



**Программа для ЭВМ
Программный комплекс «Salt.Config.Box»**

Руководство администратора

Листов: 70

Дата: 10.07.2025

АННОТАЦИЯ

В настоящем документе содержится руководство системного администратора по установке и настройке программы для ЭВМ «Программный комплекс «Salt.Config.Box»» (далее – «Программный комплекс»).

В разделе «Введение» указаны область применения и основные возможности Программного комплекса.

В разделе «Установка и настройка Программного комплекса» приведены требования к характеристикам оборудования, на которое производится установка Программного комплекса, порядок установки и первичной настройки Программного комплекса.

В разделе «Процедуры по обслуживанию Программного комплекса» приведено описание ежедневных процедур и процедур, выполняемых при необходимости, таких как запуск, останов Программного комплекса, резервное копирование данных и мониторинг состояния.

В разделе «Проверка работоспособности компонентов Программного комплекса» приведено описание способов проверки, позволяющих дать общее заключение о работоспособности Программного комплекса.

Структура и оформление настоящего документа соответствует ГОСТ 19.503-79.

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	2
Дата созд.	10.07.2025		Страниц	70

СОДЕРЖАНИЕ

1	Введение	5
1.1	Область применения Программного комплекса	5
1.2	Возможности программного комплекса	5
2	Установка и настройка Программного комплекса	6
2.1	Требования к программным и техническим средствам	6
2.1.1	Требования к аппаратному обеспечению сервера Salt.Box.....	6
2.1.2	Требования к аппаратному обеспечению сервера Salt Master, подключаемого к Salt.Box	7
2.2	Установка Docker	7
2.2.1	Поддерживаемые платформы.....	7
2.2.2	Прочие дистрибутивы Linux.....	8
2.3	Развёртывание сервера Salt.Config.Box.....	8
2.3.1	Загрузка проекта и подготовка к установке.....	8
2.3.2	Настройка обратного прокси сервера Nginx с SSL-терминацией.....	11
2.3.3	Установка серверных компонентов Salt.Box с помощью скрипта.....	13
2.3.4	Пошаговая установка серверных компонентов Salt.Box	14
2.3.5	Объекты Docker, создаваемые при установке.....	14
2.3.6	Вход в веб-интерфейс Salt.Box	18
2.3.7	Очистка	18
2.3.7.1	Очистка устаревших данных Docker	18
2.4	Настройка KeyCloak.....	20
2.4.1	Вход в KeyCloak.....	20
2.4.2	Создание нового реалма	21
2.4.3	Создание клиента.....	22
2.4.3.1	Создание роли клиента	26
2.4.4	Создание пользователя	28
2.4.4.1	Назначение роли пользователю	32
2.5	Установка клиента Salt.Config.Box.....	34
2.5.1	Установка в ОС Linux	34
2.5.1.1	Установка пакета RPM	34
2.5.1.2	Установка пакета DEB	37
2.5.1.3	Установка на Альт Линукс	40
2.5.1.4	Установка на Astra Linux SE	40
2.5.1.5	Настройка.....	40
2.5.2	Установка в ОС Windows	41
2.5.2.1	Интерактивная установка	41
2.5.2.2	Установка с помощью групповых политик Microsoft Active Directory	44

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	3
Дата созд.	10.07.2025		Страниц	70

3	Процедуры по обслуживанию Программного комплекса	50
3.1	Ежедневные процедуры	52
3.1.1	Мониторинг контейнеров Docker	52
3.1.2	Синтаксис команды docker stats	52
3.2	Процедуры, проводимые при необходимости.....	54
3.2.1	Запуск и остановка контейнеров	54
3.2.2	Получение информации о контейнерах	54
3.2.3	Очистка Docker	55
3.3	Просмотр логов сервисов Программного комплекса	57
3.3.1	Синтаксис команды docker logs	57
3.4	Взаимодействие с системой	59
3.4.1	Резервное копирование томов Docker	59
3.4.1.1	Создание резервных копий томов	59
3.4.1.2	Восстановление томов из резервных копий.....	59
3.4.2	Резервное копирование базы данных KeyCloak.....	59
3.4.2.1	Создание резервной копии базы данных PostgreSQL	60
3.4.2.2	Восстановление базы данных PostgreSQL из резервной копии	63
3.4.3	Резервное копирование данных MongoDB	63
3.4.3.1	Создание резервной копии базы данных MongoDB.....	63
3.4.3.2	Восстановление базы данных MongoDB из резервной копии	64
4	Проверка работоспособности компонентов Программного комплекса.....	65
4.1	Проверка работоспособности сервера.....	65
4.2	Проверка работоспособности клиента в ОС Windows.....	65
4.3	Проверка работоспособности клиента в ОС Linux.....	66
	Приложение А. Примеры вывода команд Docker	67
A.1	Список образов	67
A.2	Список томов.....	67
A.3	Список работающих compose-контейнеров	67
A.4	Список работающих контейнеров	68
	Перечень сокращений и условных обозначений.....	70

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	4
Дата созд.	10.07.2025		Страниц	70

1 Введение

1.1 Область применения Программного комплекса

Программный комплекс «Salt.Config.Box» (далее также — «Программный комплекс») предназначен для управления программными конфигурациями рабочих станций с целью обеспечения их функционирования в требуемом согласованном состоянии, развёртывания системного и прикладного программного обеспечения на рабочих станциях, мониторинга состояния рабочих станций.

Автоматизируемые виды деятельности:

- инвентаризация аппаратного и программного обеспечения автоматизированных рабочих мест Пользователей;
- управление программными конфигурациями рабочих станций в гибридной информационно-технологической инфраструктуре;
- управление групповыми политиками;
- перевод рабочих станций на использование ОС семейства GNU/Linux и Linux-совместимого прикладного программного обеспечения;
- конфигурирование прикладного программного обеспечения.

1.2 Возможности программного комплекса

- Многопользовательский режим работы с управлением доступом на основе ролей.
- Получение подробной информации о парке вычислительной техники предприятия, статистической информации об аппаратном и программном обеспечении рабочих станций.
- Развитые средства фильтрации и группировки рабочих станций на основе их аппаратных и программных характеристик, состояния программной конфигурации, различных отслеживаемых параметров.
- Асинхронное параллельное выполнение задач управления программными конфигурациями на отдельных физических или виртуальных машинах или на группах машин.
- Единый веб-интерфейс управления рабочими станциями, задачами и шаблонами задач, сценариями управления и отчётами.

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	5
Дата созд.	10.07.2025		Страниц	70

2 Установка и настройка Программного комплекса

2.1 Требования к программным и техническим средствам

Программный комплекс «Salt.Config.Box» распространяется в виде набора образов для контейнеризатора приложений Docker и файлов скриптов развёртывания многоконтейнерных приложений (Docker Compose).

Установка Программного комплекса «Salt.Config.Box» может производиться на виртуальный или физический сервер с установленными на нем компонентами Docker.

Серверу Salt.Box требуется один высокоскоростной сетевой интерфейс. Должен быть предоставлен один IP-адрес, корректно настроенный DNS-сервер и шлюз с доступом к сети Интернет.

Для Клиентских устройств должны быть доступны следующие TCP-порты сервера: **1022 (SSH), 4505, 4506**.

Пользователю Программного комплекса должны быть доступны TCP-порты **80 (HTTP), 443 (HTTPS)** сервера.

Трансляция сетевых адресов (NAT) между сервером и Клиентскими устройствами должна отсутствовать для обеспечения работоспособности функции отслеживания онлайн-статуса Клиентских устройств.

Сервер Salt.Box активно взаимодействует с управляемыми устройствами с использованием нескольких сетевых протоколов и должен быть добавлен в разрешающий список системы предотвращения вторжений.

В случае необходимости создания резервных копий и образов дисков (в некоторых сценариях перевода Клиентских устройств на использование отечественного ПО) необходимо предусмотреть дополнительный ресурс для подключаемой файловой системы SSHFS.

Для выполнения одновременного перевода Клиентских устройств на использование отечественного ПО с созданием виртуальной машины рекомендуется обеспечить ресурс файлового хранилища не менее 1 Тбайт для каждых десяти Клиентских устройств, планируемых к переводу.

2.1.1 Требования к аппаратному обеспечению сервера Salt.Box

В таблице 1 указаны минимальные и рекомендованные требования к аппаратному обеспечению сервера для установки всех контейнеров Программного комплекса, за исключением **saltbox-salt-master**.

В таблице 2 указаны рекомендованные требования к аппаратному обеспечению сервера для работы одного контейнера **saltbox-salt-master**.

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	6
Дата созд.	10.07.2025		Страниц	70

Таблица 1 – Требования к аппаратному обеспечению сервера Salt.Box

№	Параметр	Минимальное значение	Рекомендуемое значение
1.	Архитектура процессора	x86_64	x86_64
2.	Количество ядер процессора, шт.	4	8
3.	Объем оперативной памяти, ГБ	4	16
4.	Объем дискового пространства для размещения объектов Docker, ГБ	10	50
5.	Пропускная способность сетевого интерфейса, Мбит/с	100	1000

2.1.2 Требования к аппаратному обеспечению сервера Salt Master, подключаемого к Salt.Box

Таблица 2 – Требования к аппаратному обеспечению сервера для одного контейнера saltbox-salt-master

№	Параметр	до 10000 клиентов	10000 и более клиентов
1.	Архитектура процессора	x86_64	x86_64
2.	Количество ядер процессора, шт.	2	4
3.	Объем оперативной памяти, ГБ	4	4
4.	Пропускная способность сетевого интерфейса, Мбит/с	1000 (рекомендуется)	1000

2.2 Установка Docker

Текущая минимальная совместимая версия **Docker Compose: 2.22.0.**

Текущая минимальная совместимая версия **Docker Engine: 25.**

Ниже приведены ссылки на онлайн-документацию проекта Docker, описывающую, как установить необходимые компоненты Docker на различные операционные системы.

2.2.1 Поддерживаемые платформы

Ниже приведены ссылки на Порядок установки на онлайн-руководства по установке Docker на основных дистрибутивах GNU/Linux.

CentOS <https://docs.docker.com/engine/install/centos/>

Debian <https://docs.docker.com/engine/install/debian/>

Fedora <https://docs.docker.com/engine/install/fedora/>

Ubuntu <https://docs.docker.com/engine/install/ubuntu/>

2.2.2 Прочие дистрибутивы Linux

К СВЕДЕНИЮ

Проект Docker не тестирует и не проверяет установку на производных дистрибутивах.

- Если вы используете производные Debian, такие как LMDE (Mint на основе Debian), следуйте инструкциям по установке в соответствующем выпуске ОС Debian: <https://docs.docker.com/engine/install/debian/>.

Чтобы узнать, какой выпуск Debian соответствует версии вашего производного дистрибутива, обратитесь к документации вашего дистрибутива.

- Аналогично, если вы используете производные Ubuntu, такие как Kubuntu, Lubuntu или Xubuntu, следуйте инструкциям по установке в соответствующем выпуске ОС Ubuntu: <https://docs.docker.com/engine/install/ubuntu/>.

Чтобы узнать, какой выпуск Debian соответствует версии вашего производного дистрибутива, обратитесь к документации вашего дистрибутива.

- Некоторые дистрибутивы Linux предоставляют пакет Docker Engine через свои репозитории пакетов.

Эти пакеты собираются и поддерживаются сопровождающими пакетов дистрибутива Linux и могут иметь различия в конфигурации или собираются из измененного исходного кода.

Docker предоставляет двоичные файлы для ручной установки Docker Engine.

Эти двоичные файлы статически связаны, и вы можете использовать их в любом дистрибутиве Linux.

2.3 Развёртывание сервера Salt.Config.Box

2.3.1 Загрузка проекта и подготовка к установке

ПРЕДУПРЕЖДЕНИЕ

Если Вы выполняете развёртывание сервера **Salt.Box** на виртуальной машине, убедитесь, что процессор виртуальной машины поддерживает расширения набора команд **avx** и **avx2**. Данные расширения требуются для работы **MongoDB** версии **5.0** и выше.

Проверить наличие поддержки можно следующей командой:

```
cat /proc/cpuinfo
```

В выводе команды в блоке **flags** должны быть указаны флаги **avx** и **avx2**.

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	8
Дата созд.	10.07.2025		Страниц	70

1. Загрузите и разархивируйте исходный код актуального релиза Программного комплекса по ссылке:

<https://dev.saltbox.pro/saltbox/saltbox-compose/-/releases>

2. Войдите в каталог проекта.
3. В корневом каталоге проекта создайте копию файла **example.env** с именем **.env**.
Настройте его перед запуском **docker compose**.

Далее приведены описания некоторых переменных файла .env, переопределение которых может потребоваться пользователю для корректной работы Salt.Box в его сетевой среде.

- а. Если Вы планируете развёртывание Программного комплекса «Salt.Config.Box» на виртуальной машине или на хосте, отличном от **localhost**, то в файле **.env** должны быть переопределены переменные, значением которых по умолчанию является **'localhost'**:

Порт хост-системы, которому будет сопоставлен порт контейнера Docker, обслуживающего веб-интерфейс Salt.Box, например:

```
WEB_SERVER_PORT=80
```

Переменной **WEB_SERVER_OUTER_SOCKET** должен быть присвоен сетевой адрес или сетевое имя хоста, на котором веб-интерфейс Salt.Box будет доступен пользователю. Опционально может быть указано также номер порта:

```
WEB_SERVER_OUTER_SOCKET=<HOST> | <HOST>:<PORT>
```

К СВЕДЕНИЮ

Если развёртывание **Salt.Box** производится на виртуальной машине, переменной **WEB_SERVER_OUTER_SOCKET** должен быть присвоен сетевой адрес её интерфейса в хостовой системе.

Если доступ к Salt.Box будет осуществляться без использования обратного прокси-сервера, то в **WEB_SERVER_OUTER_SOCKET** должен быть указан тот же порт, что и в **WEB_SERVER_PORT**. Если порт не указан явно, будет использовано значение по умолчанию для протокола (80/TCP для HTTP, 443/TCP для HTTPS).

- б. Сокет сервера Redis (для подключения к нему мастера).

Для подключения внешних мастеров следует указать **0.0.0.0:6379** (все сетевые интерфейсы) либо указать адрес одного из сетевых интерфейсов:

```
REDIS_SALT_EXPOSE_SOCKET=<IP_ADDRESS>:6379
```

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	9
Дата созд.	10.07.2025		Страниц	70

- в. Переключение веб-интерфейса Salt.Box на использование протокола SSL.

Протокол SSL накладывается внешним веб-сервером (обратным прокси-сервером).

```
WEB_SERVER_SCHEME=https
WEB_SERVER_WS_SCHEME=wss
```

Настройте обратный прокси-сервер, как указано в разделе 2.3.2 «Настройка обратного прокси сервера Nginx с SSL-терминацией».

- г. Теги образов, с которых осуществляется развёртывание Программного комплекса «Salt.Config.Box»:

```
CORE_AUTOTESTS_TAG=<vX.Y.Z>
CORE_IMAGE_TAG=<vX.Y.Z>
FRONTEND_IMAGE_TAG=<vX.Y.Z>
KEYCLOAK_IMAGE_TAG=<vX.Y.Z>
NGINX_IMAGE_TAG=<vX.Y.Z>
OPA_IMAGE_TAG=<vX.Y.Z>
REDIS_IMAGE_TAG=<vX.Y.Z>
SALT_MASTER_IMAGE_TAG=<vX.Y.Z>
SSHFS_IMAGE_TAG=<vX.Y.Z>
```

где <vX.Y.Z> — номер версии программного комплекса «Salt.Config.Box»

- д. Время ожидания остановки контейнера до принудительной остановки:

```
STOP_GRACE_PERIOD=5s
```

Уменьшение данного значения может ускорить остановку, но повышает риски потери врёменных данных.

- е. Уровень журналирования контейнера главного мастера SaltStack:

```
SALT_MASTER_LOG_LEVEL='warning'
SALT_MINION_LOG_LEVEL='warning'
```

Обозначения уровней журналирования:

Уровень	Числовое значение	Описание
critical	50	Критические ошибки
error	40	Ошибки
warning	30	Предупреждения
info	20	Важная информация, предоставляемая приложением при нормальном выполнении
debug	10	Информация для диагностики проблем и устранения неполадок
all	0	Все события

ПРЕДУПРЕЖДЕНИЕ

Любой уровень журнала ниже уровня "info" **НЕБЕЗОПАСЕН** и может регистрировать конфиденциальные данные.

Подробную информацию об уровнях журналирования смотрите в документации проекта SaltStack:

<https://docs.saltproject.io/en/3006/ref/configuration/logging/index.html>.

- ж. Пути к каталогам в файловой системе хоста Docker, в которых будут храниться конфигурации для клиентов.

Каталоги могут быть перенесены, например, на примонтированное хранилище:

```
SSHFS_STORAGE_PATH='./_sshfs-storage/'
```

SSHFS-хранилище предназначено для хранения шаблонов и некоторых других типов файлов, используемых при работе Salt.Box.

2.3.2 Настройка обратного прокси сервера Nginx с SSL-терминацией

Для корректной работы веб-интерфейса на удаленных хостах HTTP-соединение должно иметь SSL-терминацию.

Для этого можно использовать веб-сервер **Nginx** в качестве обратного HTTP-прокси.

Во-первых, обязательно установите `WEB_SERVER_OUTER_SOCKET` в соответствии с `server_name` и портом обратного прокси.

Nginx может быть установлен на том же хосте, что и **Salt.Box**, либо на отдельном хосте.

В последнем случае убедитесь, что **Salt.Box** доступен для хоста Nginx, например, с помощью команды:

```
curl http://<SALTBOX_HOST>:<SALTBOX_WEB_SERVER_PORT>/auth/keycloak/realms/salt.box/.well-known/openid-configuration
```

Команда должна вернуть длинный ответ в формате JSON.

Ниже описан порядок установки и конфигурирования обратного прокси-сервера.

1. Внесите изменения в файл `.env`.

```
WEB_SERVER_PORT=<PORT> # незанятый порт, например, 8090
```

```
WEB_SERVER_SCHEME='https'
```

```
WEB_SERVER_WS_SCHEME='wss'
```

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	11
Дата созд.	10.07.2025		Страниц	70

2. Установите веб-сервер **Nginx**.

Для Debian/Ubuntu:

```
sudo apt update
```

```
sudo apt install nginx -y
```

3. Сгенерируйте сертификат.

