

Система Регистрации Диспетчерских Команд АО «СО ЕЭС»

ИНСТРУКЦИЯ ПО УСТАНОВКЕ И НАСТРОЙКЕ РАСПРЕДЕЛЕННОЙ ЧАСТИ СРДК (ПРИКЛАДНОЕ ПО)

Редакция 1.2 от 01.08.2024

Москва, 2024

Содержание

1. Настройка сервера для запуска приложений системы(все сервера приложений)	3
1.1 Подготовка конфигураций приложения	3
1.2 Установка приложений	6
1.3 Настройка отказоустойчивости приложения	7
1.4 Настройка NArpoxu	10
2. Настройка системы в интерфейсе:	11
3. Лист регистрации изменений.....	13

1. Настройка сервера для запуска приложений системы(все сервера приложений)

Для установки сервисов Системы (back-end и front-end) необходимо подготовить рабочие конфигурации приложений, установить сами приложения, а также настроить `keepalived` для обеспечения отказоустойчивости приложения на всех серверах приложений.

1.1 Подготовка конфигураций приложения

Для запуска сервиса необходимо произвести предварительные настройки.

```
#обновить список доступных пакетов  
sudo apt-get update
```

Далее необходимо установить пакет конфигураций.

Далее необходимо перейти в директорию с конфигурацией (`cd ./config/`) и произвести настройку.

Для настройки конфигурации необходимо скопировать файл с шаблоном конфигурации (`cp ./env-branch-example ./env`) и заполнить в нем переменные, указанные в таблице 4.

Таблица 1. Список переменных, используемых в env-example

Переменные	Пример	Комментарии
\$BRANCH_DB_HOST	ip-address	IP адрес, используется локальный адрес текущего сервера приложения
\$BRANCH_DB_PORT	port	Порт
\$BRANCH_DB_NAME	bd_name	Название базы данных распределенной части
\$BRANCH_DB_USER	bd_user	Пользователь базы данных распределенной части
\$BRANCH_DB_PASS	password	Пароль пользователя базы данных распределенной части
\$BACK_END_PORT	port	Порт, на котором работает бэкэнд
\$FRONT_END_PORT	port	Порт, на котором работает фронтэнд
\$FRONT_END_SSL_PORT	port	Порт, на котором работает фронтэнд на защищенном протоколе
\$LDAP_ODUXX_URL	domen.local	Адрес расположения ldap сервера
\$AUTH_ODUXX_USER	domen\login	УЗ используемая для сканирования ldap каталогов, и авторизации пользователей
\$AUTH_ODUXX_PASS	password	Пароль от УЗ AUTH_ODUXX_USER
\$SEARCH_GROUP_ODU_XX	OU=SRDK,	LDAP Путь по каталога с группой доступа ADMIN_SRDK_GROUP
\$SERVER_IP_ADDRESS	ip-address	IP адрес виртуального интерфейса
\$CENTER_IP_ADDRESS	ip-address	IP адрес сервера центральной части
\$DC_NAME	ЮЖНОЕ ОДУ	имя ДС из корпоративной НСИ, и полностью посимвольно соответствовать ей.
\$SERVER_ID	ODUVOST	Уникальная аббревиатура, однозначно определяющая ваш филиал
\$SK11_USER	domen\login	УЗ используемая для подключения к СК-11
\$SK11_PASS	password	Пароль от УЗ SK11_USER
\$SK11_MODEL	xxxxxxxx-0000-0000-xxxx-000000xxxxxx	Идентификатор объектной модели СК-11 к объектам которой будут осуществляться доступ.
\$ADMIN_SRDK_GROUP	group	Группа доступа

