TERMIDESK
CONNECT

Версия v1.3, 2026-04-27

Оглавление

СЛЕТ.101001-01 90 01. РУКОВОДСТВО АДМИНИСТРАТОРА. УСТАНОВКА TERMIDESK CONNECT

ОБЩИЕ СВЕДЕНИЯ

Назначение и область применения

Назначение

Termidesk Connect является многофункциональным сетевым устройством, обеспечивающим балансировку нагрузки для масштабирования инфраструктурных приложений, оптимизацию их работы за счет гибких настроек коммуникационных протоколов, а также георезервирование инфраструктуры.

Область применения

Termidesk Connect является посредником между пользователем и сервером приложений в сетевой инфраструктуре. Получив запрос пользователя на подключение, Termidesk Connect отправляет его на сервер приложений. При этом Termidesk Connect может использоваться как:

- шлюз, для обеспечения единой точки доступа пользователя к приложениям,серверам, сетевым ресурсам из внутренней сети организации;
- балансировщик нагрузки, для распределения запросов пользователей по нескольким серверам и оптимизации использования сетевых ресурсов;
- глобальный балансировщик нагрузки, для обеспечения доступности приложений в географически распределенной инфраструктуре. В этом случае Termidesk Connect распределяет запросы между центрами обработки данных (ЦОД) и направляет их на наиболее производительный или наименее загруженный ЦОД.

Основные характеристики

Termidesk Connect обеспечивает:

- как шлюз:
  - создание и применение правил фильтрации трафика;
  - сбор статистики подключений пользователей;
  - отправку статистики подключений на сторонние сервисы;
- как балансировщик нагрузки:
  - балансировку сетевого трафика на уровнях: транспортном L4 (поддержка Source NAT и протокола TCP), прикладном L7 (поддержка Full Proxy и протоколов HTTP, HTTPS, WS);
  - балансировку сетевого трафика алгоритмами: Round Robin, Least Connection;
  - проверку доступности серверов балансировки несколькими методами: ping, TCP, HTTP/HTTPS, специально подготовленными исполняемыми файлами;
  - сохранение сессии пользователя;
  - локальную отказоустойчивость серверов приложений;
  - географическую отказоусточивость нескольких ЦОД;
  - перенаправление запросов в зависимости от их содержимого;
- управление устройством с использованием командной строки,веб-интерфейса, интерфейсов API и NETCONF.

Требования к уровню подготовки персонала

Для штатной эксплуатации Termidesk Connect требуется следующий персонал:

- системный администратор;
- специалист по обслуживанию комплекса технических средств.

Системный администратор должен иметь опыт администрирования серверов с операционной системой (ОС) Astra Linux Special Edition и знать стек TCP/IP.

Основными обязанностями системного администратора являются:

- установка, настройка и мониторинг работоспособности Termidesk Connect;
- выполнение регламентных работ;
- восстановление работоспособности Termidesk Connect после устранения неисправностей комплекса технических средств.

Специалист по обслуживанию комплекса технических средств должен иметь опыт работы с ОС Astra Linux Special Edition, знать и понимать принципы работы сетей передачи данных, а также владеть знаниями по обслуживанию комплекса технических средств.

Основными обязанностями специалиста по обслуживанию комплекса технических средств являются:

- настройка, модернизация и проверка состояния комплекса технических средств;
- диагностика типовых неисправностей комплекса технических средств;
- настройка сетевых подключений.

Требования к аппаратному и программному обеспечению

Минимальные аппаратные требования виртуальной машины (ВМ), на которой функционирует Termidesk Connect, должны соответствовать следующим:

- оперативная память, не менее 12 ГБ;
- виртуальный процессор (vCPU), не менее 1 шт.

В свойствах ВМ должен быть активирован режим EFI для корректной загрузки Termidesk Connect.

Требования к настройке инфраструктуры

При использовании службы доменных имен (DNS) в Termidesk Connect нужно убедиться, что DNS-сервер доступен и исправно функционирует.

Для обеспечения точной синхронизации времени и корректной работы служб рекомендуется настроить в сетевой инфраструктуре NTP-сервер и добавить его в Termidesk Connect.

Типографские соглашения

Приняты следующие типографские соглашения:

- моноширинный шрифт – используется для выделения фрагментов текста программ, наименований файлов и папок (директорий), наименований пакетов, путей перемещения, строк комментариев, различных программных элементов (объект, класс, тип, переменная, команда, макрос и т. д.), а также вводимого и выводимого текста в режиме командной строки;

- «кавычки» – текст, заключенный в кавычки, используется для обозначения наименований документов, названий компонентов Termidesk, пунктов меню, наименований окон, вкладок, полей, других элементов графического интерфейса, а также вводимого и выводимого текста в режиме графического интерфейса;
- **[квадратные скобки]** – текст, заключенный в квадратные скобки, используется для наименования экранных кнопок;
- **<угловые скобки>** – текст, заключенный в угловые скобки, используется для наименования клавиш клавиатуры.

ПОЛУЧЕНИЕ TERMIDESK CONNECT

Получение Termidesk Connect

Получить Termidesk Connect можно двумя способами:

- заполнив форму запроса на сайте Termidesk: https://termidesk.ru/demonstration/;
- через личный кабинет: https://lk.astra.ru/.

Комплект поставки Termidesk Connect

Termidesk Connect распространяется в следующих форматах:

- образ `.img` для обновления устройства;
- образ диска ВМ формата `.qcow2` для гипервизоров QEMU/KVM;
- упакованный образ ВМ формата `.ova`, представляющий собой виртуальное устройство (Virtual Appliance).

Лицензирование Termidesk Connect

Получение лицензий Termidesk Connect

Termidesk Connect лицензируется по количеству используемых виртуальных процессоров (vCPU).

Получить лицензию Termidesk Connect можно через личный кабинет: https://lk.astra.ru/.

Для получения лицензии в личном кабинете:

- создать подписку, для чего перейти в раздел «Лицензии и сертификаты – Активация лицензий» и нажать экранную кнопку **[Создать подписку]** (см.Создание подписки);

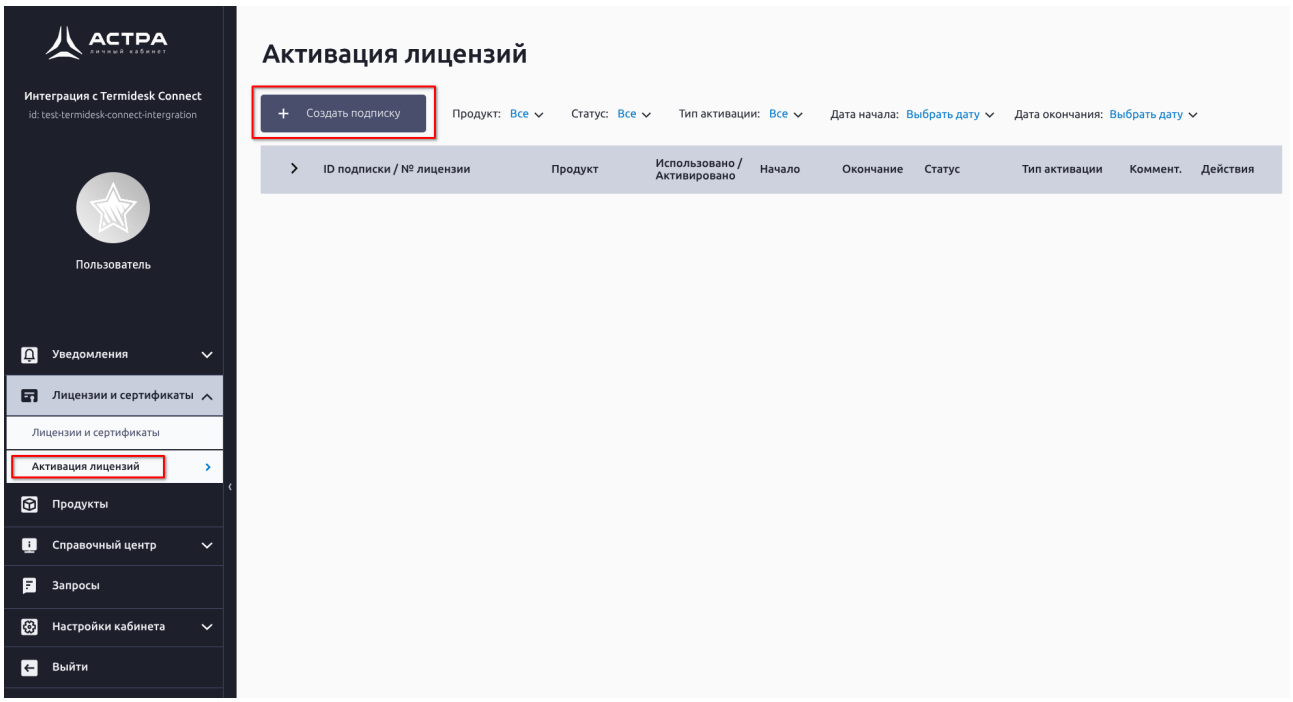

Рисунок 1. Создание подписки

- в открывшемся окне «Конструктор подписок» для определенного номера лицензии в поле «Использовать» указать количество лицензий для активации, заполнить поле «Описание» (опционально), далее нажать экранную кнопку **[Собрать и активировать]** (см.Конструктор подписок);

Одно устройство может использовать только одну лицензию.

Последнее число в номере лицензии соответствует максимально доступному количеству vCPU.

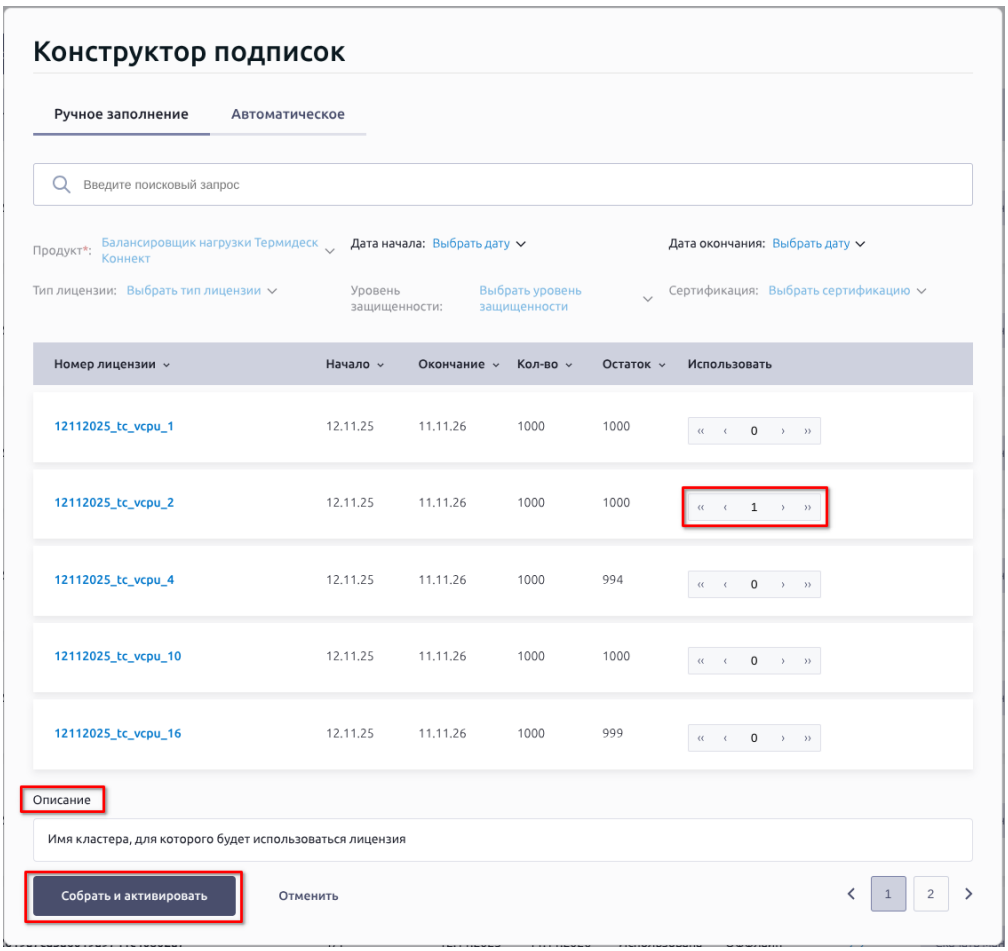

Рисунок 2. Конструктор подписок

- создать манифест, для чего в области созданной подписки нажать экранную кнопку **[Скачать манифест]** (см.Создание манифеста);

Необходимо предварительно получить HWID устройства одним из способов:

- в веб-интерфейсе Termidesk Connect перейти в раздел «Настройки – Система – Лицензия» и скопировать значение в поле «HWID»;
- в интерфейсе командной строки Termidesk Connect выполнить команду `show license attributes` и скопировать значение `HWID`.

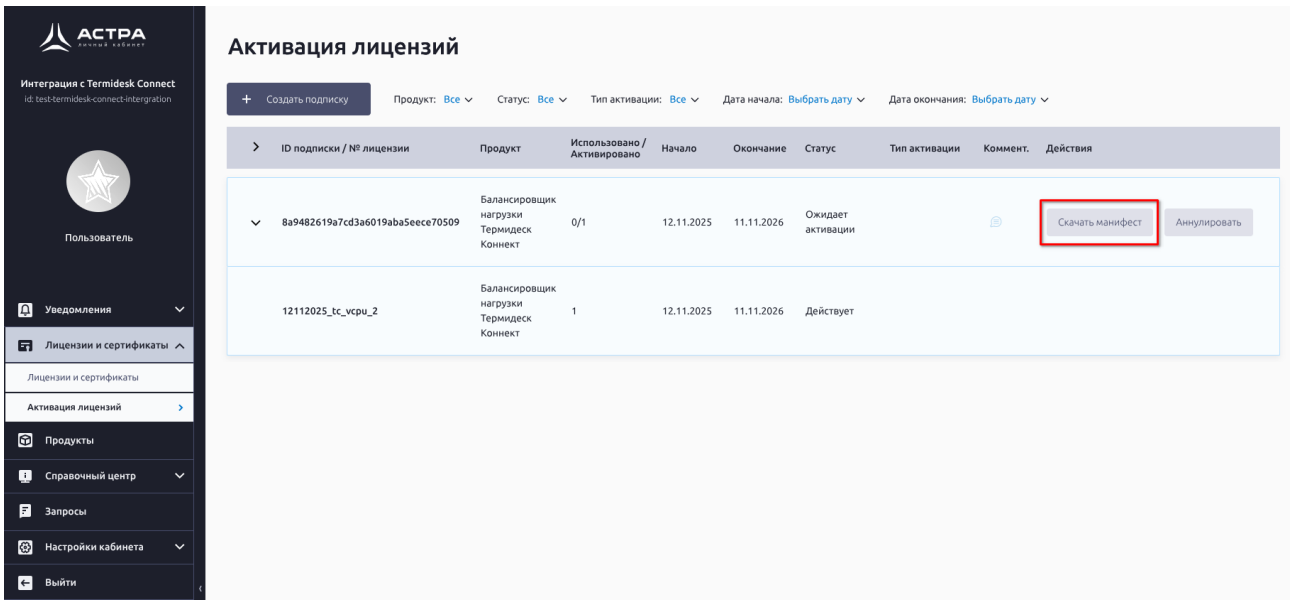

Рисунок 3. Создание манифеста

- в открывшемся окне «Скачать манифест» в поле ввода указать HWID устройства и нажать экранную кнопку **[Скачать]** (см.Скачивание манифеста).

Каждый манифест уникален для одного устройства и не может быть использован на другом. Манифест представляет собой файл в формате `.zip` с данными о лицензии.

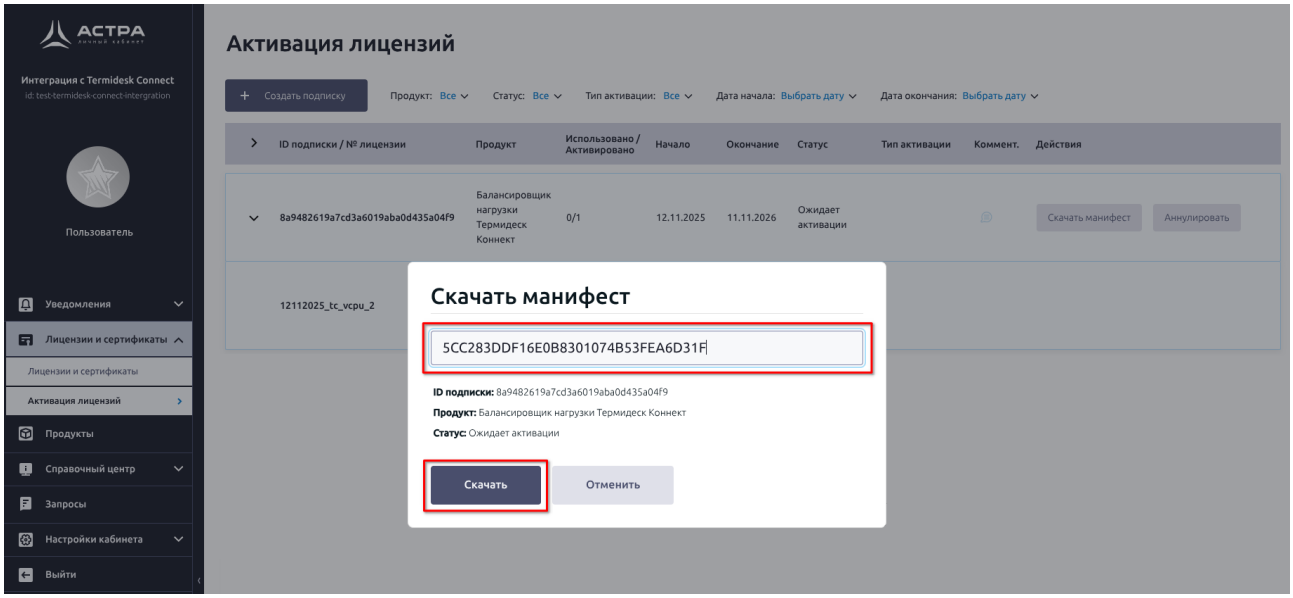

Рисунок 4. Скачивание манифеста

Состав лицензий Termidesk Connect

Состав лицензии	Termidesk Connect	Termidesk Connect Basic
Веб-интерфейс	да	да
Интерфейс командной строки	да	да
Статистика	да	да
Шлюз	да	да

Состав лицензии	Termidesk Connect	Termidesk Connect Basic
Локальная балансировка	да	-
Геобалансировка (GSLB)	да	-
Динамическая маршрутизация и RHI	да	-
Безопасность	да	-
Предаутентификация HTTP	да	-
Отказоустойчивость	да	-
Максимально доступное количество виртуальных процессоров	20	2

Лицензия «Termidesk Connect Basic» подходит для использования в программном комплексе Termidesk (Termidesk VDI).

Для лицензии «Termidesk Connect» доступен полный функционал возможностей, представленных на сайте: **Ключевые возможности Termidesk Connect**.

ПОДГОТОВКА К РАБОТЕ

Порядок загрузки на платформу виртуализации

Для загрузки Termidesk Connect на платформу виртуализации нужно:

- выполнить импорт образа Termidesk Connect на платформу виртуализации;
- создать ВМ, удовлетворяющую требованиям (см. подраздел **Требования к аппаратному обеспечению**). При создании ВМ выбрать в качестве диска импортированный образ;

В свойствах создаваемой ВМ должен быть активирован режим загрузки EFI, и должна использоваться эмуляция IDE.

- выполнить запуск ВМ;

Если ВМ не запускается, необходимо проверить, что в свойствах ВМ выбраны корректные параметры: тип ОС – «Linux», версия – «Other Linux (64-bit)». Параметры могут отличаться от приведенных, в зависимости от платформы виртуализации и ее версии.

- выбрать в меню пункт «Virtual Appliance Termidesk Connect» (по умолчанию) (см. Выбор варианта загрузки) и нажать клавишу **<ENTER>**;

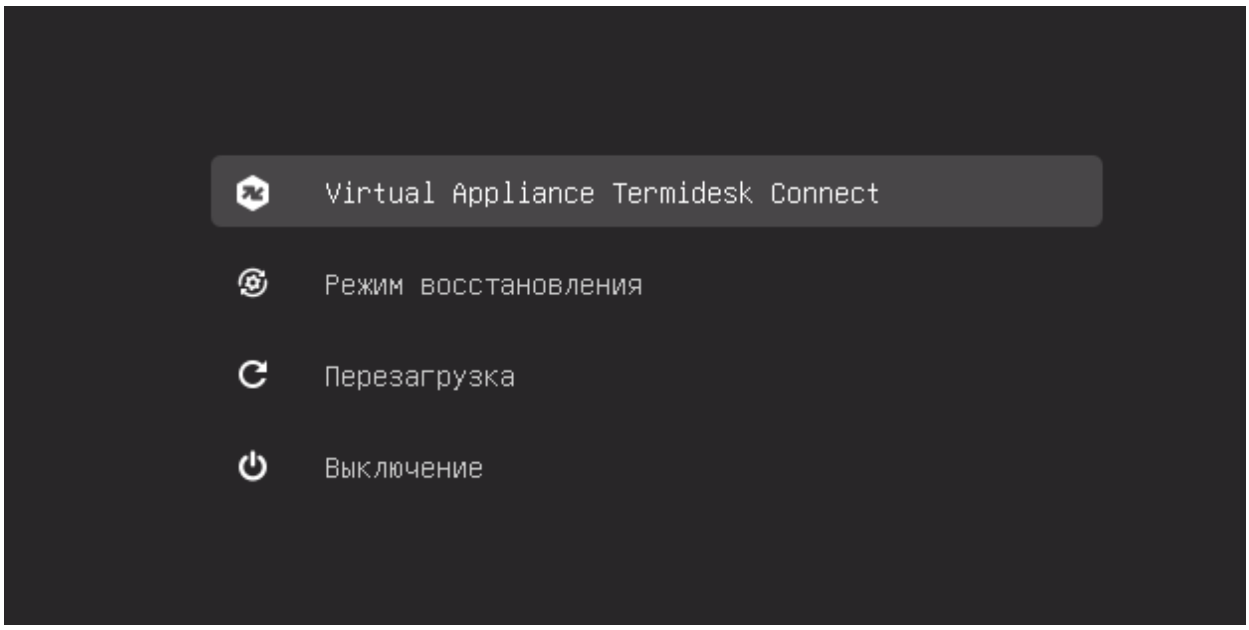

Рисунок 5. Выбор варианта загрузки

- выбрать загружаемый образ (см. Выбор загружаемого образа) и нажать клавишу **<ENTER>**;

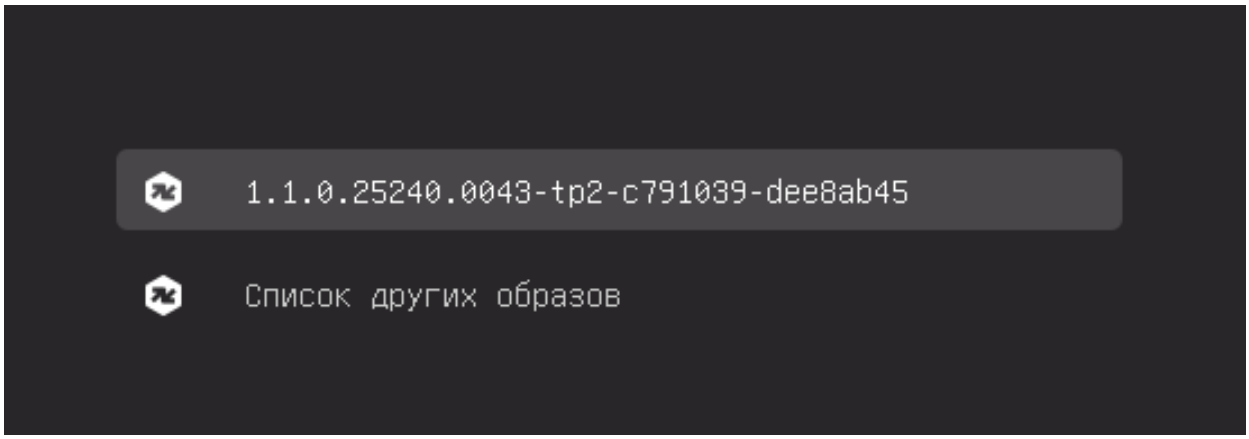

Рисунок 6. Выбор загружаемого образа

- прочитать и принять условия лицензионного соглашения (см. Лицензионное соглашение),переключившись на экранную кнопку **[OK]** и нажав клавишу **<ENTER>**;

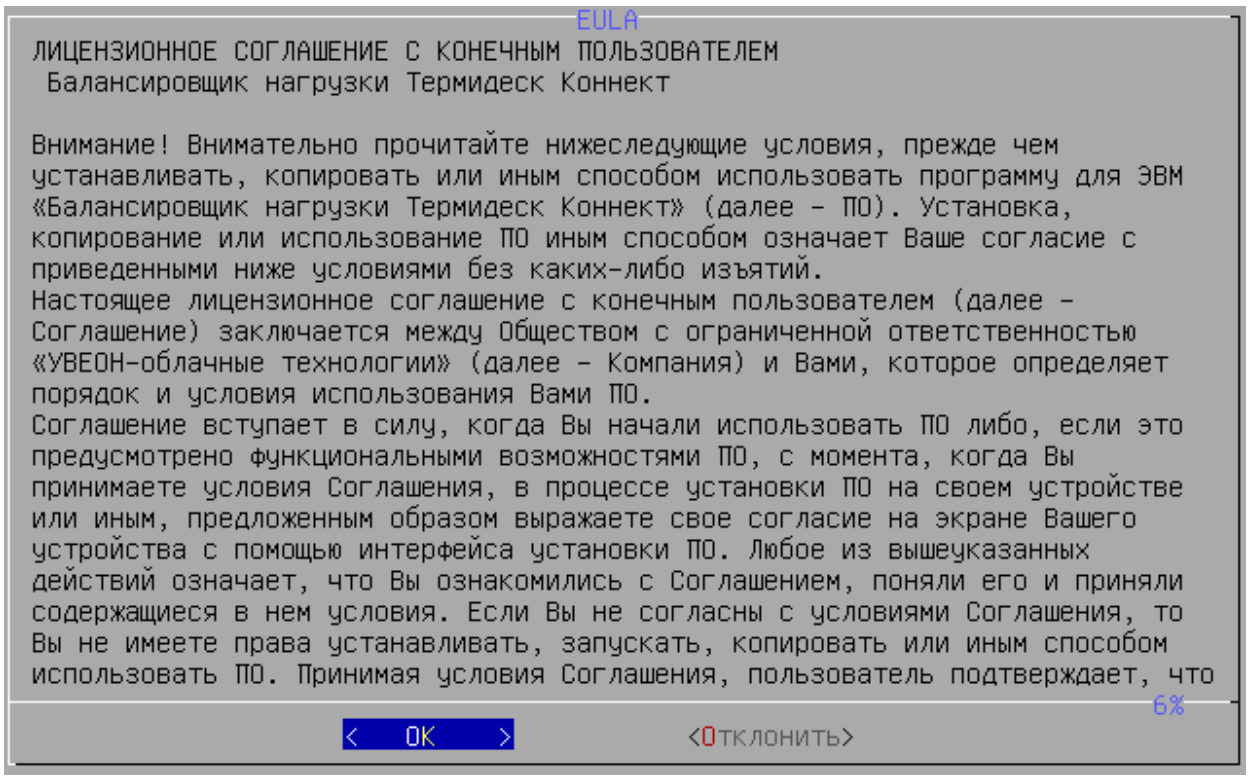

Рисунок 7. Лицензионное соглашение

Переключение между пунктами меню выполняется клавишей **<TAB>**. Подтверждение выбора выполняется клавишами **<ENTER>** или **<SPACE>**.

- дождаться появления информационного сообщения о сертификатах (см. Информационное сообщение) и нажать клавишу **<ENTER>**;

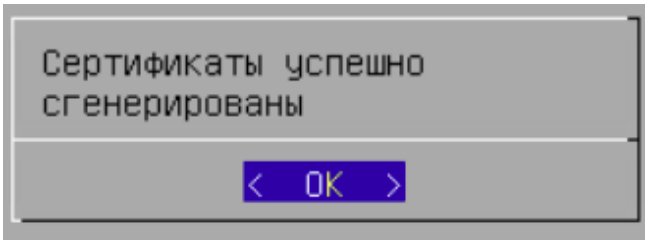

Рисунок 8. Информационное сообщение

- дождаться отображения главного окна Termidesk Connect (см. Главное окно Termidesk Connect). Затем выполнить настройку согласно подразделу **Первоначальная настройка Termidesk Connect**.

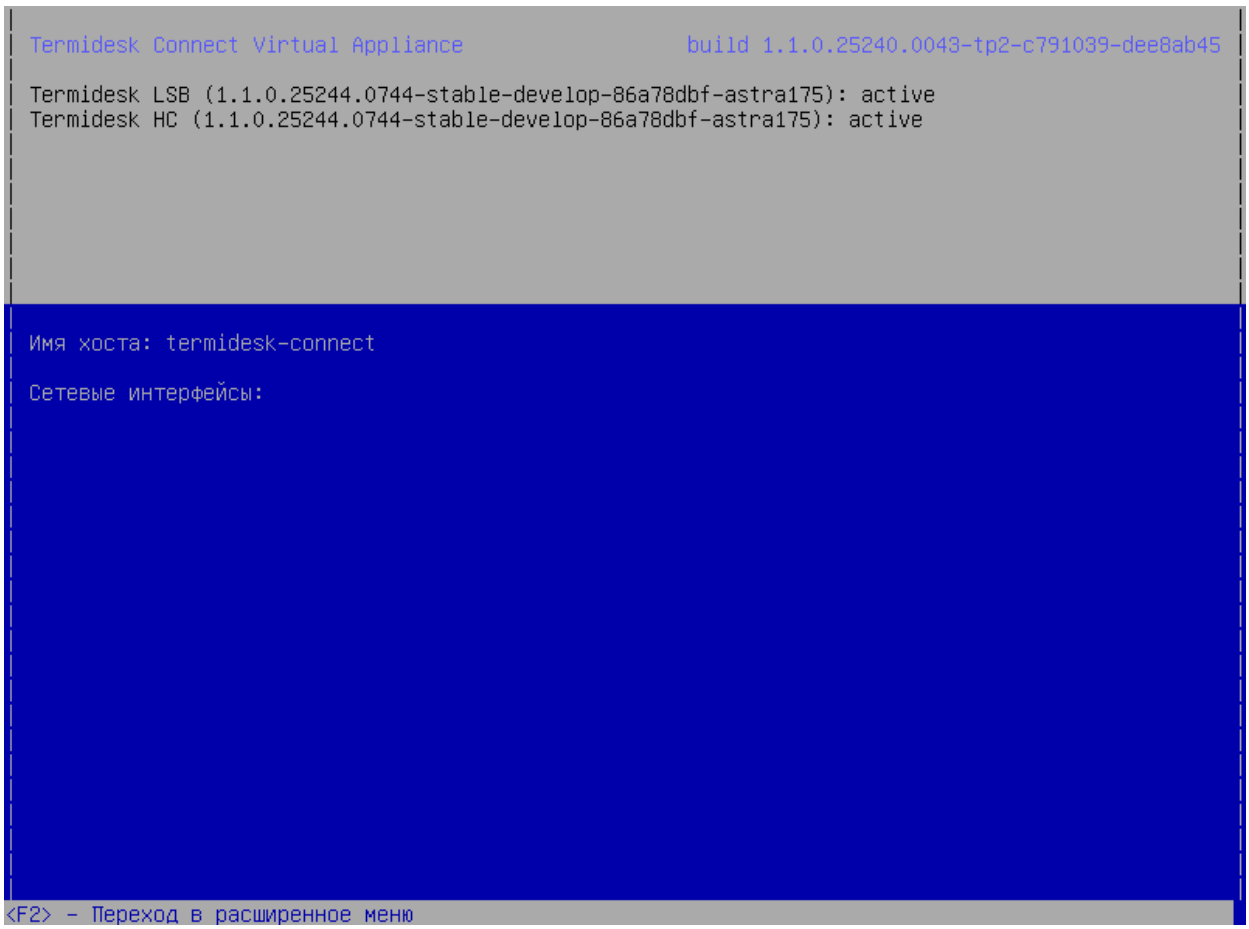

Рисунок 9. Главное окно Termidesk Connect

Загрузка на примере платформы виртуализации VMware vShere

Для загрузки Termidesk Connect на платформу виртуализации нужно:

- выполнить импорт образа диска Termidesk Connect формата `.vmdk` в хранилище платформы виртуализации;
- в веб-интерфейсе VMware vSphere Client выбрать хост и пул ресурсов, в котором будет создана новая ВМ, затем нажать экранную кнопку **[Actions]** и выбрать пункт «Deploy OVF Template...» (см. Переход к созданию новой ВМ);

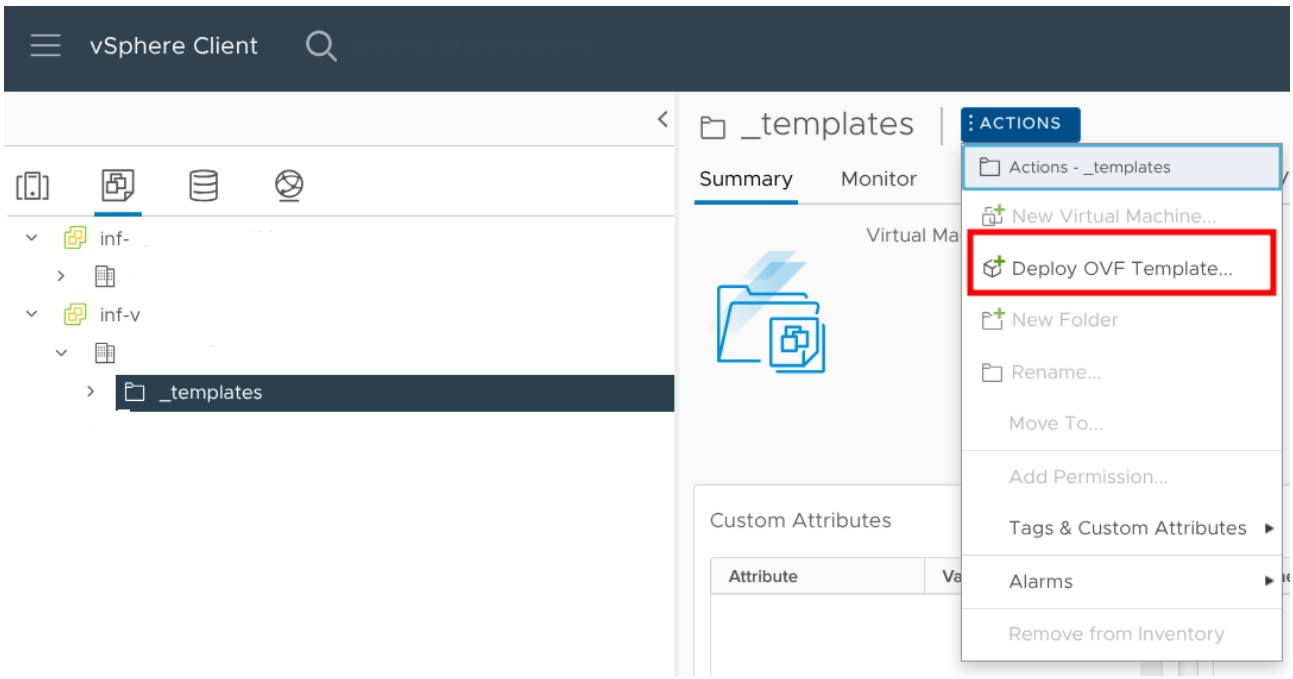

Рисунок 10. Переход к созданию новой ВМ

- далее выбрать пункт «Local file» и нажать экранную кнопку **[Upload]**. После загрузки файла нажать экранную кнопку **[NEXT]** (см. Создание новой ВМ);

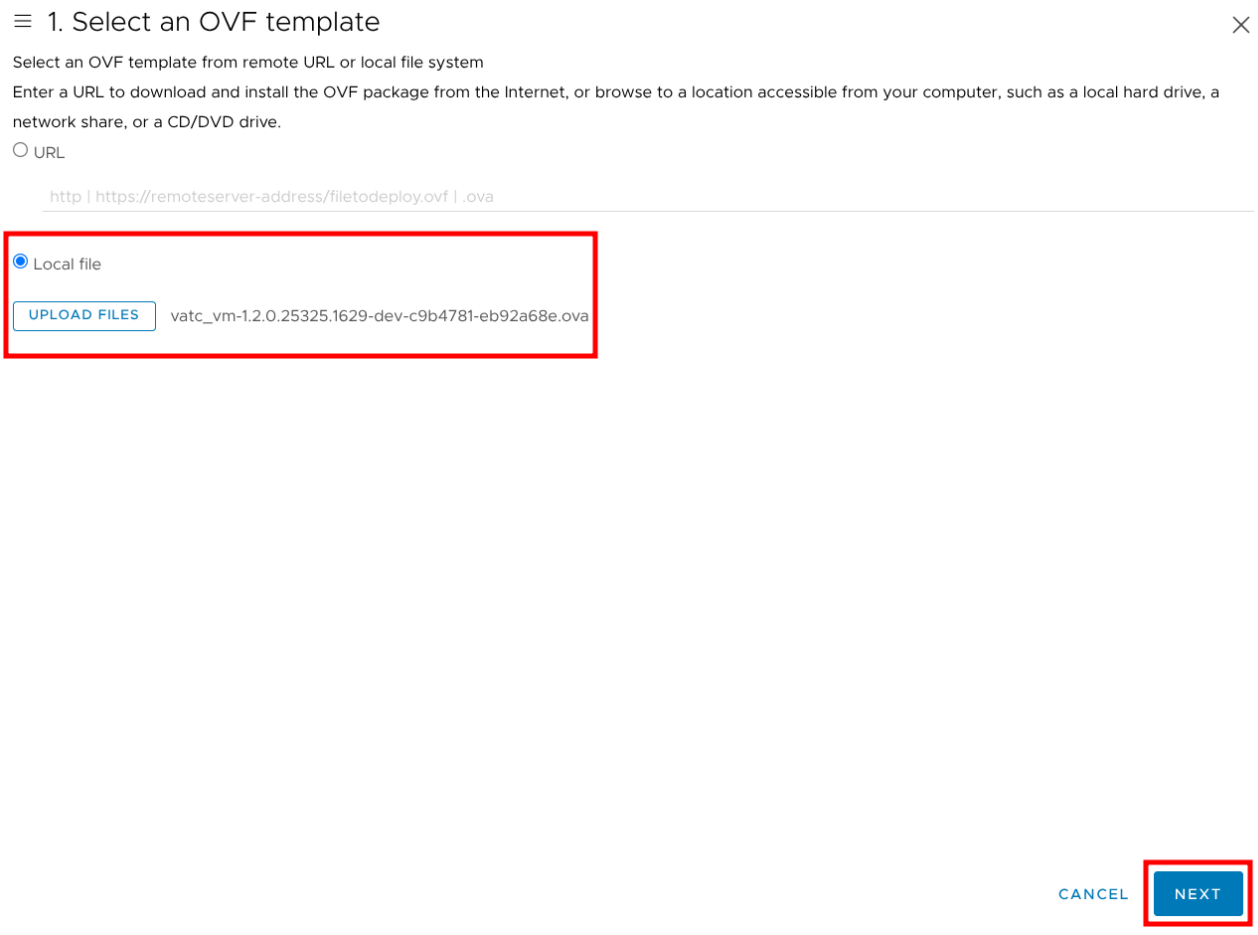

Рисунок 11. Создание новой ВМ

- заполнить имя создаваемой ВМ, выбрать каталог для размещения ВМ и нажать экранную кнопку **[NEXT]** (см. Имя новой ВМ);

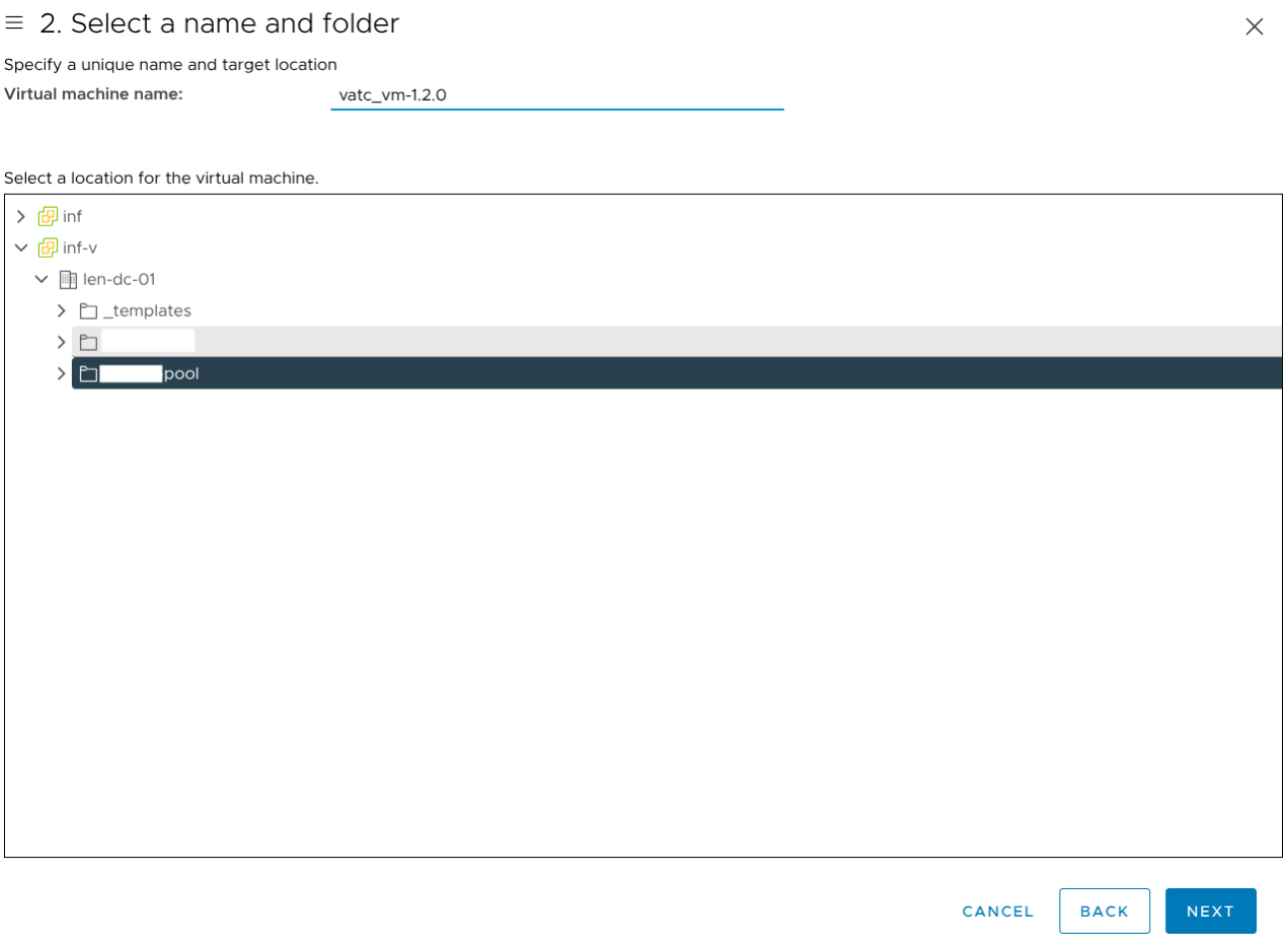

Рисунок 12. Имя новой ВМ

- выбрать вычислительный ресурс для размещения ВМ и нажать экранную кнопку **[NEXT]** (см. Выбор вычислительного ресурса);

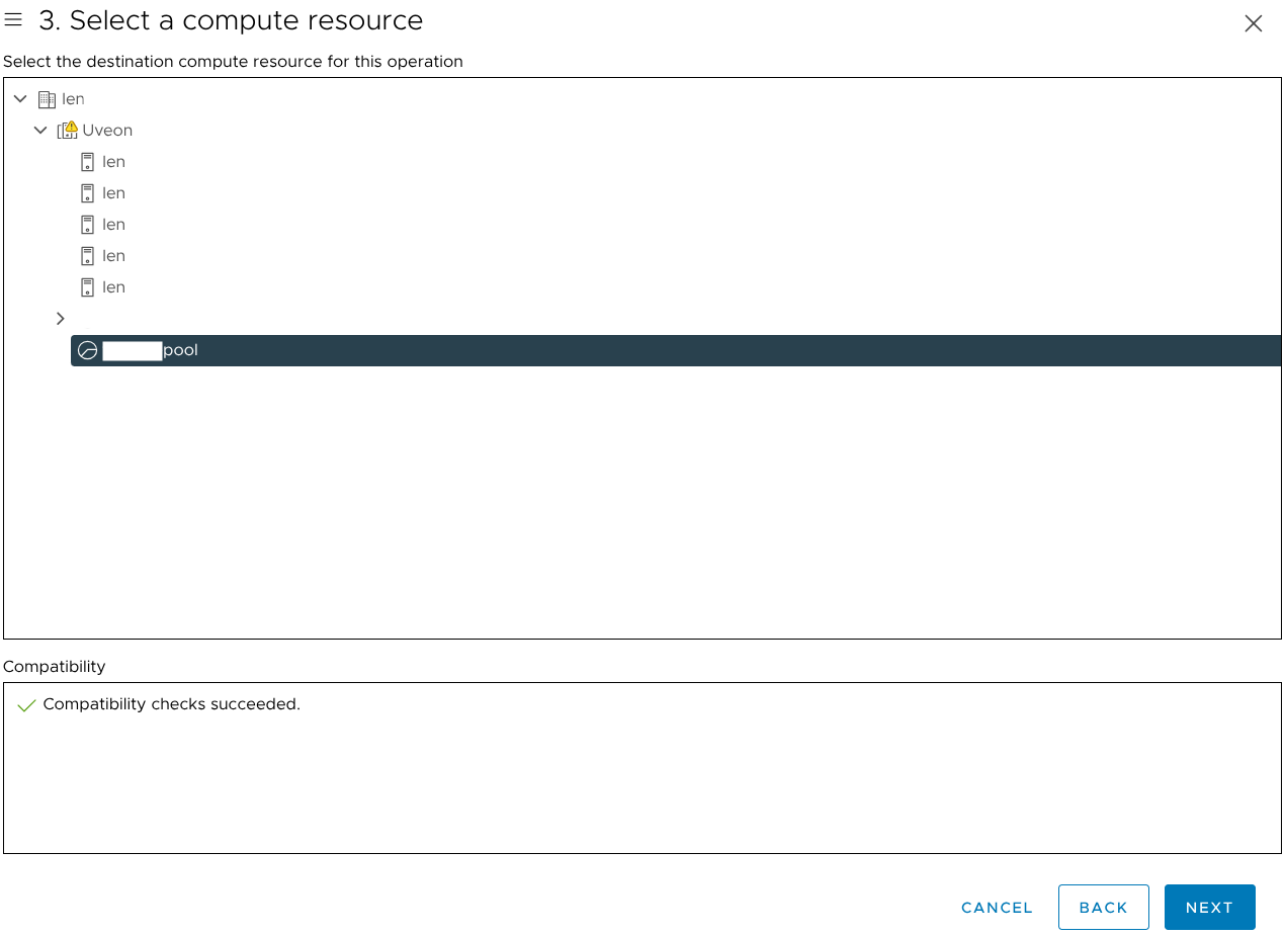

Рисунок 13. Выбор вычислительного ресурса

- ознакомиться с информацией и нажать **[NEXT]** (см. Ознакомление с деталями);

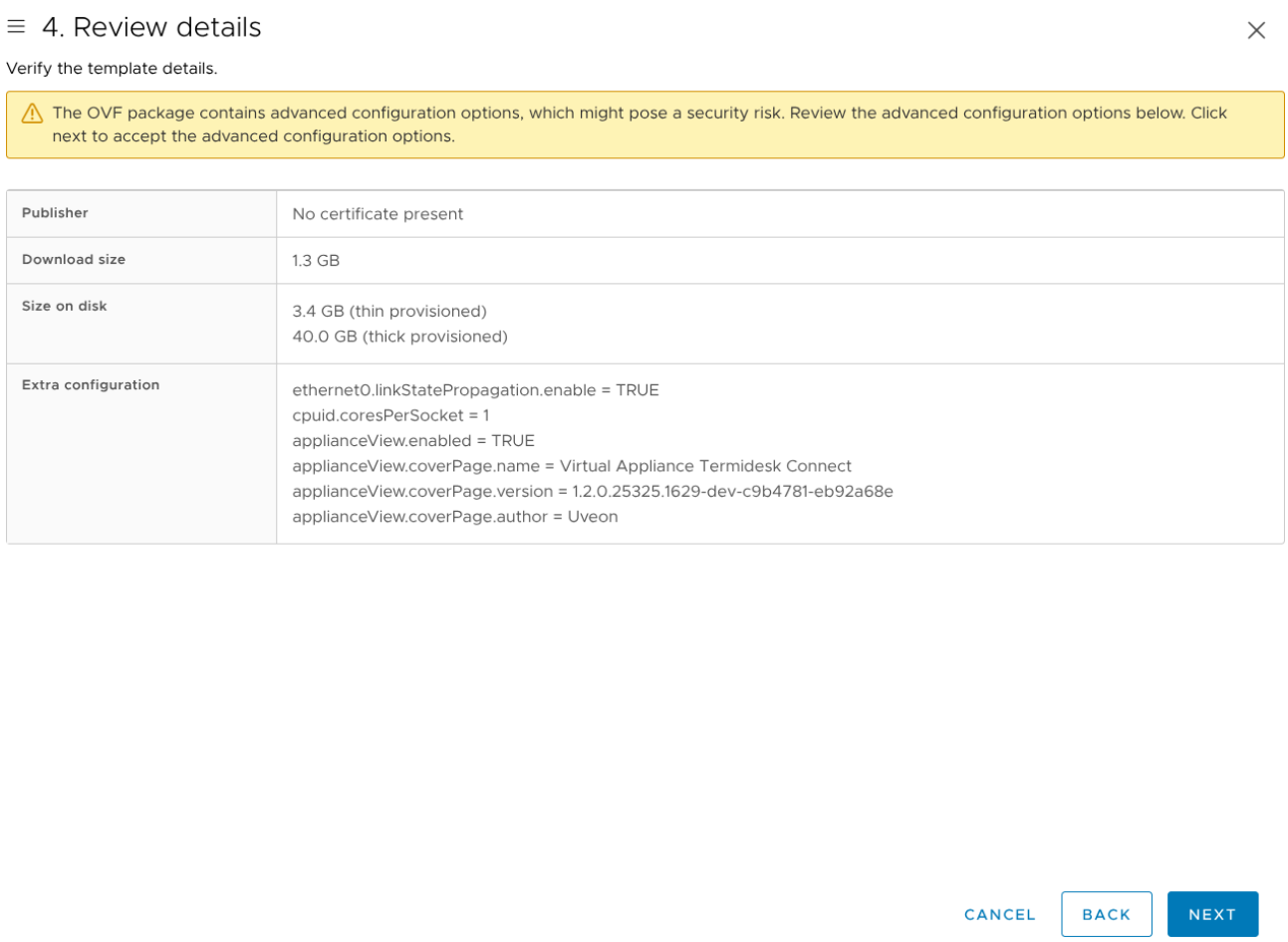

Рисунок 14. Ознакомление с деталями

- выбрать хранилище для размещения ВМ и нажать экранную кнопку **[NEXT]** (см. Выбор хранилища);

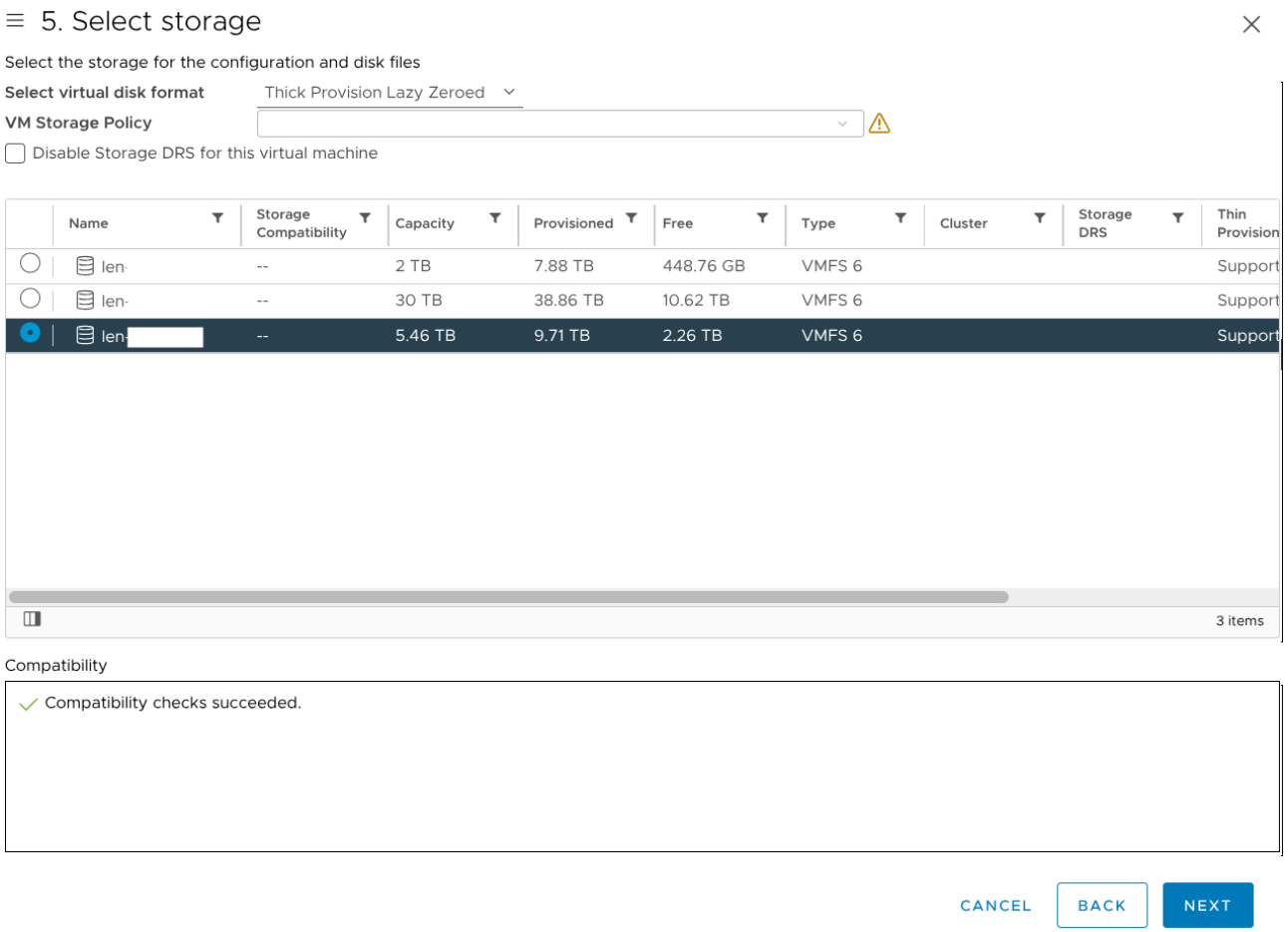

Рисунок 15. Выбор хранилища

- выбрать сеть для размещения ВМ и нажать экранную кнопку **[NEXT]** (см. Выбор сети);

