

**Документация, содержащая информацию, необходимую для  
установки «Система централизованного управления  
средствами виртуализации «AccentOS CE».**

**Казань**

**2024**

## Оглавление

1 Инструкция по установке экземпляра программного обеспечения, предоставленного для проведения экспертной проверки.	3
1.1 Системные требования	3
1.2 Получение токена	3
1.3 Настройка окружения	4
1.4 Установка	7
1.5 Лицензирование	10
1.6 Настройка	10
1.7 Файлы конфигурации	11
1.8 Установка модуля Client	16
1.9 Установка модуля CloudManager	16
1.10 Установка модуля Agent	17
1.11 Установка модуля Scheduler	18
1.12 Установка модуля RSserver	18
1.13 Установка модуля RSclient	19
1.14 Установка модуля Journal	24
1.15 Установка модуля Dashboard	25
1.16 Установка модуля Clouds	26
1.17 Установка модуля OTPsender	26
1.18 Установка модуля AppLevel	27
1.19 Установка модуля TKcontrol	27

# 1 Инструкция по установке экземпляра программного обеспечения, предоставленного для проведения экспертной проверки.

## 1.1 Системные требования

Для установки и работы программы «Система централизованного управления средствами виртуализации «AccentOS SE» требуется наличие настроенных, функционирующих и доступных компонентов:

1. Python версии 3.7 и выше;
2. Система управления пакетами - pip3;
3. Операционная система (Astra Linux SE 1.7, ALT Linux);
4. Установленная и настроенная платформа OpenStack поддерживаемая выбранной версией ПО (Train).

Для корректной работы модулей также нужен установленный и настроенный Redis.

## 1.2 Получение токена

Токен необходим для получения доступа к контроллеру. Ни один из REST API запросов невозможен без токена.

Способы получения токена:

### Примечание

Все параметры в командах указаны для примера, при использовании подставляйте актуальные параметры.

1. Получение запросом к API с использованием curl.

Команда:

```
curl -i \  
-H "Content-Type: application/json" \  
-d '{  
  "auth": {  
    "identity": {  
      "methods": ["password"],  
      "password": {  
        "user": {  
          "name": "admin",  
          "domain": { "name": "default" },  
          "password": "adminpwd"  
        }  
      }  
    }  
  }  
' \  
http://controller:5000/v3/auth/tokens ; echo
```

2. Посредством консольной утилиты Openstack Client.

Команда:

```
openstack --os-auth-url http://controller:5000/v3 --os-project-domain-name default --
os-user-domain-name default --os-project-name admin --os-username admin token issue
```

Способы получения токена на проект:

Токены на проект предназначены для разрешения на работу в рамках определенного проекта. Они содержат каталог услуг, набор ролей и подробную информацию о проекте, на работу в котором выдается разрешение.

### Примечание

Все параметры в командах указаны для примера, при использовании подставляйте актуальные параметры.

1. Получение токена на проект запросом к API с использованием curl.

Команда:

```
curl -i \
-H "Content-Type: application/json" \
-d '{
  "auth": {
    "identity": {
      "methods": ["password"],
      "password": {
        "user": {
          "name": "admin",
          "domain":
            { "name": "default" }
        }
      }
    },
    "scope": {
      "project": {
        "name": "demo",
        "domain":
          { "name": "default" }
        }
      }
    }
  }' \
"http://controller:5000/v3/auth/tokens" ; echo
```

2. Получение токена на проект посредством консольной утилиты Openstack Client.

Команда:

```
openstack --os-username=admin --os-user-domain-name=default --os-password=123456 --os-
project-name=admin --os-project-domain-name=default token issue
```

## 1.3 Настройка окружения

Настроенное окружение необходимо для корректной установки и настройки модулей.

### Важно

Все команды выполняются только от суперпользователя.

Режим суперпользователя:

```
sudo -i
```

Для настройки следуйте нижеописанным пунктам:

1. Установка пакетов выполняется с помощью команды:

```
# Debian:  
apt-get install -y python3-pip python3-dev gcc
```

Для установки пакетов на Astra Linux (Smolensk) выполните следующее:

```
sudo apt-get install -y python3-pip python3-dev gettext python3-pbr python3-setuptools  
crudini
```

2. Настройка переменных окружения:

```
export PIP_INDEX_URL=http://pypi.accentos.ru/simple/  
export PIP_TRUSTED_HOST=pypi.accentos.ru
```

Для AccentOS для Astra Linux (Smolensk) подключите предоставленный Вам репозиторий с пакетами AccentOS.

3. Установка лицензии:

Поместите предоставленный Вам файл license в папку /etc/aos.

4. Установка пакетов:

```
pip3 install python-subunit  
pip3 install packaging  
pip3 install requestsexceptions  
pip3 install sqlalchemy-migrate  
pip3 install jsonpath-rw-ext
```

Для AccentOS для Astra Linux (Smolensk) установка пакетов не требуется.

5. Перезапуск служб:

```
# Debian, Astra Linux:  
systemctl restart nova-api.service  
systemctl restart apache2
```

6. Назначение пароля к базе данных для пользователя root:

```
mysqladmin -u root password 123456
```

### **Важно**

Не допускается использование не ASCII символов в логинах/паролях для базы данных.

7. После настройки окружения можно приступать к установке модулей. Рекомендуемый порядок установки модулей при инсталляции без использования модуля Autosetup:
  1. Client;
  2. CloudManager;
  3. Agent;
  4. Scheduler;

5. RSserver;
6. RScient;
7. Journal;
8. Dashboard;
9. Clouds;
10. OTPsender;
11. AppLevel;
12. TKcontrol.

Отсутствие доступа к сети интернет

В случае установки модулей на сервер без доступа к сети интернет нужно выполнить следующие шаги:

- На сервере с доступом в интернет скачать модули и их зависимости (предварительно необходимо настроить переменные окружения):
- `mkdir packages`
- `pip3 download dashboard cloud-manager scheduler rs-server dashboard-theme pip -d packages`
- Скопировать директорию packages на сервер.

### Важно

На сервере без доступа к сети интернет необходимо обеспечить наличие пакетов, перечисленных в файле: `system_packages.txt`.

### Установка и настройка Redis для Debian 10 и OpenStack Train

Redis — это быстрое хранилище данных типа «ключ-значение», которое характеризуется гибкостью, высокой производительностью и широким выбором поддерживаемых языков.

Для установки Redis и подготовки его к работе с модулями—программой «Система централизованного управления средствами виртуализации «AccentOS CE» выполните следующие шаги:

1. Обновите локальный кэш пакетов и установите Redis с помощью следующей команды:

```
apt update
apt install redis-server
```

2. Настройте Redis для работы через systemd, для этого в конфигурационный файл `/etc/redis/redis.conf` внесите следующие изменения:

```
sed -i 's/supervised no/supervised systemd/' /etc/redis/redis.conf
```

3. Для задания пароля для работы с Redis необходимо в конфигурационном файле `/etc/redis/redis.conf` раскомментировать директиву `requirepass` и добавить к ней необходимый пароль. Например:

```
sed -i 's/# requirepass foobared/requirepass password/' /etc/redis/redis.conf
```

где password - необходимый пароль.