Создайте каталог:

```
sudo mkdir -p /etc/nginx/ssl
```

и сгенерируйте сертификат (показан пример значения параметра **-subj**, не забудьте указать нужные значения в заполнителях **< ... >**):

```
sudo openssl req -x509 -nodes -days 365 \
```

```
-newkey rsa:2048 \
```

```
-keyout /etc/nginx/ssl/saltbox-selfsigned.key \
```

```
-out /etc/nginx/ssl/saltbox-selfsigned.crt \
```

```
-subj "/C=RU/ST=Saltbox/L=Local/O=Saltbox Dev/CN=<IP-адрес, по которому будет доступен Salt.Box>"
```

4. Создайте конфигурационный файл для веб-интерфейса Salt.Box.

```
sudo touch /etc/nginx/sites-available/saltbox.conf
```

Содержание конфигурационного файла:

```
server {
    listen 80;

    server_name <IP-адрес, по которому будет осуществляться вход в Salt.Box (указан в файле
.env в переменной WEB_SERVER_PORT)>;

    return 301 https://$host$request_uri;
}

server {
    listen 443 ssl http2;
    server_name <IP-адрес, по которому будет осуществляться вход в Salt.Box (указан в файле
.env в переменной WEB_SERVER_PORT)>;

    ssl_certificate /etc/nginx/ssl/saltbox-selfsigned.crt;
    ssl_certificate_key /etc/nginx/ssl/saltbox-selfsigned.key;

    client_max_body_size 256m;

    location / {
        proxy_pass http://localhost:<PORT>; # Порт, который указан в файле .env в переменной
WEB_SERVER_PORT
```

```
proxy_set_header Host $host;
proxy_set_header X-Real-IP $remote_addr;
proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
proxy_set_header X-Forwarded-Proto https;
proxy_set_header X-Forwarded-Port 443;
proxy_set_header Upgrade $http_upgrade;
proxy_set_header Connection "upgrade";
}
}
```

Не забудьте указать нужные значения в заполнителях < ... >.

5. Активируйте конфигурацию

Для Debian/Ubuntu:

```
sudo ln -s /etc/nginx/sites-available/saltbox.conf /etc/nginx/sites-enabled/
```

6. Проверьте корректность конфигурации и перезагрузите **nginx**.

```
sudo nginx -t
```

```
sudo systemctl reload nginx
```

2.3.3 Установка серверных компонентов Salt.Box с помощью скрипта

Самый простой способ запустить систему — выполнить скрипт автоматической установки:

```
sudo ./bin/update_and_run.sh
```

Скрипт автоматической установки:

1. выполняет слияние файлов **example.env** и **override.env** (если файл **override.env** существует) в файл конфигурации **.env**
2. создаёт набор файлов с секретами
3. выполняет обновление образов
4. запускает экземпляр Salt.Box

Используйте файл **override.env** для переопределения значений по умолчанию из файла **example.env**.

Используйте флаг **-h** или **--help** для просмотра опций скрипта автоматической установки.

Альтернативным способом развёртывания сервера Salt.Box является пошаговая установка (см. раздел 2.3.4).

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	13
Дата созд.	10.07.2025		Страниц	70

2.3.4 Пошаговая установка серверных компонентов Salt.Box

1. В подкаталоге проекта `./secrets/` должны быть созданы файлы с секретами.

Создать файлы можно с помощью вспомогательного скрипта:

```
python3 ./bin/make_secrets.py
```

2. Находясь в каталоге проекта, запустите сборку:

```
sudo docker compose up --build -d
```

ПРЕДУПРЕЖДЕНИЕ

Проверьте, нет ли предупреждений о неустановленных переменных, чтобы избежать ошибок сборки.

3. Для обновления внешних образов выполните команду:

```
sudo docker compose pull
```

затем перезапустите сборку командой:

```
sudo docker compose down && sudo docker compose up -d
```

Если указан флаг `--pull=always`, перед сборкой будут загружены актуальные образы:

```
sudo docker compose up --build --pull=always
```

Для развёртывания рабочей версии продукта рекомендуется установить режим обработки перерасхода памяти с помощью команд:

```
sudo sh -c "echo 'vm.overcommit_memory=1' > /etc/sysctl.d/salt-box.conf"
```

```
sudo sysctl -p /etc/sysctl.d/salt-box.conf
```

Значение `vm.overcommit_memory=1` является требованием Redis.

2.3.5 Объекты Docker, создаваемые при установке

После успешного завершения процесса сборки создаётся ряд объектов Docker, приведённых в таблицах 3, 4, 5.

Таблица 3 – Загружаемые образы Docker

Образ Docker	Размер	Описание
saltbox-proxy	208 Мбайт	
saltbox-make-redis-salt-certs	109 Мбайт	
registry.saltbox.pro/saltbox/saltbox-compose/keycloak-init	820 Мбайт	
registry.saltbox.pro/saltbox/saltbox-compose/keycloak	808 Мбайт	

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	14
Дата созд.	10.07.2025		Страниц	70

Образ Docker	Размер	Описание
registry.saltbox.pro/saltbox/saltbox-compose/sshfs	357 Мбайт	Подключаемая файловая система SSHFS
registry.saltbox.pro/saltbox/saltbox-compose/opa	84 Мбайт	Open Policy Agent
registry.saltbox.pro/saltbox/saltbox-compose/redis	61 Мбайт	Redis server
registry.saltbox.pro/saltbox/saltbox-core	488 Мбайт	Серверные компоненты (ядро) Salt.Box
registry.saltbox.pro/saltbox/saltbox-bridge/salt-master	336 Мбайт	Сервер SaltStack (Salt Master)
registry.saltbox.pro/saltbox/saltbox-frontend	215 Мбайт	Пользовательский интерфейс Salt.Box
postgres	394 Мбайт	СУБД PostgreSQL
mongo	1.1 Гбайт	СУБД MongoDB
rabbitmq	389 Мбайт	Брокер сообщений RabbitMQ

Таблица 4 – Контейнеры Docker, создаваемые при установке

Контейнер Docker	Статус	Описание
saltbox-mongo-1	Up	СУБД MongoDB. Используется бэкендом Salt.Box. В БД хранятся такие сущности, как клиенты, коллекции и т. п.
saltbox-sshfs-1	Up	Сервер SSH. Используется для монтирования файловой системы sshfs на миньонах
saltbox-opa-1	Up	Сервис Open Policy Agent. ОПА используется для создания политик доступа к различным компонентам веб-приложения
saltbox-keycloak-db-1	Up	Сервис KeyCloak. Обеспечивает централизованную аутентификацию в веб-интерфейсе Salt.Box
saltbox-saltbox-frontend-1	Up	Пользовательский интерфейс Salt.Box (фронтенд)
saltbox-make-redis-salt-certs-1	Exited	Модуль, предоставляющий сервис выпуска сертификатов для аутентификации на основе сертификатов Redis
saltbox-saltbox-core-tasks-watcher-1	Up	Модуль, ожидающий публикации задач в шине обмена SaltStack
saltbox-saltbox-core-taskiq-scheduler-1	Up	Модуль, отвечающий за работу с распределённой асинхронной очередью задач
saltbox-saltbox-core-taskiq-worker-1	Up	Процесс taskiq worker. Отвечает за получение, отправку, выполнение задач из очереди
saltbox-rabbitmq-1	Up	Сервис RabbitMQ — брокер сообщений. Отвечает за сбор данных из нескольких источников и отправляет их в сервисы для дальнейшей обработки
saltbox-redis-taskiq-1	Up	Сервер Redis, используемый как брокер и хранилище результатов для Taskiq

Контейнер Docker	Статус	Описание
saltbox-saltbox-core-faststream-redis-1	Up	Сервер Redis, используемый как брокер для Faststream
saltbox-redis-salt-1	Up	Сервер Redis. Используется для хранения результатов выполнения процессов, связанных с SaltStack
saltbox-keycloak-1	Up	Подсистема идентификации и управления доступом Salt.Box
saltbox-saltbox-core-1	Up	Главный серверный компонент Salt.Box (бэкенд)
saltbox-salt-master-1	Up	Сервис salt-master для управления миньонами
saltbox-keycloak-init-1	Exited	
saltbox-proxy-1	Up	Nginx reverse proxy. Перенаправляет запросы клиента на внутренние сервисы в контейнерах

В таблице 4 указаны статусы контейнеров в нормальном режиме работы Salt.Box:

Up — Контейнер запущен.

Exited — Контейнер запущен, затем нормально остановлен до запуска по требованию.

Таблица 5 – Тома Docker, создаваемые при установке

Том Docker	Начальный размер	Описание
<256-битное имя>	< 1 MB	Данные сервиса RabbitMQ (используется контейнером saltbox-rabbitmq-1)
saltbox_saltbox_core_var	< 1 MB	Ключи GPG (используется контейнерами: saltbox-saltbox-core-faststream-redis-1, saltbox-saltbox-core-tasks-watcher-1, saltbox-saltbox-core-1, saltbox-saltbox-core-taskiq-scheduler-1, saltbox-saltbox-core-taskiq-worker-1)
saltbox_keycloak-db_data	67 MB	Данные сервиса KeyCloak (используется контейнером saltbox-keycloak-db-1)
saltbox_salt_master_pki	< 1 MB	Хранилище ключей сервиса salt-master (используется контейнером saltbox-salt-master-1)
saltbox_redis_taskiq_data	< 1 MB	Данные сервиса taskiq (используется контейнером saltbox-redis-taskiq-1)
saltbox_mongo_config	< 1 MB	Конфигурационные данные СУБД MongoDB (используется контейнером saltbox-mongo-1)
saltbox_mongo_data	< 311 MB	Данные СУБД MongoDB (используется контейнером saltbox-mongo-1)
saltbox_redis_salt_data	< 1 MB	Данные сервера Redis (используется контейнером saltbox-redis-salt-1)
saltbox_redis_salt_certs	< 1 MB	Хранилище сертификатов сервера Redis (используется контейнерами: saltbox-saltbox-core-faststream-redis-1, saltbox-saltbox-core-tasks-watcher-1, saltbox-saltbox-core-1, saltbox-saltbox-core-taskiq-scheduler-1, saltbox-saltbox-core-taskiq-worker-1, saltbox-redis-salt-1, saltbox-make-redis-salt-certs-1, saltbox-salt-master-1)
saltbox_salt_master_saltbox_var	< 1 MB	 (используется контейнером saltbox-salt-master-1)
saltbox_salt_repos	< 1 MB	Схемы JSON для различных функций SaltStack (используется контейнерами: saltbox-saltbox-core-faststream-redis-1, saltbox-saltbox-core-tasks-watcher-1, saltbox-saltbox-core-1, saltbox-saltbox-core-taskiq-scheduler-1, saltbox-saltbox-core-taskiq-worker-1, saltbox-sshfs-1)

2.3.6 Вход в веб-интерфейс Salt.Box

В веб-браузере перейдите по адресу:

`http://<Адрес, указанный в переменной WEB_SERVER_OUTER_SOCKET файла .env>`

если сервер **Salt.Box** установлен локально, или:

`https://<Адрес, указанный в переменной WEB_SERVER_OUTER_SOCKET файла .env>`

если доступ к серверу **Salt.Box** осуществляется через обратный прокси-сервер.

Для подключения к веб-интерфейсу с внешнего хоста без использования SSL необходимо в используемом браузере указать исключения для небезопасных источников.

Ниже приведён пример соответствующей настройки для браузеров Chrome/Chromium:

- 1) Перейдите по адресу
`chrome://flags/#unsafely-treat-insecure-origin-as-secure`
- 2) Включите опцию **"Insecure origins treated as secure"**
- 3) В поле ввода укажите URL с адресом веб-интерфейса: `http://<IP_ADDRESS>`
- 4) Перезапустите браузер

При подключении к веб-интерфейсу с внешнего хоста с использованием SSL веб-браузер выдаст предупреждение о самоподписанном сертификате. Сделайте исключение для данного сайта.

2.3.7 Очистка

После внесения изменений контейнеры и тома могут стать несовместимыми с текущим кодом без специальных операций миграции объектов.

Чтобы исправить проблемы с запуском в среде разработки, остановите контейнеры нажатием комбинации клавиш **Control**-**c** и закройте их.

```
sudo docker compose -f compose.yaml -f compose-dev-override.yaml down --volumes
```

К СВЕДЕНИЮ

Если указан флаг **--volumes**, то будут удалены подключённые тома, что приведёт к потере данных.

Убедитесь, что Вы не потеряете важные данные.

2.3.7.1 Очистка устаревших данных Docker

При изменении кода и файлов конфигурации создаются новые слои и другие объекты Docker.

Для освобождения ресурсов Вы можете выполнить команду очистки неиспользуемых объектов Docker.

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	18
Дата созд.	10.07.2025		Страниц	70

ПРЕДУПРЕЖДЕНИЕ

Если у Вас есть остановленные контейнеры, которые Вы планируете перезапустить позже, они будут удалены.

То же касается образов, томов и сетей, которые не используются прямо сейчас.

Для управления удалением не используйте ключи **--all** и **--force**.

Команда очистки:

```
sudo docker system prune
```

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	19
Дата созд.	10.07.2025		Страниц	70

2.4 Настройка KeyCloak

Подсистема управления доступом и её пользовательский интерфейс в Salt.Box реализованы на основе программного продукта с открытым исходным кодом KeyCloak, распространяемого по лицензии *Apache License 2.0*.

2.4.1 Вход в KeyCloak

Войдите в веб-интерфейс подсистемы управления доступом по адресу:

https://<адрес_сервера_Salt.Box>/auth/keycloak/

Если развёртывание Программного комплекса «Salt.Config.Box» выполнено в контейнеризаторе приложений Docker, используйте пароль учётной записи ``admin``, сохранённый в файле `/secrets/keycloak-admin-password` в папке проекта (Рисунок 1).

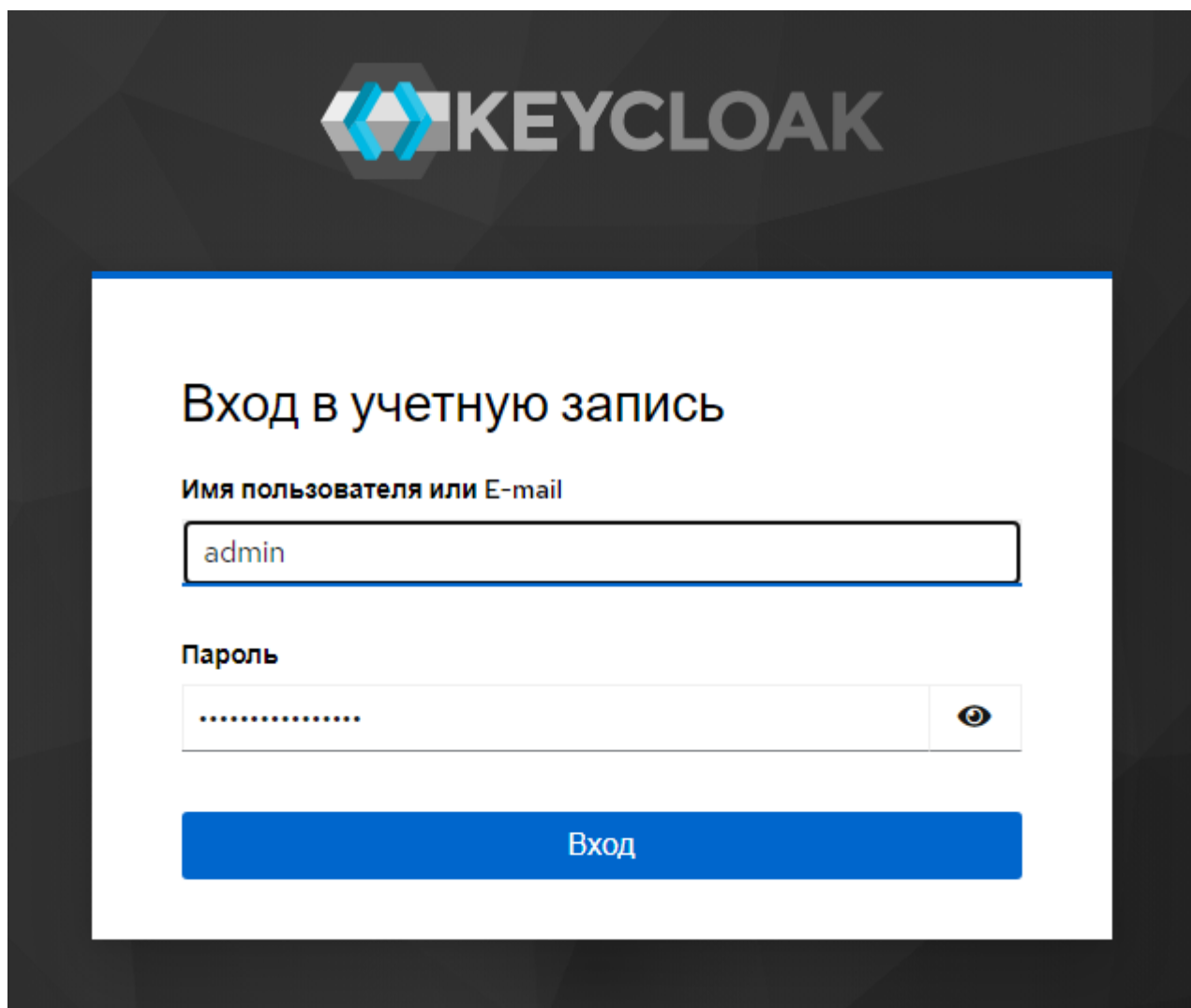


Рисунок 1 – Окно входа в KeyCloak

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	20
Дата созд.	10.07.2025		Страниц	70

2.4.2 Создание нового реалма

Учётная запись `admin` принадлежит реалму Keycloak по умолчанию с идентификатором `Master`.

По соображениям безопасности не рекомендуется создавать пользователей Salt.Box в данном реалме.

Откройте список реалмов (Рисунок 2 [1]) и нажмите кнопку **Создание реалма** для создания нового реалма.

Задайте имя реалма, например, salt.box, как на рисунке 3, и нажмите кнопку **Создать**.

Новый реалм будет создан.

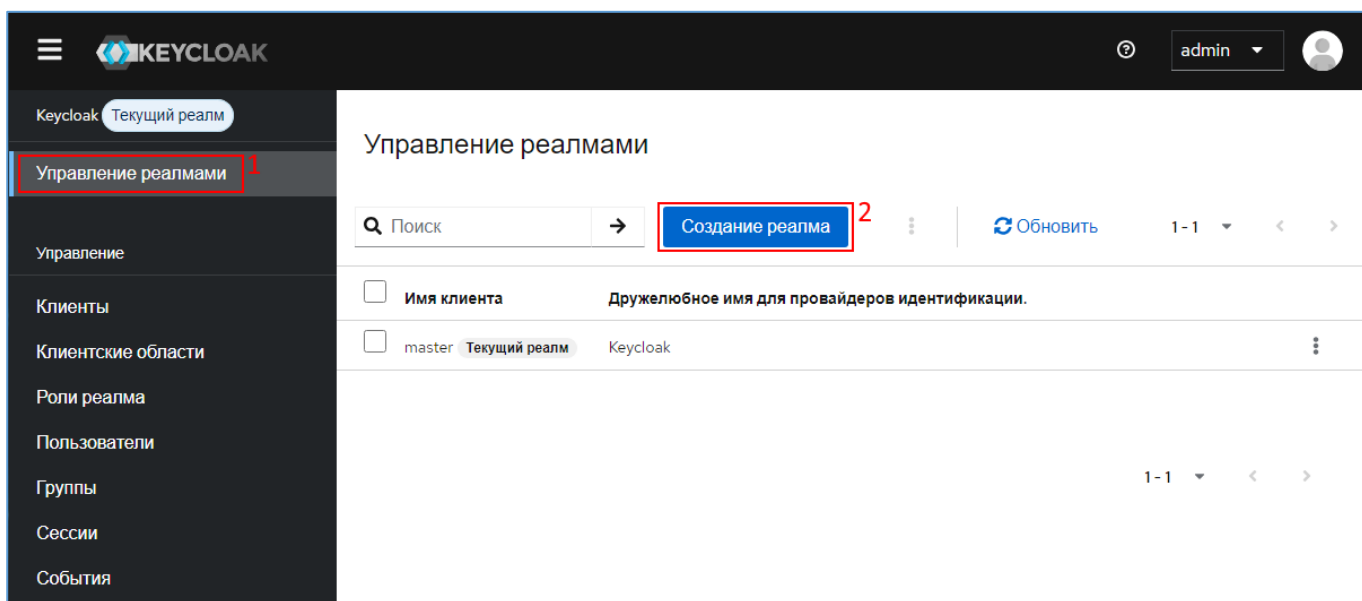


Рисунок 2 – Страница реалма по умолчанию (Master)

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	21
Дата созд.	10.07.2025		Страниц	70

Создание реалма

Реалм управляет набором пользователей, учетных данных, ролей и групп. Пользователь принадлежит к реалму и входит в него. Реалмы изолированы друг от друга и могут управлять и аутентифицировать только тех пользователей, которых они контролируют.