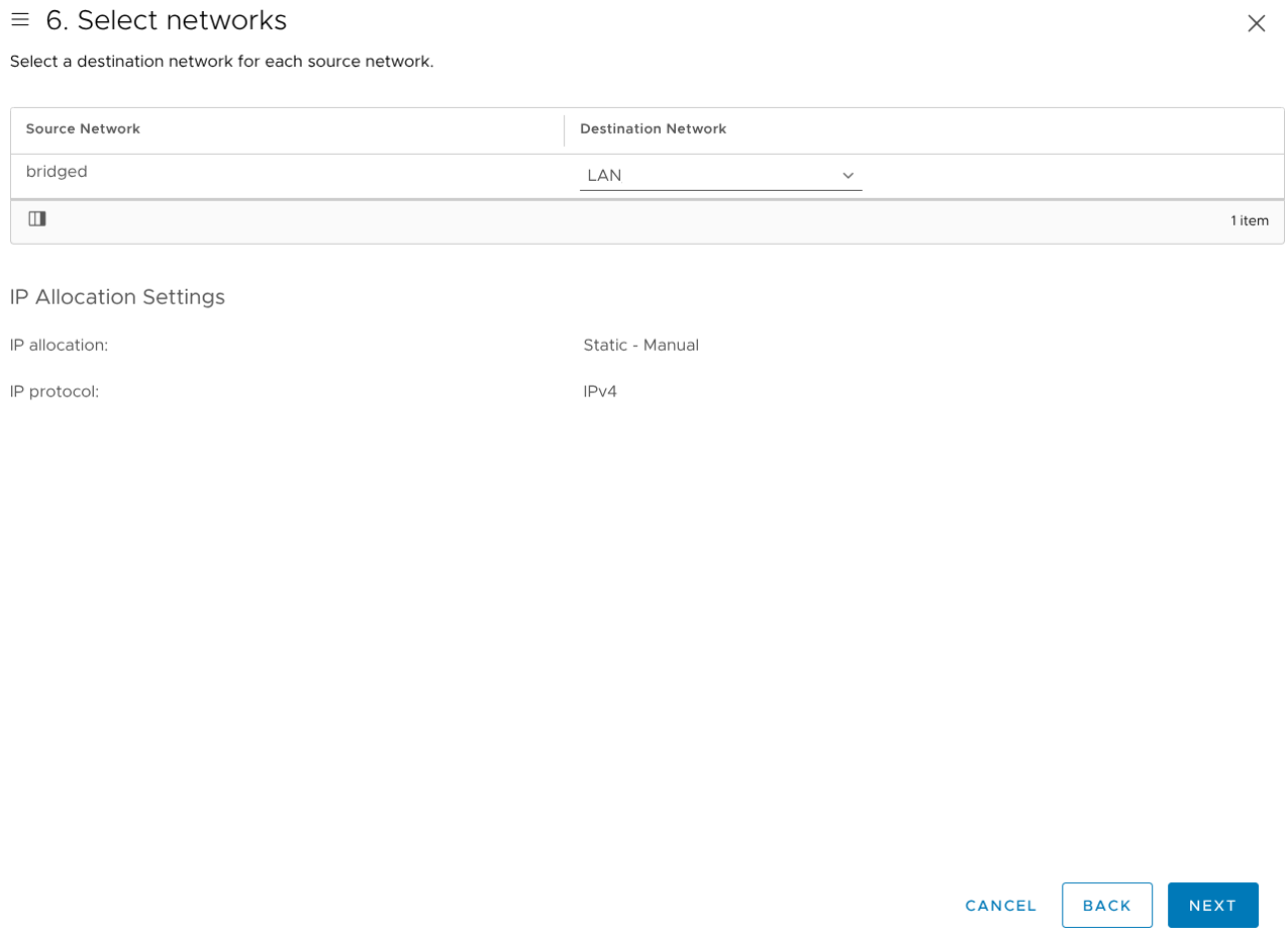

Рисунок 16. Выбор сети

- завершить настройку, нажав экранную кнопку **[NEXT]** (см. Завершение настройки);

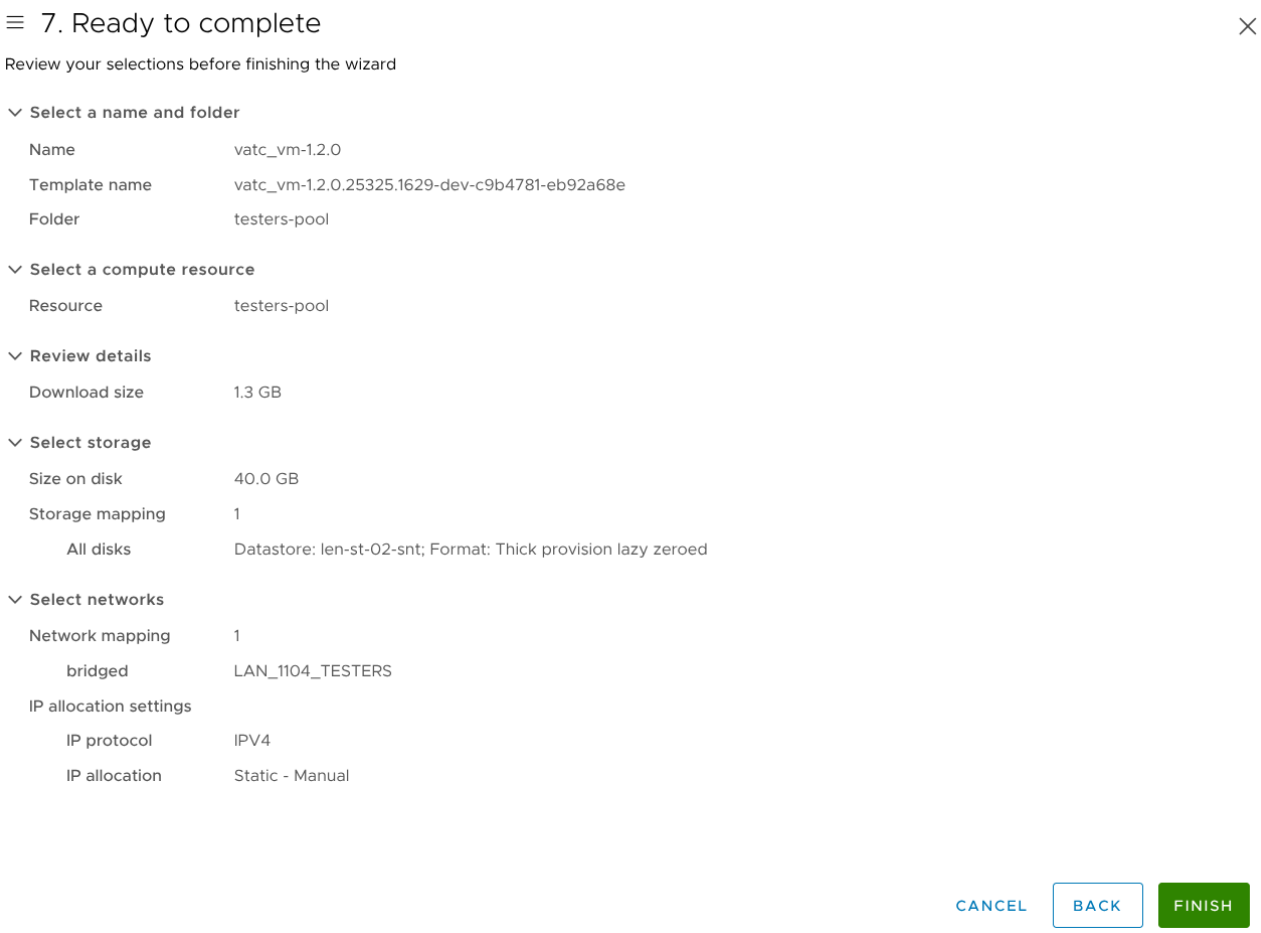

Рисунок 17. Завершение настройки

- после конфигурации параметров ВМ нужно дождаться её создания, затем выполнить запуск ВМ.

После запуска ВМ:

- дождаться автоматического выбора варианта загрузки (см. Выбор варианта загрузки). Если в период таймера нажать любую клавишу, то таймер остановится, нужно будет выбрать пункт «Virtual Appliance Termidesk Connect» и нажать клавишу **<ENTER>**;

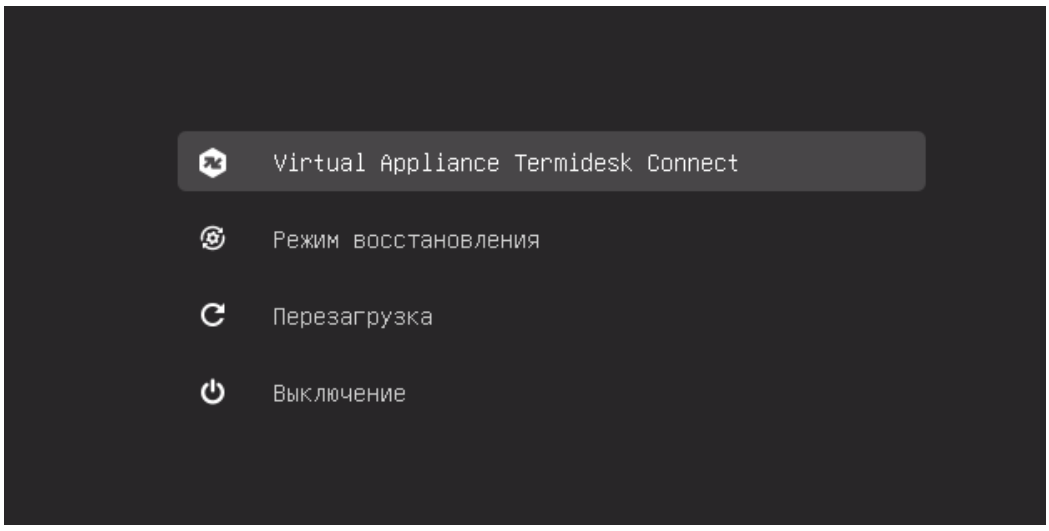

Рисунок 18. Выбор варианта загрузки

- выбрать версию образа Termidesk Connect (см. Выбор загружаемого образа) (при

первоначальной загрузке доступна только одна версия) и нажать клавишу **<ENTER>**;

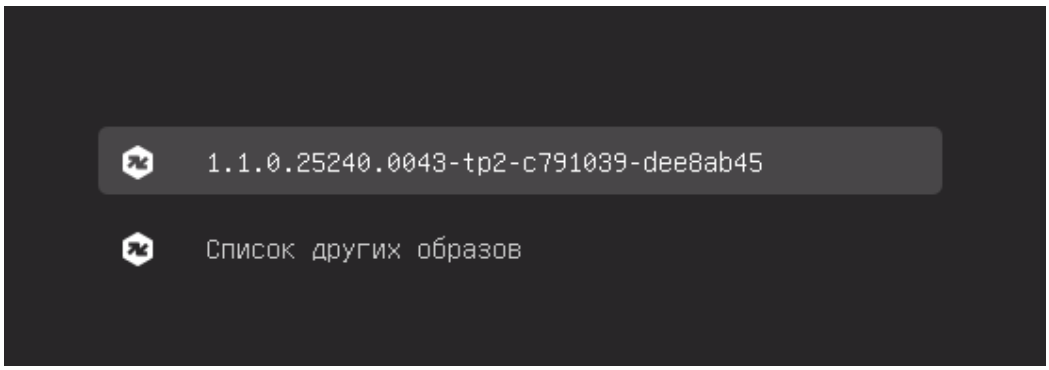

Рисунок 19. Выбор загружаемого образа

- прочитать и принять условия лицензионного соглашения (см. Лицензионное соглашение), переключившись на экранную кнопку **[OK]** и нажав клавишу **<ENTER>**;

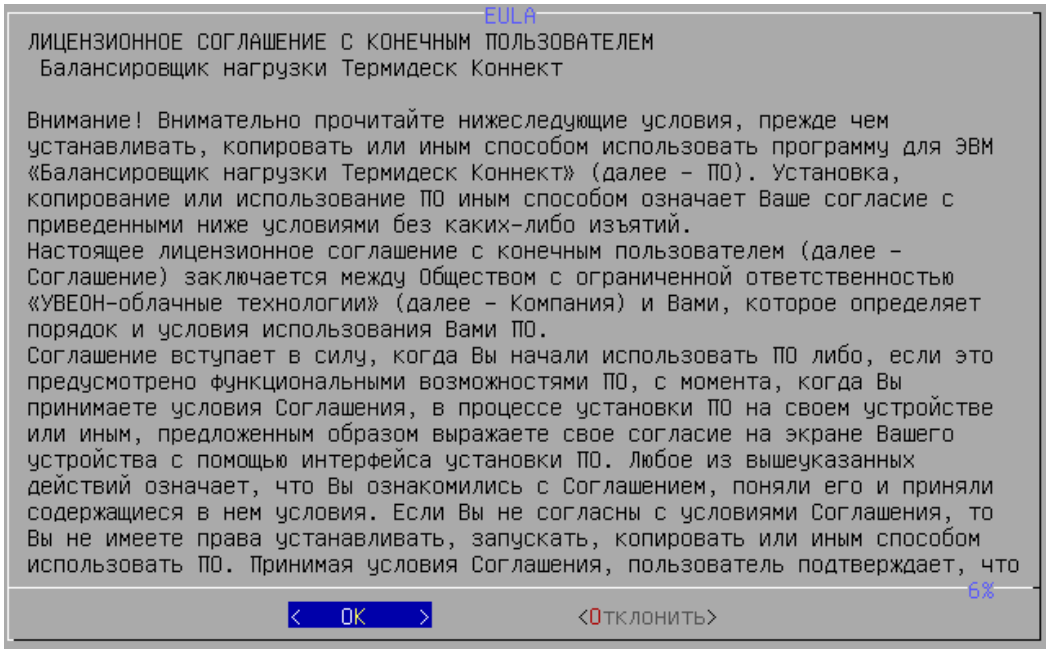

Рисунок 20. Лицензионное соглашение

- дождаться появления информационного сообщения о сертификатах (см. Информационное сообщение) и нажать клавишу **<ENTER>**;

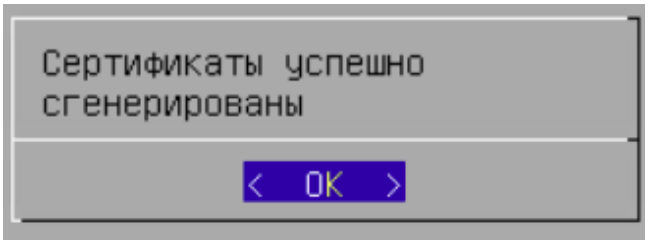

Рисунок 21. Информационное сообщение

- дождаться отображения главного окна Termidesk Connect. Затем выполнить настройку согласно подразделу **Первоначальна настройка Termidesk Connect**.

Загрузка на примере платформы виртуализации zVirt MAX

Для загрузки Termidesk Connect на платформу виртуализации zVirt MAX выполнить

действия:

- в веб-интерфейсе zVirt MAX перейти в раздел «Хранилище – Диски», далее нажать экранную кнопку **[Загрузить]** и выбрать элемент раскрывающегося списка «Начать» (см. Переход к загрузке образа диска);

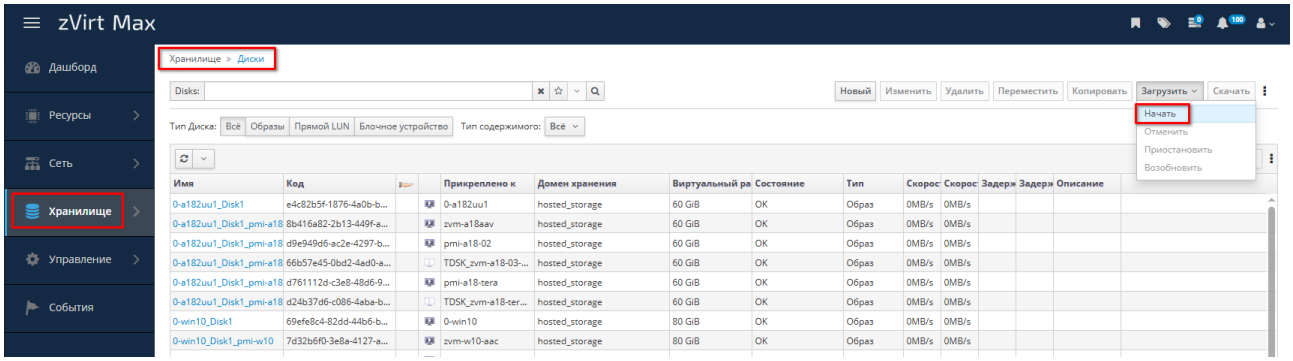

Рисунок 22. Переход к загрузке образа диска

- в открывшемся окне «Загрузить образ» нажать экранную кнопку **[Выберите файл]** и выбрать файл формата `.qcow2`, далее нажать экранную кнопку **[ОК]** и дождаться загрузки образа диска (см. Загрузка образа диска);

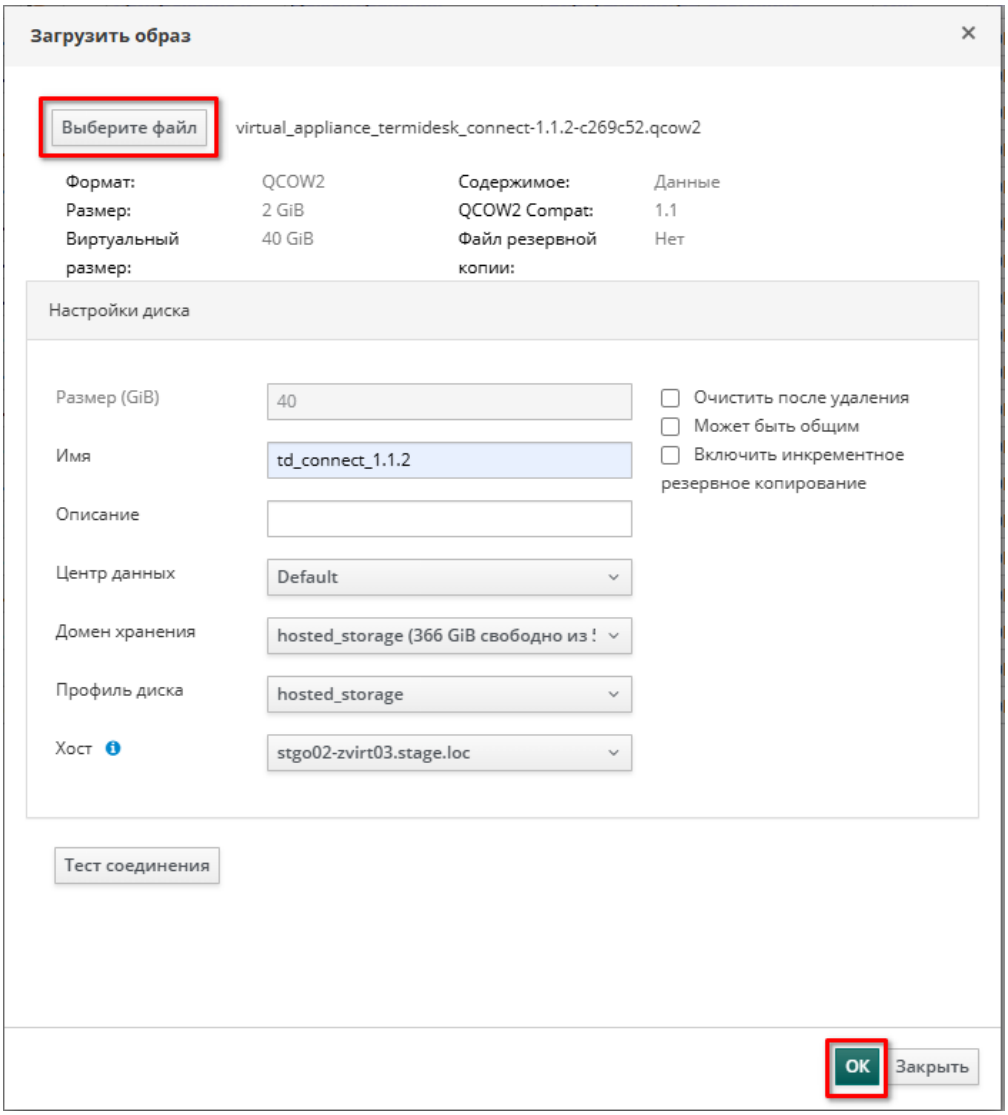

Рисунок 23. Загрузка образа диска

- перейти в раздел «Ресурсы – Виртуальные машины», далее нажать экранную кнопку **[Создать]** (см. Переход к созданию ВМ);

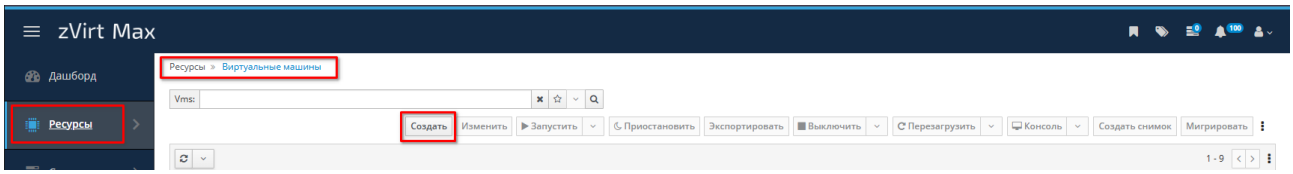

Рисунок 24. Переход к созданию ВМ

- в открывшемся окне «Новая виртуальная машина» (раздел «Общее») указать необходимые параметры, далее в области «Виртуальные диски» нажать экранную кнопку **[Прикрепить]** (см. Создание ВМ);

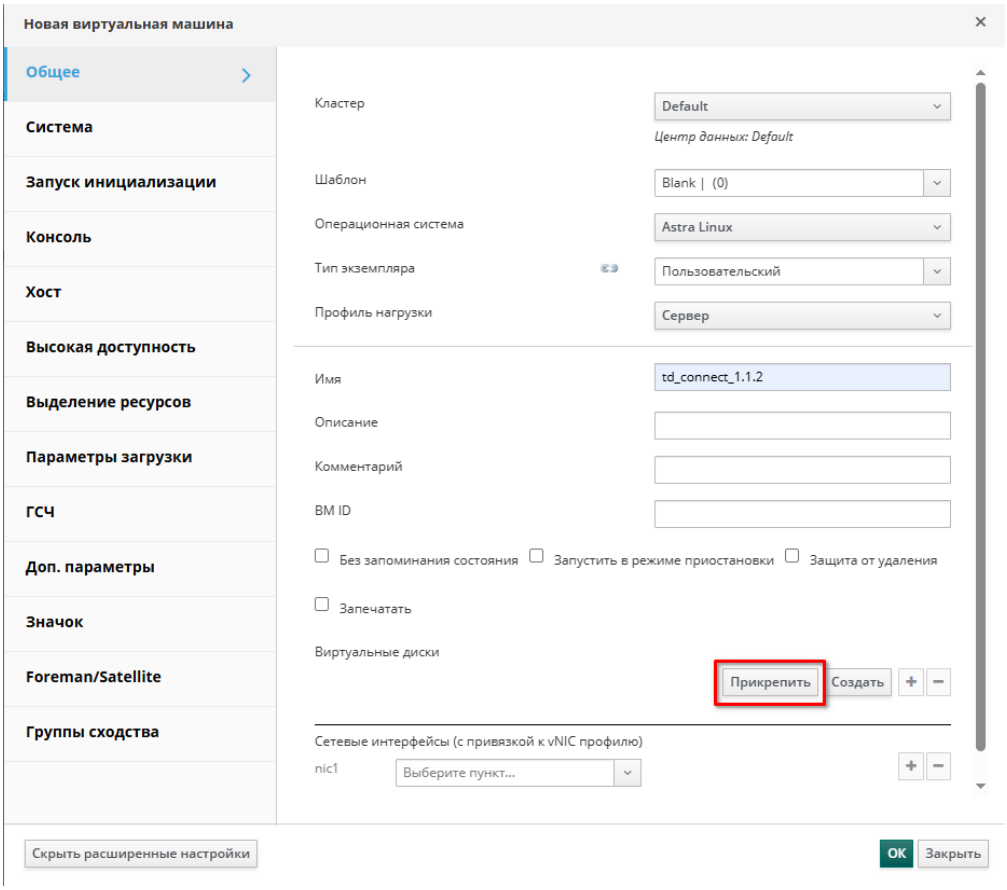

Рисунок 25. Создание ВМ

- в открывшемся окне «Прикрепить виртуальные диски» выбрать в списке недавно загруженный виртуальный диск и нажать экранную кнопку **[OK]** (см. Прикрепление виртуального диска);

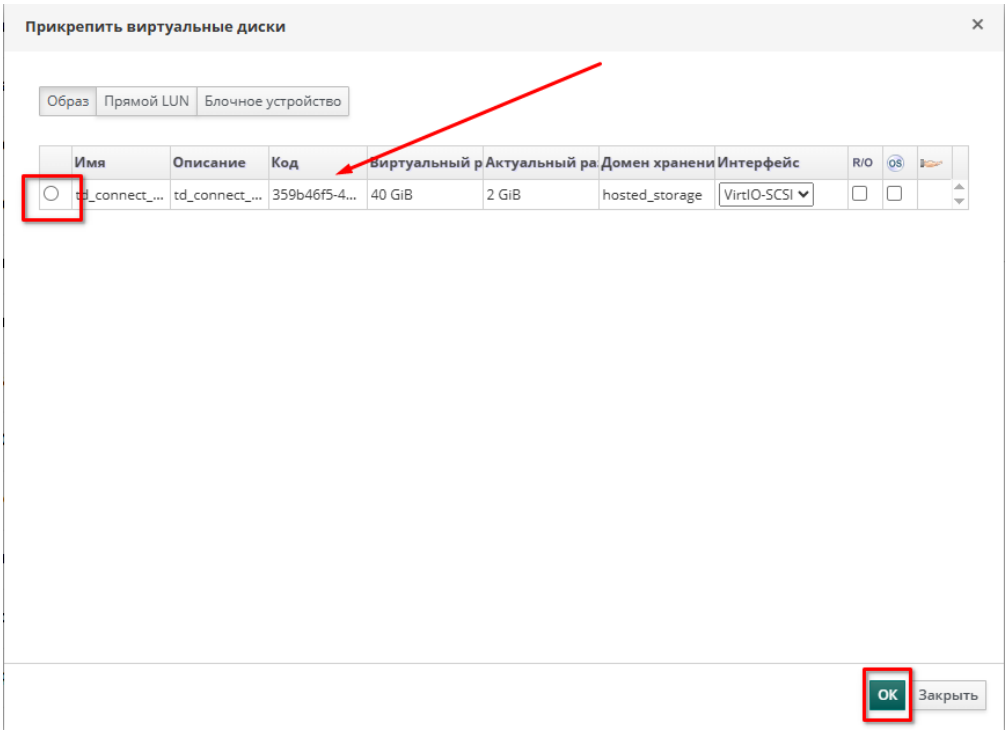

Рисунок 26. Прикрепление виртуального диска

- перейти в раздел «Система», далее в раскрывающемся списке «Тип BIOS» выбрать «Чипсет Q35 с UEFI» и нажать экранную кнопку **[ОК]** (см. Настройка параметров);

Минимальные аппаратные требования ВМ, на которой функционирует Termidesk Connect, должны соответствовать следующим:

- оперативная память – 12 ГБ и более;
- виртуальный процессор – 1 шт. и более.

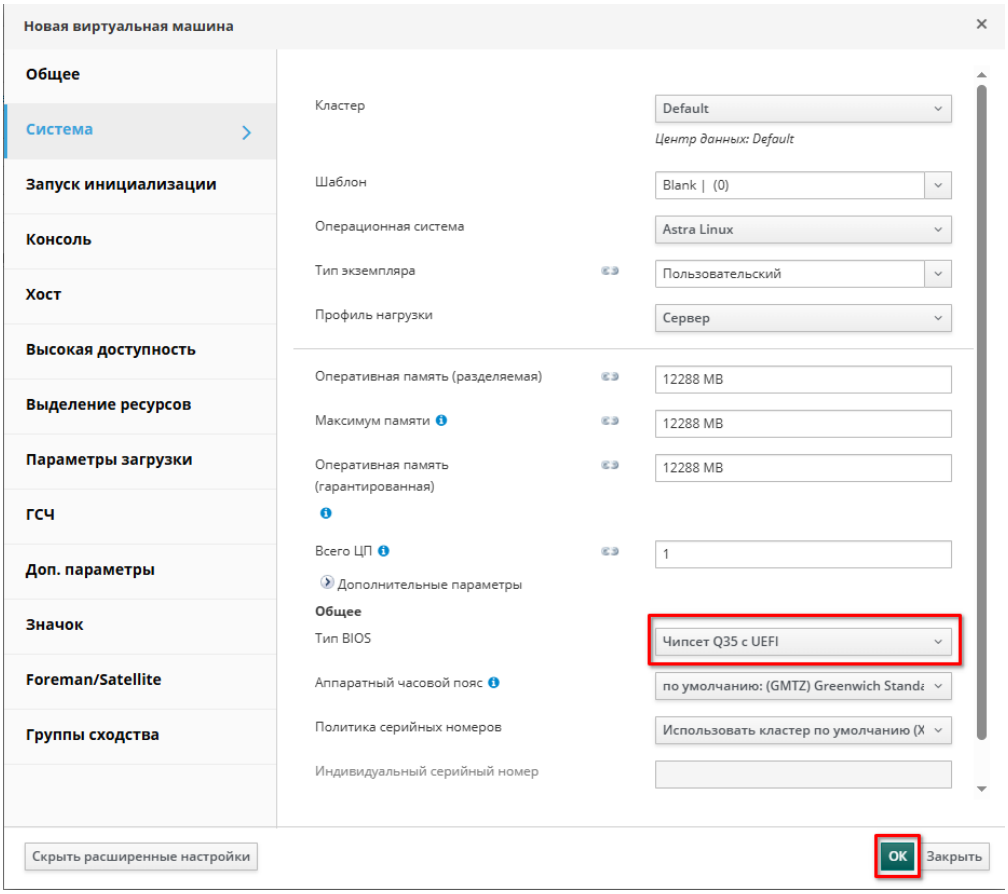

Рисунок 27. Настройка параметров

- в разделе «Ресурсы – Виртуальные машины» выбрать строку с ВМ Termidesk Connect и нажать экранную кнопку **[Запустить]** (см. Запуск ВМ).

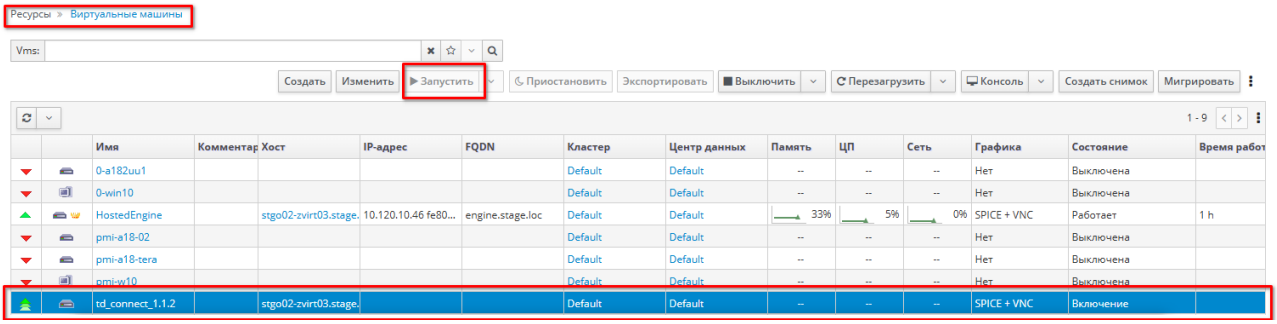

Рисунок 28. Запуск ВМ

После запуска ВМ:

- выбрать в меню пункт «Virtual Appliance Termidesk Connect» (по умолчанию) и нажать клавишу **<ENTER>** (см. Выбор варианта загрузки);

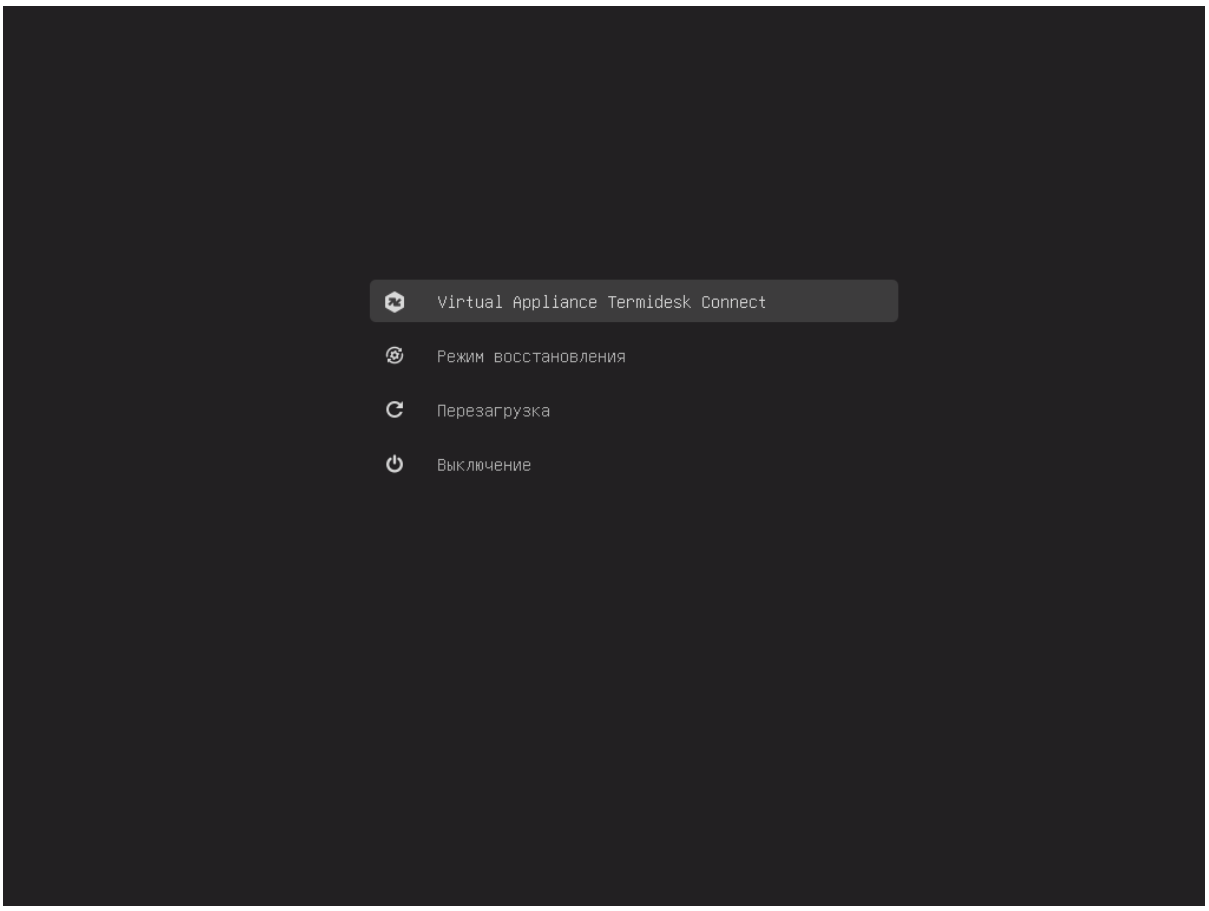

Рисунок 29. Выбор варианта загрузки

- выбрать загружаемый образ и нажать клавишу **<ENTER>** (см. Выбор загружаемого образа);

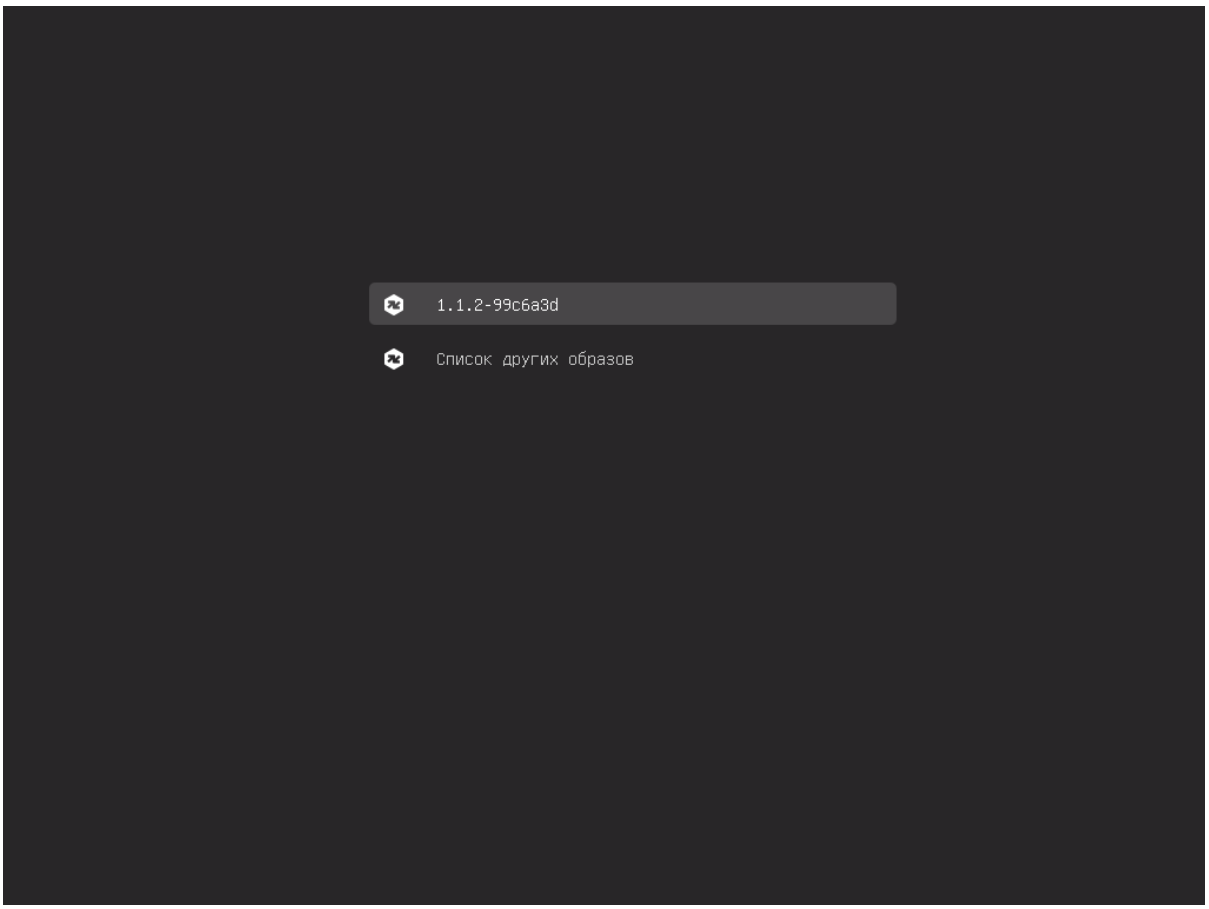

Рисунок 30. Выбор загружаемого образа

- прочитать и принять условия лицензионного соглашения, для этого переключитсья на экранную кнопку **[OK]** и нажать клавишу **<ENTER>** (см. Лицензионное соглашение);

Переключение между пунктами меню выполняется клавишей **<TAB>**. Подтверждение выбора выполняется клавишами **<ENTER>** или **<SPACE>**.

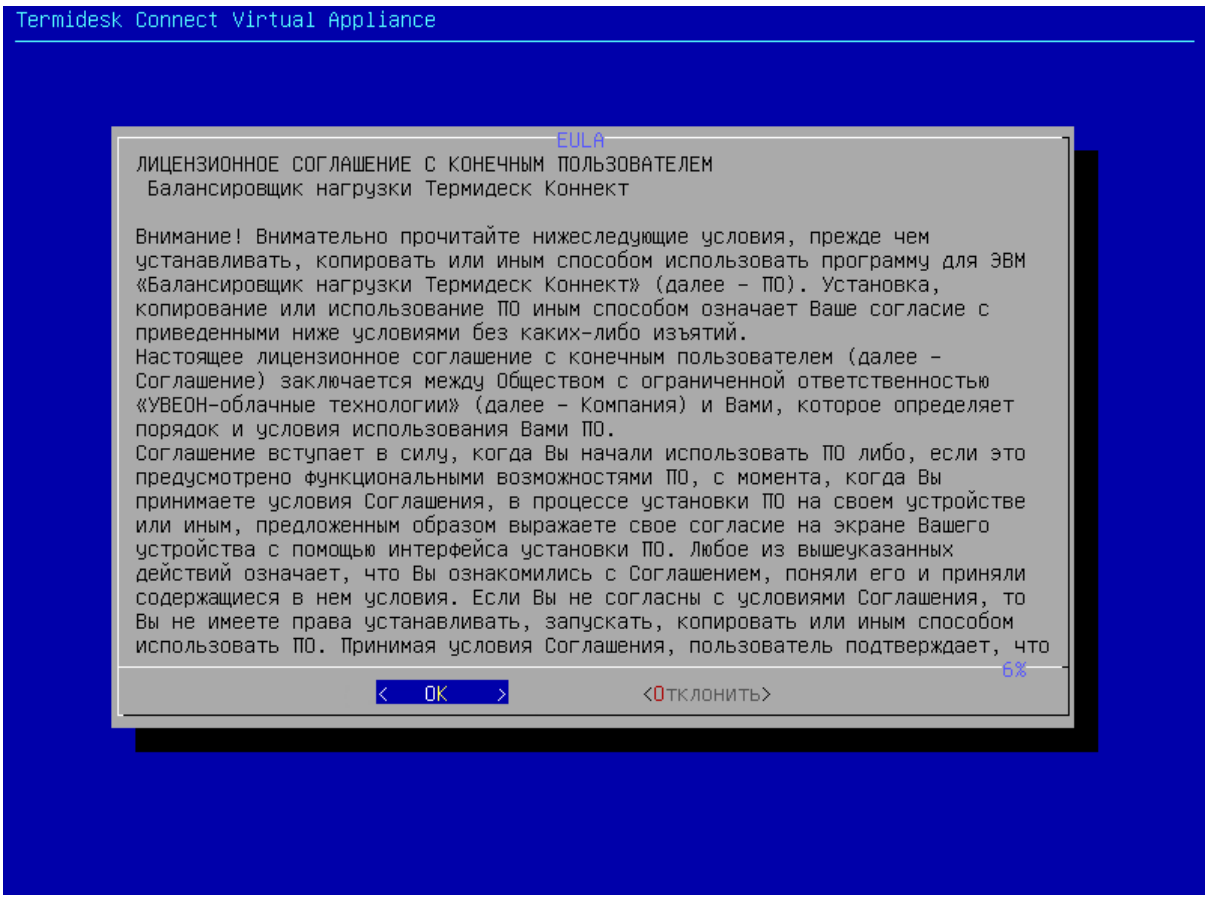

Рисунок 31. Лицензионное соглашение

- дождаться появления информационного сообщения о сертификатах и нажать клавишу **<ENTER>** (см. Информационное сообщение);

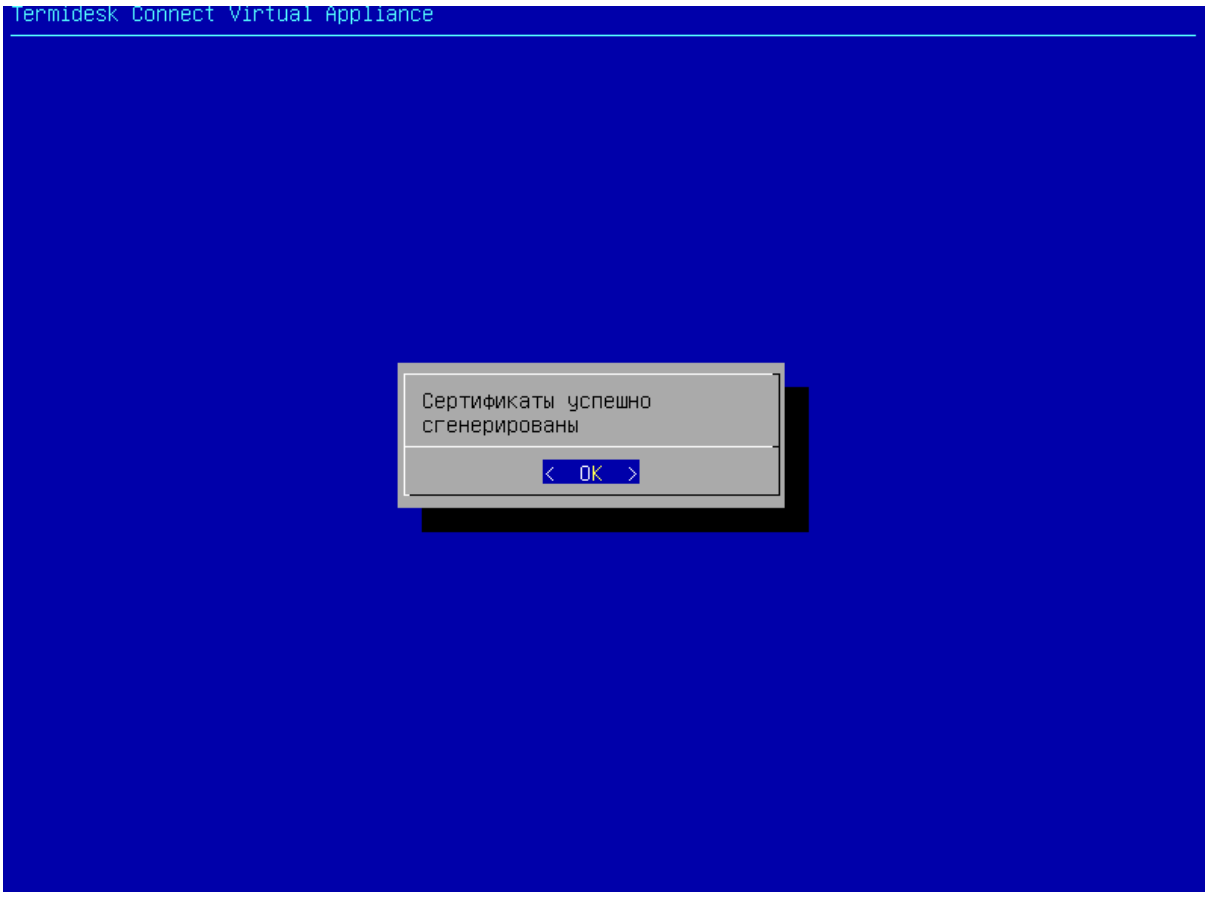

Рисунок 32. Информационное сообщение

- дождаться отображения главного окна Termidesk Connect (см. Главное окно Termidesk Connect). Затем выполнить настройку согласно подразделу **Первоначальная настройка Termidesk Connect**.