### Примечание

При выборе пароля стоит учесть, что Redis работает очень быстро и способен перебирать до 150 тысяч паролей в секунду. В связи с этим для продуктового использования рекомендуется выбирать сложные пароли.

4. Перезапустите службы:

```
systemctl restart redis.service redis-server.service
```

## 1.4 Установка

Имеются следующие варианты самостоятельного развертывания [программы «Система централизованного управления средствами виртуализации «AccentOS»](#):

Способ установки	Описание	Размер инсталляции
<i>Ручной:</i> Установка платформы по инструкции.	Требуется установить платформу OpenStack в соответствии с официальным руководством, исходя из требований инфраструктуры площадки.  На подготовленную платформу установить модули в соответствии с официальной документацией	От одного до десяти вычислительных узлов.
<i>Автоматизированный:</i> Доставка платформы в виде образа виртуальной машины.	Требуется развернуть первый гипервизор в инфраструктуре предприятия.  На предварительно подготовленном гипервизоре запускается Управляющий Узел из предоставленного образа виртуальной машины.  Остальные гипервизоры настраиваются аналогично первому, либо загружаются в бездисковом режиме (по протоколу PXE) с запущенного Управляющего Узла.	От десяти до пятидесяти вычислительных узлов.
<i>Автоматический:</i> Установка платформы с помощью служебного модуля FirstBoot.	Первоначальная настройка FirstBoot под инфраструктуру конкретной площадки.  Установка и настройка платформы в автоматическом режиме.  Дальнейшая поддержка, масштабирование и обновление платформы в автоматическом режиме.	От пятидесяти вычислительных узлов и выше.

## Ручная установка

Каждый модуль может быть установлен самостоятельно с помощью команды:

```
pip3 install module_name
```

### Пример ручной установки модуля

#### **Важно**

Перед началом установки необходимо произвести настройку окружения. Все команды выполняются только от суперпользователя.

Режим суперпользователя:

```
sudo -i
```

#### **Важно**

Установка производится на управляющий узел. Для работы модуля необходимо установить службу OpenStack Nova API.

1. Перед началом установки сохраните список установленных ранее пакетов, это позволит Вам безболезненно восстановить систему в случае ее повреждения. Для этого выполните следующие команды:

```
mkdir -p /tmp/rollback/cloud_manager  
pip3 freeze > /tmp/rollback/cloud_manager/
```

После чего в каталоге /tmp/rollback/cloud\_manager будет находиться файл pip\_before.txt с перечнем установленных приложений.

2. Также сохраните версии миграции:

```
openstack aos db list -n cloud_manager > /tmp/rollback/cloud_manager/migrations.txt
```

Где:

- /tmp/rollback/cloud\_manager/ - директория файла;
- migrations.txt - наименование файла с версиями миграций.

3. Установите пакет CloudManager:

- из репозитория Python-пакетов:

```
pip3 install cloud-manager
```

4. Сохраните список установленных пакетов после установки для возможности отката изменений:

```
pip3 freeze > /tmp/rollback/cloud_manager/pip_after.txt
```

5. Добавьте пользователя:

```
useradd -m aos  
passwd aos
```

#### **Примечание**

Для установки CloudManager на Astra Linux SE 1.6 (Smolensk) выполните следующее:

1. Подключите предоставленный Вам репозиторий с пакетами программы «Система централизованного управления средствами виртуализации «AccentOS CE».
2. Установите пакет командой:

```
sudo apt install -y cloud-manager
```

Установка на двух и более контроллерах

При установке CloudManager на двух и более контроллерах необходимо:

1. Реплицировать базу данных на каждый из контроллеров;
2. Устанавливать модуль с **одинаковыми параметрами** на каждый из контроллеров;
3. Процессы мониторинга и управления и синхронизации вычислительных узлов должны быть активны только на одном контроллере. У этого же контроллера должен быть настроен доступ к серверу брокера сообщений;
4. Инициализацию вычислительных узлов и устройств управления питанием необходимо проводить только на одном из контроллеров.

### Примечание

Удаление и диагностика модуля на каждом контроллере происходит таким же образом, как и в случае с одним контроллером.

Остальные модули программы «Система централизованного управления средствами виртуализации «AccentOS CE» устанавливаются аналогично, с описанием установки можно ознакомиться в официальной документации:

<http://docs.accentos.ru>

### Автоматическая установка

Автоматическую установку и настройку модулей программы «Система централизованного управления средствами виртуализации «AccentOS CE» обеспечивает модуль **Autosetup**.

Запуск автоустановки модулей производится утилитой autosetup.

Все необходимые параметры для установки модулей могут быть указаны в качестве аргументов этой утилиты, а также в переменных окружения. Установка продуктов производится утилитой pip, которая устанавливает пакеты с заданного хоста в зависимости от операционной системы. Подробнее ознакомиться с процессом установки модулей можно в разделе «Функционал модуля Autosetup».

Помимо самих продуктов, в том числе устанавливаются системные пакеты, необходимые для корректной работы продуктов.

Перечень устанавливаемых продуктов:

- Client (в качестве зависимости для остальных модулей);
- Dashboard;
- CloudManager;
- Scheduler;
- RSserver;

- Journal
- Clouds;
- OTPsender;
- AppLevel;
- TKcontrol.

При установке любого из модулей необходимо, чтобы были выполнены системные требования для устанавливаемого модуля.

## 1.5 Лицензирование

Лицензируемые модули:

- **CloudManager;**
- **RSserver.**

Лицензии модулей - это инструмент, определяющий легальное использование и распространение программы «Система централизованного управления средствами виртуализации «AccentOS CE». По истечению срока действия лицензии функционал модуля ограничивается, но информация и данные, созданные во время срока действия лицензии сохраняются. Для восстановления функциональности достаточно установить актуальную лицензию модуля.

**Подробнее:** [http://docs.accentos.ru/ru/modules/general-information/general\\_information/licensing.html](http://docs.accentos.ru/ru/modules/general-information/general_information/licensing.html)

## 1.6 Настройка

Для настройки модулей реализована специальная утилита:

**openstack aos configure** - консольная утилита автоматической настройки модулей программы «Система централизованного управления средствами виртуализации «AccentOS CE» после установки.

При запуске утилиты существующие конфигурационные файлы сервисов systemd не перезаписываются.

Аргументы утилиты:

Параметр	Описание
-h, --help	Вывод справки. Необязательный параметр.
-l, --list	Вывод перечня всех модулей, поддерживающих настройку. Необязательный параметр.
-n, --names MODULE_NAMES	Запуск утилиты для конкретного модуля. Наименование задается параметром MODULE_NAMES. При нескольких наименованиях параметр разделяется пробелом. Необязательный параметр.

Параметр	Описание
-v, --verbose	Вывод в таблицу детализированного сообщения об ошибке в случае её возникновения. Необязательный параметр.
-r, --rewrite	Возможность перезаписи файлов служб. Необязательный параметр.

Команда:

```
openstack aos configure --list
```

Запуск настройки для одного модуля

Пример запуска настройки для Dashboard:

```
openstack aos configure --names dashboard
```

## 1.7 Файлы конфигурации

Файлы конфигурации позволяют настраивать работу модулей.

