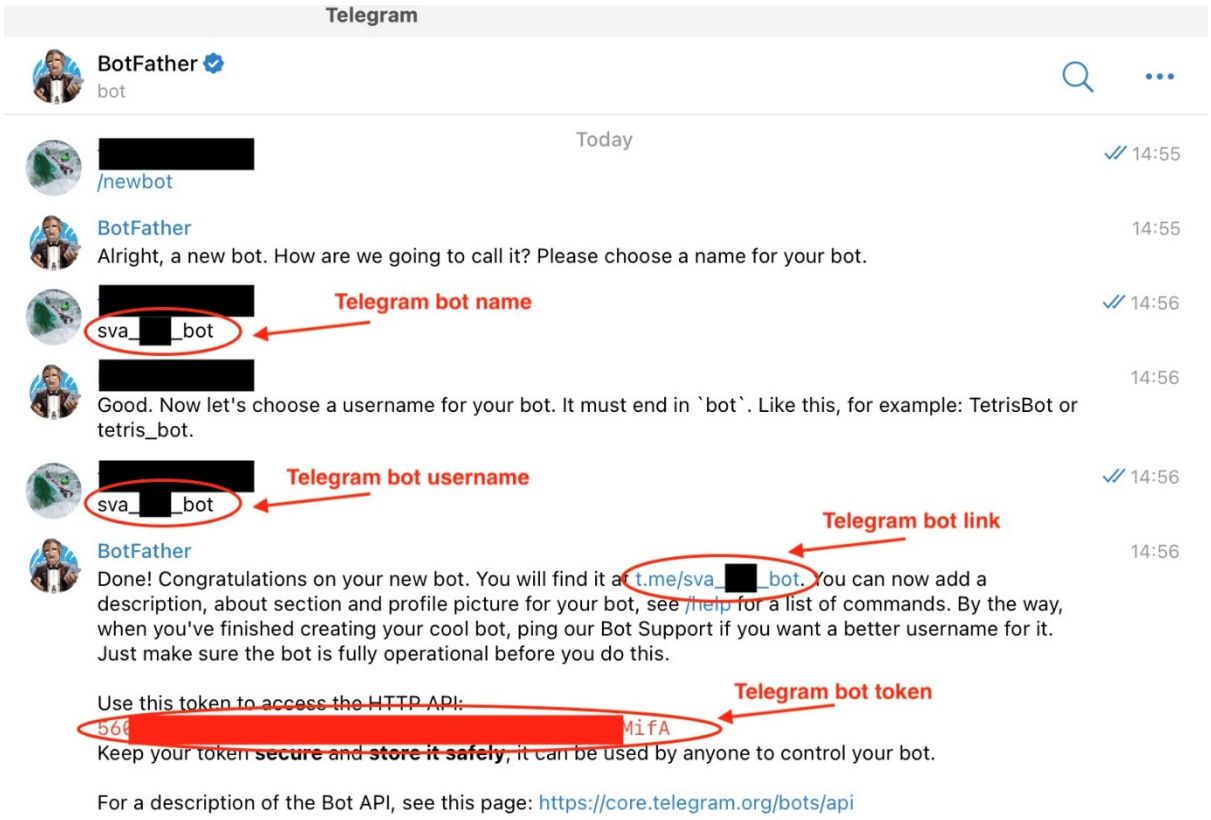
Решение VideoChat поддерживает разные средства доставки коротких ссылок до Клиентов. При необходимости под требования заказчика могут разрабатываться персональные шлюза, из настройка выходит за рамки этого описания. Из публичных сервисов поддерживается:

- Отправка ссылок через СМС – провайдер SMSC.ru (https://smsc.ru)
- Отправка ссылок через телеграмм-бот – используется для тестирования

4.2.3.1. Настройка шлюза Telegram

Шлюз телеграмм рекомендуется использовать только в тестовых целях, поскольку он позволяет осуществлять отправку сообщений только подписанным на него пользователям.

1) Для использования шлюза телеграмм необходимо создать бота телеграмм, для этого подключиться к боту https://t.me/BotFather и создать нового бота:



2) в файле ./local/group_vars/vc_msg указать следующие настройки:

```
vc_msg_providers: |2

DEFAULT {
      type = "telegram"
      description = "Default SMS Provider"
      telegram {
        message-prefix = "Подключись к Altuera Smart Video Agent "
        bot {
          link = "Telegram bot link"
          token = "Telegram bot token"
          name = "Telegram bot name"
          username = "Telegram bot username"
          button-text = "Отправить мой контакт"
          start-message = "Поделитесь своим контактом, чтобы получать ссылки от Altuera Smart Video Agent!"
        }
      }
    }
```

Где:

- значения Telegram bot link, Telegram bot token, Telegram bot name и Telegram bot username – соответствуют полученным при создании бота

- button-text – текст кнопки бота, запрашивающей контактную информацию пользователя
- start-message – первое сообщение, при регистрации в боте
- В поле message-prefix указать текст сообщения, который предшествует ссылке

4.2.3.2. Настройка шлюза SMSC.ru

Для настройки шлюза SMSC необходимо в файле ./local/group_vars/vc_msg указать следующие настройки:

```
vc_msg_providers: |2

    DEFAULT {
     type="smsc-ru-api"
     description="Default SMS Provider"
     smsc {
       host = "https://smsc.ru/sys/send.php"
       status-host = "https://smsc.ru/sys/status.php"
       login = "SMSC_LOGIN"
       psw = "SMSC_PASSWORD"
       id = ""
       sender = "SMSC_SENDER"
       message-prefix = "Подключись к видео-поддержке "
       tinyurl = 0
     }
    }
```

Где:

- SMSC_LOGIN – логин в SMSC.ru
- SMSC_PASSWORD – пароль для SMSC.ru
- SMSC_SENDER – имя для отправки SMSC.ru
- В поле message-prefix указать текст СМС, который предшествует ссылке

4.2.4. Настройка логирования

Настройки логирования задаются в файле ./local/group_vars/all со следующими параметрами:

- log__maxsize: 100MB (максимальный размер одного файла)
- log__history: 31 (количество дней для хранения лог файлов)
- log__totalsize: 1GB (максимальный суммарный объем файлов одного программного компонента)
- log__level: DEBUG (уровень логирования, один из: ERROR, WARN, INFO, DEBUG, TRACE, OFF, ALL)

4.3. Установка VideoChat

Запуск установки осуществляется из директории дистрибутива. Пользователь, осуществляющий установку должен иметь или права суперпользователя или возможность повышения прав до уровня суперпользователя (sudo) с паролем или без пароля

Запуск установки VideoChat суперпользователем или с доступом к sudo без пароля:

```
ansible-playbook deploy_sva.yml
```

Запуск установки VideoChat с доступом к sudo с паролем:

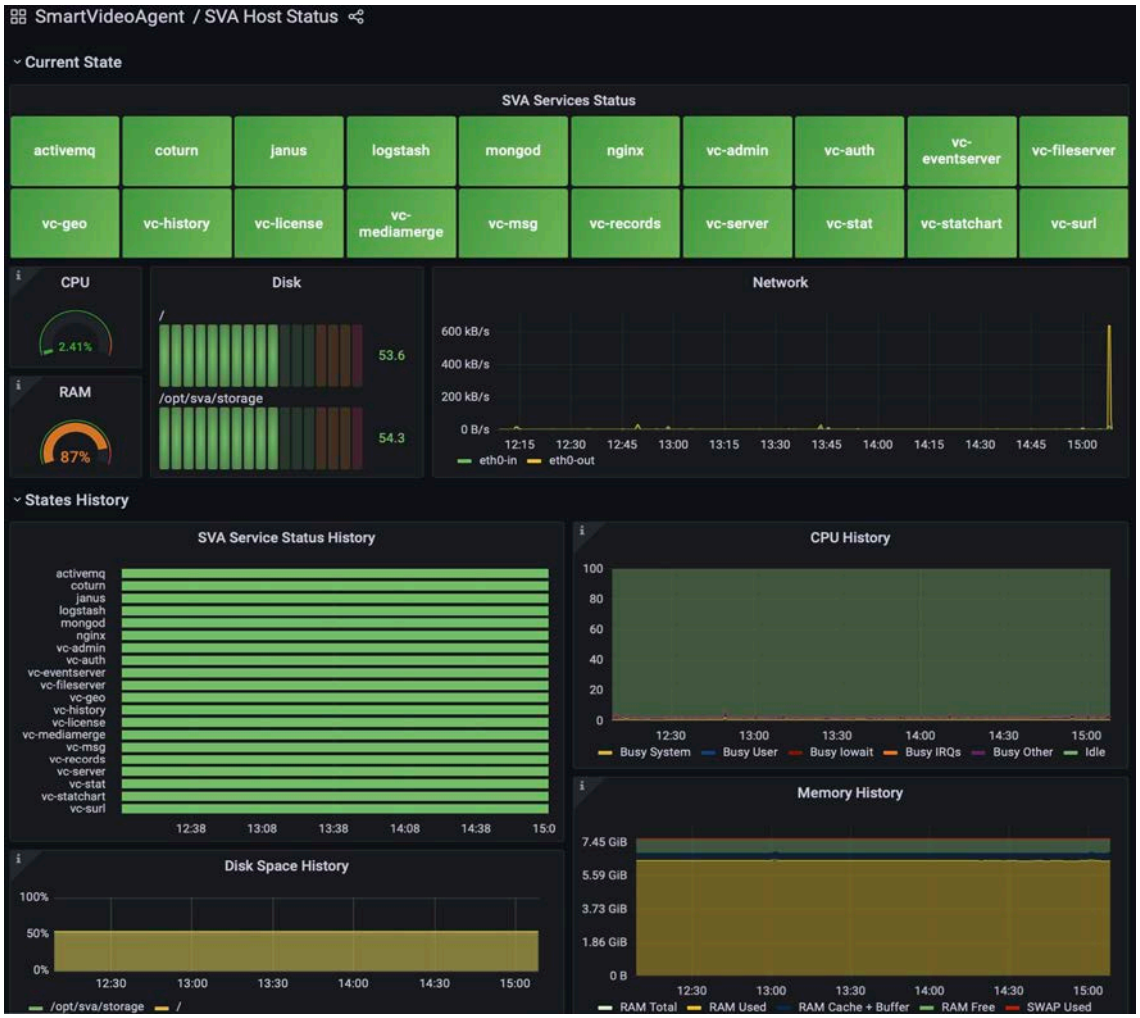
```
ansible-playbook deploy_sva.yml -K
```

