



РУКОВОДСТВО АДМИНИСТРАТОРА

СЛЕТ.10001-01 90 06

Версия 4.1.1. Выпуск от июня 2023

Настройка компонента «Оркестратор»

ОГЛАВЛЕНИЕ

1 . ОБЩИЕ СВЕДЕНИЯ.....	3
1.1 . О документе.....	3
1.2 . Назначение компонента «Оркестратор»	3
1.3 . Требования к программному и аппаратному обеспечению	3
1.4 . Типографские соглашения	4
2 . УСТАНОВКА И УДАЛЕНИЕ КОМПОНЕНТА	5
2.1 . Настройки среды функционирования	5
2.2 . Получение пакетов установки в ОС Astra Linux Special Edition	5
2.3 . Установка Оркестратора.....	7
2.4 . Удаление Оркестратора	8
2.5 . Обновление Оркестратора	8
3 . НАСТРОЙКА КОМПОНЕНТА	9
3.1 . Настройка apache2	9
3.2 . Настройка Оркестратора	9
3.3 . Проверка заданных настроек	11
3.4 . Запуск служб	12
3.5 . Проверка работоспособности	12
3.6 . Журналирование	14
4 . ПЕРЕЧЕНЬ СОКРАЩЕНИЙ	15

1 . ОБЩИЕ СВЕДЕНИЯ

1.1 . О документе

Настоящий документ является шестой частью руководства администратора на программный комплекс «Диспетчер подключений виртуальных рабочих мест Termidesk» (далее - Termidesk). Документ предназначен для администраторов системы и сети.

В этом руководстве приведено назначение, установка и настройка компонента «Оркестратор». Для того, чтобы получить информацию о месте компонента в программном комплексе, необходимо обратиться ко второй части руководства администратора - СЛЕТ.10001-01 90 02 «Руководство администратора. Настройка программного комплекса».

1.2 . Назначение компонента «Оркестратор»

Компонент «Оркестратор» (далее - Оркестратор) входит в состав Termidesk.

Оркестратор отвечает за согласованную работу всех компонентов Termidesk при децентрализованном развертывании в целях отказоустойчивости и комплексирования с облачными службами.

Оркестратор построен по архитектуре «Клиент - Сервер», выступая по отношению к облачным службам как в роли клиента, так и в роли сервера. Оркестратор также является API-шлюзом: управляющие запросы, пришедшие из облачной инфраструктуры, транслируются (транзитом) в запросы к Termidesk, а ответы (включая ошибки) возвращаются в облачную инфраструктуру.

1.3 . Требования к программному и аппаратному обеспечению

Для установки Оркестратора минимальные аппаратные требования узла должны соответствовать следующим:

- процессор архитектуры Intel x86 с разрядностью 64 бит;
- оперативная память, не менее 4 ГБ ;
- свободное дисковое пространство, не менее 1 ГБ;
- сетевое соединение, не менее 100 Мбит/с.

Среда функционирования Оркестратора должна соответствовать следующим требованиям:

- должна быть установлена операционная система (ОС) Astra Linux Special Edition версии 1.7 и выше;
- должен быть установлен Python 3.7.3 из состава ОС;
- должен быть доступен API компонента Keystone из состава OpenStack для проверки токена приложения;

- должен быть доступен API диспетчера Termidesk.

1.4 . Типографские соглашения

В настоящем документе приняты следующие типографские соглашения:

- моноширинный шрифт – используется для выделения фрагментов текста программ, наименований файлов и папок (директорий), путей перемещения, строк комментариев, различных программных элементов (объект, класс, тип, переменная, команда, макрос и т. д.), а также вводимого и выводимого текста в режиме командной строки;
- «кавычки» – текст, заключенный в кавычки, используется для обозначения наименований документов, названий компонентов Termidesk, пунктов меню, наименований окон, вкладок, полей, других элементов графического интерфейса, а также вводимого и выводимого текста в режиме графического интерфейса;
- **[квадратные скобки]** – текст, заключенный в квадратные скобки, используется для наименования экранных кнопок;
- **<угловые скобки>** – текст, заключенный в угловые скобки, используется для наименования клавиш клавиатуры.