Конфигурационные файлы описывают авторизационные данные для подключения к службам (OpenStack, DB, RabbitMQ и др.), а также параметры работы модулей.

Файлы имеют формат .conf.

Конфигурационные файлы продуктов AccentOS® делятся на два типа:

- общий файл конфигурации;
- модульный файл конфигурации.

При установке модулей создаются образцы файлов конфигурации с расширением conf.example. Файлы содержат минимальный набор секций, необходимых для работы модулей. После внесения изменений в файл conf.example, в целях сохранения образца настроек следует сохранить их под другим именем.

Ключи в файлах конфигурации являются регистронезависимыми. Также не зависят от регистра значения булевых параметров (TRUE/FALSE) и уровни логирования, остальные параметры, включая названия секций, чувствительны к регистру.

При установке любого из модулей устанавливаются оба вида конфигурационных файлов: модульный и общий, с расширением .conf.example в директорию /etc/aos/.

### Общий файл конфигурации

Данный вид файла конфигурации используется всеми модулями, что позволяет настроить сразу все модули в одном файле. Общий файл aos.conf находится в директории /etc/aos/. В файле aos.conf.example перечислены секции и их опции. Если в описании параметров не указано иное, значения параметров чувствительны к регистру.

Общий файл конфигурации появляется после установки одного из модулей: Dashboard, Journal, CloudManager, RSserver, Scheduler.

Параметры общего конфиг файла описаны в документации:

Данный вид файла конфигурации используется всеми модулями, что позволяет настроить сразу все модули в одном файле. Общий файл `aos.conf` находится в директории `/etc/aos/`. В файле `aos.conf.example` перечислены секции и их опции. Если в описании параметров не указано иное, значения параметров чувствительны к регистру.

Общий файл конфигурации появляется после установки одного из модулей: Dashboard, Journal, CloudManager, RSserver, Scheduler.

Изменение параметров файла конфигурации описано в соответствующем разделе документации.

Содержание файла конфигурации

Файл конфигурации состоит из следующих секций:

Секция	Описание
DEFAULT	Секция по умолчанию для параметров, не включенных в другие секции.
keystone	Настройки аутентификации в службе Keystone.
database	Настройки подключения к базе данных.
messaging	Настройки службы обмена сообщениями.
os_api	Настройки точек доступа к сервисам OpenStack.
api_auth	Настройки аутентификации служб AccentOS.

Секция DEFAULT

Состоит из следующих опций:

Опция	Описание	По умолчанию
log_level	Указывает установленный уровень логирования. Подробное описание уровней доступно в разделе «Уровни логирования».	info
log_rotation_type	Указывает установленный тип ротации логов. Подробное описание ротации доступно в разделе «Ротация логов».	size
max_logfile_size_mb	При установленном типе ротации <i>size</i> , указывает максимальный размер файлов логирования в МВ.	512

Опция	Описание	По умолчанию
max_logfile_count	Максимальное количество файлов логирования после ротации.	10
log_rotate_interval_type	При установленном типе ротации <i>interval</i> , указывает тип интервала ротации. Допустимые значения: Seconds, Minutes, Hours, Days, Weekday, Midnight.	Day
log_rotate_interval	При установленном типе ротации <i>interval</i> , указывает интервал ротации.	1

Секция keystone

Состоит из следующих опций:

Опция	Описание	По умолчанию
auth_url	URL авторизации в службе Keystone.	http://localhost:5000
username	Имя пользователя.	admin
password	Пароль пользователя.	
project_name	Название проекта OpenStack.	admin
user_domain_name	Название домена пользователя.	default
project_domain_name	Название домена проекта.	default

Секция database

Состоит из следующих опций:

Опция	Описание	По умолчанию	Возможные значения
max_pool_size	Максимальное количество открытых подключений в пуле.	5	Любые неотрицательные числа, при значении 0 - количество подключений не ограничено.
max_overflow	Число соединений, на которое можно	50	Любые неотрицательные числа, при значении -1 - количество соединений не ограничено.

Опция	Описание	По умолчанию	Возможные значения
	превысить max_pool_size.		
pool_recycle	Таймаут превышения соединения (в секундах) - интервал времени, по истечении которого открытое соединение удаляется из пула.	3600	Любые неотрицательные числа, при значении -1 - соединения не удаляются.
pool_timeout	Максимальный интервал времени ожидания получения соединения из пула (в секундах).	30	Любые неотрицательные числа.
retry_interval	Время ожидания между попытками соединения (в секундах).	2	Любые неотрицательные числа.
max_retries	Максимальное количество попыток подключения.	5	Любые неотрицательные числа, при значении 0 - не осуществляются попытки переключиться, при -1 - попытки подключения осуществляются бесконечно

Секция messaging

Состоит из следующих опций:

Опция	Описание	По умолчанию
url	URL подключения к службе обмена сообщениями.	amqp://aos:password@localhost:5672/aos

Секция redis

Состоит из следующих опций:

Опция	Описание	По умолчанию
url	URL подключения к NoSQL базе данных Redis.	redis://:password@localhost:6379/0

Секция os\_api

Состоит из следующих опций:

Опция	Описание	По умолчанию
nova_api_version	Версия API службы Nova.	2.60
cinder_api_version	Версия API службы Cinder.	3.50
gnocchi_api_version	Версия API службы Gnocchi.	1
glance_api_version	Версия API службы Glance.	2
neutron_api_version	Версия API службы Neutron.	2
compute_service_type	Тип службы Nova.	compute
image_service_type	Тип службы Glance.	image
network_service_type	Тип службы Neutron.	network
volume_service_type	Тип службы Cinder.	volumev3
gnocchi_service_type	Тип службы Gnocchi.	metric

Секция api\_auth

Опция	Описание	По умолчанию
auth_url	URL авторизации в службе Keystone.	
username	Имя пользователя.	aos
password	Пароль пользователя.	
project_name	Название проекта OpenStack.	service

Опция	Описание	По умолчанию
user_domain_name	Название домена пользователя.	default
project_domain_name	Название домена проекта.	default

Изменение параметров файла конфигурации

### Важно

Сначала необходимо произвести настройку окружения. Все команды выполняются только от суперпользователя.

Скопируйте образец конфигурационного файла:

```
cp /etc/aos/aos.conf.example /etc/aos/aos.conf
```

После выполнения команды вы можете редактировать файл, его можно найти в директории /etc/aos/.

Для того, чтобы изменения в файле конфигурации вступили в силу, необходимо перезапустить веб-сервер и *systemd* службы модулей:

```
# Debian:
systemctl restart apache2
systemctl restart aos-*
```

Модульный файл конфигурации

Данный вид файлов используется конкретным модулем. Файл конфигурации находится в директории /etc/aos/. Название файла соответствует названию модуля.

### Примечание

В случае, если в конфигурационном файле модуля не указаны путь и имена файлов для сбора логов, то логирование действий производится средствами системы, например, *journalctl*.

## 1.8 Установка модуля Client

### Примечание

Модуль также устанавливается в процессе установки других модулей.

