УТВЕРЖДЕНО
Генеральным директором
ООО «ЮНИС Лабс Решение»
А.Г. Ахтырченко

РУКОВОДСТВО ПО РАЗВЕРТЫВАНИЮ

СКРнД «ИСКРА»

(Система Коллективной Работы

над Документами «ИСКРА»)

на Альт Сервер 10

г. Москва
Действительно с 01.11.2024 года

Данная инструкция описывает процесс установки СКРнД «Искра» (далее ИС, Система) в режиме «демонстрационный стенд», когда сама ИС и все необходимые компоненты устанавливаются на одной машине и настраиваются минимально необходимым образом.

Инструкция предназначена для демонстрации логики и порядка действий на простом примере. При развертывании системы в продуктивной среде, в зависимости от вашей инфраструктуры, уже развернутых компонентов и их конфигурации, требований безопасности и других условий, вам потребуется адаптировать данную инструкцию под вашу ситуацию, выполнить дополнительные шаги и/или изменить настройки информационной системы.

Данная инструкция предназначена для исполнения квалифицированными техническими специалистами.

Данная инструкция не включает настройку SSL.

Данная инструкция скорректирована под особенности дистрибутива Альт Сервер 10 (Alt Linux Server 10).
https://www.basealt.ru/alt-server

Оглавление

Вступление

В документе имеется два сценария установки ИС:

- процесс установки по компонентам;
- установка демонстрационной виртуальной машины.

Установка по компонентам — это пример установки ИС и всех необходимых смежных компонент с их минимальной настройкой.

Для этого сценария вам потребуется установленная ОС Альт Сервер 10, и дистрибутив ИС, который вы можете получить по ссылкам с сайта компании-разработчика.

Если вы начинаете внедрение системы в вашей организации, это базовый сценарий, на основе которого вы должны подготовить сценарий развертывания и интеграции компонентов, учитывая особенности вашей инфраструктуры. В таком случае сценарий с установкой виртуальной машины вам не потребуется.

Установка демонстрационной виртуальной машины представляет собой инструкцию «быстрого старта» для тех, кто хочет ознакомиться с системой в кратчайшие сроки. Виртуальная машина является результатом выполнения сценария «установка по компонентам» без каких-либо адаптаций.

Для этого сценария вам потребуется система виртуализации и образ виртуальной машины с установленной ИС, который можно получить по ссылке с сайта компании-разработчика.

Если ваша цель — быстро получить демонстрационный стенд для ознакомления с возможностями системы, сразу переходите к разделу с установкой виртуальной машины. Сценарий с «установкой по компонентам» вам не потребуется.

Установка по компонентам

Формат инструкции

Если не сказано иное, все инструкции начинаются в консоли командной строки, под обычным пользователем, созданным в ходе установки системы.

Команды, подаваемые в консоль указаны в сером блоке, и содержат префикс приглашения командной строки, который показывает уровень пользователя:

- [>] для команд, подаваемых от имени обычного пользователя
- [#] для команд подаваемых от суперпользователя.

Например:

```
> su root
# apt-get update
```

В данном случае вы переключаетесь на суперпользователя «root» и далее выполняете команду уже как суперпользователь.

В некоторых сценариях вы можете столкнуться с другими префиксами, указывающими на то, что вы вводите команды в специализированных консолях, не системных. Например, это могут быть команды SQL-клиента или команды внутри Docker-контейнера.

Например:

```
# docker exec -u 0 -it coolwsd /bin/bash
:/$ cp /opt/collaboraoffice/share/addonScripts/Comments.py /opt/collaboraoffice/share/Scripts/python
:/$  exit
 # docker restart coolwsd
```

В этом случае вы, как суперпользователь, запускаете командную оболочку внутри Docker-контейнера, копируете файлы внутри контейнера, выходите из оболочки внутри контейнера и перезапускаете контейнер уже на хостовой машине.

Содержимое конфигурационных файлов или включений в них приводится на «кремовом» фоне.

Например:

```
{
"authentication":{
   "blockUnknown": true,
   "class":"solr.BasicAuthPlugin",
   "credentials":{"solr":"IV0EHq1OnNrj6gvRCwvFwTrZ1+z1oBbnQdiVC3otuq0=
Ndd7LKvVBAaZIF0QAVi1ekCfAJXr1GGfLtRUXhgrF8c="},
   "realm":"My Solr users",
   "forwardCredentials": false
}}
```

Допольниетльно обратите внимание на раздел связанный с особенностями ОС "Альт Сервер 10" которые могут важны в рамках данной инструкции.

Особенности Alt Server 10 для пользователей других дистрибутивов

Команда [sudo] для пользователя «user» не включена.

Команды [su] или [su root] не переключают контекст и окружение (после выполнения «su» состав доступных команд не меняется, например, команда «useradd» не будет работать).

Для перехода в «root» с изменением контекста, используйте команду [su -].

Особенности первоначальной установки

Предполагается, что вы начинаете работу на системе, установленной с ISO-образа с настройками по умолчанию. Наличие графической подсистемы не критично, хотя в сценарии предполагается, что вы её установили (это опция по умолчанию). В ходе установки системе присвоено имя «**iskra-alt**».

Установка Java

Требуется Java версии 17 или более поздней. Ниже описана установка Java 17, Open JDK из репозиториев ОС "Альт Сервер 10".

```
> su root
```

(Введите пароль рута. Для демо-виртуальной машины используйте пароль ‘toor’.)

Далее выполните следующие команды от суперпользователя:

```
# apt-get update
# apt-get install java17
```

Запрос:

```
# java -version
```

должен выдать ответ, начинающийся с «openjdk version "17.0.12"»

Установка Postgres

```
> su root
```

(Введите пароль рута. Для демо-виртуальной машины используйте пароль ‘toor’.)

Далее выполните следующие команды от суперпользователя:

```
# apt-get update
# apt-get install postgresql16-server
# /etc/init.d/postgresql initdb
# systemctl enable postgresql
# systemctl start postgresql
```

(Добавьте прослушиваемые адреса для сетевого доступа к БД.)

```
# echo "listen_addresses = 'localhost'" >> /var/lib/pgsql/data/postgresql.conf
# systemctl restart postgresql
```

Первичная настройка под СКРнД

(Создайте БД и пользователей.)

```
# createdb -U postgres skrnd
# createdb -U postgres keycloak

# psql -U postgres
```

(Замените пароль на ваш собственный.)

```
postgres=# CREATE USER skrnd WITH PASSWORD 'LvdCispWW4MobXaA' SUPERUSER;
postgres=# grant all privileges on database skrnd to skrnd;
```

(Замените пароль на ваш собственный.)

```
postgres=# CREATE USER keycloak WITH PASSWORD 'yW4msJcrdCHrfYWe' SUPERUSER;
postgres=# grant all privileges on database keycloak to keycloak;
```

Не обязательный блок:

Для целей локального администрирования добавить админ-пользователя.

Замените пароль на ваш собственный.

```
postgres=# CREATE USER admin WITH PASSWORD 'admin' SUPERUSER;
postgres=# grant all privileges on database keycloak to admin;
postgres=# grant all privileges on database skrnd to admin;
postgres=# grant all privileges on database postgres to admin;
```

```
postgres=# exit;
```

(Разрешите доступ по сети. Адреса адаптируйте в соответствии с вашим контекстом.)

```
# echo "host admin all 127.0.0.1/32 md5" >> /var/lib/pgsql/data/pg_hba.conf
# echo "host skrnd all 127.0.0.1/32 md5" >> /var/lib/pgsql/data/pg_hba.conf
# echo "host keycloak all 127.0.0.1/32 md5" >> /var/lib/pgsql/data/pg_hba.conf
# systemctl restart postgresql
```

(Необязательно) Для доступа к базе данных через графический интерфейс (GUI) — установите DBeaver.

```
# apt-get intall dbeaver
```

Установка Keycloak

```
> su root
```

(Введите пароль рута. Для демо-виртуальной машины используйте пароль ‘toor’.)

```
# apt-get update
# apt-get install keycloak
```

Создайте символическую ссылку в каталоге /etc/keycloak, указывающую на /usr/share/keycloak/conf.

```
# ln -s /usr/share/keycloak/conf /etc/keycloak
```

Первичная настройка под СКРнД

Данный сценарий предполагает, что вы устанавливаете сервер KeyCloak для целей разработки и демонстрации работы ИС. Настройка сервера для нужд вашего предприятия должна выполняться специалистами, знакомыми со спецификой вашего бизнеса.

Конфигурационные файлы KeyCloak можно найти в каталоге /usr/share/keycloak/conf.

(В файле keycloak.conf дополните параметры подключения к базе данных, указав имя машины и порт запуска.)

```
# echo "db=postgres" >> /usr/share/keycloak/conf/keycloak.conf
# echo "db-username=keycloak" >> /usr/share/keycloak/conf/keycloak.conf
# echo "db-password=yW4msJcrdCHrfYWe" >> /usr/share/keycloak/conf/keycloak.conf
# echo "db-url=jdbc:postgresql://localhost/keycloak" >> /usr/share/keycloak/conf/keycloak.conf
```

(Вместо iskra-alt и порта 8282 используйте имя своей машины и соответствующий порт.)

```
# echo "hostname=iskra-alt" >> /usr/share/keycloak/conf/keycloak.conf
# echo "http-port=8282" >> /usr/share/keycloak/conf/keycloak.conf
```

```
# echo "http-relative-path=/keycloak" >> /usr/share/keycloak/conf/keycloak.conf
# echo "hostname-strict=false" >> /usr/share/keycloak/conf/keycloak.conf
# cd /usr/share/keycloak/bin/
# ./kc.sh build
```

Запустите сервер в режиме разработчика. Для демо-сервера это допустимо, для «продуктивного контура» настройте конфигурацию по своему усмотрению.

```
# ./kc.sh start-dev
```

После этого перейдите по адресу http://iskra-alt:8282/keycloak (а вот http://localhost:8282/keycloak грузиться не будет, это нормально). Система должна предложить создать пользователя-администратора.

Для демо-сервера используйте имя «kcadmin» и пароль «mNPJmtEHXNHjdn7c». (Можете указать свою комбинацию, но тогда используйте в других местах руководства свой

пароль вместо демо-пароля, указанного здесь.)

Настройка Keycloak под СКРнД

Realm с названием «skrnd»

Realm ID	skrnd
Display name	ИСКРА
HTML Display name	
Frontend URL	http://iskra-alt/keycloak/
Require SSL	none

Client-запись skrnd-front-client

Client type	**OpenID Connect**
Client ID	**skrnd-front-client**
Name	skrnd-front-client
Description	skrnd-front-client
Always display in UI	false (off)

Client authentication	off
Authorization	off
Authentication flow (отметьте галочками перечисленные опции)	**[V]** Standard flow **[V]** Implicit flow **[V]** Direct access grants

Root URL	http://iskra-alt/
Home URL	http://iskra-alt/
Valid redirect URIs	http://iskra-alt/* http://iskra-alt:8282/* http://iskra-alt:8080/*
Web origins	http://iskra-alt:8282/* http://iskra-alt:8080/* http://iskra-alt/*

После прохождения мастера и открытия карточки обратите внимание на значение адреса, по которому можно проводить администрирование. Для демо-стенда допустимо оставить этот параметр как он подставляется по умолчанию.

Admin URL	http://iskra-alt:8282

Client-запись «skrnd-localhost-test-client »

Client type	**OpenID Connect**
Client ID	skrnd-localhost-test-client
Name	**skrnd-localhost-test-client**
Description	**skrnd-localhost-test-client**
Always display in UI	false (off)

Client authentication	off
Authorization	off
Authentication flow (отметьте галочками перечисленные опции)	**[V]** Standard flow **[V]** Implicit flow **[V]** Direct access grants

Root URL	http://localhost:8080/
Home URL	http://localhost:8080/
Valid redirect URIs	http://localhost:8080/* http://localhost:8081/* http://localhost:4200/*

Раздел «Realm Roles»

Создайте 3 роли в разделе «Realm Roles».

Role name	Description
admin	Роль для администрирования системы: технический доступ ко всем данным.
pm	Руководители проектов: могут создавать новые проекты.
user	Рядовой пользователь ИС: назначается всем. Имеет право авторизовываться, смотреть списки проектов, получать права в их рамках, выполнять операции с документами. Снятие роли приводит к невозможности входа в систему, даже если авторизация пройдена.

Учетные записи демонстрационных пользователей

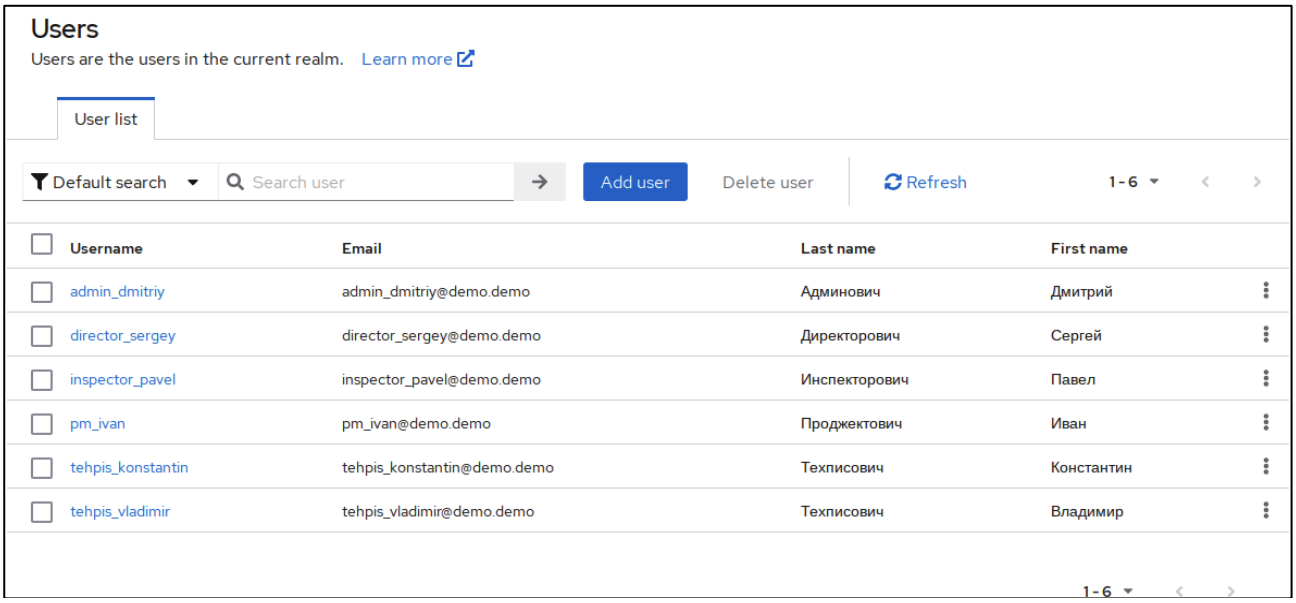
Данный пункт выполняется только при создании демонстрационного или тестового стенда.