Файл ресурса

Drag a file here or browse to upload

Browse...

Clear

Загрузить JSON-файл

Имя реалма *

salt.box

Включено

☒ Вкл

Создать

Отмена

Рисунок 3 – Диалоговое окно создания нового реалма

2.4.3 Создание клиента

- Выберите пункт меню **Клиенты** главного меню KeyCloak (Рисунок 4).

Клиентом Keycloak является веб-приложение `saltbox_core`, входящее в состав Программного комплекса. Зарегистрируйте это приложение в KeyCloak, для чего откройте диалог создания клиента, нажав кнопку **Создание клиента** (Рисунок 4 [1]).

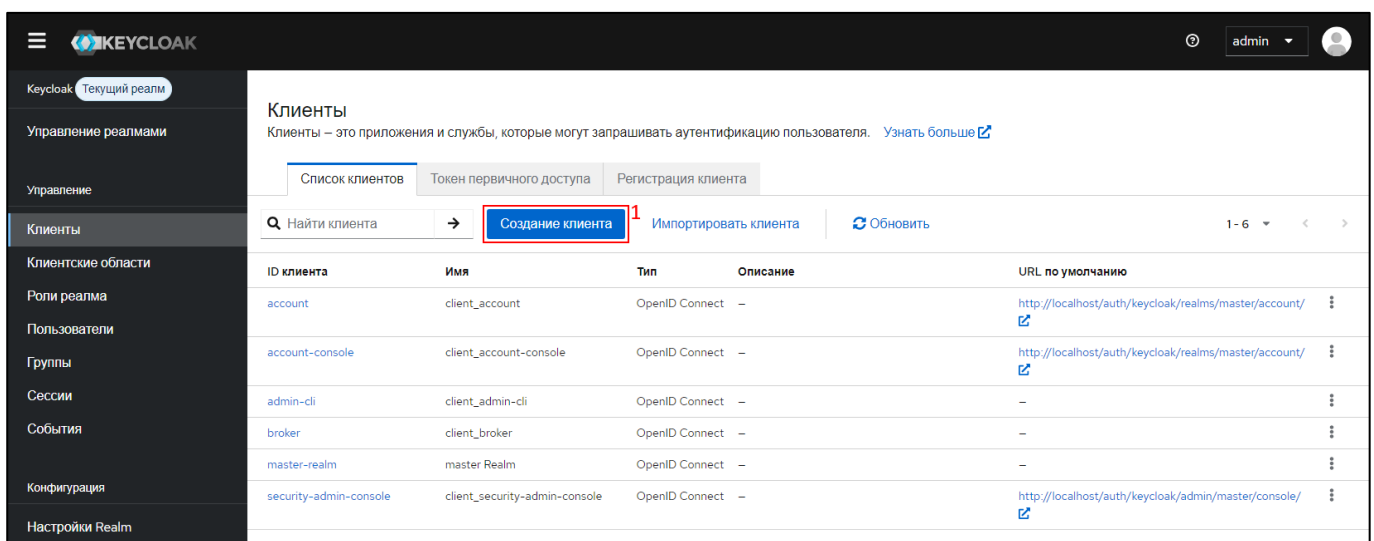


Рисунок 4 – Страница клиентов

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	22
Дата созд.	10.07.2025		Страниц	70

2. На вкладке **Общие настройки** диалога создания клиента введите параметры:

Поле	Значение	Комментарий
Тип клиента	OpenID Connect	обязательное поле
ID клиента	saltbox_core	обязательное поле
Имя клиента		любое наименование

затем нажмите **Далее**.

Клиенты > Создание клиента

Создание клиента

Клиенты – это приложения и службы, которые могут запрашивать аутентификацию пользователя.

1 Общие настройки

2 Настройка возможностей

3 Настройки входа

Тип клиента ?

OpenID Connect

ID клиента * ?

saltbox_core

Имя клиента ?

Описание ?

Всегда отображать в пользовательском интерфейсе ?

Выкл.

Назад

Далее

Отмена

Рисунок 5 – Окно создания клиента: вкладка общих настроек

3. На вкладке **Настройка возможностей** диалога создания клиента (Рисунок 6) установите флажки:

Поле	Значение
Стандартный поток	☒
Прямой доступ	☒

затем нажмите **Далее**.

Клиенты > Создание клиента

Создание клиента

Клиенты – это приложения и службы, которые могут запрашивать аутентификацию пользователя.

1 Общие настройки
2 **Настройка возможностей**
3 Настройки входа

Аутентификация клиента

☐ Выкл.

Авторизация

☐ Выкл.

Поток аутентификации

☒ Стандартный поток
☒ Прямой доступ

☐ Неявный поток
☐ Роли сервисных аккаунтов

☐ Стандартный обмен токенами

☐ Предоставление авторизации устройства OAuth 2.0

☐ Предоставление OIDC CIBA

Назад
Далее
Отмена

Рисунок 6 – Окно создания клиента: вкладка настроек функциональных возможностей

- Оставьте без изменений параметры на вкладке **Настройки входа** (Рисунок 7), нажмите **Сохранить**.

Будет создан новый клиент KeyCloak.

Клиенты > Создание клиента

Создание клиента

Клиенты – это приложения и службы, которые могут запрашивать аутентификацию пользователя.

1 Общие настройки
2 Настройка возможностей
3 **Настройки входа**

Корневой URL ?

URL по умолчанию ?

URI для перенаправления после успешного входа в систему ?

+ Добавить допустимые URI для перенаправления

URI для перенаправления после успешного выхода из системы ?

+ Добавить допустимые URI для перенаправления после выхода из системы

Web источники ?

+ Добавить веб-источники

Назад Сохранить Отмена

Рисунок 7 – Окно создания клиента: вкладка настроек параметров входа

2.4.3.1 Создание роли клиента

Программный комплекс «Salt.Config.Box» предоставляет полномочия на управление коллекциями пользователям с ролью `collections_admin`.

Создайте эту роль для клиента `saltbox_core` реалма `salt.box`.

Выполните действия, указанные ниже.

1. На странице **Клиенты** откройте созданный клиент `saltbox_core` и перейдите на вкладку **Роли**.

Нажмите кнопку **Создание роли** (Рисунок 8).

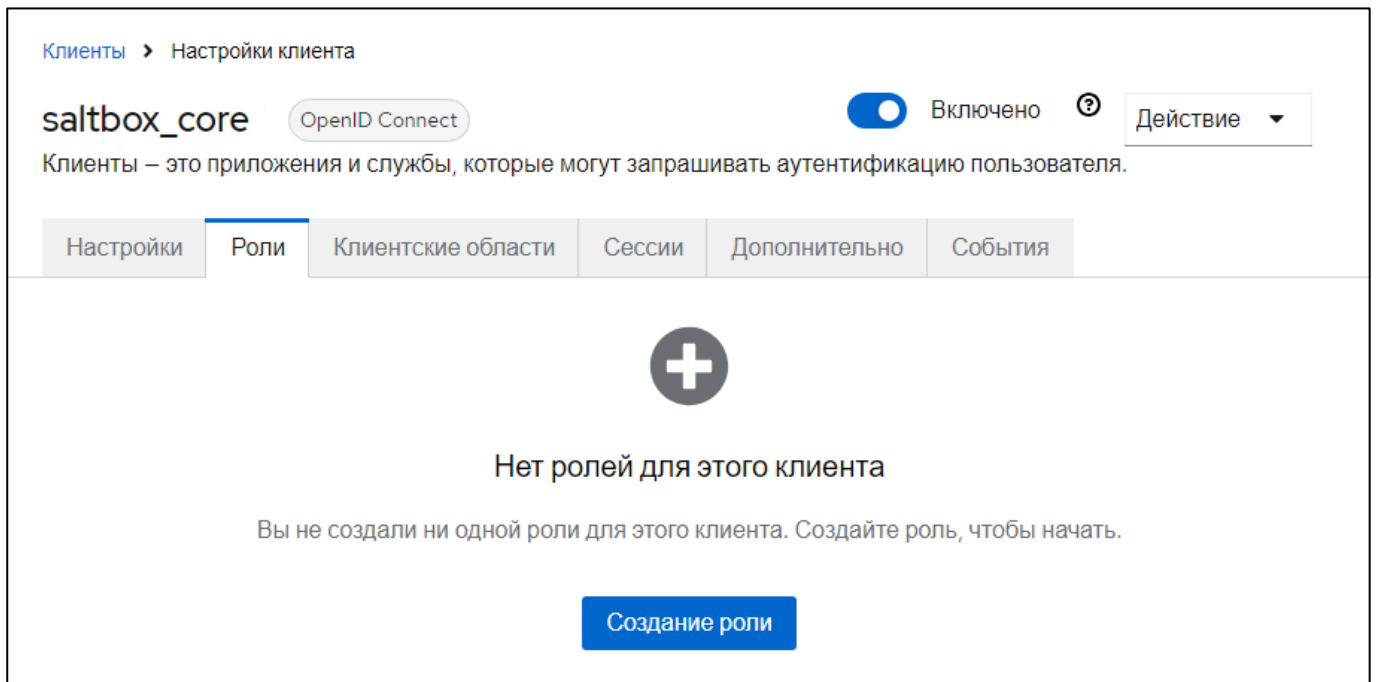


Рисунок 8 –

2. Введите параметры роли в диалоговом окне создания роли (Рисунок 9):

Поле	Значение	Комментарий
Наименование роли	collections_admin	обязательное поле
Описание		<i>произвольная строка</i>

затем нажмите **Save**.

Клиенты > Настройки клиента > Создание роли

Создание роли

Наименование роли *

Описание

Сохранить

Отмена

Рисунок 9 – Диалоговое окно создания роли

Созданную роль далее можно будет назначить любому пользователю Salt.Box: см. раздел 2.4.4.1 «Назначение роли пользователю».

Также Вы можете узнать у разработчика программного продукта наименования и назначение других встроенных ролей Salt.Box и правила нотации для доступа к ресурсам Salt.Box.

2.4.4 Создание пользователя

1. Введите данные и создайте нового пользователя:

а. Выберите пункт меню **Пользователи** главного меню **KeyCloak** (Рисунок 10).

Нажмите **Добавить пользователя** над таблицей пользователей (Рисунок 10 [1]).

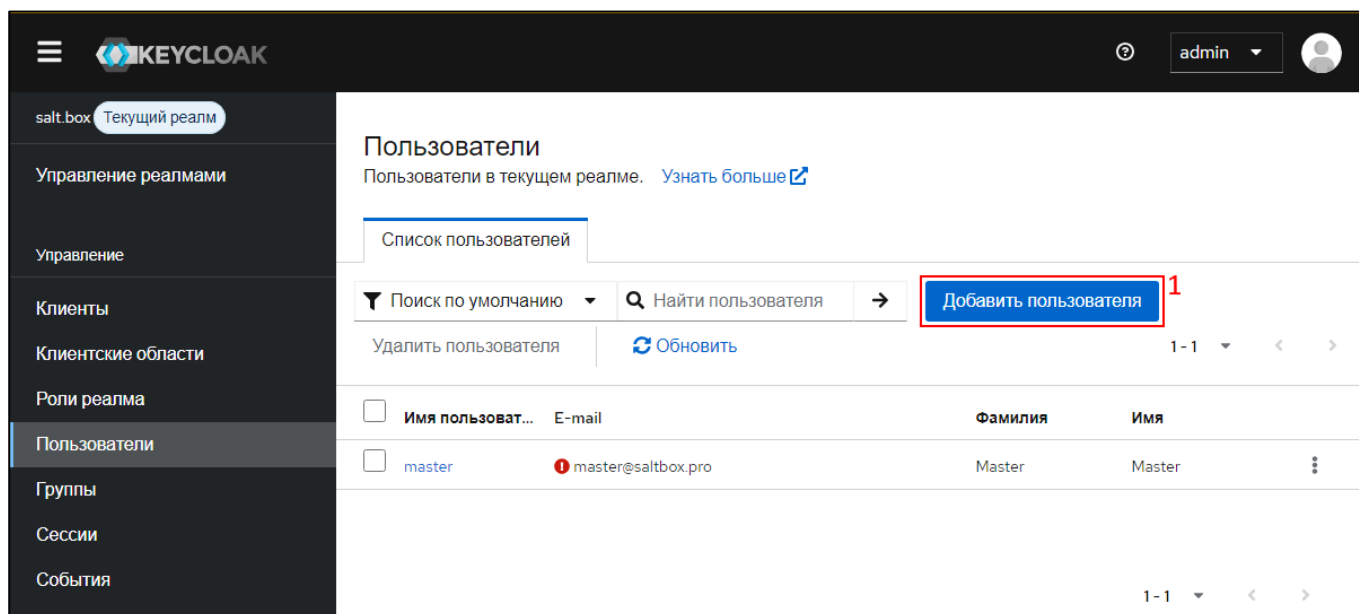


Рисунок 10 – Страница управления учётными записями пользователей

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	28
Дата созд.	10.07.2025		Страниц	70

б. В открывшемся окне введите данные пользователя (Рисунок 11):

Поле	Значение	Комментарий
Имя пользователя	user	обязательное поле
E-mail		
Имя		
Фамилия		

Пользователи > Создание пользователя

Создание пользователя

Требуемые действия от пользователя ?

Update Password x
1
Выбрать действия
x

Подтверждение E-mail ?

Выкл.

Главная

Перейти в раздел

Выбрать локаль
русский

Имя пользователя *
user

E-mail
user@example.com

Имя
Иван

Фамилия
Иванов

Группы ?

Добавить в группы

Главная

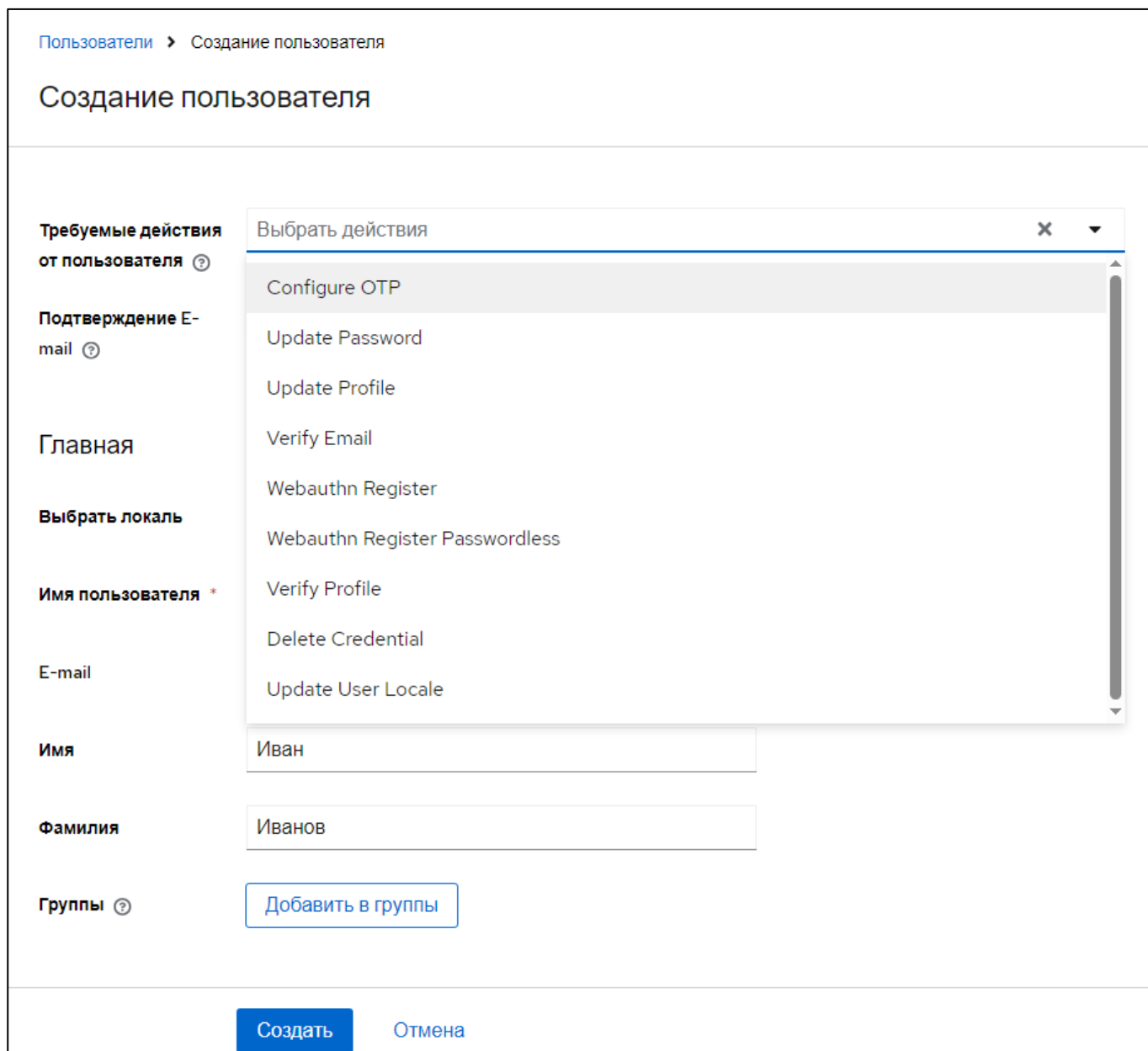
Создать

Отмена

Рисунок 11 – Ввод данных пользователя

- в. В случае необходимости укажите требуемые дополнительные действия с учётной записью пользователя, например, обязательную смену пароля при первом входе пользователя (Update Password) (Рисунок 12).

В окне создания пользователя появится вкладка **Update Password** (Рисунок 11 [1]).



Пользователи > Создание пользователя

Создание пользователя

Требуемые действия от пользователя ?

Подтверждение E-mail ?