1. Установите пакет Client:
  - из репозитория Python-пакетов:

```
pip3 install client
```

Для установки Client на Astra Linux (Smolensk) выполните следующее:

1. Подключите предоставленный Вам репозиторий с пакетами AccentOS.
2. Установите пакет командой:

```
sudo apt install -y aos-client
```

## 1.9 Установка модуля CloudManager

### Примечание

Установка производится на управляющий узел. Для работы модуля необходимо установить службу OpenStack Nova API.

1. Перед началом установки сохраните список установленных ранее пакетов, это позволит Вам безболезненно восстановить систему в случае ее повреждения. Для этого выполните следующие команды:

```
mkdir -p /tmp/rollback/cloud_manager  
pip3 freeze > /tmp/rollback/cloud_manager/
```

После чего в каталоге /tmp/rollback/cloud\_manager будет находиться файл pip\_before.txt с перечнем установленных приложений.

2. Также сохраните версии миграции:

```
openstack aos db list -n cloud_manager > /tmp/rollback/cloud_manager/migrations.txt
```

Где:

- /tmp/rollback/cloud\_manager/ - директория файла;
- migrations.txt - наименование файла с версиями миграций.

3. Установите пакет CloudManager:

- из репозитория Python-пакетов:

```
pip3 install cloud-manager
```

4. Сохраните список установленных пакетов после установки для возможности отката изменений:

```
pip3 freeze > /tmp/rollback/cloud_manager/pip_after.txt
```

5. Добавьте пользователя:

```
useradd -m aos  
passwd aos
```

### Примечание

Для установки CloudManager на Astra Linux (Smolensk) выполните следующее:

1. Подключите предоставленный Вам репозиторий с пакетами AccentOS.
2. Установите пакет командой:

```
sudo apt install -y cloud-manager
```

Установка на двух и более контроллерах

При установке CloudManager на двух и более контроллерах необходимо:

1. Реплицировать базу данных на каждый из контроллеров;
2. Устанавливать модуль с **одинаковыми параметрами** на каждый из контроллеров;

3. Процессы мониторинга и управления и синхронизации вычислительных узлов должны быть активны только на одном контроллере. У этого же контроллера должен быть настроен доступ к серверу брокера сообщений;
4. Инициализацию вычислительных узлов и устройств управления питанием необходимо проводить только на одном из контроллеров.

### Примечание

Удаление и диагностика модуля на каждом контроллере происходит таким же образом, как и в случае с одним контроллером.

## 1.10 Установка модуля Agent

### Примечание

Выполняется на всех вычислительных узлах, коммутаторах Mellanox.

1. Установите пакет Agent:
  - из репозитория Python-пакетов:

```
pip3 install agent
```

Для установки Agent на Astra Linux (Smolensk) выполните следующее:

1. Подключите предоставленный Вам репозиторий с пакетами AccentOS.
2. Установите пакет командой:

```
sudo apt install -y aos-agent
```

## 1.11 Установка модуля Scheduler

### Примечание

Установка производится на управляющий узел.

1. Перед началом установки сохраните список установленных ранее пакетов, это позволит Вам безболезненно восстановить систему в случае ее повреждения. Для этого выполните следующие команды:

```
mkdir -p /tmp/rollback/scheduler  
pip3 freeze > /tmp/rollback/scheduler/pip_before.txt
```

После чего в каталоге /tmp/rollback/scheduler будет находиться файл pip\_before.txt с перечнем установленных приложений.

2. Также сохраните версии миграции:

```
django-admin showmigrations --settings=scheduler.settings.django_settings >  
/tmp/rollback/scheduler/migrations.txt
```

Где:

- /tmp/rollback/scheduler/ - директория файла;
- migrations.txt - наименование файла с версиями миграций.

3. Установите пакет Scheduler:

- из репозитория Python-пакетов:

```
pip3 install scheduler
```

4. Сохраните список установленных пакетов после установки для возможности отката изменений:

```
pip3 freeze > /tmp/rollback/scheduler/pip_after.txt
```

Для установки Scheduler на Astra Linux (Smolensk) выполните следующее:

1. Подключите предоставленный Вам репозиторий с пакетами AccentOS.
2. Установите пакет командой:

```
sudo apt install -y aos-scheduler
```

## 1.12 Установка модуля RSserver

1. Перед началом установки сохраните список установленных ранее пакетов, это позволит Вам безболезненно восстановить систему в случае ее повреждения. Для этого выполните следующие команды:

```
mkdir -p /tmp/rollback/rs_server  
pip3 freeze > /tmp/rollback/rs_server/pip_before.txt
```

После чего в каталоге /tmp/rollback/rs\_server будет находиться файл pip\_before.txt с перечнем установленных приложений.

2. Также сохраните версии миграции:

```
openstack aos db list -n rs_sever > /tmp/rollback/rs_server/migrations.txt
```

Где:

- /tmp/rollback/rs\_server/ - директория файла;
- migrations.txt - наименование файла с версиями миграций.

3. Установите пакет RSserver:

- из репозитория Python-пакетов:

```
pip3 install rs-server
```

4. Сохраните список установленных пакетов после установки для возможности отката изменений:

```
pip3 freeze > /tmp/rollback/rs_server/pip_after.txt
```

5. Добавьте пользователя aos:

```
useradd -m aos
```

```
passwd password
```

Для установки RSserver на Astra Linux (Smolensk) выполните следующее:

1. Подключите предоставленный Вам репозиторий с пакетами AccentOS.
2. Установите пакет командой:

```
sudo apt install -y aos-rs-server
```

## 1.13 Установка модуля RSclient

Для Linux

### Внимание

Имеется поддержка аутентификации пользователя с использованием смарт-карты. Подробнее [здесь](#).

1. Установите пакет RSclient:
  - из репозитория Python-пакетов:

```
pip3 install rs-client
```

Для Astra Linux 1.7

Для установки RSclient на Astra Linux выполните следующее:

1. Подключите предоставленный Вам репозиторий с пакетами AccentOS.
2. Установите пакет командой:

```
sudo apt install -y aos-rs-client
```

### Внимание

Функционал получения виртуальных машин по тикетам Kerberos на Для Astra Linux 1.7 доступен только при установке пакета через pip.

Для корректной установки RSclient через pip на Astra Linux 1.7 выполните следующее:

1. Подключите представленные Вам репозитории с пакетами AccentOs для Astra Linux 1.7.
2. Установите пакеты, необходимые для Linux.
3. Дополнительно установите необходимые для работы пакеты командой:

```
apt install python3-pyside2.qt3dcore python3-pyside2.qt3dinput python3-pyside2.qt3dlogic python3-pyside2.qt3drender python3-pyside2.qtconcurrent \
python3-pyside2.qtc core python3-pyside2.qtgui python3-pyside2.qthelp python3-pyside2.qtllocation python3-pyside2.qtmultimedia python3-pyside2.qtmultimediawidgets \
python3-pyside2.qtnetwork python3-pyside2.qtopengl python3-pyside2.qtpositioning python3-pyside2.qtprintsupport python3-pyside2.qtqml python3-pyside2.qtquick \
python3-pyside2.qtquickwidgets python3-pyside2.qtscript python3-pyside2.qtscripttools python3-pyside2.qtsensors python3-pyside2.qtsql python3-pyside2.qtsvg \
python3-pyside2.qtttest python3-pyside2.qtttexttospeech python3-pyside2.qttuitools python3-pyside2.qtwebchannel python3-pyside2.qtwebenginecore \
```

```
python3-pyside2.qtwebenginewidgets python3-pyside2.qtwebsockets python3-
pyside2.qtwidgets python3-pyside2.qtx11extras python3-pyside2.qtxml python3-
pyside2.qtxmlpatterns
```

4. Создайте симлинк для недостающей библиотеки:

```
sudo ln -s /usr/lib/x86_64-linux-gnu/libxcb-util.so.0 /usr/lib/x86_64-linux-gnu/libxcb-
util.so.1
```

5. Установите пакет командой:

```
pip3 install -y aos-rs-client
```

Для Windows

### Внимание

Поддерживается только 64-разрядная версия Windows.