Username и **Email**	**Отображаемое имя** (First name + Last name)	**Глобальные роли**	Рекомендуемые роли в проекте
director_sergey director_sergey@demo.demo	Сергей Директорович	user	**Куратор** Согласующий
pm_ivan pm_ivan@demo.demo	Иван Проджектович	user, pm	Куратор, **Ответственный**, Согласующий
tehpis_vladimir tehpis_vladimir@demo.demo	Владимир Техписович	user	Ответственный, **Уч.раб.группы**
tehpis_konstantin tehpis_konstantin@demo.demo	Константин Техписович	user	Ответственный, **Уч.раб.группы**
inspector_pavel inspector_pavel@demo.demo	Павел0 Инспекторович	user	**Согласующий**
admin_dmitriy admin_dmitriy@demo.demo	Дмитрий Администрович	admin, user	-

Для всех демонстрационных пользователей установите один пароль: «**123qwe»** и выставьте опцию “Email verified ” в “Yes”.

Примечание: ранее существовавший пользователь «kcadmin» с паролем «mNPJmtEHXNHjdn7c» используется для администрирования Keycloak и относится к «Realm» Keycloak. Авторизоваться под ним в ИС не получится.

В результате должно получиться следующее:



Создание службы Keycloak

! В данном разделе описывается упрощенная конфигурация Keycloak.
! Не используйте её для продуктивных конфигураций.

Скопируйте файл описания сервиса Keycloak из папки дистрибутива: ./keycloak/keycloak_devmode_quarkus_alt10.service в папку /etc/systemd/system/

```
# su root

# cp /home/user/skrnd_distr/keycloak/keycloak_devmode_quarkus_alt10.service
/etc/systemd/system/
```

Создайте рабочую директорию и назначьте права на каталоги, которые должны быть доступны пользователю, под которым будет работать сервис.

```
# mkdir /var/keycloak
# chown -R user:user /var/keycloak
# chmod -R 664 /var/keycloak
# chown -R user:user /usr/share/keycloak/lib
# chmod -R 664 /usr/share/keycloak/lib
# chmod -R a+X
```

Включите сервис и запустите его.

```
# systemctl enable keycloak_devmode_quarkus_alt10
# systemctl start keycloak_devmode_quarkus_alt10
```

Проверьте работу сервиса с помощью HTTP-запроса https://localhost:8282/ и диагностируйте ошибки запуска с помощью команды:

```
# journalctl -xeu keycloak_devmode_quarkus_alt10.service
```

Известные проблемы

Может возникнуть конфликт при запуске coolwsd в Docker, если сначала запустить coolwsd. В этом случае Keycloak может выдавать ошибки при авторизации, требуя HTTPS. Чтобы решить эту проблему, необходимо остановить контейнер coolwsd, перезапустить Keycloak, а затем запустить coolwsd снова.

```
# systemctl stop coolwsd_docker_alt10
# systemctl stop docker

# systemctl stop keycloak_devmode_quarkus_alt10
# systemctl start keycloak_devmode_quarkus_alt10

# systemctl start docker
# systemctl start coolwsd_docker_alt10
```

Установка LibreOffice

Скачайте с официального сайта (https://www.libreoffice.org/download/download-libreoffice/) rpm-пакет актуальной версии LibreOffice и сохраните его во временный каталог в домашнем каталоге пользователя (~/tmp).

Распакуйте .tar.gz архив рядом.

Скачайте языковой пакет при необходимости.

На момент написания данного документа актуальной была версия 24.8.2

Повторить описанное выше в командах можно так:

```
> mkdir ~/tmp
> cd ~/tmp
> wget http://libreoffice-
mirror.rbc.ru/pub/libreoffice/libreoffice/stable/24.8.2/rpm/x86_64/LibreOffice_24.8.2_Linux_x86-64_rpm.tar.gz
```

```
> tar -xf LibreOffice_24.8.2_Linux_x86-64_rpm.tar.gz
> cd ./LibreOffice_24.8.2.1_Linux_x86-64_rpm/RPMS/
> su root
# apt-get install *.rpm
```

Установите языковой пакет:

```
> cd ~/tmp
> wget http://libreoffice-
mirror.rbc.ru/pub/libreoffice/libreoffice/stable/24.8.2/rpm/x86_64/LibreOffice_24.8.2_Linux_x86-64_rpm_langpack_ru.tar.gz
```

```
> tar -xf  LibreOffice_24.8.2_Linux_x86-64_rpm_langpack_ru.tar.gz
> cd ./LibreOffice_24.8.2.1_Linux_x86-64_rpm_langpack_ru/RPMS/
> su root
# apt-get install *.rpm
```

LibreOffice был установлен в каталог /opt/libreoffice24.8/program. Этот каталог потребуется при настройке переменных окружения при запуске Tomcat.

Установка Collbora/CODE

Установка Docker

Обновите индекс пакетов и установите Docker.

```
# apt-get update
# apt-get install docker-engine
```

Запустите сервис контейнеризации Docker и добавьте его в автозагрузку.

```
# systemctl enable --now docker
```

Проверьте установку, запустив Docker и статус запущенной службы.

```
# systemctl status docker
```

Статус должен отображать «**active (running)**».

Чтобы получить информацию об установленном Docker, выполните следующую команду:

```
# docker info
```

При правильной настройке вы получите соответствующий ответ от сервиса Docker.

Установка Collabora CODE «в контейнере»

Скачайте образ Collabora CODE из репозитория докер-образов.

```
# docker pull collabora/code
```

Убедитесь, что образ успешно скачан.

```
# docker images
```

Создайте временный контейнер для извлечения конфигурации.

```
# docker run -it -d --name tmpcool -p 9980:9980 collabora/code
```

Создайте копию конфигурационных файлов из контейнера, скопируйте каталог скриптов, удалите временный контейнер и исправьте права доступа:

```
# mkdir /etc/coolwsd/
# docker cp tmpcool:/etc/coolwsd/ /etc/

# mkdir -p /opt/collaboraoffice/share/addonScripts/


# docker stop tmpcool
# docker remove tmpcool

# chown -R root:docker /etc/coolwsd
# chmod -R 774 /etc/coolwsd
```

Пересоздайте контейнер с постоянным именем, подключите конфигурацию с хостовой машины и настройте другие опции.

```
# docker run -it -d -v /etc/coolwsd:/etc/coolwsd -v
/opt/collaboraoffice/share/addonScripts:/opt/collaboraoffice/share/addonScripts  --name
coolwsd  -p 9980:9980 -e 'domain=iskra-alt' --restart always --cap-add MKNOD --add-host
"iskra-alt:host-gateway"  -e "username=admin" -e "password=pass" collabora/code
```

(Эта команда создания контейнера используется в нескольких местах руководства, но приведена только здесь в единственном экземпляре, и потому она выделена, чтобы не затеряться.)

Обратите внимание: тут задан пользователь и пароль доступа к админской консоли Collabora: «**admin | pass**».

Убедитесь, что контейнер создан, и проверьте его статус.

```
# docker ps -a
```

При необходимости, можно проверить конфигурацию контейнера

```
# docker inspect coolwsd
```

Проверить работу Collabora через HTTP-вызов https://localhost:9980/ должно вернуть «**ok**» на странице.

Установка скриптов для Collabora/CODE

Установите python-скрипты. Скопируйте файлы скриптов (./coolwsd_python/*) из каталога дистрибутива (/home/user/skrnd_distr/) в каталог скриптов Collabora-сервера.

```
# cp -rT  /home/user/skrnd_distr/coolwsd_python/ /opt/collaboraoffice/share/addonScripts
```

Установите в Docker дополнительные пакеты для поддержки скриптов.

```
# docker exec -u 0 -it coolwsd /bin/bash
:/$ apt-get update
:/$ apt-get install collaboraofficebasis-python-script-provider
:/$ apt-get install collaboraofficebasis-pyuno
```

Продолжая сеанс в контейнере, выполните **docker restart coolwsd** и установите права для пользователя **cool** и группы **cool** (из-за несовпадения пользователей у хостовой машины и контейнера это нужно сделать «изнутри»).

```
:/$ cp /opt/collaboraoffice/share/addonScripts/Comments.py /opt/collaboraoffice/share/Scripts/python
:/$ chown cool:cool /opt/collaboraoffice/share/Scripts/python/Comments.py
:/$ chmod 664  /opt/collaboraoffice/share/Scripts/python/Comments.py

:/$  exit
```

Перезапустите контейнер.

```
# docker restart coolwsd
```

Настройка Collabora/CODE

Замените **имя сервера:**

```
# docker exec -it coolwsd /bin/bash
:/$ coolconfig set server_name iskra-alt
:/$ exit
```

или установите это имя через установку значения server_name в /etc/coolwsd/coolwsd.xml

(! в описании ниже приводятся не все атрибуты, только имена тегов без дополнительных атрибутов !)

```
<server_name desc="External hostname:port of the server running coolwsd. ..." >iskra-alt</server_name>
```

Установите Content Secure Policy в /etc/coolwsd/coolwsd.xml

(! в описании ниже приводятся только имена тегов без дополнительных атрибутов !)

```
<content_security_policy ... >
frame-ancestors 'self' http://127.0.0.1:9980/;
frame-ancestors 'self' http://iskra-alt:*/;
frame-ancestors 'self' http://localhost:*/;
frame-ancestors 'self' https://localhost:*/;
</content_security_policy>
```

Выставить режим доступа **alias_groups** в **groups** в **/etc/coolwsd/coolwsd.xml**. Найдите одноименный тег и замените там значение на **groups**.

Добавьте группы хостов с допустимыми источниками в теге «**Backend storage**» в теге **group /etc/coolwsd/coolwsd.xml**.

(! в описании ниже приводятся только имена тегов, без дополнительных атрибутов!)
(Добавляемое выделено жирным.)

```
<storage desc="Backend storage">
...
  <wopi ... >
    ...
    <alias_groups ... >
      <group>
       <host  ...>https://iskra-alt:443</host>
       <alias ...>http://iskra-alt</alias>
       <alias ...>http://127.0.0.1:80</alias>
       <alias ...>https://localhost:80</alias>
      </group>
    </alias_groups>
    ...
  </wopi >
</storage>
```

В теге **ssl** - исправьте **enable** на **false** и **termination** на **true** в **/etc/coolwsd/coolwsd.xml**.

(! в описании ниже приводятся только имена тегов без дополнительных атрибутов !)

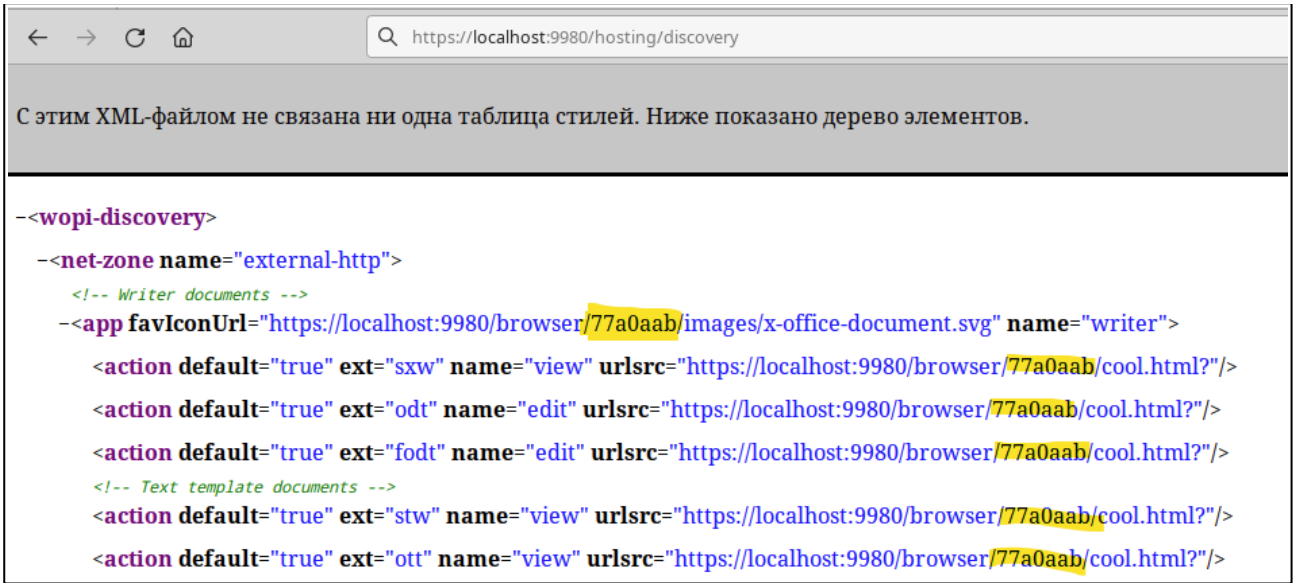
```
<ssl ... >
    <enable      ...>false</termination>
    <termination ...>true</termination>
</ssl ... >
```

По окончании — перезапустить coolwsd:

```
# docker restart coolwsd
```

Получить код инсталляции

Для обращения к серверу Collabora нам потребуется «код инсталляции». Получить его можно из «wopi-discovery» XML, который можно получить через http://localhost:9980/hosting/discovery.