и ввести пароль для sudo

4.4. Контроль состояния системы после установки

По окончанию установки открыть систему мониторинга по адресу https://fqdn/graf/dashboards (где fqdn - полное доменное имя сервера VideoChat) и перейти в дашборд VideoChat -> SVA Host Status

убедиться, что все сервисы запущены (зеленые)



4.5. Обновление файлов лицензий

При необходимости установить новые файлы лицензий после установки решения необходимо:

- Скопировать новые (или заменить старые) лицензионные файлы в папку дистрибутива `/local/.data/license`
- Запустить процесс обновления лицензий из папки дистрибутива командой `ansible-playbook deploy_sva.yml --tags vc_license`

При обновлении лицензий возможна кратковременная приостановка сервиса – невозможность создавать новые взаимодействия. Обслуживание текущих взаимодействий останется без изменений.

4.6. Обновление сертификатов

Если выпуск сертификатов автоматизирован (параметр окружения при установке `acme=true`), то перевыпуск и установка сертификатов осуществляется автоматически. При невозможности обновить сертификат, за 14 дней до истечения его срока действия на указанный при установке решения e-mail адрес `acme_email` придёт соответствующее письмо.

Если обновления требуют предварительно выпущенные сертификаты, то для этого необходимо:

- Заменить файлы сертификата и ключа соответственно разделу 4.2.2 Настройка TLS сертификатов
- Запустить процесс установки сертификатов из папки дистрибутива командой `ansible-playbook deploy_sva.yml --tags certs`

5. Описание расположения файлов компонент VideoChat

- Все файлы, относящиеся к проекту расположены в /opt/sva
- Все бэкенд сервисы запускаются под непривилегированным пользователем **sva**
- группа-владелец файлов в /opt/sva - **sva** (установлен setgid bit)
- Лог-файлы
  - nginx /var/log/nginx
  - mongodb /var/log/mongodb
  - coturn /var/log/coturn
  - janus /var/log/janus
  - activemq /opt/activemq/data/activemq.log
  - logstash /var/log/logstash
  - grafana /var/log/grafana
  - логи компонент VideoChat /var/log/sva -> /opt/sva/log

Структура каталога SVA

- backend - директория расположения сервисов VideoChat
  - vc-* - директория с названием сервиса
    - conf - директория с файлами конфигурации сервиса
      - application.conf, application.yaml - основной файл конфигурации сервиса
      - logback.xml - файл конфигурации логов сервиса
      - прочие файлы конфигурации
    - log - символическая ссылка на директорию лог-файлов /opt/sva/log/backend/vc-*
    - service.sh - файл запуска сервиса
- frontend - директория расположения статического веб контента
  - admin-panel – административный интерфейс
  - agent – интерфейс агента
  - user – интерфейс мобильного клиента
  - desktop_user – интерфейс клиента для браузеров ПК
- log - директория с лог-файлами
  - backend - логи сервисов VideoChat
  - elk - логи, клиентских веб-интерфейсов, собираемые logstash

- mongo - директория хранения данных mongodb
- nginx - директория конфигов nginx для VideoChat
- storage - примонтированный диск для хранения файлов
  - records - целевая директория для записи вызовов (janus-gateway)
  - fileserver - целевая директория для хранения файлов vc-fileserver
- prometheus - директория данных prometheus
- grafana - директория данных и провиженинга grafana