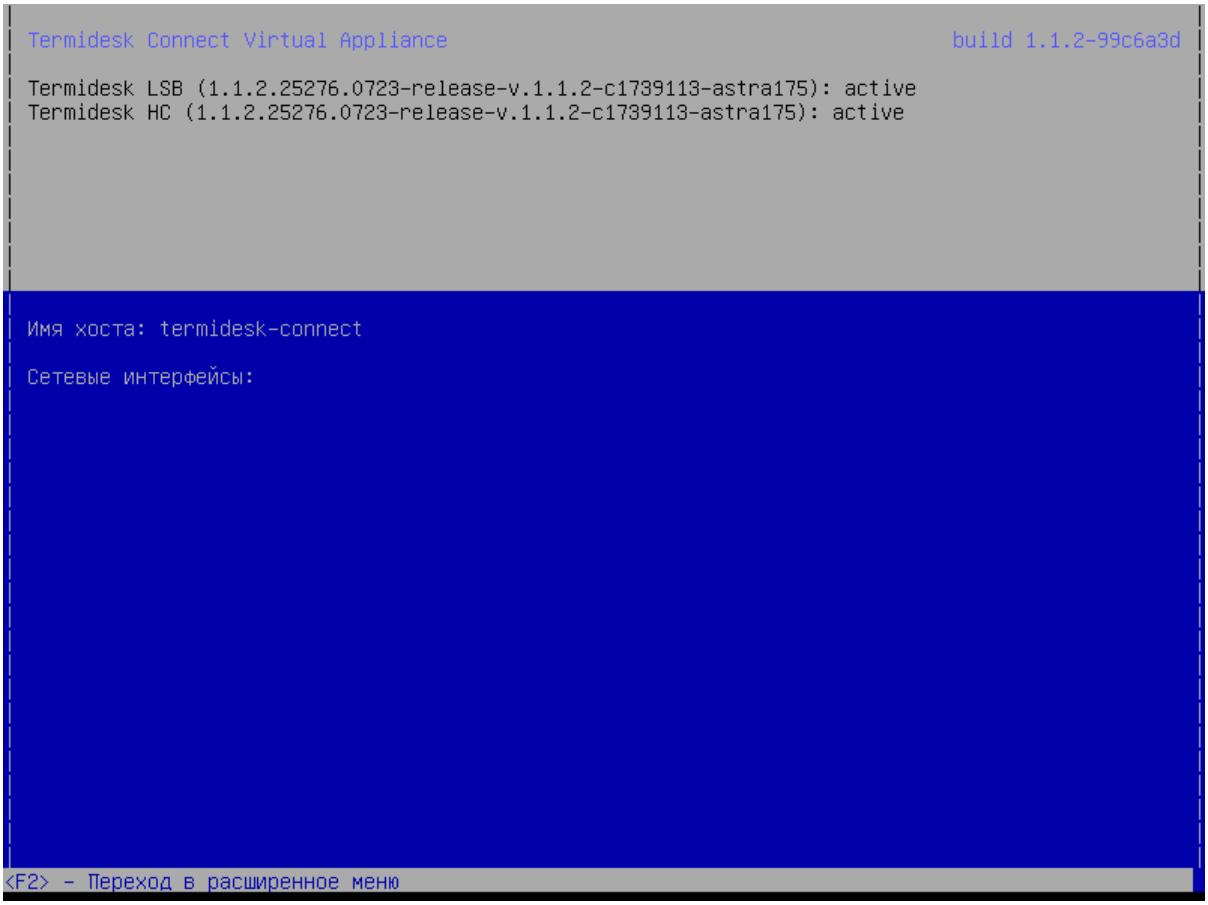

Рисунок 33. Главное окно Termidesk Connect

Загрузка на примере платформы виртуализации VMmanager

Для загрузки Termidesk Connect на платформу виртуализации VMmanager выполнить действия:

- в веб-интерфейсе VMmanager перейти в раздел «Виртуальные машины», далее нажать экранную кнопку **[Создать VM]** (см. Переход к созданию виртуальной машины);

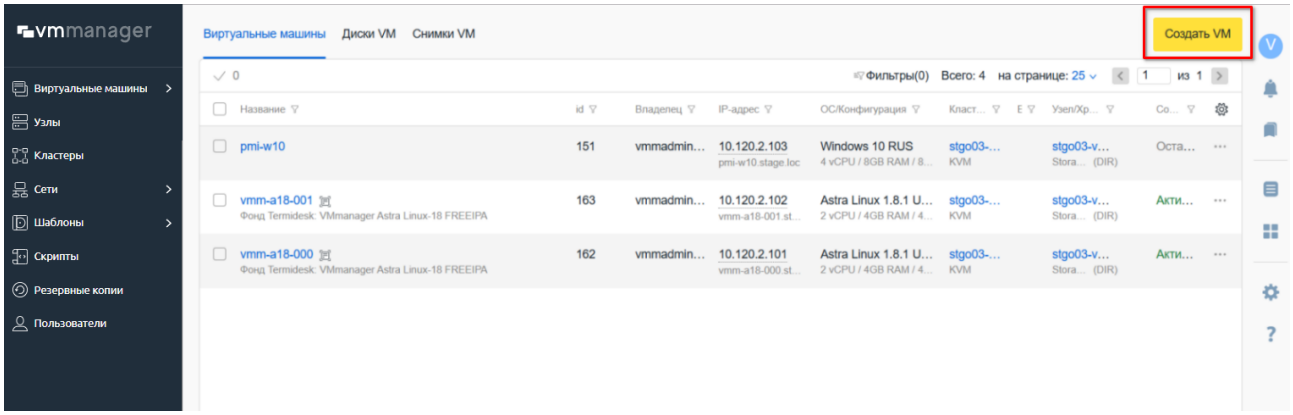

Рисунок 34. Переход к созданию виртуальной машины

- в открывшемся окне «Новая виртуальная машина», в области «Операционные системы кластера KVM» выбрать поле «NoOS» (см. Настройка конфигурации);

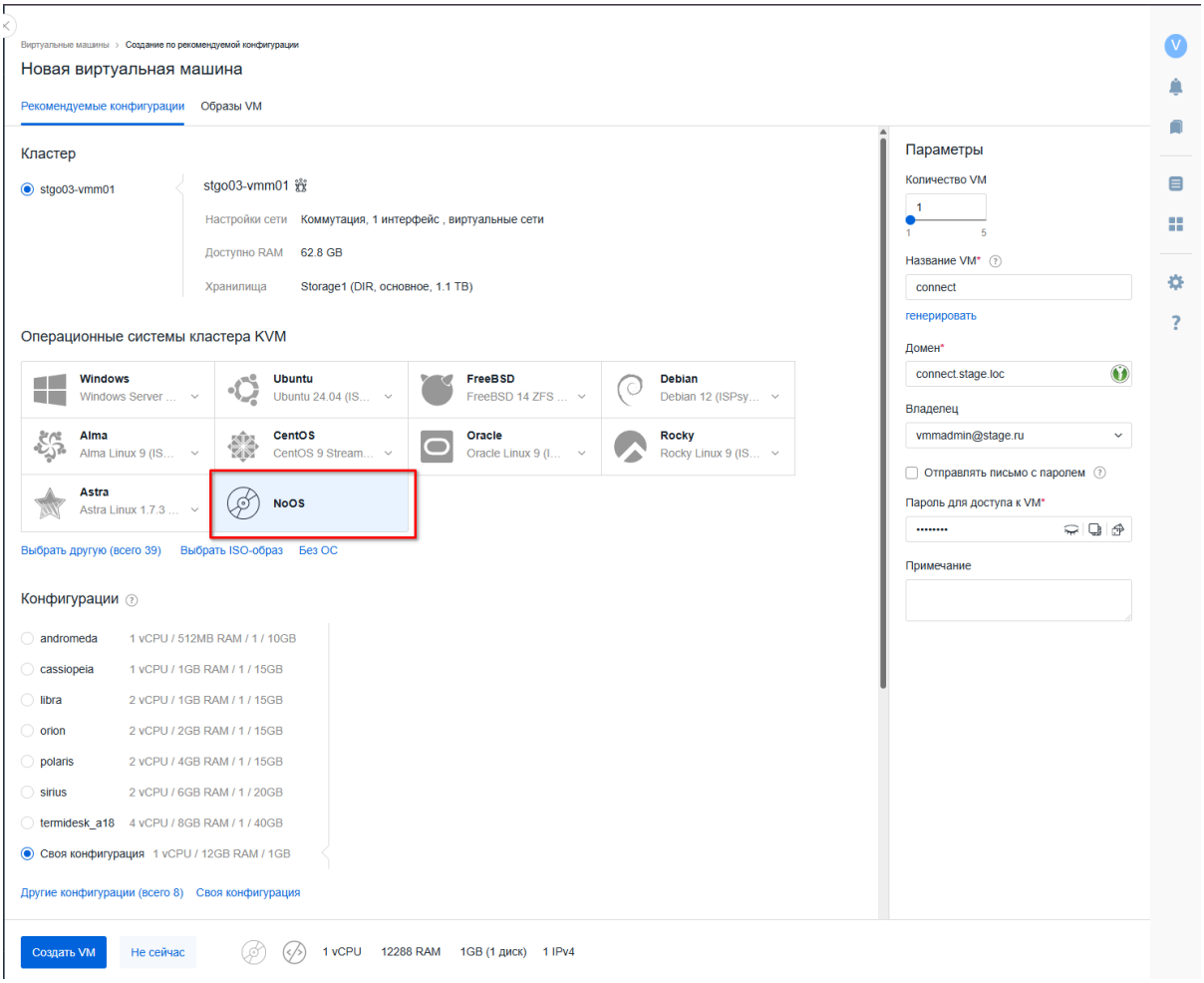

Рисунок 35. Настройка конфигурации

- далее в списке области «Конфигурации» выбрать строку «Своя конфигурация» и перейти по ссылке «Своя конфигурация» в нижней части области (см. Выбор конфигурации);

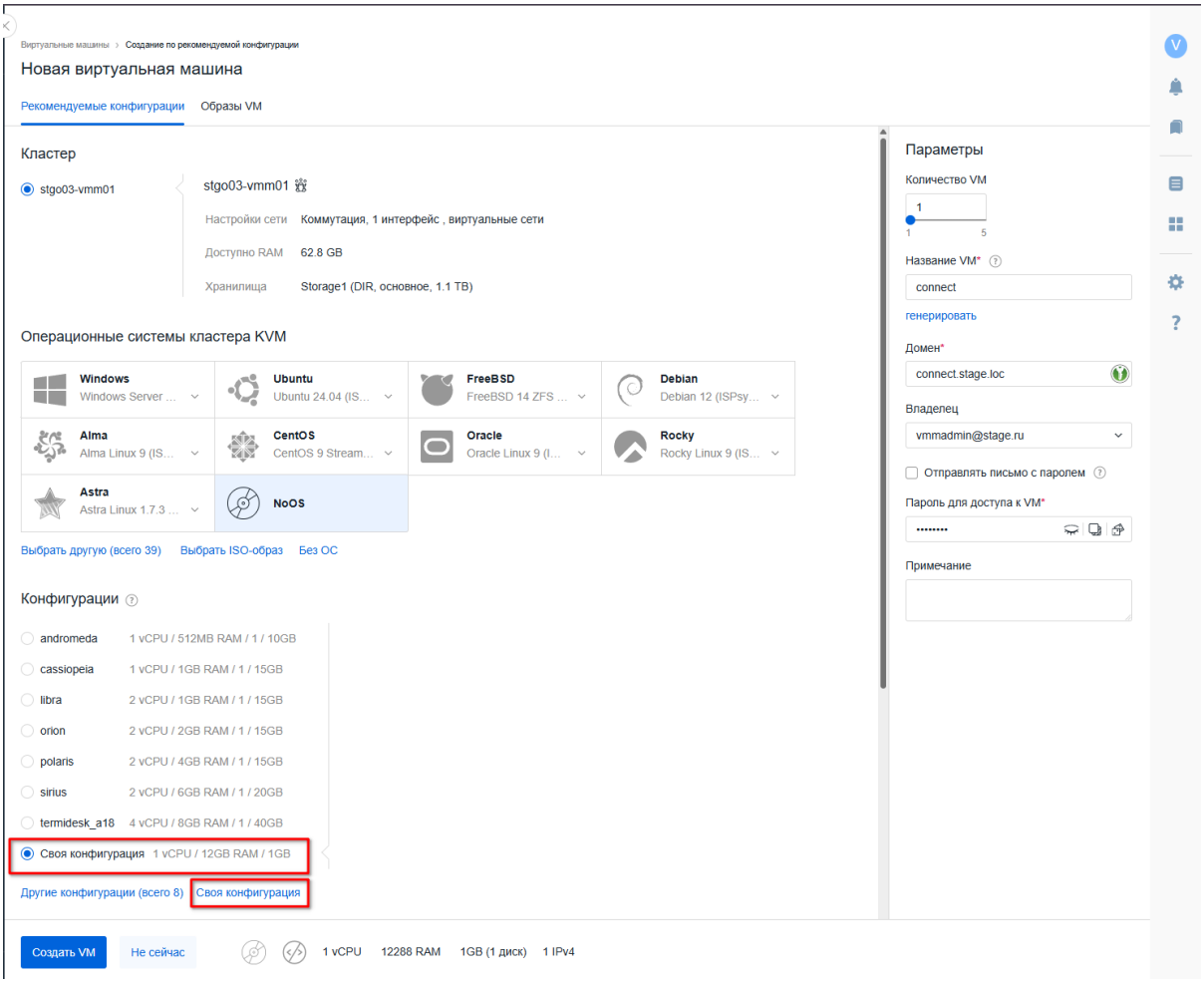

Рисунок 36. Выбор конфигурации

- в открывшемся окне «Своя конфигурация VM», в области «Ресурсы» нажать экранную кнопку **[Изменить]**, установить значение «12 GB RAM» и размер диска (размер диска должен быть не меньше, чем у образа диска с Termidesk Connect) (см. Выбор ресурсов);

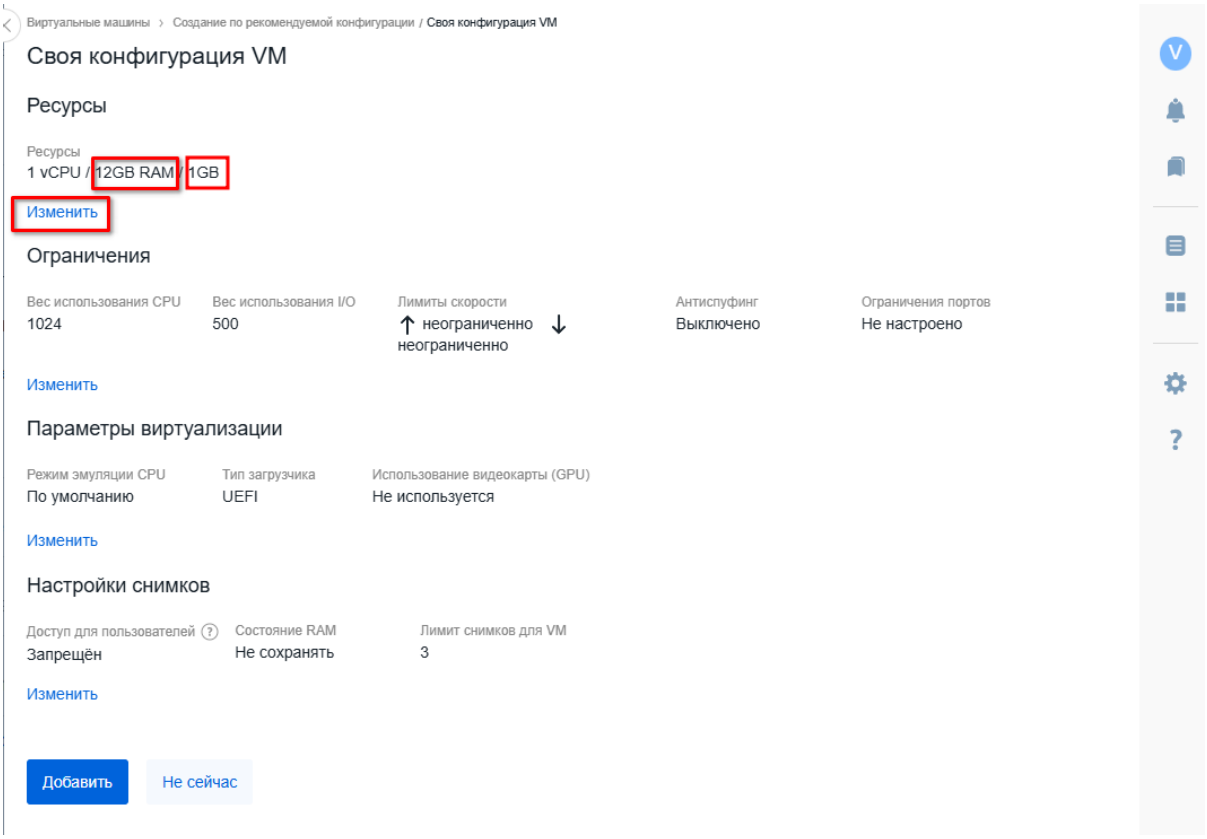

Рисунок 37. Выбор ресурсов

- далее в области «Параметры виртуализации» нажать экранную кнопку **[Изменить]** и установить для параметра «Тип загрузчика» значение «UEFI» (см. Параметры виртуализации);

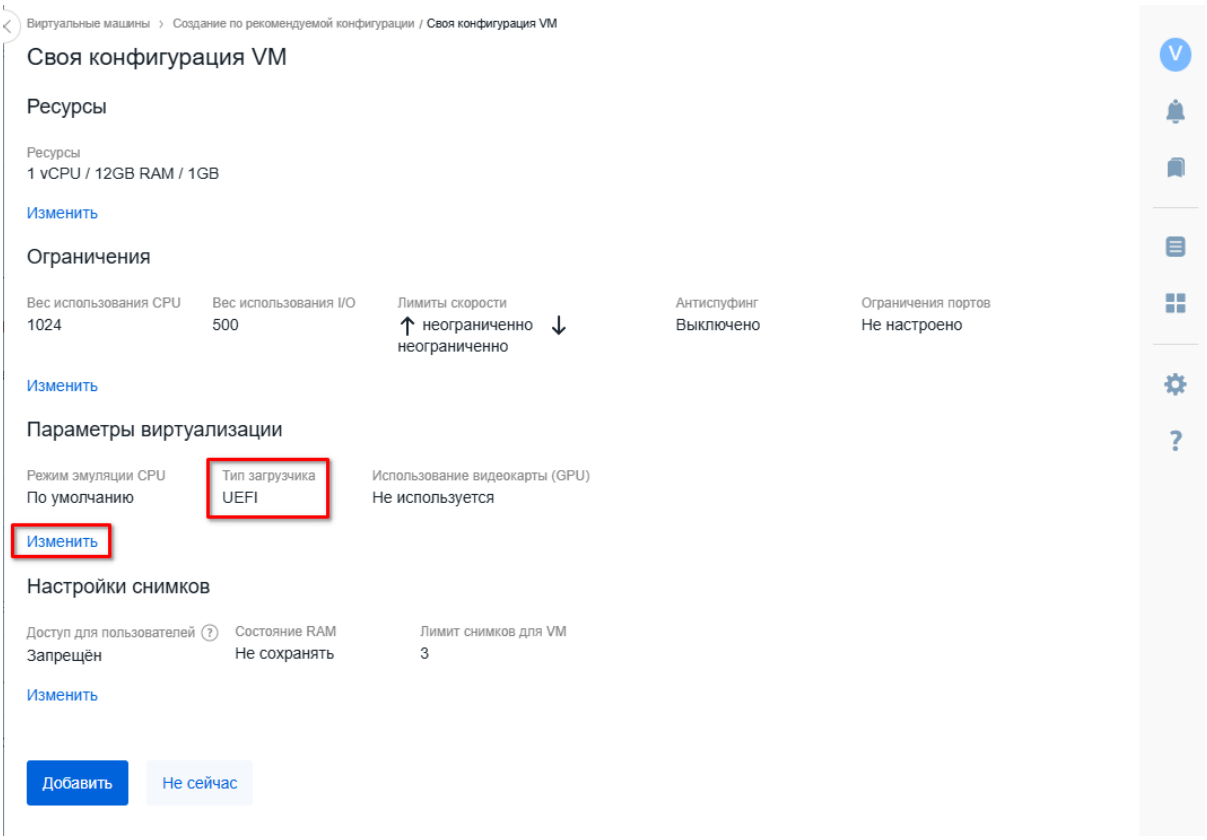

Рисунок 38. Параметры виртуализации

- далее нажать экранную кнопку **[Добавить]** (см. Добавление конфигурации);

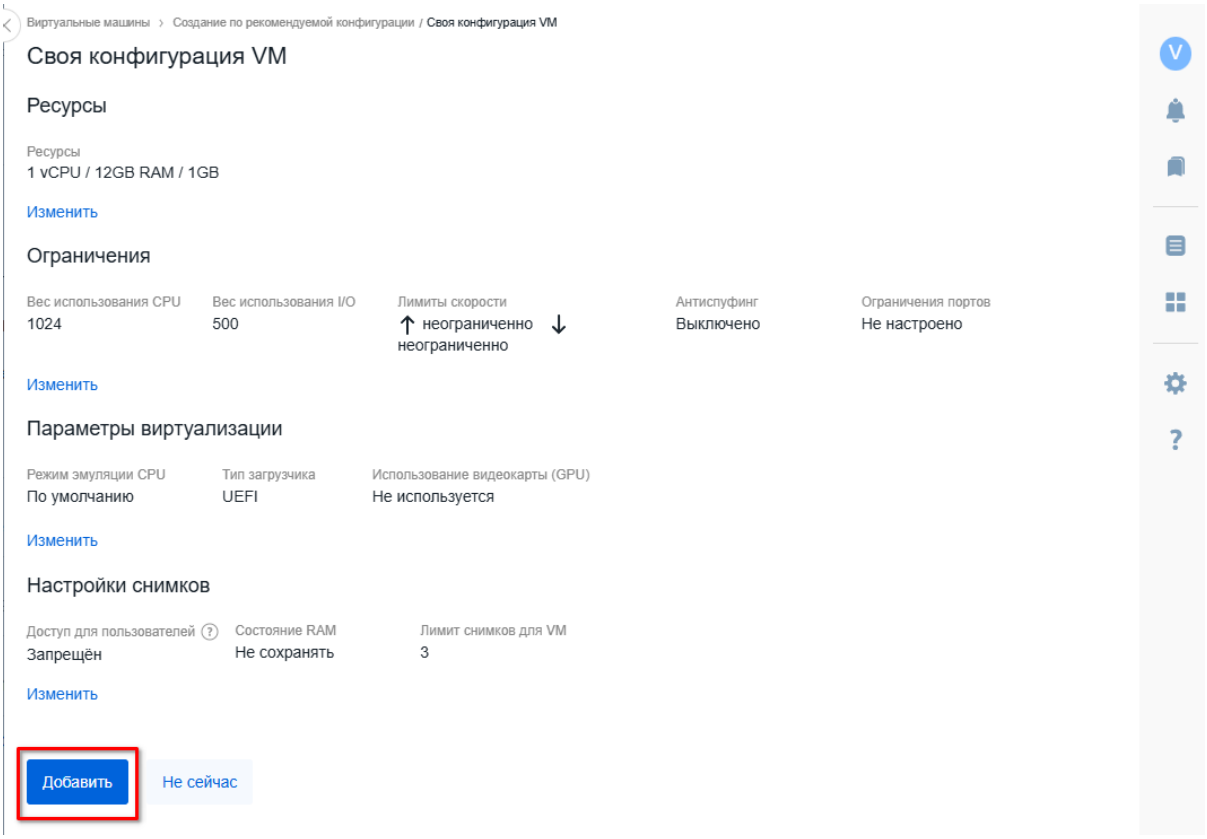

Рисунок 39. Добавление конфигурации

- в окне «Новая виртуальная машина», в области «Параметры» для полей «Название VM» и «Пароль для доступа к VM» ввести соответствующие значения (см. Параметры ВМ);

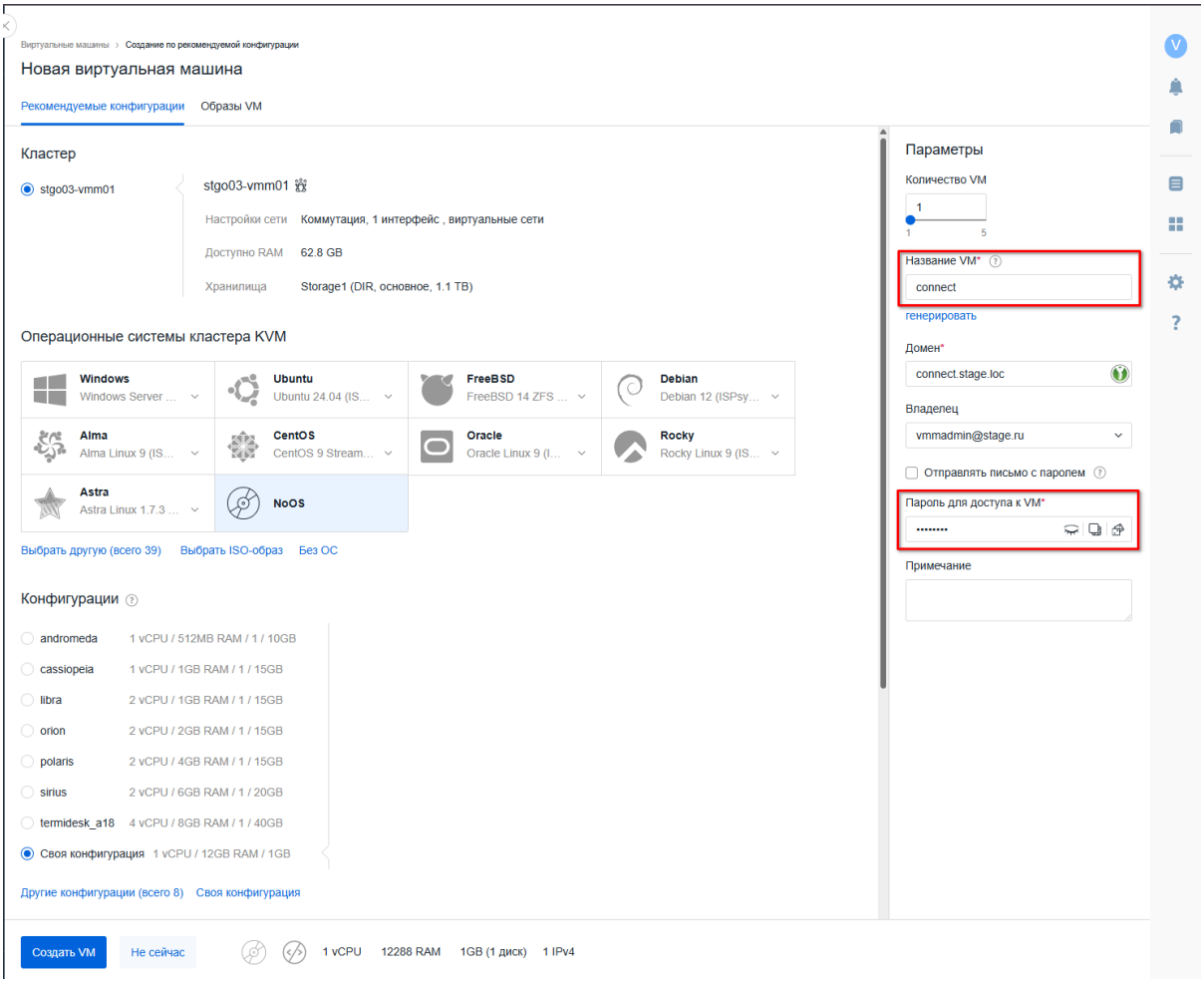

Рисунок 40. Параметры ВМ

- далее нажать экранную кнопку **[Создать VM]** (см. Создание ВМ);

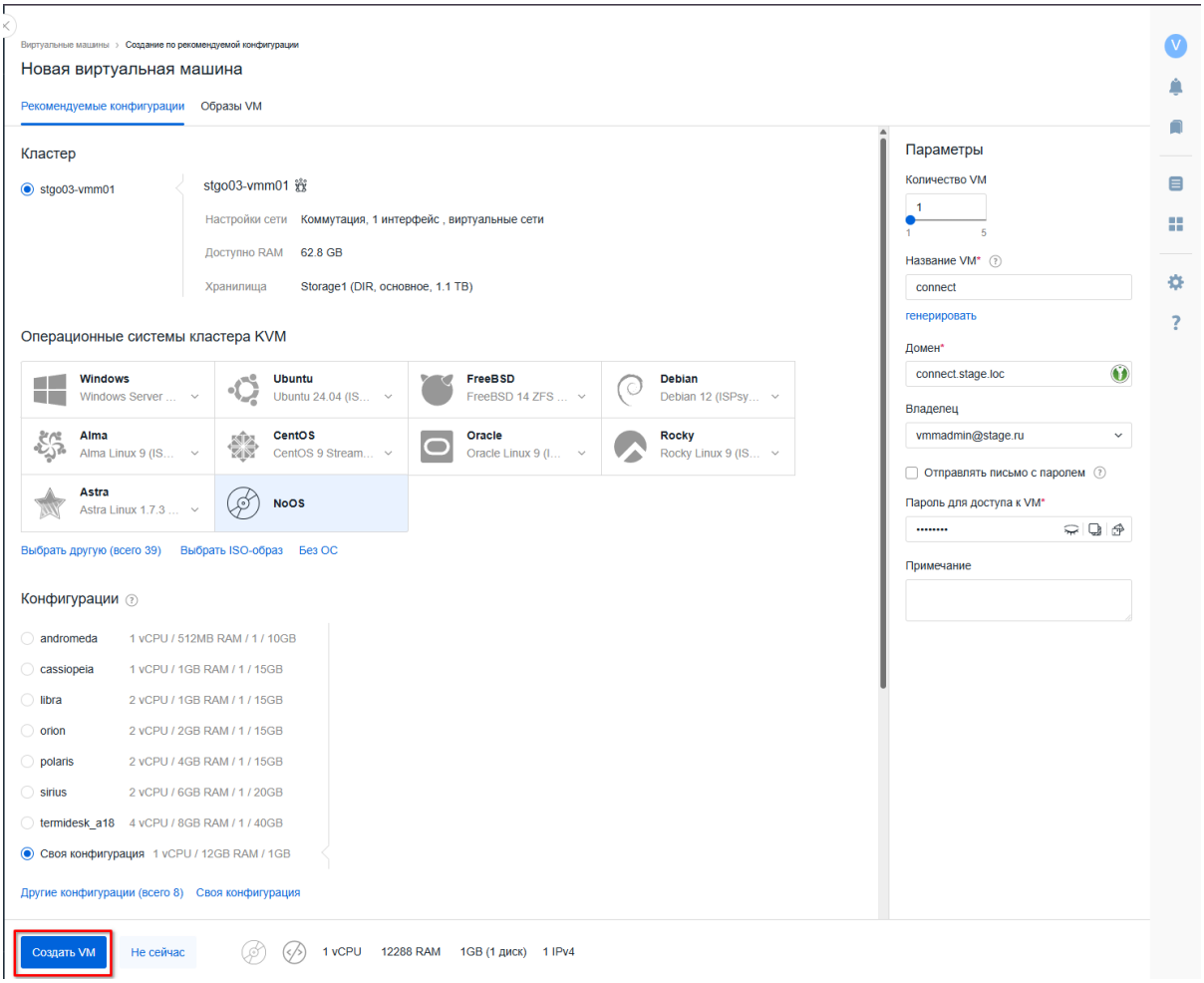

Рисунок 41. Создание ВМ

- выполнить копирование образа диска Termidesk Connect формата `.qcow2` на узел, где создана новая ВМ;
- в корневой директории `vm` заменить созданный файл VMmanager на скопированный образ диска Termidesk Connect (см. Замена файла);

Имена образов дисков Termidesk Connect не должны изменяться.

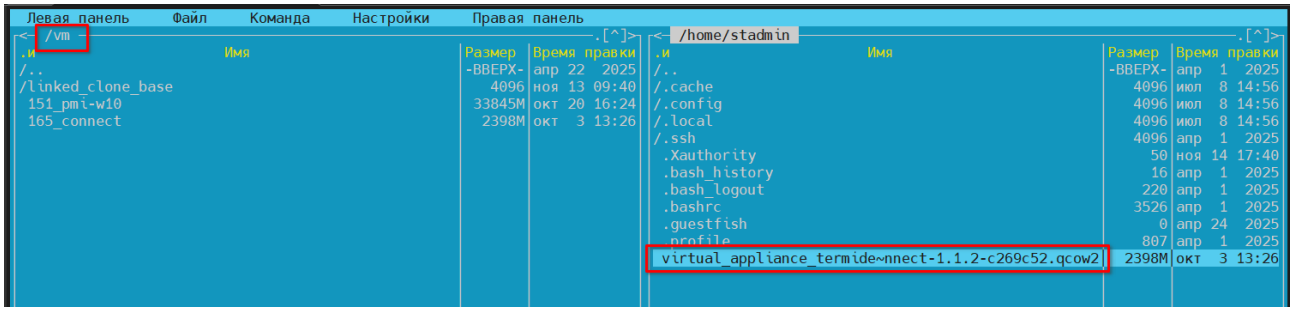

Рисунок 42. Замена файла

- выполнить запуск двойным нажатием левой кнопкой мыши по строке с созданной ВМ в списке (см. Запуск ВМ).

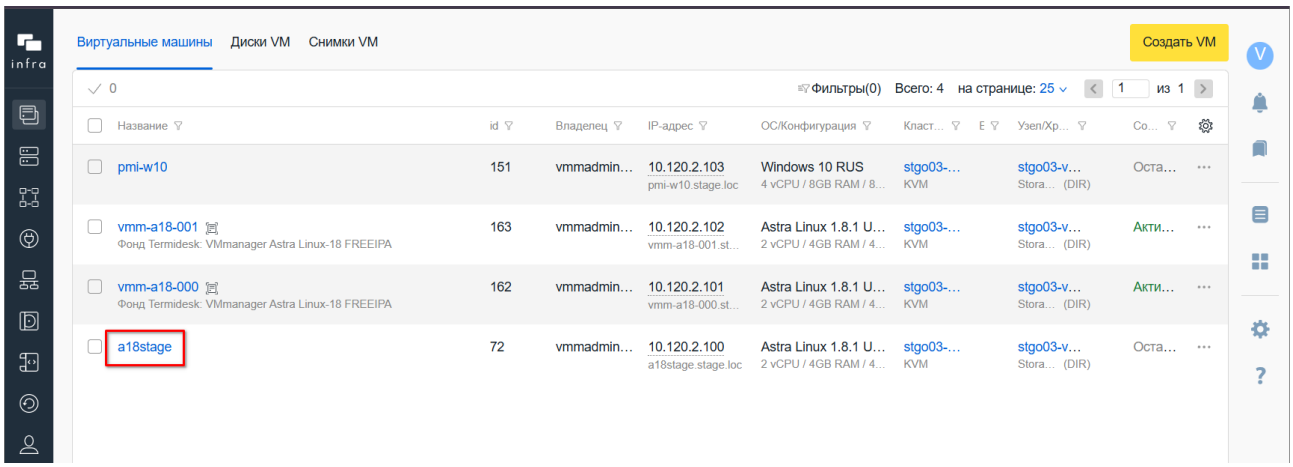

Рисунок 43. Запуск ВМ

После запуска ВМ:

- выбрать в меню пункт «Virtual Appliance Termidesk Connect» (по умолчанию) и нажать клавишу **<ENTER>** (см. Выбор варианта загрузки);

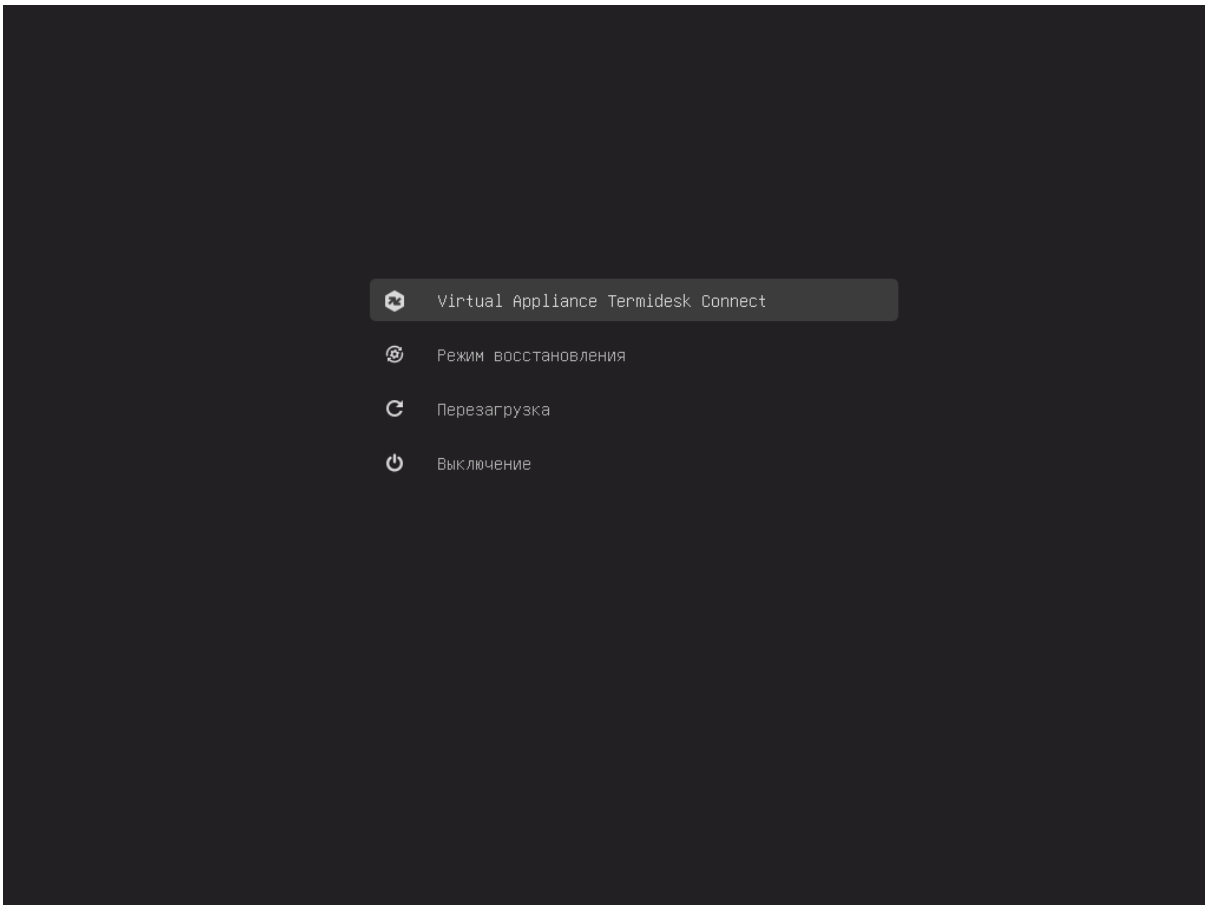

Рисунок 44. Выбор варианта загрузки

- выбрать загружаемый образ и нажать клавишу **<ENTER>** (см. Выбор загружаемого образа);

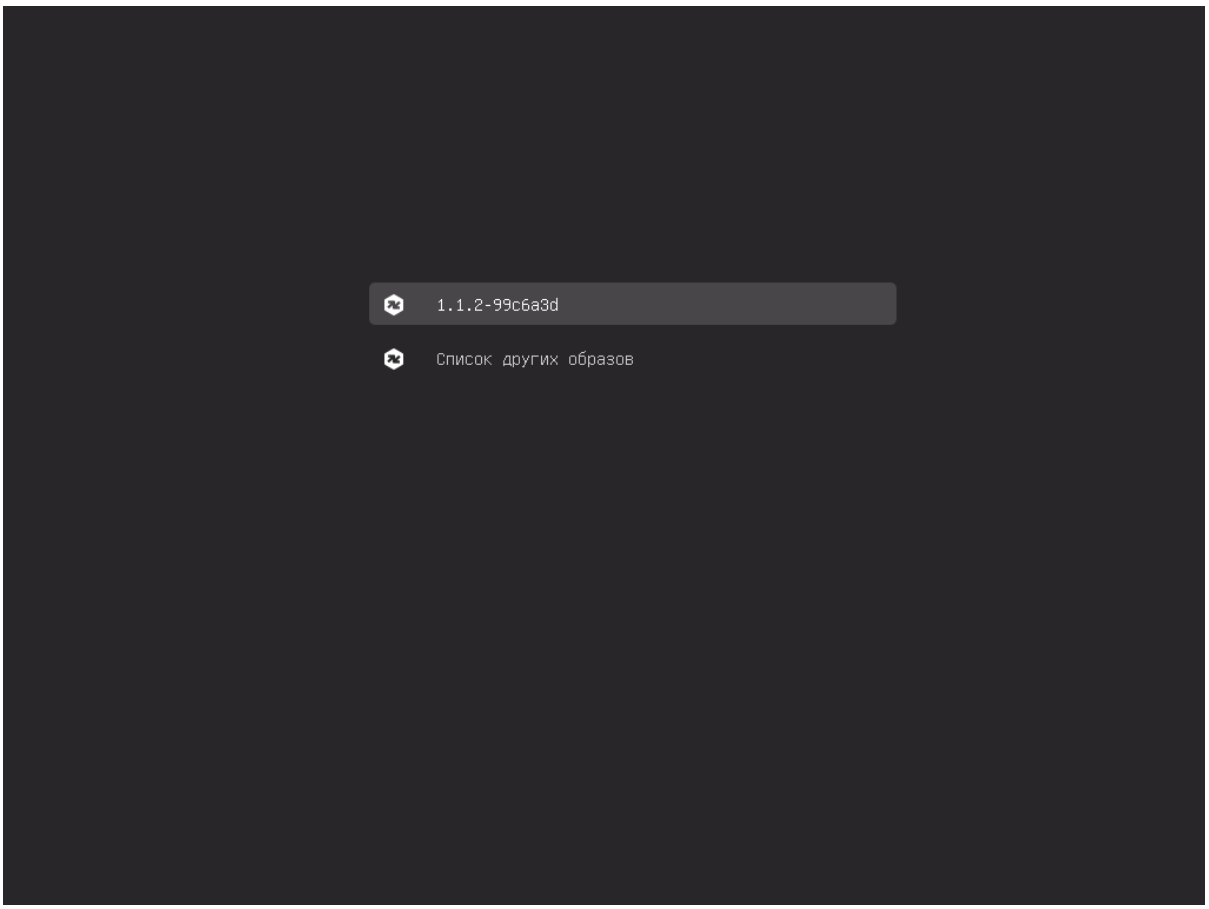

Рисунок 45. Выбор загружаемого образа

- прочитать и принять условия лицензионного соглашения, для этого переключитсья на экранную кнопку **[OK]** и нажать клавишу **<ENTER>** (см. Лицензионное соглашение);

Переключение между пунктами меню выполняется клавишей **<TAB>**. Подтверждение выбора выполняется клавишами **<ENTER>** или **<SPACE>**.

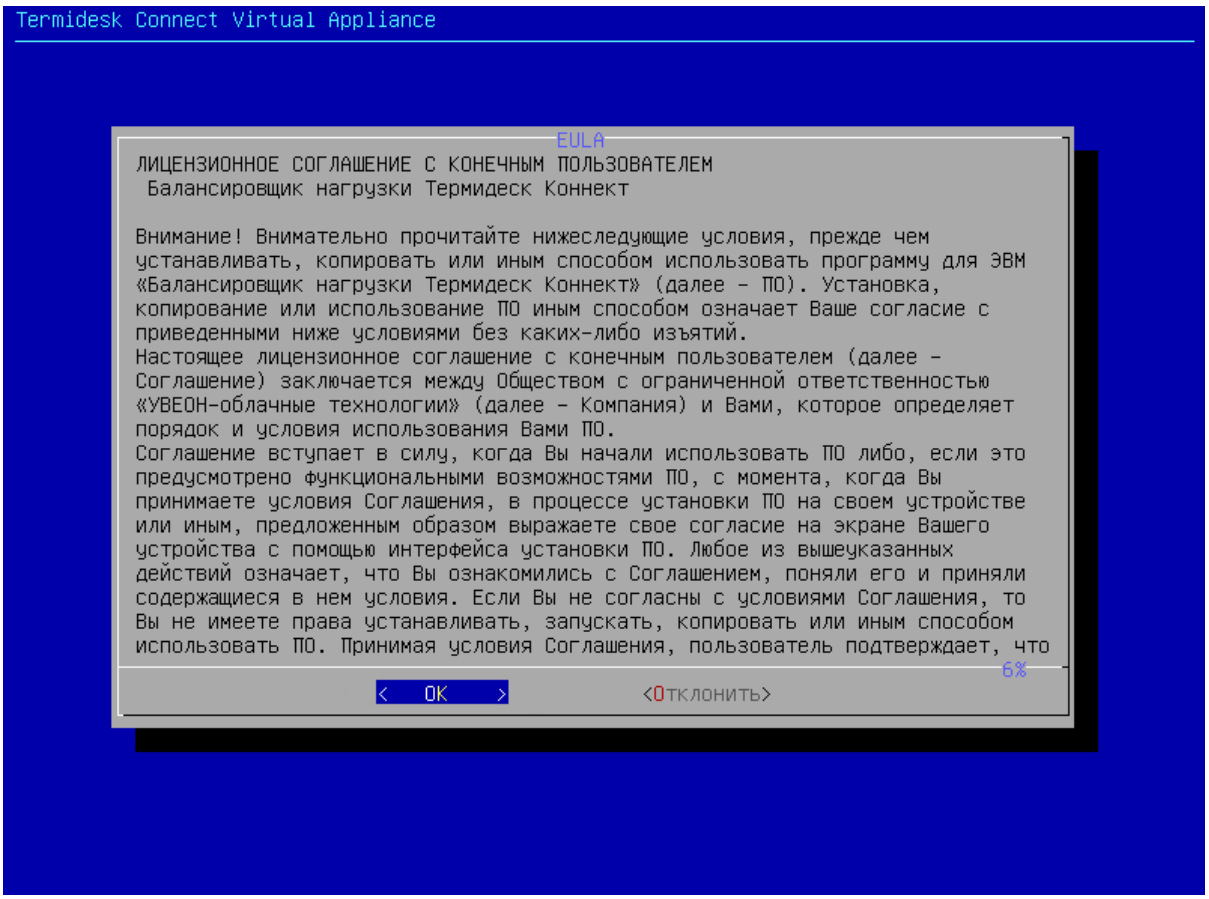

Рисунок 46. Лицензионное соглашение

- дождаться появления информационного сообщения о сертификатах и нажать клавишу **<ENTER>** (см. Информационное сообщение);

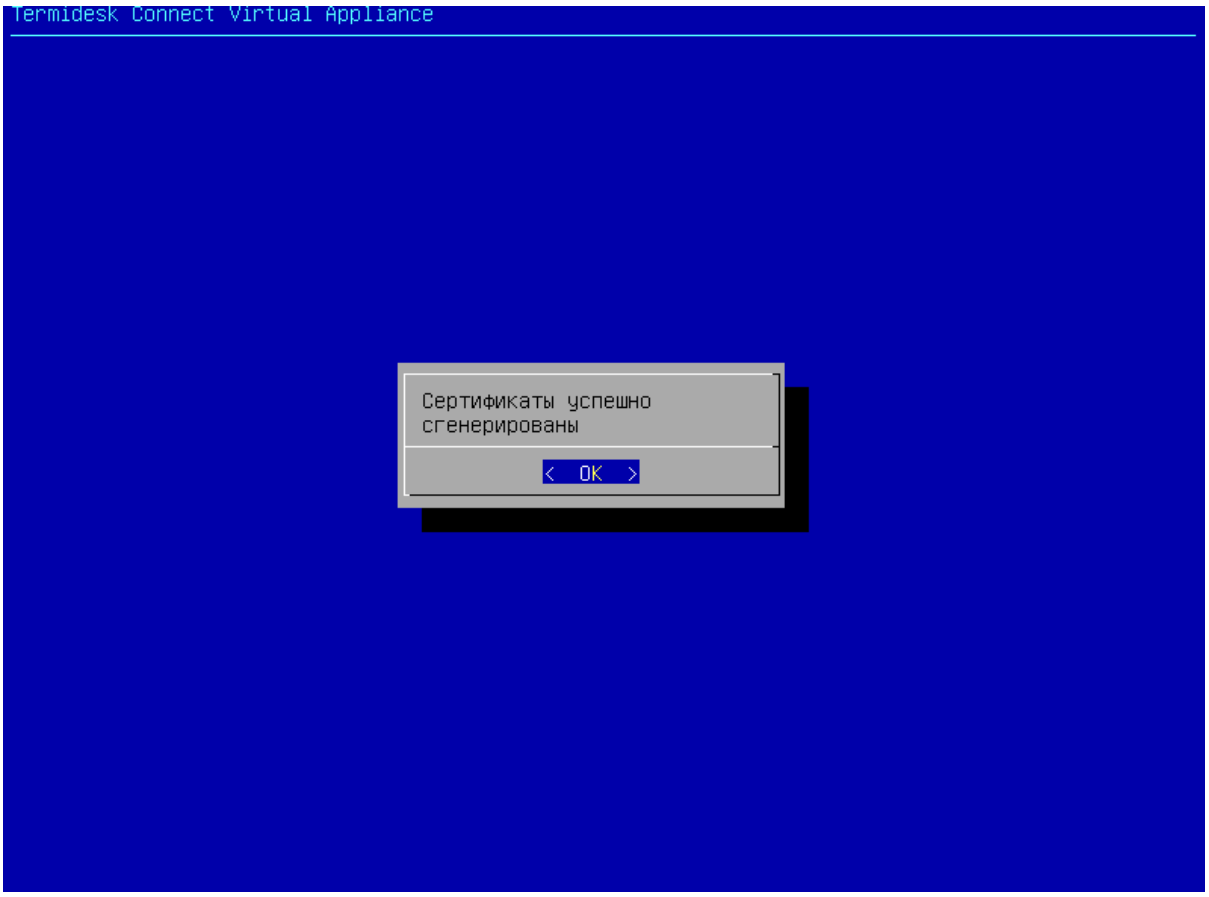

Рисунок 47. Информационное сообщение

- дождаться отображения главного окна Termidesk Connect (см. Главное окно Termidesk Connect). Затем выполнить настройку согласно подразделу **Первоначальная настройка Termidesk Connect**.

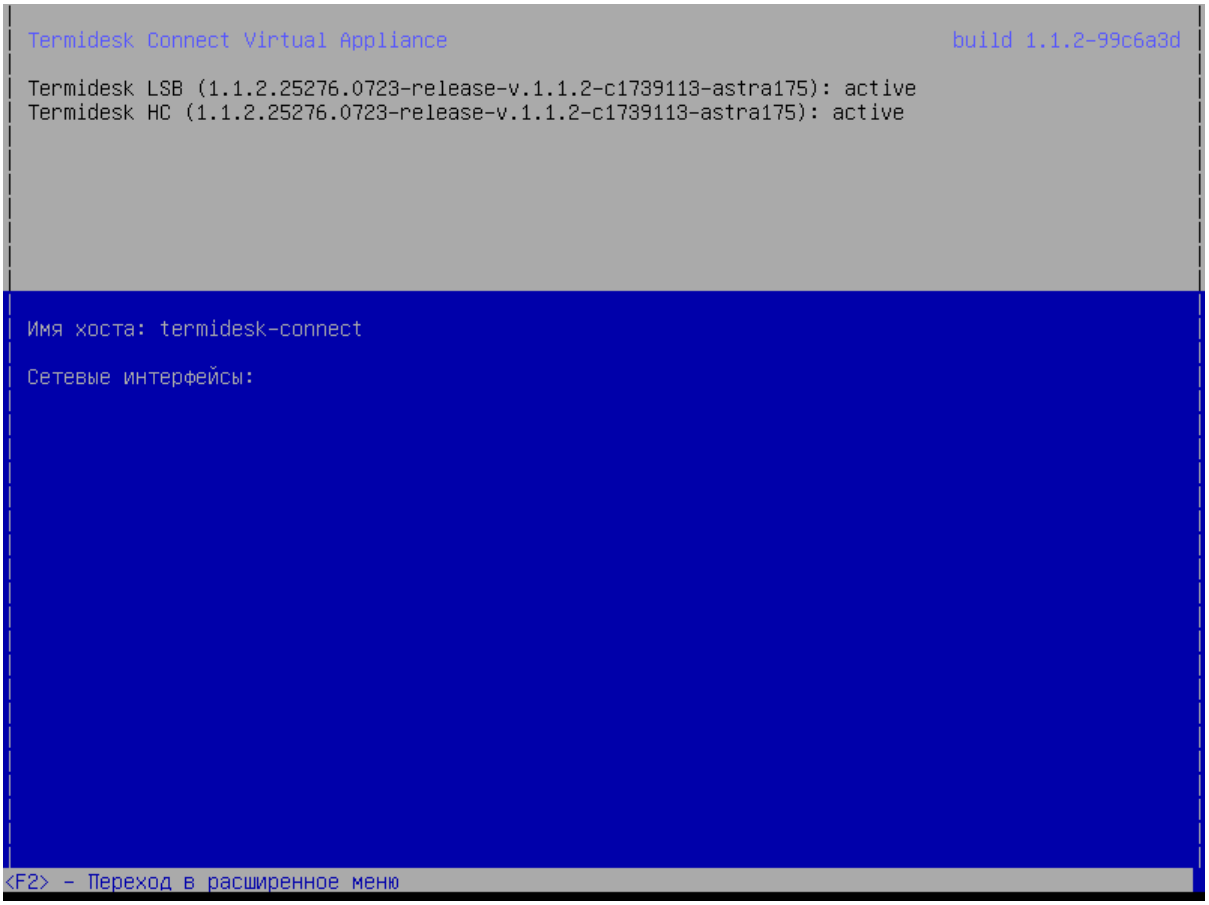

Рисунок 48. Главное окно Termidesk Connect

Автоматическая настройка при помощи `cloud-init`

`Cloud-init` – это инструмент, позволяющий применить параметры конфигурации и передать необходимые файлы (сертификаты, скрипты, токены) при запуске ВМ и тем самым ускорить и автоматизировать процесс настройки Termidesk Connect.

Поддерживаются следующие модули для проведения операций на разных этапах загрузки ВМ:

- `bootcmd`;
- `write_files`;
- `runcmd`;
- `scripts_user`;
- `eula` – дополнительный модуль Termidesk Connect;
- `tdc` – дополнительный модуль Termidesk Connect.

Для работы Termidesk Connect с `cloud-init` требуется подготовить обязательные файлы:

- `meta-data`, содержащий сведения об идентификаторе узла (`instance-id`) и его имени (`local-hostname`). Пример содержимого файла:

Параметры `instance-id` и `local-hostname` для каждого конкретного узла будут отличаться от приведенного примера.

```
instance-id: tdc-001
```

```
local-hostname: tdc-node
```

- `user-data`, содержащий команды настройки Termidesk Connect для запуска `cloud-init`. Пример содержимого файла:

```
#cloud-config

runcmd:
  - mount -o ro /dev/cdrom /mnt
  - cp /mnt/<cert_name>.crt /etc/ssl/tdc/
  - cp /mnt/<key_name>.key /etc/ssl/tdc/
  - chmod 644 /etc/ssl/tdc/<cert_name>.crt
  - chmod 600 /etc/ssl/tdc/<key_name>.key
  - chown root:root /etc/ssl/tdc/<cert_name>.crt /etc/ssl/tdc/<key_name>.key
  - umount /mnt

eula:
  accept: true

tdc:
  commands:
    - set ip address <ip> /<mask> if-ethernet <eth_name>
    - set ip route default <network> <gw>
    - set system mgmt ip <ip>
    - set system mgmt webui-port <port>
    - commit
    - write
```

После подготовки обязательных файлов нужно собрать `iso`-образ с ними командой:

Инструмент для сборки `iso`-образа может быть любой, в примере используется утилита `genisoimage`.

`CIDATA` или `cidata` – это идентификатор устройства (`label`) для доступа к нему. Параметр обязательно должен быть указан при сборке `iso`-образа.

```
genisoimage -output tdc.iso -volid CIDATA -joliet -rock meta-data user-data
```

После этого подготовленный `iso`-образ Termidesk Connect готов для добавления в инфраструктуру с поддержкой `cloud-init`.

Первоначальная настройка Termidesk Connect

Для первоначальной настройки Termidesk Connect нужно:

- в главном меню Termidesk Connect нажать клавишу **<F2>**, ввести текущий пароль администратора;

После установки по умолчанию используется логин `tdadmin` с паролем `tdadmin` для доступа к ряду функций управления Termidesk Connect. Необходимо сменить пароль после первоначальной настройки Termidesk Connect согласно подразделу **Смена пароля администратора** документа СЛЕТ.10101-01 90 02 «Руководство администратора. Настройка Termidesk Connect».

- далее выбрать пункт «CLI» и нажать клавишу **<ENTER>**. Отобразится строка приглашения `termidesk-connect#`, свидетельствующая об успешном переходе в интерфейс командной строки;
- выполнить настройку сетевых параметров:
  - настроить IP-адрес:

```
set ip address <IP-адрес> <длина_префикса_сети>
```

Пример команды:

```
set ip address 192.0.1.1 /24
```

  - назначить сетевой интерфейс для добавленного IP-адреса:

Сетевой интерфейс появляется в списке доступных автоматически после загрузки Termidesk Connect.

```
set ip address <IP-адрес> <длина_префикса_сети> if-ethernet <имя_интерфейса>
```

  - настроить маршрут по умолчанию:

```
set ip route default 0.0.0.0/0 <IP-адрес_шлюза>
```

- выполнить настройку параметров для доступа к интерфейсу управления Termidesk Connect:
  - назначить IP-адрес из списка заданных адресов, на котором будет доступен интерфейс управления:

Заданный IP-адрес будет использоваться как для доступа к веб-интерфейсу,так и для удаленного доступа к Termidesk Connect.

```
set system mgmt ip <IP-адрес>
```

  - назначить порт, на котором будет доступен веб-интерфейс Termidesk Connect:

```
set system mgmt webui-port <порт>
```

- применить заданные настройки:

```
commit
```

- сохранить заданные настройки:

```
write
```

После выполнения настроек Termidesk Connect будет доступен по протоколу SSH, также будет доступен веб-интерфейс управления.

Дальнейшая настройка Termidesk Connect приведена в документе СЛЕТ.10101-01 90 02 «Руководство администратора. Настройка Termidesk Connect».

ЗАВЕРШЕНИЕ РАБОТЫ

Завершение работы

Для завершения работы Termidesk Connect и выключения ВМ следует:

- перейти в интерфейс расширенного меню, нажав клавишу **<F2>** в главном меню Termidesk Connect;
- выбрать пункт «Выключение» и нажать клавишу **<Enter>**;
- подтвердить действие (см. Подтверждение выключения Termidesk Connect), нажав экранную кнопку **[Да]**.

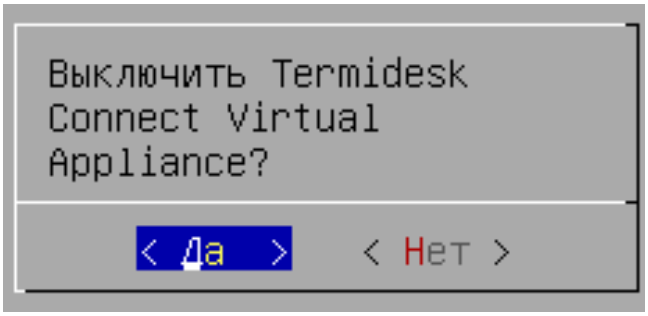

Рисунок 49. Подтверждение выключения Termidesk Connect

ТЕРМИНЫ

Full Proxy	Режим, при котором Termidesk Connect является посредником между клиентом и Реальным Сервером и полностью обрабатывает входящие и исходящие соединения
Termidesk (Termidesk VDI)	Программный комплекс «Диспетчер подключений виртуальных рабочих мест»
Termidesk Connect	Программа для электронной вычислительной машины «Балансировщик нагрузки Термидеск Коннект»
Реальный Сервер	Узел с установленным приложением, доставку которого обеспечивает Termidesk Connect
Группа Реальных Серверов	Объединение нескольких Реальных Серверов и их

периодических проверок

СОКРАЩЕНИЯ

ВМ	Виртуальная машина
ОС	Операционная система
ЦОД	Центр обработки данных
API	Application Programming Interface (программный интерфейс приложения)
EFI	Unified Extensible Firmware Interface (унифицированный расширяемый микропрограммный интерфейс)
HTTP	HyperText Transfer Protocol (протокол передачи гипертекста)
HTTPS	Hypertext Transfer Protocol Secure (расширение протокола HTTP для поддержки шифрования)
IP	Internet Protocol (межсетевой протокол)
L4	Четвертый (транспортный) уровень сетевой модели OSI
L7	Седьмой (прикладной) уровень сетевой модели OSI
NAT	Network Address Translation (преобразование сетевых адресов)
NETCONF	Network Configuration Protocol (протокол управления сетевыми устройствами)
OSI	The Open Systems Interconnection model (модель стека сетевых протоколов)
SSH	Secure Shell Protocol (протокол защищенной передачи информации)
TCP	Transmission Control Protocol (протокол управления передачей)
vCPU	Virtual Central Processing Unit (виртуальный центральный процессор)
WS	WebSocket (двунаправленный протокол, позволяющий клиенту установить связь с сервером)

СЛЕТ.101001-01 90 02. РУКОВОДСТВО АДМИНИСТРАТОРА. НАСТРОЙКА TERMIDESK CONNECT

КОМПОНЕНТЫ TERMIDESK CONNECT И ИХ ВЗАИМОДЕЙСТВИЕ

Основные компоненты и порядок их взаимодействия

Termidesk Connect состоит из группы объектов (компонентов):

- Виртуальный Сервер – представляет собой абстракцию, терминирующую на себя трафик от пользователя. Для Виртуального Сервера задаются IP-адрес, порт, тип, правила перенаправления трафика на Сервера Балансировки, правила модификации трафика и другие параметры. Помимо указанных параметров к Виртуальному Серверу привязываются:
  - SSL-Профиль, представляющий собой набор настроек протокола SSL (сертификат, ключ, используемые алгоритмы);
  - TCP-Профиль, представляющий собой набор настроек протокола TCP;
  - HTTP-Профиль, представляющий набор настроек протокола HTTP;
- Сервер Балансировки – представляет собой абстракцию, которая не имеет собственного IP-адреса и порта. Для Сервера Балансировки задаются тип, алгоритм балансировки, действия при недоступности Реальных Серверов, правила модификации трафика, а также привязываются одна или несколько Групп Реальных Серверов. Помимо указанных параметров к Серверу Балансировки привязываются:
  - SSL-Профиль;
  - TCP-Профиль;
  - HTTP-Профиль;
- группа Реальных Серверов – объединение нескольких Реальных Серверов и их периодических Проверок. Реальный Сервер – это узел с установленным приложением, доставку которого обеспечивает Termidesk Connect. Для включения Реального Сервера в список доставки Termidesk Connect задаются его IP-адрес, порт.