В указанном примере это «77a0aab». В вашем случае будет другой код. Запишите его, он потребуется при настройке конфигурации ИС.

Проверка работоспособности

После настройки по данному сценарию, Collabora будет использовать незащищенный режим:

- http://localhost:9980/ - отдаст «ок»;
- http://localhost:9980/hosting/discovery - покажет «wopi-discovery» XML;
- http://iskra-alt:9980/ - отдаст «ок»;
- http://iskra-alt:9980/hosting/discovery - покажет «wopi-discovery» XML.

А после настройки nginx на последующих шагах будет работать:

- http://iskra-alt/hosting/discovery - покажет «wopi-discovery» XML.

Работоспособность последней точки доступа является целевым состоянием.

Диагностика проблем

Если имеются ошибки, запустите контейнер с выводом логов в консоль и внимательно изучите полученный вывод.

```
# docker start -a coolwsd
```

Лог работы Collabora можно получить так:

```
# docker logs coolwsd -f
```

Запуск консоли внутри контейнера:

```
# docker exec -it coolwsd /bin/bash
```

Запуск консоли внутри контейнера с правами суперпользователя (root):

```
# docker exec -u 0 -it coolwsd /bin/bash
```

Доступ к контрольной панели сервера Collabora можно получить по адресу: http://iskra-alt:9980/browser/dist/admin/admin.html. Имя пользователя и пароль — как указаны в параметрах запуска контейнера (admin | pass).

В контрольной панели сервера Collabora, в частности, можно изменить уровень логирования системы.

Исправление известных ошибок при использовании collabora в docker-контейнере

В ситуации переполнения дискового пространства в разделе **/var** из-за ошибок в работе образа Collabora (диагностируется через команду **df -h** и наблюдении быстрого роста заполненного места на файловых разделах, связанных с Docker), выполните следующие действия:

- Остановите контейнер coolwsd;
- Удалите его;
- Создайте контейнер заново.

```
# systemctl stop docker
# systemctl start docker
# docker stop coolwsd
# docker remove coolwsd
```

Далее пересоздайте контейнер с помощью команды, указанной в разделе **«Установка Collabora/CODE в контейнере»** в пункте «**Пересоздайте контейнер уже с постоянным именем»**.

Если это не освобождает занятое место, можно полностью **сбросить рабочее состояние Docker — удалить все его рабочие каталоги, все закэшированные образы, конт**ейнеры и другие материалы.

```
# systemctl stop docker
# rm -rf /var/lib/docker/
# systemctl start docker
```

После этого потребуется выкачать образ Collabora и заново создать контейнер (повторить часть операций, описанных выше).

```
# docker pull collabora/code
```

Далее пересоздайте контейнер с помощью команды, указанной в разделе **«Установка Collabora/CODE в контейнере»** в пункте «**Пересоздайте контейнер уже с постоянным именем»**.

Установка Apache Solr

Создайте пользователя для запуска Solr:

```
# su -

# adduser --system --home /var/solr --create-home --shell /bin/bash solr
# exit
```

Загрузите актуальный дистрибутив Solr с https://solr.apache.org/downloads.html и поместите его в папку /opt. На данный момент это solr-9.7.0.tgz. Можно также, если сервер имеет доступ к интернету, загрузить дистрибутив непосредственно на сервер:

```
> su

# cd ~/tmp
# wget -O solr-9.7.0.tgz https://www.apache.org/dyn/closer.lua/solr/solr/9.7.0/solr-
9.7.0.tgz?action=download
```

Распакуйте дистрибутив и создайте симлинк:

```
# cd ~/tmp
# tar xzf solr-9.7.0.tgz
# mv ./solr-9.7.0 /opt
# ln -s /opt/solr-9.7.0 /opt/solr
```

Подготовьте рабочие папки solr:

```
# mkdir -p /opt/solr_wrkdir/data
# mkdir -p /opt/solr_wrkdir/logs
# chown -R solr:solr /opt/solr_wrkdir
```

Скопируйте файлы конфигурации и службы из дистрибутива solr:

```
> cp /opt/solr/bin/init.d/solr /etc/init.d
> cp /opt/solr/bin/solr.in.sh /etc/default/
```

Добавьте в конец файла **/etc/default/solr.in.sh** строки:

```
SOLR_JETTY_HOST="iskra-alt"
SOLR_HOME="/opt/solr_wrkdir/data"
SOLR_LOGS_DIR="/opt/solr_wrkdir/logs"
SOLR_PID_DIR="/var/solr"
```

Разрешить запуск службы Solr, запустите службу, убедитесь, что служба запущена:

```
# systemctl enable solr
# systemctl start solr
# systemctl status solr
```

Проверить доступность ссылки приложения http://iskra-alt:8983. Доступ может быть не сразу. Интерфейс будет доступен без авторизации.

Диагностика запуска и логи

Логи сервиса можете смотреть через:

```
# sournalctl -u solr -f
```

Первичная настройка

Включение аутентификации (basic authentication)

Необходимо создать файл /opt/solr_wrkdir/data/security.json

```
> touch /opt/solr_wrkdir/data/security.json
> chown solr:solr /opt/solr_wrkdir/data/security.json
```

и наполнить его следующим содержанием:

```
{
"authentication":{
   "blockUnknown": true,
   "class":"solr.BasicAuthPlugin",
   "credentials":{"solr":"IV0EHq1OnNrj6gvRCwvFwTrZ1+z1oBbnQdiVC3otuq0=
Ndd7LKvVBAaZIF0QAVi1ekCfAJXr1GGfLtRUXhgrF8c="},
   "realm":"My Solr users",
   "forwardCredentials": false
},
"authorization":{
   "class":"solr.RuleBasedAuthorizationPlugin",
   "permissions":[{"name":"security-edit",
      "role":"admin"}],
   "user-role":{"solr":"admin"}
}}
```

Это задает единственного пользователя «**solr**» с паролем «**SolrRocks**».

Перезапустите службу:

```
> systemctl restart solr
```

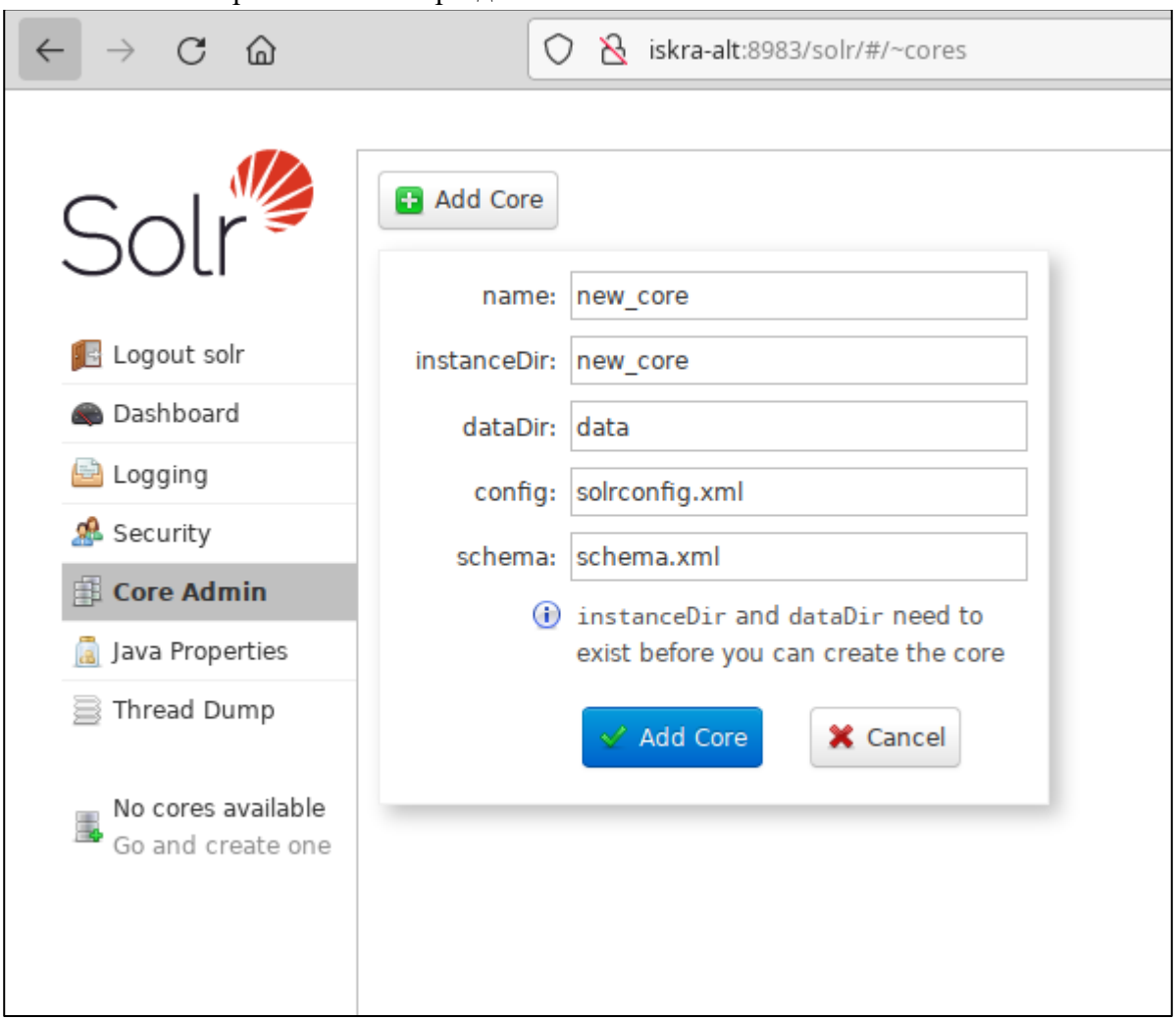
Проверьте аутентификацию: http://iskra-alt:8983. После успешной аутентификации пользователи могут быть добавлены или изменены через веб-интерфейс.

Создание коллекции «newcollection»

Скопируйте дефолтную конфигурацию на доступные серверы.

```
# cp -r /opt/solr/server/solr/configsets/_default/conf/* /opt/solr/server/resources
```

Зайдите в панель администрирования Solr по адресу http://iskra-alt:8983 и авторизуйтесь как solr. Затем переключитесь в раздел «Core Admin».

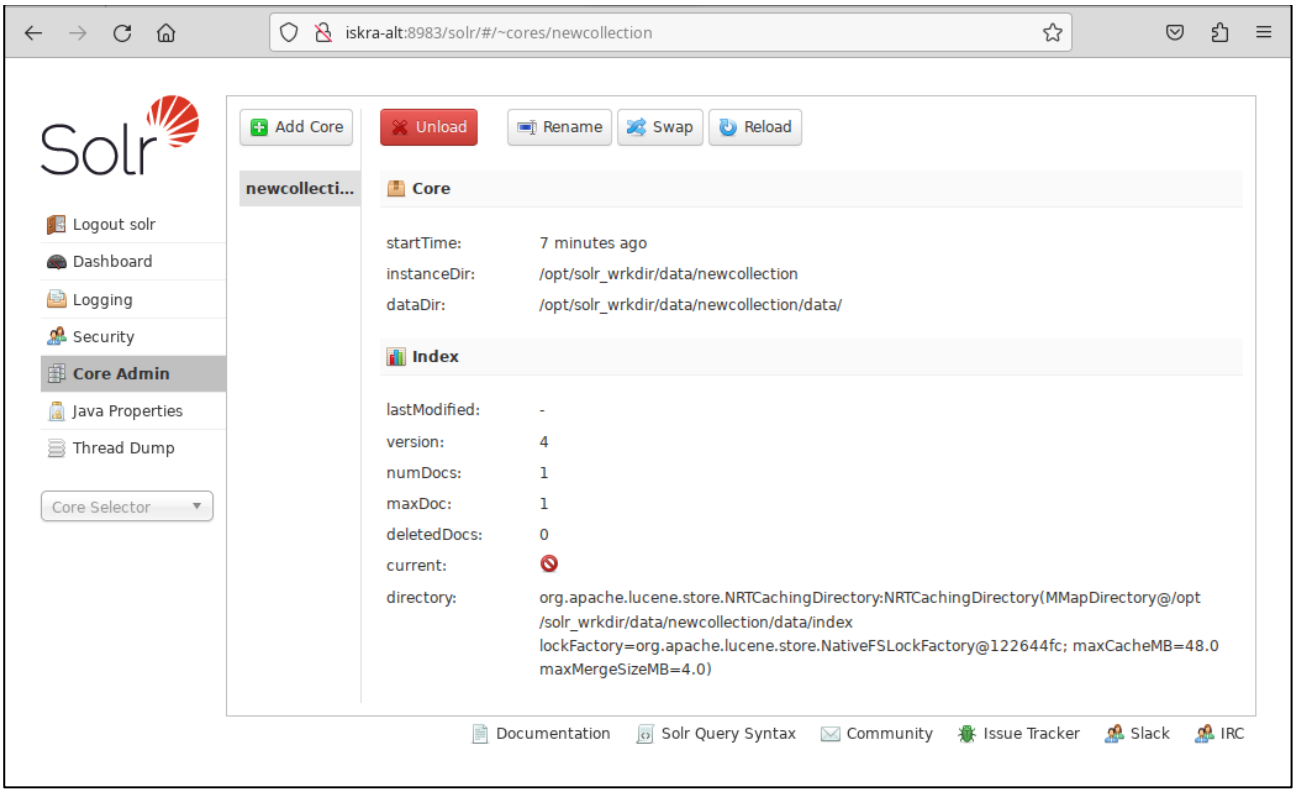


и создайте новое «ядро» (коллекцию).

параметр	значение
name	newcollection
instanceDir	/opt/solr_wrkdir/data/newcollection
dataDir	/opt/solr_wrkdir/data/newcollection/data

Параметры «**config**» и «**schema**» оставьте без изменений.