Главная

Выбрать локаль

Имя пользователя *

E-mail

Имя

Фамилия

Группы ?

Выбор действий

- Configure OTP
- Update Password
- Update Profile
- Verify Email
- Webauthn Register
- Webauthn Register Passwordless
- Verify Profile
- Delete Credential
- Update User Locale

Иван

Иванов

Добавить в группы

Создать Отмена

Рисунок 12 – Добавление требуемых действий пользователя

- г. Нажмите **Создать** (Рисунок 11).

Будет создан новый пользователь, отобразится окно **Настройки пользователя** (Рисунок 13).

2. Задайте пароль для созданного пользователя.

На странице **Пользователи** откройте созданный аккаунт user и перейдите на вкладку **Учетные данные**.

а. Нажмите **Установить пароль** (Рисунок 13).

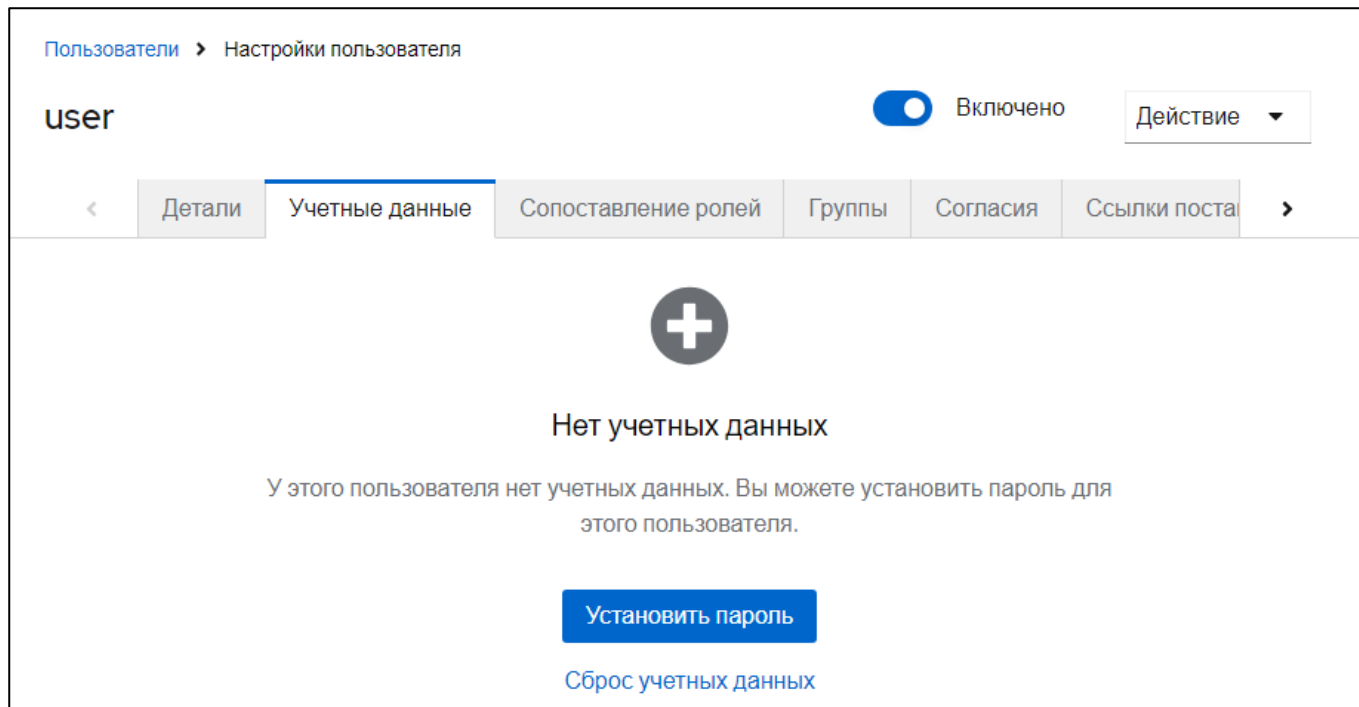


Рисунок 13 – Вкладка управления учётными данными пользователя

б. Задайте пароль пользователя в диалоговом окне (Рисунок 14).

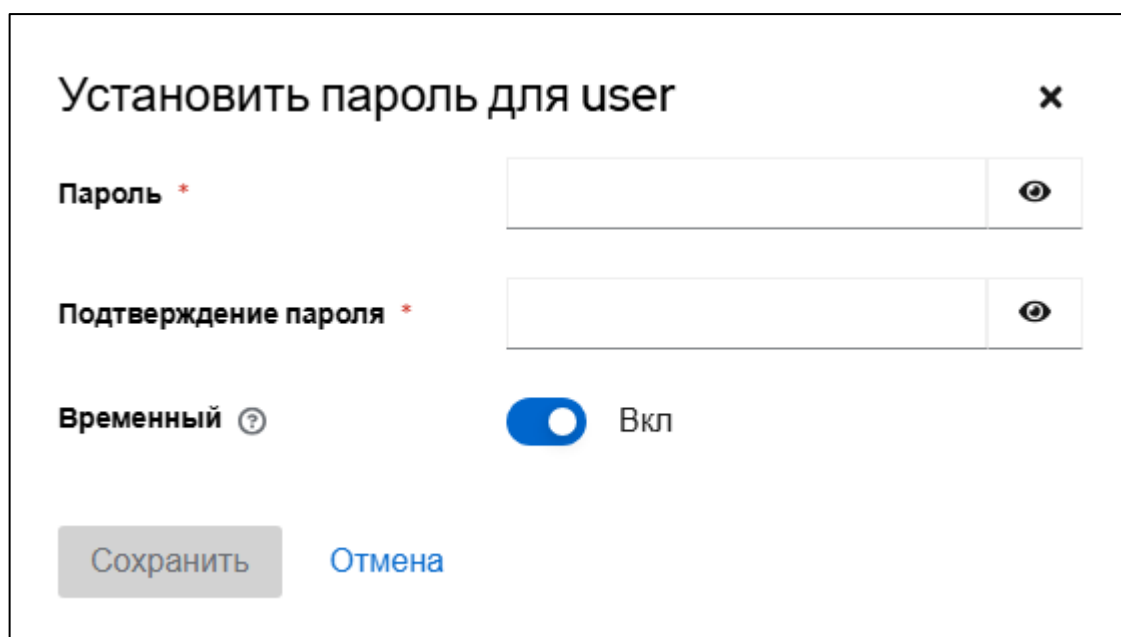


Рисунок 14 – Установка пароля пользователя

2.4.4.1 Назначение роли пользователю

Теперь назначьте ранее созданную роль `collections_admin` (см. раздел 2.4.3.1 «Создание роли клиента») учетной записи `user`, чтобы предоставить ей разрешения на управление коллекциями.

1. На странице **Пользователи** откройте созданный аккаунт `user` и перейдите на вкладку **Сопоставление ролей**.

Нажмите **Назначить роль** (Рисунок 15).

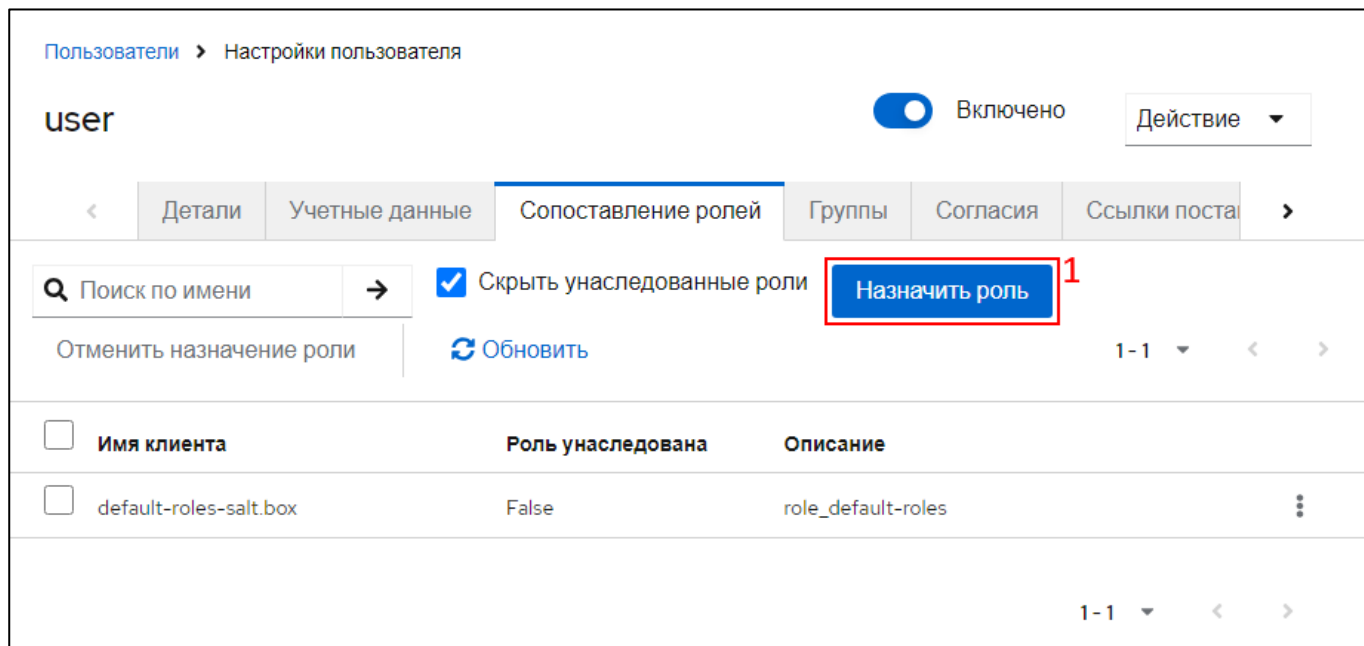


Рисунок 15 – Вкладка сопоставления ролей

- Введите название роли `collections_admin` в строку поиска и нажмите кнопку  (Рисунок 16 [1]).

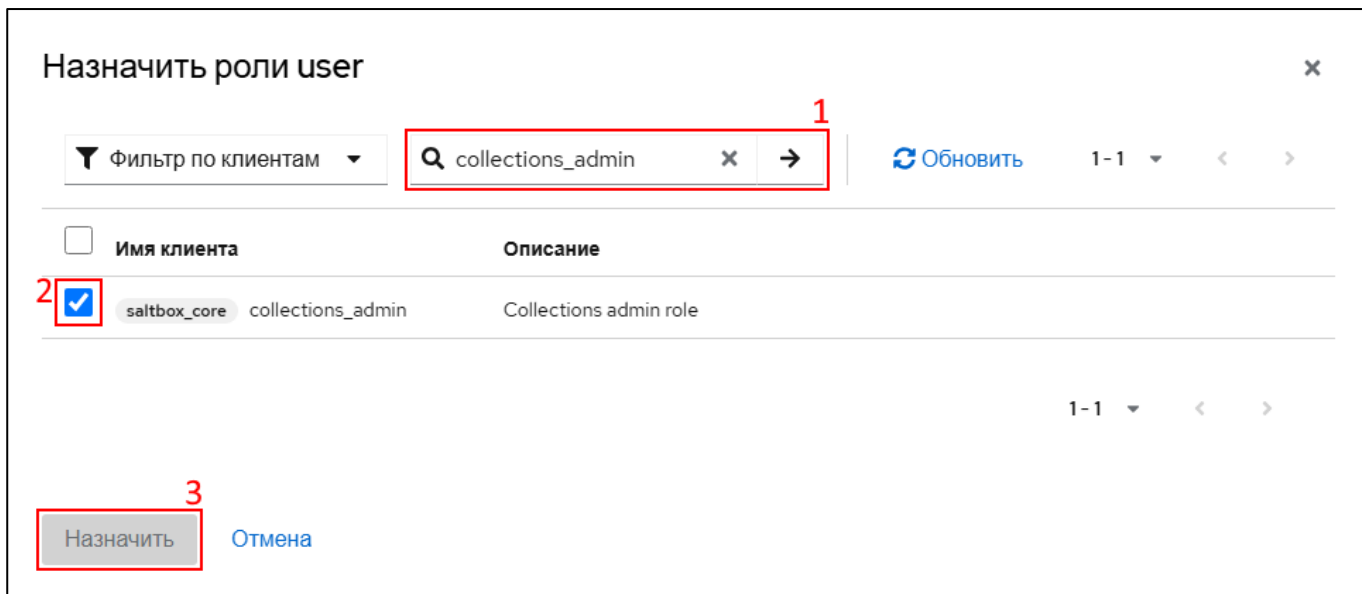


Рисунок 16 – Назначение роли пользователю

- Поставьте флажок на записи с ролью `collections_admin`, ассоциированной с клиентом `saltbox_core` (Рисунок 16 [2]).
- Нажмите **Назначить** (Рисунок 16 [3]).

Роль `collections_admin` будет назначена пользователю `user`.

Пользователю `user` будут доступны объекты корневой коллекции.

2.5 Установка клиента Salt.Config.Box

ПРЕДУПРЕЖДЕНИЕ

В дистрибутиве **Salt.Box** используются компоненты **SaltStack 3006.9 LTS**.

Данный релиз **SaltStack** имеет долгосрочную поддержку.

Это означает, что релиз будет поддерживаться в течение более длительного времени, чем обычные версии.

Подразумевается, что в течение этого периода будут выходить обновления, включающие исправления безопасности и критические исправления ошибок, но не новые функции, которые могут нарушить совместимость.

Для обеспечения полной совместимости клиента **Salt Minion** используйте на управляемых устройствах клиент той же версии — **SaltStack 3006.9 LTS**.

2.5.1 Установка в ОС Linux

2.5.1.1 Установка пакета RPM

1. Для установки репозитория **Salt Project** выполните следующую команду:

```
curl -fsSL https://github.com/saltstack/salt-install-guide/releases/latest/download/salt.repo |  
sudo tee /etc/yum.repos.d/salt.repo
```

ПРИМЕЧАНИЕ

Из-за наличия в **EPEL** классических пакетов Salt возможна загрузка классических пакетов старых версий Salt вместо пакетов onedir.

Во время разрешения зависимостей при выборе лучшего поставщика среди нескольких **dnf** учитывает приоритет репозитория каждого поставщика.

Значение представляет собой целое число от 1 до 99, где 1 — наиболее предпочтительный репозиторий, а 99 — наименее предпочтительный.

По умолчанию все репозитории имеют приоритет 80.

EPEL рассматривал репозиторий Salt как 99. Чтобы решить эту проблему, Salt изменил свой уровень приоритета на 10 для **RHEL 8** и **9**.

2. Чтобы очистить метаданные репозитория, выполните:

```
sudo dnf clean expire-cache
```

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	34
Дата созд.	10.07.2025		Страниц	70

3. Установка пакета **salt-minion**:

ПРЕДУПРЕЖДЕНИЕ

STS-релизы не рекомендуются для продуктивной среды.

Salt Project рекомендует развёртывать **LTS**-релизы в продуктивной среде.

- **3006 LTS**

Установите пакет:

```
sudo dnf install salt-minion
```

Если вы хотите установить конкретный релиз **salt-minion**, укажите суффикс полной версии релиза Salt.

Например:

```
sudo dnf install salt-minion-3006.9
```

Для закрепления версии пакета, когда необходимо исключить его обновление при выполнении `dnf upgrade`, может быть использована команда `dnf versionlock`.

```
sudo dnf install 'dnf-command(versionlock)'
```

```
sudo dnf versionlock add salt-minion
```

- **3007 STS**

Для установки релиза **Salt 3007 STS** необходимо разрешить его загрузку:

```
# Включить репозиторий Salt 3007 STS
```

```
sudo dnf config-manager --set-disable salt-repo-*
```

```
sudo dnf config-manager --set-enabled salt-repo-3007-sts
```

Установите пакет:

```
sudo dnf install salt-minion
```

Если вы хотите установить конкретный релиз **salt-minion**, укажите суффикс полной версии релиза Salt.

Например:

```
sudo dnf install salt-minion-3007.5
```

Для закрепления версии пакета, когда необходимо исключить его обновление при выполнении `dnf upgrade`, может быть использована команда `dnf versionlock`.

```
sudo dnf install 'dnf-command(versionlock)'
```

```
sudo dnf versionlock add salt-minion
```

- **LATEST Available**

Для установки релиза **LTS** или **STS**, в зависимости от того, какой из них является актуальным, необходимо разрешить загрузку из репозитория с суффиксом 'latest':

```
# Включить репозиторий Salt LATEST
```

```
sudo dnf config-manager --set-disable salt-repo-*
```

```
sudo dnf config-manager --set-enabled salt-repo-latest
```

Установите пакет:

```
sudo dnf install salt-minion
```

Если вы хотите установить конкретный релиз **salt-minion**, укажите суффикс полной версии релиза Salt.

Например:

```
sudo dnf install salt-minion-3007.5
```

Для закрепления версии пакета, когда необходимо исключить его обновление при выполнении `dnf upgrade`, может быть использована команда `dnf versionlock`.

```
sudo dnf install 'dnf-command(versionlock)'
```

```
sudo dnf versionlock add salt-minion
```

4. Включите и запустите службу salt-minion:

```
sudo systemctl enable salt-minion && sudo systemctl start salt-minion
```

ПРИМЕЧАНИЕ

При установке версии Salt onedir (**3006 и более поздние**) Salt устанавливает собственную локальную версию Python и зависимости, необходимые для основных функций Salt.

После установки версии Salt onedir в Вашей системе будет присутствовать как глобальная версия Python на системном уровне, так и локальная версия Python, используемая Salt.

Это изменение означает, что пути Salt onedir для Python отличаются, и Вам необходимо изменить способ установки сторонних зависимостей Python, которые Вы используете с Salt, включая ваши файлы состояния.

Для получения дополнительной информации см. **«Установка зависимостей»**:

<https://docs.saltproject.io/salt/install-guide/en/latest/topics/install-dependencies.html#install-dependencies>

2.5.1.2 Установка пакета DEB

1. Для установки репозитория Salt Project выполните следующие команды:

```
# Убедиться, что каталог keyrings существует
```

```
mkdir -p /etc/apt/keyrings
```

```
# Загрузить открытый ключ
```

```
curl -fsSL https://packages.broadcom.com/artifactory/api/security/keypair/SaltProjectKey/public  
| sudo tee /etc/apt/keyrings/salt-archive-keyring.gpg
```

```
# Создать целевую конфигурацию репозитория
```

```
curl -fsSL https://github.com/saltstack/salt-install-guide/releases/latest/download/salt.sources |  
sudo tee /etc/apt/sources.list.d/salt.sources
```

2. Для обновления метаданных выполните:

```
sudo apt update
```

3. Установка пакета **salt-minion**:

ПРЕДУПРЕЖДЕНИЕ

STS-релизы не рекомендуются для продуктивной среды.

Salt Project рекомендует развёртывать **LTS**-релизы в продуктивной среде.