Помимо перечисленных компонентов в состав включаются следующие, если Termidesk Connect используется для геораспределенной балансировки:

- Виртуальный Сервер геобалансировки – представляет собой абстракцию, терминирующую на себя трафик. Для него задаются алгоритм балансировки, время жизни (TTL) и другие параметры;
- Сервис – это объект (абстракция) с назначенным IP-адресом, доступ к которому нужно предоставить пользователю в рамках геобалансировки. Для Сервиса задается локальный IP-адрес, отправляющийся в DNS-ответе, если пользователь подключается из внутренней сети организации, и общедоступный IP-адрес для остальных случаев.

Процесс подключения пользователя выглядит следующим образом (см. Общий принцип взаимодействия):

- запрос от пользователя приходит на Виртуальный Сервер;
- Виртуальный Сервер на основе заданных настроек определяет, как будет обработан поступивший запрос;
- далее на основании правил балансировки уровней L4 или L7 запрос передается на Сервер Балансировки;
- Сервер Балансировки на основе заданных настроек балансировки обрабатывает запрос, выполняет проверку Группы Реальных Серверов и направляет подключение на один из Реальных Серверов для доступа к приложению.

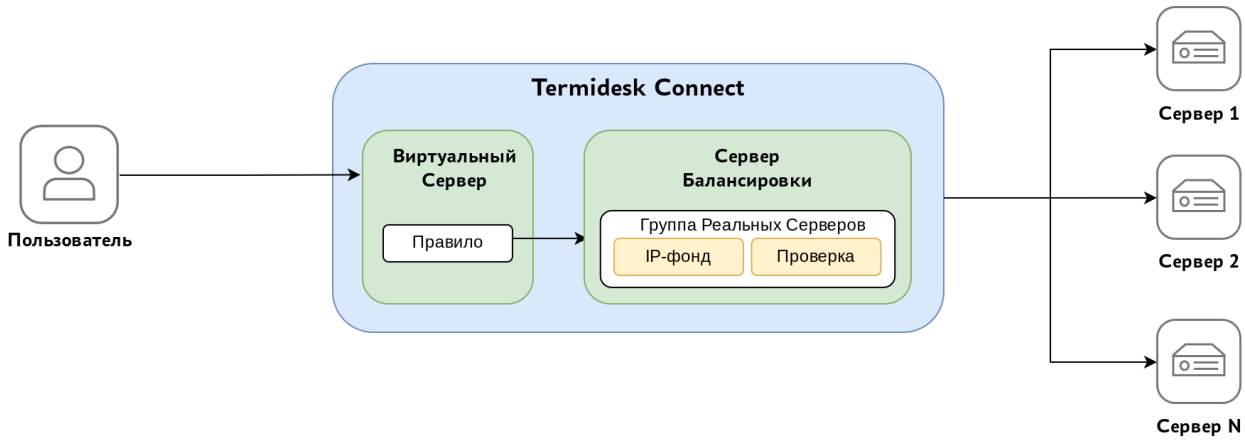

Рисунок 50. Общий принцип взаимодействия

Хранилища конфигураций Termidesk Connect

Termidesk Connect использует несколько хранилищ конфигураций для настройки и управления заданными параметрами:

- предварительная конфигурация («candidate») – представляет собой временное хранилище конфигурации, в котором вносятся изменения перед их применением. Изменения в такой конфигурации можно редактировать, проверять и отменять без влияния на рабочую конфигурацию Termidesk Connect. Чтобы изменения вступили в силу, их нужно записать командой `commit`;
- рабочая конфигурация («running») – представляет собой конфигурацию, в данный момент использующуюся Termidesk Connect. Рабочая конфигурация будет обновлена, если для предварительной конфигурации выполнить команду `commit`. Для сохранения изменений в рабочей конфигурации используется команда `write`. Если сохранение изменений не будет выполнено, то при перезагрузке Termidesk Connect изменения в рабочей конфигурации будут сброшены;
- сохраненная конфигурация («startup») – представляет собой конфигурацию, загружаемую при старте Termidesk Connect. После перезагрузки Termidesk Connect сохраненная конфигурация копируется в рабочую. Чтобы рабочая конфигурация была загружена при следующем запуске (стала сохраненной конфигурацией), нужно выполнить команду `write`.

Взаимодействие хранилищ конфигураций при настройке Termidesk Connect из интерфейса командной строки, приведено на рисунке (см. Взаимодействие хранилищ конфигурации).

При настройке Termidesk Connect из веб-интерфейса изменение параметров применяется сразу же и на рабочую конфигурацию («running»). Запись рабочей конфигурации в сохраненную («startup») осуществляется через экранную кнопку **[Сохранить]**.

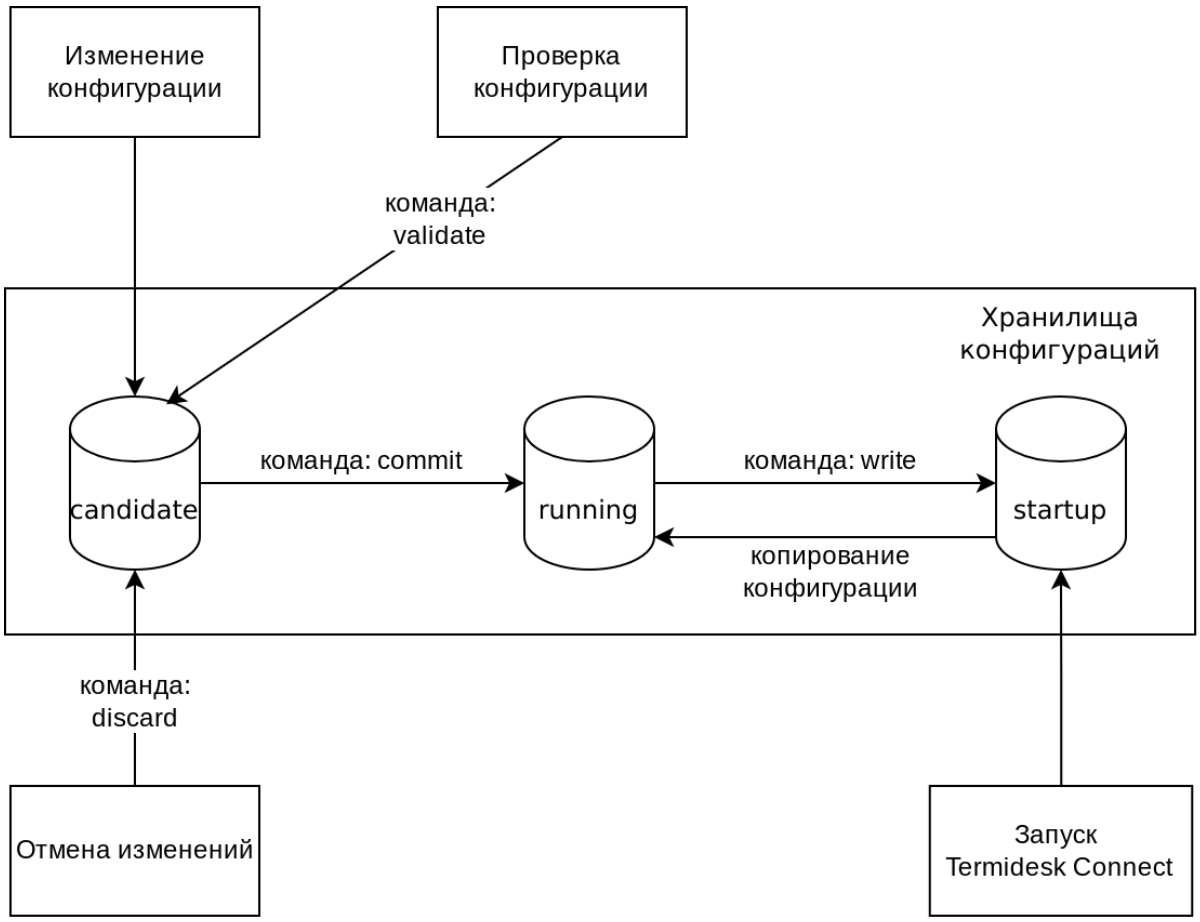

Рисунок 51. Взаимодействие хранилищ конфигурации

НАСТРОЙКИ СИСТЕМЫ

Лицензионное соглашение Termidesk Connect

Принятие условий лицензионного соглашения осуществляется при загрузке и первоначальной настройке Termidesk Connect.

Получение лицензии приведено в поздразделе **Лицензирование Termidesk Connect**.

Лицензия может быть добавлена в конфигурацию Termidesk Connect одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Лицензионное соглашение**).

Для добавления лицензии используются команды:

- загрузка манифеста в формате `.zip` на устройство:

```
scp -P 222 <имя_манифеста>.zip tdadmin@<IP-адрес_Termidesk_Connect>:~/
```

- перемещение манифеста в хранилище лицензий:

В Termidesk Connect хранилище лицензий расположено в папке `/var/lib/tdc/license`.

```
sudo cp <имя_манифеста>.zip /var/lib/tdc/license
```

- активация лицензии на устройстве:

```
set system license <имя_манифеста>.zip
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

- просмотр параметров используемой лицензии:

```
show license attributes

HWID: 1AE02A267D277A5D48DB22F49DDFBF90
Customer: test-termidesk-connect-integration
Redaction: termidesk_connect
Since: 2025-11-12T00:00:00+0000
Until: 2026-11-11T23:59:59+0000
End version: 1.2
CPU: 1
```

DNS

В сетевой инфраструктуре может быть настроен DNS-сервер, обеспечивающий разрешение доменных имен в IP-сетях.

Действующий DNS-сервер может быть добавлен в конфигурацию Termidesk Connect одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. DNS**).

Для добавления DNS-сервера используются команды:

Описание параметров также приведено в подразделе **Объект** `dns`.

- добавление DNS-сервера:

```
set dns ip-address <IP-адрес>
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

Дополнительно можно указать в конфигурации Termidesk Connect домен поиска для автоматического добавления доменного суффикса при попытке разрешения неполного доменного имени.

Для этого используются команды:

- добавление домена поиска:

```
set dns search-domain <имя_домена>
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

NTP

В сетевой инфраструктуре должен быть настроен NTP-сервер, обеспечивающий синхронизацию времени для компонентов Termidesk Connect.

Действующий NTP-сервер должен быть добавлен в конфигурацию Termidesk Connect одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect.

Для добавления NTP-сервера используются команды:

Описание параметров также приведено в подразделе **Объект** `ntp`.

- добавление NTP-сервера:

```
set ntp server <IP-адрес_или_FQDN>
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

Отказоустойчивость

Общие сведения об отказоустойчивости Termidesk Connect

Для обеспечения постоянства доступа пользователя к приложениям, размещенным в инфраструктуре организации, Termidesk Connect может быть настроен в отказоустойчивой (высокодоступной) конфигурации.

Для настройки отказоустойчивой (высокодоступной) конфигурации необходимо использовать от двух узлов Termidesk Connect (поддерживается до 255 узлов).

Termidesk Connect в отказоустойчивой конфигурации обеспечивает:

- синхронизацию конфигураций между двумя и более узлами Termidesk Connect. При этом синхронизируется не вся конфигурация: часть параметров является индивидуальной для каждого узла;
- резервирование IP-адресов Виртуальных Серверов;
- переключение трафика между узлами Termidesk Connect при выходе какого-либо из них из строя.

Для обеспечения отказоустойчивости устройства Termidesk Connect объединяются в одноранговый кластер и имеют одинаковые права и возможности. Однако для упрощения настройки принято, что существуют следующие узлы:

- мастер-узел («Active») – устройство обрабатывает трафик пользователей в настоящий момент;
- резервный узел («Standby») – устройство в настоящий момент не обрабатывает трафик пользователей. Резервных узлов может быть несколько.

Единовременно может работать только один мастер-узел, все остальные являются резервными.

Решение о назначении мастер-узла принимается на основе его рейтинга:

- все узлы обмениваются своим рейтингом;
- рейтинг узла конвертируется в число. Чем больше число, тем выше рейтинг;
- когда узел становится мастер-узлом, в его структуре данных выставляется определенный бит. Если узел будет назначен активным вручную, то его рейтинг будет выше, чем у текущего мастер-узла, и он будет считаться главным мастер-узлом;
- узел, который был перезагружен (или отключен), будет иметь наименьший рейтинг.

Управление кластером осуществляется через общий IP-адрес с типом «ha-

type SHARED», добавляемый на мастер-узле.

Все параметры конфигурации синхронизируются между узлами Termidesk Connect, кроме указанных:

- IP-адрес с типом «ha-type LOCAL» или привязанный к VLAN с типом «ha-type LOCAL»;
- имя устройства (hostname);
- VLAN с типом «ha-type LOCAL»;
- маршруты с типом «ha-type LOCAL».

Удаление любых скриптов или файлов в синхронизируемых директориях на мастер-узле не приводит к их удалению на резервном узле.

Общий алгоритм настройки отказоустойчивой (высокодоступной) конфигурации (примеры команд приведены в подразделе **Настройка отказоустойчивой конфигурации**):

- убедиться, что устройства Termidesk Connect имеют сетевую доступность между собой;
- убедиться, что наименования сетевых интерфейсов для устройств Termidesk Connect идентичны на всех узлах;
- убедиться, что версия Termidesk Connect одинакова на всех узлах кластера;
- убедиться, что доступны следующие порты для взаимодействия:
  - TCP 222;
  - TCP/UDP 322. Этот порт может быть изменен на другой;
- настроить мастер-узел Termidesk Connect:
  - (опционально) сконфигурировать IP-адрес управления кластером с типом «ha-type SHARED» (настройка IP-адреса приведена в подразделе **Сеть**);

    Интерфейс, на который назначен локальный IP-адрес, должен быть сконфигурирован с VRF `default`.

  - указать уникальный числовой идентификатор мастер-узла;
  - указать локальный IP-адрес, используемый для взаимодействия с другим устройством Termidesk Connect. Выбранный IP-адрес должен соответствовать типу «ha-type LOCAL» (настройка IP-адреса приведена в подразделе **Сеть**);

    Интерфейс, на который назначен локальный IP-адрес, должен быть сконфигурирован с VRF `default`.

  - указать UDP-порт, используемый для взаимодействия;
  - указать интервал периодических запросов. Такой же интервал должен быть задан при настройке резервного узла Termidesk Connect;
  - указать параметры подключения к резервному узлу;
- настроить зеркально резервные узлы Termidesk Connect:
  - указать уникальный числовой идентификатор резервного узла;
  - указать локальный IP-адрес, используемый для взаимодействия с другим устройством Termidesk Connect. Выбранный IP-адрес должен соответствовать типу «ha-type LOCAL» (настройка IP-адреса приведена в подразделе **Сеть**);

    Интерфейс, на который назначен локальный IP-адрес, должен быть сконфигурирован с VRF `default`.

- указать UDP-порт, используемый для взаимодействия;
- указать интервал периодических запросов, который был задан на мастер-узле Termidesk Connect;
- указать параметры подключения к мастер-узлу.

Условия переключения узлов отказоустойчивой конфигурации

В отказоустойчивом кластере Termidesk Connect принятие решения о переключении узлов (`failover`) может осуществляться не только по недоступности мастер-ноды. Реализована опциональная возможность анализа состояния мастер-узла на основе объектов отслеживания. Невыполнение любого из условий отслеживания объектов (при условии исправности резервного узла) инициирует процесс переключения.

Переключение происходит по порядку, согласно рейтингу узла. В случае недоступности мастер-узел понижает свой рейтинг и происходит переключение на новый узел из списка резервных. Переключение осуществляется на резервный узел с большим рейтингом. При попытке принудительного переключения узла в состояние `ACTIVE` с меньшим рейтингом переключение не произойдет.

Для оценки состояния мастер-узла может отслеживаться:

- состояние сетевых интерфейсов. Если сетевой интерфейс переходит в состояние `DOWN`, то узел считается неактивным. Позволяет предотвратить ситуацию, когда узел остается активным, но теряет связность из-за падения интерфейса. Для включения механизма сетевой интерфейс должен быть помечен как отслеживаемый;
- доступность IP-адресов. Если один или несколько IP-адресов перестали быть доступными (не ответили на ICMP-запрос) в течение заданного количества попыток, то узел считается неактивным. Для включения механизма IP-адрес должен быть добавлен в список отслеживаемых;
- состояние Сервера Балансировки. Если состояние Сервера Балансировки на мастер-узле меняется на `OFFLINE`, а на резервной ноде оно `ONLINE`, то текущий мастер-узел считается неактивным. Для включения механизма Сервер Балансировки должен быть помечен как отслеживаемый.

Для включения функционала оценки состояния мастер-узла:

- в настройках следующих объектов используется параметр `ha-monitor`:
  - сетевого интерфейса (`ethernet`);
  - Проверки (`health-check`);
  - Сервера Балансировки (`lbs`);
- на каждом узле отказоустойчивой конфигурации доступна настройка отслеживаемых IP-адресов (параметр `ip-monitor`)

Настройка отказоустойчивой конфигурации

Настройка выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Отказоустойчивость**).

Для настройки используются команды:

Описание параметров также приведено в подразделе **Объект `ha`**.

Команды выполняются на каждом узле отказоустойчивого кластера.

- задание локальных настроек узла:
  - задание уникального числового идентификатора узла для отказоустойчивой конфигурации:

Уникальные числовые идентификаторы не должны повторяться в одноранговом кластере! Узел с минимальным идентификатором будет мастер-узлом.

```
set ha id <идентификатор>
```

  - (опционально) указание общего IP-адреса управления кластером:

Общий IP-адрес управления кластером:

- указывается только на мастер-узле;
- должен быть создан и настроен на мастер-узле, и должен соответствовать типу «ha-type SHARED» (см. настройку IP-адресов в подразделе **Сеть**);
- интерфейс, на который назначен локальный IP-адрес, должен быть сконфигурирован с VRF `default`.

```
set ha cluster-ip <IP-адрес>
```

  - указание локального IP-адреса узла для отказоустойчивой конфигурации:

Выбранный IP-адрес должен соответствовать типу «ha-type LOCAL» (см. настройку IP-адресов в подразделе **Сеть**).

Интерфейс, на который назначен локальный IP-адрес, должен быть сконфигурирован с VRF `default`.

```
set ha ip <IP-адрес>
```

  - задание UDP-порта узла для отказоустойчивой конфигурации (по умолчанию – `322`):

```
set ha port <порт>
```

  - задание интервала (в миллисекундах) периодических запросов к узлам для отказоустойчивой конфигурации (по умолчанию – `500`):

Интервал должен быть одинаковым на всех узлах.

```
set ha interval <значение>
```

- задание настроек соединения с узлом Termidesk Connect, с которым будет осуществляться взаимодействие (удаленным узлом):
  - задание имени удаленного узла Termidesk Connect:

```
set ha remote <имя_узла>
```

  - указание идентификатора удаленного узла Termidesk Connect:

```
set ha remote <имя_узла> id <идентификатор>
```

  - задание IP-адреса удаленного узла Termidesk Connect:

```
set ha remote <имя_узла> ip <IP-адрес>
```

  - задание UDP-порта удаленного узла Termidesk Connect:

```
set ha remote <имя_узла> port <порт>
```

  - задание пароля пользователя удаленного узла Termidesk Connect:

Должен быть указан пароль пользователя `tdadmin` удаленного узла.

Так же пароль может быть получен из Хранилища секретов. Пример значения параметра:

- `kv://secret/ha#password` – для `kv` версии 1;
- `kv://secret/data/na#password` – для `kv` версии 2.

```
set ha remote <имя_узла> seckey <пароль>
```

- (опционально) задание отслеживаемых IP-адресов для готовности узла к переходу в состояние `ACTIVE`:

Может быть задано несколько IP-адресов. Подробное описание условий переключения узлов отказоустойчивой конфигурации приведено в подразделе **Условия переключения узлов отказоустойчивой конфигурации**.

  - задание IP-адреса:

```
set ha ip-monitor <IP-адрес>
```

  - (опционально) задание VRF для IP-адреса (по умолчанию – `default`):

```
set ha ip-monitor <IP-адрес> vrf <имя_VRF>
```

- проверка заданных настроек:

```
validate
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml ha
```

- просмотр выполненных команд:

```
show configuration cli ha
```

В таблице (см. Пример команд для отказоустойчивой конфигурации) приведен пример команд настройки для трех устройств Termidesk Connect.

В примере 192.0.2.1 – общий IP-адрес управления кластером, который создан на мастер-узле NodeTDC15-1.

Таблица 1. Пример команд для отказоустойчивой конфигурации

Мастер-узел NodeTDC15-1	Резервный узел NodeTDC15-2	Резервный узел NodeTDC15-3
Локальные настройки узлов		
`set ha id 1`	`set ha id 2`	`set ha id 3`
`set ha ip 192.0.2.10`	`set ha ip 192.0.2.11`	`set ha ip 192.0.2.12`
`set ha port 322`	`set ha port 322`	`set ha port 322`
`set ha interval 500`	`set ha interval 500`	`set ha interval 500`
`set ha cluster-ip 192.0.2.1`	-	-
Настройка взаимодействия с другими узлами		
`set ha remote NodeTDC15-2 id 2`	`set ha remote NodeTDC15-1 id 1`	`set ha remote NodeTDC15-1 id 1`

Мастер-узел NodeTDC15-1	Резервный узел NodeTDC15-2	Резервный узел NodeTDC15-3
`set ha remote NodeTDC15-2 ip 192.0.2.11`	`set ha remote NodeTDC15-1 ip 192.0.2.10`	`set ha remote NodeTDC15-1 ip 192.0.2.10`
`set ha remote NodeTDC15-2 port 322`	`set ha remote NodeTDC15-1 port 322`	`set ha remote NodeTDC15-1 port 322`
`set ha remote NodeTDC15-2 seckey P@ss`	`set ha remote NodeTDC15-1 seckey P@ss`	`set ha remote NodeTDC15-1 seckey P@ss`
`set ha remote NodeTDC15-3 id 3`	`set ha remote NodeTDC15-3 id 3`	`set ha remote NodeTDC15-2 id 2`
`set ha remote NodeTDC15-3 ip 192.0.2.12`	`set ha remote NodeTDC15-3 ip 192.0.2.12`	`set ha remote NodeTDC15-2 ip 192.0.2.11`
`set ha remote NodeTDC15-3 port 322`	`set ha remote NodeTDC15-3 port 322`	`set ha remote NodeTDC15-2 port 322`
`set ha remote NodeTDC15-3 seckey P@ss`	`set ha remote NodeTDC15-3 seckey P@ss`	`set ha remote NodeTDC15-2 seckey P@ss`
Проверка и применение конфигурации		
`validate`	`validate`	`validate`
`commit`	`commit`	`commit`
`write`	`write`	`write`
Принудительный перевод узла в активное состояние (необязательно при первоначальной настройке)		
`ha set-active`	-	-

Проверка состояний узлов в отказоустойчивой конфигурации выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect:

```
show status ha
```

- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Отказоустойчивость**).

Вывод команды отобразит:

- роль узла, на котором выполнена команда:
  - `STANDBY` – резервный узел, не обрабатывающий в данный момент трафик пользователей;
  - `ACTIVE` – мастер-узел, обрабатывающий в данный момент трафик пользователей;
- информацию об узле, на котором выполнена команда: идентификатор, IP-адрес, порт, дата и время последней синхронизации, статус синхронизации;
- информацию об удаленном узле, с которым настроена отказоустойчивая конфигурация: имя узла, его доступность, роль узла, идентификатор, IP-адрес, порт, дата и время последней синхронизации,статус синхронизации.

Если статус синхронизации соответствует `false`, необходимо подключиться к узлу с ролью

`STANDBY` и выполнить поиск причин ошибки в журналах.

Статус синхронизации `false` означает, что синхронизация конфигурации между узлами не выполнена. Такая ситуация может возникнуть, например, если при внесении изменений в настройках ADNS был указан не общий IP-адрес кластера.

Пример вывода команды:

```
########################
#  Отказоустойчивость  #
########################

               Роль: ACTIVE
                 ID: 0
          IP-адрес: 192.0.2.1
              Порт: 322
        Время синх: 2025-08-11 17:46:06.749 +0000
       Статус синх: true

============================================================
(1)            Узел: node2
       Доступность: доступен
               Роль: STANDBY
                 ID: 1
          IP-адрес: 192.0.2.2
              Порт: 322
        Время синх: 2025-08-11 17:46:06.749 +0000
       Статус синх: true
```

Сеть

Общие сведения по настройке сети

Для интеграции Termidesk Connect в существующую сеть организации нужно выполнить настройку:

- VRF (опционально);
- сетевых интерфейсов;
- агрегации каналов (опционально);
- VLAN;
- IP-адресов;
- маршрутизации;
- IP-Фондов (опционально);
- динамической маршрутизации (опционально);
- ARP (опционально).

Настройка VRF

Настройка VRF является необязательной и выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Сеть**).

Для создания VRF используются команды:

Описание параметров также приведено в подразделе **Объект** `vrf`.

- создание VRF и назначение таблицы маршрутизации:

```
set vrf name <имя> table-id <номер_таблицы>
```

Номер таблицы задает идентификатор таблицы маршрутизации, которая будет ассоциирована с этим VRF. Номер таблицы должен быть уникальным.

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml vrf name <имя>
```

- просмотр выполненных команд:

```
show configuration cli vrf name <имя>
```

Настройка интерфейсов

Настройка интерфейсов выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Сеть**).

Имя интерфейса определяется автоматически после загрузки Termidesk Connect. MAC-адрес также будет назначен автоматически.

Для настройки интерфейсов используются команды:

Описание параметров также приведено в подразделе **Объект** `ethernet`.

- (опционально) назначение VRF сетевому интерфейсу:

```
set ethernet name <имя> vrf <имя_VRF>
```

- (опционально) назначение DDoS-Профиля сетевому интерфейсу:

```
set ethernet name <имя> ddos-profile-id <DDoS-Профиль>
```

- (опционально) привязка сетевого интерфейса к агрегированному каналу:

```
set ethernet name <имя> agg <имя_агрегированного_интерфейса>
```

- (опционально) назначение списка контроля доступа сетевому интерфейсу:

```
set ethernet name <имя> acl-id <имя_списка>
```

- (опционально) задание MTU (значение от 0 до 9216, по умолчанию – `1500`):

```
set ethernet name <имя> mtu <значение>
```

- (опционально) активация или отключение сетевого интерфейса (по умолчанию – `ENABLED`):

```
set ethernet name <имя> state <ENABLED/DISABLED>
```

- (опционально) включение или отключение отслеживания состояния интерфейса для готовности узла к переходу в состояние `ACTIVE` (по умолчанию – `false`):

Подробное описание условий переключения узлов отказоустойчивой конфигурации приведено в подразделе **Отказоустойчивость**.

```
set ethernet name <имя> USER ha-monitor <true/false>
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml ethernet name <имя>
```

- просмотр выполненных команд:

```
show configuration cli ethernet name <имя>
```

Настройка агрегации каналов

Агрегированный канал объединяет несколько физических сетевых интерфейсов в один логический LAG-интерфейс, что позволяет повысить пропускную способность сетевого соединения и обеспечить его отказоустойчивость.

Termidesk Connect поддерживает протокол LACP, описанный в стандарте IEEE 802.3ad, для настройки LAG-интерфейсов.

Настройка агрегации каналов выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Сеть**).

Для настройки агрегации каналов используются команды:

Описание параметров также приведено в подразделе **Объект** `aggregation`.

- создание LAG-интерфейса:

```
set aggregation name <имя>
```

- назначение типа LAG-интерфейса:

Возможные значения:

- `LACP` – будет использоваться протокол LACP. После того как агрегированный канал сформирован, за поддержание статуса канала отвечает LACP;
- `Active-backup` – будет использоваться только активный интерфейс из объединенных. При отказе активного интерфейса выполняется автоматическое переключение на резервный интерфейс.

Не допускается изменение типа для уже созданного и настроенного интерфейса.

Для использования типа LACP он должен поддерживаться сетевым коммутатором.

```
set aggregation name <имя> type <тип>
```

- (опционально, если задан тип LACP) назначение режима работы LAG-интерфейса (по умолчанию –slow):

Возможные значения:

- fast – режим работы, при котором частота отправки LACPDU-пакетов составляет один раз в секунду для более быстрого обнаружения изменений в сети;
- slow – режим работы, при котором частота отправки LACPDU-пакетов составляет один раз в 30 секунд.

```
set aggregation name <имя> lacp-rate <режим>
```

- (опционально, если задан тип Active-backup) назначение частоты (в миллисекундах) мониторинга MII (Media Independent Interface) (по умолчанию – 100):

```
set aggregation name <имя> miimon <значение>
```

- (опционально) назначение VRF для LAG-интерфейса:

```
set aggregation name <имя> vrf <имя_VRF>
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

- просмотр заданных настроек (указывается формат вывода – XML, JSON или TXT):

```
show configuration xml aggregation name <имя>
```

- просмотр выполненных команд:

```
show configuration cli aggregation name <имя>
```

После настройки LAG-интерфейса требуется назначить его физическим сетевым интерфейсам (см. настройку интерфейсов в подразделе **Сеть**).

Также может потребоваться настройка VLAN (см. настройку VLAN в подразделе **Сеть**).

Настройка VLAN

Настройка VLAN является необязательной и выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Сеть**).

Для создания и настройки VLAN используются команды:

Описание параметров также приведено в подразделе **Объект** `vlan`.

- создание VLAN и назначение ему идентификатора:

```
set vlan name <имя> vlan-id <идентификатор>
```

- назначение VLAN сетевому интерфейсу:

Сетевые интерфейсы определяются автоматически при загрузке Termidesk Connect.

```
set vlan name <имя> if-ethernet <имя_интерфейса>
```

- (опционально) назначение VRF для VLAN:

```
set vlan name <имя> vrf <имя_VRF>
```

- (опционально, если используется отказоустойчивая конфигурация) задание типа синхронизации:

Тип может быть:

- `LOCAL` – VLAN используется в локальной конфигурации, не синхронизируется для отказоустойчивой конфигурации;
- `SHARED` – VLAN синхронизируется для отказоустойчивой конфигурации.

```
set vlan name <имя> ha-type <значение>
```

- (опционально) назначение VLAN агрегированному интерфейсу (см. подраздел **Настройка агрегации каналов**):

```
set vlan name <имя> if-aggregation <имя_агрегированного_интерфейса>
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml vlan name <имя>
```

- просмотр выполненных команд:

```
show configuration cli vlan name <имя>
```

Настройка IP-адресов

Настройка IP-адресов является обязательной и выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Сеть**).

Для добавления и настройки IP-адресов используются команды:

Описание параметров также приведено в подразделе **Объект `ip`**.

- добавление IP-адреса в формате CIDR:

```
set ip address <IP-адрес> <длина_префикса>
```

Пример:

```
set ip address 172.16.0.1 /16
```

- назначение сетевого интерфейса для добавленного IP-адреса:

Сетевые интерфейсы определяются автоматически при загрузке Termidesk Connect.

```
set ip address <IP-адрес> <длина_префикса> if-ethernet <имя_интерфейса>
```

Пример:

```
set ip address 172.16.0.1 /16 if-ethernet eth0
```

- или назначение VLAN для добавленного IP-адреса:

VLAN должен быть предварительно создан.

```
set ip address <IP-адрес> <длина_префикса> if-vlan <имя_VLAN>
```

- (опционально, если используется отказоустойчивая конфигурация) задание типа синхронизации:

Тип может быть:

- `LOCAL` – IP-адрес используется в локальной конфигурации, не синхронизируется для отказоустойчивой конфигурации;
- `SHARED` – IP-адрес для отказоустойчивой конфигурации.

```
set ip address 172.16.0.1 /16 ha-type <значение>
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml ip
```

- просмотр выполненных команд:

```
show configuration cli ip
```

Настройка маршрутизации

Настройка маршрутизации является обязательной и выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Сеть**).

Для добавления маршрута используются команды:

Описание параметров также приведено в подразделе **Объект** `ip`.

- добавление маршрута с указанием IP-адреса в формате CIDR:

```
set ip route <имя_VRF> <IP-адрес><длина_префикса> <IP-адрес_шлюза>
```

Пример:

```
set ip route namevrf 0.0.0.0/8 172.16.0.1
```

- (опционально, если используется отказоустойчивая конфигурация) задание типа синхронизации:

Тип может быть:

- `LOCAL` – маршрут используется в локальной конфигурации, не синхронизируется для отказоустойчивой конфигурации;
- `SHARED` – маршрут синхронизируется для отказоустойчивой конфигурации.

```
set ip route <имя_VRF> <IP-адрес><длина_префикса> <IP-адрес_шлюза> ha-type
<значение>
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml ip
```

- просмотр выполненных команд:

```
show configuration cli ip
```

Настройка IP-Фондов

IP-Фонд – это группа IP-адресов, которые может использовать Termidesk Connect для взаимодействия с Реальными Серверами. Настройка IP-Фонда необходима для:

- определения перечня IP-адресов, с которых Termidesk Connect будет подключаться к Реальным Серверам;
- снятия существующих сетевых ограничений по количеству открытых TCP-сессий со стороны Termidesk Connect к Реальным Серверам.

Добавление IP-Фондов в общем случае помогает решить ситуацию, когда количество TCP-сессий, открываемых между Termidesk Connect (работающим в режиме Full Proxy) и Реальными Серверами, ограничивается максимально возможным их количеством (65 536) для одного IP-адреса.

Настройка IP-Фондов является необязательной и выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Сеть**).

Для создания и настройки IP-Фондов используются команды:

Описание параметров также приведено в подразделе **Объект** `ipset`.

- создание IP-Фонда:

```
set ipset id <имя>
```

- добавление IP-адреса в IP-Фонд:

IP-адрес может быть любым: он необязательно должен быть из списка тех, что уже существуют для физических сетевых интерфейсов.

Для одного IP-Фонда может быть задано несколько IP-адресов. Команда позволяет за один раз добавить только один IP-адрес.

```
set ipset id <имя> ips <IP-адрес>
```

- добавление VRF в IP-Фонд (по умолчанию – `default`):

```
set ipset id <имя> vrf <IP-адрес>
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

- просмотр заданных настроек IP-Фонда (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml ipset id <имя>
```

- просмотр заданных настроек IP-адреса в IP-Фонде (указывается формат вывода – XML, JSON или TXT):

```
show configuration xml ipset id <имя> ips <IP-адрес>
```

- просмотр выполненных команд:

```
show configuration cli ipset id <имя>
```

Настройка динамической маршрутизации

Общие сведения по настройке динамической маршрутизации

Termidesk Connect поддерживает управление динамической маршрутизацией с использованием ПО Free Range Routing (FRR) – набора служб маршрутизации с поддержкой протоколов BGP, OSPF и других.

Настройки динамической маршрутизации выполняются из интерфейса VTYSH (см. подраздел **Общие сведения по работе с VTYSH**), для этого:

- перейти в интерфейс VTYSH, последовательно выполнив:

```
bash
sudo vtysh
```

- перейти в режим настройки:

```
configure terminal
```

- выполнить требуемые настройки динамической маршрутизации;

Ручное изменение маршрутов недопустимо: состояние службы frr не отслеживается автоматические, поэтому при ручном изменении какого-либо из маршрутов поведение станет неопределенным.

Для работы анонсирования маршрута (RHI) для определенного VRF нужно:

- выполнить настройку маршрутизатора BGP:

```
router bgp <номер_системы> vrf <имя_VRF>
```

где:

<номер_системы> – числовой идентификатор автономной системы (AS

Number, ASN) – группы сетей, использующих BGP и находящихся под единым административным управлением. Часто ASN – это идентификатор сети провайдера;

<имя_VRF> – имя VRF;

- при настройке eBGB нужно:
  - либо настроить политики протокола;
  - либо выполнить (до ввода команды должна быть выполнена настройка маршрутизатора BGP):

```
no bgp ebgp-requires-policy
```

При работе RHI в конфигурацию добавятся строки следующего вида, которые не следует менять вручную:

```
network <виртуальный_IP-адрес>/32
```

- после настройки последовательно выйти из режима конфигурирования config-router:

```
exit
exit
```

- убедиться, что отображена строка приглашения termidesk-connect-dr#;
- записать изменения:

```
write
```

- убедиться в правильности настройки:

```
show running-config
```

- из termidesk-connect-dr# перейти в стандартный интерфейс командной строки Linux Shell:

```
exit
```

- перезапустить службу frr:

```
sudo systemctl restart frr
```

Пример настройки минимальной конфигурации для динамической маршрутизации

Пример настройки:

- перейти в интерфейс VTYSH, последовательно выполнив:

```
bash
sudo vtysh
```

- перейти в режим настройки:

```
configure terminal
```

- выполнить настройку маршрутизатора BGP:

```
router bgp <номер_системы>
```

где:

`<номер_системы>` – числовой идентификатор автономной системы (ASN) – группы сетей, использующих BGP и находящихся под единым административным управлением. Часто ASN – это идентификатор сети провайдера;

- отключить политики протокола для eBGP:

```
no bgp ebgp-requires-policy
```

- добавить информацию о соседнем маршрутизаторе BGP:

```
neighbor <IP-адрес_соседа> remote-as <номер_системы_соседа>
```

- после настройки последовательно выйти из режима конфигурирования `config-router`:

```
exit
exit
```

- убедиться, что отображена строка приглашения `termidesk-connect-dr#`;
- записать изменения:

```
write
```

- убедиться в правильности настройки:

```
show running-config
```

```
Building configuration...

Current configuration:
!
frr version 10.3.1
frr defaults traditional
hostname lb-name
log syslog informational

service integrated-vtysh-config
!
router bgp 10
 no bgp ebgp-requires-policy
 neighbor 192.168.0.2 remote-as 22
exit
!
end
```

- из `termidesk-connect-dr#` перейти в стандартный интерфейс командной строки Linux Shell:

```
exit
```

- перезапустить службу `frr`:

```
sudo systemctl restart frr
```

Создание статической записи ARP

Создание статической записи ARP является необязательной и выполняется из интерфейса командной строки Termidesk Connect.

Для создания записи ARP используются команды:

Описание параметров также приведено в подразделе **Объект** `arp`.

- создание статической записи ARP:

```
set arp static <IP-адрес> mac <MAC-адрес>
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml arp static <IP-адрес> mac <MAC-адрес>
```

- просмотр выполненных команд:

```
show configuration cli arp static <IP-адрес> mac <MAC-адрес>
```

Управление системой

Общие параметры управления

После установки Termidesk Connect необходимо выполнить начальную конфигурацию веб-сервера Apache для настройки доступа к веб-интерфейсу.

Настройка доступа к веб-интерфейсу выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Управление**).

Описание параметров также приведено в подразделе **Объект** `system`.

Для настройки доступа используются команды:

- задание IP-адреса веб-интерфейса управления из списка существующих на узле:

```
set system mgmt ip <IP-адрес>
```

- задание порта для подключения к веб-интерфейсу (по умолчанию – `443`):

```
set system mgmt webui-port <порт>
```

- задание сертификата для подключения к веб-интерфейсу по протоколу HTTPS (по умолчанию – `tdc_ss_public_key.pem`):

Загрузка сертификатов и ключей выполняется с помощью протокола SFTP или через веб-интерфейс (см. подраздел **Веб. TLS**).

После загрузки файлы будут расположены в каталоге `/etc/ssl/tdc/`.

Так же файл может быть получен из Хранилища секретов. Пример значения параметра: `pki://pki/issue/example-dot-com?common_name=\"app.example.com\"&ttl=\"24h\"#certificate`.

```
set system mgmt webui-cert <файл>
```

- задание закрытого ключа к сертификату для подключения к веб-интерфейсу по протоколу HTTPS (по умолчанию – `tdc_ss_private_key.key`):

Так же файл может быть получен из Хранилища секретов. Пример значения параметра: `pki://pki/issue/example-dot-com?common_name=\"app.example.com\"&ttl=\"24h\"#private_key`.

```
set system mgmt webui-key <файл>
```

- задание времени жизни (в секундах) сессии пользователя (по умолчанию – `3600`):

```
set system mgmt webui-timeout <значение>
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

Ролевая модель доступа

По умолчанию после установки Termidesk Connect используется следующая ролевая модель:

- роль «Администратор»: роль, которой доступны настройка и управление Termidesk Connect после успешного прохождения процедуры идентификации и аутентификации. По умолчанию после установки с ролью «Администратор» ассоциируется локальный пользователь ОС `tdadmin`, состоящий в группе `tdadmin`;

Запрещается удалять пользователя `tdadmin` и группу `tdadmin`, поскольку это приведет к неработоспособности системы.

- роль «Оператор»: роль, которой доступен только просмотр настроек Termidesk Connect после успешного прохождения процедуры идентификации и аутентификации. По умолчанию после установки с ролью «Оператор» ассоциируется локальный пользователь ОС `tdoperator`, состоящий в группе `tdoperator`.

По умолчанию все операции (авторизация, просмотр и изменение настроек) для неидентифицированных пользователей запрещены.

Администратор Termidesk Connect может гибко настроить:

- список групп и пользователей (см. подраздел **Создание групп и пользователей**);
- правила доступа (см. подраздел **Создание набора правил доступа**).

Конфигурация по умолчанию после установки:

- просмотр конфигурации:

```
show configuration cli nacm
```

- пример вывода:

```
set nacm enable-nacm true
set nacm read-default deny
set nacm write-default deny
set nacm exec-default deny
set nacm rule-list admin
set nacm rule-list admin group tdadmin
set nacm rule-list admin rule allow-all
set nacm rule-list admin rule allow-all path /
set nacm rule-list admin rule allow-all access-operations *
set nacm rule-list admin rule allow-all action permit
set nacm rule-list operator
set nacm rule-list operator group tdoperator
set nacm rule-list operator rule allow-read
set nacm rule-list operator rule allow-read path /
set nacm rule-list operator rule allow-read access-operations read
set nacm rule-list operator rule allow-read action permit
```

Создание групп и пользователей

Создание и настройка пользователей и группы выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Управление**).

По умолчанию после установки доступны пользователи и группы `tdadmin` и `tdoperator`. Запрещается удалять пользователя и группу `tdadmin`, поскольку это приведет к неработоспособности системы.

Для создания и настройки пользователей и групп используются команды:

Описание параметров также приведено в подразделах **Объект `user`** и **Объект `groups`**.

- создание группы:

Имя группы должно соответствовать заданной в службе каталогов, если настроено подключение к ней (см. подраздел **Аутентификация и авторизация пользователей через службу каталогов**).

```
set groups name <имя_группы>
```

- создание пользователя и добавление его в группу:

```
set user name <имя_пользователя> groups <имя_группы>
```

- применение конфигурации (после создания пользователя команду выполнять обязательно):

```
commit
```

- назначение пароля пользователю:

```
set user name <имя_пользователя> password
```

Пароль вводится в интерактивном режиме и отправляется после нажатия клавиши **<ENTER>**. Длина пароля должна составлять от 8 до 72 символов.

В пароле нельзя использовать символы: ", ', \, `, $, !, <, >, ;, {, }, (,), [,], |, *, ?, ~, &, ^, а также управляющие символы – табуляции,переноса строки и возврата каретки.

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml user name <имя>
```

- просмотр выполненных команд:

```
show configuration cli user name <имя>
```

Создание набора правил доступа

Правила доступа используются для разграничения доступа пользователей к настройке и управлению Termidesk Connect (см. подраздел **Ролевая модель доступа**).

Для создания и настройки списка правил доступа используются команды:

Описание параметров также приведено в подразделе **Объект** `nacm`.

- создание набора правил доступа:

По умолчанию есть следующие преднастроенные наборы:

- `admin` – разрешены все действия по настройке Termidesk Connect;
- `operator` – разрешен только просмотр, действия по настройке Termidesk Connect запрещены.

```
set nacm rule-list <имя_набора>
```

- назначение набора правил доступа группе пользователей (группа пользователей должна быть предварительно создана, см. подраздел **Создание групп и пользователей**):

Может быть назначено несколько групп пользователей.

```
set nacm rule-list <имя_набора> group <имя_группы>
```

- создание непосредственно правила доступа для набора:

Может быть создано несколько правил доступа.

```
set nacm rule-list <имя_набора> rule <имя_правила>
```

- настройка добавленного правила доступа для набора:
  - указание пути к элементам конфигурации Termidesk Connect (по умолчанию – `/`):

Предполагает выражения для указания конкретных путей обхода XML-дерева, расположенного в хранилище конфигурации. Осуществляется на языке запросов XPath, определенного в RFC 5261. По заданному выражению определяются права доступа к конкретным элементам конфигурации.

Значение `/` указывает на все возможное содержимое хранилища данных.

```
set nacm rule-list <имя_набора> rule <имя_правила> path <путь>
```

  - определение операции доступа, которая будет связана с правилом (по умолчанию – `*`):

Операцией может быть:

- `edit` – изменение конфигурации Termidesk Connect;
- `read` – чтение конфигурации Termidesk Connect;
- `exec` – выполнение определенных операций. Например: `commit`, `write`, `restore-broken-config` и т.д.;
- `*` – все операции и команды.

```
set nacm rule-list <имя_набора> rule <имя_правила> access-operations
<операция>
```

- определение действия по доступу, которое будет связано с правилом:

Действием может быть:

- `deny` – запретить операции;
- `permit` – разрешить операции.

```
set nacm rule-list <имя_набора> rule <имя_правила> action <действие>
```

- указание текстового описания правила доступа:

```
set nacm rule-list <имя_набора> rule <имя_правила> comment <комментарий>
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml nacm rule-list <имя_набора>
```

- просмотр выполненных команд:

```
show configuration cli nacm rule-list <имя_набора>
```

Пример создания набора правил, разрешающего все действия с объектами для созданной группы `users`:

```
set groups name users
set rule-list allow-change group users
set rule-list allow-change rule allow-change
set rule-list allow-change rule allow-change action permit
set rule-list allow-change rule allow-change access-operations *
```

Аутентификация и авторизация пользователей через службу каталогов

В качестве средства аутентификации и авторизации пользователей (администраторов Termidesk Connect) может использоваться служба каталогов, существующая и настроенная в инфраструктуре организации.

Особенности работы при подключении к службе каталогов:

- для аутентификации и авторизации пользователей используется PAM-модуль;
- права пользователей к Termidesk Connect назначаются в зависимости от правил доступа группы, в которую входит пользователь (см. подраздел **Создание набора правил доступа**);
- подключенная служба каталогов не исключает функции локальных администраторов, существующих в Termidesk Connect.

Настройка подключения к службе каталогов выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Управление**).

Для настройки подключения к службе каталогов используются команды:

Описание параметров также приведено в подразделе **Объект** `ldap`.

- указание типа подключаемой службы каталогов:

Возможные значения:

- `AD` – служба каталогов Active Directory Domain Services;
- `FreeIPA` – служба каталогов FreeIPA;
- `OpenLDAP` – служба каталогов OpenLDAP Directory Services.

```
set ldap type <тип_службы_каталогов>
```

- указание доменного имени сервера службы каталогов или его IP-адреса:

Пример:

```
set ldap domain example.loc
```

```
set ldap domain <доменное_имя_сервера_или_IP>
```

- указание времени ожидания (в секундах) ответа от службы каталогов (по умолчанию – `30`):

```
set ldap timeout <значение>
```

- указание типа безопасности для подключения к службе каталогов (по умолчанию –

TEXT):

Возможные значения:

- TEXT – незащищенное подключение;
- SSL – защищенное подключение. Перед обменом данными будет установлена TLS-сессия.

```
set ldap security <тип_безопасности>
```

- (опционально, если используется защищенное подключение к службе каталогов) указание файла сертификата УЦ:

Файл сертификата УЦ должен быть предварительно загружен на Termidesk Connect (см. подраздел **TLS**).

Так же сертификат УЦ может быть получен из Хранилища секретов. Пример значения параметра: pki://pki/issue/example-dot-com?common_name=\"app.example.com\"&ttl=\"24h\"#issuing_ca.

```
set ldap ca-cert <имя_файла>
```

- указание порта для подключения к службе каталогов (по умолчанию – 389):

```
set ldap port <значение>
```

- указание корня поиска в службе каталогов (Base DN):

Вводимое значение не должно содержать пробелов:

- в начале и конце строки;
- рядом с разделителями (запятыми);
- в элементах пути (например, «DC=company name,DC=de»).

Пример:

```
set ldap base-dn DC=example,DC=loc
```

```
set ldap base-dn <значение>
```

- указание учетной записи (с правами на чтение) в формате DN, используемой для подключения к службе каталогов:

Вводимое значение не должно содержать пробелов:

- в начале и конце строки;

- рядом с разделителями (запятыми);
- в элементах пути (например, «DC=company name,DC=de»).

Пример:

```
set ldap administrator-bind-dn CN=admin,OU=Users,DC=example,DC=loc
```

Так же учетная запись может быть получена из Хранилища секретов. Пример значения параметра:

- `kv://secret/data/my_ldap#dn` – для `kv` версии 2;
- `ad://ldap/static-cred/my_ldap#dn` – для `ldap`.

```
set ldap administrator-bind-dn <значение>
```

- указание пароля учетной записи:

Так же пароль может быть получен из Хранилища секретов. Пример значения параметра:

- `kv://secret/data/my_ldap#last_password` – для `kv` версии 2;
- `ad://ldap/static-cred/my_ldap#last_password` – для `ldap`.

```
set ldap password <пароль>
```

- указание атрибута уникального имени или идентификатора пользователя в службе каталогов (по умолчанию – `userPrincipalName`):

Возможные значения:

- `uid` – уникальный идентификатор учетной записи;
- `userPrincipalName` – логин пользователя в формате `user@example.loc`;
- `SamAccountName` – короткое имя пользователя;
- `cn` – отображаемое имя пользователя (иногда – имя для входа).

```
set ldap user-name-attribute <атрибут_имени_пользователя>
```

- указание атрибута группы (по умолчанию – `memberOf`):

Возможные значения:

- `memberOf` – список участников группы, в котором каждый участник указывается в виде полного DN. Универсальный атрибут;
- `uniqueMember` – аналог атрибута выше, но используется в некоторых реализациях LDAP, например, OpenLDAP Directory Services.

```
set ldap group-attribute <атрибут_группы>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml ldap
```

- просмотр выполненных команд:

```
show configuration cli ldap
```

После настройки подключения должно быть выполнено добавление группы пользователей службы каталогов (см. подраздел **Создание групп и пользователей**).

Для разделения полномочий в Termidesk Connect также должна быть выполнена настройка правил доступа (см. подраздел **Создание набора правил доступа**).

Диагностика

Диагностика позволяет выявить сетевые неисправности при возникновении проблем с подключением через Termidesk Connect.

Диагностика сети в Termidesk Connect выполняется через стандартный интерфейс командной строки Linux Shell. Для перехода в интерфейс следует вызвать из CLI команду:

```
bash
```

Для диагностики сети могут использоваться команды Linux Shell:

- для проверки сетевого соединения к узлу:

```
sudo ping <IP-адрес>
```

- для анализа и фильтрации трафика:

```
sudo tcpdump -i <интерфейс>
```

- для проверки маршрута до узла:

```
sudo tcptraceroute <IP-адрес> <порт>
```

- для анализа и фильтрации трафика SSL/TLS:

```
sudo ssldump
```

Аудит

Общие сведения о системе аудита

Система аудита фиксирует действия пользователей, системных служб и изменения конфигурации Termidesk Connect. Полученные сведения могут использоваться для анализа событий, устранения неисправностей и внутреннего контроля.

Информация о событиях записывается в журнал и может отправляться на внешний syslog-сервер. Состав и объём фиксируемых событий определяется настройками уровня и категории сообщений.

Создание и настройка syslog-сервера

Для отправки событий на syslog-сервер он должен быть задан в конфигурации Termidesk Connect. Добавление syslog-сервера в конфигурацию выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Аудит**).

Для добавления syslog-сервера используются команды:

Описание параметров также приведено в подразделе **Объект** `audit`.

- добавление syslog-сервера:

```
set audit syslog <имя>
```

- задание IP-адреса syslog-сервера:

```
set audit syslog ip <IP-адрес>
```

- задание порта syslog-сервера:

```
set audit syslog port <порт>
```

- задание протокола для передачи событий на syslog-сервер:

```
set audit syslog transport <протокол>
```

При использовании значения `tls` нужно задать значения параметрам `ca-cert`, `ca-key` и `peer-verify`.

- указание пути к файлу корневого сертификата для защищенного соединения SSL/TLS:

Так же файл может быть получен из Хранилища секретов. Пример значения параметра: `pki://pki/issue/example-dot-com?common_name=\"app.example.com\"&ttl=\"24h\"#certificate`.

```
set audit syslog ca-cert <путь>
```

- указание пути к файлу ключа для защищенного соединения SSL/TLS:

Так же файл может быть получен из Хранилища секретов. Пример значения параметра: `pki://pki/issue/example-dot-com?common_name=\"app.example.com\"&ttl=\"24h\"#private_key`.

```
set audit syslog ca-key <путь>
```

- настройка проверки подлинности сертификата при запросах к syslog-серверу:

```
set audit syslog peer-verify <значение>
```

- указание категории событий для записи в журнал:

```
set audit syslog facility <категория>
```

- указание уровня событий для записи в журнал:

```
set audit syslog level <уровень>
```

- указание службы, события которой будут записаны в журнал:

```
set audit syslog service <служба>
```

- задание IP-адреса, используемого для отправки событий на syslog-сервер:

```
set audit syslog local-address <IP-адрес>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – XML, JSON или TXT):

```
show configuration xml audit syslog <имя>
```

- просмотр выполненных команд:

```
show configuration cli audit syslog <имя>
```

SNMP

Настройки SNMP

Настройка SNMP выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. SNMP**).

Для настройки SNMP выполнить:

Описание параметров также приведено в подразделе **Объект** `snmp`.

- добавление адреса агента, на котором SNMP-сервис будет ожидать запросы:

Указывается настроенный IP-адрес с типом Shared.

Для <порт> значение по умолчанию – 161.

Для <протокол> значение по умолчанию – UDP.