По завершении должно получиться так:



Установка Tomcat и «бек»-модуля «Искры»

Требуется Tomcat версии 10 или более поздней. Установим Tomcat 10 из стандартных программных репозиториев.

```
> su root
```
(Введите пароль рута. Для демо-виртуальной машины используйте пароль ‘toor’.)

Далее выполните следующие команды от суперпользователя:

```
# apt-get update
# apt-get install tomcat10 tomcat10-admin-webapps
```

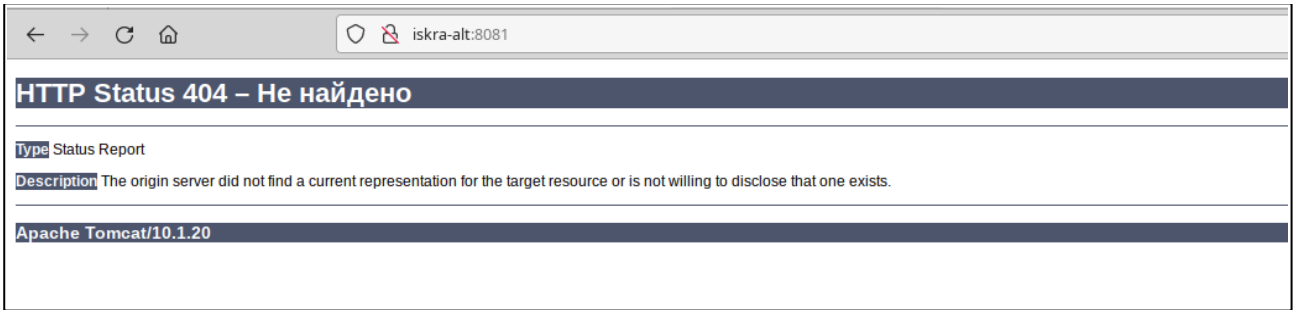
Так как на порту 8080 в Alt Server 10 работает сервис администрирования, необходимо изменить порт 8080 в файле /etc/tomcat/server.xml на другой. Используйте порт 8081:

```
<Connector port="8081" protocol="HTTP/1.1"
           connectionTimeout="20000"
           redirectPort="8443" />
```

Сервис Tomcat:

```
# systemctl enable --now tomcat
```

После этого запрос в браузере по http://iskra-alt:8081/ должен отдать страницу ошибки 404, но сгенерированную сервером Tomcat:



Диагностика проблем запуска

Лог Tomcat в Alt Server 10 пишется в журнал служб, и для его просмотра используйте команду journalctl:

```
# journalctl -u tomcat -f
```

В данном случае это аналог команды tail -f ./catalina.out, используемой в других дистрибутивах, которая выполняется в каталоге логов Tomcat.

Для Tomcat в Alt Server 10 в дефолтных настройках в каталоге логов вообще отсутствует файл catalina.out. Вместо этого есть только файлы по датам, но они не обновляются непрерывно.

Переменные окружения для запуска ИС

Для запуска ИС в текущей версии требуется указать ряд параметров подключения через переменные окружения.

Сам файл сервиса размещается в /lib/systemd/system/timcat.service и импортирует файл настроек из /etc/tomcat/tomcat.conf.

Добавьте в конец /etc/tomcat/tomcat.conf следующие строки:

```
#####################################################
# ISKRA ATTRS

COLLABORA_URL=http://iskra-alt
DB_HOST=127.0.0.1
DB_NAME=skrnd
DB_PASSWORD=skrnd
DB_PORT=5432
DB_SCHEMA=public
DB_USER=skrnd

#это адрес, по которому сервер будет связываться с кейклоаком для проверки jwt-токена
KEYCLOAK_BASE_URL=http://iskra-alt/keycloak

#это строка, по которой будет узначаться «свой» jwt-токен.
#должно быть равно одноименному полю в jwt.
KEYCLOAK_ISSUER_URI=http://iskra-alt/keycloak/realms/skrnd

KEYCLOAK_REALM=skrnd

LOCAL_STORAGE_LOCATION=/opt/storage
SOLR_URL=http://iskra-alt:8983/solr/newcollection
SOLR_BASE_URL=http://iskra-alt:8983/solr/
#SOLR_PASSWORD=QAZwsx123
SOLR_PASSWORD=SolrRocks
SOLR_USERNAME=solr

#тут надо указать путь до каталога с бинарными файлами либореофис
CLASSPATH=/opt/libreoffice24.8/program

#тут тоже надо указать путь до каталога с бинарными файлами либореофис
LIBREOFFICE_BIN_PATH=/opt/libreoffice24.8/program

# //ISKRA ATTRS
#####################################################
```

Обратите внимание: данный набор значений соответствует порядку установки, изложенному в данном сценарии, и строго соответствует данному дистрибутиву.

Путь до каталога LibreOffice настраивается в двух местах:

1. Надо прописать путь до LibreOffice в двух переменных: LIBREOFFICE_BIN_PATH здесь и в CLASSPATH, который используется Java. Эти значения должны быть одинаковыми;
2. Если у вас другой каталог, вам необходимо указать тот, в котором находится исполняемый бинарный файл soffice (например, soffice.exe для Windows).

Затем необходимо перезапустить сервис:

```
# systemctl daemon-reload
# systemctl restart tomcat
```

Рабочие каталоги для запуска СКРнД

Создайте каталоги для хранения файлов в /opt/storage.

Затем необходимо перезапустить сервис.

```
# mkdir p /opt/storage
# chown -R tomcat:tomcat /opt/storage
# chmod -R 775 /opt/storage
```

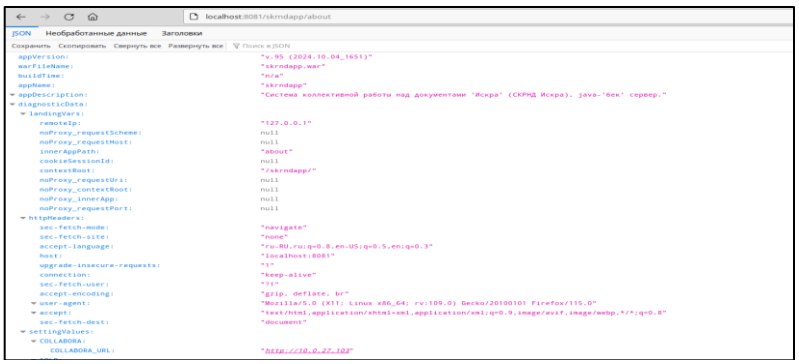
Установка «back»-модуля «Искры»

Из каталога дистрибутива (/home/user/skrnd_distr/) возьмите файл .war-архива бэк-приложения (./back/skrndapp.war) и скопируйте его в /usr/share/tomcat/webapps/.

```
# chmod a+r-x /home/user/skrnd_distr/back/skrndapp.war
# cp /home/user/skrnd_distr/back/skrndapp.war /usr/share/tomcat/webapps/
```

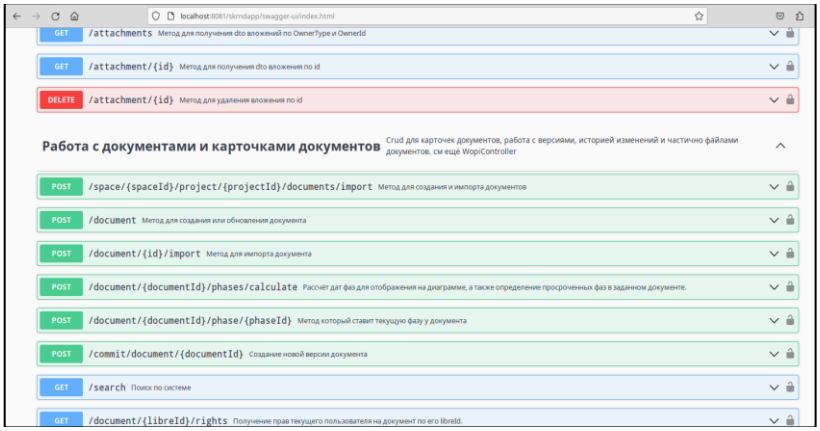
При успешном запуске — вы увидите появление каталога ./SKRNDAPP в каталоге приложений. Также вы сможете просмотреть конфигурацию бэк-системы по запросу http://localhost:8081/skrndapp/about.

Используйте этот экран для контроля значений переменных, которые сейчас используются в ИС.



Убедитесь, что примененные параметры соответствуют тем, что были указаны в конфигурации в переменных окружения.

Далее убедитесь, что работает swagger-описание API для фронта по адресу http://localhost:8081/skrndapp/swagger-ui/index.html.



После первого запуска данное описание может загрузиться не сразу. Если не получилось, повторите запрос через полминуты.

Установка Nginx

Установите пакет nginx:

```
> su root
# apt-get update
# apt-get install nginx
```

Запустить Nginx и добавить его в автозагрузку:

```
# systemctl enable --now nginx
```

Установка «front»-модуля «Искры»

Создайте каталоги для веб-интерфейса:

```
# mkdir -p /var/www/skrnd.config
# mkdir -p /var/www/skrnd.front
```

Скопируйте файлы конфигурации фронтового приложения из каталога дистрибутива (./front_config/front-kc-config.json и ./front_config/collabora_config.json) в каталог настроек фронтового приложения /var/www/skrnd.config/.

Считаем, что дистрибутив размещен в каталоге /home/user/skrnd_distr:

```
# chmod a+r-x /home/user/skrnd_distr/front_config/front-kc-config.json
# cp /home/user/skrnd_distr/front_config/front-kc-config.json /var/www/skrnd.config/

# chmod a+r-x /home/user/skrnd_distr/front_config/collabora_config.json
# cp /home/user/skrnd_distr/front_config/collabora_config.json /var/www/skrnd.config/

# chown -R tomcat:tomcat /var/www/skrnd.config
```

Отредактируйте файл collabora_config.json — замените код инсталляции Collabora на тот, который вы наблюдаете при проверке работоспособности Collabora (http://localhost:9980/hosting/discovery — см. раздел по развертыванию Collabora, ссылка на «wopi discovery» XML):

```
{
  "collaboraUrl": "http://iskra-alt:9980/browser/77a0aab/cool.html?lang=ru&WOPISrc=http://iskra-
```

```
alt:8081/skrndapp/wopi/files/"
}
```

Проверьте содержимое front-kc-config.json, если имя сервера или порты, используемые вами, отличаются от указанных в данном руководстве.

Скопируйте файл фронтового приложения из каталога дистрибутива (./front/app.zip) в каталог фронтового приложения /var/www/skrnd.front/ и распакуйте его.

```
# chmod a+r-x /home/user/skrnd_distr/front/app.zip
# cp /home/user/skrnd_distr/front/app.zip /var/www/skrnd.front/
# cd /var/www/skrnd.front/
# unzip ./app.zip
# rm ./app.zip
# rm -rf ./WEB-INF
```

Установка конфигурации nginx

Скопируйте файл конфигурации nginx для Alt Server 10 из дистрибутива (./nginx/iskra.alt10.conf) в каталог /etc/nginx/sites-available.d/.

```
# cp /home/user/skrnd_distr/nginx/skrnd.alt10.conf /etc/nginx/sites-available.d/
```

Выключите Apache2:

```
# systemctl stop httpd2
# systemctl disable httpd2
```

Скопируйте файл конфигурации как единственный конфиг в каталог активированных конфигураций сайтов:

```
# ln -s /etc/nginx/sites-available.d/skrnd.alt10.conf  /etc/nginx/sites-enabled.d/
```

Перезапустите nginx:

```
# systemctl restart nginx
```

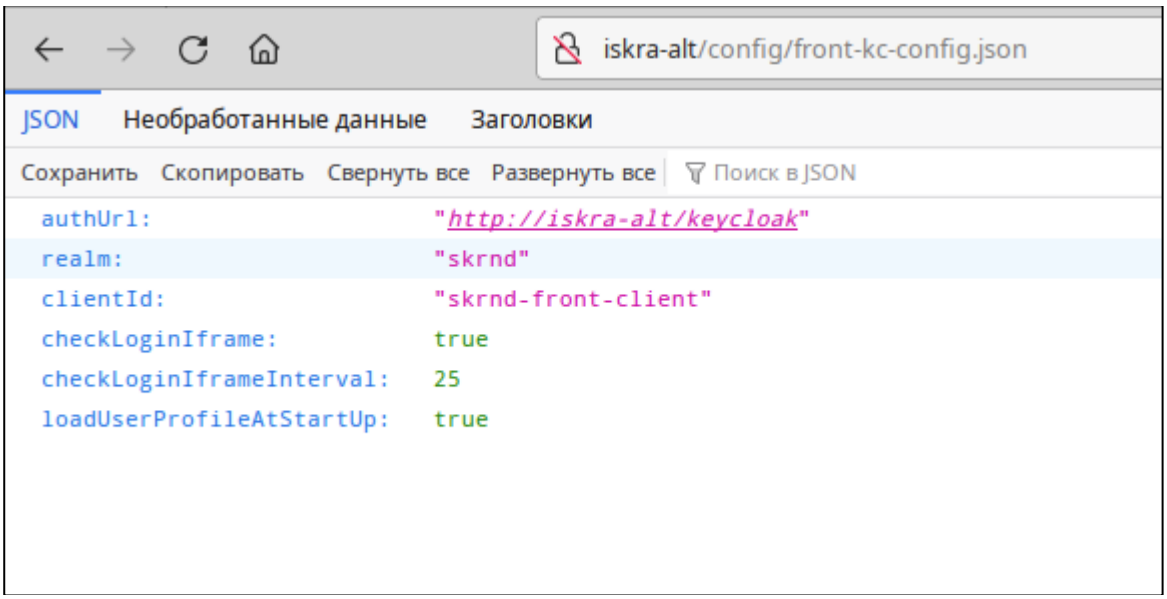
```
# systemctl status nginx
```

Проверьте, что конфигурация работает:

1. Сделайте базовые запросы файлов:

- http://iskra-alt/config/front-kc-config.json
- http://iskra-alt/config/collabora_config.json

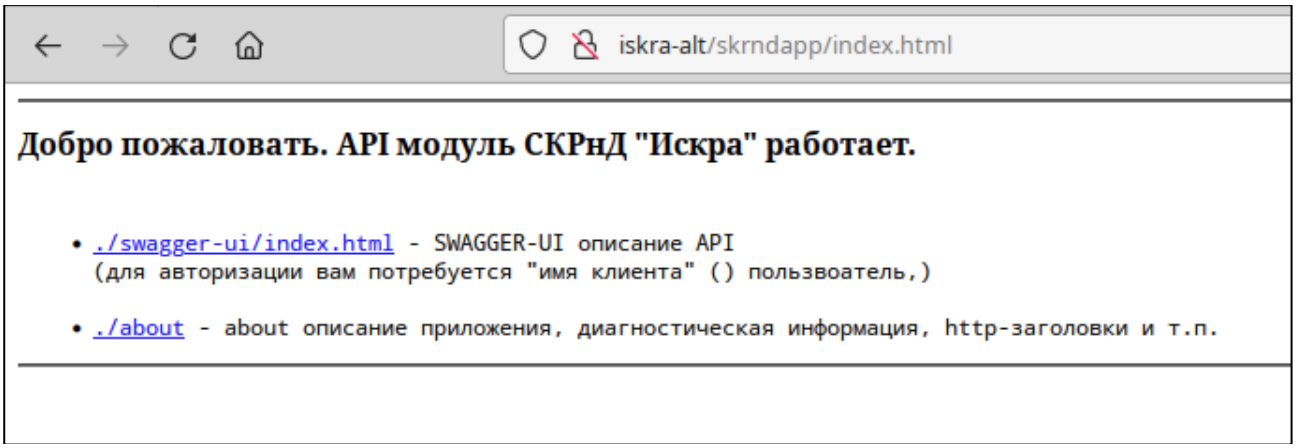
Эти запросы должны выдать файлы конфигурации для «фронта».