		администраторов к системе
\$ELK_USER	login	Учетная запись системы сбора логов
\$ELK_PASSWORD	password	Пароль УЗ ELK_USER
\$ESM_ADDRESS	ip-address	Адрес подключения к системе ЕСМ
\$ESM_PORT	port	Порт подключения к системе ЕСМ
\$ESM_COMMUNITY	public	Сообщество чтения ЕСМ
\$ESM_APPOID	5560000	OID системы в ЕСМ
\$PASSWORD_ENCODE_PASSWORD	1111	Ключ шифрования паролей в БД
\$PASSWORD_ENCODE_SALT	2222	Дополнительный ключ шифрования паролей в БД
\$APP_MODES_BASEURL	http://server-name.local:port	Адрес подключения к системе модес терминал.
\$MAIL_DEFAULT_HOST	server-name.local	Адрес почтового сервера
\$MAIL_DEFAULT_PORT	port	Порт почтового сервера
\$MAIL_DEFAULT_AUTH	true	Использование авторизации
\$MAIL_DEFAULT_USERNAME	login	УЗ для подключения к почтовому серверу
\$MAIL_DEFAULT_PASSWORD	password	Пароль для подключения к почтовому серверу
\$MAIL_DEFAULT_FROM	login@domen.local	Имя почтового ящика, с которого производится отправка почтовых уведомлений
\$MAIL_DEFAULT_STARTTLS	true	Использование TLS-шифрования
\$APP_MEM_USAGE	X G	Использование максимального количества оперативной памяти приложением. Необходимо ½ от всей.
\$APP_MODES_ID_LAYER	id	Идентификатор слоя ПБР-24
\$GRAPH_TIMEZONE	3	Часовой пояс для отображения графиков
\$APP_MODES_MAX_IN_MEMORY_KBYTES	XXXX	объем данных полученного из modes
\$APP_HTTP_REQUEST_LOGGING_ENABLED	true	Включает логирование входящих запросов.
\$APP_HTTP_REQUEST_LOGGING_STORAGE	/var/log/SRDK/xxxx	Директория для хранения входящих запросов
\$APP_HTTP_REQUEST_LOGGING_ENDPOINTS	/api/v1/xxxx/saveDataPG	Список API URI, по которым необходимо для логирования
\$APP_HTTP_REQUEST_LOGGING_AGE	P7D	Максимальный срок хранения файлов логов

\$APP_TIME_ZONE	Europe/Moscow	Часовой пояс
\$APP_SK11_MONITORING_ENABLED	false	Включен ли мониторинг
\$APP_SK11_MONITORING_STORAGE	/var/log/SRDК/xxxx	Директория хранения логов в СК-11
\$APP_SK11_MONITORING_AGE	P1D	Максимальный срок хранения файлов логов
\$APP_SK11_MONITORING_ENDPOINT_PATTERNS	'/**/xxxx/**'	Список шаблонов API запросов для мониторинга
\$APP_DISTRIBUTED_PART_READ_TIMEOUT	120000	Время ожидания входящего трафика от распределенной части в миллисекундах.
\$GRAPH_USE_GOU_ZONE	false	Определение временной зоны для ГОУ графика.
\$APP_SK11_CERT		Зашифрованный контейнер
\$APP_CENTRAL_PART_URL_SUFFIX	""	Эту переменную необходимо добавить, но значение присвоить пустое.
\$APP_CENTRAL_PART_URL_SUFFIX		Зашифрованный контейнер
\$APP_CENTRAL_PART_SSL_KEY_STORE_PASSWORD		Пароль от контейнера

Необходимо заполнить все поля LDA_XX_URL, AUTH_XX_USER, AUTH_XX_PASS и SEARCH_GROUP_ODU_XX т.к. это необходимо для корректного запуска системы. Значение можно использовать те же, что и для вашего домена.

Далее необходимо внести правки в конфигурационный файл балансировщика:

Переходим в каталог nginx

```
cd ./nginx/
```

Копируем пример конфигурации

```
cp ./default.conf.example ./default.conf
```

Затем исправляем в полученном файле переменные:

`${SERVER_IP_ADDRESS}` – адрес виртуального интерфейса

`${URL}` – доменное имя системы

Затем необходимо добавить сертификат и приватный ключ для корректной работы https. Сертификат и ключ необходимо положить в каталог ./config/nginx/.

1.2 Установка приложений

Для установки всех приложений достаточно запустить скрипт установки:

```
# Переходим в каталог проекта
cd ./config
# Повышаем привелегии пользователя
sudo -s
# Запуск скрипта установки
./start-script.sh
```

Данный скрипт проверит уже установленные сервисы, и установит свежие версии приложений.

1.3 Настройка отказоустойчивости приложения

Данные действия аналогично выполняются на всех серверах приложений с незначительными отличиями в конфигурации. Для реализации отказоустойчивости, необходимо выделить дополнительный виртуальный IP адрес, через который и будет производиться подключение пользователей к системе. В дальнейшем для настройки будут использоваться следующие переменные, которые необходимо будет заменить в конфигурациях:

- \$IP_MASTER – IP адрес основного сервера;
- \$IP_SLAVE – IP адрес резервного сервера;
- \$VIRT_IP – основной IP адрес системы.