- **3006 LTS**

Чтобы ограничить обновления мажорной версией **Salt 3006 LTS**, создайте файл `/etc/apt/preferences.d/salt-pin-1001` с помощью следующей команды:

```
echo 'Package: salt-*
```

```
Pin: version 3006.*
```

```
Pin-Priority: 1001' | sudo tee /etc/apt/preferences.d/salt-pin-1001
```

Установите пакет:

```
sudo apt-get install salt-minion
```

Если вы хотите установить конкретную версию **salt-minion**, укажите суффикс полной версии релиза Salt.

Например:

```
sudo apt-get install salt-minion=3006.9
```

ПРЕДУПРЕЖДЕНИЕ

Конфликты зависимостей Salt

Если Вы используете не последний релиз целевой мажорной версии, Вам может потребоваться установить другие пакеты salt в закреплённом виде.

Например, чтобы установить **salt-minion**, потребуется установить **salt-common** той же версии:

```
sudo apt-get install salt-minion=3006.9 salt-common=3006.9
```

- **3007 STS**

Чтобы ограничить обновления мажорной версией **Salt 3007 STS**, создайте файл `/etc/apt/preferences.d/salt-pin-1001` с помощью следующей команды:

```
echo 'Package: salt-*
```

```
Pin: version 3007.*
```

```
Pin-Priority: 1001' | sudo tee /etc/apt/preferences.d/salt-pin-1001
```

Установите пакет:

```
sudo apt-get install salt-minion
```

Если вы хотите установить конкретную версию **salt-minion**, укажите суффикс полной версии релиза Salt.

Например:

```
sudo apt-get install salt-minion=3007.5
```

Пример команды для закрепления текущего второстепенного пакета, чтобы он пропусклся во время общесистемного обновления apt-get

```
sudo apt-mark hold salt-minion
```

ПРЕДУПРЕЖДЕНИЕ

Конфликты зависимостей Salt

Если Вы используете не последний релиз целевой мажорной версии, Вам может потребоваться установить другие пакеты salt в закреплённом виде.

Например, чтобы установить **salt-minion**, потребуется установить **salt-common** той же версии:

```
sudo apt-get install salt-minion=3007.5 salt-common=3007.5
```

- **LATEST Available**

Установка релиза **LTS** или **STS**, в зависимости от того, какой из них является актуальным.

Установите пакет:

```
sudo apt-get install salt-minion
```

Если вы хотите установить конкретную версию **salt-minion**, укажите суффикс полной версии релиза Salt.

Например:

```
sudo apt-get install salt-minion=3007.5
```

Пример команды для закрепления текущего второстепенного пакета, чтобы он пропусклся во время общесистемного обновления apt-get

```
sudo apt-mark hold salt-minion
```

ПРЕДУПРЕЖДЕНИЕ

Если Вы используете не последний релиз целевой мажорной версии, Вам может потребоваться установить другие пакеты salt в закреплённом виде.

Например, чтобы установить **salt-minion**, потребуется установить **salt-common** той же версии:

```
sudo apt-get install salt-minion=3007.5 salt-common=3007.5
```

4. Включите и запустите службу salt-minion:

```
sudo systemctl enable salt-minion && sudo systemctl start salt-minion
```

ПРИМЕЧАНИЕ

При установке версии Salt onedir (**3006 и более поздние**) Salt устанавливает собственную локальную версию Python и зависимости, необходимые для основных функций Salt.

После установки версии Salt onedir в Вашей системе будет присутствовать как глобальная версия Python на системном уровне, так и локальная версия Python, используемая Salt.

Это изменение означает, что пути Salt onedir для Python отличаются, и Вам необходимо изменить способ установки сторонних зависимостей Python, которые Вы используете с Salt, включая ваши файлы состояния.

Для получения дополнительной информации см. «**Установка зависимостей**»:

<https://docs.saltproject.io/salt/install-guide/en/latest/topics/install-dependencies.html#install-dependencies>

2.5.1.3 Установка на Альт Линукс

В операционных системах Альт Линукс версий 10.4, 11.0 будет установлен пакет **salt-minion** версии **3007.1** из официального репозитория.

```
sudo apt-get update
```

```
sudo apt-get install salt-minion
```

2.5.1.4 Установка на Astra Linux SE

В операционных системах Astra Linux SE версий 1.7.7, 1.8.2 будет установлен пакет **salt-minion** версии **3006.9** из официального репозитория.

```
sudo apt-get update
```

```
sudo apt-get install salt-minion
```

2.5.1.5 Настройка

После установки salt-minion необходимо настроить его для связи с Master-сервером.

Для этого в файле **/etc/salt/minion** укажите адрес или имя хоста **Salt Master**:

```
master: <IP_ADDRESS> | <HOST>
```

После изменения файла конфигурации перезапустите службу salt-minion:

```
sudo systemctl restart salt-minion
```

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	40
Дата созд.	10.07.2025		Страниц	70

2.5.2 Установка в ОС Windows

2.5.2.1 Интерактивная установка

1. Загрузите файл установки клиента Salt Minion, соответствующий разрядности Вашей ОС, с сайта проекта Salt Project:

<https://docs.saltproject.io/salt/install-guide/en/latest/topics/install-by-operating-system/windows.html>

Архитектура	Тип файла	Тип релиза	Файл установки
3006 LTS			
AMD64	exe	LTS	https://packages.broadcom.com/artifactory/saltproject-generic/windows/3006.9/Salt-Minion-3006.9-Py3-AMD64-Setup.exe
AMD64	msi	LTS	https://packages.broadcom.com/artifactory/saltproject-generic/windows/3006.9/Salt-Minion-3006.9-Py3-AMD64.msi
x86	exe	LTS	https://packages.broadcom.com/artifactory/saltproject-generic/windows/3006.9/Salt-Minion-3006.9-Py3-x86-Setup.exe
x86	msi	LTS	https://packages.broadcom.com/artifactory/saltproject-generic/windows/3006.9/Salt-Minion-3006.9-Py3-x86.msi
3007 STS			
AMD64	exe	STS	https://packages.broadcom.com/artifactory/saltproject-generic/windows/3007.5/Salt-Minion-3007.5-Py3-AMD64-Setup.exe
AMD64	msi	STS	https://packages.broadcom.com/artifactory/saltproject-generic/windows/3007.5/Salt-Minion-3007.5-Py3-AMD64.msi
x86	exe	STS	https://packages.broadcom.com/artifactory/saltproject-generic/windows/3007.5/Salt-Minion-3007.5-Py3-x86-Setup.exe
x86	msi	STS	https://packages.broadcom.com/artifactory/saltproject-generic/windows/3007.5/Salt-Minion-3007.5-Py3-x86.msi

2. Для установки клиента **Salt.Box** в интерактивном режиме запустите файл установки без указания параметров запуска.

3. Нажмите **Next >** в стартовом окне инсталлятора (Рисунок 17).

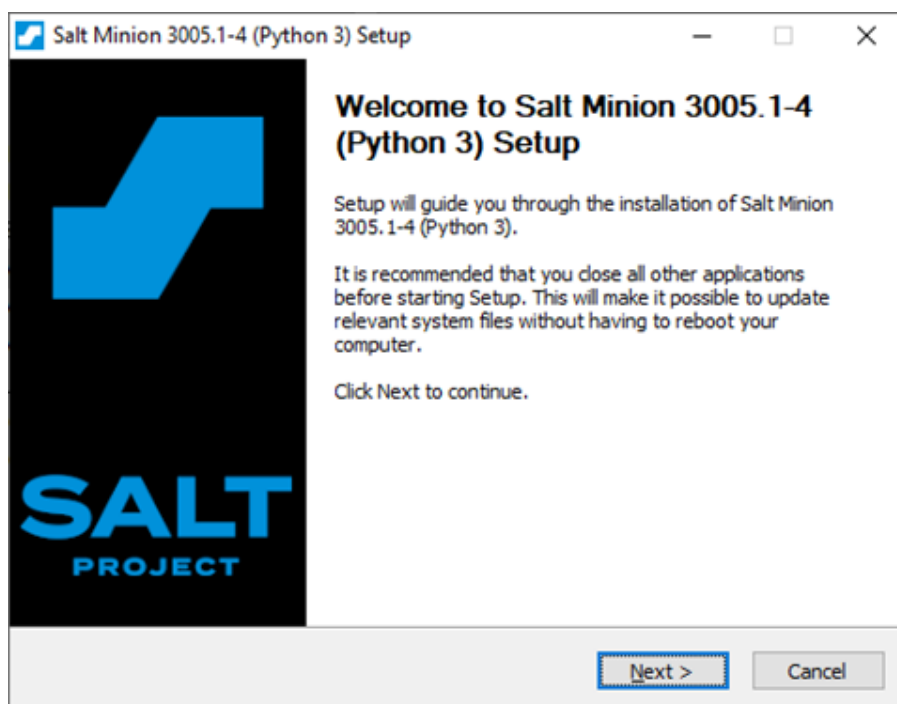


Рисунок 17 – Стартовое окно инсталлятора службы salt-minion

4. Нажмите **I Agree** в окне лицензионного соглашения (Рисунок 18).

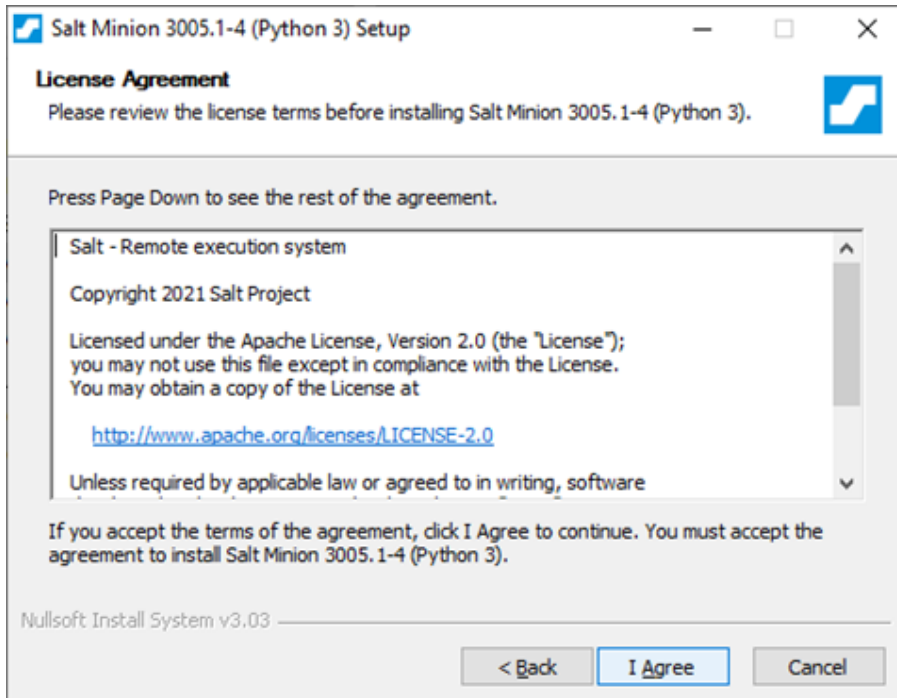


Рисунок 18 – Окно лицензионного соглашения

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	42
Дата созд.	10.07.2025		Страниц	70

5. Укажите путь к папке инсталляции клиента (Рисунок 19).

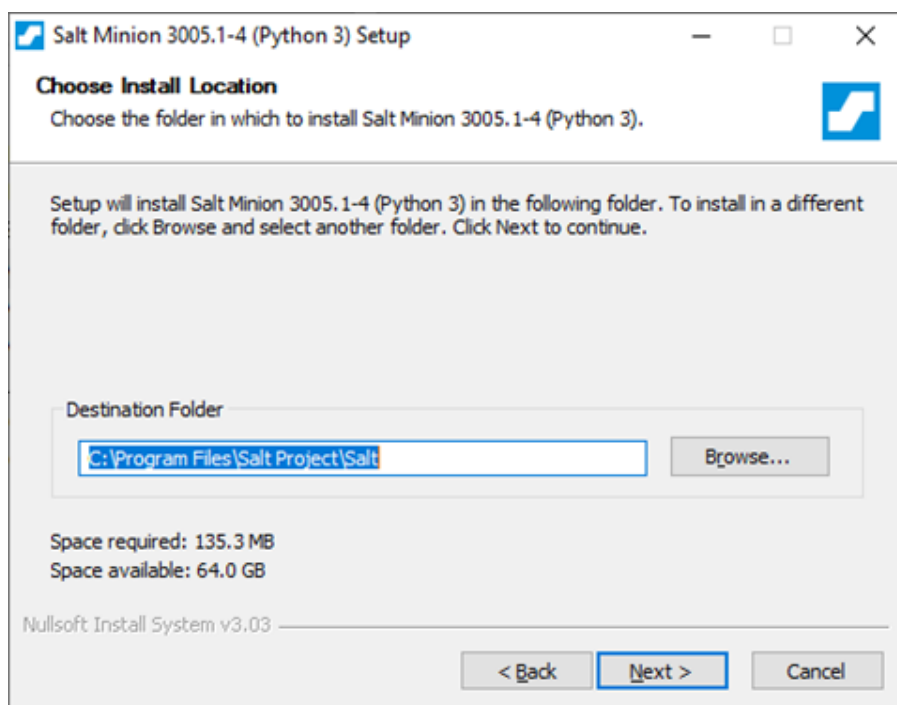


Рисунок 19 – Окно выбора папки назначения

6. В окне установки параметров службы salt-minion в первом поле укажите выделенный IP-адрес сервера.

Параметры **Minion Name: hostname** и **Default config** оставьте без изменений (Рисунок 20). Нажмите **Install** и дождитесь окончания установки клиента.

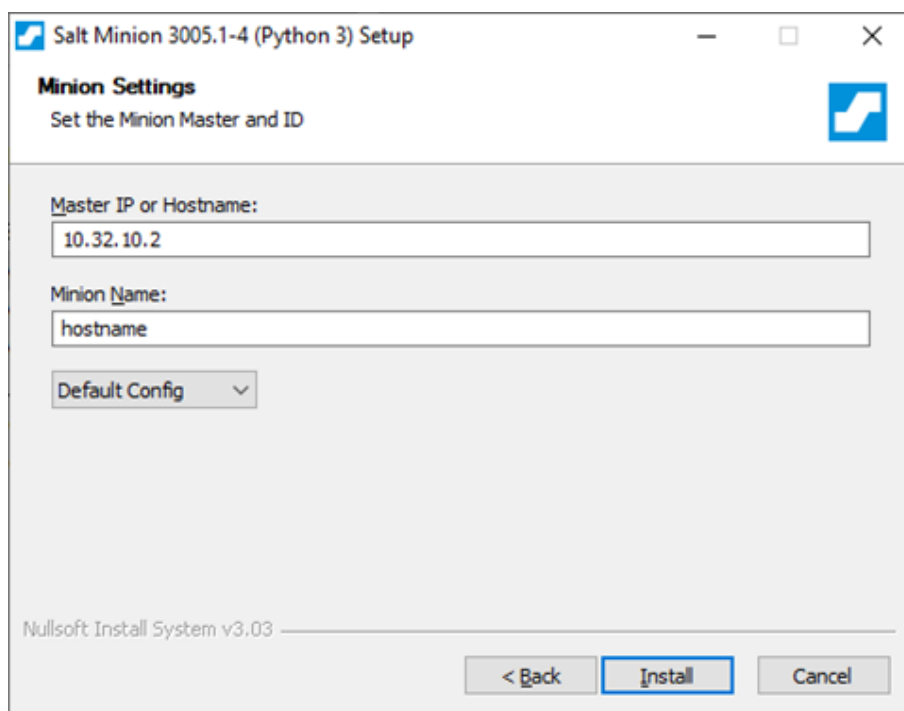


Рисунок 20 – Окно установки параметров службы salt-minion

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	43
Дата созд.	10.07.2025		Страниц	70

- Нажмите **Finish** в окне завершения установки (Рисунок 21).

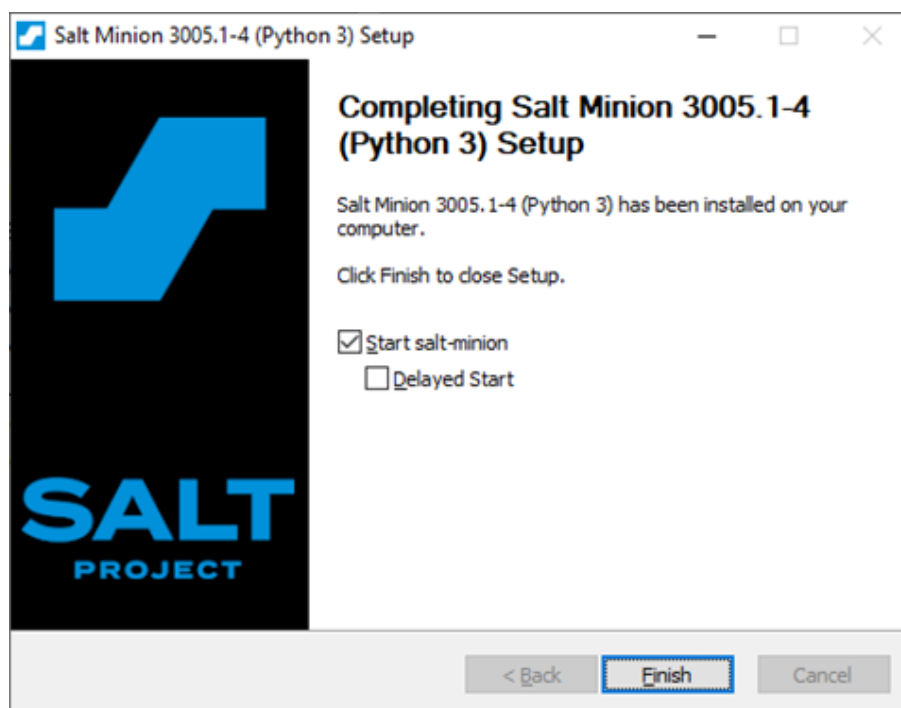


Рисунок 21 – Окно завершения установки

2.5.2.2 Установка с помощью групповых политик Microsoft Active Directory

- В меню **Пуск** выберите пункт **Administrative Tools -> Group Policy Management**, либо выберите пункт **Выполнить** и запустите консоль **gpmmc.msc** (Рисунок 22).

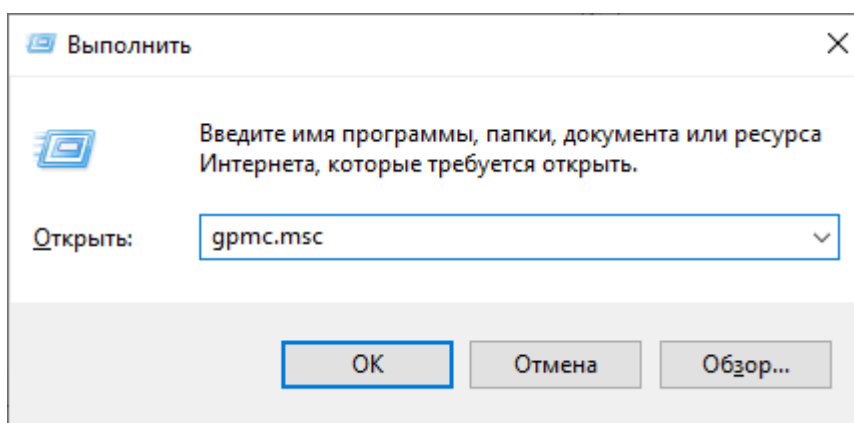


Рисунок 22 – Запуск оснастки управления групповой политикой

- В дереве объектов домена кликните правой кнопкой мыши на узле **Объекты групповой политики**, выберите пункт **Создать** (Рисунок 23).