2. Редирект на «бек-модуль»:

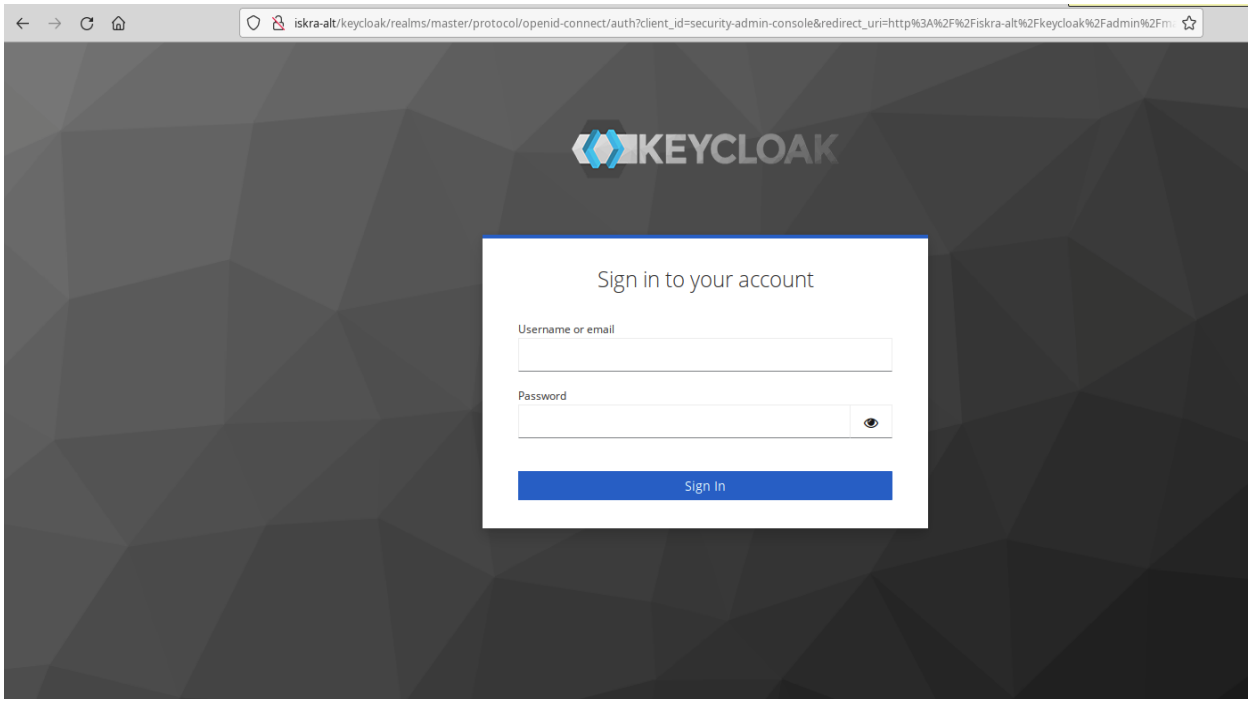
- http://iskra-alt/skrndapp — выдаст лендинг страницу бек-модуля.

Проверьте, что обе ссылки работают.



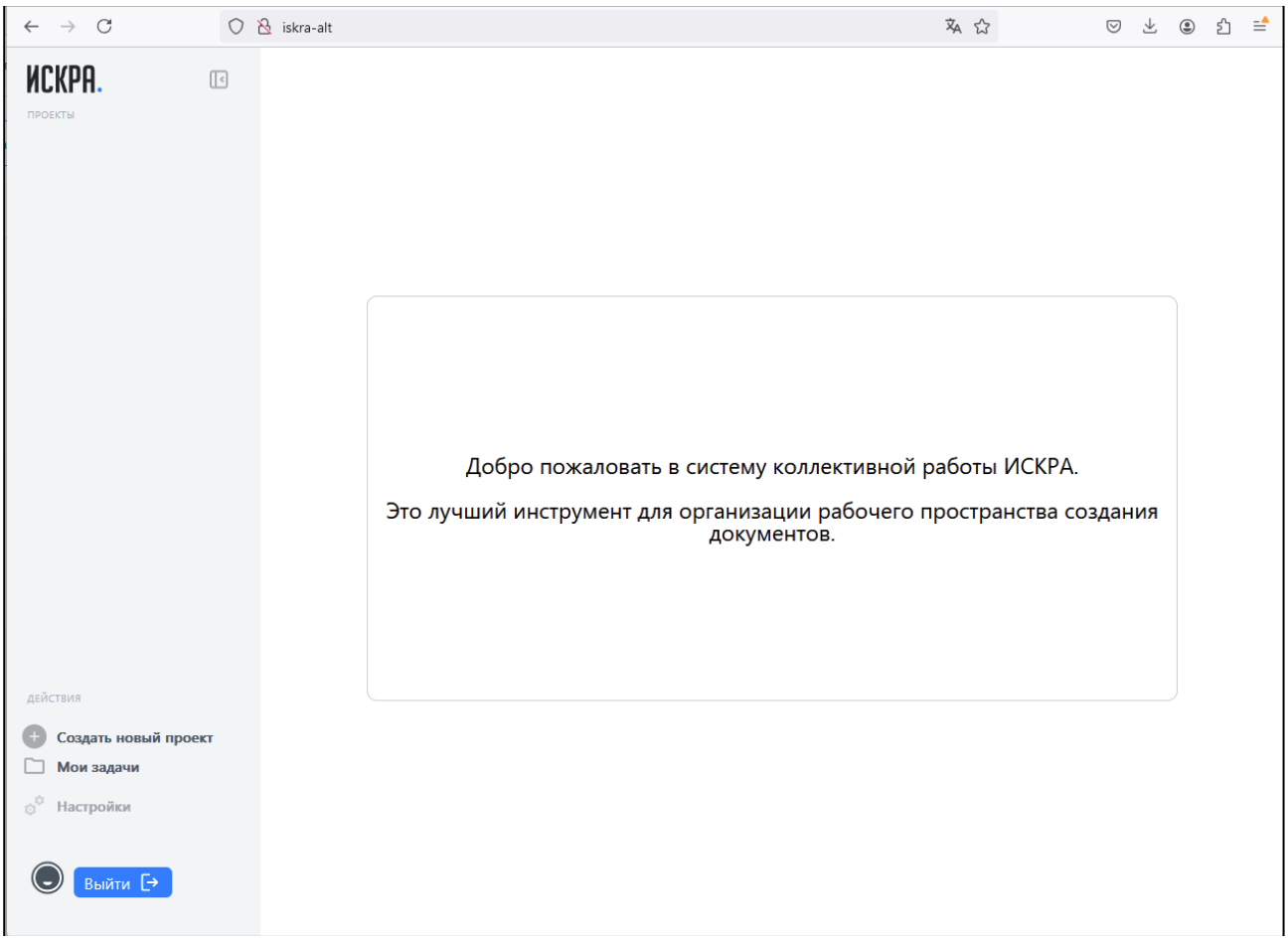
3. Редирект на админ панель Keycloak:

- http://iskra-alt/keycloak/ должен выдать логин на Keycloak, как и http://iskra-alt:8282/keycloak/.



4. **Авторизация во фронте и отображение стартового экрана:**

- http://iskra-alt/ направит на авторизацию в Keycloak. Используйте любого демо-пользователя, определенного при настройке Keycloak, после чего отобразится рабочее окно «Искры».



Обновление СКРнД «Искра»

Для обновления «Искры», установленной в рамках описанного сценария, необходимо выполнить следующие шаги:

1. Заменить файл веб-архива бэк-приложения. Файл веб-архива серверного приложения (./back/skrndapp.war) необходимо скопировать в каталог приложений Tomcat: /usr/share/tomcat/webapps/.
2. Заменить файлы «фронта». Необходимо очистить содержимое каталога /var/www/skrnd.front и разместить в нем содержимое архива фронтового приложения ./front/app.zip из комплекта дистрибутива.
3. Перезапустить Tomcat:

```
# systemctl restart tomcat
```

Перезапуск Tomcat необходим из-за использования нативных библиотек LibreOffice в API-клиенте LibreOffice. Диагностика проблем запуска осуществляется аналогичным образом, как описано в разделе установки Tomcat.

Виртуальная машина с предустановленной системой «Искра»

Общие сведения

Для демонстрации системы и развертывания на ресурсах клиента доступен образ виртуальной машины с установленной и настроенной ИС и пакетом программных продуктов.

Эта машина создана в соответствии с инструкцией по развертыванию демонстрационного сервера системы и содержит все необходимые компоненты, работающие в минимальной конфигурации.

Внимание! Не используйте образ виртуальной демо-машины для продуктивных задач!

Виртуальная машина поставляется как образ в «формате открытой виртуализации» (.ova), версии 1.0.

Образ системы, построенной на базе ОС Альт Сервер 10, поставляется в файле с именем вида [iskra-alt]_ISKRA_AltLinux10_DEMO.2024-10-24_2130.ova1.0.ova. Год, месяц и день генерации будут различаться для разных версий образа.

Рекомендуется предоставить для виртуальной машины следующие ресурсы:

- 4-8 вычислительных ядер процессора;
- 10-15 ГБ оперативной памяти;
- выделить 150 ГБ свободного места на диске (это максимальный объем дискового пространства, определенный в настройках виртуальной машины).

Использование

Имя машины: **iskra-alt** (для системы на базе Альт Сервер 10).

Суперпользователь «**root»**, пароль «**toor»**.

Образ содержит графическую подсистему, с автологином пользователя «**user»** (без пароля).

При желании можно использовать графический режим внутри виртуальной машины для тестирования. Однако рекомендуемый режим ознакомления с продуктом — по сети с машин локальной сети.

Рекомендуется настроить сетевые адаптеры в системе виртуализации в режиме «мост», чтобы виртуальная машина получила IP-адрес от DHCP-сервера локальной сети и её сервисы были доступны на машинах локальной сети.

Сценарии, описываемые в руководстве, предполагают, что виртуальная машина доступна «по имени» в локальной сети или, как минимум, на хостовой системе. То есть достаточно открыть браузер и перейти по адресу http://iskra-alt, чтобы получить стартовое окно СКРнД «Искра».

Для подключения используется простой HTTP, без SSL.

Для доступа по сети к сервисам машины, возможно потребуется прописать в настройках сопоставление имени «iskra-alt» и того IP-адреса, которое было назначено в рамках вашей локальной сети. При подключении по IP-адресу возможно будут наблюдаться сбои в работе части механизмов веб-интерфейса.