Для установки системы необходимо установить keepalived:

```
apt-get update
apt-get install keepalived
```

Далее создаем скрипт для проверки состояния приложения /etc/keepalived/script.sh следующего содержания:

```
#!/bin/bash
lockfile=/tmp/keepalived_lock

if ( set -o noclobber; echo "$$" > "$lockfile") 2> /dev/null;
then

front_code=$(curl --noproxy \* -k -s -o /dev/null -w %{http_code}
-m 1 https://localhost)
back_code=`curl --noproxy \* -s -m 1 /dev/null
http://localhost:port/api/v1/actuator/health/keepalived|\python
3 -c "import sys, json; print(json.load(sys.stdin)['status'])"`
if [[ $front_code -eq 200 && $back_code == UP ]]
then
    echo 0 > "/tmp/chk_keep_status"
    rm -f "$lockfile"
    trap - INT TERM EXIT
    exit 0
else
    echo 1 > "/tmp/chk_keep_status"
    rm -f "$lockfile"
    trap - INT TERM EXIT
    exit 1
fi
```

```
else
    echo "Failed to acquire lockfile: $lockfile."
    kill -9 $(cat $lockfile) 2>/dev/null
    rm -f "$lockfile"
    exit $(cat /tmp/chk_keep_status)
fi
```

После создания файла, нужно разрешить для данного файла выполнение, командой:

```
chmod 777 /etc/keepalived/script.sh
```

Далее создаем непосредственно конфигурационный файл /etc/keepalived/config.conf:

```
vrrp_script chk_app {
    script "/etc/keepalived/script.sh"
    user nobody
    interval 2
    fall 3
    rise 3
    timeout 3
}

global_defs {
    router_id uMASTER
    enable_script_security
}

vrrp_instance VI_1 {
    state MASTER
    interface eth0
    virtual_router_id 230
    priority 101
    advert_int 1
    unicast_src_ip $IP_MASTER
    unicast_peer {
        $IP_SLAVE
    }

    virtual_ipaddress {
        $VIRT_IP/32 dev eth0 label eth0:1
    }

    track_script {
        script
    }
}
```

Конфигурация резервного сервера аналогична, но с незначительными отличиями:

```
vrrp_script chk_app {
```

Система Регистрации Диспетчерских Команд АО «СО ЕЭС»

```
script "/etc/keepalived/script.sh"
user nobody
interval 2
fall 3
rise 3
timeout 3
}

global_defs {
    router_id uBACKUP
    enable_script_security
}

vrrp_instance VI_1 {
    state BACKUP
    interface eth0
    virtual_router_id 230
    priority 100
    advert_int 1
    unicast_src_ip $IP_SLAVE
    unicast_peer {
        $IP_MASTER
    }

    virtual_ipaddress {
        $VIRT_IP /32 dev eth0 label eth0:1
    }

    track_script {
        script
    }
}
```

После сохранения конфигурационных файлов необходимо запустить службу `keepalived`, а также настроить автоматический запуск при старте системы:

```
systemctl start keepalived
systemctl enable keepalived
```

Для проверки установки необходимо выполнить команду:

```
systemctl status keepalived | grep active
```

Ожидаемый ответ:

```
Active: active (running)
```

Если в логах будет ошибка выполнения скрипта, потребуется проверить, доступна ли на сервере команда `python3`:

```
curl --noproxy \* -s -m 1 /dev/null
```

```
http://localhost:port/api/v1/xxxx/keepalived|\python3 -c "import sys,
json; print(json.load(sys.stdin) ['status'])"
```

Если при запуске будет ошибка – нужно будет проверить, что python3 установлен и работает.

1.4 Настройка HAProxy

Для настройки сервиса необходимо установить haproxy:

```
apt-get install haproxy -y
```

и внести конфигурацию в файл /etc/haproxy/config.txt:

```
global
    maxconn 100

defaults
    log global
    mode tcp
    retries 2
    timeout client 30m
    timeout connect 4s
    timeout server 30m
    timeout check 5s

listen stats
    mode http
    bind *:port
    stats enable
    stats uri /

listen postgres
    bind *:port
    option httpchk
    http-check expect status 200
    default-server inter 3s fall 3 rise 2 on-marked-down
shutdown-sessions
    server node1 <ip node1>:port maxconn 100 check port
    server node2 <ip node2>:port maxconn 100 check port
    server node3 <ip node3>:port maxconn 100 check port
```

В указанной конфигурации необходимо заменить параметры <ip node1,2,3> - на ip-адреса узлов кластера СУБД

После этого перезапускаем сервис:

```
systemctl restart haproxy
systemctl enable haproxy
```