```
set snmp listener <ip-адрес> <порт> <протокол>
```

- добавление сообщества для доступа к данным SNMP-сервиса:

Для <исходный_узел> значение по умолчанию – 0.0.0.0/0.

```
set snmp community <имя> <исходный_узел>
```

- указание уровня доступа сообщества:

Для <доступ> значение по умолчанию – RO.

```
set snmp community <имя> <исходный_узел> access <доступ>
```

- (опционально) описание SNMP-сервиса (по умолчанию – Termidesk Connect):

Для описания допустимы только латинские буквы. В случае использования пробелов следует заключать текст описания в двойные кавычки.

```
set snmp description <описание>
```

- (опционально) описание физического расположения устройства (по умолчанию – DC):

Для описания допустимы только латинские буквы. В случае использования пробелов следует заключать текст описания в двойные кавычки.

```
set snmp location <расположение>
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

- просмотр заданных настроек (указывается формат вывода – XML, JSON или TXT):

```
show configuration xml snmp
```

- просмотр выполненных команд:

```
show configuration cli snmp
```

Пример настройки SNMP:

```
set snmp listener 192.0.2.1 161 TCP
set snmp listener 192.0.2.1 161 UDP
set snmp community SNMPcom 0.0.0.0/0
set snmp community SNMPcom 0.0.0.0/0 access RW
set snmp description "TDC 1"
```

```
set snmp location "Data Center"
```

Пользователи SNMP

Добавление пользователя для SNMPv3 выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. SNMP**).

Для добавления пользователя SNMPv3 выполнить:

Описание параметров также приведено в подразделе **Объект** `snmpv3`.

- указание имени пользователя (доступно значение от 1 до 64 символов):

Так же имя пользователя может быть получено из Хранилища секретов.

Пример значения параметра:

- `kv://secret/snmp#username` – для `kv` версии 1;
- `kv://secret/data/snmp#username` – для `kv` версии 2.

```
set snmpv3 user <имя>
```

- указание протокола аутентификации (по умолчанию – `NONE`):

```
set snmpv3 user <имя> auth-protocol <протокол_аутентификации>
```

- указание пароля аутентификации:

Пароль аутентификации указывается в случае, если значение протокола аутентификации отлично от `NONE`.

Так же пароль может быть получен из Хранилища секретов.

Пример значения параметра:

- `kv://secret/snmp#auth_key` – для `kv` версии 1;
- `kv://secret/data/snmp#auth_key` – для `kv` версии 2.

```
set snmpv3 user <имя> auth-key <пароль_аутентификации>
```

- указание протокола конфиденциальности (по умолчанию – `NONE`):

```
set snmpv3 user <имя> priv-protocol <протокол_конфиденциальности>
```

- указание пароля конфиденциальности:

Пароль конфиденциальности указывается в случае, если значение протокола конфиденциальности отлично от `NONE`.

Так же пароль может быть получен из Хранилища секретов.

Пример значения параметра:

- `kv://secret/snmp#priv_key` – для `kv` версии 1;
- `kv://secret/data/snmp#priv_key` – для `kv` версии 2.

```
set snmpv3 user <имя> priv-key <пароль_конфиденциальности>
```

- указание уровня доступа пользователя (по умолчанию – `RO`):

```
set snmpv3 user <имя> permissions <доступ>
```

- указание типа идентификатора агента SNMP для сценариев, где требуется его уникальность:

```
set snmpv3 engine-id-type <идентификатор>
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml snmpv3
```

- просмотр выполненных команд:

```
show configuration cli snmpv3
```

Пример настройки SNMPv3:

```
set snmpv3 user user1
set snmpv3 user user1 auth-protocol MD5
```

```
set snmpv3 user user1 auth-key passExample
set snmpv3 user user1 priv-protocol DES
set snmpv3 user user1 priv-key passExample
set snmpv3 user user1 permissions RW
set snmpv3 user user2
set snmpv3 user user2 auth-protocol MD5
set snmpv3 user user2 auth-key passv3Example
set snmpv3 user user2 priv-protocol NONE
set snmpv3 user user2 permissions RO
set snmpv3 engine-id-type 1
```

Резервное копирование и обновление

Общие сведения по управлению образами и резервными копиями

Для управления образами и резервными копиями в Termidesk Connect используются инструменты:

- `vamgr`;
- `snag`.

Работа с инструментами `vamgr` и `snag` доступна в Termidesk Connect с версии 1.1 и полностью исключает работу с устаревшим инструментом `va` из Termidesk Connect версии 1.0.

При переходе необходимо:

- ввести в эксплуатацию версию 1.1;
- удалить устройства версии 1.0 из эксплуатации.

Инструмент `vamgr` позволяет:

- устанавливать, обновлять и удалять образы;
- предоставлять интерактивный доступ к образам;
- управлять приоритетами загрузки образов.

Инструмент `snag` позволяет:

- создавать, управлять и восстанавливать резервные копии данных;
- импортировать, экспортировать, создавать и восстанавливать резервные копии;
- управлять правилами отслеживания файлов.

При установке или обновлении образов инструмент `vamgr` позволяет импортировать состояние конфигураций с помощью инструмента `snag` из существующего хранилища резервных копий.

Порядок взаимодействия инструментов `vamgr` и `snag` представлен на рисунке (см. Взаимодействие инструментов `vamgr` и `snag`).

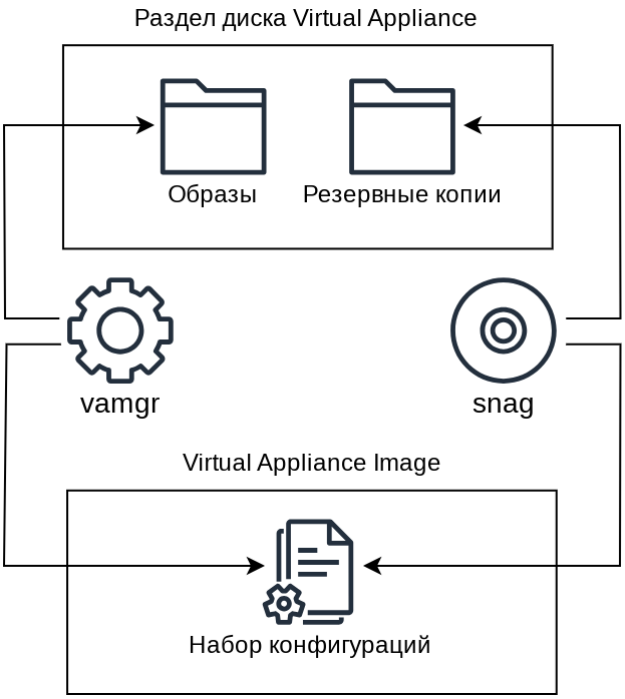

Рисунок 52. Взаимодействие инструментов `vamgr` и `snag`

Управление резервными копиями через инструмент `snag`

Для управления резервными копиями в Termidesk Connect используется инструмент `snag`.

Инструмент `snag` доступен через интерфейс командной строки Linux Shell.

Для перехода в интерфейс следует вызвать из CLI команду:

```
bash
```

Для управления резервными копиями требуются привилегированные права пользователя.

Основной формат команды инструмента `snag`:

```
snag <команда>
```

Для получения информации по основным командам инструмента `snag` выполнить:

```
snag -h
```

Для указания пути к файлу конфигурации, отличного от стандартного, используется опция `-c`. Пример:

```
snag -c <путь_к_файлу_конфигурации> <команда>
```

Доступные команды инструмента `snag` приведены в таблице (см. Основные команды инструмента `snag`).

Таблица 2. Основные команды инструмента `snag`

Команда	Описание
`init`	Инициализация репозитория для хранения резервных копий
`create`	Создание новой резервной копии
`import`	Импорт резервной копии из архива формата `tar.gz`
`export`	Экспорт резервной копии в архив формата `tar.gz`
`restore`	Восстановление состояния из указанной резервной копии
`list`	Вывод списка резервных копий
`diff`	Отображение измененных данных
`status`	Проверка статуса отслеживаемых файлов
`size`	Отображение размера резервных копий
`rules`	Управление правилами отслеживания

Примеры команд инструмента `snag` приведены в таблице (см. Примеры использования команд инструмента `snag`).

Таблица 3. Примеры использования команд инструмента `snag`

Пример команды	Описание
`snag init`	Инициализация репозитория для хранения резервных копий
`snag init -f`	Инициализация с перезаписью существующего репозитория
`snag create`	Создание новой резервной копии
`snag create --no-pre`	Создание новой резервной копии без приостановки процессов из файла конфигурации
`snag create --no-post`	Создание новой резервной копии без возобновления процессов из файла конфигурации

`snag create -c <комментарий> -a <автор> -e <адрес_почты>`	Создание новой резервной копии с указанием комментария, автора и адреса электронной почты. Пример:
	`snag create -c <комментарий> -a <автор> -e <адрес_почты>`
	`Снимок успешно создан: 45b768a`

`snag import <путь_к_архиву>`	Импорт резервной копии из архива формата `tar.gz`. Пример:
	`snag import <путь_к_архиву>`
	`Импорт успешно завершен: 6yb73da`

`snag import --no-pre <путь_к_архиву>`	Импорт резервной копии из архива формата `tar.gz` без приостановки процессов из файла конфигурации
`snag import --no-post <путь_к_архиву>`	Импорт резервной копии из архива формата `tar.gz` без возобновления процессов из файла конфигурации

Пример команды	Описание
`snag import -c <комментарий> -a <автор> -e <адрес_почты> <путь_к_архиву>`	Импорт резервной копии из архива формата `tar.gz` с указанием комментария, автора и адреса электронной почты
`snag export <путь_к_папке>`	Экспорт резервной копии в архив формата `tar.gz`

`snag export -s <хеш> <путь_к_папке>`	Экспорт в архив формата `tar.gz` с указанием хеша резервной копии. Пример:
	`snag export -s <хеш> <путь_к_папке>`
	`Экспорт в архив успешно завершен:`
	`/va/backups/37542290654-45b768a.tar.gz`

`snag restore <хеш>`	Восстановление состояния из указанной резервной копии. Пример:
	`snag restore <хеш>`
	`Резервная копия успешно восстановлена: 45b768a`

`snag restore --no-pre <хеш>`	Восстановление состояния из указанной резервной копии без приостановки процессов из файла конфигурации
`snag restore --no-post <хеш>`	Восстановление состояния из указанной резервной копии без возобновления процессов из файла конфигурации
`snag list`	Вывод списка резервных копий

`snag list -c -a -e`	Вывод списка резервных копий с указанием комментария, автора и электронной почты. Пример:
	`snag list -c -a -e`
	`> 45b768a 2025.08.13 14:21:02 Создание стандартного снимка snag user@site.domain`

`snag diff`	Отображение измененных данных
`snag diff -s <хеш>`	Отображение измененных данных с указанием хеша резервной копии
`snag status`	Проверка статуса отслеживаемых файлов
`snag size`	Отображение размера резервных копий
`snag rules save`	Сохранение правил отслеживания
`snag rules show`	Отображение текущих правил отслеживания
`snag rules show -c`	Отображение правил отслеживания из файла конфигурации
`snag rules update`	Обновление правил отслеживания
`snag rules update -r`	Обновление правил отслеживания с удалением ранее отслеживаемых файлов Требуется повышенная осторожность при работе с опцией `-r`.
`snag rules reset`	Сброс правил отслеживания до состояния внесенных изменений

Пример команды	Описание
`snag rules clear`	Очистка правил отслеживания

Управление образами через инструмент `vamgr`

Для управления образами в Termidesk Connect используется инструмент `vamgr`.

Управление текущим образом невозможно, можно управлять только другими образами из текущего.

Инструмент `vamgr` доступен через интерфейс командной строки Linux Shell. Для перехода в интерфейс следует вызвать из CLI команду:

```
bash
```

Для управления образами требуются привилегированные права пользователя.

Основной формат команды инструмента `vamgr`:

```
vamgr <команда>
```

Для получения информации по основным командам инструмента `vamgr` выполнить:

```
vamgr -h
```

Для получения справочной информации возможно использовать `-h` или `--help` с любой командой.

Основные команды инструмента `vamgr` приведены в таблице (см. Основные команды инструмента `vamgr`).

Таблица 4. Основные команды инструмента `vamgr`

Команда	Описание
`chroot`	Предоставление интерактивного доступа к образу
`current`	Установка приоритета загрузки образа
`update`	Обновление существующего образа
`resize`	Увеличение размера образа
`install`	Установка существующего образа формата `img`
`upgrade`	Установка нового образа
`remove`	Удаление образа
`version`	Проверка текущей версии образа
`list`	Отображение списка установленных образов

Убедитесь, что перед использованием команд указаны верные пути и имена образов.

Примеры использования команд инструмента `vamgr` приведены в таблице (см. Примеры использования команд инструмента `vamgr`).

Таблица 5. Примеры использования команд инструмента `vamgr`

Пример	Описание
`vamgr chroot <имя_или_путь_к_образу>`	Предоставление интерактивного доступа к указанному образу
`vamgr chroot --host-path <директория> <имя_или_путь_к_образу>`	Монтирование указанной директории в образ и предоставление интерактивного доступа
`vamgr current`	Отображение приоритетного для загрузки образа
`vamgr current <имя_образа>`	Установка указанного образа как приоритетного для загрузки
`vamgr update -s <хеш_резервной_копии> <имя_или_путь_к_образу>`	Обновление существующего образа с указанным хешом резервной копии для распаковки в образ
`vamgr update -f -s <хеш_резервной_копии> <имя_или_путь_к_образу>`	Обновление существующего образа с указанным хешом резервной копии для распаковки в образ и принудительная повторная инициализация репозитория
`vamgr resize [-h] -s <размер_образа> <имя_или_путь_к_образу>`	Увеличение размера указанного образа  Размер образа указывается в гибибайтах с возможным значением от 5.1 ГиБ до 25 ГиБ. Необходимо указывать при каждом следующем вызове команды число больше предыдущего.
`vamgr install <путь_к_образу>`	Установка существующего образа формата `img`
`vamgr install --mod-uuid <путь_к_образу>`	Установка существующего образа формата `img` с изменением уникального идентификатора устройства образа
`vamgr upgrade <путь_к_архиву>`	Установка нового образа из архива формата `tar.gz`  Устанавливаемый образ получает приоритет загрузки.
`vamgr upgrade --not-remove <путь_к_архиву>`	Установка нового образа из архива формата `tar.gz` без удаления образа формата `img` при возникновении ошибок, если он был распакован
`vamgr upgrade --mod-uuid -s <хеш_резервной_копии> <путь_к_архиву>`	Установка нового образа из архива формата `tar.gz` с указанным хешом резервной копии, изменяя уникальный идентификатор устройства образа
`vamgr remove <имя_образа>`	Удаление указанного образа
`vamgr remove --save-backup <имя_образа>`	Удаление указанного образа с сохранением его резервной копии
`vamgr version`	Отображение текущей версии используемого образа
`vamgr list`	Отображение списка установленных образов
`vamgr list -p`	Отображение простого списка установленных образов без форматирования

Процедуры обновления образа Termidesk Connect

Общие сведения по обновлению

Обновление выполняется из интерфейса командной строки Linux Shell, для этого следует вызвать из CLI команду:

```
bash
```

Для выполнения процедур обновления требуются привилегированные права пользователя.

Обновление выполняется с использованием инструментов `vamgr` и `snag`.

Обновление образа с восстановлением резервной копии

Для обновления образа с восстановлением резервной копии выполнить:

- создать резервную копию:

```
snag create
```

- посмотреть список резервных копий:

```
snag list -c -a -e
   >   952b797 2025.08.13 14:22:06     Создание стандартного снимка    snag
snag@site.domain
       45b768a 2025.08.13 14:21:02     custom                          user
user@site.domain
```

- скопировать архив в папку с образами:

```
scp -P 222 <путь_к_архиву> <имя_пользователя>@<IP-
адрес_интерфейса_управления>:<путь_к_папке_с_образами>
```

- установить новый образ из архива с указанием хеша резервной копии:

```
vamgr upgrade -s <хеш_резервной_копии> <путь_к_архиву_с_образом>
```

Устанавливаемый образ получает приоритет загрузки.

- вывести список установленных образов:

```
vamgr list
[Virtual Appliance Termidesk Connect Images]
 ├──[1]  *  [5.0G 2025.08.13-14:40]  1.0.22.25210.1550-dev-287a1c1
 └──[2] [ ] [5.0G 2025.08.13-15:39]  1.0.22.25210.1657-dev-a789dba
```

где:

* – текущий образ;

[ ] – приоритетный для загрузки образ;

- выполнить перезагрузку:

```
reboot
```

Обновление конфигурации существующего образа из резервной копии

Для обновления конфигурации существующего образа из резервной копии выполнить:

- создать резервную копию:

```
snag create
```

- посмотреть список резервных копий:

```
snag list -c -a -e
   >   952b797 2025.08.13 14:22:06     Создание стандартного снимка    snag
snag@site.domain
       45b768a 2025.08.13 14:21:02     custom                          user
user@site.domain
```

- обновить существующий образ с указанием хеша резервной копии:

```
vamgr update -s <хеш_резервной_копии> <имя_образа>
```

Переключение текущего образа на предыдущий

Для переключения текущего образа на предыдущий выполнить:

- вывести список установленных образов:

```
vamgr list
[Virtual Appliance Termidesk Connect Images]
 ├──[1] [*] [5.0G 2025.08.13-14:40]  1.0.22.25210.1550-dev-287a1c1
 └──[2]     [5.0G 2025.08.13-15:39]  1.0.22.25210.1657-dev-a789dba
```

где:

[*] – текущий и приоритетный для загрузки образ;

- установить указанный образ как приоритетный для загрузки:

```
vamgr current <имя_образа>
```

- отобразить приоритетный для загрузки образ:

```
vamgr current
Текущий загрузочный образ: 1.0.22.25210.1657-dev-a789dba
```

Журналирование

Журналирование в Termidesk Connect ведется в файлах:

- `/var/log/syslog`;
- `/var/log/mail.info`;
- `/var/log/mail.warn`;
- `/var/log/mail.err`;
- `/var/log/mail.log`;
- `/var/log/daemon.log`;
- `/var/log/kern.log`;
- `/var/log/auth.log`;
- `/var/log/user.log`;
- `/var/log/lpr.log`;
- `/var/log/cron.log`;
- `/var/log/debug`;
- `/var/log/messages`;
- `/var/log/error`.

Конфигурация ротации журналов описана в `/etc/logrotate.d`:

- `alternatives`;
- `apache2`;
- `aptitude`;
- `btmp`;
- `dpkg`;
- `frr`;
- `rsyslog.disabled`;
- `syslog-ng-mod-astra`;
- `syslog-ng-override`;
- `syslog-ng.disabled`;
- `wtmp`.

Запуск сервиса ротации журналов происходит ежечасно.

Настройки по умолчанию для ротации журналов:

- в системе может хранится до 24 файлов ротации;
- ротация журнала происходит еженедельно при условии, что он не пуст;
- при превышении размера журнала в 100 Мбайт ротация происходит принудительно,

игнорируя еженедельную ротацию.

Для просмотра журналов возможно использовать утилиту `journalctl`. Пример просмотра журнала работы службы `frr`:

```
journalctl -u frr -f
```

Для оперативного мониторинга журналов возможно использовать утилиту `tail`. Пример отслеживания системного журнала `syslog` в режиме реального времени:

```
tail -f /var/log/syslog
```

Для быстрого просмотра без предварительной распаковки журналов из архива возможно использовать утилиту `zcat`. Пример поиска в архиве системного журнала `syslog` командой `grep`:

```
zcat syslog* | grep lbs
```

Для изменения уровня журналирования используется команда:

```
set logging level <уровень>
```

Поддерживаются следующие уровни журналирования:

- `INFO` (по умолчанию) – журналируется общая информация о работе. Включает в себя сообщения уровней: `critical`, `error`, `warning`, `info`;
- `DEBUG` – журналируется отладочная информация о работе. Включает в себя сообщения уровней: `critical`, `error`, `warning`, `info`, `debug`.

Уровни сообщений и их описание приведены в таблице (см. Уровни сообщений).

Таблица 6. Уровни сообщений

Уровень	Описание
`critical`	Критические ошибки, при возникновении которых приложение не может продолжать работу
`error`	Ошибки, возникновение которых негативно влияет на функционирование приложения, но работа не прекращается
`warning`	Предупреждения
`info`	Общая информация по функционированию
`debug`	Отладочная информация по функционированию

УПРАВЛЕНИЕ ТРАФИКОМ

Проверки

Общие сведения о Проверках

Проверка – это набор правил для Проверки балансируемых (Реальных) Серверов. При создании Проверка не выполняется.

Выполнение Проверки представлено на рисунке (см. Выполнение Проверки).

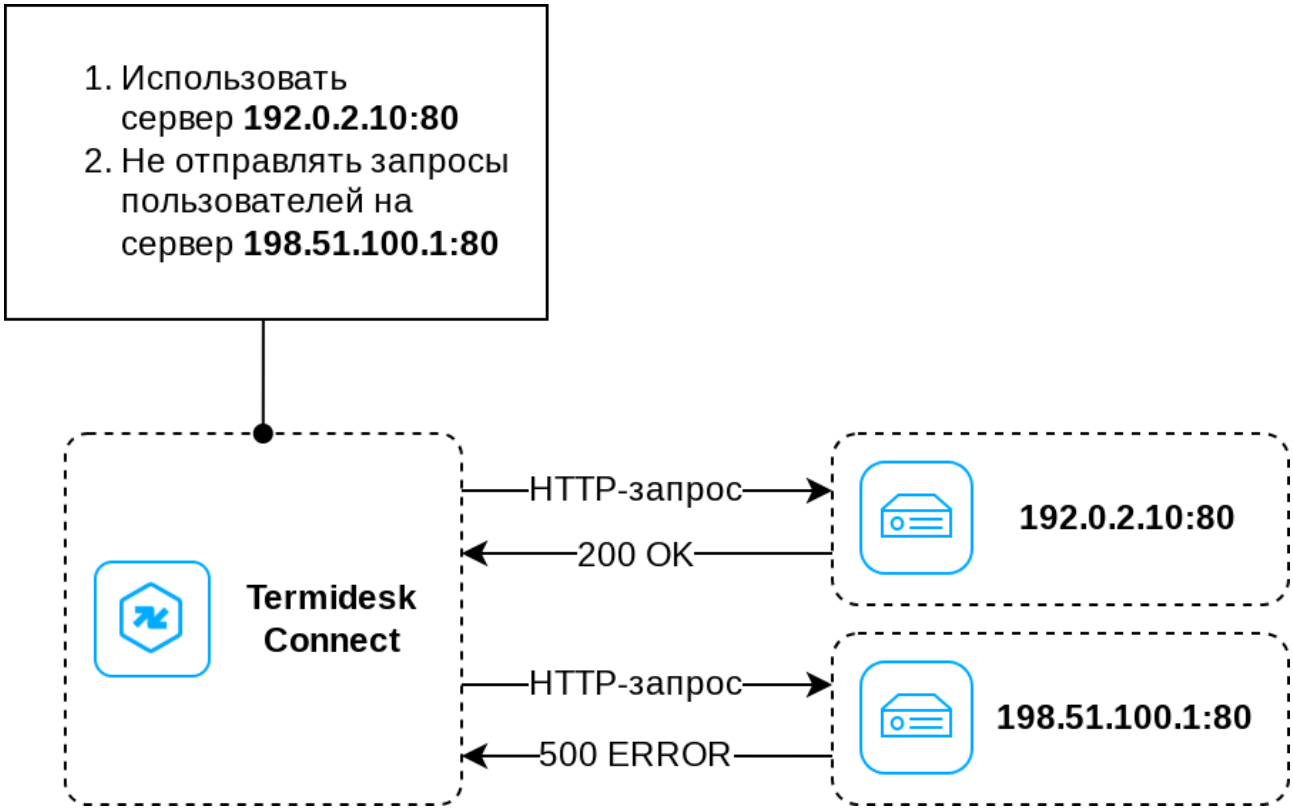

Рисунок 53. Выполнение Проверки

Проверки являются методами тестирования и диагностики сетевых соединений и работоспособности сервисов. Termidesk Connect позволяет создать:

- ICMP-Проверки, основанные на использовании протокола ICMP для определения состояния сетевого соединения. Такие Проверки выполняют отправку ICMP-запросов на узел, и на основании ответов определяют, доступен он или нет;
- TCP-Проверки, основанные на использовании протокола TCP. Такие Проверки позволяют определить, доступен ли сервис на указанном порту;
- HTTP-Проверки, основанные на использовании протокола HTTP. Такие Проверки выполняют отправку HTTP-запросов на узел, определяя доступность и работоспособность сервиса на основании ответов;
- HTTPS-Проверки, основанные на использовании защищенного протокола HTTPS. Такие Проверки позволяют преобразовать данные для защиты информации. Защищенный протокол HTTPS использует шифрование SSL/TLS, что обеспечивает целостность данных;
- Пользовательские Проверки, основанные на использовании пользовательских скриптов. Такие Проверки выполняются по логике, определенной пользователем, и позволяют реализовать нестандартные способы проверки доступности узлов или сервисов;
- Комбинированные Проверки, основанные на использовании нескольких Проверок с заданием их приоритетов (через логические выражения `AND/OR`). В такой Проверке могут быть заданы ранее созданные ICMP-Проверки, TCP-Проверки, HTTP-Проверки или Пользовательские Проверки.

Создание и настройка ICMP-Проверки

Создание и настройка ICMP-Проверок выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Проверки**).

Для создания и настройки ICMP-Проверки используются команды:

Описание параметров также приведено в подразделе **Объект** `health-check`.

- создание Проверки:

```
set health-check id <имя>
```

- назначение протокола для Проверки:

```
set health-check id <имя> ICMP
```

- задание интервала (в секундах) Проверки (по умолчанию – `10`):

```
set health-check id <имя> ICMP interval <значение>
```

- задание времени ожидания ответа на запрос (в секундах) (по умолчанию – `5`):

```
set health-check id <имя> ICMP timeout <значение>
```

- задание количества повторных Проверок в случае отсутствия ответа (по умолчанию – `1`):

```
set health-check id <имя> ICMP try <значение>
```

- задание необходимого количества успешных Проверок после восстановления связности (по умолчанию – `1`):

```
set health-check id <имя> ICMP success-try <значение>
```

- указание VRF для Проверки (по умолчанию – `default`):

```
set health-check id <имя> ICMP vrf <имя_VRF>
```

- указание IP-адреса источника (Termidesk Connect), с которого будет выполнена Проверка:

Указание отдельных IP-адресов для Проверок используется для логического разделения трафика:

- для перенаправления пользовательских подключений к Реальным Серверам Termidesk Connect будет использовать один пул IP-адресов (IP-Фонд);
- соответственно, для Проверок будут использоваться другие IP-адреса, указанные непосредственно в Проверках.

```
set health-check id <имя> ICMP source-ip <IP-адрес>
```

- (опционально) указание IP-адреса назначения, если он отличается от IP-адреса, указанного в настройках Реального Сервера:

```
set health-check id <имя> ICMP target-ip <IP-адрес>
```

- (опционально) включение или отключение отслеживания Проверки для готовности узла к переходу в состояние `ACTIVE` (по умолчанию – `false`):

Подробное описание условий переключения узлов отказоустойчивой конфигурации приведено в подразделе **Отказоустойчивость**.

```
set health-check id <имя> ICMP ha-monitor <true/false>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml health-check id <имя>
```

- просмотр выполненных команд:

```
show configuration cli health-check id <имя>
```

Пример конфигурации ICMP-Проверки:

```
set health-check id icmp-new
```

```
set health-check id icmp-new ICMP
set health-check id icmp-new ICMP interval 10
set health-check id icmp-new ICMP timeout 5
set health-check id icmp-new ICMP try 1
set health-check id icmp-new ICMP success-try 1
set health-check id icmp-new ICMP target-ip 192.0.2.10
```

Создание и настройка TCP-Проверки

Создание и настройка TCP-Проверок выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Проверки**).

Для создания и настройки TCP-Проверки используется следующий набор команд:

Описание параметров также приведено в подразделе **Объект** `health-check`.

- создание Проверки:

```
set health-check id <имя>
```

- назначение протокола для Проверки:

```
set health-check id <имя> TCP
```

- задание интервала (в секундах) Проверки (по умолчанию – `10`):

```
set health-check id <имя> TCP interval <значение>
```

- задание времени (в секундах) ожидания ответа на запрос (по умолчанию – `5`):

```
set health-check id <имя> TCP timeout <значение>
```

- задание количества повторных Проверок в случае отсутствия ответа (по умолчанию – `1`):

```
set health-check id <имя> TCP try <значение>
```

- задание необходимого количества успешных Проверок после восстановления связности (по умолчанию – `1`):

```
set health-check id <имя> TCP success-try <значение>
```

- указание VRF для Проверки (по умолчанию – `default`):

```
set health-check id <имя> TCP vrf <имя_VRF>
```

- (опционально) указание IP-адреса источника (Termidesk Connect), с которого будет выполнена Проверка:

```
set health-check id <имя> TCP source-ip <IP-адрес>
```

- (опционально) указание IP-адреса назначения, если он отличается от IP-адреса, указанного в настройках Реального Сервера:

```
set health-check id <имя> TCP target-ip <IP-адрес>
```

- (опционально) порт назначения Проверки (по умолчанию используется порт Реального Сервера из Группы Реальных Серверов):

```
set health-check id <имя> TCP target-port <порт>
```

- (опционально) включение или отключение отслеживания Проверки для готовности узла к переходу в состояние `ACTIVE` (по умолчанию – `false`):

Подробное описание условий переключения узлов отказоустойчивой конфигурации приведено в подразделе **Отказоустойчивость**.

```
set health-check id <имя> TCP ha-monitor <true/false>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml health-check id <имя>
```

- просмотр выполненных команд:

```
show configuration cli health-check id <имя>
```

Пример конфигурации TCP-Проверки:

```
set health-check id tcp-new
set health-check id tcp-new TCP
set health-check id tcp-new TCP interval 10
set health-check id tcp-new TCP timeout 5
set health-check id tcp-new TCP try 1
set health-check id tcp-new TCP success-try 1
set health-check id tcp-new TCP target-port 88
```

Создание и настройка HTTP-Проверки

Создание и настройка HTTP-Проверок выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Проверки**).

Для создания и настройки HTTP-Проверки используются команды:

Описание параметров также приведено в подразделе **Объект** `health-check`.

- создание Проверки:

```
set health-check id <имя>
```

- назначение протокола для Проверки:

```
set health-check id <имя> HTTP
```

- задание интервала (в секундах) Проверки (по умолчанию – `10`):

```
set health-check id <имя> HTTP interval <значение>
```

- задание времени (в секундах) ожидания ответа на запрос (по умолчанию – `5`):

```
set health-check id <имя> HTTP timeout <значение>
```

- задание количества повторных Проверок в случае отсутствия ответа (по умолчанию – `1`):

```
set health-check id <имя> HTTP try <значение>
```

- задание необходимого количества успешных Проверок после восстановления связности (по умолчанию – 1):

```
set health-check id <имя> HTTP success-try <значение>
```

- указание VRF для Проверки (по умолчанию – default):

```
set health-check id <имя> HTTP vrf <имя_VRF>
```

- (опционально) указание IP-адреса источника (Termidesk Connect), с которого будет выполнена Проверка:

```
set health-check id <имя> HTTP source-ip <IP-адрес>
```

- (опционально) указание IP-адреса назначения, если он отличается от IP-адреса, указанного в настройках Реального Сервера:

```
set health-check id <имя> HTTP target-ip <IP-адрес>
```

- (опционально) порт назначения Проверки (по умолчанию используется порт Реального Сервера из Группы Реальных Серверов):

```
set health-check id <имя> HTTP target-port <порт>
```

- указание метода запроса, по которому выполняется Проверка (по умолчанию – HEAD):

Возможные значения:

- GET – запрос с получением тела ответа;
- HEAD – запрос с получением заголовка ответа;
- POST – отправка данных с телом запроса.

```
set health-check id <имя> HTTP method <метод_запроса>
```

- указание пути, по которому выполняется Проверка (по умолчанию – /):

```
set health-check id <имя> HTTP uri <URI_сервиса>
```

- указание ожидаемых кодов ответов (по умолчанию – 200):

```
set health-check id <имя> HTTP status-codes <коды_ответов>
```

- (опционально) указание строки в ответе, по которой выполняется Проверка:

```
set health-check id <имя> HTTP response-string <строка_ответа>
```

- (опционально) указание заголовка, по которому выполняется Проверка:

```
set health-check id <имя> HTTP headers <заголовок>:<значение>
```

- (опционально) указание кодов ответов для перевода Реального Сервера в режим технического обслуживания:

```
set health-check id <имя> HTTP maintenance-codes <коды_ответов>
```

- (опционально) инверсия результата Проверки (по умолчанию – `false`):

```
set health-check id <имя> HTTP reverse true
```

- (опционально) включение или отключение отслеживания Проверки для готовности узла к переходу в состояние `ACTIVE` (по умолчанию – `false`):

Подробное описание условий переключения узлов отказоустойчивой конфигурации приведено в подразделе **Отказоустойчивость**.

```
set health-check id <имя> HTTP ha-monitor <true/false>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml health-check id <имя>
```

- просмотр выполненных команд:

```
show configuration cli health-check id <имя>
```

Пример конфигурации HTTP-Проверки:

```
set health-check id http_hc
set health-check id http_hc HTTP
set health-check id http_hc HTTP interval 10
set health-check id http_hc HTTP timeout 5
set health-check id http_hc HTTP try 1
set health-check id http_hc HTTP success-try 1
set health-check id http_hc HTTP source-ip 192.0.2.1
set health-check id http_hc HTTP method POST
set health-check id http_hc HTTP uri /
set health-check id http_hc HTTP status-codes 200
set health-check id http_hc HTTP status-codes 202
set health-check id http_hc HTTP response-string
set health-check id http_hc HTTP headers Host:123
set health-check id http_hc HTTP reverse false
```

Создание и настройка HTTPS-Проверки

Создание и настройка HTTPS-Проверок выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Проверки**).

Для создания и настройки HTTPS-Проверки используются команды:

Описание параметров также приведено в подразделе **Объект** `health-check`.

- создание Проверки:

```
set health-check id <имя>
```

- назначение протокола для Проверки:

```
set health-check id <имя> HTTP
```

- задание интервала (в секундах) Проверки (по умолчанию – `10`):

```
set health-check id <имя> HTTP interval <значение>
```

- задание времени (в секундах) ожидания ответа на запрос (по умолчанию – `5`):

```
set health-check id <имя> HTTP timeout <значение>
```

- задание количества повторных Проверок в случае отсутствия ответа (по умолчанию –

1):

```
set health-check id <имя> HTTP try <значение>
```

- задание необходимого количества успешных Проверок после восстановления связности (по умолчанию – 1):

```
set health-check id <имя> HTTP success-try <значение>
```

- указание VRF для Проверки (по умолчанию – `default`):

```
set health-check id <имя> HTTP vrf <имя_VRF>
```

- (опционально) указание IP-адреса источника (Termidesk Connect), с которого будет выполнена Проверка:

```
set health-check id <имя> HTTP source-ip <IP-адрес>
```

- (опционально) указание IP-адреса назначения, если он отличается от IP-адреса, указанного в настройках Реального Сервера:

```
set health-check id <имя> HTTP target-ip <IP-адрес>
```

- (опционально) порт назначения Проверки (по умолчанию используется порт Реального Сервера из Группы Реальных Серверов):

```
set health-check id <имя> HTTP target-port <порт>
```

- указание метода запроса, по которому выполняется Проверка (по умолчанию – `HEAD`):

Возможные значения:

- `GET` – запрос с получением тела ответа;
- `HEAD` – запрос с получением заголовка ответа;
- `POST` – отправка данных с телом запроса.

```
set health-check id <имя> HTTP method <метод_запроса>
```

- указание пути, по которому выполняется Проверка (по умолчанию – `/`):

```
set health-check id <имя> HTTP uri <URI_сервиса>
```

- указание ожидаемых кодов ответов (по умолчанию – `200`, может быть задано несколько):

```
set health-check id <имя> HTTP status-codes <коды_ответов>
```

- (опционально) указание строки в ответе, по которой выполняется Проверка:

```
set health-check id <имя> HTTP response-string <строка_ответа>
```

- (опционально) указание заголовка, по которому выполняется Проверка:

```
set health-check id <имя> HTTP headers <заголовок>:<значение>
```

- (опционально) указание кодов ответов для перевода Реального Сервера в режим технического обслуживания:

```
set health-check id <имя> HTTP maintenance-codes <коды_ответов>
```

- (опционально) инверсия результата Проверки:

```
set health-check id <имя> HTTP reverse true
```

- указание Профиля защищенного соединения, используемого при выполнении Проверки:

Для настройки Профиля защищенного соединения HTTPS-Проверки подходит только Клиентский SSL-Профиль.

```
set health-check id <имя> HTTP ssl-profile-id <имя_Профиля>
```

- (опционально) включение или отключение отслеживания Проверки для готовности узла к переходу в состояние `ACTIVE` (по умолчанию – `false`):

Подробное описание условий переключения узлов отказоустойчивой конфигурации приведено в подразделе **Отказоустойчивость**.

```
set health-check id <имя> HTTP ha-monitor <true/false>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml health-check id <имя>
```

- просмотр выполненных команд:

```
show configuration cli health-check id <имя>
```

Пример конфигурации HTTPS-Проверки:

```
set health-check id https_hc
set health-check id https_hc HTTP
set health-check id https_hc HTTP interval 10
set health-check id https_hc HTTP timeout 5
set health-check id https_hc HTTP try 1
set health-check id https_hc HTTP success-try 1
set health-check id https_hc HTTP source-ip 192.0.2.1
set health-check id https_hc HTTP method POST
set health-check id https_hc HTTP uri /
set health-check id https_hc HTTP status-codes 200
set health-check id https_hc HTTP status-codes 202
set health-check id https_hc HTTP response-string
set health-check id https_hc HTTP headers Host:123
set health-check id https_hc HTTP reverse false
set health-check id https_hc HTTP ssl-profile-id client_default
```

Создание и настройка Пользовательской Проверки

Создание и настройка Пользовательской Проверки выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Проверки**).

Для создания и настройки Пользовательской Проверки используется следующий набор команд:

Описание параметров также приведено в подразделе **Объект** `health-check`.

- создание Проверки:

```
set health-check id <имя>
```

- задание типа Проверки:

```
set health-check id <имя> USER
```

- задание интервала (в секундах) Проверки (по умолчанию – 10):

```
set health-check id <имя> USER interval <значение>
```

- задание времени (в секундах) ожидания ответа на запрос (по умолчанию – 5):

```
set health-check id <имя> USER timeout <значение>
```

- задание количества повторных Проверок в случае отсутствия ответа (по умолчанию – 1):

```
set health-check id <имя> USER try <значение>
```

- задание необходимого количества успешных Проверок после восстановления связности (по умолчанию – 1):

```
set health-check id <имя> USER success-try <значение>
```

- указание VRF для Проверки (по умолчанию – default):

```
set health-check id <имя> USER vrf <имя_VRF>
```

- (опционально) указание IP-адреса источника (Termidesk Connect), с которого будет выполнена Проверка:

```
set health-check id <имя> USER source-ip <IP-адрес>
```

- (опционально) указание IP-адреса назначения, если он отличается от IP-адреса, указанного в настройках Реального Сервера:

```
set health-check id <имя> USER target-ip <IP-адрес>
```

- (опционально) порт назначения Проверки (по умолчанию используется порт Реального Сервера из Группы Реальных Серверов):

```
set health-check id <имя> USER target-port <порт>
```

- указание имени файла скрипта, согласно которому будет выполнена Проверка:

```
set health-check id <имя> USER script <имя_файла_скрипта>
```

- (опционально) включение или отключение отслеживания Проверки для готовности узла к переходу в состояние `ACTIVE` (по умолчанию – `false`):

Подробное описание условий переключения узлов отказоустойчивой конфигурации приведено в подразделе **Отказоустойчивость**.

```
set health-check id <имя> USER ha-monitor <true/false>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
 show configuration xml health-check id <имя>
```

- просмотр выполненных команд:

```
show configuration cli health-check id <имя>
```

Примеры конфигураций Пользовательских Проверок с различными скриптами представлены в таблице (см. таблицу Примеры конфигураций для Пользовательских Проверок).

Таблица 7. Примеры конфигураций для Пользовательских Проверок

Формат скрипта	Конфигурация

`py`	Синтаксис python. Пример:
	`set health-check id user01`
	`set health-check id user01 USER`
	`set health-check id user01 USER interval 7`
	`set health-check id user01 USER timeout 5`
	`set health-check id user01 USER try 2`
	`set health-check id user01 USER success-try 2`
	`set health-check id user01 USER script httptcp.py`

Формат скрипта	Конфигурация

sh	Синтаксис bash. Пример:
	`set health-check id user02`
	`set health-check id user02 USER`
	`set health-check id user02 USER interval 5`
	`set health-check id user02 USER timeout 2`
	`set health-check id user02 USER try 2`
	`set health-check id user02 USER success-try 2`
	`set health-check id user02 USER target-ip 192.0.2.10`
	`set health-check id user02 USER script httptcp.sh`

Создание и настройка Комбинированной Проверки

Создание и настройка Комбинированных Проверок выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Проверки**).

Для создания и настройки Комбинированной Проверки используются команды:

Описание параметров также приведено в подразделе **Объект** `health-check`.

- создание Проверки:

```
set health-check id <имя>
```

- назначение типа Проверки:

```
set health-check id <имя> COMBO
```

- добавление базовой Проверки:

Могут быть добавлены ранее созданные ICMP-Проверки, TCP-Проверки, HTTP-Проверки или Пользовательские Проверки.

  - назначение идентификатора базовой Проверки:

```
set health-check id <имя> COMBO hc-ids <id-Проверки>
```

  - назначение Веса базовой Проверки (от 1 до 255):

Вес базовой Проверки указывает какой вклад вносит привязанная базовая Проверка в определение состояния Комбинированной Проверки. Вес базовой Проверки — параметр, определяющий насколько состояние базовой Проверки влияет на оценку состояния Реального Сервера (`ONLINE` или `OFFLINE`).

```
set health-check id <имя> COMBO hc-ids <id-Проверки> weight <вес_базовой_Проверки>
```

- задание значения общего Веса Проверки:

Общий Вес Проверки — пороговое значение успеха для суммы Весов базовых Проверок. Если к Комбинированной Проверке привязано несколько базовых Проверок, их Вес сравнивается с общим Весом, указанным в настройках Комбинированной Проверки. Реальный Сервер считается `ONLINE`, если сумма Весов всех успешных базовых Проверок равна или превышает значение общего Веса Комбинированной Проверки.

```
set health-check id <имя> COMBO threshold <общий_вес>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml health-check id <имя>
```

- просмотр выполненных команд:

```
show configuration cli health-check id <имя>
```

Скрипты Проверок

Общие сведения о скриптах Проверок

Скрипты Проверок – это файлы, содержащие пользовательскую логику Проверки работоспособности сервиса или узла.

Для создания скрипта Проверки используются файлы в формате:

- `.py` – для python-скрипта;
- `.sh` – для bash-скрипта.

Файлы скриптов Проверок располагаются в каталоге `/var/lib/tdc/lbscripts/hc/`.

Скрипты Проверок позволяют выполнять Проверки по протоколам:

- HTTP/HTTPS;
- TCP.

Параметры Проверки передаются скрипту в качестве аргумента командной строки в формате JSON, и обязательно содержат поля:

- source-ip;
- target-ip;
- target-port.

Для каждой выполняемой Проверки запускается отдельный процесс с собственной копией скрипта Проверки. Большое количество таких процессов может негативно сказаться на производительности системы.

Пример использования python-скрипта:

Описание полей приведено в таблице (см. Описание полей python-скрипта Проверки).

```
import requests
import sys
import json

config = json.loads(sys.argv[1])
target_ip = config["target-ip"]
target_port = config["target-port"]

response = requests.get(f"http://{target_ip}:{target_port}", timeout=5)
response.raise_for_status()
sys.exit(0)
```

Таблица 8. Описание полей python-скрипта Проверки

Поле	Описание
`import requests`	Импортирует модуль `requests` для выполнения HTTP-запросов
`import sys`	Импортирует модуль `sys` для работы с аргументами командной строки
`import json`	Импортирует модуль `json` для обработки данных в формате `JSON`
`config = json.loads(sys.argv[1])`	Читает первый аргумент командной строки, содержащий строку в формате `JSON`, и преобразует его в объект Python
`target_ip = config["target-ip"]`	Извлекает из объекта `config` значение параметра `target-ip`
`target_port = config["target-port"]`	Извлекает из объекта `config` значение параметра `target-port`
`response = requests.get(f"http://{target_ip}:{target_port}", timeout=5)`	Выполняет HTTP-запрос по адресу, составленному из значений параметров `target-ip` и `target-port`, с временем ожидания 5 секунд

Поле	Описание
`response.raise_for_status()`	Генерирует исключение, если код ответа сервера не входит в диапазон значений 200 – 299
`sys.exit(0)`	Завершает работу скрипта с кодом `0`, если Проверка успешна

Проверку с использованием python-скрипта можно упростить, используя логику обработки аргументов командной строки в конфигурационном файле `config.py`.

Файл `config.py` должен располагаться в каталоге `/var/lib/tdc/lbscripts/hc/`.

Пример содержимого файла `config.py` :

Описание полей приведено в таблице (см. Описание полей конфигурационного файла `config.py`).

```
import json

class Config:

    source_ip: str
    target_ip: str
    target_port: str

    def __init__(self, argv):
        if len(argv) != 2:
            raise ValueError(f"List must be the length of two, but has {len(argv)}
elements")

        try:
            config = json.loads(argv[1])
            self.source_ip = config.get("source-ip")
            self.target_ip = config.get("target-ip")
            self.target_port = config.get("target-port")
        except json.JSONDecodeError:
            raise TypeError("List's second element must contain valid JSON object")
```

Таблица 9. Описание полей конфигурационного файла `config.py`

Поле	Описание
`import json`	Импортирует модуль `json` для обработки данных в формате `JSON`
`class Config:`	Задает класс `Config` для хранения параметров Проверки
`source_ip: str`	Атрибут для хранения IP-адреса источника запроса
`target_ip: str`	Атрибут для хранения IP-адреса назначения
`target_port: str`	Атрибут для хранения порта назначения
`def init(self, argv):`	Конструктор класса, принимающий список аргументов командной строки
`if len(argv) != 2:`	Проверяет, что список аргументов содержит ровно два элемента

Поле	Описание
`raise ValueError(…)`	Генерирует исключение, если количество аргументов не равно двум
`config = json.loads(argv[1])`	Читает второй аргумент командной строки, содержащий строку в формате `JSON`, и преобразует его в объект Python
`self.source_ip = config.get("source-ip")`	Сохраняет значение поля `source-ip` в атрибуте `source_ip`
`self.target_ip = config.get("target-ip")`	Сохраняет значение поля `target-ip` в атрибуте `target_ip`
`self.target_port = config.get("target-port")`	Сохраняет значение поля `target-port` в атрибуте `target_port`
`except json.JSONDecodeError:`	Обрабатывает ошибку, если второй аргумент не является корректной JSON-строкой
`raise TypeError(…)`	Генерирует исключение, если аргумент не содержит корректный JSON-объект

Пример python-скрипта с использованием конфигурационного файла `config.py`:

Описание полей приведено в таблице (см. Описание полей python-скрипта с использованием файла `config.py`).

```
import sys
import requests
import config # Импорт файла с классом Config

def check_http(target_ip, target_port):
    url = f"http://{target_ip}:{target_port}"
    try:
        response = requests.head(url, timeout=3)
        print("HEAD: OK")
        response = requests.get(url, timeout=3)
        print("GET: OK")
        return True
    except requests.exceptions.RequestException:
        print("NOT OK")
        return False

config = config.Config(sys.argv) # Использование класса
source_ip = config.source_ip # Получение source-ip
target_ip = config.target_ip # Получение target-ip
target_port = config.target_port # Получение target-port

print(f"Config:")
print(f"  Source IP: {source_ip}")
print(f"  Target IP: {target_ip}")
print(f"  Target Port: {target_port}")
print(f"Checking HTTP server at {target_ip}:{target_port}...", end=" ")

if not check_http(target_ip, target_port):
    sys.exit(1)
```

```
sys.exit(0)
```

Таблица 10. Описание полей python-скрипта с использованием файла `config.py`

Поле	Описание
`import requests`	Импортирует модуль `requests` для выполнения HTTP-запросов
`import sys`	Импортирует модуль `sys` для работы с аргументами командной строки
`import config`	Импортирует модуль `config`, содержащий класс `Config` для обработки аргументов командной строки
`def check_http(target_ip, target_port):`	Определяет функцию `check_http` для выполнения HTTP-Проверки работоспособности узла или сервиса
`url = f"http://{target_ip}:{target_port}"`	Формирует URL-адрес из значений параметров `target_ip` и `target_port`
`response = requests.head(url, timeout=3)`	Выполняет HEAD-запрос по указанному адресу с временем ожидания 3 секунды
`print("HEAD: OK")`	Выводит сообщение об успешном выполнении HEAD-запроса
`response = requests.get(url, timeout=3)`	Выполняет GET-запрос по указанному адресу с временем ожидания 3 секунды
`print("GET: OK")`	Выводит сообщение об успешном выполнении GET-запроса
`return True`	Возвращает значение `True`, если оба запроса завершились без ошибок
`except requests.exceptions.RequestException:`	Перехватывает исключения, возникающие при выполнении HTTP-запросов
`print("NOT OK")`	Выводит сообщение о неуспешной проверке
`return False`	Возвращает значение `False` в случае ошибки при выполнении запросов
`config = config.Config(sys.argv)`	Создает объект класса `Config` и передает ему аргументы командной строки
`source_ip = config.source_ip`	Получает значение параметра `source-ip` из объекта `Config`
`target_ip = config.target_ip`	Получает значение параметра `target-ip` из объекта `Config`
`target_port = config.target_port`	Получает значение параметра `target-port` из объекта `Config`
`print(f"Config:")`	Выводит заголовок конфигурации
`print(f" Source IP: {source_ip}")`	Выводит IP-адрес источника запроса
`print(f" Target IP: {target_ip}")`	Выводит IP-адрес назначения
`print(f" Target Port: {target_port}")`	Выводит порт назначения Проверки
`print(f"Checking HTTP server at {target_ip}:{target_port}…", end=" ")`	Выводит сообщение о начале HTTP-Проверки
`if not check_http(target_ip, target_port):`	Проверяет результат выполнения функции `check_http` для указанного узла или сервиса
`sys.exit(1)`	Завершает работу скрипта с кодом ошибки `1`, если Проверка неуспешна
`sys.exit(0)`	Завершает работу скрипта с кодом `0`, если Проверка успешна

Пример использования bash-скрипта:

Описание полей приведено в таблице (см. Описание полей bash-скрипта Проверки).

```
#!/bin/bash

json_input="$1"
target_ip=$(jq -r '.["target-ip"]' <<< "$json_input")
target_port=$(jq -r '.["target-port"]' <<< "$json_input")

curl -s -o /dev/null "http://${target_ip}:${target_port}"
exit $?
```

Таблица 11. Описание полей bash-скрипта Проверки

Поле	Описание
`json_input="$1"`	Сохраняет первый аргумент командной строки в переменную `json_input`
`target_ip=$(jq -r '.["target-ip"]' <<< "$json_input")`	Извлекает значение параметра `target-ip` из переменной `json_input` с использованием утилиты `jq`
`target_port=$(jq -r '.["target-port"]' <<< "$json_input")`	Извлекает значение параметра `target-port` из переменной `json_input` с использованием утилиты `jq`
`curl -s -o /dev/null "http://${target_ip}:${target_port}"`	Выполняет HTTP-запрос с использованием утилиты `curl` по адресу, составленному из значений параметров `target-ip` и `target-port`, без вывода содержимого ответа и информации о процессе выполнения
`exit $?`	Завершает выполнение скрипта с кодом ответа, который вернула утилита `curl`

Добавление скрипта Проверки

Добавление скрипта Проверки выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см.подраздел **Веб. Проверки**).

Пример команды добавления скрипта Проверки:

```
set health-check id <имя> USER script <имя_файла_скрипта>
```

Файл должен располагаться в каталоге `/var/lib/tdc/lbscripts/hc/`. Для каждой выполняемой Проверки запускается отдельный процесс с собственной копией скрипта Проверки. Большое количество таких процессов может негативно сказаться на производительности системы.

Примеры bash-скриптов для Проверок

Примеры приведены в таблице (см. Примеры bash-скриптов для выполнения Проверки).

Таблица 12. Примеры bash-скриптов для выполнения Проверки

Скрипт	Описание

`#!/bin/bash`
 	
`json_input="$1"`
`target_ip=$(jq -r '.["target-ip"]' <<< "$json_input")`
`target_port=$(jq -r '.["target-port"]' <<< "$json_input")`
 	
`curl -s -o /dev/null "http://${target_ip}:${target_port}"`
`exit $?`
	HTTP/HTTPS-Проверка с использованием утилит `jq` и `curl`

`#!/bin/bash`
 	
`json_input="$1"`
`target_ip=$(jq -r '.["target-ip"]' <<< "$json_input")`
`target_port=$(jq -r '.["target-port"]' <<< "$json_input")`
 	
`nc -z -w 3 "$target_ip" "$target_port"`
	TCP-Проверка с использованием утилит `jq` и `netcat`

Примеры python-скриптов для Проверок

Примеры приведены в таблице (см. Примеры python-скриптов для выполнения Проверки).

Таблица 13. Примеры python-скриптов для выполнения Проверки

Скрипт	Описание

`import requests`
`import sys`
`import json`
 	
`config = json.loads(sys.argv[1])`
`target_ip = config["target-ip"]`
`target_port = config["target-port"]`
 	
`response = requests.get(f"http://{target_ip}:{target_port}", timeout=5)`
`response.raise_for_status()`
`sys.exit(0)`
	HTTP/HTTPS-Проверка

Скрипт	Описание

`import socket`
`import sys`
`import json`
 	
`config = json.loads(sys.argv[1])`
`target_ip = config["target-ip"]`
`target_port = config["target-port"]`
 	
`socket.create_connection((target_ip,`
`target_port), timeout=5)`
`sys.exit(0)`
	TCP-Проверка

Группы Реальных Серверов

Общие сведения о Реальных Серверах

Реальный Сервер в Termidesk Connect представляет собой IP-адрес (или доменное имя) и порт узла с установленным приложением (ресурсом), доступ к которому предоставляется пользователю. Доступ пользователя к Реальным Серверам балансируется Серверами Балансировки.