Пример установки образа виртуальной машины с предустановленной СКРнД «Искра»

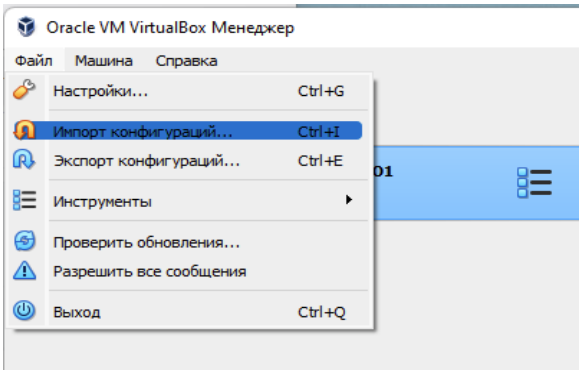
Процесс установки образа виртуальной машины зависит от используемой вами системы виртуализации и деталей её установки. **В качестве примера ниже приведен сценарий для «хостовой» ОС Windows и системы виртуализации VirtualBox 7.0.22.**

Импорт образа и запуск виртуальной машины

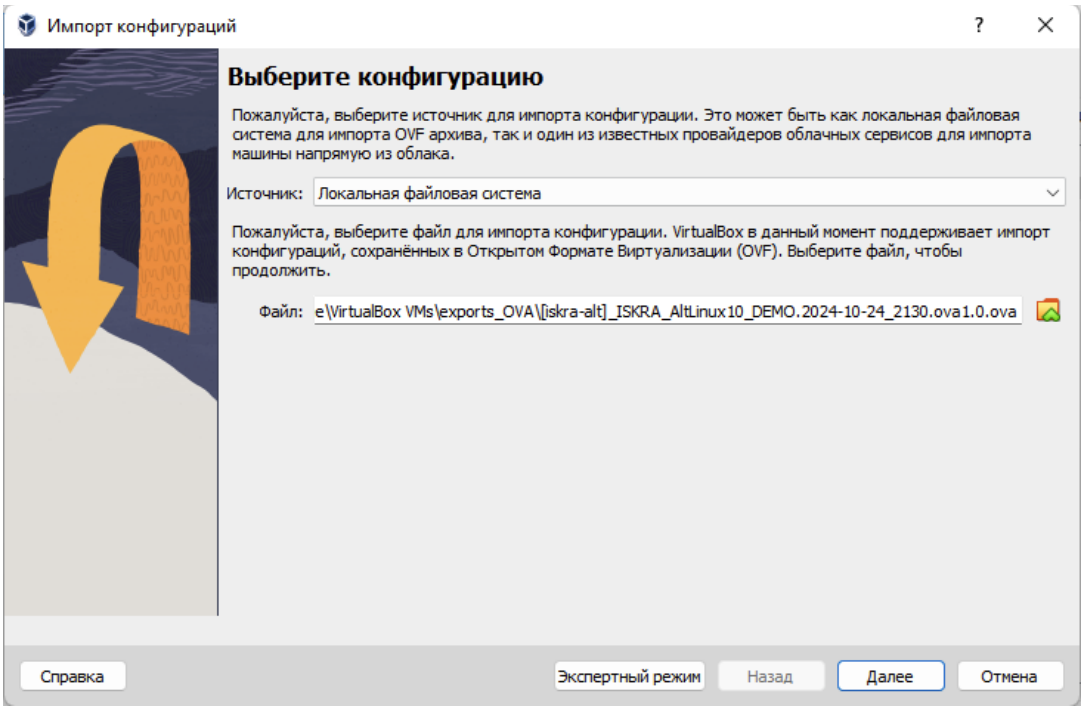
1. Убедитесь, что у вас имеются необходимые ресурсы:

- 8 ядер процессора, которые вы можете выделить для задач виртуальной машины,
- до 16 ГБ свободной памяти, которые вы можете выделить для задач виртуальной машины,
- 150 ГБ свободного места на диске, доступного для системы виртуализации.

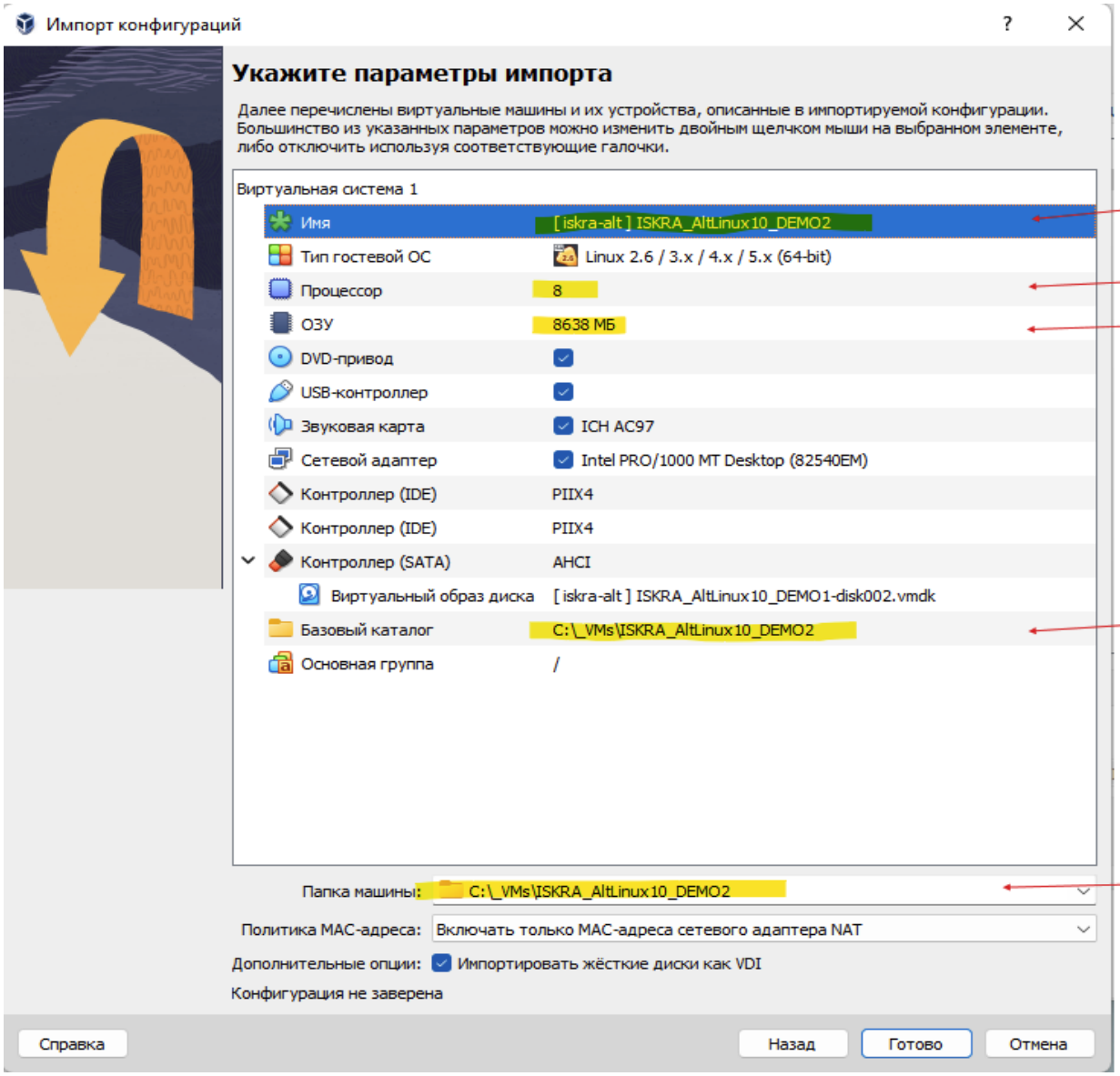
2. Откройте VirtualBox Manager, выберите «Импорт конфигураций».



3. Выберите файл образа виртуальной машины. (в нашем случае это: «**[iskra-alt]_ISKRA_AltLinux10_DEMO.2024-10-24_2130.ova1.0.ova**»)



4. **Скорректируйте параметры виртуальной машины** : отображаемое имя виртуальной машины, выделяемые ресурсы — число ядер процессора и объем памяти, и каталог для размещения машины.

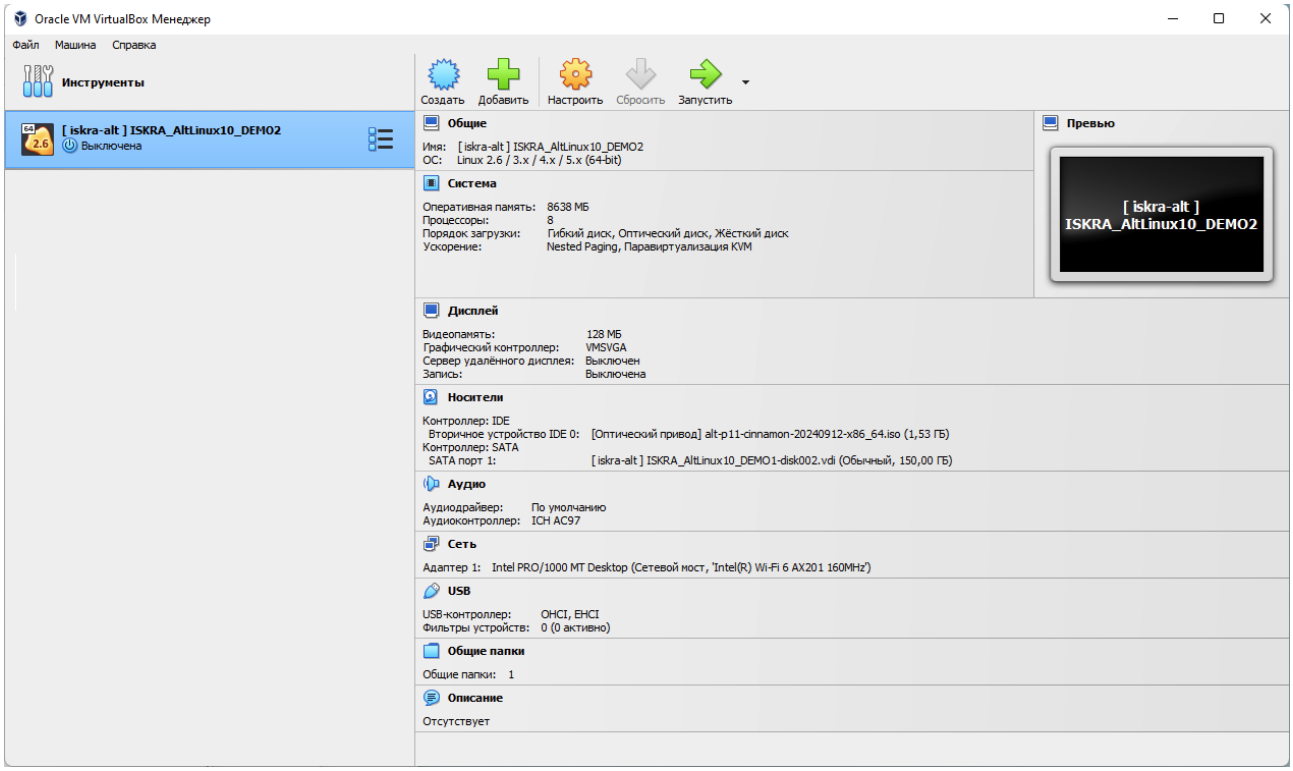


Если это единственная виртуальная машина или у вас нет предпочтений, можете оставить значения по умолчанию. Однако, для избежания замедления работы системы в виртуальной машине, не рекомендуется выделять менее 4-х ядер и 8 мегабайт памяти.

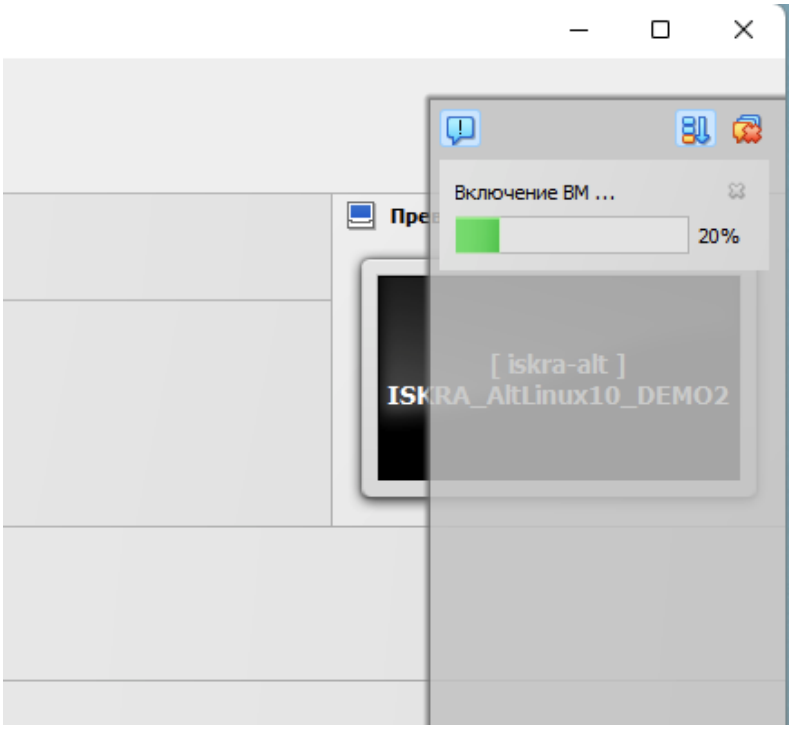
5. **Дождитесь окончания импорта образа.**

После завершения работы мастера в правой части отобразится окно статуса. Дождитесь окончания импорта, это может занять несколько минут, в зависимости от мощности вашей системы.

После окончания импорта вы увидите машину в списке машин с её характеристики

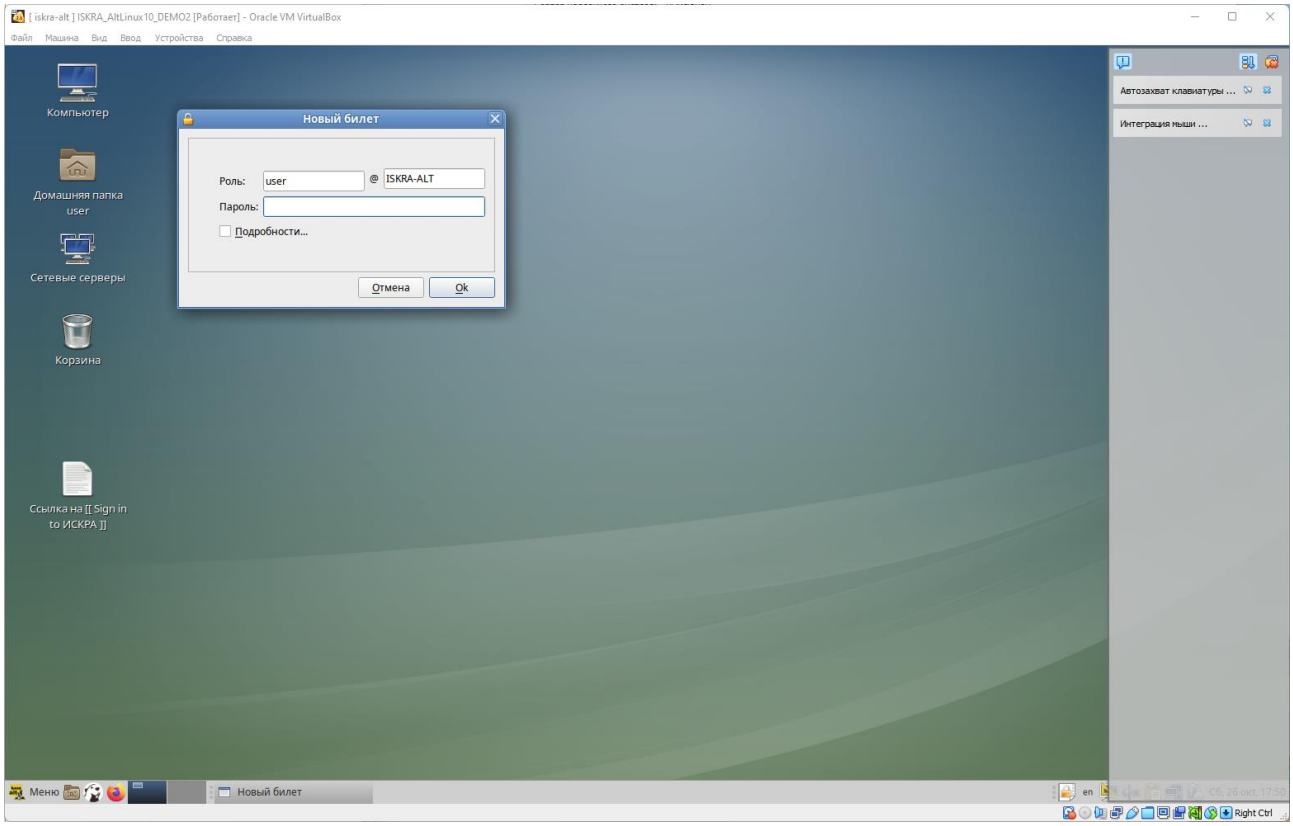


6. **Запустите виртуальную машину** кнопкой запустить.