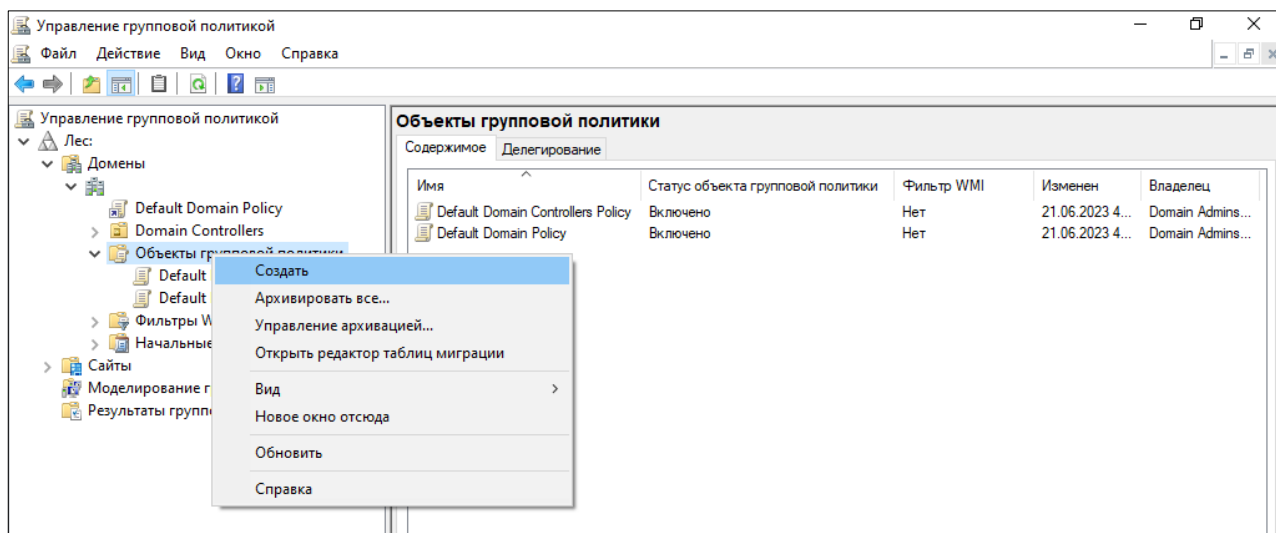


Рисунок 23 – Создание объекта групповой политики

3. В диалоговом окне введите наименование нового объекта групповой политики и нажмите **ОК**. В данном примере объект групповой политики называется **"Установка клиента"** (Рисунок 24).

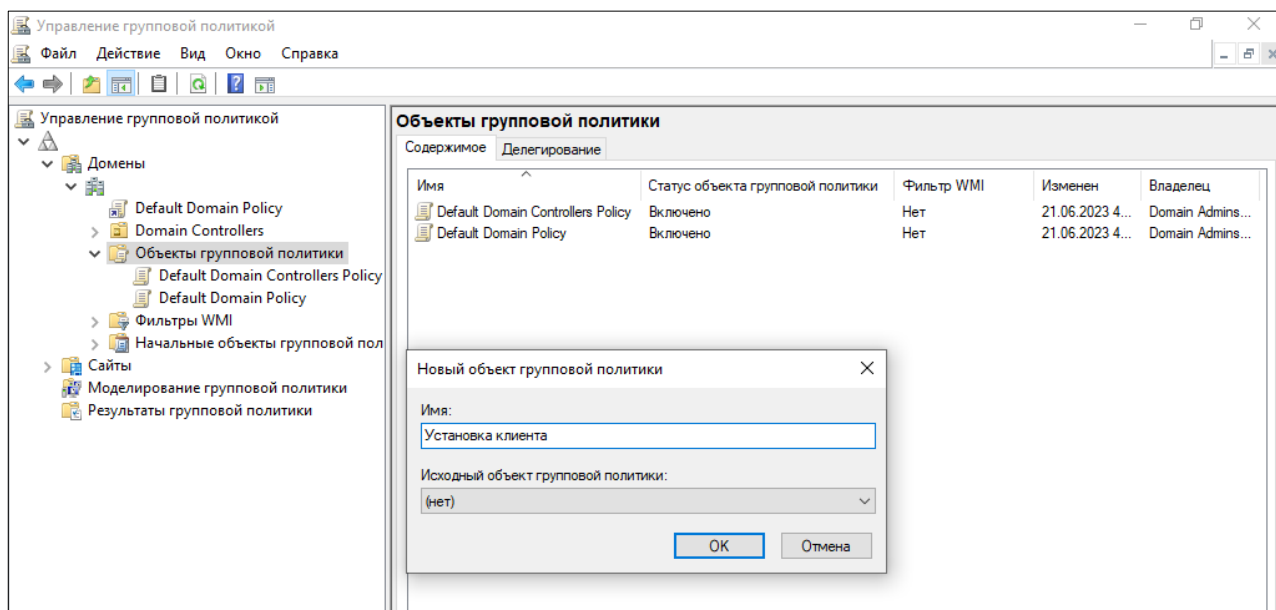


Рисунок 24 – Ввод наименования нового объекта групповой политики

4. В дереве объектов домена кликните правой кнопкой мыши на нужном организационном подразделении (OU) (в данном примере — **"Тестовое подразделение"**), выберите пункт **Связать существующий объект групповой политики** (Рисунок 25).

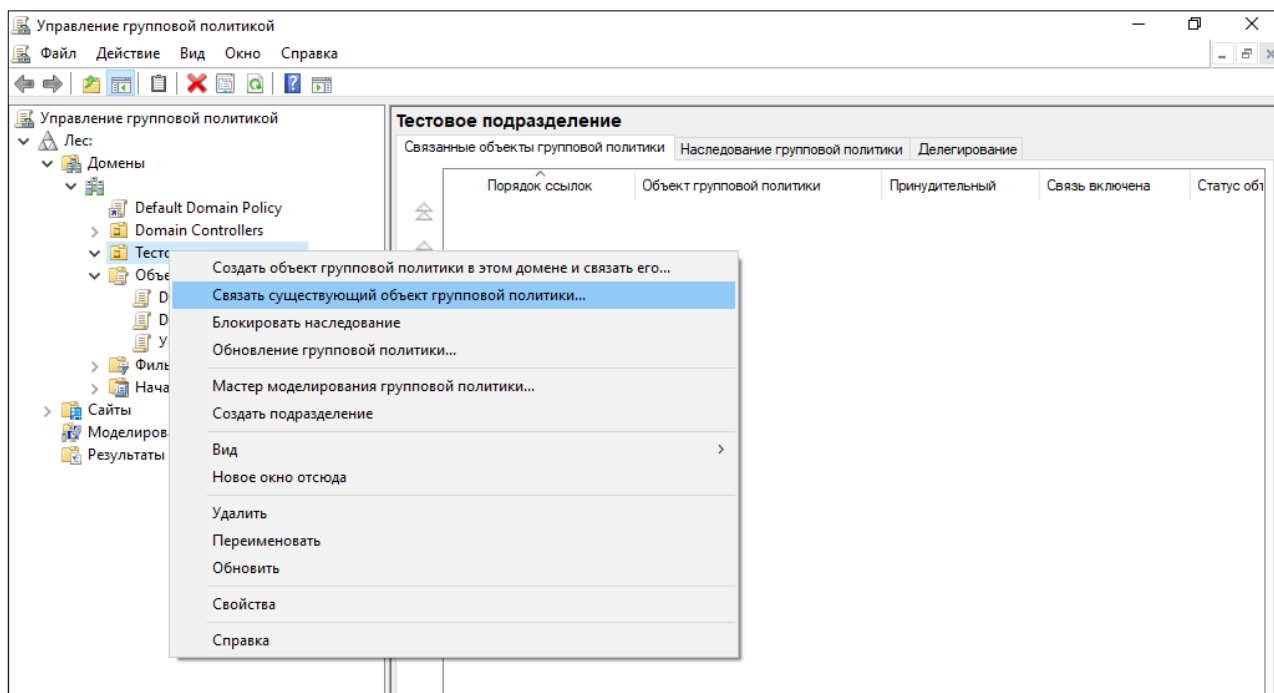


Рисунок 25 – Создание связи Organizational Unit с объектом групповой политики

5. В окне выбора объекта групповой политики выберите **"Установка клиента"**.

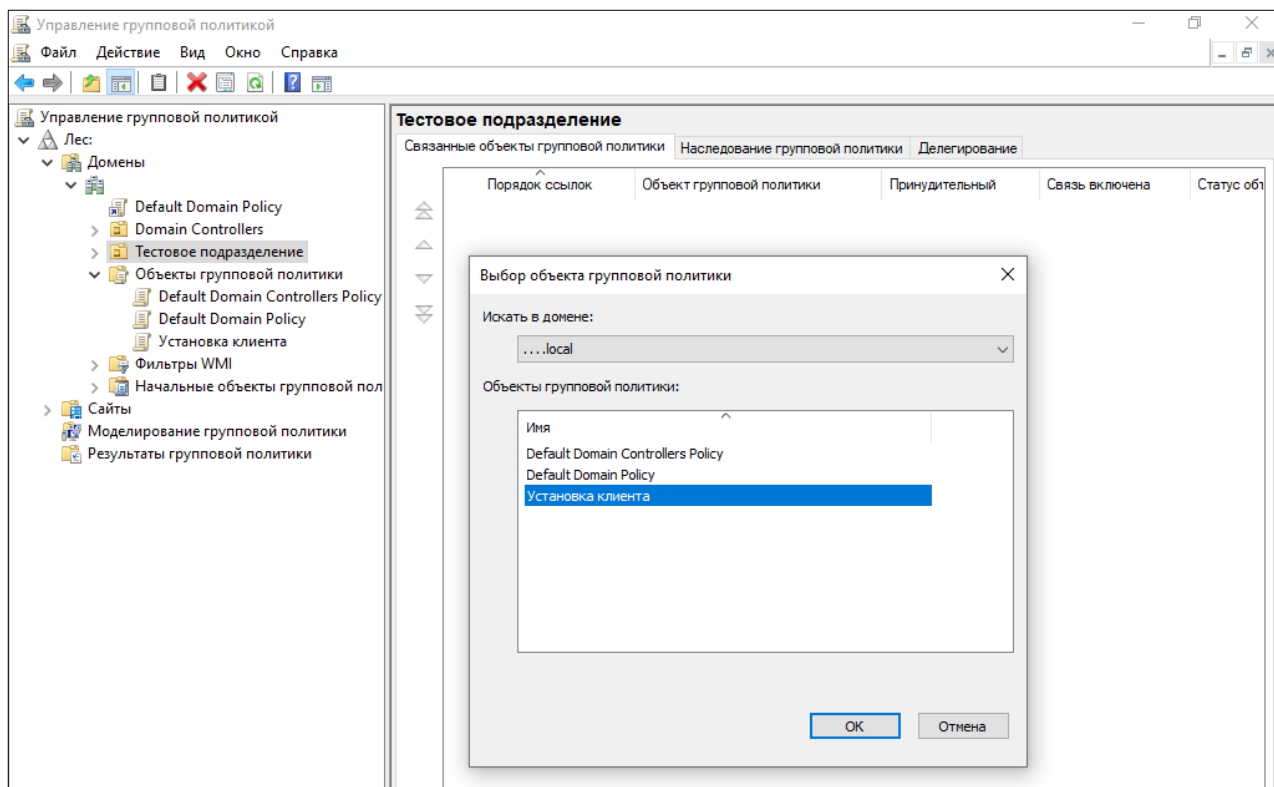


Рисунок 26 – Выбор объекта групповой политики

6. В списке объектов групповой политики домена кликните правой кнопкой мыши на объекте **"Установка клиента"**, выберите пункт **Изменить** (Рисунок 27).

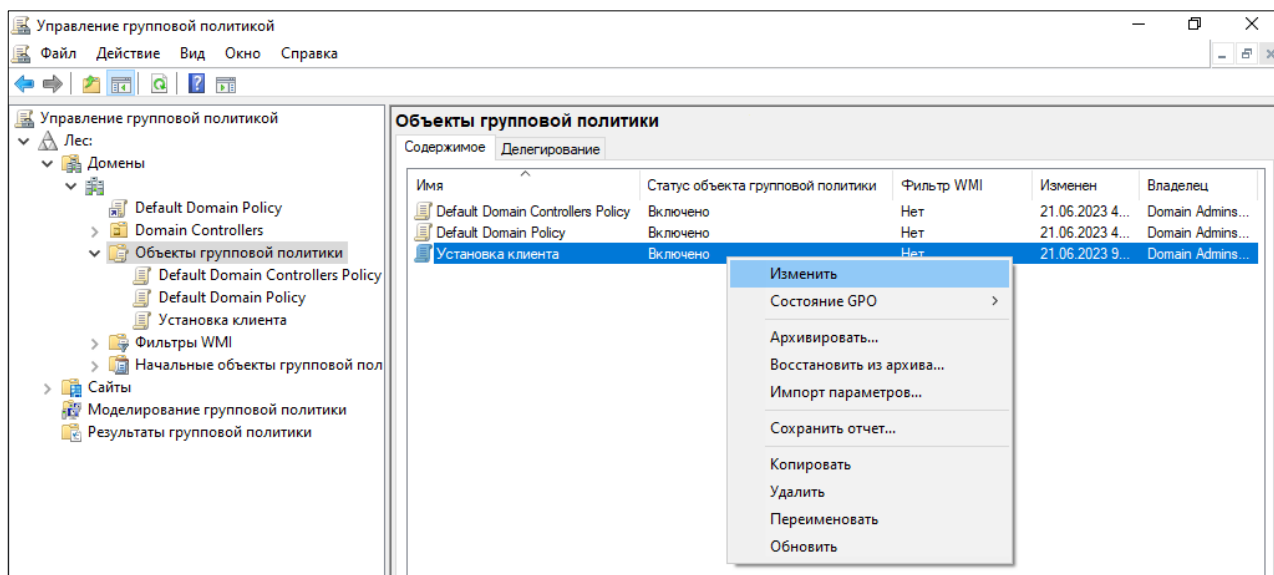


Рисунок 27 – Изменение объекта групповой политики

7. В окне редактора управления групповыми политиками в узле Конфигурация компьютера -> Конфигурация Windows выберите пункт **Сценарии** (Рисунок 28).
8. В окне со списком сценариев кликните правой кнопкой мыши на сценарии **"Автозагрузка"**, выберите пункт **Свойства** (Рисунок 28).

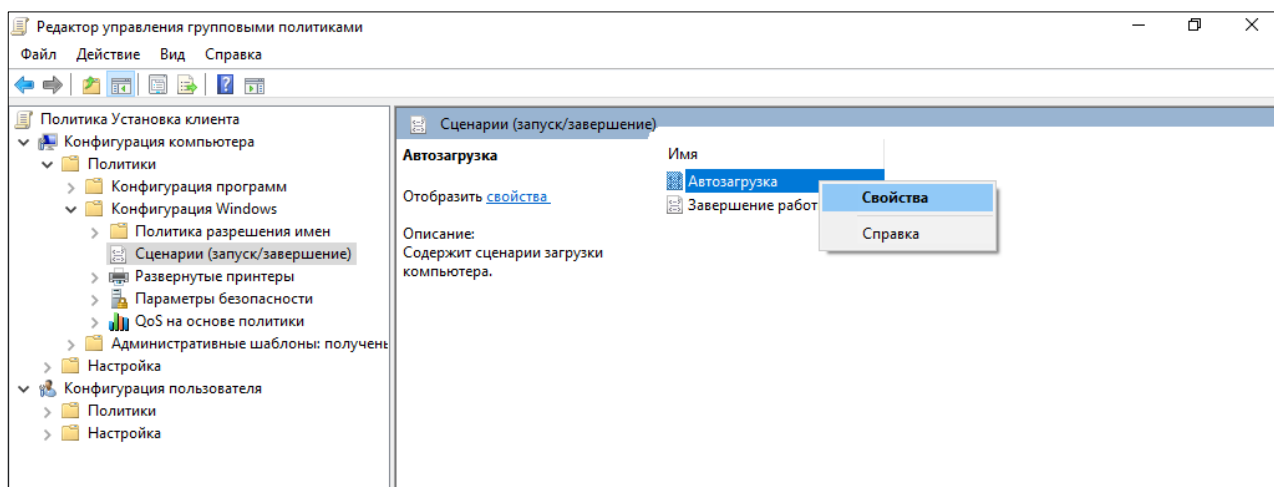


Рисунок 28 – Изменение свойств сценария автозагрузки

9. В открывшемся окне свойств сценария автозагрузки на вкладке **Сценарии** нажмите кнопку **Показать файлы** (Рисунок 29).