2. УСТАНОВКА И УДАЛЕНИЕ КОМПОНЕНТА

2.1. Настройки среды функционирования

Узлу, на который будет установлен Оркестратор, необходимо присвоить имя (hostname) при помощи команды:

```
~$ sudo hostname orchestrator.test.lan
```

где `orchestrator.test.lan` - назначаемое узлу имя.

Для того, чтобы после перезагрузки операционной системы (ОС) назначенное имя сохранилось, необходимо внести его в файл `/etc/hosts`.

Пример файла `/etc/hosts` с указанием полного и короткого имени узла:

```
127.0.0.1      localhost
192.168.3.183 orchestrator.test.lan orchestrator
```

2.2. Получение пакетов установки в ОС Astra Linux Special Edition

Дистрибутив представлен бинарным файлом пакета программного обеспечения (ПО) в `deb`-формате. Установка в ОС Astra Linux Special Edition производится из локального репозитория, который распространяется в формате `iso`-образа.

Получить `iso`-образ можно двумя способами:

- заполнить запрос на получение образа через сайт Termidesk: <https://termidesk.ru/support/#request-support>;
- через личный кабинет: <https://lk-new.astralinux.ru/>.

Для подключения локального репозитория Termidesk на узле, где предполагается установка, нужно выполнить следующее:

- скопировать в домашний каталог пользователя образ диска `termidesk-vdi.XXXXX.iso`, где `XXXXX` — версия сборки;
- подключить образ диска к файловой системе в каталог `/mnt`:

```
~$ sudo mount -o loop termidesk-vdi.XXXXX.iso /mnt
```

где:

`-o loop` - параметры для привязки петлевого устройства (`/dev/loop`) к файлу `termidesk-vdi.XXXXX.iso`, устройство затем монтируется в указанный каталог `/mnt`;

- скопировать содержимое каталога `repos` подключенного образа диска в каталог `/var` локальной файловой системы:

```
~$ sudo cp -Rp /mnt/repos /var
```

где:

`-Rp` - ключ для рекурсивного копирования подкаталогов и файлов с сохранением исходных свойств;

- отключить подключенный ранее образ диска от узла:

```
~$ sudo umount /mnt
```

- установить пакет `lsb-release`:

```
~$ sudo apt install -y lsb-release
```

где:

`-y` - ключ для пропуска подтверждения установки;

- добавить локальный репозиторий Termidesk (`/var/repos/astra`) в файл `/etc/apt/sources.list.d/termidesk_local.list` через командный интерпретатор `sh`:

```
~$ sudo sh -c 'echo "deb file:/var/repos/astra $(lsb_release -cs) non-free" > /etc/apt/sources.list.d/termidesk_local.list'
```

где:

`-c` - ключ для чтения команд из вводимой строки (стандартный ввод);

`echo` - команда вывода текста, совместно с символом «>» используется для перенаправления строки `deb file:/var/repos/astra $(lsb_release -cs) non-free` в файл `/etc/apt/sources.list.d/termidesk_local.list`;

`deb file:/var/repos/astra $(lsb_release -cs) non-free` - добавляемый репозиторий, вложенная команда `$(lsb_release -cs)` подставляет версию - 1.7_x86-64;

- выполнить поиск ключа репозитория Termidesk `GPG-KEY-PUBLIC` и добавить его в ОС:

```
~$ cat /var/repos/astra/GPG-KEY-PUBLIC | sudo apt-key add -
```

- убедиться, что ключ с `uid` «`release@uveon.ru`» был успешно добавлен:

```
~$ apt-key list
```

⚠ В случае, если ключ не отображен в выводе команды, необходимо убедиться, что ключ `GPG-KEY-PUBLIC` существует:

```

:~$ cat /var/repos/astra/GPG-KEY-PUBLIC
    
```

Команда выведет содержимое ключа в формате Base64.

Если ключ все же существует, необходимо проверить правильность выполнения шагов по добавлению репозитория Termidesk в файл `/etc/apt/sources.list.d/termidesk_local.list`.

- обновить данные пакетного менеджера:

```

:~$ sudo apt update
    
```

Данную команду (`sudo apt update`) необходимо выполнять при каждом изменении списка источников пакетов или при изменении содержимого этих источников.

2.3 . Установка Оркестратора

Перед установкой необходимо подключить локальный репозиторий Termidesk, как приведено выше.