- Установка RScient при помощи установщика

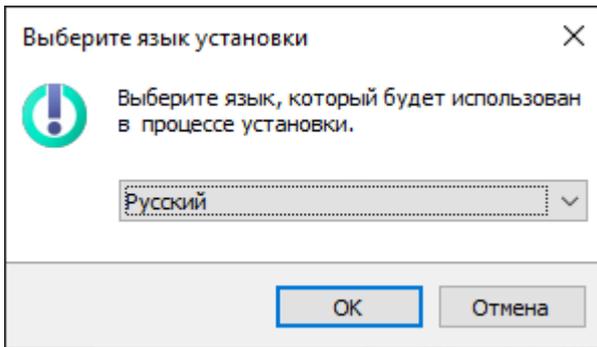
Установка RScient при помощи установщика

1. Перейдите по ссылке в репозиторий с дистрибутивами клиента (<https://files.accentos.ru/index.php/s/cmxeC6fXWMLWXJG>).
2. Выберите совместимую с операционной системой версию RScient:

Имя	Размер	Изменён
RScient.0.5.0x64.Setup.exe	41,8 MB	день назад

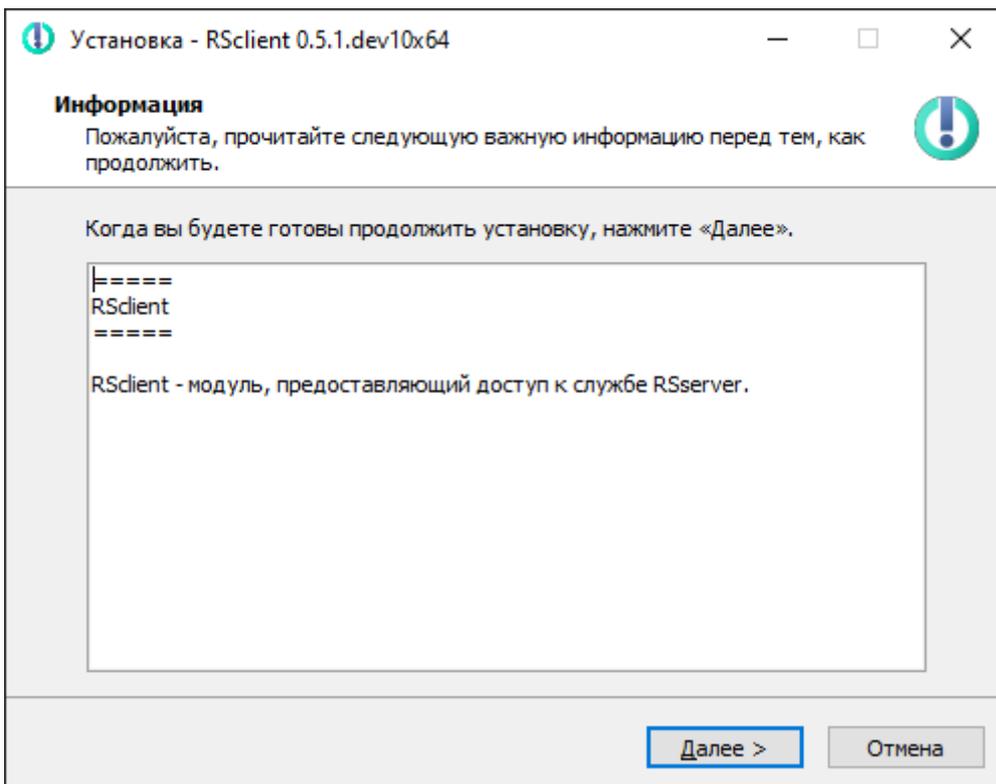
### Перечень клиентов

3. Скачайте и запустите установочную программу RScient.1.30.0x64.Setup.exe.
4. Выберите язык установки:



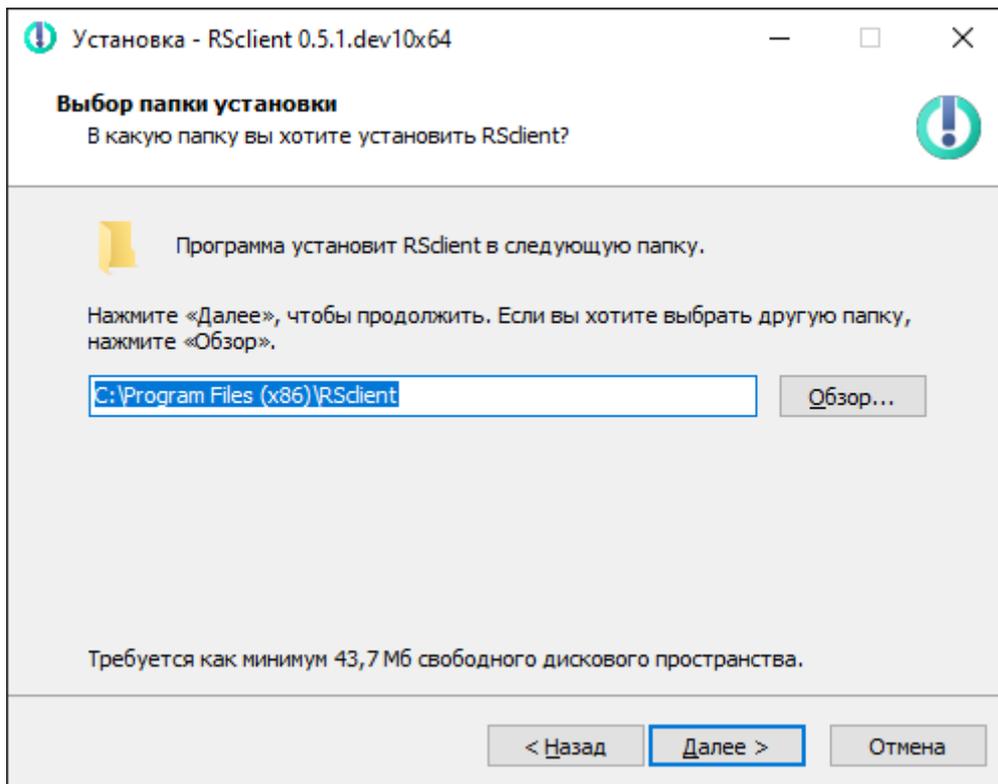
Окно выбора языка

5. Ознакомьтесь с программным продуктом:



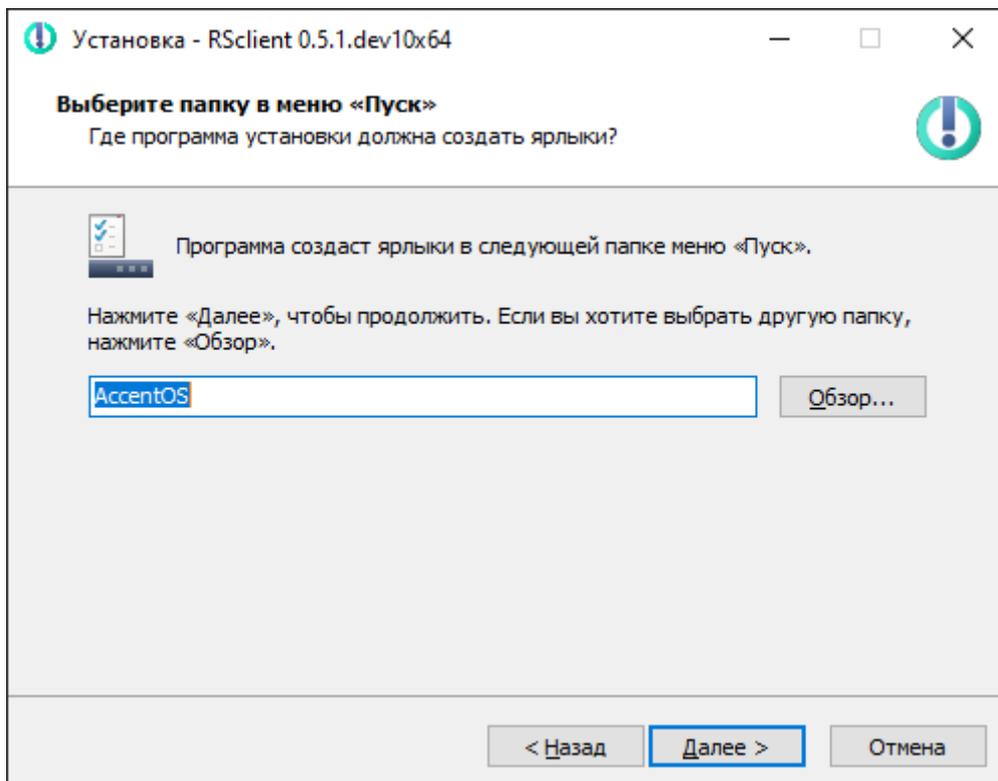
Окно с информацией о модуле

6. Выберите директорию для установки:



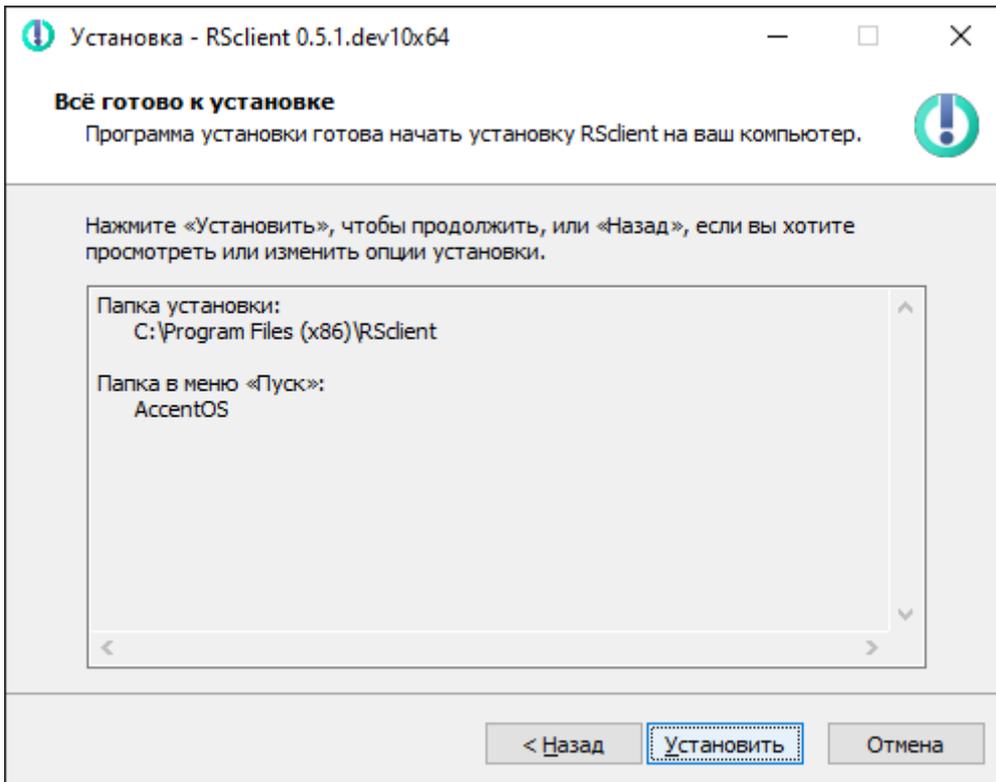
*Окно выбора директории для установки приложения*

7. Выберите папку в меню «Пуск»:



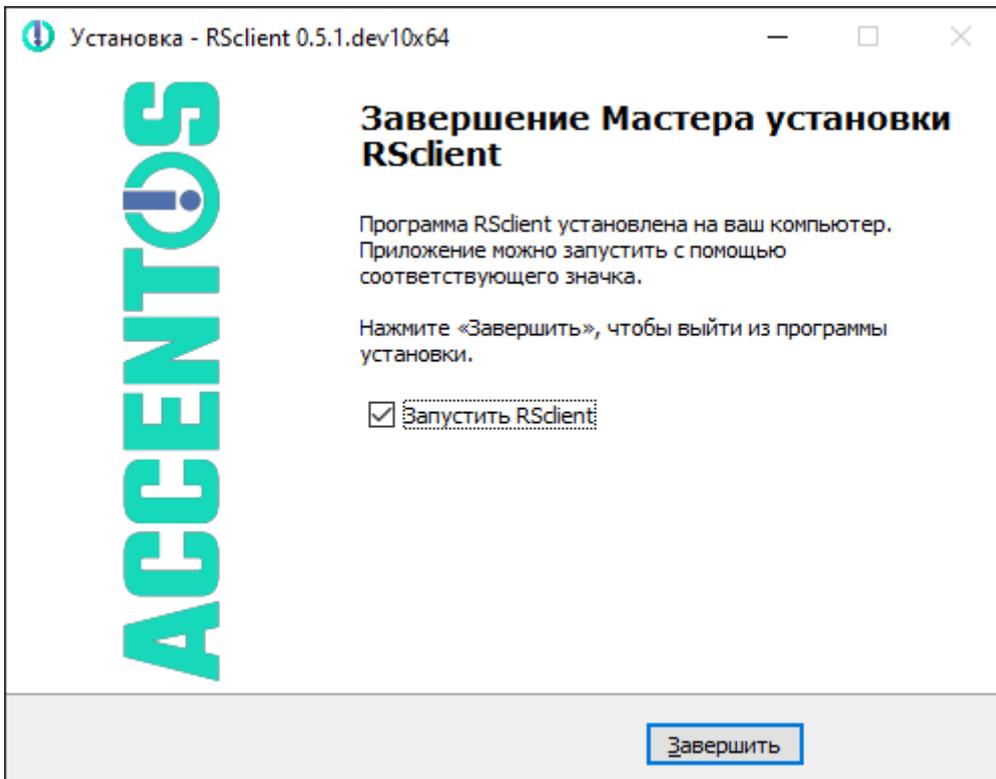
*Окно выбора директории для ярлыка приложения*

8. Подтвердите параметры установки:



Окно запуска установки

После успешной установки в окне завершения по умолчанию установлен флаг запуска программы:



Окно завершения установки

Установите необходимое значение и завершите установку кнопкой «Завершить».