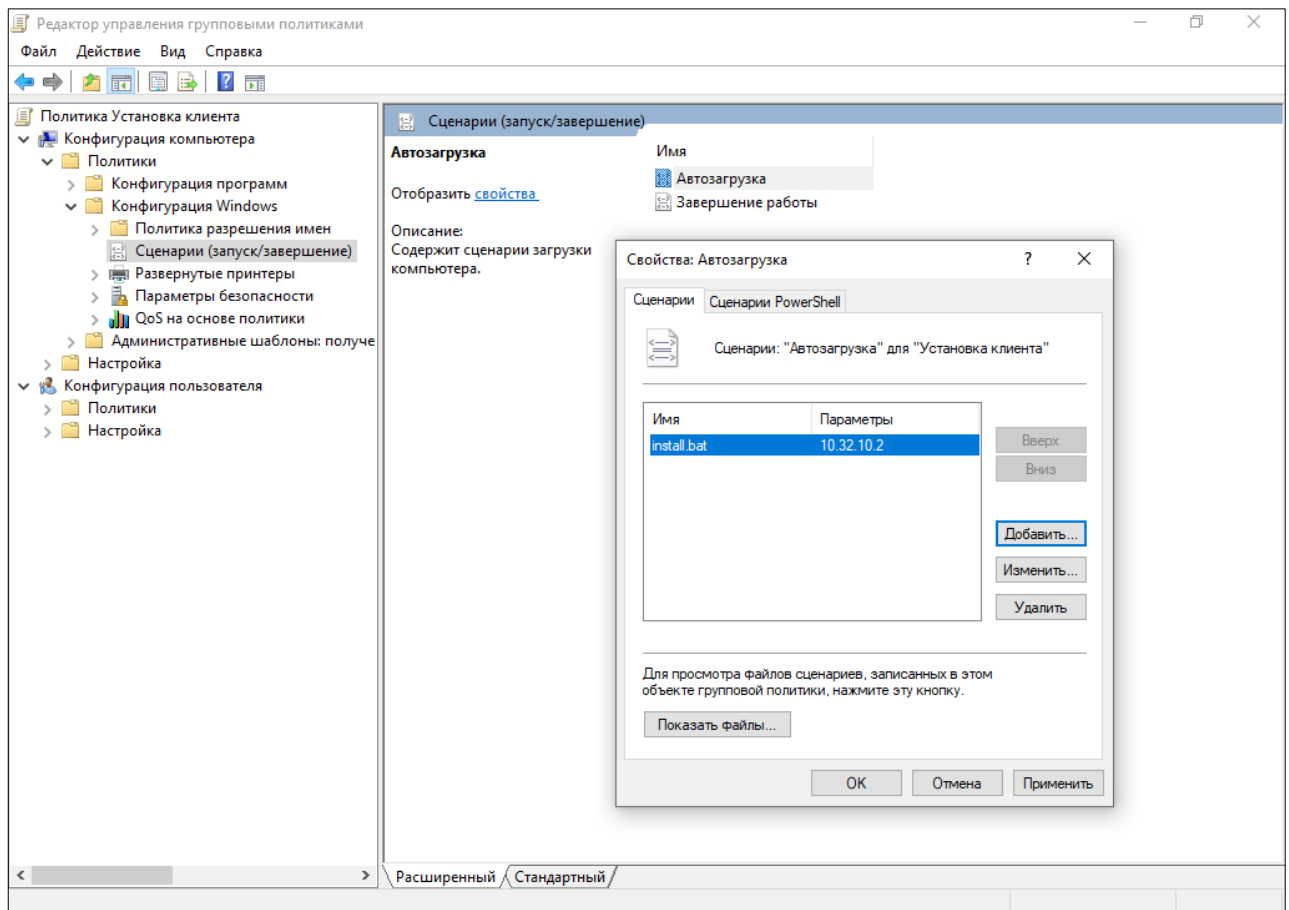


Рисунок 29 – Свойства сценария автозагрузки

10. В папку сценария автозагрузки, открывшуюся в проводнике Windows, скопируйте файл `install.bat` и `.exe`-файлы инсталляторов для ОС Windows 32- и 64-разрядной архитектуры (Рисунок 30).

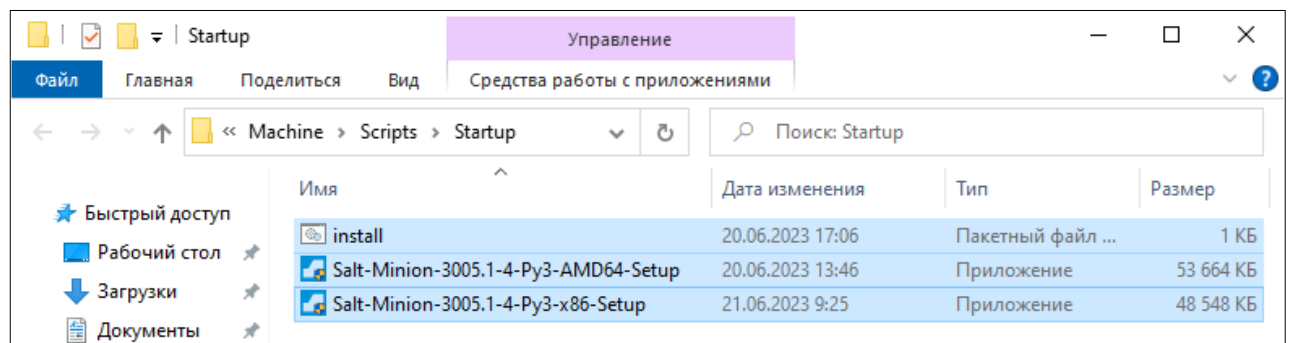


Рисунок 30 – Копирование файлов в папку сценария автозагрузки

11. Закройте проводник и нажмите кнопку **Добавить** на вкладке **Сценарии** окна свойств сценария автозагрузки (Рисунок 29).
12. Укажите имя `install.bat` и значение параметра — выделенный IP-адрес сервера — и нажмите **OK** (Рисунок 31).

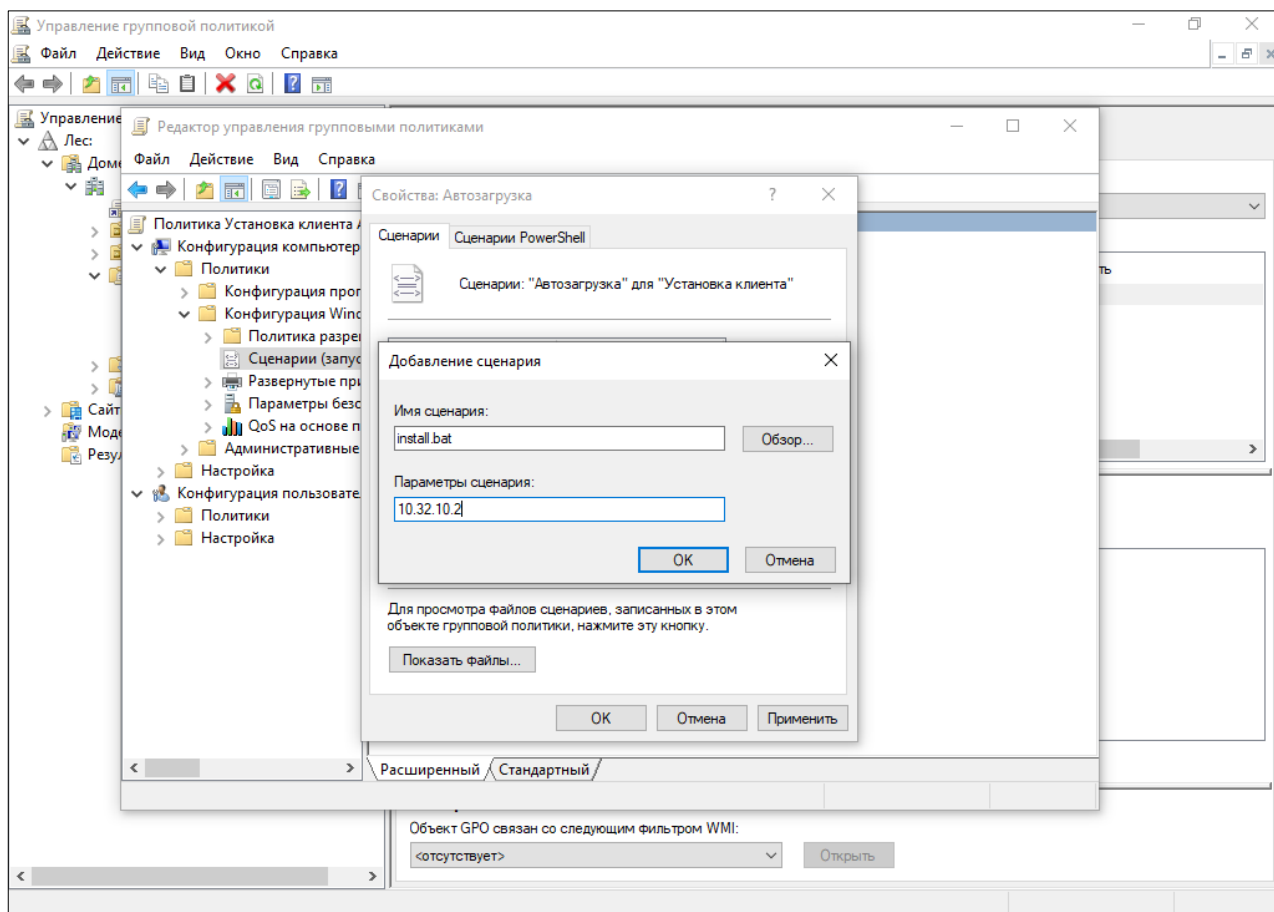


Рисунок 31 – Ввод параметров сценария автозагрузки

3 Процедуры по обслуживанию Программного комплекса

В таблице 6 приведён перечень контейнеров Программного комплекса, на который делаются ссылки в тексте текущего раздела.

Таблица 6 – Контейнеры Docker

Контейнер Docker	Описание
saltbox-mongo-1	СУБД MongoDB. Используется бэкендом Salt.Box. В БД хранятся такие сущности, как клиенты, коллекции и т. п.
saltbox-sshfs-1	Сервер SSH. Используется для монтирования файловой системы sshfs на миньонах
saltbox-opa-1	Сервис Open Policy Agent. ОПА используется для создания политик доступа к различным компонентам веб-приложения
saltbox-keycloak-db-1	Сервис KeyCloak. Обеспечивает централизованную аутентификацию в веб-интерфейсе Salt.Box
saltbox-saltbox-frontend-1	Пользовательский интерфейс Salt.Box (фронтенд)
saltbox-make-redis-salt-certs-1	Модуль, предоставляющий сервис выпуска сертификатов для аутентификации на основе сертификатов Redis
saltbox-saltbox-core-tasks-watcher-1	Модуль, ожидающий публикации задач в шине обмена SaltStack
saltbox-saltbox-core-taskiq-scheduler-1	Модуль, отвечающий за работу с распределённой асинхронной очередью задач
saltbox-saltbox-core-taskiq-worker-1	Процесс taskiq worker. Отвечает за получение, отправку, выполнение задач из очереди
saltbox-rabbitmq-1	Сервис RabbitMQ — брокер сообщений. Отвечает за сбор данных из нескольких источников и отправляет их в сервисы для дальнейшей обработки
saltbox-redis-taskiq-1	Сервер Redis, используемый как брокер и хранилище результатов для Taskiq
saltbox-saltbox-core-faststream-redis-1	Сервер Redis, используемый как брокер для Faststream
saltbox-redis-salt-1	Сервер Redis. Используется для хранения результатов выполнения процессов, связанных с SaltStack
saltbox-keycloak-1	Подсистема идентификации и управления доступом Salt.Box
saltbox-saltbox-core-1	Главный серверный компонент Salt.Box (бэкенд)
saltbox-salt-master-1	Сервис salt-master для управления миньонами
saltbox-keycloak-init-1	
saltbox-proxy-1	Nginx reverse proxy. Перенаправляет запросы клиента на внутренние сервисы в контейнерах

В таблице 7 приведён перечень томов Программного комплекса, на который делаются ссылки в тексте текущего раздела.

Таблица 7 – Тома Docker

Том Docker	Минимальный размер	Описание
<256-битное имя>	< 1 MB	Данные сервиса RabbitMQ
saltbox_saltbox_core_var	< 1 MB	Ключи GPG
saltbox_keycloak-db_data	67 MB	Данные сервиса KeyCloak
saltbox_salt_master_pki	< 1 MB	Хранилище ключей сервиса salt-master
saltbox_redis_taskiq_data	< 1 MB	Данные сервиса taskiq
saltbox_mongo_config	< 1 MB	Конфигурационные данные СУБД MongoDB
saltbox_mongo_data	< 311 MB	Данные СУБД MongoDB
saltbox_redis_salt_data	< 1 MB	Данные сервера Redis
saltbox_redis_salt_certs	< 1 MB	Хранилище сертификатов сервера Redis
saltbox_salt_master_saltbox_var	< 1 MB	
saltbox_salt_repos	< 1 MB	Схемы JSON для различных функций SaltStack

3.1 Ежедневные процедуры

3.1.1 Мониторинг контейнеров Docker

docker stats — это встроенная команда Docker, которая отображает в реальном времени статистику по использованию ресурсов запущенных контейнеров. Команда отображает такие метрики, как использование CPU, памяти, сети и диска, что позволяет быстро оценить нагрузку и выявить возможные проблемы.

3.1.2 Синтаксис команды **docker stats**

```
| docker stats [OPTIONS] [CONTAINER ...]
```

Здесь:

- **`CONTAINER`** — имя или идентификатор контейнера, статистику которого необходимо просмотреть. Допускается указывать список контейнеров через пробел. Если список контейнеров не задан, будет выведена таблица с текущими метриками по всем запущенным контейнерам;
- **`OPTIONS`** — дополнительные параметры для настройки вывода статистики.

В выводе команды отображаются следующие параметры и метрики:

- **CONTAINER ID / NAME** — идентификатор или имя контейнера
- **CPU %** — процент использования процессора
- **MEM USAGE / LIMIT** — объём использованной и доступной памяти
- **MEM %** — процент использования памяти
- **NET I/O** — объём входящих и исходящих данных по сети
- **BLOCK I/O** — объём данных, прочитанных и записанных на диск

Перечень контейнеров, для которых применима данная команда, приведён в таблице 6.

Параметры команды **docker logs**:

7. Показать все контейнеры:

-a, --all

По умолчанию отображаются только запущенные контейнеры.

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	52
Дата созд.	10.07.2025		Страниц	70

8. Форматирование вывода:

--format string

Строка шаблона форматирования:

`table`: вывести в формате таблицы с заголовками столбцов (по умолчанию)**`table TEMPLATE`:** вывести в формате таблицы с использованием заданного шаблона Go**`json`:** вывести в формате JSON**`TEMPLATE`:** форматировать вывод с использованием заданного шаблона Go. Подробнее о форматировании вывода с помощью шаблонов см. на странице <https://docs.docker.com/go/formatting/>

9. Отключить потоковую передачу статистики и отображать только первый результат

--no-stream

10. Не отсекай вывод

--no-trunc

3.2 Процедуры, проводимые при необходимости

3.2.1 Запуск и остановка контейнеров

Запуск контейнера

```
| docker start <container>
```

Перезапуск контейнера

```
| docker restart <container>
```

Пауза (приостановка всех процессов контейнера)

```
| docker pause <container>
```

Снятие паузы

```
| docker unpause <container>
```

Остановка контейнера

```
| docker stop <container>
```

Отправка SIGKILL (завершающего сигнала)

```
| docker kill <container>
```

Отправка другого сигнала

```
| docker kill -s HUP <container>
```

Подключение к существующему контейнеру

```
| docker attach <container>
```

3.2.2 Получение информации о контейнерах

Вывод подробной информации о контейнере

```
| docker inspect <container>
```

События контейнера

```
| docker events <container>
```

Публичные порты контейнера

```
| docker port <container>
```

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	54
Дата созд.	10.07.2025		Страниц	70

Список процессов, выполняющихся в контейнере

```
| docker top <container>
```

Перечень изменений в файлах и каталогах файловой системы контейнера

```
| docker diff <container>
```

3.2.3 Очистка Docker

Удаление работающего контейнера

```
| docker rm <container>
```

Удаление контейнера и его тома (volume)

```
| docker rm -v <container>
```

Удаление всех контейнеров со статусом exited

```
| docker rm $(docker ps -a -f status=exited -q)
```

Удаление всех остановленных контейнеров

```
| docker container prune
```

Удаление образа

```
| docker rmi <image>
```

Удаление неиспользуемых (dangling) образов

```
| docker image prune
```

Удаление всех образов

```
| docker rmi $(docker images -a -q)
```

Остановка и удаление всех контейнеров

```
| docker stop $(docker ps -a -q) && docker rm $(docker ps -a -q)
```

Удаление неиспользуемых (dangling) томов

```
| docker volume prune
```

Удаление всех неиспользуемых объектов

```
| docker system prune [--volumes]
```

Подробная информация о назначении и синтаксисе команд Docker размещена на официальном ресурсе проекта Docker:

<https://docs.docker.com/>

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	56
Дата созд.	10.07.2025		Страниц	70

3.3 Просмотр логов сервисов Программного комплекса

Для просмотра логов приложений, контейнеризованных с помощью ПО Docker, используется команда **docker logs**.

Команда **docker logs** выводит логи указанного контейнера. Эти логи включают все сообщения, записанные процессами, выполняемыми внутри контейнера, в стандартные потоки вывода (**stdout**) и ошибок (**stderr**).

3.3.1 Синтаксис команды **docker logs**

```
docker logs [OPTIONS] CONTAINER
```

Здесь:

- **CONTAINER** — имя или идентификатор контейнера, логи которого необходимо просмотреть;
- **OPTIONS** — дополнительные параметры для настройки вывода логов.

Перечень томов, для которых применима данная команда, приведён в таблице 7.

Параметры команды **docker logs**:

1. Просмотр логов в реальном времени:

Для просмотра логов в реальном времени используется параметр **-f (follow)**.

Логи отображаются по мере их появления, аналогично команде **tail -f**.

```
docker logs -f <container>
```

2. Ограничение количества выводимых строк:

Количество строк, выводимых командой **docker logs**, можно ограничить с помощью параметра **--tail**, например:

```
docker logs --tail 100 <container>
```

Будут выведены последние 100 строк логов контейнера **<container>**.

3. Вывод логов с определенного времени:

Параметр **--since** позволяет выводить логи, начиная с указанного времени.

Время может быть указано в формате временной метки (например, **2025-05-26T00:00:00**) или относительном формате (например, **1h** для логов за последний час).

```
docker logs --since "2025-05-26T00:00:00" <container>
```

или

```
docker logs --since 1h <container>
```

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	57
Дата созд.	10.07.2025		Страниц	70

4. Вывод логов до определенного времени:

Параметр `--until` позволяет выводить логи до указанного времени.

Формат времени аналогичен параметру `--since`.

```
| docker logs --until "2025-05-26T01:00:00" <container>
```

или

```
| docker logs --until 10m <container>
```

5. Разделение потоков `stdout` и `stderr`:

По умолчанию команда `docker logs` объединяет в своём выводе стандартный поток вывода (`stdout`) и стандартный поток ошибок (`stderr`).

Используя параметры `--stdout` и `--stderr`, можно указать, какой поток требуется вывести, например:

```
| docker logs --stdout <container>
```

3.4 Взаимодействие с системой

3.4.1 Резервное копирование томов Docker

Предпочтительным способом резервного копирования данных приложений, контейнеризованных с помощью ПО Docker, является создание резервных копий томов Docker (Docker volumes). Данные тома Docker, такие как базы данных или файлы, хранятся вне контейнера.

3.4.1.1 Создание резервных копий томов

Для создания резервной копии тома Docker выполните следующие действия:

1. Найдите название нужного тома командой:

```
docker volume ls
```

Перечень томов, для которых необходимо сделать резервные копии, приведён в таблице 7.

2. Создайте временный контейнер для копирования данных тома:

```
docker run --rm -v your_volume_name:/volume -v $(pwd):/backup busybox tar czf /backup/volume_backup.tar.gz -C /volume .
```

Здесь:

- `your\_volume\_name` — название тома;
- `\$(pwd)` — текущий каталог, где должен быть сохранён архив;
- `busybox` — минимальный контейнер для выполнения команды.

3.4.1.2 Восстановление томов из резервных копий

Для восстановления тома с данными из резервной копии выполните команду:

```
docker run --rm -v your_volume_name:/volume -v $(pwd):/backup busybox tar xzf /backup/volume_backup.tar.gz -C /volume
```

Здесь:

- `your\_volume\_name` — название тома;
- `\$(pwd)` — текущий каталог, где сохранён архив;
- `busybox` — минимальный контейнер для выполнения команды.

3.4.2 Резервное копирование базы данных KeyCloak

Данные подсистемы управления доступом KeyCloak хранятся на томе `saltbox_keycloak-db_data` контейнера `saltbox-keycloak-db-1`.