Установка Оркестратора выполняется командой:

```

:~$ sudo apt install termidesk-orchestrator
    
```

Установку можно также выполнить из deb-пакета, не подключая локальный репозиторий, при помощи команды:

```

:~$ sudo apt install /home/user/termidesk-orchestrator_XXX-astra17_amd64.deb
    
```

где `/home/user/termidesk-orchestrator_XXX-astra17_amd64.deb` - путь к deb пакету.

В процессе установки необходимо принять лицензионное соглашение, доступное в псевдографическом интерфейсе. Для возможности использования псевдографического интерфейса должен быть установлен пакет `dialog`.

Дополнительно установятся следующие пакеты (зависимости):

- `apache2`;
- `ssl-cert`;
- `memcached`;
- `openssl`;
- `libcurl3-nss`;
- `libcairo2`;
- `gettext`;
- `python3-click`;
- `python3-distutils`.

2.4 . Удаление Оркестратора

Удаление Оркестратора из среды ОС Astra Linux Special Edition 1.7 выполняется командой:

```
:~$ sudo aptitude purge -y termidesk-orchestrator
```

где:

-y - ключ для пропуска подтверждения удаления.

После удаления необходимо очистить оставшиеся зависимости и конфигурации командой:

```
:~$ sudo aptitude purge ~c -y
```

Команда полностью удалит оставшиеся настройки и зависимости уже удаленных пакетов.

2.5 . Обновление Оркестратора

Обновление Оркестратора выполняется установкой новой версии поверх предыдущей.

3. НАСТРОЙКА КОМПОНЕНТА

3.1 . Настройка apache2

Вместе с Оркестратором будет установлен сервер apache2, если он не был установлен ранее.

После установки Оркестратора необходимо скорректировать файл конфигурации веб-сервера / etc/apache2/apache2.conf. Для этого следует найти и раскомментировать строку с параметром AstraMode, далее присвоить данному параметру значение off, точно соблюдая отступы и пробелы в файле.

Пример:

```
# Astra security mode.
#
AstraMode off
```

После перезапустить веб-сервер командой:

```
:~$ sudo systemctl restart apache2
```

3.2 . Настройка Оркестратора

Для настройки оркестратора необходимо отредактировать разделы Termidesk params и OpenStack params в файле /opt/termidesk/share/termidesk-orchestrator/src/config/orc_config.py.

При первой установке этот конфигурационный файл отсутствует. Его можно скопировать из шаблона /opt/termidesk/share/termidesk-orchestrator/src/config/orc_config.py.dist, для этого выполнить следующие действия:

- перейти в каталог с конфигурационными файлами:

```
:~$ cd /opt/termidesk/share/termidesk-orchestrator/src/config
```

- переключиться на пользователя termidesk:

```
:~$ sudo -u termidesk bash
```

- скопировать шаблон в новый файл:

```
:~$ cp orc_config.py.dist orc_config.py
```

- выйти из окружения пользователя termidesk:

```
:~$ exit
```

Параметры внутри разделов конфигурационного файла `/opt/termidesk/share/termidesk-orchestrator/src/config/orc_config.py` имеют следующую структуру:

```
ВНУТРЕННЯЯ_ПЕРЕМЕННАЯ = os.getenv("ПЕРЕМЕННАЯ ОКРУЖЕНИЯ",
"ЗНАЧЕНИЕ_ВВОДИМОЕ_ВРУЧНУЮ")
```

где:

`ПЕРЕМЕННАЯ ОКРУЖЕНИЯ` - наименование переменной окружения, назначенной через команду `export`. Если переменная в окружении не задана, она добавится с указанным значением;
`ЗНАЧЕНИЕ_ВВОДИМОЕ_ВРУЧНУЮ` - новое значение переменной.