Дождитесь окончания загрузки системы. После завершения процедур запуска вы

увидите рабочий стол ОС

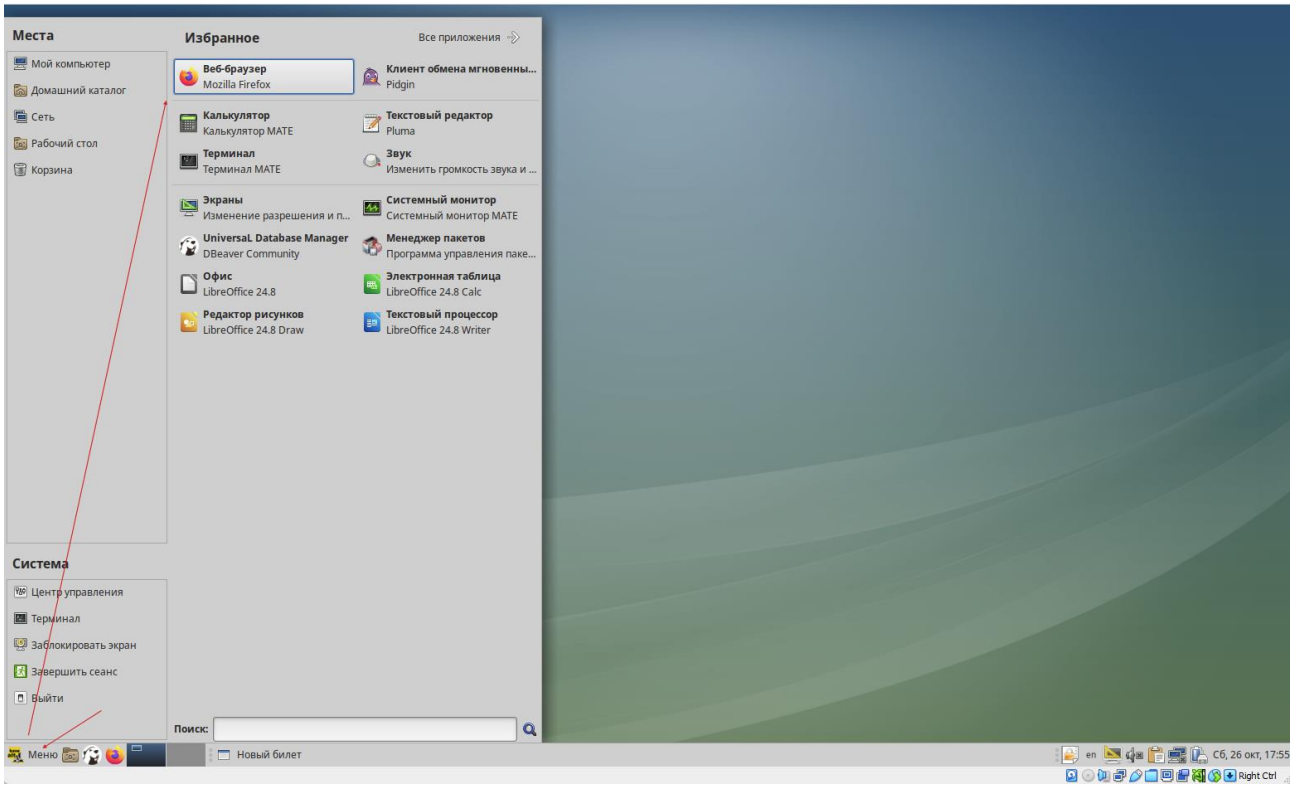


Учетные записи демонстрационных пользователей :

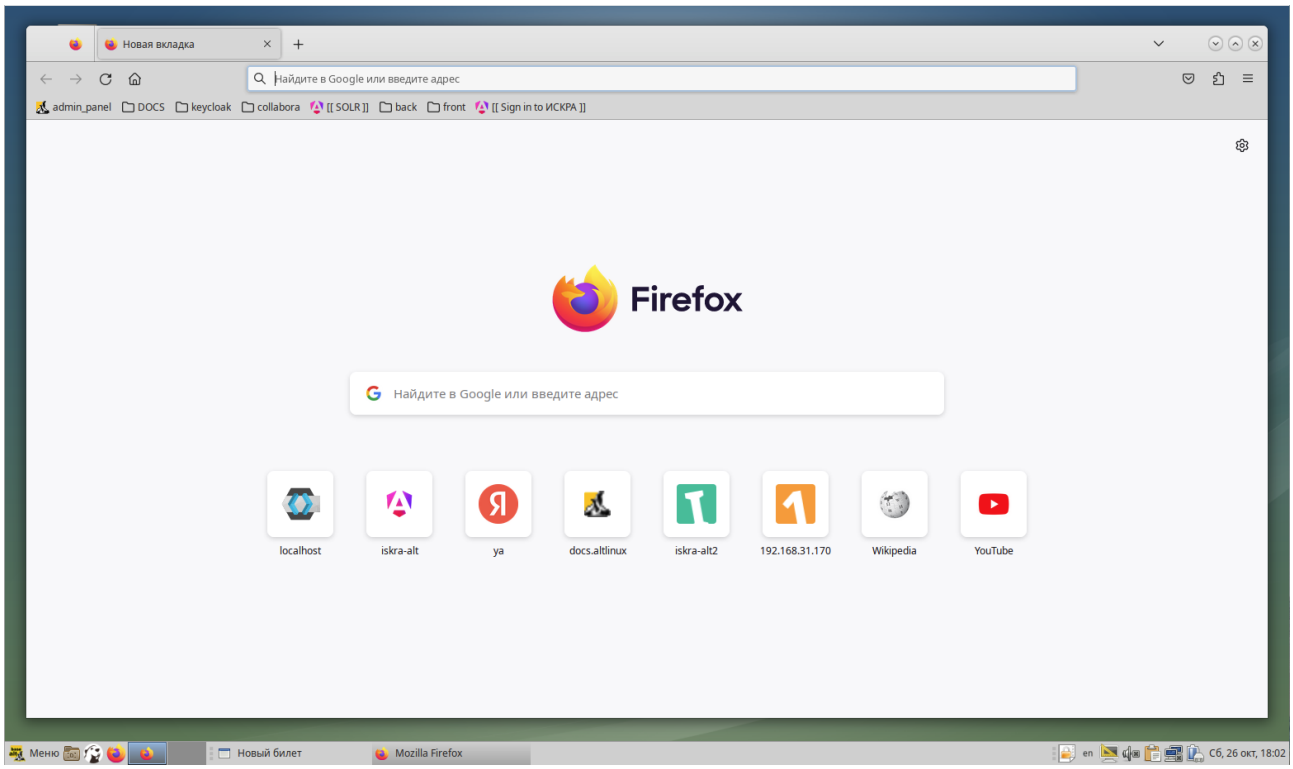
Имя пользователя и **пароль**	**Отображаемое имя**	**Глобальные роли**	**Рекомендуемые роли в проекте**
director_sergey **123qwe**	Сергей Директорович	user	**Куратор** Согласующий
pm_ivan **123qwe**	Иван Проджектович	user, pm	Куратор, **Ответственный**, Согласующий
tehpis_vladimir **123qwe**	Владимир Техписович	user	Ответственный, **Уч.раб.группы**
tehpis_konstantin **123qwe**	Константин Техписович	user	Ответственный, **Уч.раб.группы**
inspector_pavel **123qwe**	Павел0 Инспекторович	user	**Согласующий**
admin_dmitriy **123qwe**	Дмитрий Администрович	admin, user	-

Проверка доступности внутри виртуальной машины

Внутри виртуальной машины запустите браузер Firefox

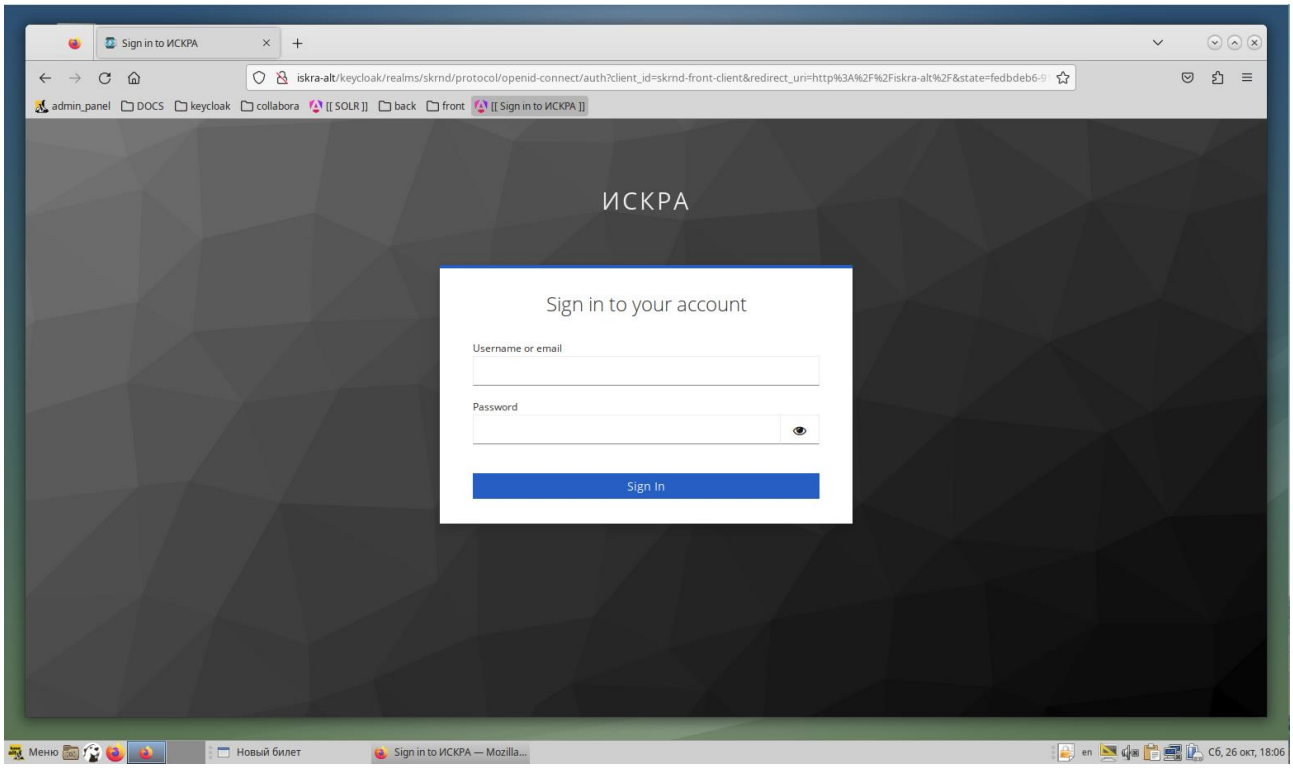


После запуска браузера введите в адресной строке http://iskra-alt или воспользуйтесь готовой ссылкой «[[Sign in to ИСКРА]]» на панели закладок браузера.