Группа Реальных Серверов – объединение нескольких Реальных Серверов и их периодических проверок.

Взаимодействие с Группой Реальных Серверов представлено на рисунке (см. Взаимодействие с Группой Реальных Серверов).

Рисунок 54. Взаимодействие с Группой Реальных Серверов

Создание и настройка Группы Реальных Серверов

Создание и настройка Группы Реальных Серверов выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;

- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Группы Реальных Серверов**).

Для создания и настройки Группы Реальных Серверов используются команды:

Описание параметров также приведено в подразделе **Объект** `rs-pool`.

- создание Группы Реальных Серверов:

Имя создаваемого объекта не должно содержать дефисов, нижнее подчеркивание использовать разрешено.

```
set rs-pool id <имя>
```

- назначение Реального Сервера Группе (может быть задано несколько узлов, доступны разные варианты назначения):
  - либо по IP-адресу:

```
set rs-pool id <имя> rs <IP-адрес> <порт>
```

  - либо по доменному имени узла:

Задание Группы Реальных Серверов по доменному имени узла полезно для следующих сценариев:

- IP-адрес Реального Сервера периодически меняется. В этом случае исключается необходимость ручного редактирования конфигурации при изменении IP-адресов;
- требуется автоматическое масштабирование Группы Реальных Серверов;
- инфраструктура организации настроена таким образом, что все узлы взаимодействуют друг с другом исключительно по доменному имени.

```
set rs-pool id <имя> rs-domain <доменное_имя_узла> <порт>
```

Поддерживается сохранение и передача порта из запроса на Реальный Сервер без изменений. Для этого нужно указать `0` при назначении Реального Сервера:

```
set rs-pool id <имя> rs <IP-адрес> 0
set rs-pool id <имя> rs-domain <доменное_имя_узла> 0
```

- если Группа Реальных Серверов задана по IP-адресу, доступны настройки:
  - задание Веса Реального Сервера (по умолчанию – 1):

```
set rs-pool id <имя> rs <IP-адрес> <порт> weight <значение>
```

- если Группа Реальных Серверов задана по доменному имени узла, доступны настройки:
  - (опционально) задание автоматического масштабирования Группы Реальных Серверов (по умолчанию – `false`):

> При добавлении Реального Сервера по доменному имени выполняется разрешение имени в IP-адрес или IP-адреса, которые используются в алгоритме балансировки до истечения времени, заданного в TTL. Выбор целевых IP-адресов Реального Сервера определяется настройками автоматического масштабирования:
>
> - если выбрано значение `false`, то целевым IP-адресом выбирается первый IP-адрес в DNS-ответе (если их было несколько);
> - если выбрано значение `true`, то целевыми IP-адресами выбираются все IP-адреса в DNS-ответе (если их было несколько).

```
set rs-pool id <имя> rs-domain <доменное_имя_узла> <порт> autoscale
<true/false>
```

  - задание времени жизни (в секундах) информации о доменном имени (по умолчанию – `60`):

> По истечении времени TTL Termidesk Connect делает новый запрос и обновляет данные об IP-адресе (IP-адресах) Реального Сервера.

```
set rs-pool id <имя> rs-domain <доменное_имя_узла> <порт> TTL <значение>
```

  - задание Веса Реального Сервера (по умолчанию – `1`):

```
set rs-pool id <имя> rs-domain <доменное_имя_узла> <порт> weight <значение>
```

- (опционально) назначение ранее созданной Проверки для Группы Реальных Серверов:

```
set rs-pool id <имя> hc-id <имя>
```

- (опционально) указание IP-Фонда (см. подраздел **Сеть**):

```
set rs-pool id <имя> ipset-id <имя>
```

- (опционально) привязка Профиля ограничения скорости (см. подраздел **Профили ограничения скорости**):

```
set rs-pool id <имя> rl-profile-id <имя_Профиля>
```

- (опционально) задание комментария, который будет привязан к Группе Реальных Серверов:

Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.

```
set rs-pool id <имя> description <комментарий>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml rs-pool id <имя>
```

- просмотр выполненных команд:

```
show configuration cli rs-pool id <имя>
```

Пример конфигурации Группы Реальных Серверов, заданной по IP-адресу:

```
set id rs_port_22
set id rs_port_22 rs 192.0.2.1 22 weight 1
set id rs_port_22 ipset-id ha_node_1
set id rs_port_22 hc-id combo
set id rs_port_22 description "tcp rs"
```

Состояние Реальных Серверов

Одной из основных функций Termidesk Connect является интеллектуальный выбор Реального Сервера для подключения пользователя к нему.

Состояние Реального Сервера определяется Проверкой, однако для гибкого управления составом балансируемых Реальных Серверов предусмотрены следующие действия, выполняемые вручную:

- перевод в выключенное состояние;
- перевод во включенное состояние;
- перевод в режим техобслуживания. Режим техобслуживания предназначен для проведения плановых регламентных или аварийных работ на Реальном Сервере.

Описание состояний приведены в таблице (см. Состояния Реального Сервера).

Таблица 14. Состояния Реального Сервера

Состояние	Описание
«В работе»	Реальный Сервер участвует в выборе алгоритма балансировки и может быть использован для привязки сессии пользователя (persistence)
«Отключен»	Реальный Сервер не может быть выбран при работе алгоритма балансировки и не может быть использован для привязки сессии пользователя. Все записи о привязке, ассоциированные с этим Реальным Сервером, удаляются
«Обслуживание»	Уже существующие сессии пользователей не обрываются, но новые сессии не направляются на этот Реальный Сервер. Записи о привязке сессии пользователя не удаляются
«Неизвестно»	Проверка не назначена Группе Реальных Серверов. Реальный Сервер участвует в выборе алгоритма балансировки и может быть использован для привязки сессии пользователя

Состояние Группы Реальных Серверов также зависит от состояния Реальных Серверов в ней (см. Состояния Группы Реального Сервера).

Таблица 15. Состояния Группы Реального Сервера

Состояние	Описание
«В работе»	Все Реальные Серверы в состоянии «В работе»
«Частично в работе»	Хотя бы один Реальный Сервер в состоянии «В работе»
«Отключен»	Все Реальные Серверы в состоянии «Отключен»
«Обслуживание»	Все Реальные Серверы в состоянии «Обслуживание»
«Неизвестно»	Проверка не назначена Группе Реальных Серверов или не применимы другие состояния

Ручное изменение состояния Реального Сервера может выполняться:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Группы Реальных Серверов**).

Для ручного перевода Реального Сервера в соответствующий режим используются команды:

Описание параметров также приведено в подразделе **Объект** `rs-pool`.

- перевод в выключенное состояние:
  - если Реальный Сервер задан по IP-адресу:

```
set rs-pool id <имя> rs <IP-адрес> <порт> state DISABLE
```

  - если Реальный Сервер задан по доменному имени узла:

```
set rs-pool id <имя> rs-domain <доменное_имя> <порт> state DISABLE
```

- перевод во включенное состояние:
  - если Реальный Сервер задан по IP-адресу:

```
set rs-pool id <имя> rs <IP-адрес> <порт> state ENABLE
```

  - если Реальный Сервер задан по доменному имени узла:

```
set rs-pool id <имя> rs-domain <доменное_имя> <порт> state ENABLE
```

- перевод в режим техобслуживания:
  - если Реальный Сервер задан по IP-адресу:

```
set rs-pool id <имя> rs <IP-адрес> <порт> state MAINTENANCE
```

  - если Реальный Сервер задан по доменному имени узла:

```
set rs-pool id <имя> rs-domain <доменное_имя> <порт> state MAINTENANCE
```

Для режима техобслуживания доступно указание времени (в секундах), по истечении которого все сессии (если они остались), сбросятся, а записи привязки сессии пользователя удалятся. Время указывается для Группы Реальных Серверов (по умолчанию – `0`, отсчет времени отключен):

```
set rs-pool id <имя> maintenance-timeout <значение>
```

Профили

Общие сведения о Профилях

Профили определяют настройки и параметры, используемые для обработки и управления трафиком. Важно понимать различия между Серверными и Клиентскими Профилями:

- Серверный Профиль настраивается для взаимодействия между пользователем и Termidesk Connect;
- Клиентский Профиль настраивается для взаимодействия между Termidesk Connect и Реальным Сервером.

В Termidesk Connect существуют следующие Профили для обработки и управления трафиком:

- TCP-Профиль, используемый для TCP-соединения;
- HTTP-Профиль, используемый для работы с HTTP-запросами;
- Профиль сохранения сессий, используемый для привязки пользователя к Реальному

Серверу.

TCP-Профили

Создание и настройка Серверного TCP-Профиля

Создание и настройка Серверного TCP-Профиля выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Профили**).

Описание параметров также приведено в подразделе **Объект** `tcp-profile`.

Для создания и настройки Серверного TCP-Профиля используются команды:

- создание Серверного TCP-Профиля:

```
set tcp-profile server <имя>
```

По умолчанию доступен Серверный TCP-Профиль `tcpp-server-default`, который нельзя удалить или изменить.

- задание времени ожидания (в секундах) записи в сокет (по умолчанию – `60`):

```
set tcp-profile server <имя> write-timeout <значение>
```

- активация или отключение использования алгоритма Нейгла (по умолчанию – `true`):

```
set tcp-profile server <имя> tcp-nodelay <true/false>
```

- задание размера буфера (в байтах) для чтения (по умолчанию – `32768`):

```
set tcp-profile server <имя> buffer-size <значение>
```

- указание алгоритма предотвращения перегрузок (по умолчанию – `CUBIC`):

Алгоритм может быть:

- `BBR`;
- `BIC`;
- `CDG`;
- `CUBIC`;
- `DCTCP`;
- `HIGHSPEED`;
- `HTCP`;

- HYBLA;
- ILLINOIS;
- LP;
- NV;
- RENO;
- VEGAS;
- VENO;
- WESTWOOD;
- YEAH.

```
set tcp-profile server <имя> cc <алгоритм>
```

- (опционально) настройка PROXY-протокола для входящего соединения:

Если активирован PROXY-протокол, Termidesk Connect ожидает от пользователя получение заголовка PROXY-протокола перед началом обмена данными.

Спецификация PROXY-протокола: https://www.haproxy.org/download/1.8/doc/proxy-protocol.txt.

- указание версии PROXY-протокола:

Для Серверного TCP-Профиля возможно указать обе версии PROXY-протокола: 1 и 2. Команда позволяет за один раз добавить только одну версию.

Если указаны одновременно обе версии, Termidesk Connect автоматически определит версию заголовка по его сигнатуре.

Версия может быть:

- v1;
- v2.

```
set tcp-profile client <имя> proxy-protocol <версия>
```

- указание времени ожидания (в секундах) получения заголовка PROXY-протокола от пользователя (по умолчанию – 10):

```
set tcp-profile client <имя> proxy-protocol timeout <время>
```

- разрешение подключения пользователя без передачи заголовка PROXY-протокола (по умолчанию – false):

```
set tcp-profile client <имя> proxy-protocol allow-con-without-proxy-protocol
```

```
<true/false>
```

- (опционально) если выбран PROXY-протокол версии 1, выполняется его настройка:
  - разрешение приема заголовка `PROXY UNKNOWN` (по умолчанию – `false`):

```
set tcp-profile client <имя> proxy-protocol v1 allow-unknown <true/false>
```

- (опционально) если выбран PROXY-протокол версии 2, выполняется его настройка:
  - разрешение приема заголовка, содержащего адрес `AF_UNIX` (по умолчанию – `false`):

```
set tcp-profile client <имя> proxy-protocol v1 allow-af-unix <true/false>
```

  - разрешение приема заголовка, содержащего адрес `UNSPEC` (по умолчанию – `false`):

```
set tcp-profile client <имя> proxy-protocol v1 allow-af-unspec <true/false>
```

  - разрешение приема заголовка, содержащего команду `LOCAL` (по умолчанию – `false`):

```
set tcp-profile client <имя> proxy-protocol v1 allow-local <true/false>
```

- (опционально) настройка проверки активности соединения:
  - активация или отключение проверки активности соединения (по умолчанию – `false`):

```
set tcp-profile server <имя> keep-alive enable <true/false>
```

  - указание времени бездействия (в секундах) соединения перед отправкой пакетов проверки (по умолчанию – `900`):

```
set tcp-profile server <имя> keep-alive timeout <значение>
```

  - указание интервала (в секундах) отправки пакетов проверки (по умолчанию – `75`):

```
set tcp-profile server <имя> keep-alive interval <значение>
```

  - указание количества пакетов проверки, которые следует отправить при отсутствии подтверждения от узла (по умолчанию – `3`):

Если узел в какой-то момент перестанет присылать подтверждение, то Termidesk Connect отправит указанное количество пакетов проверки, прежде чем считать узел вышедшим из строя.

```
set tcp-profile server <имя> keep-alive probe <значение>
```

- (опционально) задание комментария, который будет привязан к Серверному TCP-Профилю:

Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.

```
set tcp-profile server <имя> description <комментарий>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml tcp-profile server <имя>
```

- просмотр выполненных команд:

```
show configuration cli tcp-profile server <имя>
```

Создание и настройка Клиентского TCP-Профиля

Создание и настройка Клиентского TCP-Профиля выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Профили**).

Описание параметров также приведено в подразделе **Объект** `tcp-profile`.

Для создания и настройки Клиентского TCP-Профиля используются команды:

- создание Клиентского TCP-Профиля:

```
set tcp-profile client <имя>
```

По умолчанию доступен Клиентский TCP-Профиль `tcpp-client-default`,

который нельзя удалить или изменить.

- задание времени ожидания (в секундах) записи в сокет (по умолчанию – 60):

```
set tcp-profile client <имя> write-timeout <значение>
```

- активация или отключение использования алгоритма Нейгла (по умолчанию – true):

```
set tcp-profile client <имя> tcp-nodelay <true/false>
```

- задание размера буфера (в байтах) для чтения (по умолчанию – 32768):

```
set tcp-profile client <имя> buffer-size
```

- указание алгоритма предотвращения перегрузок (по умолчанию – CUBIC):

Алгоритм может быть:

- BBR;
- BIC;
- CDG;
- CUBIC;
- DCTCP;
- HIGHSPEED;
- HTCP;
- HYBLA;
- ILLINOIS;
- LP;
- NV;
- RENO;
- VEGAS;
- VENO;
- WESTWOOD;
- YEAH.

```
set tcp-profile client <имя> cc <алгоритм>
```

- указание времени ожидания (в секундах) соединения с Реальным Сервером (по умолчанию – 10):

```
set tcp-profile client <имя> connect-timeout <значение>
```

- указание времени (в секундах) отсутствия данных в сессии (по умолчанию – 120):

```
set tcp-profile client <имя> idle-timeout <значение>
```

- (опционально) настройка PROXY-протокола для исходящего соединения:

Данные PROXY-протокола из входящих соединений доступны в lua -скриптах (см. подраздел **Сценарии**).

Спецификация PROXY-протокола: https://www.haproxy.org/download/1.8/doc/proxy-protocol.txt.

  - активация PROXY-протокола версии 1 или 2:

Для Клиентского TCP-Профиля возможно указать только одну версию PROXY-протокола: 1 или 2.

```
set tcp-profile client <имя> proxy-protocol <v1/v2>
```

  - активация или отключение формирования адреса для заголовка PROXY-протокола (по умолчанию – true):

Значение может быть:

    - true – Termidesk Connect передаст на Реальный Сервер адрес источника и адрес назначения, полученные из заголовка PROXY-протокола пользовательского подключения (настроенного на Серверном TCP-Профиле);
    - false – Termidesk Connect сформирует адреса по соответствию:
        - адрес источника — адрес пользователя;
        - адрес назначения — адрес Реального Сервера.

```
set tcp-profile client <имя> proxy-protocol pass-through-addr <true/false>
```

- (опционально) если выбран PROXY-протокол версии 2, выполняется его настройка:

Особые случаи:

  - если указан тип type=0x03 (PP2_TYPE_CRC32C) (с любым значением параметра value), Termidesk Connect автоматически добавит TLV с рассчитанной контрольной суммой;
  - если указан тип type=0x05 (PP2_TYPE_UNIQUE_ID) с пустым значением параметра value, Termidesk Connect автоматически сформирует уникальный идентификатор;
  - если указано одно и тоже значение типа TLV в параметрах added-tlv

и `pass-through-tlvs`, то приоритет отдается значению из параметра `pass-through-tlvs`.

- указание типа TLV, который будет передан из пользовательского подключения в сторону Реального Сервера:

  Может быть задано несколько TLV.

  Команда позволяет за один раз добавить только один тип TLV.

```
set tcp-profile client <имя> proxy-protocol v2 pass-through-tlvs <тип>
```

- указание типа TLV для добавления в заголовок PROXY-протокола:

  Может быть задано несколько TLV.

  Команда позволяет за один раз добавить только один тип TLV.

```
set tcp-profile client <имя> proxy-protocol v2 added-tlv <тип>
```

- указание формата TLV для добавления в заголовок PROXY-протокола (по умолчанию – `STR`):

  Формат может быть:

  - `STR` – строка;
  - `BASE64` – бинарные данные закодированные в формате `Base64`.

```
set tcp-profile client <имя> proxy-protocol v2 added-tlv <тип> format
<формат>
```

- (опционально) указание значения TLV для добавления в заголовок PROXY-протокола:

```
set tcp-profile client <имя> proxy-protocol v2 added-tlv <тип> value
<значение>
```

- (опционально) настройка проверки активности соединения:
  - активация или отключение проверки активности соединения (по умолчанию – `false`):

```
set tcp-profile client <имя> keep-alive enable <true/false>
```

  - указание времени бездействия (в секундах) соединения перед отправкой пакетов проверки (по умолчанию – `900`):

```
set tcp-profile client <имя> keep-alive timeout <значение>
```

- указание интервала (в секундах) отправки пакетов проверки (по умолчанию – 75):

```
set tcp-profile client <имя> keep-alive interval <значение>
```

- указание количества пакетов проверки, которые следует отправить при отсутствии подтверждения от узла (по умолчанию – 3):

Если узел в какой-то момент перестанет присылать подтверждение, то Termidesk Connect отправит указанное количество пакетов проверки, прежде чем считать узел вышедшим из строя.

```
set tcp-profile client <имя> keep-alive probe <значение>
```

- (опционально) задание комментария, который будет привязан к Клиентскому TCP-Профилю:

Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.

```
set tcp-profile client <имя> description <комментарий>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – XML, JSON или TXT):

```
show configuration xml tcp-profile client <имя>
```

- просмотр выполненных команд:

```
show configuration cli tcp-profile client <имя>
```

HTTP-Профили

Создание и настройка Серверного HTTP-Профиля

Создание и настройка Серверного HTTP-Профиля выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Профили**).

Описание параметров также приведено в подразделе **Объект** `http-profile`.

Для создания и настройки Серверного HTTP-Профиля используются команды:

- создание Серверного HTTP-Профиля:

По умолчанию доступен Серверный HTTP-Профиль `httpp-server-default`, который нельзя удалить или изменить.

```
set http-profile server <имя>
```

- задание времени ожидания (в секундах) чтения из сокета (по умолчанию – `60`):

```
set http-profile server <имя> read-timeout <значение>
```

- задание максимального числа заголовков в запросе, при котором он считается валидным (по умолчанию – `128`):

```
set http-profile server <имя> max-header-count <значение>
```

- задание максимального размера (в байтах) всех заголовков (по умолчанию – `32768`):

```
set http-profile server <имя> max-headers-size <значение>
```

- задание максимального размера (в байтах) одного заголовка (по умолчанию – `24820`):

```
set http-profile server <имя> max-header-length <значение>
```

- (опционально) настройка повторного использования HTTP-соединений:
  - активация или отключение постоянного HTTP-соединения (по умолчанию – `true`):

    ```
    set http-profile server <имя> keep-alive enable <true/false>
    ```

  - указание времени ожидания (в секундах) следующего запроса в соединении (по умолчанию – `20`):

```
set http-profile server <имя> keep-alive timeout <значение>
```

- указание максимального количества запросов, которое может быть отправлено в этом соединении перед его закрытием (по умолчанию – `100`):

```
set http-profile server <имя> keep-alive max-req-count <значение>
```

- указание Сценария ошибок (по умолчанию – `error-reply.lua`):

```
set http-profile server <имя> error-reply <наименование_Сценария>
```

- задание максимального размера (в байтах) тела HTTP-запроса (по умолчанию – `0`):

Возможные значения:

- `0` – размер тела запроса не ограничивается. В Сценарий балансировки значение тела запроса не передается (передается только заголовок запроса);
- значение больше `0` – тело запроса считывается в оперативную память и передается в Сценарий балансировки. При этом параметр `proxy-100` Клиентского HTTP-Профиля не учитывается: Termidesk Connect всегда самостоятельно отправляет клиенту `100-Continue`. Если размер тела запроса, переданного клиентом, больше указанного лимита, то соединение будет сброшено, Termidesk Connect отправит ответ `404`.

```
set http-profile server <имя> body-size-limit <значение>
```

- (опционально) задание комментария, который будет привязан к Клиентскому HTTP-Профилю:

Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.

```
set http-profile server <имя> description <комментарий>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml http-profile server <имя>
```

- просмотр выполненных команд:

```
show configuration cli http-profile server <имя>
```

Создание и настройка Клиентского HTTP-Профиля

Создание и настройка Клиентского HTTP-Профиля выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Профили**).

Описание параметров также приведено в подразделе **Объект** `http-profile`.

Для создания и настройки Клиентского HTTP-Профиля используются команды:

- создание Клиентского HTTP-Профиля:

По умолчанию доступен Клиентский HTTP-Профиль `httpp-client-default`, который нельзя удалить или изменить.

```
set http-profile client <имя>
```

- указание значений (может быть несколько) из заголовка Upgrade, для которых разрешена смена протокола (по умолчанию – `websocket`):

```
set http-profile client <имя> upgrade-types <значение>
```

- задание времени ожидания (в секундах) чтения из сокета (по умолчанию – `60`):

```
set http-profile client <имя> read-timeout <значение>
```

- задание максимального числа заголовков в запросе, при котором он считается валидным (по умолчанию – `128`):

```
set http-profile client <имя> max-header-count <значение>
```

- задание максимального размера (в байтах) всех заголовков (по умолчанию – `32768`):

```
set http-profile client <имя> max-header-size <значение>
```

- задание максимального размера (в байтах) одного заголовка (по умолчанию – `24820`):

```
set http-profile client <имя> max-header-length <значение>
```

- активация или отключение проксирования метода CONNECT с переключением на TCP (по умолчанию – `false`):

```
set http-profile client <имя> allow-connect <true/false>
```

- активация или отключение обработки заголовка `Expect: 100-continue` (по умолчанию – `false`):

Возможные значения:

- `true` – заголовок `Expect: 100-continue` будет передан на Реальный Сервер, а ответ будет транслирован пользователю;
- `false` – заголовок `Expect: 100-continue` будет удален, а Termidesk Connect сгенерирует ответ `100-Continue`

```
set http-profile client <имя> proxy-100 <true/false>
```

- задание максимального размера (в байтах) тела HTTP-ответа (по умолчанию – `0`):

Возможные значения:

- `0` – размер тела ответа не ограничивается. В Сценарий модификации ответа значение тела ответа не передается;
- значение больше `0` – тело ответа считывается в оперативную память и передается в Сценарий модификации ответа.

```
set http-profile client <имя> body-size-limit <значение>
```

- (опционально) задание комментария, который будет привязан к Клиентскому HTTP-Профилю:

Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.

```
set http-profile client <имя> description <комментарий>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – XML, JSON или TXT):

```
show configuration xml http-profile client <имя>
```

- просмотр выполненных команд:

```
show configuration cli http-profile client <имя>
```

Профили сохранения сессий

Общие сведения о Профилях сохранения сессий

Компоненты приложения могут быть доступны через Виртуальный Сервер или Сервер Балансировки. При этом для работоспособности приложения необходимо сохранять за пользователем ранее выбранный Реальный Сервер независимо от того, через какой Виртуальный Сервер или Сервер Балансировки пользователь на него попадает. Для решения этой задачи используется Профиль сохранения сессий.

Профиль сохранения сессий задает настройки параметров для привязки пользователя к Реальному Серверу и привязывается к Серверу Балансировки (с типом TCP или HTTP). Профиль может быть переиспользован на нескольких Серверах Балансировки.

При выборе Профиля сохранения сессий с типом, который не поддерживается на Сервере Балансировки, сохранение сессии не будет поддерживаться на этом Сервере Балансировки.

Все Реальные Серверы в группе должны иметь разные IP-адреса (это не касается портов) для успешного сохранения сессий. Взаимодействие с Реальным Сервером через Профиль сохранения сессий представлено на рисунке (см. рисунок Взаимодействие с Реальным Сервером через Профиль сохранения сессий).

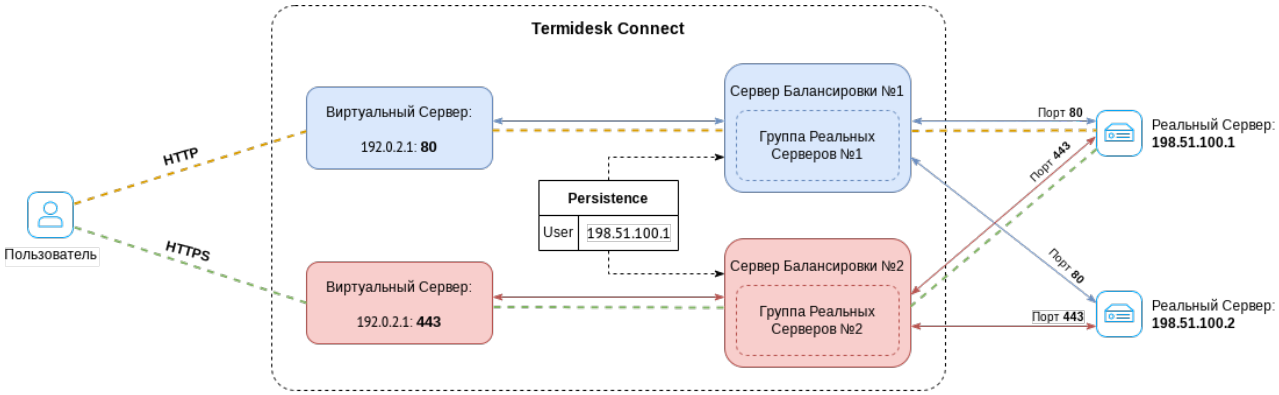

Рисунок 55. Взаимодействие с Реальным Сервером через Профиль сохранения сессий

Создание и настройка Профиля сохранения сессий

Создание и настройка Профиля сохранения сессий выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Профили**).

Описание параметров также приведено в подразделе **Объект** `persistence-profile`.

Для создания и настройки Профиля сохранения сессий используются команды:

- создание Профиля сохранения сессий:

```
set persistence-profile id <имя>
```

- указание типа Профиля сохранения сессий (по умолчанию – `IPSOURCE`):

Возможные значения:

- `COOKIE` – привязка по cookie, который получен в ответе Реального Сервера. Значение cookie, указанного в конфигурации, записывается в персистентную таблицу. Последующие запросы пользователя с данным cookie отправляются на этот Реальный Сервер;
- `HEADER` – привязка по значению заголовка, указанного в конфигурации. Этот алгоритм независим от TCP(IP) параметров подключения;
- `IPSOURCE` – привязка по IP-адресу источника запроса, т.е. IP-адресу пользователя;
- `SSLSESSION` – привязка пользователя к Реальному Серверу по идентификатору SSL-сессии, являющегося частью процесса установления соединения с выбранным сервером. Последующие запросы пользователя с данным идентификатором отправляются на ранее выбранный Реальный Сервер.

```
set persistence-profile id <имя> algorithm <значение>
```

- (опционально, если задан тип `IPSOURCE`) задание времени ожидания (в секундах), в течение которого повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер (по умолчанию – `60`):

```
set persistence-profile id <имя> ipsource-param timeout <значение>
```

- (опционально, если задан тип `COOKIE`) задание имени cookie, который ожидается в ответе Реального Сервера для повторного подключения пользователя на этот Реальный Сервер:

```
set persistence-profile id <имя> cookie-param cookie-name <значение>
```

- (опционально, если задан тип `COOKIE`) задание времени ожидания (в секундах), в

течение которого повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер (по умолчанию – 60):

```
set persistence-profile id <имя> cookie-param timeout <значение>
```

- (опционально, если задан тип HEADER) задание имени HTTP-заголовка, по которому повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер:

```
set persistence-profile id <имя> header-param header-name <значение>
```

- (опционально, если задан тип HEADER) задание времени ожидания (в секундах), в течение которого повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер (по умолчанию – 60):

```
set persistence-profile id <имя> header-param timeout <значение>
```

- (опционально, если задан тип SSLSESSION) задание времени ожидания (в секундах), в течение которого повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер (по умолчанию – 60):

```
set persistence-profile id <имя> sslsession-param timeout <значение>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – XML, JSON или TXT):

```
show configuration xml persistence-profile id <имя>
```

- просмотр выполненных команд:

```
show configuration cli persistence-profile id <имя>
```

Серверы Балансировки

Общие сведения о Серверах Балансировки

Сервер Балансировки в Termidesk Connect – это объект (абстракция), с заданным алгоритмом балансировки и другими параметрами, реализующий перенаправление подключения пользователя на один из Реальных Серверов.

Взаимодействие с Сервером Балансировки представлен на рисунке (см. рисунок Взаимодействие с Сервером Балансировки).

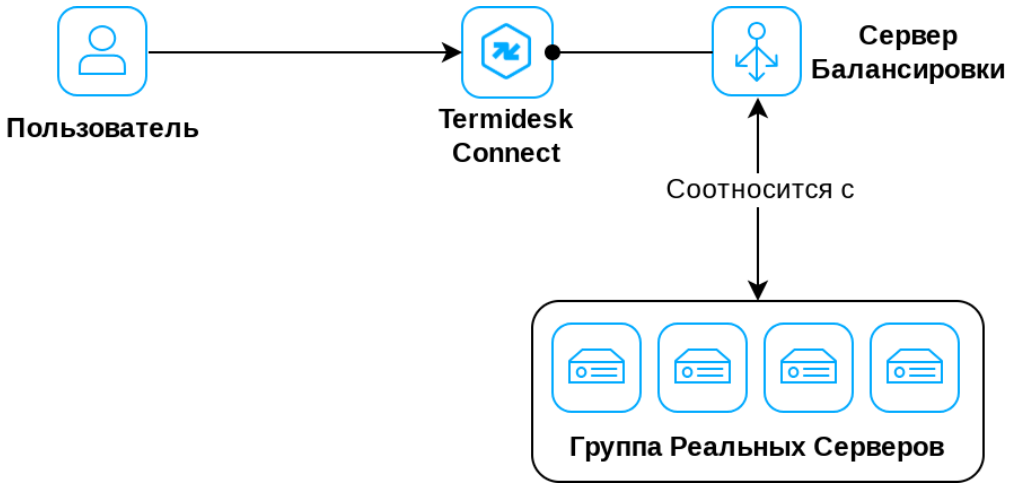

Рисунок 56. Взаимодействие с Сервером Балансировки

Сервер Балансировки может работать в следующих режимах обработки клиентского IP-адреса:

- сохранение клиентского IP-адреса (см. рисунок Режим сохранения клиентского IP-адреса). В этом случае Termidesk Connect выбирает Сервер Балансировки из списка и подменяет IP-адрес назначения, пришедший в пакете, на IP-адрес назначения Реального Сервера;

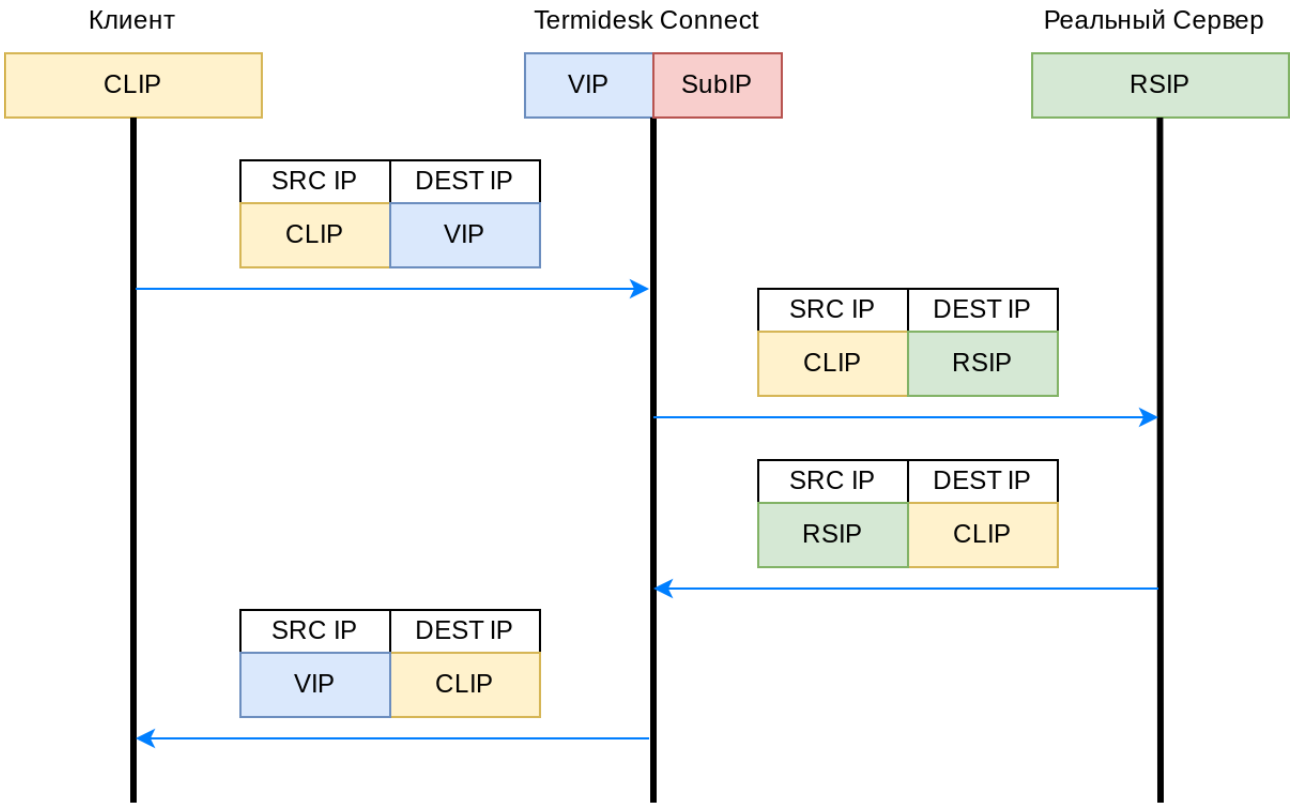

Рисунок 57. Режим сохранения клиентского IP-адреса

- подмена клиентского IP-адреса (см. рисунок Режим подмены клиентского IP-адреса). В этом случае Termidesk Connect меняет как IP-адрес назначения, так и IP-адрес источника (клиента). В данном случае в качестве IP-адреса источника на Реальном Сервере будет виден:
  - или IP-адрес из IP-Фонда (если он используется);
  - или IP-адрес Termidesk Connect, выбираемый согласно таблице маршрутизации.

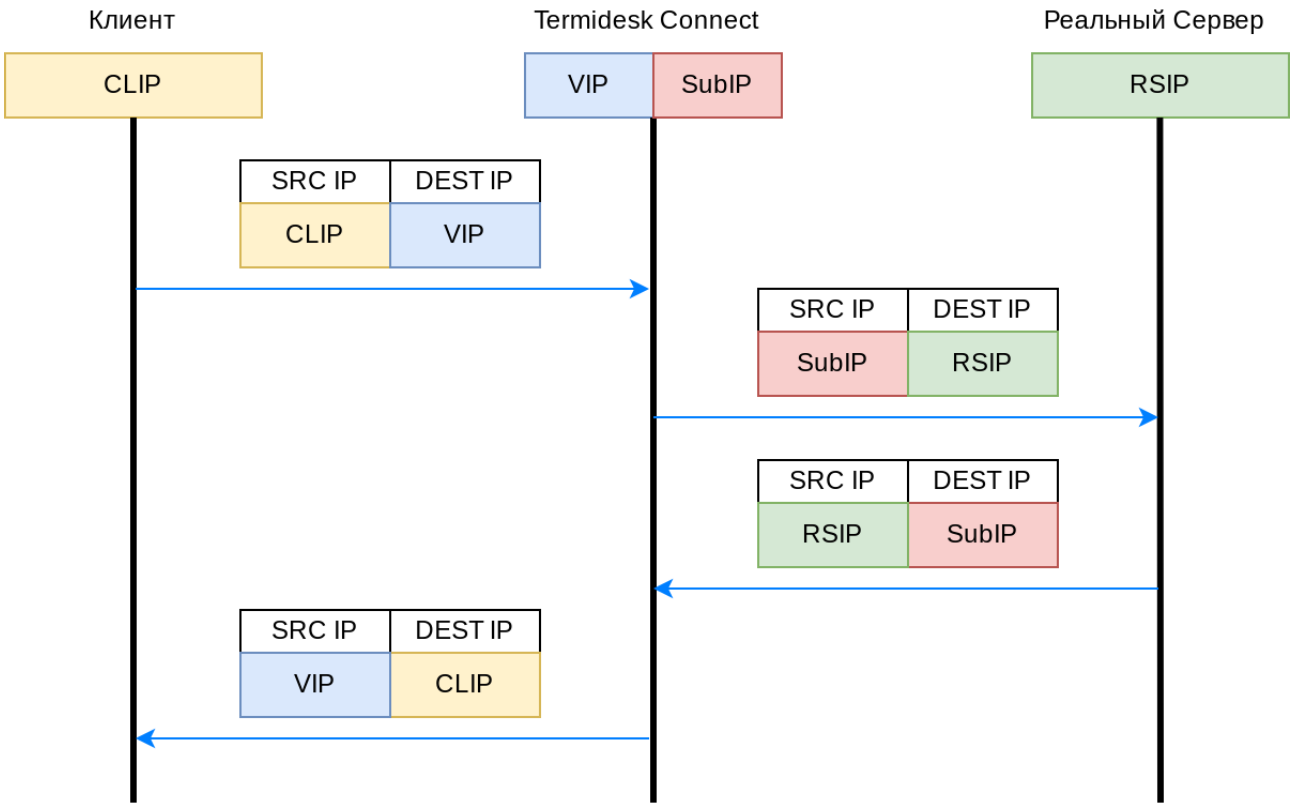

Рисунок 58. Режим подмены клиентского IP-адреса

Если для сохранения клиентского IP-адреса необходимо использовать VRF, отличный от `default`, то настройка Группы Реальных Серверов осуществляется через IP-Фонд.

Режим DSR

Общие сведения о режиме DSR

При создании и настройке Сервера Балансировки с типом RAPID-TCP и RAPID-UDP доступен режим работы DSR.

DSR (Direct Server Return) – это режим, при котором пакет от пользователя передается на Реальный Сервер без изменения IP-адресов (источника и назначения). Ответ от Реального Сервера отправляется пользователю напрямую, минуя Termidesk Connect.

Существуют разновидности DSR:

- L2 DSR;
- L3 DSR.

L2 DSR

L2 DSR – режим с подменой MAC-адресов (`MAC`). В этом режиме IP-адреса во входящем пакете остаются неизменными, Termidesk Connect подменяет в пакете только MAC-адреса (MAC-адрес источника – Termidesk Connect, MAC-адрес назначения – Реальный Сервер) и отправляет этот пакет по MAC-адресам на сервер.

Для работы L2 DSR должны быть выполнены условия:

- на Реальном Сервере должен быть настроен loopback-интерфейс, чтобы он знал о виртуальном IP-адресе;

- на loopback-интерфейсе должен быть отключен ARP-фильтр;
- Termidesk Connect и Реальный Сервер должны находится в одной сети.

Схема работы L2 DSR представлена на рисунке (см. рисунок Схема работы L2 DSR).

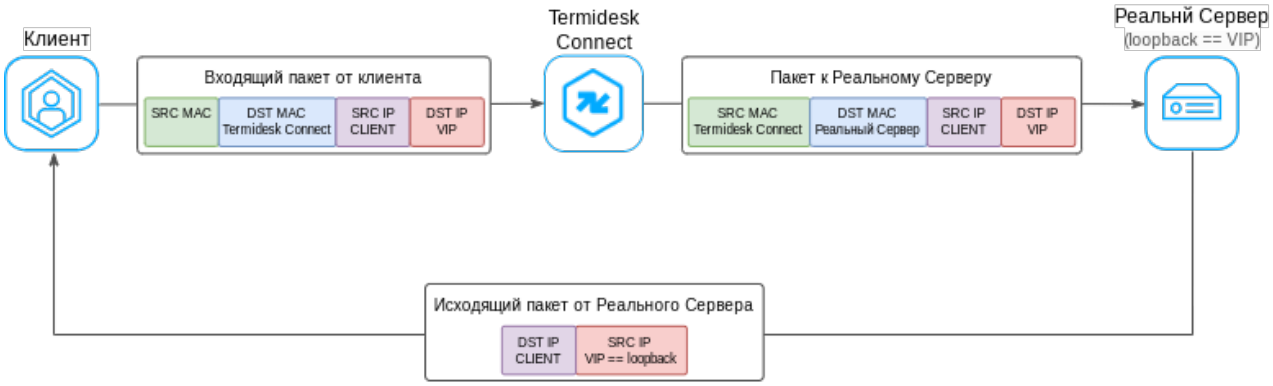

Рисунок 59. Схема работы L2 DSR

Для работы с L2 DSR необходима предварительная настройка Реального Сервера. Пример:

При настройке IP-адреса на loopback-интерфейсе учесть, что этот IP-адрес должен соответствовать IP-адресу Виртуального Сервера.

```
sudo ip addr add 192.168.2.11/24 dev lo label lo
```

Если Реальный Сервер под управлением ОС Astra Linux Special Edition, необходимо выполнить дополнительную настройку для отключения ARP-фильтра. Пример:

```
echo 1 > /proc/sys/net/ipv4/conf/all/arp_ignore
echo 2 > /proc/sys/net/ipv4/conf/all/arp_announce
echo 2 >/proc/sys/net/ipv4/conf/all/rp_filter
echo 2 >/proc/sys/net/ipv4/conf/lo/rp_filter
echo 2 >/proc/sys/net/ipv4/conf/ens192/rp_filter
sysctl -w net.ipv4.ip_forward=1
```

L3 DSR

L3 DSR – режим (IPIP), при котором входящий пакет инкапсулируется в IPIP-туннель и направляется на Реальный Сервер. Далее Реальный Сервер декапсулирует IP-пакет и видит IP-адрес пользователя и IP-адрес Виртуального Сервера.

Для работы L3 DSR должны быть выполнены условия:

- на Реальном Сервере должен быть настроен loopback-интерфейс, чтобы он знал о виртуальном IP-адресе;
- на loopback-интерфейсе должен быть отключен ARP-фильтр;
- на Реальном Сервере должен быть настроен туннельный интерфейс для декапсуляции пакетов.

Схема работы L3 DSR представлена на рисунке (см. рисунок Схема работы L3 DSR).

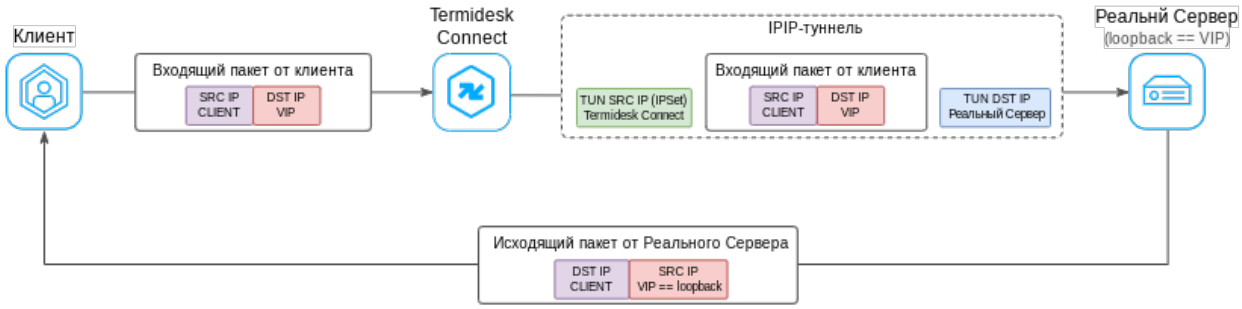

Рисунок 60. Схема работы L3 DSR

Для работы с L3 DSR необходима предварительная настройка Реального Сервера:

- настройка IP-адреса на loopback-интерфейсе. Пример:

При настройке IP-адреса на loopback-интерфейсе учесть, что этот IP-адрес должен соответствовать IP-адресу Виртуального Сервера.

```
sudo ip addr add 192.0.2.11/24 dev lo label lo
```

- настройка туннельного интерфейса. Пример:

Для ОС семейства Debian на туннеле необходимо настроить IP-адрес, который может быть любым.

```
sudo ip link add name ipip0 type ipip external
sudo ip link set up ipip0
sudo ip addr add 127.0.0.99/24 dev ipip0
```

Если Реальный Сервер под управлением ОС Astra Linux Special Edition, необходимо выполнить дополнительную настройку для отключения ARP-фильтра. Пример:

```
echo 1 > /proc/sys/net/ipv4/conf/all/arp_ignore
echo 2 > /proc/sys/net/ipv4/conf/all/arp_announce
echo 2 >/proc/sys/net/ipv4/conf/all/rp_filter
echo 2 >/proc/sys/net/ipv4/conf/lo/rp_filter
echo 2 >/proc/sys/net/ipv4/conf/ipip0/rp_filter
sysctl -w net.ipv4.ip_forward=1
```

Создание и настройка Сервера Балансировки для протокола TCP

Создание и настройка Сервера Балансировки выполняется выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Серверы Балансировки**).

Для создания и настройки Сервера Балансировки используются команды:

Описание параметров также приведено в подразделе **Объект** `lbs`.

- создание Сервера Балансировки:

```
set lbs TCP <имя>
```

- назначение Группы Реальных Серверов для Сервера Балансировки:

В текущей версии Termidesk Connect предполагается, что один Сервер Балансировки работает с одной группой Реальных Серверов.

```
set lbs TCP <имя> rs-pool-id <имя>
```

- указание минимального количества активных Реальных Серверов, при котором Сервер Балансировки также будет активен (по умолчанию – `1`):

Если Сервер Балансировки становится неактивным (переходит в статус «Отключен»), то в текущей версии Termidesk Connect подключение пользователя к Реальному Серверу будет сброшено.

```
set lbs TCP <имя> min-rs <значение>
```

- задание алгоритма балансировки (по умолчанию – `LEASTCONN`):

Алгоритм балансировки может быть:

- `ROUNDROBIN` – подключения пользователей в этом случае распределяются последовательно между Реальными Серверами, что обеспечивает их равномерное распределение;
- `LEASTCONN` – подключения пользователей в этом случае распределяются оптимизировано, с учетом количества текущих активных соединений на каждом Реальном Сервере. Для подключения пользователя выбирается Реальный Сервер с наименьшим количеством текущих активных соединений, что обеспечивает более равномерное распределение нагрузки и помогает избежать перегрузки отдельных Реальных Серверов;
- `WEIGHTEDROUNDROBIN` – подключения пользователей в этом случае распределяются последовательно между Реальными Серверами пропорционально их Весу, что обеспечивает их равномерное распределение;
- `WEIGHTEDLEASTCONN` – подключения пользователей в этом случае распределяются оптимизировано, с учетом соотношения количества текущих активных соединений к Весу для каждого Реального Сервера, что помогает избежать перегрузки отдельных Реальных Серверов;
- `WEIGHTEDLEASTCONNECTTIME` – подключения пользователей в этом случае распределяются оптимизировано, с учетом соотношения наименьшего среднего времени соединения и количества текущих сессий к Весу для каждого Реального Сервера;
- `RANDOM` – для подключения пользователей в этом случае выбирается

случайный Реальный Сервер;

- `POWEROFTWORANDOM` – для подключения пользователей в этом случае выбирается Реальный Сервер с наименьшим числом соединений из двух Реальных Серверов, выбранных случайным образом.

```
set lbs TCP <имя> algorithm <алгоритм>
```

- при использовании алгоритмов `LEASTCONN`, `WEIGHTEDLEASTCONN`, `WEIGHTEDLEASTCONNECTTIME` указывается время (в секундах), на которое производится смена алгоритмов на `ROUNDROBIN` и `WEIGHTEDROUNDROBIN` соответственно при изменении количества серверов с состоянием «В работе» в привязанной Группе Реальных Серверов (по умолчанию – `0`):

Параметр `starttime` указывает, на какое время после изменения Группы Реальных Серверов Сервер Балансировки перейдет на работу по алгоритму `ROUNDROBIN` или `WEIGHTEDROUNDROBIN` для выравнивания количества подключений между Реальными Серверами.

Это позволяет исключить перегрузку Реального Сервера в случае, когда состав группы изменился (добавился новый узел), и все подключения пользователей были перенаправлены на новый Реальный Сервер.

```
set lbs TCP <имя> leastconn-param starttime <значение>
```

- (опционально) задание алгоритма привязки пользователя к Реальному Серверу на время активной сессии (по умолчанию – `NONE`):

Алгоритм привязки пользователя может быть:

- `NONE` – привязка не используется;
- `IPSOURCE` – привязка по IP-адресу источника запроса, т.е. IP-адресу пользователя;
- `SSLSESSION` – привязка пользователя к Реальному Серверу по идентификатору SSL-сессии, являющегося частью процесса установления соединения с выбранным сервером. Последующие запросы пользователя с данным идентификатором отправляются на ранее выбранный Реальный Сервер.

Настройка привязки пользователя используется, когда нужно сохранить соединение пользователя на определенном Реальном Сервере. Это актуально, например, в электронной коммерции – Интернет-магазинах и др., чтобы алгоритмы балансировки не перенаправили пользователя на другой Реальный Сервер.

```
set lbs TCP <имя> persistence algorithm <тип>
```

- (опционально, если задан алгоритм `IPSOURCE`) задание времени ожидания (в секундах), в течение которого повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер (по умолчанию – `60`):

```
set lbs TCP <имя> persistence ipsource-param timeout <значение>
```

- (опционально, если задан алгоритм `SSLSESSION`) задание времени ожидания (в секундах), в течение которого повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер (по умолчанию – `60`):

```
set lbs TCP <имя> persistence sslsession-param timeout <значение>
```

Пример: пользователь запросил доступ к приложению, а Termidesk Connect направил его на Реальный Сервер N. Пользователь получил приложение, поработал с ним и отключился, но позже запросил доступ снова.

Если задано время ожидания, то:

- в течение этого времени Termidesk Connect будет «помнить» о том, что пользователь ранее подключался к Реальному Серверу N;
- если время ожидания не истекло и пользователь повторно запросил доступ, то Termidesk Connect перенаправит подключение пользователя на Реальный Сервер N.

- (опционально) привязка Профиля сохранения сессий к Серверу Балансировки (см. подраздел **Профили**):

Все Реальные Серверы в группе должны иметь разные IP-адреса (это не касается портов) для успешного сохранения сессий.

```
set lbs TCP <имя> persistence-profile <имя_Профиля>
```

- (опционально) привязка Клиентского SSL-Профиля к Серверу Балансировки (см. подраздел **TLS**):

```
set lbs TCP <имя> ssl-profile-id <имя_Профиля>
```

- привязка Клиентского TCP-Профиля к Серверу Балансировки (см. подраздел **Профили**):

При добавлении Сервера Балансировки по умолчанию привязан Клиентский TCP-Профиль `tcpp-client-default`. Это Клиентский TCP-Профиль по умолчанию, который не может быть изменен или удален.

```
set lbs TCP <имя> tcp-profile-id <имя_Профиля>
```

- активация или отключение режима подмены IP-адреса клиента (по умолчанию – `false`):

Режим подмены IP-адреса клиента работает следующим образом:

- если режим активирован (`true`), то Termidesk Connect не будет подменять IP-адрес клиента. Следовательно, при взаимодействии с Реальным Серверов в источник запроса будет помещен IP-адрес клиента. При этом сетевой инфраструктурой должна обеспечиваться правильная обработка такого запроса: Реальный Сервер должен иметь возможность ответить на него;
- если режим отключен (`false`), то Termidesk Connect подменяет IP-адрес источника (клиента) и взаимодействует с Реальным сервером со своего IP-адреса. При этом в Группе Реальных Серверов обязательно должен быть задан IP-Фонд (см. подраздел **Группы Реальных Серверов**).

Если для сохранения клиентского IP-адреса необходимо использовать VRF, отличный от `default`, то настройка Группы Реальных Серверов осуществляется через IP-Фонд.

```
set lbs TCP <имя> use-cip <true/false>
```

- (опционально) настройка функции перебалансировки:
  - разрешение или запрет перебалансировки. Перебалансировка (попытка выбрать другой Реальный Сервер) осуществляется в случае ошибки подключения к Реальному Серверу (по умолчанию – `false`):

В случае выбора другого Реального Сервера в результате перебалансировки, если была запись в персистентной таблице, она будет удалена (заменена) на другой Реальный Сервер.

```
set lbs TCP <имя> re-balancing enable <true/false>
```

  - указание максимального количества попыток перебалансировки (значение от 1 до 10, по умолчанию – 1):

```
set lbs TCP <имя> re-balancing max-attempts <значение>
```

- (опционально) включение или отключение отслеживания состояния Сервера Балансировки для готовности узла к переходу в состояние `ACTIVE` (по умолчанию – `false`):

Подробное описание условий переключения узлов отказоустойчивой конфигурации приведено в подразделе **Отказоустойчивость**.

```
set lbs TCP <имя> ha-monitor <true/false>
```

- (опционально) привязка Профиля ограничения скорости (см. подраздел **Профили ограничения скорости**):

```
set lbs TCP <имя> rl-profile-id <имя_Профиля>
```

- (опционально) задание комментария, который будет привязан к Серверу

Балансировки:

Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.

```
set lbs TCP <имя> description <комментарий>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml lbs TCP <имя>
```

- просмотр выполненных команд:

```
show configuration cli lbs TCP <имя>
```

Создание и настройка Сервера Балансировки для протокола HTTP

Создание и настройка Сервера Балансировки выполняется выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Серверы Балансировки**).

Для создания и настройки Сервера Балансировки используются команды:

Описание параметров также приведено в подразделе **Объект `lbs`**.

- создание Сервера Балансировки:

```
set lbs HTTP <имя>
```

- назначение Группы Реальных Серверов для Сервера Балансировки:

В текущей версии Termidesk Connect предполагается, что один Сервер Балансировки работает с одной группой Реальных Серверов.

```
set lbs HTTP <имя> rs-pool-id <имя>
```

- указание минимального количества активных Реальных Серверов, при котором Сервер Балансировки также будет активен (по умолчанию – 1):

Если Сервер Балансировки становится неактивным (переходит в статус «Отключен»), то в текущей версии Termidesk Connect подключение пользователя к Реальному Серверу будет сброшено.