Пример файла `/opt/termidesk/share/termidesk-orchestrator/src/config/orc_config.py`:

```
# Termidesk params
TERMIDESK_HOST = os.getenv("TSRV_HOST", '')
TERMIDESK_PORT = os.getenv("TERMIDESK_PORT", "")
TERMIDESK_USERNAME = os.getenv("TSRV_USERNAME", '')
TERMIDESK_PASSWORD = os.getenv("TSRV_PASSWORD", '')
#Termidesk auth domain ID: authSmallName
TERMIDESK_AUTH = os.getenv("TSRV_AUTH", '')
TERMIDESK_AUTH_URL = os.getenv('TSRV_AUTH_URI', '/api/auth/draft/legacy/login')

# OpenStack params
# The name of application credentials
APP_CREDENTIALS_NAME = os.getenv("APP_CREDENTIALS_NAME", "")
# The secret for application credentials
APP_CREDENTIALS_SECRET = os.getenv("APP_CREDENTIALS_SECRET", "")
# The user name for application credentials
APP_CREDENTIALS_USER_NAME = os.getenv("APP_CREDENTIALS_USER_NAME", "")
# The name of user domain for application credentials
APP_CREDENTIALS_USER_DOMAIN = os.getenv("APP_CREDENTIALS_USER_DOMAIN", "")
# The url of application credentials authentication. Example: http://0.0.0.0/identity/v3/
auth/tokens
OPENSTACK_AUTH_URL = os.getenv("OPENSTACK_AUTH_URL", "")
```

Список доступных параметров конфигурационного файла приведен в таблице.

Таблица 1 – Описание параметров конфигурационного файла Оркестратора

Параметр	Назначение	Значение по умолчанию
Параметры универсального диспетчера Termidesk (секция <code>Termidesk params</code>)		
<code>TERMIDESK_HOST</code>	FQDN или IP-адрес универсального диспетчера Termidesk	Не задано
<code>TERMIDESK_PORT</code>	Номер порта универсального диспетчера Termidesk (как правило, 80 или 443)	Не задано

TERMIDESK_USERNAME	Имя пользователя с правами администратора в универсальном диспетчере Termidesk	Не задано
TERMIDESK_PASSWORD	Пароль пользователя с правами администратора в универсальном диспетчере Termidesk	Не задано
TERMIDESK_AUTH	Идентификатор домена аутентификации (authSmallName, метка) в универсальном диспетчере Termidesk, в который входит пользователь TERMIDESK_USERNAME	Не задано
TERMIDESK_AUTH_URL	<p>Данный URL используется для проверки состояния (health check) универсального диспетчера Termidesk.</p> <p>Если страница доступна (получен ответ: HTTP_200_OK), то считается, что универсальный диспетчер Termidesk функционирует правильно</p>	'/api/auth/draft/legacy/login'
Параметры для проверки пароля приложений OpenStack (секция OpenStack params)		
APP_CREDENTIALS_NAME	Имя приложения для подключения к OpenStack	Не задано
APP_CREDENTIALS_SECRET	Ключ-пароль для подключения к OpenStack. Может быть задан или автоматически сгенерирован при создании доступа для приложения. Отображается в интерфейсе OpenStack только один раз, и, в случае утери, требует регенерации	Не задано
APP_CREDENTIALS_USER_NAME	Имя пользователя, предоставляющего доступ приложению. Уровень пользователя должен быть не ниже «member». Возможные варианты уровня: «member», «admin», «owner»	Не задано
APP_CREDENTIALS_USER_DOMAIN	Имя (не идентификатор) домена OpenStack, в который входит пользователь APP_CREDENTIALS_USER_NAME	Не задано
OPENSTACK_AUTH_URL	URL компонента Identity, располагается в OpenStack по пути «Проект - Доступ к API - Identity»	Не задано

3.3 . Проверка заданных настроек

Для проверки корректности заданных настроек применяется исполняемый файл /opt/termidesk/share/termidesk-orchestrator/src/self_check.py.

Для его использования необходимо выполнить команды:

```

:~$ cd /opt/termidesk/share/termidesk-orchestrator/src/
:~$ python3 self_check.py
    
```

Если во время проверки возникает ошибка, подробную информацию о ней можно найти в файле /tmp/termidesk_self_check_crash.log. Файл журнала перезаписывается при каждом запуске проверок.

3.4 . Запуск служб

Запуск служб осуществляется после задания настроек и проверки их корректности.

Добавить в автоматический запуск службу Оркестратора:

```

:~$ sudo systemctl enable termidesk-orchestrator.service
    
```

Затем запустить службу:

```

:~$ sudo systemctl start termidesk-orchestrator.service
    
```

3.5 . Проверка работоспособности

Для проверки работоспособности необходимо:

- получить самоподписанный сертификат для обращения по протоколу HTTPS (опционально);
- получить токен приложения (обязательно).