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	59
Дата созд.	10.07.2025		Страниц	70

Предпочтительным способом резервного копирования базы данных, реализованной в СУБД PostgreSQL, является использование утилиты **pg_basebackup** для создания «физической копии» базы данных.

Утилитой **pg_basebackup** можно выполнять резервное копирование работающего кластера баз данных PostgreSQL. Результирующий бинарный файл можно использовать для репликации или восстановления на определенный момент в прошлом. Утилита создает резервную копию всего экземпляра базы данных и не дает возможности создавать слепки данных отдельных сущностей. Копия создаётся через обычное подключение к PostgreSQL при помощи протокола репликации с полномочиями суперпользователя или с правом REPLICATION.

3.4.2.1 Создание резервной копии базы данных PostgreSQL

Полная резервная копия создается следующей командой:

```
pg_basebackup -h <postgresql-server-address> -p 5432 -U postgres -D /backup/<backup-postgresql-folder-name> -Ft -z -Xs -P
```

Чтобы выполнить команду **pg_basebackup** с удаленного сервера, нужно разрешить это подключение. Для этого в файл **pg_hba.conf** необходимо добавить строку:

```
host replication all <адрес сервера>/32 md5
```

После внесения изменений в **pg_hba.conf** необходимо перезапустить PostgreSQL для их применения.

Для непрерывного (PiTR) архивирования PostgreSQL требуется включить архивирование **WAL**. Для этого в конфигурационном файле **/var/lib/postgresql/data/postgresql.conf** необходимо настроить следующие параметры:

```
wal_level = replica
archive_mode = on
archive_command = 'test ! -f /backup/<backup-postgresql-folder-name>/postgresql_archives/%f && cp %p /backup/<backup-postgresql-folder-name>/postgresql_archives/%f'
archive_timeout = 300
```

Там же необходимо установить значение параметра **data_directory**, если оно не определено. Иначе модуль резервного копирования не сможет определить местоположение файлов СУБД.

После внесения изменений необходимо перезапустить PostgreSQL при помощи команды:

```
service postgresql restart
```

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	60
Дата созд.	10.07.2025		Страниц	70

К СВЕДЕНИЮ

В базовой конфигурации Программного комплекса порт **5432/TCP** не опубликован на Docker host, поэтому для работы утилиты **pg_basebackup** необходимо в файле **compose.yaml** указать:

```
services:
  keycloak-db:
    ports:
      - '5432:5432'
```

Параметры утилиты pg_basebackup:

-d <строка_подключения>, --dbname=строка_подключения
определение базы данных в виде строки для подключения.

-h <сервер>, --host=сервер
имя сервера с базой данных.

-p <порт>, --port=порт
TCP-порт, через который СУБД принимает подключения.

-s <интервал>, --status-interval=интервал
количество секунд между отправками статусных пакетов.

-U <пользователь>, --username=пользователь
установка имени пользователя для подключения.

-w, --no-password
отключение запроса на ввод пароля.

-W, --password
принудительный запрос пароля.

-V, --version
вывод версии утилиты pg_basebackup.

-, --help
вывод справки по утилите pg_basebackup.

-D каталог, --pgdata=каталог
директория записи данных.

-F <формат>, --format=формат
формат вывода. Допустимые варианты:

p, plain
значение для записи выводимых данных в текстовые файлы;

t, tar
значение, указывающее на необходимость записи в целевую директорию в формате tar.

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	61
Дата созд.	10.07.2025		Страниц	70

-r <скорость_передачи>, --max-rate=скорость_передачи

предельная скорость передачи данных в Кб/с.

-R, --write-recovery-conf

записать минимальный файл recovery.conf в директорию вывода.

-S <имя_слота>, --slot=имя_слота

задание слота репликации при использовании WAL в режиме потоковой передачи.

-T <каталог_1=каталог_2>, --tablespace-mapping=каталог_1=каталог_2

активация миграции табличного пространства из одного каталога в другой каталог при копировании.

--xlogdir=каталог_xlog

директория хранения журналов транзакций.

-X <метод>, --xlog-method=метод

активация вывода файлов журналов транзакций WAL в резервную копию на основе следующих методов:

f, fetch

включение режима сбора файлов журналов транзакций при окончании процесса копирования;

s, stream

включение передачи журнала транзакций в процессе создания резервной копии.

-z, --gzip

активация gzip-сжатия результирующего tar-файла.

-Z <уровень>, --compress=уровень

определение уровня сжатия механизмом gzip.

-c, --checkpoint=fast|spread

активация режима реперных точек.

-l <метка>, --label=метка

установка метки резервной копии.

-P, --progress

активация в вывод отчета о прогрессе.

-v, --verbose

режим подробного логирования.

3.4.2.2 Восстановление базы данных PostgreSQL из резервной копии

Для восстановления или запуска на новом месте резервной копии, полученной в результате работы **pg_basebackup**, необходимо остановить сервер PostgreSQL, распаковать и подменить содержимое его директории данных базовой резервной копией и архивом WAL:

```
systemctl stop postgresql
mv /var/lib/postgresql/data /var/lib/postgresql/data.old
mkdir /var/lib/postgresql/data
tar xvf /backup/<backup-postgresql-folder-name>/base.tar.gz -C /var/lib/postgresql/data
tar xvf /backup/<backup-postgresql-folder-name>/pg_wal.tar.gz -C /var/lib/postgresql/data/pg_wal
chown -R postgres:postgres /var/lib/postgresql
chmod -R go-rwx /var/lib/postgresql/
systemctl start postgresql
```

Для восстановления кластера PostgreSQL до состояния, в котором он находился в определённый момент времени, необходимо остановить сервер PostgreSQL, восстановить последнюю полную резервную копию, смонтировать файловую систему с архивными файлами **WAL** и создать файл **/var/lib/postgresql/data/recovery.conf** со следующим содержанием:

```
restore_command = 'cp /backup/<backup-postgresql-folder-name>/postgresql_archives/%f%p'
recovery_target_time = '2025-05-26 11:20:00 UTC'
recovery_target_inclusive = false
```

Затем запустить сервер PostgreSQL.

3.4.3 Резервное копирование данных MongoDB

Статические данные Программного комплекса хранятся на томе **saltbox_mongo_data** контейнера **saltbox-mongo-1**.

Предпочтительным способом резервного копирования базы данных, реализованной в СУБД MongoDB, является использование утилиты **mongodump**.

Утилита **mongodump** — это базовый инструмент логического резервного копирования, входящий в состав MongoDB, который создает двоичный экспорт содержимого базы данных или коллекций.

3.4.3.1 Создание резервной копии базы данных MongoDB

Резервная копия создаётся следующей командой:

```
mongodump --uri <mongodbURL> --gzip --quiet --out /backup/<backup-mongodb-folder-name>
```

Организация	ООО «ДТ-Софт»	Руководство администратора	Страница	63
Дата созд.	10.07.2025		Страниц	70

Можно также включить флаг **--oplog** в случае резервного копирования набора реплик. Преимущество этого заключается в том, что выходные данные также будут включать в файл **oplog.bson** записи **oplog**, которые фиксируют данные, возникающие во время операции **mongodump**.

Файл **oplog.bson** создаётся как часть вывода **mongodump**. Файл содержит записи, которые произошли во время операции **mongodump**. Это обеспечивает эффективный моментальный снимок состояния экземпляра **mongod** на момент времени. Эта опция работает только для узлов, поддерживающих **oplog**, т. е. для всех членов набора реплик.

3.4.3.2 Восстановление базы данных MongoDB из резервной копии

Для восстановления базы данных используется утилита **mongorestore**. Восстановление выполняется следующим образом:

```
mongorestore --uri <mongodbURL> --dir /backup/<backup-mongodb-folder-name> --drop --gzip  
--preserveUUID --excludeCollection=head.settings.view
```

Для восстановления базы данных на момент времени применяется **Oplogs**. Скопируйте в каталог для восстановления только файл **oplog.bson**, полученный ранее с помощью команды:

```
mongodump --oplog
```

Запустите **mongorestore** с параметром **--oplogReplay** (убедитесь, что каталог содержит только **oplog.bson**):

```
mongorestore --uri <mongodbURL> --oplogReplay --dir /backup/<backup-mongodb-oplog-  
folder-name>
```

4 Проверка работоспособности компонентов Программного комплекса

4.1 Проверка работоспособности сервера

После развёртывания сервера Salt.Box следует убедиться в том, что:

1. Необходимые образы загружены на хост-систему.

Пример корректного вывода команды `docker image ls` приведён в Приложении А, п. А.1.

2. Созданы необходимые тома.

Пример корректного вывода команды `docker volume ls` приведён в Приложении А, п. А.2.

3. Созданы и запущены контейнеры.

Пример корректного вывода команды `docker compose ls -a` приведён в Приложении А, п. А.3.

Пример корректного вывода команды `docker container ls -a` приведён в Приложении А, п. А.4.

4.2 Проверка работоспособности клиента в ОС Windows

Для выявления проблем в работе клиента Программного комплекса «Salt.Config.Box» в ОС Windows необходимо:

- убедиться, что клиент Программы установлен – в списке установленных программ должна присутствовать запись **salt minion**;
- убедиться, что клиент Программы запущен – при выполнении в командной строке команды `sc query salt-minion` должно отобразиться состояние **«RUNNING»**.

Пример корректного вывода:

```
Имя_службы: salt-minion
Тип        : 10  WIN32_OWN_PROCESS
Состояние   : 4  RUNNING
             (STOPPABLE, NOT_PAUSABLE, ACCEPTS_SHUTDOWN)
Код_выхода_Win32 : 0 (0x0)
Код_выхода_службы : 0 (0x0)
Контрольная_точка : 0x0
Ожидание     : 0x0
```

- убедиться, что есть связь с сервером:
 - При выполнении команды `ping <IP-адрес сервера Salt.Box>` должно выводиться сообщение об отсутствии потерь пакетов (0% потерь)
 - При выполнении команды `salt-call test.ping` должен выдаваться ответ:

```
local:
True
```

- В выводе команды `netstat -an | findstr 450` должна присутствовать строка:
`<IP-адрес сервера>:4505 ESTABLISHED`
 либо
`<IP-адрес сервера>:4506 ESTABLISHED`
- убедиться, что архитектура системы и клиента совпадают, для этого запустить:
`C:\Program Files\Salt Project\Salt\bin\python.exe`
 На экране в 64-разрядных системах должна появиться строка, содержащая текст:
`64 bit (AMD64)`

4.3 Проверка работоспособности клиента в ОС Linux

В Таблице 8 представлен перечень процессов и утилит, выполняемых на рабочей станции, с указанием формируемых ими файлов журналов.

Таблица 8 – Процессы, выполняемые на рабочей станции

№	Процесс	Назначение	Файлы журналов
1	salt-minion	Служба клиента SaltStack	/var/log/salt/minion

В файлах журналов не должно быть сообщений о повторяющихся приоритетных ошибках (уровень «**ERROR**» и выше).

Приложение А. Примеры вывода команд Docker

А.1 Список образов

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
registry.saltbox.pro/saltbox/saltbox-bridge/salt-master	master	6e6a65ae6d5c	14 hours ago	305MB
registry.saltbox.pro/saltbox/saltbox-core	master	dc990973edb6	15 hours ago	486MB
registry.saltbox.pro/saltbox/saltbox-frontend	master	cb046f5104d3	22 hours ago	215MB
saltbox-proxy	latest	0a7a69e50215	9 days ago	208MB
registry.saltbox.pro/saltbox/saltbox-compose/keycloak-init	master	c3ad347807b2	11 days ago	819MB
registry.saltbox.pro/saltbox/saltbox-compose/keycloak	master	cce07c6262d2	11 days ago	808MB
registry.saltbox.pro/saltbox/saltbox-compose/sshfs	master	6d4bcd308df	11 days ago	218MB
registry.saltbox.pro/saltbox/saltbox-compose/opa	master	9f4e528e02fb	11 days ago	82.6MB
registry.saltbox.pro/saltbox/saltbox-compose/redis	master	2cb6940da7ee	11 days ago	60.7MB
mongo	7	8c30f14b362a	2 weeks ago	1.1GB
postgres	16-alpine	3b057e1c2c6d	6 weeks ago	394MB
saltbox-make-redis-salt-certs	latest	f82ed3daf965	6 weeks ago	109MB
rabbitmq	3-management	49895caee5d3	6 months ago	389MB

А.2 Список томов

DRIVER	VOLUME NAME
local	f85f6baa5742b4b1bf7b3b7ca86929361c529aee6e06bff9b1a5208d76469345
local	saltbox_keycloak-db_data
local	saltbox_mongo_config
local	saltbox_mongo_data
local	saltbox_redis_salt_certs
local	saltbox_redis_salt_data
local	saltbox_redis_taskiq_data
local	saltbox_saltbox_core_var
local	saltbox_salt_master_pki

А.3 Список работающих compose-контейнеров

NAME	STATUS	CONFIG FILES
saltbox	exited(2), running(16)	root/saltbox-compose/compose.yaml

Организация	ООО «ИТ-Софт»
Дата созд.	10.07.2025
Руководство администратора	Приложение Б
Страница	67
Страниц	70

A.4 Список работающих контейнеров

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
88dc531cc401	saltbox-proxy	"/docker/entrypoint...."	22 minutes ago	Up 21 minutes	0.0.0.0:80->80/tcp	
saltbox-proxy-1						
9027787dcf67	registry.saltbox.pro/saltbox/saltbox-core:master	"entrypoint.sh salt-..."	22 minutes ago	Up 22 minutes	8000/tcp	
saltbox-saltbox-core-faststream-redis-1						
372acea8af4a	registry.saltbox.pro/saltbox/saltbox-core:master	"/usr/local/bin/uvic..."	22 minutes ago	Up 21 minutes	8000/tcp	
saltbox-saltbox-core-1						
04ccd3ce9c7c	registry.saltbox.pro/saltbox/saltbox-bridge/salt-master:master	"/usr/local/bin/entr..."	22 minutes ago	Up 22 minutes	0.0.0.0:4505-4506->4505-4506/tcp, 8000/tcp	
saltbox-saltmaster-1						
a864aec0556a	registry.saltbox.pro/saltbox/saltbox-core:master	"entrypoint.sh salt-..."	22 minutes ago	Up 22 minutes	8000/tcp	
saltbox-saltbox-core-tasks-watcher-1						
8ed8856d31c1	registry.saltbox.pro/saltbox/saltbox-compose/keycloak-init:master	"/usr/local/bin/crea..."	22 minutes ago	Exited (0) 21 minutes ago		
saltbox-keycloak-init-1						
332bc7787b30	registry.saltbox.pro/saltbox/saltbox-core:master	"/usr/local/bin/task..."	22 minutes ago	Up 21 minutes	8000/tcp	
saltbox-saltbox-core-taskiq-scheduler-1						
140c425ca829	registry.saltbox.pro/saltbox/saltbox-core:master	"/usr/local/bin/task..."	22 minutes ago	Up 21 minutes	8000/tcp	
saltbox-saltbox-core-taskiq-worker-1						
f8b895c79ec2	registry.saltbox.pro/saltbox/saltbox-compose/redis:master	"entrypoint.sh '--sa..."	22 minutes ago	Up 22 minutes (healthy)	127.0.0.1:6379->6379/tcp	
saltbox-redis-salt-1						
05bb59635b38	registry.saltbox.pro/saltbox/saltbox-compose/keycloak:master	"/usr/local/bin/star..."	22 minutes ago	Up 21 minutes (healthy)	8080/tcp	
saltbox-keycloak-1						
b68811fa1e30	postgres:16-alpine	"docker-entrypoint.s..."	22 minutes ago	Up 22 minutes (healthy)	5432/tcp	
saltbox-keycloak-db-1						
e7091bac8749	rabbitmq:3-management	"docker-entrypoint.s..."	22 minutes ago	Up 22 minutes (healthy)	4369/tcp, 5671-5672/tcp, 15671-15672/tcp, 15691-15692/tcp, 25672/tcp	
saltbox-rabbitmq-1						
6673e7cb5a06	mongo:7	"docker-entrypoint.s..."	22 minutes ago	Up 22 minutes	127.0.0.1:27017->27017/tcp	
saltbox-mongo-1						
8cd31f8c27ad	saltbox-make-redis-salt-certs	"entrypoint.sh /root..."	22 minutes ago	Exited (0) 22 minutes ago		
saltbox-make-redis-salt-certs-1						
b1ba6ea04662	registry.saltbox.pro/saltbox/saltbox-compose/redis:master	"entrypoint.sh '--sa..."	22 minutes ago	Up 22 minutes	6379/tcp	
saltbox-redis-taskiq-1						
77199f687853	registry.saltbox.pro/saltbox/saltbox-frontend:master	"/docker/entrypoint...."	22 minutes ago	Up 22 minutes	80/tcp	
saltbox-saltbox-frontend-1						
c727e1df0c0b	registry.saltbox.pro/saltbox/saltbox-compose/sshfs:master	"/usr/local/bin/sshd..."	22 minutes ago	Up 22 minutes	0.0.0.0:1022->1022/tcp	

Организация	ООО «ИТ-Софт»	Руководство администратора Приложение Б		Страница	69
Дата созд.	10.07.2025			Страниц	70

saltbox-sshfs-1

53f1daabbabd registry.saltbox.pro/saltbox/saltbox-compose/opa:master

"/opa run --server -..." 22 minutes ago

Up 22 minutes

0.0.0.0:8181->8181/tcp

saltbox-opa-1

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ

Salt.Box, Программный комплекс, ПК	программа для ЭВМ «Программный комплекс «Salt.Config.Box»» (Свидетельство о государственной регистрации программы для ЭВМ № 2025617606, дата государственной регистрации в Реестре программ для ЭВМ 27 марта 2025 г.)
Salt.Box Bridge	компонент Salt.Box, устанавливаемый на серверах Salt Master, предназначен для подключения серверов Salt Master к ПК с целью масштабирования решения
Salt Master	серверный компонент Системы управления конфигурациями SaltStack, подключаемый к ПК с помощью компонента Salt.Box Bridge. Через серверы Salt Master ПК осуществляет управление программными конфигурациями Клиентов ПК
Salt Minion, Клиент ПК	клиентский компонент Системы управления конфигурациями SaltStack – программа, обеспечивающая подключение клиентского устройства к ПК
БД	База данных
Клиентское устройство, Управляемое устройство	физический или виртуальный компьютер/ сервер, на котором установлен и используется Клиент ПК
ОС	Операционная система
ПО	Программное обеспечение
Расширение Salt.Box	модуль или подсистема, расширяющая функциональность ПК, имеющая собственный стандартизованный для ПК API
СУБД	Система управления базами данных
Центр управления	набор компонентов ПК в составе: ядра ПК, подсистемы единого входа, подсистемы аутентификации и авторизации, расширений Salt.Box (при наличии), компонентов общего назначения (веб- приложение, API-шлюз, обратный прокси-сервер). В состав Центра управления не входит компонент Salt.Box Bridge, устанавливаемый на подключаемые к ПК сервера Salt Master