```
set lbs HTTP <имя> min-rs <значение>
```

- задание алгоритма балансировки (по умолчанию – LEASTCONN):

Алгоритм балансировки может быть:

- ROUNDROBIN – подключения пользователей в этом случае распределяются последовательно между Реальными Серверами, что обеспечивает их равномерное распределение;
- LEASTCONN – подключения пользователей в этом случае распределяются оптимизировано, с учетом количества текущих активных соединений на каждом Реальном Сервере. Для подключения пользователя выбирается Реальный Сервер с наименьшим количеством текущих активных соединений, что обеспечивает более равномерное распределение нагрузки и помогает избежать перегрузки отдельных Реальных Серверов;
- WEIGHTEDROUNDROBIN – подключения пользователей в этом случае распределяются последовательно между Реальными Серверами пропорционально их Весу, что обеспечивает их равномерное распределение;
- WEIGHTEDLEASTCONN – подключения пользователей в этом случае распределяются оптимизировано, с учетом соотношения количества текущих активных соединений к Весу для каждого Реального Сервера, что помогает избежать перегрузки отдельных Реальных Серверов;
- WEIGHTEDLEASTCONNECTTIME – подключения пользователей в этом случае распределяются оптимизировано, с учетом соотношения наименьшего среднего времени соединения и количества текущих сессий к Весу для каждого Реального Сервера;
- WEIGHTEDLEASTRESPONSETIME – подключения пользователей в этом случае распределяются оптимизировано, с учетом соотношения наименьшего среднего времени соединения, наименьшим средним временем получения первого байта ответа и количества текущих сессий к Весу для каждого Реального Сервера;
- RANDOM – для подключения пользователей в этом случае выбирается случайный Реальный Сервер;
- POWEROFTWORANDOM – для подключения пользователей в этом случае выбирается Реальный Сервер с наименьшим числом соединений из двух Реальных Серверов, выбранных случайным образом.

```
set lbs HTTP <имя> algorithm <алгоритм>
```

- при использовании алгоритмов LEASTCONN, WEIGHTEDLEASTCONN, WEIGHTEDLEASTCONNECTTIME, WEIGHTEDLEASTRESPONSETIME указывается время (в секундах), на которое производится смена алгоритмов на ROUNDROBIN и WEIGHTEDROUNDROBIN соответственно при изменении количества серверов с состоянием «В работе» в привязанной Группе Реальных Серверов (по умолчанию – 0):

Параметр starttime указывает, на какое время после изменения Группы Реальных Серверов Сервер Балансировки перейдет на работу по алгоритму ROUNDROBIN или WEIGHTEDROUNDROBIN для выравнивания количества подключений между Реальными Серверами.

Это позволяет исключить перегрузку Реального Сервера в случае, когда состав группы изменился (добавился новый узел), и все подключения пользователей были перенаправлены на новый Реальный Сервер.

```
set lbs HTTP <имя> leastconn-param starttime <значение>
```

- (опционально) задание алгоритма привязки пользователя к Реальному Серверу на время активной сессии (по умолчанию – NONE):

Алгоритм привязки пользователя может быть:

- NONE – привязка не используется;
- IPSOURCE – привязка по IP-адресу источника запроса, т.е. IP-адресу пользователя;
- COOKIEINSERT – привязка по cookie, который помещается в HTTP-ответ, направляемый пользователю. Обеспечивает постоянство выбора Реального Сервера путем автоматической вставки cookie в HTTP-ответ. Последующие запросы пользователя с этим cookie перенаправляются на тот же Реальный Сервер. В случае, когда пользователь не сохраняет cookie в HTTP, его запросы не будут содержать cookie для отправки Termidesk Connect. Для такого пользователя данный метод привязки не подходит, требуется настройка альтернативного метода;
- HEADER – привязка по значению заголовка, указанного в конфигурации. Этот алгоритм независим от TCP/IP параметров подключения;
- COOKIE – привязка по cookie, который получен в ответе Реального Сервера. Значение cookie, указанного в конфигурации, записывается в персистентную таблицу. Последующие запросы пользователя с данным cookie отправляются на этот Реальный Сервер;
- SSLSESSION – привязка пользователя к Реальному Серверу по идентификатору SSL-сессии, являющегося частью процесса установления соединения с выбранным сервером. Последующие запросы пользователя с данным идентификатором отправляются на ранее выбранный Реальный Сервер.

Настройка привязки пользователя используется, когда нужно сохранить соединение пользователя на определенном Реальном Сервере. Это актуально, например, в электронной коммерции – Интернет-магазинах и др., чтобы алгоритмы балансировки не перенаправили пользователя на другой Реальный Сервер.

COOKIEINSERT – это не табличный метод привязки, cookie не хранится в Termidesk Connect. Решение о выборе Реального Сервера принимается именно на основании данных, закодированных в cookie с информацией о

Виртуальном Сервере, IP-адресе и порте Реального Сервера.

```
set lbs HTTP <имя> persistence algorithm <тип>
```

Пример: пользователь запросил доступ к приложению, а Termidesk Connect направил его на Реальный Сервер N. Пользователь получил приложение, поработал с ним и отключился, но позже запросил доступ снова.

Если задано время ожидания, то:

- в течение этого времени Termidesk Connect будет «помнить» о том, что пользователь ранее подключался к Реальному Серверу N;
- если время ожидания не истекло и пользователь повторно запросил доступ, то Termidesk Connect перенаправит подключение пользователя на Реальный Сервер N.

- (опционально, если задан алгоритм IPSOURCE) задание времени ожидания (в секундах), в течение которого повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер (по умолчанию – 60):

```
set lbs HTTP <имя> persistence ipsource-param timeout <значение>
```

- (опционально, если задан алгоритм COOKIEINSERT) задание времени ожидания (в секундах), в течение которого повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер (по умолчанию – 60):

```
set lbs HTTP <имя> persistence ci-param timeout <значение>
```

- (опционально, если задан алгоритм HEADER) задание заголовка, по которому повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер:

```
set lbs HTTP <имя> persistence header-param header-name <значение>
```

- (опционально, если задан алгоритм HEADER) задание времени ожидания (в секундах), в течение которого повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер (по умолчанию – 60):

```
set lbs HTTP <имя> persistence header-param timeout <значение>
```

- (опционально, если задан алгоритм COOKIE) задание имени cookie, который ожидается в ответе Реального Сервера для повторного подключения пользователя на этот Реальный Сервер:

```
set lbs HTTP <имя> persistence cookie-param cookie-name <значение>
```

- (опционально, если задан алгоритм `COOKIE`) задание времени ожидания (в секундах), в течение которого повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер (по умолчанию – `60`):

```
set lbs HTTP <имя> persistence cookie-param timeout <значение>
```

- (опционально, если задан алгоритм `SSLSESSION`) задание времени ожидания (в секундах), в течение которого повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер (по умолчанию – `60`):

```
set lbs HTTP <имя> persistence sslsession-param timeout <значение>
```

- (опционально) привязка Профиля сохранения сессий к Серверу Балансировки (см. подраздел **Профили**):

Все Реальные Серверы в группе должны иметь разные IP-адреса (это не касается портов) для успешного сохранения сессий.

```
set lbs HTTP <имя> persistence-profile <имя_Профиля>
```

- (опционально) привязка Клиентского SSL-Профиля к Серверу Балансировки (см. подраздел **TLS**):

```
set lbs HTTP <имя> ssl-profile-id <имя_Профиля>
```

- привязка Клиентского HTTP-Профиля к Серверу Балансировки (см. подраздел **Профили**):

При добавлении Сервера Балансировки по умолчанию привязан Клиентский HTTP-Профиль `httpp-client-default`. Это преднастроенный Клиентский HTTP-Профиль, который не может быть изменен или удален.

```
set lbs HTTP <имя> http-profile-id <имя_Профиля>
```

- привязка Клиентского TCP-Профиля к Серверу Балансировки (см. подраздел **Профили**):

При добавлении Сервера Балансировки по умолчанию привязан Клиентский TCP-Профиль `tcpp-client-default`. Это преднастроенный Клиентский TCP-Профиль, который не может быть изменен или удален.

```
set lbs HTTP <имя> tcp-profile-id <имя_Профиля>
```

- активация или отключение режима подмены IP-адреса клиента (по умолчанию – `false`):

Режим подмены IP-адреса клиента работает следующим образом:

- если режим активирован (`true`), то Termidesk Connect не будет подменять IP-адрес клиента. Следовательно, при взаимодействии с Реальным Серверов в источник запроса будет помещен IP-адрес клиента. При этом сетевой инфраструктурой должна обеспечиваться правильная обработка такого запроса: Реальный Сервер должен иметь возможность ответить на него;
- если режим отключен (`false`), то Termidesk Connect подменяет IP-адрес источника (клиента) и взаимодействует с Реальным сервером со своего IP-адреса. При этом в Группе Реальных Серверов обязательно должен быть задан IP-Фонд (см. подраздел **Группы Реальных Серверов**).

Если для сохранения клиентского IP-адреса необходимо использовать VRF, отличный от `default`, то настройка Группы Реальных Серверов осуществляется через IP-Фонд.

```
set lbs HTTP <имя> use-cip <true/false>
```

- (опционально) задание Сценария (исполняемого файла формата `LUA`):

Параметр `<приоритет>` задает приоритет применения Сценария: чем ниже число, тем выше приоритет, и тем раньше Сценарий будет обработан. Подробное описание работы со Сценариями приведено в подразделе **Сценарии**.

Файл должен быть расположен в каталоге `/var/lib/tdc/lbscripts/response-modifying/`.

```
set lbs HTTP <имя> luarules <приоритет> script <имя_файла>
```

- (опционально) настройка функции перебалансировки:
  - разрешение или запрет перебалансировки. Перебалансировка (попытка выбрать другой Реальный Сервер) осуществляется в случае ошибки подключения к Реальному Серверу (по умолчанию – `false`):

В случае выбора другого Реального Сервера в результате перебалансировки, если была запись в персистентной таблице, эта запись будет удалена (заменена) на другой Реальный Сервер.

```
set lbs HTTP <имя> re-balancing enable <true/false>
```

  - указание максимального количества попыток перебалансировки (значение от 1 до 10, по умолчанию – `1`):

```
set lbs HTTP <имя> re-balancing max-attempts <значение>
```

- (опционально) включение или отключение отслеживания состояния Сервера Балансировки для готовности узла к переходу в состояние `ACTIVE` (по умолчанию – `false`):

Описание параметров также приведено в подразделе **Объект** `lbs`.

Подробное описание условий переключения узлов отказоустойчивой конфигурации приведено в подразделе **Отказоустойчивость**.

```
set lbs HTTP <имя> ha-monitor <true/false>
```

- (опционально) привязка Профиля ограничения скорости (см. подраздел **Профили ограничения скорости**):

```
set lbs HTTP <имя> rl-profile-id <имя_Профиля>
```

- (опционально) задание комментария, который будет привязан к Серверу Балансировки:

Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.

```
set lbs HTTP <имя> description <комментарий>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml lbs HTTP <имя>
```

- просмотр выполненных команд:

```
show configuration cli lbs HTTP <имя>
```

Создание и настройка Сервера Балансировки для протокола RAPID-TCP

Создание и настройка Сервера Балансировки выполняется выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Серверы Балансировки**).

Для создания и настройки Сервера Балансировки используются команды:

Описание параметров также приведено в подразделе **Объект lbs**.

Часть настроек выполняется аналогично Серверу Балансировки для протокола TCP.

- создание Сервера Балансировки:

```
set lbs RAPID-TCP <имя>
```

- назначение Группы Реальных Серверов для Сервера Балансировки:

В текущей версии Termidesk Connect предполагается, что один Сервер Балансировки работает с одной группой Реальных Серверов.

```
set lbs RAPID-TCP <имя> rs-pool-id <имя>
```

- указание минимального количества активных Реальных Серверов, при котором Сервер Балансировки также будет активен (по умолчанию – 1):

```
set lbs RAPID-TCP <имя> min-rs <значение>
```

- задание алгоритма балансировки (по умолчанию – LEASTCONN):

Алгоритм балансировки может быть:

- ROUNDROBIN – подключения пользователей в этом случае распределяются последовательно между Реальными Серверами, что обеспечивает их равномерное распределение;
- LEASTCONN – подключения пользователей в этом случае распределяются оптимизировано, с учетом количества текущих активных соединений на каждом Реальном Сервере. Для подключения пользователя выбирается Реальный Сервер с наименьшим количеством текущих активных соединений, что обеспечивает более равномерное распределение нагрузки и помогает избежать перегрузки отдельных Реальных Серверов;
- WEIGHTEDROUNDROBIN – подключения пользователей в этом случае распределяются последовательно между Реальными Серверами пропорционально их Весу, что обеспечивает их равномерное распределение;
- WEIGHTEDLEASTCONN – подключения пользователей в этом случае распределяются оптимизировано, с учетом соотношения количества текущих активных соединений к Весу для каждого Реального Сервера, что помогает избежать перегрузки отдельных Реальных Серверов;
- WEIGHTEDLEASTCONNECTTIME – подключения пользователей в этом случае распределяются оптимизировано, с учетом соотношения наименьшего среднего времени соединения и количества текущих сессий к Весу для каждого Реального Сервера;
- RANDOM – для подключения пользователей в этом случае выбирается

случайный Реальный Сервер;

- POWEROFTWORANDOM – для подключения пользователей в этом случае выбирается Реальный Сервер с наименьшим числом соединений из двух Реальных Серверов, выбранных случайным образом.

```
set lbs RAPID-TCP <имя> algorithm <тип>
```

- при использовании алгоритмов LEASTCONN, WEIGHTEDLEASTCONN, WEIGHTEDLEASTCONNECTTIME, указывается время (в секундах), на которое производится смена алгоритмов на ROUNDROBIN и WEIGHTEDROUNDROBIN соответственно при изменении количества серверов с состоянием «В работе» в привязанной Группе Реальных Серверов (по умолчанию – 0):

Параметр starttime указывает, на какое время после изменения Группы Реальных Серверов Сервер Балансировки перейдет на работу по алгоритму ROUNDROBIN или WEIGHTEDROUNDROBIN для выравнивания количества подключений между Реальными Серверами.

Это позволяет исключить перегрузку Реального Сервера в случае, когда состав группы изменился (добавился новый узел), и все подключения пользователей были перенаправлены на новый Реальный Сервер.

```
set lbs RAPID-TCP <имя> leastconn-param starttime <значение>
```

- (опционально) задание алгоритма привязки пользователя к Реальному Серверу на время активной сессии (по умолчанию – NONE):

Алгоритм привязки пользователя может быть:

- NONE – привязка не используется;
- IPSOURCE – привязка по IP-адресу источника запроса, т.е. IP-адресу пользователя.

Настройка привязки пользователя используется, когда нужно сохранить соединение пользователя на определенном Реальном Сервере. Это актуально, например, в электронной коммерции – Интернет-магазинах и др., чтобы алгоритмы балансировки не перенаправили пользователя на другой Реальный Сервер.

```
set lbs RAPID-TCP <имя> persistence algorithm <тип>
```

- (опционально, если задан алгоритм IPSOURCE) задание времени ожидания (в секундах), в течение которого повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер (по умолчанию – 60):

```
set lbs RAPID-TCP <имя> persistence ipsource-param timeout <значение>
```

- активация или отключение режима сохранения IP-адреса из IP-Фонда для взаимодействия с Реальным Сервером (по умолчанию – false):

Режим сохранения IP-адреса из IP-Фонда работает следующим образом:

- если режим активирован (`true`), то Termidesk Connect будет сохранять ранее выбранный IP-адрес из пула адресов IP-Фонда для взаимодействия с Реальным Сервером;
- если режим отключен (`false`), то Termidesk Connect будет выбирать случайный IP-адрес из пула адресов IP-Фонда для взаимодействия с Реальным Сервером.

```
set lbs RAPID-TCP <имя> persistence ipsource-param ipset-src-persist <true/false>
```

- задание времени жизни сессии (в секундах) после получения управляющего флага FIN в заголовке пакета (по умолчанию – `2`):

```
set lbs RAPID-TCP <имя> fin-timeout <значение>
```

- задание времени жизни сессии (в секундах) при бездействии, т.е. отсутствии пакетов (по умолчанию – `60`):

```
set lbs RAPID-TCP <имя> connection-idle <значение>
```

- активация или отключение режима подмены IP-адреса клиента (по умолчанию – `false`):

Режим подмены IP-адреса клиента работает следующим образом:

- если режим активирован (`true`), то Termidesk Connect не будет подменять IP-адрес клиента. Следовательно, при взаимодействии с Реальным Серверов в источник запроса будет помещен IP-адрес клиента. При этом сетевой инфраструктурой должна обеспечиваться правильная обработка такого запроса: Реальный Сервер должен иметь возможность ответить на него;
- если режим отключен (`false`), то Termidesk Connect подменяет IP-адрес источника (клиента) и взаимодействует с Реальным сервером со своего IP-адреса. При этом в Группе Реальных Серверов обязательно должен быть задан IP-Фонд (см. подраздел **Группы Реальных Серверов**).

Если для сохранения клиентского IP-адреса необходимо использовать VRF, отличный от `default`, то настройка Группы Реальных Серверов осуществляется через IP-Фонд.

```
set lbs RAPID-TCP <имя> use-cip <true/false>
```

- (опционально) задание режима работы Сервера Балансировки (по умолчанию – `OFF`):

Режим работы Сервера Балансировки может быть:

- `OFF` – режим работы, при котором DSR отключен;
- `MAC` – режим с подменой MAC-адресов (L2 DSR). В этом режиме IP-адреса во входящем пакете остаются неизменными, Termidesk Connect

подменяет в пакете только MAC-адреса (MAC-адрес источника – Termidesk Connect, MAC-адрес назначения – Реальный Сервер) и отправляет этот пакет по MAC-адресам на сервер;

- `IPIP` – режим (L3 DSR), при котором входящий пакет инкапсулируется в IPIP-туннель и направляется на Реальный Сервер. Далее Реальный Сервер декапсулирует IP-пакет и видит IP-адрес пользователя и IP-адрес Виртуального Сервера.

Описание режимов DSR представлено в подразделе **Режим DSR**.

```
set lbs RAPID-TCP <имя> dsr-mode <режим_работы>
```

- задание параметров TTL IP-пакета в заголовке:

TTL – поле в заголовке IP-пакета, указывающее максимальное число переходов, через которое можем пройти пакет переж тем, как будет отброшен.

По умолчанию Termidesk Connect не меняет TTL, но опционально можно изменить его для пакетов, пережаваемых в сторону Реального Сервера или в сторону источника запроса (клиента).

  - TTL пакета к Реальному Серверу (по умолчанию – `0`):

```
set lbs RAPID-TCP <имя> ttl to-rs <значение>
```

  - TTL пакета к источнику запроса (по умолчанию – `0`):

```
set lbs RAPID-TCP <имя> ttl to-client <значение>
```

- (опционально) включение или отключение отслеживания состояния Сервера Балансировки для готовности узла к переходу в состояние `ACTIVE` (по умолчанию – `false`):

Подробное описание условий переключения узлов отказоустойчивой конфигурации приведено в подразделе **Отказоустойчивость**.

```
set lbs RAPID-TCP <имя> ha-monitor <true/false>
```

- (опционально) задание комментария, который будет привязан к Серверу Балансировки:

Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.

```
set lbs RAPID-TCP <имя> description <комментарий>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml lbs RAPID-TCP <имя>
```

- просмотр выполненных команд:

```
show configuration cli lbs RAPID-TCP <имя>
```

Создание и настройка Сервера Балансировки для протокола RAPID-UDP

Создание и настройка Сервера Балансировки выполняется выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Серверы Балансировки**).

Для создания и настройки Сервера Балансировки используются команды:

Описание параметров также приведено в подразделе **Объект `lbs`**.

Часть настроек выполняется аналогично Серверу Балансировки для протокола TCP.

- создание Сервера Балансировки:

```
set lbs RAPID-UDP <имя>
```

- назначение Группы Реальных Серверов для Сервера Балансировки:

В текущей версии Termidesk Connect предполагается, что один Сервер Балансировки работает с одной группой Реальных Серверов.

```
set lbs RAPID-UDP <имя> rs-pool-id <имя>
```

- указание минимального количества активных Реальных Серверов, при котором Сервер Балансировки также будет активен (по умолчанию – `1`):

```
set lbs RAPID-UDP <имя> min-rs <значение>
```

- задание алгоритма балансировки (по умолчанию – LEASTCONN):

Алгоритм балансировки может быть:

- ROUNDROBIN – подключения пользователей в этом случае распределяются последовательно между Реальными Серверами, что обеспечивает их равномерное распределение;
- LEASTCONN – подключения пользователей в этом случае распределяются оптимизировано, с учетом количества текущих активных соединений на каждом Реальном Сервере. Для подключения пользователя выбирается Реальный Сервер с наименьшим количеством текущих активных соединений, что обеспечивает более равномерное распределение нагрузки и помогает избежать перегрузки отдельных Реальных Серверов;
- WEIGHTEDROUNDROBIN – подключения пользователей в этом случае распределяются последовательно между Реальными Серверами пропорционально их Весу, что обеспечивает их равномерное распределение;
- WEIGHTEDLEASTCONN – подключения пользователей в этом случае распределяются оптимизировано, с учетом соотношения количества текущих активных соединений к Весу для каждого Реального Сервера, что помогает избежать перегрузки отдельных Реальных Серверов;
- RANDOM – для подключения пользователей в этом случае выбирается случайный Реальный Сервер;
- POWEROFTWORANDOM – для подключения пользователей в этом случае выбирается Реальный Сервер с наименьшим числом соединений из двух Реальных Серверов, выбранных случайным образом.

```
set lbs RAPID-UDP <имя> algorithm <тип>
```

- при использовании алгоритмов LEASTCONN, WEIGHTEDLEASTCONN указывается время (в секундах), на которое производится смена алгоритмов на ROUNDROBIN и WEIGHTEDROUNDROBIN соответственно при изменении количества серверов с состоянием «В работе» в привязанной Группе Реальных Серверов (по умолчанию – 0):

Параметр starttime указывает, на какое время после изменения Группы Реальных Серверов Сервер Балансировки перейдет на работу по алгоритму ROUNDROBIN или WEIGHTEDROUNDROBIN для выравнивания количества подключений между Реальными Серверами.

Это позволяет исключить перегрузку Реального Сервера в случае, когда состав группы изменился (добавился новый узел), и все подключения пользователей были перенаправлены на новый Реальный Сервер.

```
set lbs RAPID-UDP <имя> leastconn-param starttime <значение>
```

- (опционально) задание алгоритма привязки пользователя к Реальному Серверу на время активной сессии (по умолчанию – NONE):

Алгоритм привязки пользователя может быть:

- NONE – привязка не используется;
- IPSOURCE – привязка по IP-адресу источника запроса, т.е. IP-адресу пользователя.

Настройка привязки пользователя используется, когда нужно сохранить соединение пользователя на определенном Реальном Сервере. Это актуально, например, в электронной коммерции – Интернет-магазинах и др., чтобы алгоритмы балансировки не перенаправили пользователя на другой Реальный Сервер.

```
set lbs RAPID-UDP <имя> persistence algorithm <тип>
```

- (опционально, если задан алгоритм IPSOURCE) задание времени ожидания (в секундах), в течение которого повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер (по умолчанию – 60):

```
set lbs RAPID-UDP <имя> persistence ipsource-param timeout <значение>
```

- активация или отключение режима сохранения IP-адреса из IP-Фонда для взаимодействия с Реальным Сервером (по умолчанию – false):

Режим сохранения IP-адреса из IP-Фонда работает следующим образом:

- если режим активирован (true), то Termidesk Connect будет сохранять ранее выбранный IP-адрес из пула адресов IP-Фонда для взаимодействия с Реальным Сервером;
- если режим отключен (false), то Termidesk Connect будет выбирать случайный IP-адрес из пула адресов IP-Фонда для взаимодействия с Реальным Сервером.

```
set lbs RAPID-UDP <имя> persistence ipsource-param ipset-src-persist <true/false>
```

- задание времени жизни сессии (в секундах) при бездействии, т.е. отсутствии пакетов (по умолчанию – 60):

```
set lbs RAPID-UDP <имя> connection-idle <значение>
```

- активация или отключение режима подмены IP-адреса клиента (по умолчанию – false):

Режим подмены IP-адреса клиента работает следующим образом:

- если режим активирован (true), то Termidesk Connect не будет подменять IP-адрес клиента. Следовательно, при взаимодействии с Реальным Серверов в источник запроса будет помещен IP-адрес клиента. При этом сетевой инфраструктурой должна обеспечиваться правильная обработка такого запроса: Реальный Сервер должен иметь возможность ответить на него;

- если режим отключен (`false`), то Termidesk Connect подменяет IP-адрес источника (клиента) и взаимодействует с Реальным сервером со своего IP-адреса. При этом в Группе Реальных Серверов обязательно должен быть задан IP-Фонд (см. подраздел **Группы Реальных Серверов**).

Если для сохранения клиентского IP-адреса необходимо использовать VRF, отличный от `default`, то настройка Группы Реальных Серверов осуществляется через IP-Фонд.

```
set lbs RAPID-UDP <имя> use-cip <true/false>
```

- (опционально) задание режима работы Сервера Балансировки (по умолчанию – `OFF`):

Режим работы Сервера Балансировки может быть:

- `OFF` – режим работы, при котором DSR отключен;
- `MAC` – режим с подменой MAC-адресов (L2 DSR). В этом режиме IP-адреса во входящем пакете остаются неизменными, Termidesk Connect подменяет в пакете только MAC-адреса (MAC-адрес источника – Termidesk Connect, MAC-адрес назначения – Реальный Сервер) и отправляет этот пакет по MAC-адресам на сервер;
- `IPIP` – режим (L3 DSR), при котором входящий пакет инкапсулируется в IPIP-туннель и направляется на Реальный Сервер. Далее Реальный Сервер декапсулирует IP-пакет и видит IP-адрес пользователя и IP-адрес Виртуального Сервера.

Описание режимов DSR представлено в подразделе **Режим DSR**.

```
set lbs RAPID-UDP <имя> dsr-mode <режим_работы>
```

- задание параметров TTL IP-пакета в заголовке:

TTL – поле в заголовке IP-пакета, указывающее максимальное число переходов, через которое можем пройти пакет переж тем, как будет отброшен.

По умолчанию Termidesk Connect не меняет TTL, но опционально можно изменить его для пакетов, пережаваемых в сторону Реального Сервера или в сторону источника запроса (клиента).

  - TTL пакета к Реальному Серверу (по умолчанию – `0`):

```
set lbs RAPID-UDP <имя> ttl to-rs <значение>
```

  - TTL пакета к источнику запроса (по умолчанию – `0`):

```
set lbs RAPID-UDP <имя> ttl to-client <значение>
```

- (опционально) включение или отключение отслеживания состояния Сервера Балансировки для готовности узла к переходу в состояние `ACTIVE` (по умолчанию –

`false`):

Подробное описание условий переключения узлов отказоустойчивой конфигурации приведено в подразделе **Отказоустойчивость**.

```
set lbs RAPID-UDP <имя> ha-monitor <true/false>
```

- (опционально) задание комментария, который будет привязан к Серверу Балансировки:

Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.

```
set lbs RAPID-UDP <имя> description <комментарий>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml lbs RAPID-UDP <имя>
```

- просмотр выполненных команд:

```
show configuration cli lbs RAPID-UDP <имя>
```

TLS

Общие сведения о TLS

TLS используется в Termidesk Connect для:

- настройки доступа к веб-интерфейсу по протоколу HTTPS;
- настройки SSL-соединения;
- настройки функционала SSL Offload;
- настройки взаимной аутентификации по протоколу mTLS.

Подразумевается, что SSL и TLS – два неразрывно связанных

протокола.Поэтому здесь и далее они равнозначны: например, SSL Offload и TLS Offload представляют один и тот же процесс, но далее будет использовано только наименование SSL Offload.

SSL Offload – это процесс, в котором преобразование трафика SSL/TLS выполняется не на серверах приложений (Реальных Серверах), а на Termidesk Connect, что также позволяет снизить нагрузку с Реальных Серверов. Процесс выглядит следующим образом:

- Termidesk Connect терминирует на себе сессию пользователя;
- в зависимости от настроек Termidesk Connect:
  - либо преобразует полученные данные к открытому виду для последующего их анализа и обработки;
  - либо не преобразует полученные данные, поскольку они уже пришли в открытом виде;
- далее, в зависимости от настроек, Termidesk Connect:
  - либо отправляет на Реальный Сервер данные без преобразования;
  - либо преобразует данные, а потом отправляет их на Реальный Сервер.

Для реализации этого функционала создаются SSL-Профили.

Файлы для SSL/TLS

Загрузка сертификатов, ключей, CRL и других файлов выполняется с помощью протокола SFTP или через веб-интерфейс (см. подраздел **Веб. TLS**).

После загрузки файлы будут расположены в каталоге `/etc/ssl/tdc/`.

Профили SSL/TLS

Профили определяют настройки и параметры, используемые для преобразования данных. В контексте управления трафиком важно понимать различия между Серверными и Клиентскими Профилями:

- Серверный SSL-Профиль настраивается для обеспечения безопасного соединения между пользователем и Termidesk Connect;
- Клиентский SSL-Профиль настраивается для обеспечения безопасного соединения между Termidesk Connect и Реальным Сервером.

Создание и настройка Серверного SSL-Профиля

Создание и настройка Серверного SSL-Профиля выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. TLS**).

Команды настройки Серверного SSL-Профиля включают в себя аналогичные параметры, что и Клиентский SSL-Профиль, с дополнительной настройкой идентификации имени сервера и управления режимом взаимной аутентификации (mTLS).

Пример команд настройки Серверного SSL-Профиля:

Описание параметров также приведено в подразделе **Объект** `ssl-profile`.

- создание SSL-Профиля:

```
set ssl-profile server <имя_Профиля>
```

- указание времени ожидания (в секундах) ответа от пользователя (значение от 1 до 60, по умолчанию – 5):

```
set ssl-profile server <имя_Профиля> handshake-timeout <значение>
```

- (опционально) указание значения поля SNI из TLS Hello, для которого требуются особые настройки обработки:

SNI – расширение протокола TLS, позволяющее пользователю сообщать имя узла, с которым он хочет установить соединение во время процесса «рукопожатия».

Это позволяет серверу (в Termidesk Connect – Виртуальному Серверу) предоставлять несколько сертификатов на одном IP-адресе и TCP-порту, и, следовательно, позволяет работать нескольким HTTPS-сайтам или другим сервисам поверх TLS на одном IP-адресе без использования одного и того же сертификата на всех сайтах. Имя узла передается в TLS Hello пользователя.

Если требуется задать правила для нескольких SNI, то для каждого из них должна использоваться индивидуальная настройка.

Поддерживается задание шаблонов SNI по формату: `*.<домен>`. При этом:

- астериск может быть расположен только слева и должен быть разделен от домена точкой;
- астериск означает, что любой хост домена (не включая оригинальный домен) удовлетворяет шаблону. Для профиля оригинального домена требуется отдельный хост SNI;
- при выборе SNI приоритетным будет тот, у которого совпадет больше уровней доменов или имеется полное совпадение по шаблону.

```
set ssl-profile server <имя_Профиля> host <имя>
```

- (опционально) указание файла сертификата УЦ, который используются для проверки подлинности клиентского сертификата пользователя (используется, если активирована взаимная аутентификация по протоколу mTLS):

Указывается полное имя файла с расширением.

Для одного правила SNI может быть задано несколько файлов сертификатов УЦ. Команда позволяет за один раз добавить только один файл.

Так же файлы могут быть получены из Хранилища секретов. Пример значения параметра: `pki://pki/issue/example-dot-com?common_name=\"app.example.com\"&ttl=\"24h\"#issuing_ca`.

```
set ssl-profile server <имя_Профиля> host <имя> setting ca-certs <имя_файла>
```

- указание файла сертификата сервера, который будет использоваться для аутентификации Termidesk Connect:

Указывается полное имя файла с расширением. Пример значения параметра: `server.crt`.

Так же файл может быть получен из Хранилища секретов. Пример значения параметра: `pki://pki/issue/example-dot-com?common_name=\"app.example.com\"&ttl=\"24h\"#certificate`.

```
set ssl-profile server <имя_Профиля> host <имя> setting cert <имя_файла>
```

- (опционально) указание файла параметров Диффи-Хеллмана, использующегося для безопасного обмена ключами при установлении SSL-соединения:

Указывается полное имя файла с расширением.

```
set ssl-profile server <имя_Профиля> host <имя> setting dh-params <имя_файла>
```

- указание файла закрытого ключа, соответствующего серверному сертификату:

Указывается полное имя файла с расширением. Пример значения параметра: `crt.key`.

Так же файл может быть получен из Хранилища секретов. Пример значения параметра: `pki://pki/issue/example-dot-com?common_name=\"app.example.com\"&ttl=\"24h\"#private_key`.

```
set ssl-profile server <имя_Профиля> host <имя> setting key <имя_файла>
```

- (опционально) если ключ зашифрован, то указать пароль для расшифровки закрытого ключа:

Так же пароль может быть получен из Хранилища секретов. Пример значения параметра:

- `kv://secret/cert#password` – для `kv` версии 1;
- `kv://secret/data/snmp#password` – для `kv` версии 2.

```
set ssl-profile server <имя_Профиля> host <имя> setting password <пароль>
```

- (опционально) активация поддержки взаимной аутентификации по протоколу mTLS:

```
set ssl-profile server <имя_Профиля> host <имя> setting mtls true
```

- если активирован протокол mTLS, указать метод проверки отзыва для клиентских сертификатов (по умолчанию – NONE):

Значение может быть:

- NONE – отсутствует проверка на отзыв клиентского сертификата;
- CRL – проверка методом CRL, когда отозванные сертификаты перечислены в CRL-файле. Если клиент предоставляет сертификат из этого списка при создании подключения, сертификат считается отозванным и подключение сбрасывается;
- OCSP – проверка методом OCSP, когда на сервер (OCSP Responder) отправляется OCSP-запрос для получения актуального статуса сертификата.

```
set ssl-profile server <имя_Профиля> host <имя> setting mtls-check
<метод_проверки_mTLS>
```

- если выбран метод CRL, настроить параметры проверки клиентских сертификатов методом CRL:
  - указание CRL-файла для проверки клиентских сертификатов:

Указывается полное имя файла с расширением. Пример значения параметра: crl.pem.

Так же файл может быть получен из Хранилища секретов. Пример значения параметра: pki://v1/pki/cert/crl#certificate.

```
set ssl-profile server <имя_Профиля> host <имя> setting crl-param file
<имя_файла>
```

- если выбран метод OCSP, настроить параметры проверки клиентских сертификатов методом OCSP:
  - указание URL-адреса сервера (OCSP Responder):

В текущей версии Termidesk Connect для сервера (OCSP Responder) поддерживается только протокол HTTP. Пример: http://myocspserver.local/.

```
set ssl-profile server <имя_Профиля> host <имя> setting ocsp-param url <URL>
```

  - указание метода HTTP-запроса:

В текущей версии Termidesk Connect поддерживается только метод POST.

```
set ssl-profile server <имя_Профиля> host <имя> setting ocsp-param method
POST
```

- указание времени ожидания (в секундах) ответа от сервера (OCSP Responder) (по умолчанию – `10`)

```
set ssl-profile server <имя_Профиля> host <имя> setting ocsp-param timeout
<значение>
```

- (опционально) активация кеширования OCSP-ответов (по умолчанию – `true`):

```
set ssl-profile server <имя_Профиля> host <имя> setting ocsp-param cache true
```

- (опционально) активация поддержки расширения Nonce (по умолчанию – `true`):

```
set ssl-profile server <имя_Профиля> host <имя> setting ocsp-param nonce true
```

- (опционально) активация строгого режима проверки (по умолчанию – `true`):

Значение может быть:

- `true` – допуск только клиентов, чьи сертификаты разрешены;
- `false` – допуск клиентов, чьи сертификаты разрешены или чьих сертификатов нет в БД.

```
set ssl-profile server <имя_Профиля> host <имя> setting ocsp-param strict
true
```

- (опционально) активация поддержки OCSP Stapling с фоновым обновлением подписи серверного сертификата через OCSP Responder (по умолчанию – `false`):

Для OCSP Stapling не задается URL-адрес, т.к он извлекается из серверного сертификата.

В файле серверного сертификата необходимо указывать всю цепочку сертификатов, ведущую к УЦ, включая все промежуточные сертификаты.

```
set ssl-profile server <имя_Профиля> host <имя> setting stapling true
```

- определение набора алгоритмов, которые могут использоваться для преобразования данных между пользователем и Termidesk Connect:

Команда позволяет за один раз добавить только один алгоритм.

После создания SSL-Профиля по умолчанию уже назначены следующие

алгоритмы: TLS_AES_256_GCM_SHA384, TLS_CHACHA20_POLY1305_SHA25, TLS_AES_128_GCM_SHA256, ECDHE-ECDSA-AES256-GCM-SHA384, ECDHE-ECDSA-AES128-GCM-SHA256, ECDHE-ECDSA-AES256-SHA384, ECDHE-RSA-AES256-SHA384, ECDHE-ECDSA-AES128-SHA256, ECDHE-RSA-AES128-SHA256, ECDHE-ECDSA-AES256-SHA, ECDHE-RSA-AES256-SHA, ECDHE-ECDSA-AES128-SHA, ECDHE-RSA-AES128-SHA.

```
set ssl-profile server <имя_Профиля> host <имя> setting ciphers <алгоритм> true
```

- указание версии протокола TLS для данного Серверного SSL-Профиля:

```
set ssl-profile server <имя_Профиля> host <имя> setting versions <версия TLS> true
```

- использование механизма Secure Renegotiation (по умолчанию – `ENABLE`):

Механизм Secure Renegotiation применяется для протокола TLS версии 1.2 или старше и позволяет безопасно пересогласовать параметры SSL-соединения без разрыва текущего защищенного канала. Secure Renegotiation требуется в сценариях, при которых необходима периодическая смена ключей.

При активированном Secure Renegotiation:

- каждое новое повторное согласование соединения связывается с предшествующими сообщениями через специальное поле `renegotiation_info`, содержащее контрольную информацию о них;
- корректность значения поля `renegotiation_info` проверяется и если все проверки прошли успешно, то соединение продолжается. Если какая-либо из сторон соединения обнаруживает несоответствие в этом поле, то соединение обрывается.

```
set ssl-profile server <имя> host <имя> setting ssl-reneg state ENABLE
```

- задание лимита запросов в минуту на SSL-соединение (по умолчанию – `10`):

При превышении заданного лимита для всех последующих попыток будет возвращено предупреждение о невозможности согласования на уровне протокола, пока период не обновится. При значении `0` ограничение на количество допустимых запросов отсутствует.

```
set ssl-profile server <имя> host <имя> setting ssl-reneg rate <значение>
```

- использование механизма Session Reuse (по умолчанию – `STATEFUL`):

Механизм Session Reuse (повторное использование SSL-сессии) применяется для протокола TLS версии 1.2 или старше и позволяет повторно использовать ранее согласованные SSL-параметры для новых соединений, что исключает необходимость выполнения полного рукопожатия (handshake).

Значение может быть:

- NONE – повторное использование SSL-сессии не происходит;
- STATEFUL – выполняется повторное использование SSL-сессии.

```
set ssl-profile server <имя> host <имя> setting session-reuse type <значение>
```

- (если используется STATEFUL) задание времени хранения (в секундах) SSL-сессии в кеше (по умолчанию – 7200):

```
set ssl-profile server <имя> host <имя> setting session-reuse stateful-param
session-timeout <значение>
```

- (опционально) задание комментария, который будет привязан к Серверному SSL-Профилю:

Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.

```
set ssl-profile server <имя> description <комментарий>
```

После настройки Профиля необходимо применить и сохранить данные:

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – XML, JSON или TXT):

```
show configuration xml ssl-profile server <имя_Профиля>
```

- просмотр выполненных команд:

```
show configuration cli ssl-profile server <имя_Профиля>
```

Termidesk Connect позволяет использовать особую конфигурацию Серверного SSL-Профиля для соединения, не попавшего в указанные правила.

Если пришедший SNI пуст, либо не соответствует SNI, указанному в параметре `host`, то для этого соединения будет применена конфигурация `setting-default`.

Создание и настройка Клиентского SSL-Профиля

Создание и настройка Клиентского SSL-Профиля выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. TLS**).

Описание параметров также приведено в подразделе **Объект** `ssl-profile`.

Пример команд настройки Клиентского Профиля:

- создание Клиентского SSL-Профиля:

```
set ssl-profile client <имя>
```

При создании Клиентского SSL-Профиля создается минимально необходимая конфигурация для него. Все настройки, приведенные далее, относятся к переопределению конфигурации по умолчанию.

- указание версии протокола TLS для данного SSL-Профиля:

Команда позволяет за один раз добавить только одно значение протокола TLS. После создания SSL-Профиля по умолчанию уже назначены следующие версии: tls-v1, tls-v11, tls-v12.

```
set ssl-profile client <имя> versions <версия_TLS> true
```

- использование механизма Secure Renegotiation (по умолчанию – `ENABLE`):

Механизм Secure Renegotiation применяется для протокола TLS версии 1.2 или старше и позволяет безопасно пересогласовать параметры SSL-соединения без разрыва текущего защищенного канала. Secure Renegotiation требуется в сценариях, при которых необходима периодическая смена ключей.

При активированном Secure Renegotiation:

- каждое новое повторное согласование соединения связывается с предшествующими сообщениями через специальное поле `renegotiation_info`, содержащее контрольную информацию о них;
- корректность значения поля `renegotiation_info` проверяется и если все проверки прошли успешно, то соединение продолжается. Если какая-либо из сторон соединения обнаруживает несоответствие в этом поле, то соединение обрывается.

```
set ssl-profile client <имя> ssl-reneg state ENABLE
```

- определение набора алгоритмов, которые могут использоваться для преобразования данных между Termidesk Connect и Реальным Сервером:

Команда позволяет за один раз добавить только один алгоритм.

После создания SSL-Профиля по умолчанию уже назначены следующие алгоритмы: TLS_AES_256_GCM_SHA384, TLS_CHACHA20_POLY1305_SHA25, TLS_AES_128_GCM_SHA256, ECDHE-ECDSA-AES256-GCM-SHA384, ECDHE-ECDSA-AES128-GCM-SHA256, ECDHE-ECDSA-AES256-SHA384, ECDHE-RSA-AES256-SHA384, ECDHE-ECDSA-AES128-SHA256, ECDHE-RSA-AES128-SHA256, ECDHE-ECDSA-AES256-SHA, ECDHE-RSA-AES256-SHA, ECDHE-ECDSA-AES128-SHA, ECDHE-RSA-AES128-SHA.

```
set ssl-profile client <имя> ciphers <алгоритм> true
```

- указание имени сервера по умолчанию для расширения SNI:

Если `sni-default` не установлен, то по умолчанию для соединения с Реальным Сервером используется SNI из запроса пользователя. Если `sni-default` установлен, то Termidesk Connect при соединении с Реальным Сервером будет использовать SNI, указанный в `sni-default`.

```
set ssl-profile client <имя> sni-default <имя>
```

- указание времени ожидания (в секундах) установки соединения с Реальным Сервером (значение от 1 до 60, по умолчанию – `5`):

```
set ssl-profile client <имя> handshake-timeout <значение>
```

- (опционально) указание файла сертификата УЦ, используемого для проверки подлинности Реального Сервера:

Указывается полное имя файла с расширением. Может быть выбран один файл.

Так же файл может быть получен из Хранилища секретов. Пример значения параметра: `pki://pki/issue/example-dot-com?common_name=\"app.example.com\"&ttl=\"24h\"#issuing_ca`.

```
set ssl-profile client <имя> ca-cert <имя_файла>
```

- (опционально) указание файла клиентского сертификата, используемого для аутентификации на Реальном Сервере:

Указывается полное имя файла с расширением. Пример значения параметра: `client.crt`.

Так же файл может быть получен из Хранилища секретов. Пример значения параметра: `pki://pki/issue/example-dot-com?common_name=\"app.example.com\"&ttl=\"24h\"#certificate`.

```
set ssl-profile client <имя> cert <имя_файла>
```

- (опционально) указание файла параметров Диффи-Хеллмана, используемого для обмена ключами и обеспечения безопасного соединения:

Указывается полное имя файла с расширением.

```
set ssl-profile client <имя> dh-params <имя_файла>
```

- (опционально) указание файла закрытого ключа, соответствующего клиентскому сертификату:

Указывается полное имя файла с расширением. Пример значения параметра: `crt.key`.

Так же файл может быть получен из Хранилища секретов. Пример значения параметра: `pki://pki/issue/example-dot-com?common_name=\"app.example.com\"&ttl=\"24h\"#private_key`.

```
set ssl-profile client <имя> key <имя_файла>
```

- (опционально) если ключ зашифрован, то указать пароль для расшифровки закрытого ключа:

Так же пароль может быть получен из Хранилища секретов. Пример значения параметра:

- `kv://secret/cert#password` – для `kv` версии 1;
- `kv://secret/data/snmp#password` – для `kv` версии 2.

```
set ssl-profile client <имя> password <пароль>
```

- (опционально) задание комментария, который будет привязан к Клиентскому SSL-Профилю:

Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.

```
set ssl-profile client <имя> description <комментарий>
```

После настройки профилей необходимо применить и сохранить данные:

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml ssl-profile client <имя>
```

- просмотр выполненных команд:

```
show configuration cli ssl-profile client <имя>
```

Политики SSL/TLS

SSL-Политики определяют выбор SSL-Профиля и (или) Сервера Балансировки на основании данных, полученных из сообщения TLS Hello.

Создание и настройка SSl-Политики выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **SSL-Политики**).

Для создания и настройки SSL-Политики используются команды:

Описание параметров также приведено в подразделе **Объект** `ssl-policy`.

- создание SSL-Политики:

```
set ssl-policy policy <имя>
```

- указание порядкового номера для условия SSL-Политики:

Чем ниже номер, тем выше приоритет, и тем раньше условие будет обработано SSL-Политикой.

Может быть задано несколько условий для одной SSL-Политики. Команда позволяет за один раз добавить только одно условие.

```
set ssl-policy policy <имя> rules <номер>
```

- указание условия для правила SSL-Политики:

Возможные типы условий:

- `ALPN` – проверка списка ALPN из TLS Hello. Условием может быть:
  - `eq` – точное соответствие `ALPN` и <значение>;
  - `not-eq` – точное несоответствие `ALPN` и <значение>;
- `CIPHER` – проверка списка алгоритмов шифрования из TLS Hello. Проверяется последовательность в любом месте на соответствие списку. Условием может быть:
  - `contains` – содержит последовательность в <значение>;
  - `not-contains` – не содержит последовательность в <значение>;
- `SNI` – проверка поля SNI из TLS Hello. Условием может быть:
  - `contains` – содержит последовательность в <значение>;
  - `not-contains` – не содержит последовательность в <значение>;
  - `eq` – точное соответствие `SNI` и <значение>;
  - `not-eq` – точное несоответствие `SNI` и <значение>;
  - `ends-with` – заканчивается на <значение>;
  - `starts-with` – начинается с <значение>;
- `DEFAULT` – действие при невыполнении условия (не содержит условие).

```
set ssl-policy policy <имя> rules <номер> <тип> <условие> <значение>
```

- указание действия при выполнении условия:

Параметром может быть:

- `lbs-id` – выбор Сервера Балансировки;
- `ssl-profile-id` – выбор SSL-Профиля;
- `drop` – разрыв соединения. Параметр не имеет значения.

```
set ssl-policy policy <имя> rules <номер> action <параметр> <значение>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml ssl-policy policy <имя>
```

- просмотр выполненных команд:

```
show configuration cli ssl-policy policy <имя>
```

Сценарии

Общие сведения о Сценариях

Сценарии – это `lua`-скрипты, являющиеся дополнением к основным параметрам конфигурации Виртуальных Серверов.

С помощью Сценариев (`lua`-скриптов) можно:

- выбирать Сервер Балансировки (Content Switching (cs));
- изменять содержимое запросов/ответов (Rewrite);
- отвечать на HTTP-запрос (Responder);
- прерывать нежелательные соединения.

`lua`-скрипты применяются к:

- Виртуальному Серверу для:
  - выбора Сервера Балансировки (Content Switching (cs));
  - ответа на HTTP-запрос (Responder);
  - изменения содержимого запросов (Rewrite);
  - прерывания нежелательного соединения;
- Серверу балансировки для:
  - ответа на HTTP-запрос (Responder) (например, при условии получении ответа);
  - изменения содержимого ответов (Rewrite);
  - прерывания нежелательного соединения (например, при условии получении ответа).

Условия выполнения того или иного действия определяются:

- логическими выражениями `if`, `elseif`, `else`, `end`;
- любыми данными, полученными из запроса, а также их комбинацией. Например:
  - `if client.http_req.host == "abc.ru" then` – если имя хоста из запроса эквивалентно «abc.ru», то выполнить какое-либо действие;
  - `if (client.http_req.host == "abc.ru" and client.remote_p.ip == "10.140.0.200") then` – если имя хоста из запроса эквивалентно «abc.ru», и запрос пришел с IP-адреса 10.140.0.200, то выполнить какое-либо действие.

Можно использовать возможности языка для работы с полученными данными, поскольку правила с четким соответствием не всегда применимы.

Пример для `curl qwe.123.ru/abczxc` (см. Пример выражений для работы с данными):

```
> GET /abczxc HTTP/1.1
> Host: qwe.123.ru
```

```
> User-Agent: curl/7.81.0
```

Таблица 16. Пример выражений для работы с данными

Выражение	Результат	Комментарий
`if client.http_req.path == "/abcZxc"`	`true`	Эквивалентно
`if client.http_req.path == "/abc"`	`false`	
`if (client.http_req.path:find("zx"))`	`true`	Содержит значение
`client.http_req.host:match("(.*).123.ru")`	`qwe`	Вернет строку, значение ДО « .123.ru»

Использование в Сценариях балансировки

Обработка правил в Сценариях балансировки осуществляется через глобальные переменные:

- `storage` – содержит методы для работы с путями к каталогам, которые будут доступны при вызове из `lua`-скрипта;
- `client` – содержит методы для работы с клиентом и запросом.

Обработка правил Сценария останавливается после выражения: `client.action = "xxx"`.

Описание полей приведено в таблице ниже.

Таблица 17. Описание полей переменной `storage`

Поле	Тип поля	Описание
`storage:path(name)`	Строка	Возвращает путь к каталогу, к которому можно обращаться по его наименованию (`name`) из Lua-скрипта. В текущей версии `name` принимает только значение `www`. Пример: выражение `storage.path("www")` вернет `/var/lib/tdc/www`
`storage:sub_path(name, path)`	Строка	Формирует полный путь (`path`) к файлу или подкаталогу внутри указанного каталога (`name`). В текущей версии `name` принимает только значение `www`. Пример: выражение `storage:sub_path("www", "page.html")` вернет `/var/lib/tdc/www/page.html`

Таблица 18. Описание полей переменной `client`

Поле	Тип поля	Описание
client.local	Структура	Псевдоним, то же самое, что client.local_p. Доступно только для чтения
client.local_p	Структура	Содержит поля локальной точки подключения. Доступно только для чтения
client.local_p.ip	Строка	Возвращает локальный IP-адрес подключения (IP-адрес Виртуального Сервера, vip.ip)
client.local_p.port	Строка	Возвращает локальный порт подключения (порт Виртуального Сервера, vip.port)
client.remote	Структура	Псевдоним, то же самое, что client.remote_p. Доступно только для чтения
client.remote_p	Структура	Содержит поля удаленной точки подключения (клиента). Доступно только для чтения
client.remote_p.ip	Строка	Возвращает IP-адрес клиента. Доступно только для чтения
client.remote_p.port	Целое	Возвращает порт клиента. Доступно только для чтения
client.action	Строка	Определяет действие, которое необходимо выполнить. Для Сценария балансировки: • client.action = "bs" – возвращает Сервер Балансировки, определенный в client.bs; • client.action = "gw" – возвращает Шлюз, определенный в client.gw; • client.action = "respond" – возвращает ответ клиенту, не передавая запрос на Реальный Сервер. Содержимое ответа описывается в client.respond.status; • client.action = "drop" – сбрасывает соединение; Для Сценария модификации ответа: • client.action = "respond" – возвращает ответ клиенту, не передавая запрос на Реальный Сервер. Содержимое ответа описывается в client.respond.status; • client.action = "drop" – сбрасывает соединение; • client.action = "pass" – используется для модификации ответов, после этого ответ передается клиенту При использовании аутентификации пользователя: • client.action = "auth" – останавливает обработку для аутентификации пользователя

Поле	Тип поля	Описание
`client.bs`	Строка	Определяет имя Сервера Балансировки. Пример: `client.bs = "lb1"`. При использовании в Сценарии балансировки свойство чтения и записи применяется, если `client.action = "bs"`. При использовании в Сценарии модификации ответа доступно только чтение, ответ будет содержать имя Сервера Балансировки
`client.gw`	Строка	Определяет имя Шлюза. Пример: `client.gw = "gw1"`
`client:bs_state(lbs_name)`	Строка	Возвращает статус Сервера Балансировки (`lbs_name`). Пример: `client:bs_state("ws1")`. Возвращаемые значения: • `NOT_FOUND`; • `ONLINE`; • `OFFLINE`; • `UNKNOWN`
`client.http_req`	Структура или `nil`	Класс для работы с HTTP-запросами. Описание полей приведено в таблице (см. таблицу Описание полей структуры `client.http_req`) В текущей версии Termidesk Connect значение `nil` для балансировки уровня L7 вернуться не может.
`client.http_resp`	Структура	Класс для работы с HTTP-ответами. Существует только для Сценария модификации ответа. Описание полей приведено в таблице (см. таблицу Описание полей структуры `client.http_resp`)
`client.rs.ip`	Строка	Возвращает IP-адрес Реального Сервера
`client.rs.port`	Строка	Возвращает порт Реального Сервера
`client.respond`	Структура	Класс для генерации HTTP-ответа. Применяется, если `client.action == "respond"`. Описание полей приведено в таблице (см. таблицу Описание полей структуры `client.respond`)
`client.aaa`	Структура	Класс для работы с AAA. Описание полей приведено в таблице (см. таблицу Описание полей структуры `client.aaa`)
`client.proxy_proto`	Структура или `nil`	Опциональное поле. Содержит `nil` или заголовок PROXY-протокола, если он был получен. Описание полей приведено в таблице (см. таблицу Описание полей структуры `client.proxy_proto` и `client.proxy_proto.ssl`)

Поле	Тип поля	Описание
client.proxy_proto.ssl	Структура или nil	Содержит nil или значение TLV PP2_TYPE_SSL, если оно было передано. Описание полей приведено в таблице (см. таблицу Описание полей структуры client.proxy_proto и client.proxy_proto.ssl)
client.sso	Структура	Класс для работы с SSO. Описание полей приведено в таблице (см. таблицу Описание полей структуры client.sso)
client.tls	Структура	Класс для чтения информации из сертификата клиента. Содержит данные начального этапа установления TLS-соединения. Описание полей приведено в таблице (см. таблицу Описание полей структуры client.tls) Некоторые поля структуры поддерживаются только для mTLS.