Для проверки установки необходимо выполнить команду:

```
systemctl status haproxy | grep active
```

Ожидаемый ответ:

```
Active: active (running)
```

Чтобы убедиться в отсутствии ошибок, необходимо через несколько минут после завершения установки выполнить команду:

```
docker ps
```

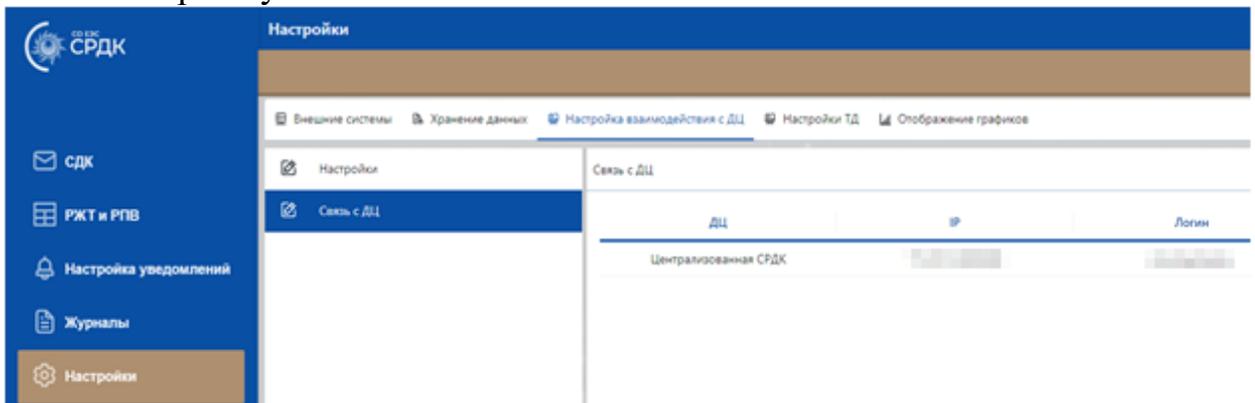
В выводе должно быть 2 контейнера с имена соответствующими названиям сервисов, а именно: back-end и front-end, и в колонке STATUS у каждого из них должно быть значение – UP N minutes (где N количество минут после установки).

2. Настройка системы в интерфейсе:

- В интерфейсе распределенной части Системы необходимо настроить взаимодействие с централизованной частью Системы:

IP = ip-address:port

Логин и пароль уточнить в СВР ТС ИА



- Необходимо создать группу AD для доступа к API распределенной части ИУС СРДК. Необходимо создать сервисного пользователя для доступа к API распределенной части ИУС СРДК и включить его в созданную группу AD для доступа к API.
- Необходимо в интерфейсе распределенной части ИУС СРДК назначить роль ROLE_REMOTE_ACCESS для созданной в п.3 группы AD.
- Необходимо сообщить логин/пароль пользователя созданного в п.3 в СВР ТС ИА
- Необходимо в интерфейсе ИУС СРДК указать параметры подключения к ИУС СРПГ, СК11 локального ДЦ

- Так же в настройках Modes Terminal ввести адрес стенда для взаимодействия. В СК11 включить все передачи данных (ползунок), а также, ввести URL, логин и пароль от соответствующего стенда.
- В разделе СК11, Получение телеметрии по РГЕ есть поле «Периодичность». Там необходимо указать желаемую периодичность запросов в СК11. По умолчанию стоит 15 секунд. При желании, для разгрузки системы можно увеличить данный параметр.

Пример настроек для распределенной части ИА указан ниже:

The image shows two screenshots of a web-based configuration interface. The top screenshot displays the configuration for 'ПАК СРПГ' (SRPG package). The left sidebar lists 'ПАК Modes Terminal', 'ОИК СК-11', and 'ПАК СРПГ'. The main area shows 'Состав настроек: ПАК СРПГ' and 'Настройки ПАК СРПГ'. Fields include 'Адрес сервиса:' with the value 'srpg-control/api/v1/ext/collation/ess/find', 'Логин:' (masked), and 'Пароль:' (empty). The bottom screenshot shows the configuration for 'ОИК СК-11'. The left sidebar lists 'ПАК Modes Terminal', 'ОИК СК-11', and 'ПАК СРПГ'. The main area shows 'Состав настроек: ОИК СК-11' and 'Настройки ПАК СРПГ'. Fields include 'Адрес сервиса:', 'Логин:', and 'Пароль:' (masked).

3. Лист регистрации изменений

№п/п	Автор	Редакция	Дата	Описание изменения