## 1.14 Установка модуля Journal

1. Перед началом установки сохраните список установленных ранее пакетов, это позволит Вам безболезненно восстановить систему в случае ее повреждения. Для этого выполните следующие команды:

```
mkdir -p /tmp/rollback/journal  
pip3 freeze > /tmp/rollback/journal/pip_before.txt
```

После чего в каталоге /tmp/rollback/journal будет находиться файл pip\_before.txt с перечнем установленных приложений.

2. Также сохраните версии миграции:

```
openstack aos db list -n journal > /tmp/rollback/journal/migrations.txt
```

3. Где:

- /tmp/rollback/journal/ - директория файла;
- migrations.txt - наименование файла с версиями миграций.

4. Установите пакет Journal:

- из репозитория Python-пакетов:

```
pip3 install journal
```

5. Сохраните список установленных пакетов после установки для возможности отката изменений:

```
pip3 freeze > /tmp/rollback/journal/pip_after.txt
```

Для установки Journal на Astra Linux (Smolensk) выполните следующее:

1. Подключите предоставленный Вам репозиторий с пакетами AccentOS.
2. Установите пакет командой:

```
sudo apt install -y aos-journal
```

## 1.15 Установка модуля Dashboard

### Примечание

Установка производится на управляющий узел.

1. Перед началом установки сохраните список установленных ранее пакетов, это позволит Вам безболезненно восстановить систему в случае ее повреждения. Для этого выполните следующие команды:

```
mkdir -p /tmp/rollback/dashboard  
pip3 freeze > /tmp/rollback/dashboard/pip_before.txt
```

После чего в каталоге /tmp/rollback/dashboard будет находиться файл pip\_before.txt с перечнем установленных приложений.

2. Также сохраните версии миграции:

```
python3 /usr/share/openstack-dashboard/manage.py showmigrations >
/tmp/rollback/dashboard/migrations.txt
```

Где:

- /tmp/rollback/dashboard/ - директория файла;
- migrations.txt - наименование файла с версиями миграций.

3. Установите пакет [OpenStack Horizon](#):

```
apt-get install openstack-dashboard
```

Для Astra Linux (Smolensk) установите пакет командой:

```
sudo apt install -y openstack-dashboard
```

4. Установите пакет Dashboard:

- из репозитория Python-пакетов:

```
pip3 install dashboard
```

5. Сохраните список установленных пакетов после установки для возможности отката изменений:

```
pip3 freeze > /tmp/rollback/dashboard/pip_after.txt
```

### Примечание

Для установки Dashboard и Dashboard Theme на Astra Linux (Smolensk) выполните следующее:

1. Подключите предоставленный Вам репозиторий с пакетами AccentOS.
2. Установите пакет командой:

```
sudo apt install -y aos-dashboard
sudo apt install -y aos-dashboard-theme
```

## 1.16 Установка модуля Clouds

### Примечание

Установка производится на управляющий узел.

1. Перед началом установки сохраните список установленных ранее пакетов, это позволит Вам безболезненно восстановить систему в случае ее повреждения. Для этого выполните следующие команды:

```
mkdir -p /tmp/rollback/clouds
pip3 freeze > /tmp/rollback/clouds/pip_before.txt
```

После чего в каталоге /tmp/rollback/clouds будет находиться файл pip\_before.txt с перечнем установленных приложений.

2. Также сохраните версии миграции:

```
openstack aos db list -n clouds > /tmp/rollback/clouds/migrations.txt
```

Где:

/tmp/rollback/clouds/ - директория файла;

migrations.txt - наименование файла с версиями миграций.

3. Установите пакет Clouds:

```
pip3 install clouds
```

4. Сохраните список установленных пакетов после установки для возможности отката изменений:

```
pip3 freeze > /tmp/rollback/clouds/pip_after.txt
```

## 1.17 Установка модуля OTPsender

1. Перед началом установки сохраните список установленных ранее пакетов, это позволит Вам безболезненно восстановить систему в случае ее повреждения. Для этого выполните следующие команды:

```
mkdir -p /tmp/rollback/otp-sender  
pip3 freeze > /tmp/rollback/otp-sender/pip_before.txt
```

После чего в каталоге /tmp/rollback/otp\_sender будет находиться файл pip\_before.txt с перечнем установленных приложений.

2. Установите пакет OTPsender:

- из репозитория Python-пакетов:

```
pip3 install otp-sender
```

3. Сохраните список установленных пакетов после установки для возможности отката изменений:

```
pip3 freeze > /tmp/rollback/otp-sender/pip_after.txt
```

### Примечание

Для установки OTPsender на Astra Linux (Smolensk) выполните следующее:

1. Подключите предоставленный Вам репозиторий с пакетами AccentOS.
2. Установите пакет командой:

```
sudo apt install -y aos-otp-sender
```

## 1.18 Установка модуля AppLevel

1. Перед началом установки сохраните список установленных ранее пакетов, это позволит Вам безболезненно восстановить систему в случае ее повреждения. Для этого выполните следующие команды:

```
mkdir -p /tmp/rollback/applevel
```

```
pip3 freeze > /tmp/rollback/applevel/pip_before.txt
```

После чего в каталоге /tmp/rollback/applevel будет находиться файл pip\_before.txt с перечнем установленных приложений.

1. Установите пакет AppLevel:
  - из репозитория Python-пакетов:

```
pip3 install applevel
```

2. Сохраните список установленных пакетов после установки для возможности отката изменений:

```
pip3 freeze > /tmp/rollback/applevel/pip_after.txt
```

### Примечание

Для установки AppLevel на Astra Linux (Smolensk) выполните следующее:

1. Подключите предоставленный Вам репозиторий с пакетами AccentOS.
2. Установите пакет командой:

```
sudo apt install -y aos-applevel
```

## 1.19 Установка модуля TKcontrol

### Mongo DB

Импортируйте публичный gpg ключ mongodb:

```
wget -qO - https://www.mongodb.org/static/pgp/server-4.4.asc | sudo apt-key add -
```

Добавьте mongodb в список репозитория:

```
echo "deb http://repo.mongodb.org/apt/debian buster/mongodb-org/4.4 main" | sudo tee /etc/apt/sources.list.d/mongodb-org-4.4.list
```

Обновите список пакетов и установите mongodb:

```
sudo apt update -y  
sudo apt install -y mongodb-org
```

Добавьте mongodb в автозапуск и включите службу:

```
sudo systemctl enable mongod  
sudo systemctl start mongod
```