Для получения самоподписанного сертификата, используемого для обращений по протоколу HTTPS, необходимо воспользоваться командой:

```

:~$ echo quit | openssl s_client -showcerts -servername orchestrator.test.lan -connect
orchestrator.test.lan:443 > /home/user/orchestrator.test.lan.pem
    
```

где:

orchestrator.test.lan - имя узла, на котором установлен Оркестратор;

/home/user/orchestrator.test.lan.pem - путь к pem файлу, в который будет сохранен сертификат.

Токен приложения используется для авторизации запросов пользователя и запрашивается на сервере OpenStack через функционал API.

Для выполнения API-запросов следует установить утилиту curl:

```

:~$ sudo apt install curl
    
```

Для запроса токена необходимо использовать следующую команду:

```

:~$ curl -i \
  -H "Content-Type: application/json" \
  -d '{
    "auth": {
      "identity": {
        "methods": ["application_credential"],
        "application_credential": {
          "name": "orchestrator",
          "secret": "orchestrator_secret_key",
          "user": {
            "name": "admin",
            "domain": { "name": "Default" }
          }
        }
      }
    }
  }' "http://IP-address/identity/v3/auth/tokens" | grep "X-Subject-Token" | sed -e 's/\r//'
  -e 's/\n//'
    
```

где:

IP-address - IP-адрес сервера OpenStack.

Вывод команды будет содержать необходимый токен:

```
X-Subject-Token: gAAAAABjdl0bymw<...cuted out...>-pJ2JcKNDI_s_Fle0LuZ0uyt-r
```

Полученный токен необходимо разместить в переменную окружения APP_TOKEN и подставлять в заголовок «X-ORC-TOKEN-APP» при каждом запросе к Оркестратору.

Для задания переменной окружения следует воспользоваться командой:

```
:~$ export APP_TOKEN="gAAAAABjdl0bymw<...cuted out...>-pJ2JcKNDI_s_Fle0LuZ0uyt-r"
```

Проверка состояния Оркестратора (health check) выполняется запросом через утилиту curl:

```
:~$ curl -s -H "X-ORC-TOKEN-APP: $APP_TOKEN" --cacert orchestrator.test.lan.pem https://
orchestrator.test.lan/api/draft/orc/health/ | python -m json.tool
```

где:

orchestrator.test.lan.pem - ранее сохраненный самоподписанный сертификат;

orchestrator.test.lan - имя узла, на котором установлен Оркестратор.

Проверка состояния универсального диспетчера Termidesk через Оркестратор выполняется запросом через утилиту curl:

```
:~$ curl -s -H "X-ORC-TOKEN-APP: $APP_TOKEN" --cacert orchestrator.test.lan.pem https://
orchestrator.test.lan/api/draft/health/ | python -m json.tool
```

При правильном функционировании запрашиваемого компонента ответ должен содержать:

```
{  
  "data": "ok"  
}
```

3.6 . Журналирование

Журнал работы Оркестратора расположен в файле `/var/log/termidesk/orchestrator.log`.

4. ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

Сокращение	Пояснение
ОС	Операционная система
ПО	Программное обеспечение
API	Application Programming Interface (интерфейс прикладного программирования)
FQDN	Fully Qualified Domain Name (полностью определенное имя домена)
HTTPS	Hypertext Transfer Protocol Secure (расширение протокола HTTP для поддержки шифрования)
IP	Internet Protocol (межсетевой протокол)
Termidesk	Программный комплекс «Диспетчер подключений виртуальных рабочих мест Termidesk»
URL	Uniform Resource Locator (унифицированный указатель ресурса)



© ООО «УВЕОН - ОБЛАЧНЫЕ ТЕХНОЛОГИИ»

Адрес: 119415, г. Москва, проспект Вернадского, дом 41, строение 1, офис 645а

Сайт: www.termidesk.ru

Телефон: +7 (495) 975-1-975

Общий e-mail: info@uveon.ru

Отдел продаж: sales@uveon.ru

Техническая поддержка: support@uveon.ru