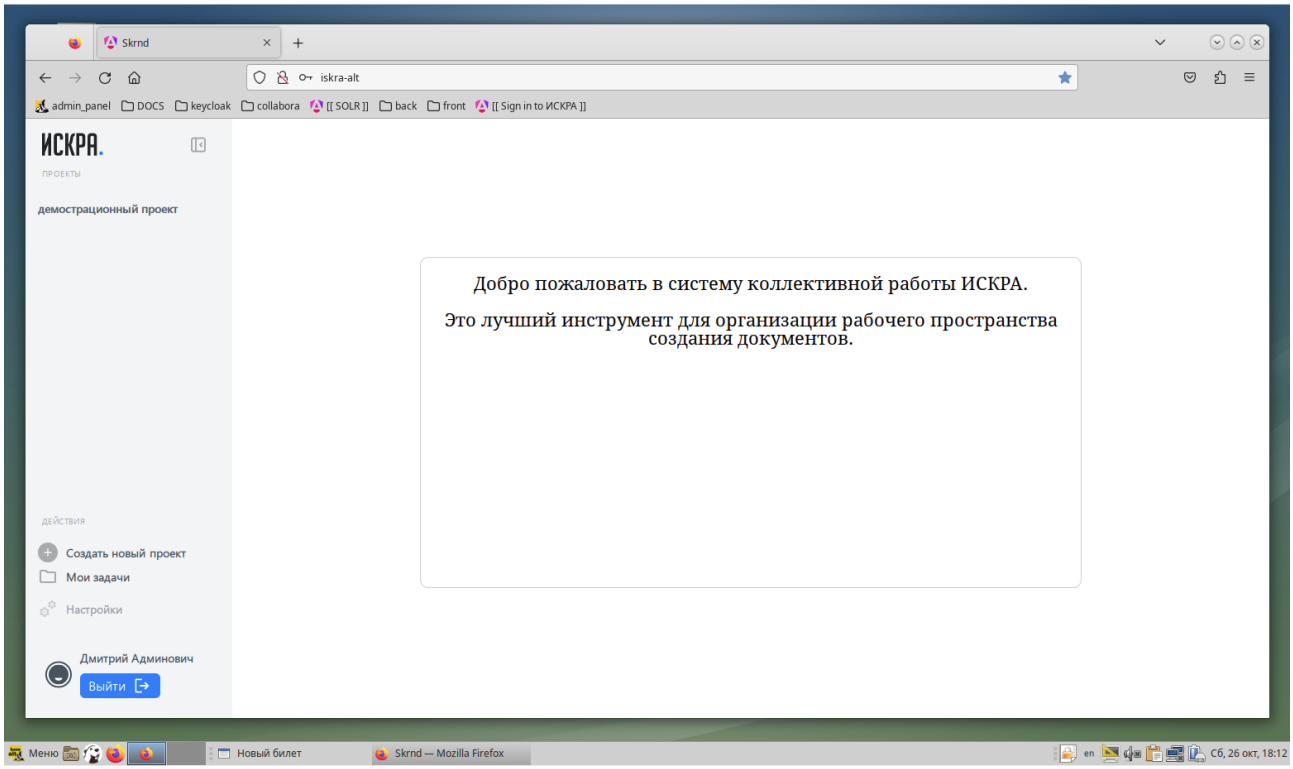


В окне логина используйте логин-пароль одного из демонстрационных пользователей

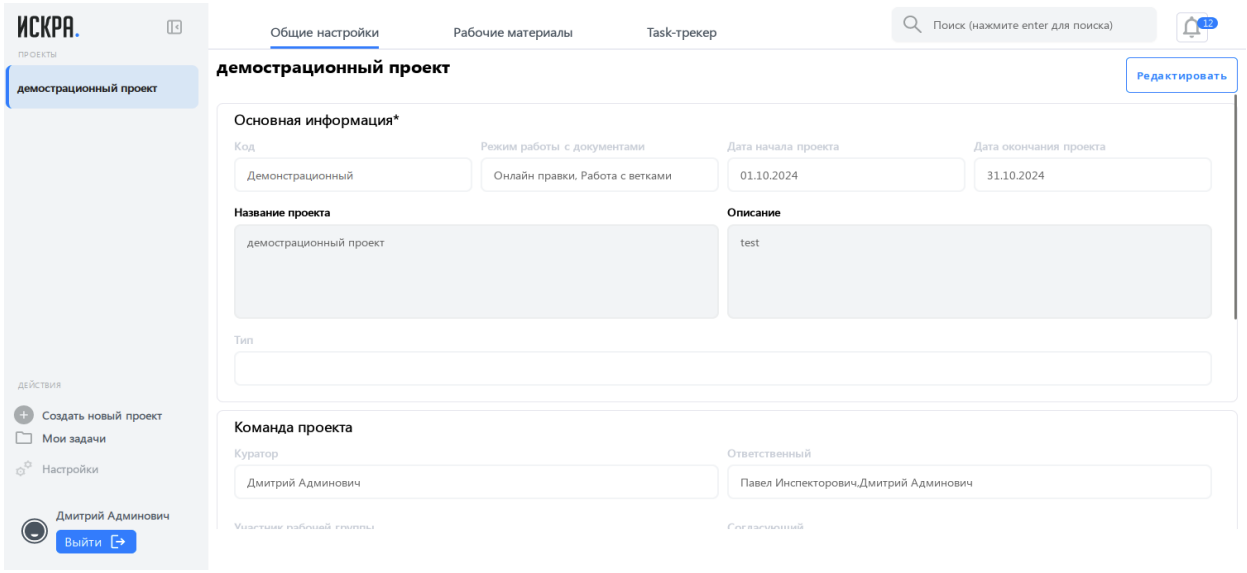
(см. раздел выше).



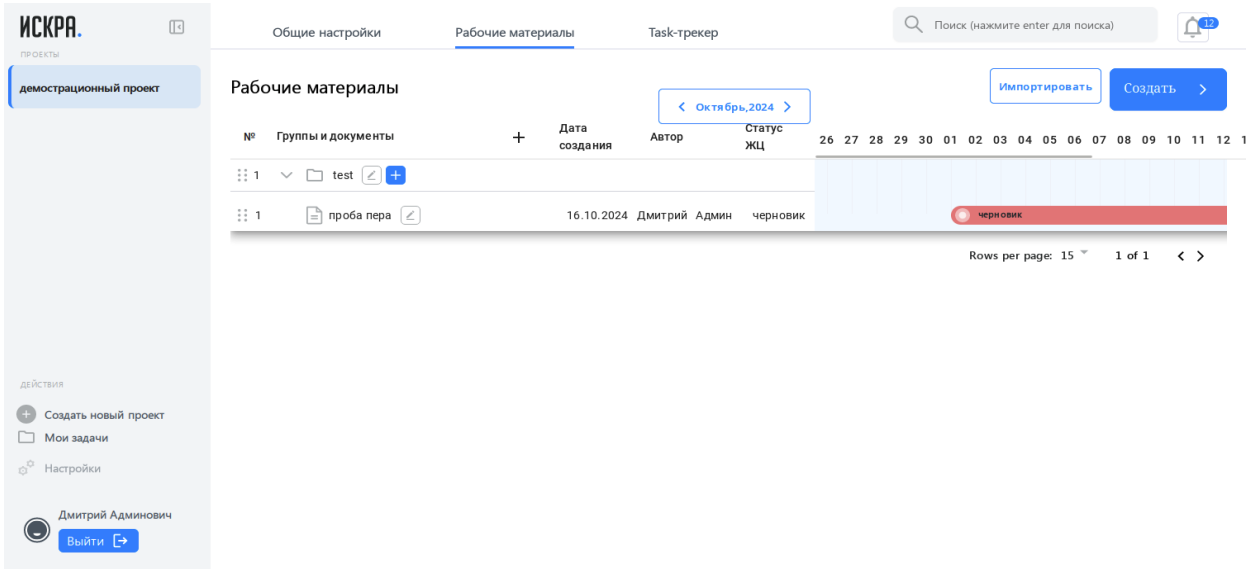
После успешной авторизации откроется стартовое окно.



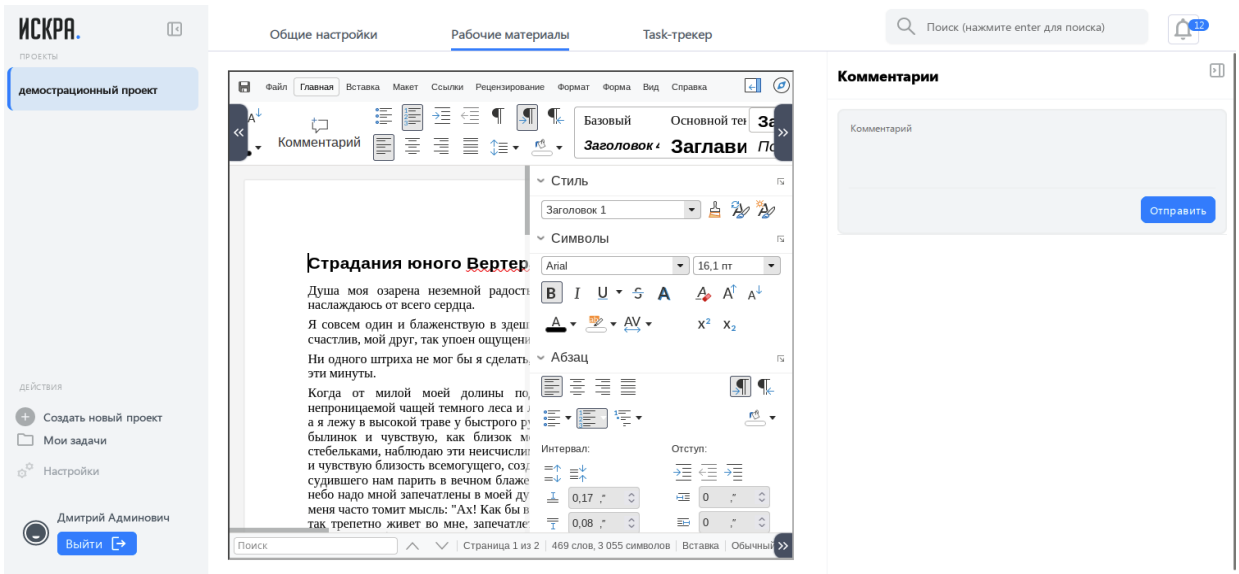
На поставляемой виртуальной машине уже заведен демонстрационный проект, выберите его нажатием.



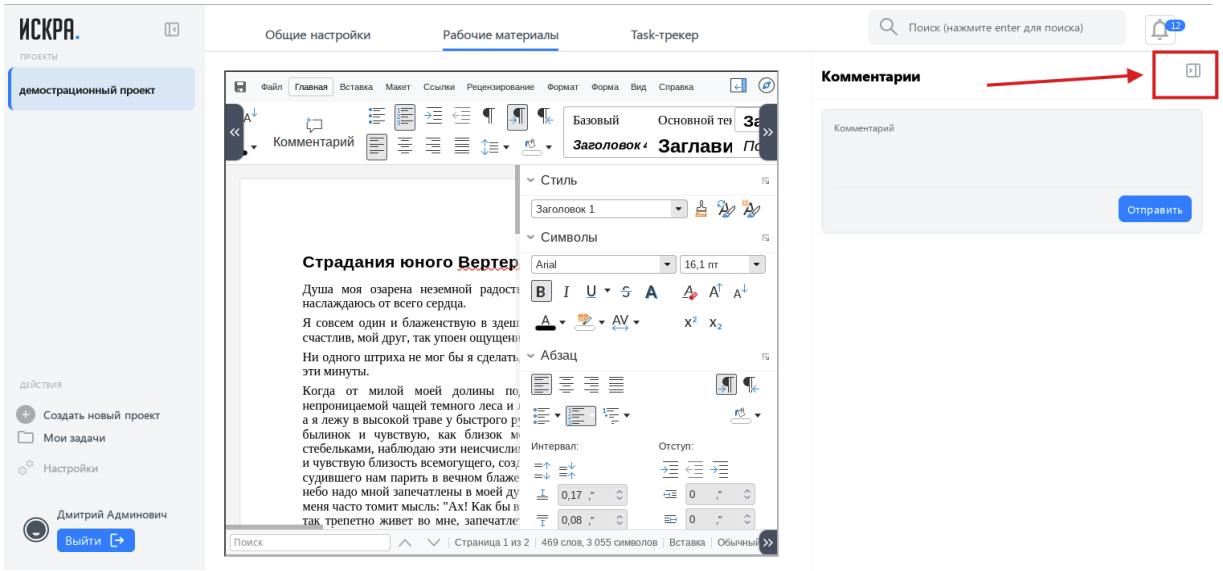
Далее перейдите в раздел «рабочие материалы».

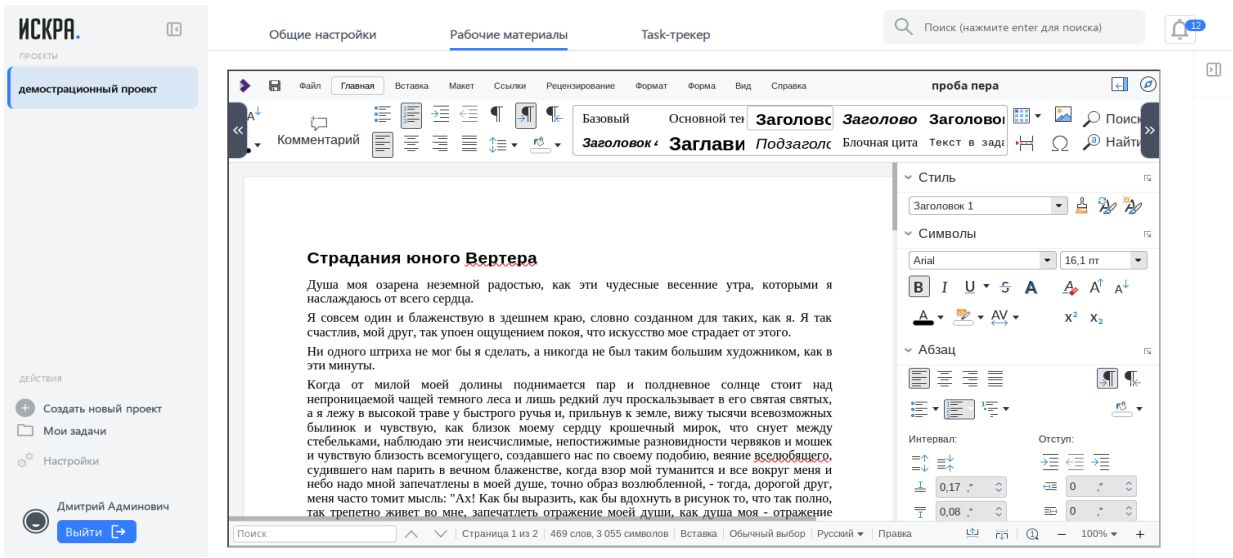


Выберите документ «проба пера».



Для удобства - отключите отображение комментариев.





Проверка доступности на хостовой системе

В Windows, на хостовой машине откройте браузер – Firefox или Chrome и введите в адресной строке http://iskra-alt (просто http, не https)

Повторите шаги в браузере аналогично тем, что описаны в предыдущем разделе (авторизуйтесь, откройте демонстрационный проект, откройте документ «проба пера»).

Установка и первичная проверка работы виртуальной машины с предустановленной СКРнД “Искра” завершены.