### Rabbit MQ

Установите Rabbit MQ Server:

```
sudo apt install -y rabbitmq-server
```

Добавьте rabbitmq в автозапуск и включите службу:

```
sudo systemctl enable rabbitmq-server
sudo systemctl start rabbitmq-server
```

## SaltStack

Установите Salt Master и Salt Api:

```
sudo apt install -y salt-master salt-api
```

Добавьте salt-master и salt-api в автозапуск и включите службы:

```
sudo systemctl enable salt-master salt-api
sudo systemctl start salt-master salt-api
```

## Websockify

Установите websockify:

```
sudo apt install -y websockify
```

## VNC Repeater

Склонируйте репозиторий uvncrepeater-ac:

```
git clone https://github.com/tenchman/uvncrepeater-ac
```

Запустите сборку приложения:

```
sudo make -C uvncrepeater-ac install
```

Добавьте пользователя для uvncrepeater:

```
sudo useradd -s /sbin/nologin uvncrep
```

## Nginx JWT Module

Установка из исходных файлов

Установите версии модулей в переменные окружения:

```
export NGINX_VERSION=1.14.2 && \  
export JANSSON_VERSION=2.10 && \  
export LIBJWT_VERSION=1.9.0
```

Установите необходимые библиотеки для сборки исходных файлов:

```
sudo apt install -y build-essential
sudo apt install -y wget unzip git cmake check autoconf libtool openssl libssl-dev
libpcre3 libpcre3-dev zlib1g zlib1g-dev libxml2 libxml2-dev libxslt-dev libgd-dev
google-perftools libgoogle-perftools-dev libperl-dev
```

Установите nginx:

```
sudo apt install -y nginx=$NGINX_VERSION-2+deb10u4
```

Остановите и отключите nginx:

```
sudo systemctl stop nginx
sudo systemctl disable nginx
```

Установите библиотеку для работы с JSON:

```
wget https://github.com/akheron/jansson/archive/v$JANSSON_VERSION.zip && \
unzip v$JANSSON_VERSION.zip && \
rm v$JANSSON_VERSION.zip && \
cd jansson-$JANSSON_VERSION && \
cmake . -DJANSSON_BUILD_SHARED_LIBS=1 -DJANSSON_BUILD_DOCS=OFF && \
make && \
make check && \
sudo make install && \
cd ..
```

Установите библиотеку для работы с JWT токенами:

```
wget https://github.com/benmcollins/libjwt/archive/v$LIBJWT_VERSION.zip && \
unzip v$LIBJWT_VERSION.zip && \
rm v$LIBJWT_VERSION.zip && \
cd libjwt-$LIBJWT_VERSION && \
autoreconf -i && \
./configure && \
make all && \
sudo make install && \
cd ..
```

Загрузите nginx модуль для работы с JWT токенами:

```
git clone https://github.com/TeslaGov/nginx-http-auth-jwt-module
```

Соберите nginx с необходимыми модулями:

```
wget http://nginx.org/download/nginx-$NGINX_VERSION.tar.gz && \
tar -xzf nginx-$NGINX_VERSION.tar.gz && \
rm nginx-$NGINX_VERSION.tar.gz && \
cd nginx-$NGINX_VERSION
```

# Конфигурируем nginx модули

```
./configure --add-dynamic-module=../ngx-http-auth-jwt-module \
--prefix=/usr/share/nginx \
--sbin-path=/usr/sbin/nginx \
--modules-path=/usr/lib/nginx/modules \
--conf-path=/etc/nginx/nginx.conf \
--error-log-path=/var/log/nginx/error.log \
--http-log-path=/var/log/nginx/access.log \
--pid-path=/run/nginx.pid \
--lock-path=/var/lock/nginx.lock \
--user=www-data \
--group=www-data \
--build=Ubuntu \
--http-client-body-temp-path=/var/lib/nginx/body \
--http-fastcgi-temp-path=/var/lib/nginx/fastcgi \
--http-proxy-temp-path=/var/lib/nginx/proxy \
--http-scgi-temp-path=/var/lib/nginx/scgi \
```

```

--http-uwsgi-temp-path=/var/lib/nginx/uwsgi \
--with-pcre \
--with-pcre-jit \
--with-compat \
--with-file-aio \
--with-threads \
--with-http_addition_module \
--with-http_auth_request_module \
--with-http_dav_module \
--with-http_flv_module \
--with-http_gunzip_module \
--with-http_gzip_static_module \
--with-http_image_filter_module=dynamic \
--with-http_mp4_module \
--with-http_random_index_module \
--with-http_realip_module \
--with-http_slice_module \
--with-http_ssl_module \
--with-http_sub_module \
--with-http_stub_status_module \
--with-http_xslt_module=dynamic \
--with-http_v2_module \
--with-http_secure_link_module \
--with-mail=dynamic \
--with-mail_ssl_module \
--with-stream=dynamic \
--with-stream_realip_module \
--with-stream_ssl_module \
--with-stream_ssl_preread_module \
--with-debug \
--with-cc-opt='-g -O2 -fdebug-prefix-map=/build/nginx-mcUg8N/nginx-
$NGINX_VERSION=. -fstack-protector-strong -Wformat -Werror=format-security -fPIC -
Wdate-time -D_FORTIFY_SOURCE=2'\
--with-ld-opt='-Wl,-Bsymbolic-functions -Wl,-z,relro -Wl,-z,now -fPIC'

```

# Билдим nginx

```

make && \
sudo make install && \
cd ..

```

# Выполняем unlink конфигурации старых модулей

```

sudo unlink /etc/nginx/modules-enabled/50-mod-http-auth-pam.conf && \
sudo unlink /etc/nginx/modules-enabled/50-mod-http-dav-ext.conf && \
sudo unlink /etc/nginx/modules-enabled/50-mod-http-echo.conf && \
sudo unlink /etc/nginx/modules-enabled/50-mod-http-geoip.conf && \
sudo unlink /etc/nginx/modules-enabled/50-mod-http-sub-filter.conf && \
sudo unlink /etc/nginx/modules-enabled/50-mod-http-upstream-fair.conf

```

# Добавляем конфигурацию для jwt модуля

```

echo "load_module modules/ngx_http_auth_jwt_module.so;" | sudo tee /etc/nginx/modules-
enabled/http-auth-jwt.conf

```

Обновите динамические библиотеки:

```

sudo /sbin/ldconfig

```

**TKControl**

Установите пакеты tkcontrol модулей:

```
sudo pip3 install --prefix /usr/local \  
--index-url http://pypi.accentos.ru/ \  
--trusted-host pypi.accentos.ru \  
tkcontrol-auth \  
tkcontrol-backend \  
tkcontrol-dbadapter \  
tkcontrol-modules \  
tkcontrol-configure \  
tkcontrol-services
```

Добавьте ссылку на gunicorn:

```
sudo ln -s /usr/local/bin/gunicorn /usr/bin/gunicorn
```

#### **ПРИМЕЧАНИЕ:**

После установки всех модулей обязательно выполнить настройку модулей с помощью утилиты openstack aos configure:

[http://docs.accentos.ru/1.17.0/ru/modules/general-information/general\\_information/configure.html](http://docs.accentos.ru/1.17.0/ru/modules/general-information/general_information/configure.html)