Примеры из таблицы ниже отражены в этом ответе на запрос к узлу termideskconnect по протоколу HTTP:

```
GET /termideskconnect?test=query HTTP/1.0
Host: connect.termidesk.ru
User-Agent: Mozilla Firefox/3.0.3
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Cookie: sessionid=100TC
```

Таблица 19. Описание полей структуры client.http_req

Поле	Тип поля	Описание
Работа с заголовками – client.http_req.header		
client.http_req.header:count()	Целое	Возвращает количество заголовков
client.http_req.header:get(i)	Строка	Возвращает пару: ключ и значение. Пример: X-Forwarded-For : 1.2.3.4
client.http_req.header:del([i])	Таблица (список строк)	Удаляет значение заголовка, индекс (i) опционален: • если индекс не задан, то удалятся все заголовки; • если индекс меньше 1, то удалится первый заголовок (равносильно i = 1); • если индекс больше, чем client.http_req.header:count(), то удалится последний заголовок
client.http_req.header:field_count(name)	Целое	Возвращает количество всех заголовков с именем name

Поле	Тип поля	Описание
`client.http_req.header:field_get(name [,i])`	Строка или `nil`	Возвращает пару: ключ (имя заголовка с учетом регистра) и значение: • если индекс не задан, то `i = client.http_req.header:size_field(name)`, возвращается последний найденный заголовок с именем `name`; • если индекс меньше `1` или больше `client.http_req.header:field_count(name)`, то ключ и значение будут `nil`. Значения параметров: • `name` – имя заголовка, строка без учета регистра; • `i` – индекс повторения заголовка `name`
`client.http_req.header:field_set(name, value [,i])`	Таблица (список строк)	Устанавливает значение заголовка (`key`, `value`) по индексу повторения (`i`): • если `value` число, то оно конвертируется в строку; • если индекс не задан, то удалятся все существующие заголовки, где `name = key`. Запишется только один заголовок; • если индекс меньше `1`, то перезапишется первый заголовок (равносильно `i = 1`); • если индекс больше `client.http_req.header:field_count(name)`, то перезапишется последнее вхождение заголовка; • если заданные `key` и `value` не найдены, то произойдет новая запись. Значения параметров: • `name` – имя заголовка, строка без учета регистра; • `value` – значение, строка или число; • `i` – индекс повторения заголовка `name`

Поле	Тип поля	Описание
`client.http_req.header:field_del(name, [i])`	Таблица (список строк)	Удаляет заголовок (`name`) по индексу повторения (`i`): • если индекс не задан, то удалятся все существующие заголовки с именем `name`; • если индекс меньше `1`, то удалится первое вхождение заголовка `name` (равносильно `i = 1`); • если индекс больше, чем `client.http_req.header:field_count(name)`, то удалится последнее вхождение заголовка `name`; • если `name` не найден, то действий не произойдет. Значения параметров: • `name` – имя заголовка, строка без учета регистра; • `i` – индекс повторения заголовка `name`, опциональный параметр
`client.http_req.header:field_push(name, value [,i])`	Строка	Добавляет (продвигает) заголовок (`key`, `value`) по индексу повторения (`i`): • если `value` число, то оно конвертируется в строку; • если индекс не задан или больше `client.http_req.header:field_count(name)`, то заголовок добавится в конец; • если индекс меньше `1`, то заголовок добавится в начало (равносильно `i = 1`); • если заданные `key` и `value` не найдены, то произойдет новая запись. Значения параметров: • `name` – имя заголовка, строка без учета регистра; • `value` – значение, строка или число; • `i` – индекс повторения заголовка `name`, опциональный параметр
`client.http_req.header[name]`	Таблица (список строк)	Возвращает таблицу строк значений заголовка `name`
`client.http_req.header[name] = value`	Таблица (список строк)	Удаляет или перезаписывает все вхождения заголовка `name`
`client.http_req.header = value`	Таблица (список строк)	Устанавливает значение для всех заголовков
Дополнительные свойства и методы		

Поле	Тип поля	Описание
`http_req.version`	Целое	Возвращает версию протокола HTTP: целое число, в котором младший десяток соответствует значению после точки, старший десяток – перед точкой. Поддерживаются версии: 1.0, 1.1. Пример: для HTTP/1.0 будет возвращен ответ: `10`
`http_req.method`	Строка	Возвращает метод запроса (GET, POST, HEAD, DELETE, PATCH, PUT, OPTIONS, CONNECT, TRACE и др.). Пример: `GET`
`http_req.path`	Строка	Возвращает путь запроса. Пример: `termideskconnect`
`http_req.query`	Строка	Возвращает запрос. Пример: `test=query`
`http_req.path_and_query`	Строка	Возвращает полный путь запроса. Пример: `termideskconnect?test=query`
`http_req.host`	Строка	Возвращает значение заголовка `Host`. Пример: `connect.termidesk.ru`
`client.http_req.has_body`	Логический	Возвращает значение `true`, если тело запроса было передано в Сценарий балансировки
`http_req.body`	Строка	Возвращает тело HTTP-запроса. Поле существует, если `client.http_req.has_body` = `true`

Таблица 20. Описание полей структуры `client.http_resp`

Поле	Тип поля	Описание
`client.http_resp.header:count()`	Целое	Возвращает количество заголовков
`client.http_resp.header:get(i)`	Строка	Возвращает пару: ключ и значение
`client.http_resp.header:del([i])`	Таблица (список строк)	Удаляет значение заголовка, индекс (`i`) опционален: • если индекс не задан, то удалятся все заголовки; • если индекс меньше `1`, то удалится первый заголовок (равносильно `i = 1`); • если индекс больше, чем `client.http_resp.header:count()`, то удалится последний заголовок
`client.http_resp.header:field_count(name)`	Целое	Возвращает количество всех заголовков с именем `name`

Поле	Тип поля	Описание
`client.http_resp.header:field_get(name [,i])`	Строка или `nil`	Возвращает пару: ключ (имя заголовка с учетом регистра) и значение: • если индекс не задан, то `i = client.http_resp.header:size_field(name)`, возвращается последний найденный заголовок с именем `name`; • если индекс меньше `1` или больше `client.http_resp.header:field_count(name)`, то ключ и значение будут `nil`. Значения параметров: • `name` – имя заголовка, строка без учета регистра; • `i` – индекс повторения заголовка `name`
`client.http_resp.header:field_set(name, value [,i])`	Таблица (список строк)	Устанавливает значение заголовка (`key`, `value`) по индексу повторения (`i`): • если `value` число, то оно конвертируется в строку; • если индекс не задан, то удалятся все существующие заголовки, где `name = key`. Запишется только один заголовок; • если индекс меньше `1`, то перезапишется первый заголовок (равносильно `i = 1`); • если индекс больше `client.http_resp.header:field_count(name)`, то перезапишется последнее вхождение заголовка; • если заданные `key` и `value` не найдены, то произойдет новая запись. Значения параметров: • `name` – имя заголовка, строка без учета регистра; • `value` – значение, строка или число; • `i` – индекс повторения заголовка `name`, параметр опционален

Поле	Тип поля	Описание
`client.http_resp.header:field_del(name, [i])`	Таблица (список строк)	Удаляет заголовок (`name`) по индексу повторения (`i`): • если индекс не задан, то удалятся все существующие заголовки с именем `name`; • если индекс меньше `1`, то удалится первое вхождение заголовока `name` (равносильно `i = 1`); • если индекс больше, чем `client.http_resp.header:field_count(name)`, то удалится последнее вхождение заголовока `name`; • если `name` не найден, то действий не произойдет. Значения параметров: • `name` – имя заголовка, строка без учета регистра; • `i` – индекс повторения заголовка `name`, опциональный параметр
`client.http_resp.header:field_push(name, value [,i])`	Строка	Добавляет (продвигает) заголовок (`key`, `value`) по индексу повторения (`i`): • если `value` число, то оно конвертируется в строку; • если индекс не задан или больше `client.http_resp.header:field_count(name)`, то заголовок добавится в конец; • если индекс меньше `1`, то заголовок добавится в начало (равносильно `i = 1`); • если заданные `key` и `value` не найдены, то произойдет новая запись. Значения параметров: • `name` – имя заголовка, строка без учета регистра; • `value` – значение, строка или число; • `i` – индекс повторения заголовка `name`, опциональный параметр
`client.http_resp.header[name]`	Таблица (список строк)	Возвращает таблицу строк значений заголовка `name`
`client.http_resp.header[name] = value`	Таблица (список строк)	Удаляет или перезаписывает все вхождения заголовка `name`
`client.http_resp.header = value`	Таблица (список строк)	Устанавливает значение для всех заголовков
`client.http_resp.status`	Целое	Возвращает или устанавливает код HTTP-ответа. Пример: `200`

Поле	Тип поля	Описание
`client.http_resp.has_body`	Логический	Возвращает `true`, если тело ответа было передано в Сценарий модификации ответов
`http_resp.body`	Строка	Возвращает тело HTTP ответа. Поле существует, если `client.http_resp.has_body = true`

Таблица 21. Описание полей структуры `client.respond`

Поле	Тип поля	Описание
Работа с заголовками		
`client.respond.header`	Структура	Класс для работы с заголовками ответа. Методы и свойства аналогичны `client.http_req.header`
`client.respond.version`	Целое	Возвращает или устанавливает версию протокола HTTP как целое число: • `11` – версия HTTP 1.1; • `10` – версия HTTP 1.0
`client.respond.status`	Целое	Возвращает или устанавливает код HTTP-ответа. Пример: `200`
`client.respond.body`	Строка	Возвращает или устанавливает тело HTTP-ответа

Таблица 22. Описание полей структуры `client.aaa`

Поле	Тип поля	Описание
`client.aaa.profile`	Строка	Задает AAA-Профиль. Название AAA-Профиля выбирается из конфигурации, в случае несовпадения выводится ошибка
`client.aaa:user_id()`	Строка	Возвращает идентификатор пользователя в формате `user@realm`. Для того, чтобы поле было непустое, необходимо установить `client.action="auth"`
`client.aaa:attempts()`	Целое	Возвращает количество неудачных попыток аутентификации за указанный в конфигурации период

Поле	Тип поля	Описание
`client.aaa:field_count(name)`	Строки	Возвращает, устанавливает или удаляет атрибуты пользователя, которые создаются во время процедуры аутентификации. Значения параметров: • `user` – имя пользователя; • `password` – пароль, введенный пользователем по запросу; • `memberOf` – атрибут группы пользователя из Active Directory. Поддерживается добавление индивидуальных атрибутов, которые будут храниться в течение жизни сессии
`client.aaa:field_set(name, value [,i])`		
`client.aaa:field_get(name [,i])`		
`client.aaa:field_push(name, value [,i])`		
`client.aaa:field_del(name [,i])`		

Таблица 23. Описание полей структуры `client.proxy_proto` и `client.proxy_proto.ssl`

Поле	Тип поля	Описание
Структура `proxy_proto`		
`proxy_proto.address_family`	Перечисление	Содержит тип адреса. Возможные значения: • `PP_AF.UNPSEC` – неопределенное; • `PP_AF.INET` – IPv4-адрес; • `PP_AF.INET6` – IPv6-адрес; • `PP_AF.UNIX` – Unix-сокет
`proxy_proto.command`	Перечисление	Возможные значения: • `PP_CMD.PROXY` – если заголовок содержит команду; • `PROXY PP_CMD.LOCAL` – если заголовок содержит команду `LOCAL` Для PROXY-протокола версии 1, всегда значение `PP_CMD.PROXY`.
`proxy_proto.dest`	Строка	Содержит адрес назначения: • если `client.proxy_proto.address_family == PP_AF.INET` или `client.proxy_proto.address_family == PP_AF.INET6`, то `client.proxy_proto.dest.ip` возвращает IP-адрес и `client.proxy_proto.dest.port` возвращает порт; • если `client.proxy_proto.address_family == PP_AF.UNIX`, то содержит строку с названием Unix-сокета; • если `client.proxy_proto.address_family == PP_AF.UNSPEC`, то содержит `nil`
`proxy_proto.is_local_command`	Логический	Значение `true`, если `client.proxy_proto.command == PP_CMD.LOCAL`

Поле	Тип поля	Описание
proxy_proto.is_proxy_command	Логический	Значение true, если client.proxy_proto.command == PP_CMD.PROXY
proxy_proto.is_sock_stream	Логический	Значение true, если client.proxy_proto.transport == PP_TRANSPORT.SOCK_STREAM
proxy_proto.is_sock_dgram	Логический	Значение true, если client.proxy_proto.transport == PP_TRANSPORT.SOCK_DGRAM
proxy_proto.is_sock_unspec	Логический	Значение true, если client.proxy_proto.transport == P_TRANSPORT.UNSPEC
proxy_proto.src	Строка, структура или nil	Содержит адрес источника: • если client.proxy_proto.address_family == PP_AF.INET или client.proxy_proto.address_family == PP_AF.INET6, то client.proxy_proto.src.ip возвращает IP-адрес и client.proxy_proto.src.port возвращает порт; • если client.proxy_proto.address_family == PP_AF.UNIX, то содержит строку с названием Unix-сокета; • если client.proxy_proto.address_family == PP_AF.UNSPEC, то содержит nil
proxy_proto:tlv_get(Type)	Строка	Возвращает TLV типа Type
proxy_proto.transport	Перечисление	Содержит тип транспортного протокола. Возможные значения: • PP_TRANSPORT.SOCK_STREAM – TCP-протокол; • PP_TRANSPORT.SOCK_DGRAM – UDP-протокол; • PP_TRANSPORT.UNSPEC — неопределенное
proxy_proto.version	Целое	Версия PROXY-протокола. Возможные значения: • 1 – версия 1; • 2 – версия 2
Структура proxy_proto.ssl		
proxy_proto.ssl.cert_in_conn	Логический	Указывает на то, что пользователь предоставил сертификат в данном соединении
proxy_proto.ssl.cert_in_session	Логический	Указывает на то, что пользователь предоставил сертификат как минимум один раз в рамках TLS-сессии данного соединения
proxy_proto.ssl.cert_verified	Логический	Указывает на то, что пользователь предъявил сертификат и он был успешно проверен
proxy_proto.ssl.cipher	Строка	Название алгоритма шифрования

Поле	Тип поля	Описание
`proxy_proto.ssl.cn`	Строка	Значение поля `Common Name` сертификата
`proxy_proto.ssl.key_algo`	Строка	Название алгоритма, используемого для генерации ключа сертификата
`proxy_proto.ssl.sig_algo`	Строка	Название алгоритма, используемого для подписи сертификата
`proxy_proto.ssl.tls`	Логический	Указывает на то, что используется TLS-соединение
`proxy_proto.ssl.version`	Строка	Версия TLS

Таблица 24. Описание полей структуры `client.sso`

Поле	Тип поля	Описание
`client.sso.profile`	Строка	Задает SSO-Профиль
`client.sso.user`	Строка	Устанавливает имя пользователя, если требуется
`client.sso.password`	Строка	Устанавливает пароль пользователя, если требуется

`client.sso:field_set`	Структура	Устанавливает параметры SSO-Профиля. Примеры:
		
		`client.sso.profile = "ss-krb-del"`
		`client.sso:field_set("user", client.aaa:field_get("user"))`
		`client.sso:field_set("service", "HTTP/krb.test.loc")`

Таблица 25. Описание полей структуры `client.tls`

Поле	Тип поля	Описание
`client.tls:certificate()`	Строка	Возвращает строку сертификата, закодированная в формате `Base64`. Поддерживается только для mTLS
`client.tls:version()`	Строка	Возвращает версию протокола TLS из сертификата. Поддерживается только для mTLS
`client.tls:serial_number()`	Строка	Возвращает серийный номер сертификата. Поддерживается только для mTLS
`client.tls:subject_get("name")`	Таблица (список строк)	Возвращает поля субъекта сертификата. Поддерживается только для mTLS. Значения параметра `name`: • `client.tls:subject_get("commonName")`; • `client.tls:subject_get("countryName")`; • `client.tls:subject_get("stateOrProvinceName")`; • `client.tls:subject_get("organizationName")`

Поле	Тип поля	Описание
`client.tls:issuer_get("name)`	Таблица (список строк)	Возвращает поля издателя сертификата. Поддерживается только для mTLS. Значения параметра `name`: • `client.tls:issuer_get("organizationName");` • `client.tls:issuer_get("stateOrProvinceName");` • `client.tls:issuer_get("countryName")`
`client.tls.sni`	Строка	Возвращает значение поля `sni` из подключения пользователя к Виртуальному Северу в сценарии, когда сесси терминируется на Termidesk Connect
`client.tls.alpn`	Строка	Возвращает значение поля `ALPN` в сценарии, когда сессия терминируется на Termidesk Connect
`client.tls.cipher`	Строка	Возвращает `cipher-suite`, выбранный при установке SSL-сессии с пользователем

Использование в Сценариях ошибок

Обработка правил Сценария ошибок осуществляется через глобальные переменные:

- `storage` – аналогично Сценариям балансировки содержит методы для работы с путями к каталогам, которые будут доступны при вызове из `lua`-скрипта;
- `client` – аналогично Сценариям балансировки содержит методы для работы с клиентом и запросом. Доступны все поля, кроме: `client.bs`, `client.action`, `client.aaa`, `client.sso`.

Описание используемых полей приведено в таблице (см. таблицу Описание полей для Сценариев ошибок).

Таблица 26. Описание полей для Сценариев ошибок

Поле	Тип поля	Описание
`context.script_errors`	Структура	Возвращает информацию об ошибке в `lua`-скрипте: • `error_message` — сообщение об ошибке; • `filename` — название файла, в котором произошла ошибка
`context.error`	Целое	Код ошибки
`context.bs `	Строка	Возвращает Сервер Балансировки
`context.rs.ip`	Строка	Возвращает IP-адрес Реального Сервера
`context.rs.port`	Целое	Возвращает порт Реального Сервера
`context.bad_header`	Строка	Заголовок, на котором был превышен лимит
`error_to_string(context.error)`	Строка	Возвращает код ошибки в виде строки
`status_to_string(status_code)`	Строка	Возвращает код HTTP-статуса в виде строки

Поддерживаемые коды ошибок перечислены в таблице (см. таблицу Описание кодов для Сценариев ошибок).

Таблица 27. Описание кодов для Сценариев ошибок

Поле	Описание
`Error.REQUEST_HEADER_LENGTH_LIMIT`	Превышено ограничение на размер одного заголовка
`Error.REQUEST_HEADERS_SIZE_LIMIT`	Превышено ограничение на общий размер заголовков
`Error.REQUEST_HEADER_COUNT_LIMIT`	Превышено ограничение на количество заголовков
`Error.REQUEST_BAD_CONTENT_LENGTH`	Неправильное значение заголовка `Content-Length`
`Error.REQUEST_MULTIPLE_CONTENT_LENGTH`	Несколько заголовков `Content-Length` в одном запросе
`Error.REQUEST_BAD_VERSION`	Неправильная или неподдерживаемая версия HTTP
`Error.REQUEST_MULTIPLE_UPGRADE`	Несколько заголовков `Upgrade` или значений
`Error.REQUEST_UPGRADE_NOT_ALLOWED`	`Upgrade` запрещен настройками (поле `upgrade_types` в HTTP-Профиле)
`Error.REQUEST_CONNECT_NOT_ALLOWED`	Метод `CONNECT` запрещен настройками (поле `allow-connect` в HTTP-Профиле)
`Error.BAD_REQUEST`	Другие ошибки разбора HTTP-запроса
`Error.SCRIPT_ERROR`	Ошибка выполнения Сценария балансировки (`lua`-скрипта)
`Error.BS_NOT_FOUND`	Сервер Балансировки, который вернул Сценарий (`lua`-скрипт), не существует
`Error.AUTHENTICATION_FAILED`	Ошибка аутентификации пользователя
`Error.BS_OFFLINE`	Сервер Балансировки, который вернул Сценарий (`lua`-скрипт), находится в состоянии `OFFLINE`
`Error.ERROR_CONNECTING_TO_RS`	Ошибка подключения к Реальному Серверу
`Error.ERROR_WRITING_REQUEST_TO_RS`	Ошибка при передаче запроса Реальному Серверу
`Error.ERROR_READING_RESPONSE_FROM_RS`	Ошибка при чтении ответа от Реального Сервера
`Error.RESPONSE_HEADER_LENGTH_LIMIT`	Превышено ограничение на размер одного заголовка
`Error.RESPONSE_HEADERS_SIZE_LIMIT`	Превышено ограничение на общий размер заголовков
`Error.RESPONSE_HEADER_COUNT_LIMIT`	Превышено ограничение на количество заголовков
`Error.RESPONSE_BAD_CONTENT_LENGTH`	Неправильное значение заголовка `Content-Length`
`Error.RESPONSE_MULTIPLE_CONTENT_LENGTH`	Несколько заголовков `Content-Length` в одном запросе
`Error.BAD_RESPONSE`	Другие ошибки разбора HTTP-ответа

Использование в Сценариях SSL

Обработка правил в Сценариях SSL осуществляется через глобальную переменную:

- `handshake` – содержит методы для работы с SSL-Политиками. Доступно только для

Сценариев SSL.

Описание полей приведено в таблице ниже.

Таблица 28. Описание полей переменной `handshake`

Поле	Тип поля	Описание
`handshake.sni`	Строка	Возвращает имя сервера, к которому подключается клиент. Поле доступно только на чтение, может быть пустым
`handshake.alpn`	Таблица (список строк)	Возвращает список протоколов в порядке очередности, которые могут быть выбраны сервером. Поле доступно только на чтение
`handshake.ciphers`	Таблица (список строк)	Возвращает список алгоритмов шифрования, поддерживаемых клиентом. Поле доступно только на чтение
`handshake.ssl-profile`	Строка	Возвращает идентификатор Серверного SSL-Профиля, используемого для установки соединения. По умолчанию поле пустое. `lua`-скрипт может установить любой валидный идентификатор, сконфигурированный в системе. Если значение остается пустым, то используется Серверный SSL-Профиль по-умолчанию из конфигурации Серевера Балансировки
`handshake.action`	Строка	Определяет действие, которое необходимо выполнить: • `handshake` — произвести рукопожатие и продолжить; • `drop` — сбросить соединение

Добавление Сценария

Добавление Сценария выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Сценарии**).

Пример команды с указанием порядка исполнения и имени файла Сценария:

```
set vs HTTP <имя> luarules 5 script test.lua
```

Файлы должны располагаться:

- Сценарии балансировки в каталоге `/var/lib/tdc/lbscripts/content-switching/`;
- Сценарии модификации ответов в каталоге

/var/lib/tdc/lbscripts/response-modifying/;

- Сценарии ошибок в каталоге /var/lib/tdc/lbscripts/error-reply/;
- Сценарии SSL в каталоге /var/lib/tdc/lbscripts/ssl.

Порядковый номер определяет очередность выполнения файла Сценария: например, Сценарий с порядковым номером 10 будет исполнен раньше, чем Сценарий с порядковым номером 15.

При этом правила в файлах Сценариев также обрабатываются по порядку:

- сначала правила из файла Сценария с наименьшим порядковым номером;
- затем правила из файла Сценария со следующим порядковым номером, если не было совпадений.

Правила Сценария могут располагаться как в одном файле, так и в нескольких.

Примеры Сценариев для изменения содержимого запросов

Пример 1. Если имя хоста содержит текст «abc» и запрос пришел из сети 192.0.2.0/24, то добавить заголовок «XFF: <IP-адрес_пользователя_при_подключении>», и перенаправить запрос на Сервер Балансировки «lb1».

Код:

```
if (client.http_req.host:find("abc") and
client.remote_p:is_network("192.0.2.0/24")) then
        client.http_req.header:field_set("XFF", client.remote_p.ip)
        client.bs = "lb1"
        client.action = "bs"
end
```

Результат:

```
WEB -- 03<p>Method GET</p><p>URL on server: /</p><p>REQ
Headers: </p>Host: abc.domain.ru
User-Agent: curl/7.81.0
Accept: */*
XFF: 192.0.2.5
```

Пример 2. Если имя хоста начинается с «abc», то добавить заголовок «XFF: <IP-адрес_пользователя_при_подключении>» и «Remote-port: <порт-источник_на_клиенте>», и перенаправить запрос на Сервер Балансировки «lb1».

Код:

```
function startswith(text, prefix)
  return text:find(prefix, 1, true) == 1
end
```

```
if startswith(client.http_req.host, "abc") then
  client.http_req.header:field_set("Remote-port", client.remote_p.port)
  client.http_req.header:field_set("XFF", client.remote_p.ip)
  client.bs = "lb1"
  client.action = "bs"
end
```

Пример 3. Если путь запроса содержит «app», то при наличии заголовка X-Forwarded-For добавить IP-адрес клиента в конец запроса. Если заголовка не было в запросе, то добавить новый. Перенаправить запрос на Сервер Балансировки «lb1».

Код:

```
if client.http_req.path:find("abc") then
        if client.http_req.header:field_count("X-Forwarded-For") > 0 then
                xff, xffip = client.http_req.header:field_get("X-Forwarded-For")
                client.http_req.header:field_set("X-Forwarded-For", xffip ..",
"..client.remote_p.ip)
        else
                client.http_req.header:field_set("X-Forwarded-For",
client.remote_p.ip)
        end
client.bs = "lb1"
client.action = "bs"

end
```

Пример 4. Подменить домен «ru» на домен «local» (например, если запрос идет к app.domain.ru, то в сторону Реального Сервера должен прийти app.domain.local). В начале пути запроса добавить «/external» и перенаправить запрос на Сервер Балансировки «lb1».

Код:

```
client.http_req.host =  client.http_req.host:match("(.*).ru") .. ".local"
client.http_req.path = "/external" .. client.http_req.path
client.bs = "lb1"
client.action = "bs"
```

Примеры Сценариев для ответа на запрос

Пример 1. Если метод не GET или не HEAD, то сбросить соединение.

Код:

```
if (client.http_req.method ~= "GET" and client.http_req.method ~= "HEAD") then
    client.action = "drop"
end
```

Пример 2. Перенаправить соединение с HTTP на HTTPS.

Код:

```
client.respond.status = 302
client.respond.header["Location"] = "https://" .. client.http_req.host ..
client.http_req.path
client.respond.header["Connection"] = "close"
client.action = "respond"
```

Пример 3. Если запрос из сети 192.0.2.0/24, то ответить HTML-страницей и параметрами запроса.

Код:

```
if client.remote_p:is_network("192.0.2.0/24") then
    client.respond.header["Connection"] = "close"
    client.respond.header["Content-type"] = 'text/html
    client.respond.body = [[<html>
    <body>
    <meta charset="UTF-8">
    <h1>Lets goodbye!</h1>
    <p>Доступ запрещен</p>
    </body>
    </html>]] .. "IP: " .. client.remote_p.ip .. "\n TRY: " .. client.http_req.host
.. client.http_req.path .. "\n" ..      " Vserver: " .. client.local_p.ip .. ":" ..
client.local_p.port
client.action = 'respond'
```

Примеры Сценариев для выбора Сервера Балансировки

Пример 1. Передача любого запроса на Сервер Балансировки «lb1».

Код:

```
client.bs = "lb1"
client.action = "bs"
```

Пример 2. Если имя хоста в запросе точно соответствует «abc.domain.ru», то направить запрос на Сервер Балансировки «lb1». В противном случае вернуть код ответа 403 (по умолчанию вернется ошибка 503).

Код:

```
if client.http_req.host == "abc.domain.ru" then
        client.bs = "lb1"
        client.action = "bs"
else
```

```
        client.respond.status = 403
        client.action = "respond"
end
```

Пример 3. Если имя хоста содержит текст «abc» и запрос пришел из сети 192.0.2.0/24, то вернуть Сервер Балансировки «lb1». В противном случае вернуть код ответа 403 (по умолчанию вернется ошибка 503).

Код:

```
if (client.http_req.host:find("abc") and
client.remote_p:is_network("192.0.2.0/24"))
then
        client.bs = "lb1"
        client.action = "bs"
else
        client.respond.status = 403
        client.action = "respond"
end
```

Пример 4. Если путь содержит:

- «red», то вернуть Сервер Балансировки «lb1»;
- «green», то вернуть Сервер Балансировки «lb2».

Если не сработало ни одно из условий выше, то вернуть Сервер Балансировки «lb-default».

Код:

```
if client.http_req.path:find("red") then
        client.bs = "lb1"
        client.action = "bs"
elseif client.http_req.path:find("green") then
        client.bs = "lb2"
        client.action = "bs"
else
        client.bs = "lb-default"
        client.action = "bs"
end
```

Виртуальные Серверы

Общие сведения о Виртуальных Серверах

Виртуальные Серверы в Termidesk Connect – это объект (абстракция), являющийся точкой входа для пользователей. Виртуальный Сервер принимает на себя подключение пользователя и далее через Серверы Балансировки принимается решение, на какой из Реальных Серверов будет направлено подключение.

Взаимодействие с Виртуальным Сервером представлено на рисунке (см. Взаимодействие с Виртуальным Серовером).

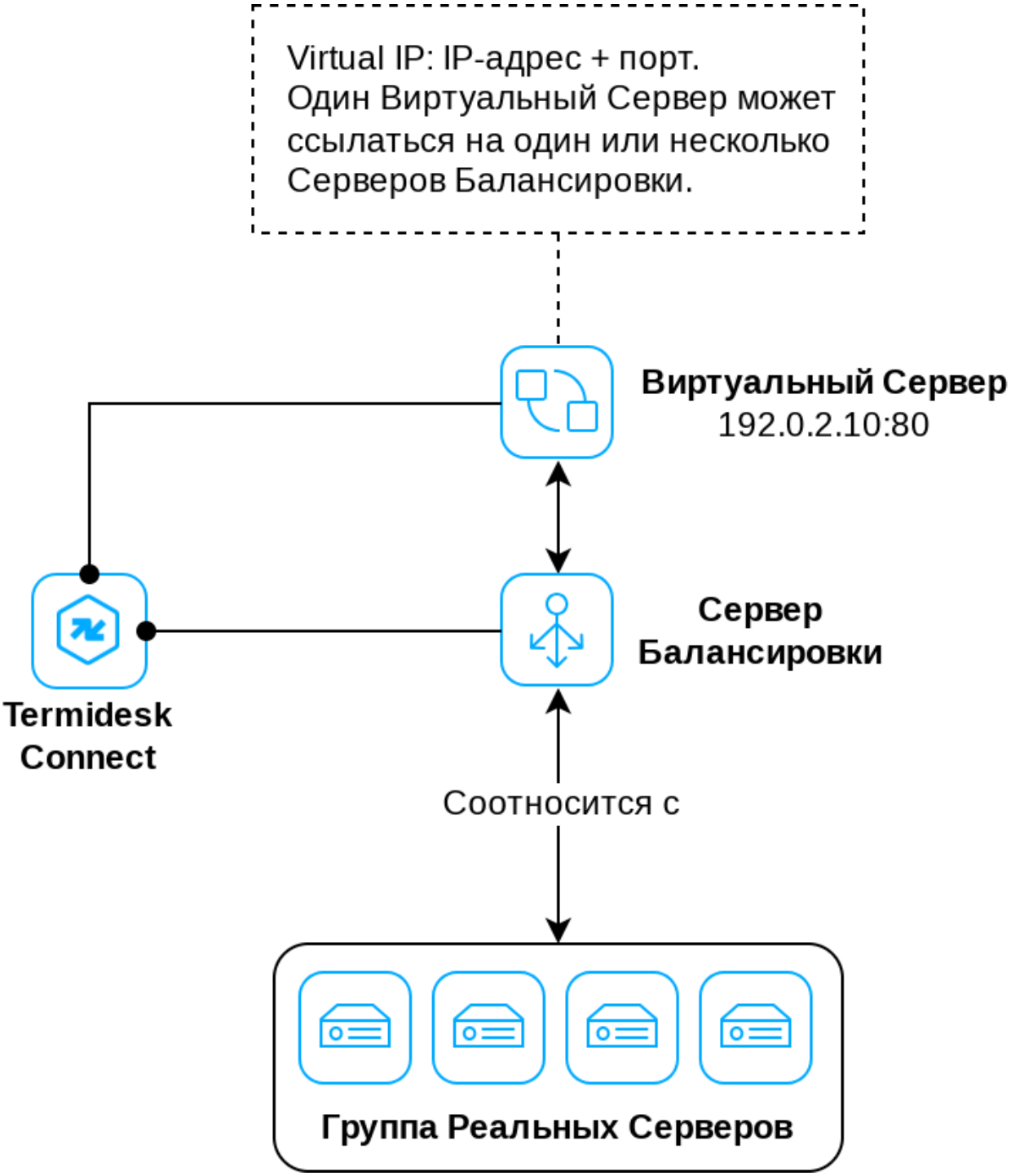

Рисунок 61. Взаимодействие с Виртуальным Серовером

Виртуальный Сервер может работать для балансировки протокола TCP (уровень L4 в модели OSI), RAPID-TCP (уровень L4 в модели OSI), RAPID-UDP (уровень L4 в модели OSI) или HTTP (уровни L3-L7 в модели OSI). Выбор протокола зависит от того, какой тип балансировки требуется реализовать:

- при балансировке уровня L4 по протоколу TCP (см. Балансировка уровня L4 для TCP) клиент устанавливает соединение с Виртуальным Сервером Termidesk Connect, далее Termidesk Connect задействует алгоритмы, определенные в Сервере Балансировки, и устанавливает TCP-соединение с Реальным Сервером. В рамках установленного TCP-соединения передаются данные, которые Termidesk Connect не фильтрует;

- при балансировке уровня L4 по протоколу RAPID-TCP (см. Балансировка уровня L4 для RAPID-TCP) клиент отправляет пакет SYN на Виртуальный Сервер Termidesk Connect. Уже на этом этапе Termidesk Connect задействует алгоритмы, определенные в Сервере Балансировки, и перенаправляет пакет SYN на Реальный Сервер. После установления TCP-соединения данные передаются в рамках него;
- при балансировке уровня L4 по протоколу RAPID-UDP (см. Балансировка уровня L4 для RAPID-UDP) клиент отправляет данные на Виртуальный Сервер Termidesk Connect. Termidesk Connect задействует алгоритмы, определенные в Сервере Балансировки, и перенаправляет данные на Реальный Сервер;
- при балансировке уровня L7 (см. Балансировка уровня L7) клиент устанавливает TCP-соединение с Виртуальным Сервером Termidesk Connect, далее Termidesk Connect ожидает от клиента HTTP-запрос. На основании переданного HTTP-запроса Termidesk Connect задействует алгоритмы, определенные в Сервере Балансировки, и устанавливает TCP-соединение с Реальным Сервером.

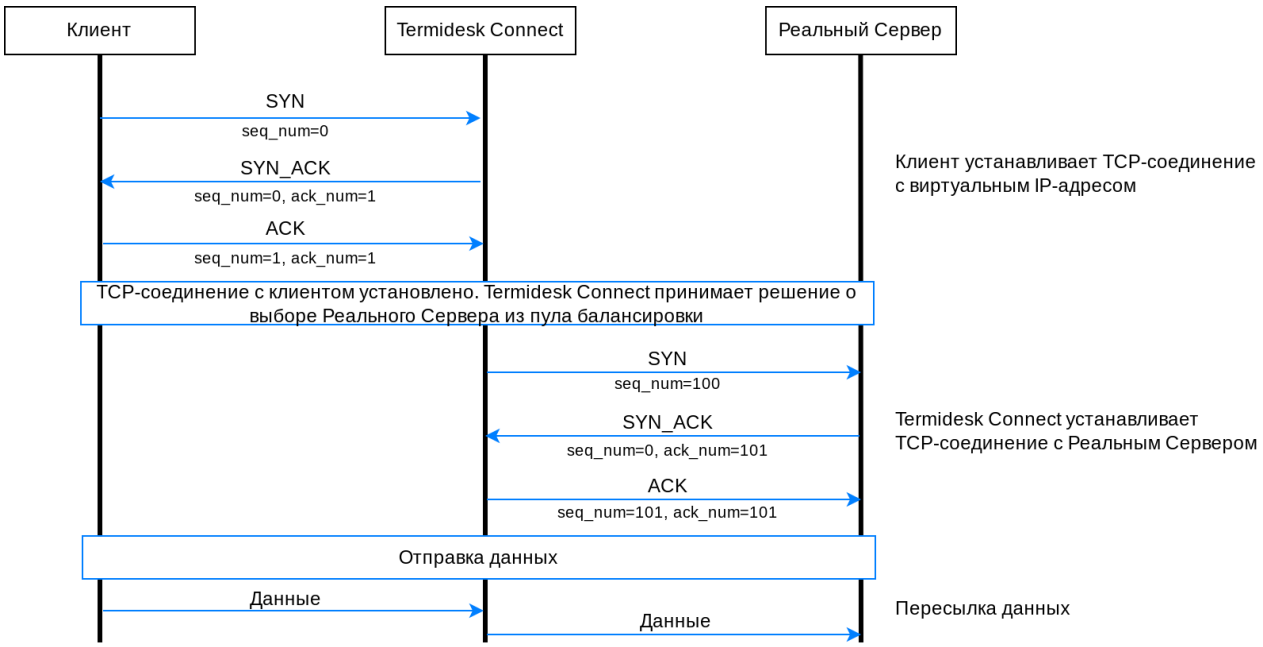

Рисунок 62. Балансировка уровня L4 для TCP

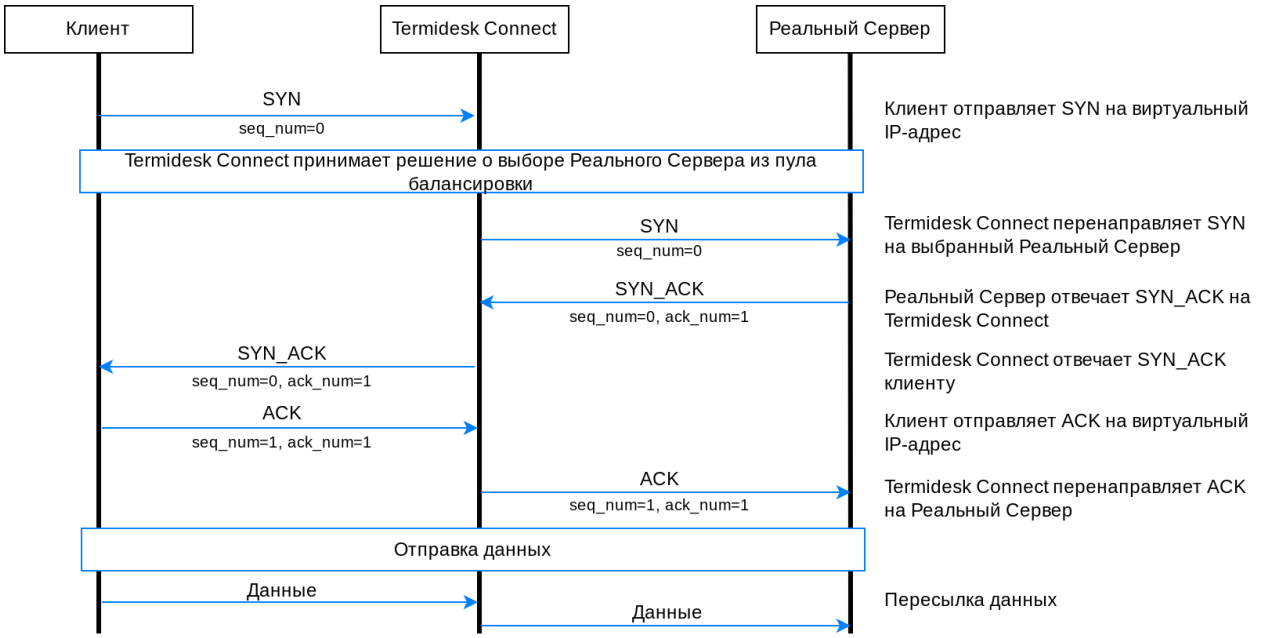

Рисунок 63. Балансировка уровня L4 для RAPID-TCP

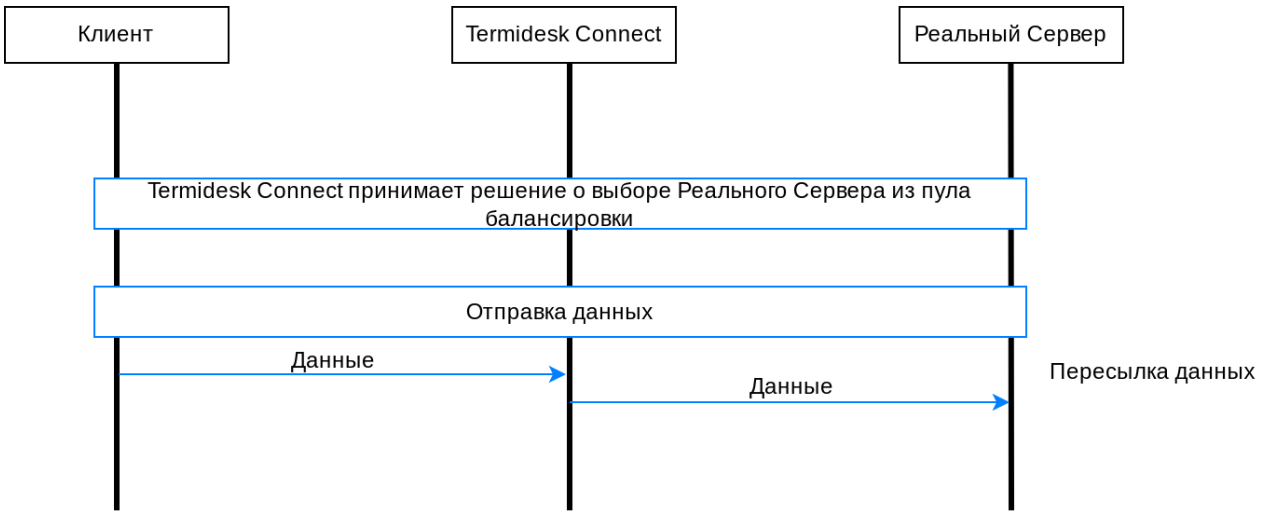

Рисунок 64. Балансировка уровня L4 для RAPID-UDP

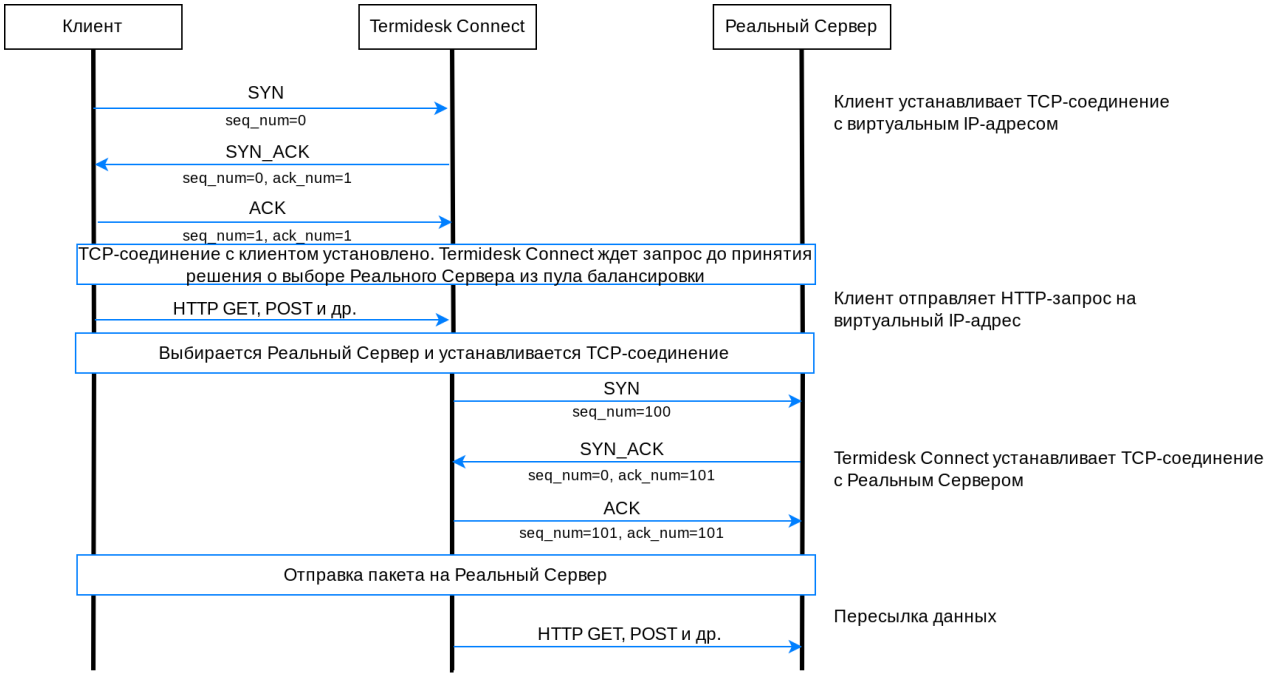

Рисунок 65. Балансировка уровня L7

Создание и настройка Виртуального Сервера для балансировки по протоколу TCP

Создание и настройка Виртуального Сервера выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см.подраздел **Веб. Виртуальные Серверы**).

Для создания и настройки Виртуального Сервера для балансировки по протоколу TCP используются команды:

Описание параметров также приведено в подразделе **Объект** `vs`.

- создание Виртуального Сервера:

```
set vs TCP <имя>
```

- задание IP-адреса Виртуального Сервера:

```
set vs TCP <имя> vip ip <IP-адрес>
```

- задание порта Виртуального Сервера:

```
set vs TCP <имя> vip port <порт>
```

Виртуальный Сервер может принимать входящие сообщения также на диапазон портов или список портов для решения сценариев:

- балансировки одновременных подключений по нескольким портам;
- ограничения занимаемых портов и использования только конкретного списка или диапазона.

В этих случаях команда назначения портов будет отличаться:

- для задания диапазона портов:

```
set vs TCP <имя> vip portrange from <начало_диапазона>
set vs TCP <имя> vip portrange to <конец_диапазона>
```

- для задания списка портов (каждый порт задается отдельной командой для добавления в список):

```
set vs TCP <имя> vip portlist <порт>
```

Если на Виртуальном Сервере настроен диапазон или список портов, а на Реальном Сервере указан конкретный порт, то трафик перенаправляется на этот порт. Если на Реальном Сервере указаны все порты или порт не задан, то порт из запроса сохраняется и передается на Реальный Сервер без изменений.

- задание алгоритма определения статуса Виртуального Сервера:

Алгоритм определения статуса может быть:

- `OR` – статус Виртуального Сервера будет «В работе», если работает хотя бы один ассоциированный с ним Сервер Балансировки;
- `AND` – статус Виртуального Сервера будет «В работе», если работают все ассоциированные с ним Серверы Балансировки;
- `NONE` – не использовать никакой из алгоритмов.

```
set vs TCP <имя> check-lbs algorithm <алгоритм>
```

- задание Сервера Балансировки (можно указать несколько узлов), по состоянию которого будет определяться статус Виртуального Сервера:

Статус Виртуального Сервера зависит от статуса ассоциированного с ним Сервера Балансировки. Алгоритм определения статуса задается в команде выше.

```
set vs TCP <имя> check-lbs lbs-ids <имя>
```

- (опционально) задание правила выбора Сервера Балансировки:

Параметр `<номер>` задает порядковый номер применения правила: чем ниже номер, тем выше приоритет, и тем раньше правило будет обработано Виртуальным Сервером.

```
set vs TCP <имя> netrules <номер>
```

- задание сети источника в формате CIDR для правила:

В правиле задается сеть источника запроса: в зависимости от того, из какой сети подключился пользователь, будет выбран тот или иной Сервер Балансировки.

```
set vs TCP <имя> netrules <номер> network <сеть><префикс>
```

- задание Сервера Балансировки для правила:

```
set vs TCP <имя> netrules <номер> lbs-id <имя>
```

- привязка Серверного TCP-Профиля к Виртуальному Серверу (см. подраздел **Профили**):

```
set lbs TCP <имя> tcp-profile-id <имя_Профиля>
```

- (опционально) привязка Серверного SSL-Профиля к Виртуальному Серверу (см. подраздел **TLS**):

```
set vs TCP <имя> ssl-profile-id <имя_Профиля>
```

- (опционально) привязка SSL-Политики к Виртуальному Серверу (см. подраздел **TLS**):

```
set lbs TCP <имя> ssl-policy-id <имя_Политики>
```

- привязка VRF к Виртуальному Серверу:

```
set vs TCP <имя> vrf <имя_VRF>
```

- (опционально) активация возможности анонсирования IP-адреса (Route Health Injection – RHI), привязанного к Виртуальному Серверу, протоколам динамической маршрутизации:

Для работы RHI необходимо, чтобы динамическая маршрутизация была включена на Termidesk Connect (см. подраздел **Сеть**).

Параметру могут быть заданы значения:

- `ON` – активация RHI. При активации RHI Termidesk Connect будет анонсировать в сеть IP-адреса Виртуальных Серверов в зависимости от их режима работы;
- `OFF` – отключение RHI.

```
set vs TCP <имя> rhi <значение>
```

- (опционально, только при активации RHI) задание режима работы RHI для Виртуального Сервера:

Параметру могут быть заданы значения:

- `ACTIVE` – активный режим RHI для Виртуального Сервера;
- `PASSIVE` – пассивный режим RHI для Виртуального Сервера.

Идентификатором маршрута являются пары «VRF – IP-адрес» (поддерживаются множественные VRF), при этом состояние маршрута определяется условиями:

- если все Виртуальные Серверы по данному маршруту находятся в режиме `PASSIVE`, то Termidesk Connect всегда будет объявлять маршрут для виртуального IP-адреса;
- если хотя бы один Виртуальный Сервер находятся в режиме `ACTIVE` и в состоянии «В работе», то Termidesk Connect будет объявлять маршрут для виртуального IP-адреса;
- в остальных случаях Termidesk Connect не будет объявлять маршрут.

```
set vs TCP <имя> rhi-state <значение>
```

- (опционально) привязка Профиля ограничения скорости (см. подраздел **Профили ограничения скорости**):

```
set vs TCP <имя> rl-profile-id <имя_Профиля>
```

- (опционально) задание комментария, который будет привязан к Виртуальному Серверу:

Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.

```
set vs TCP <имя> description <комментарий>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml vs TCP <имя>
```

- просмотр выполненных команд:

```
show configuration cli vs TCP <имя>
```

Создание и настройка Виртуального Сервера для балансировки по протоколу HTTP

Создание и настройка Виртуального Сервера выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см.подраздел **Веб. Виртуальные Серверы**).

Для создания и настройки Виртуального Сервера для балансировки по протоколу HTTP используются команды:

Описание параметров также приведено в подразделе **Объект** `vs`.

- создание Виртуального Сервера:

```
set vs HTTP <имя>
```

- задание IP-адреса:

```
set vs HTTP <имя> vip ip <IP-адрес>
```

- задание порта:

```
set vs HTTP <имя> vip port <порт>
```

Виртуальный Сервер может принимать входящие сообщения также на диапазон портов или список портов для решения сценариев:

- балансировки одновременных подключений по нескольким портам;
- ограничения занимаемых портов и использования только конкретного списка или диапазона.

В этих случаях команда назначения портов будет отличаться:

- для задания диапазона портов:

```
set vs HTTP <имя> vip portrange from <начало_диапазона>
set vs HTTP <имя> vip portrange to <конец_диапазона>
```

- для задания списка портов (каждый порт задается отдельной командой для добавления в список):

```
set vs HTTP <имя> vip portlist <порт>
```

Если на Виртуальном Сервере настроен диапазон или список портов, а на Реальном Сервере указан конкретный порт, то трафик перенаправляется на этот порт. Если на Реальном Сервере указаны все порты или порт не задан, то порт из запроса сохраняется и передается на Реальный Сервер без изменений.

- задание алгоритма определения статуса Виртуального Сервера:

```
set vs HTTP <имя> check-lbs algorithm <алгоритм>
```

- задание Сервера Балансировки (можно указать несколько узлов), по состоянию которого будет определяться статус Виртуального Сервера:

```
set vs HTTP <имя> check-lbs lbs-ids <имя>
```

- задание Сценария (исполняемого файла формата LUA), который определяет правила обработки трафика, проходящего через Виртуальный Сервер:

Файл формата LUA должен быть расположен в каталоге

/var/lib/tdc/lbscripts/content-switching/.

```
set vs HTTP <имя> luarules <номер> script <имя_файла>
```

Параметр <номер> задает порядковый номер применения Сценария: чем ниже номер, тем выше приоритет, и тем раньше Сценарий будет обработан Виртуальным Сервером.

Подробное описание работы со Сценариями приведено в подразделе **Сценарии**.

- задание Сценария (исполняемого файла формата LUA), который определяет выбор SSL-Профиля или Сервера Балансировки на основании данных, полученных из сообщения TLS Hello:

Файл формата LUA должен быть расположен в каталоге /var/lib/tdc/lbscripts/ssl.

```
set vs HTTP <имя> sslrules <номер> script <имя_файла>
```

- привязка Серверного HTTP-Профиля к Виртуальному Серверу (см. подраздел **Профили**):

```
set vs HTTP <имя> http-profile-id <имя_Профиля>
```

- привязка Серверного TCP-Профиля к Виртуальному Серверу (см. подраздел **Профили**):

```
set vs HTTP <имя> tcp-profile-id <имя_Профиля>
```

- (опционально) привязка Серверного SSL-Профиля к Виртуальному Серверу (см. подраздел **TLS**):

```
set vs HTTP <имя> ssl-profile-id <имя_Профиля>
```

- (опционально) привязка VRF к Виртуальному Серверу:

```
set vs HTTP <имя> vrf <имя_VRF>
```

- (опционально) активация возможности анонсирования IP-адреса (Route Health Injection – RHI), привязанного к Виртуальному Серверу, протоколам динамической маршрутизации:

Для работы RHI необходимо, чтобы динамическая маршрутизация была включена на Termidesk Connect (см. подраздел **Сеть**).

Параметру могут быть заданы значения:

- `ON` – активация RHI. При активации RHI Termidesk Connect будет анонсировать в сеть IP-адреса Виртуальных Серверов в зависимости от их режима работы;
- `OFF` – отключение RHI.

```
set vs HTTP <имя> rhi <значение>
```

- (опционально, только при активации RHI) задание режима работы RHI для Виртуального Сервера:

Параметру могут быть заданы значения:

- `ACTIVE` – активный режим RHI для Виртуального Сервера;
- `PASSIVE` – пассивный режим RHI для Виртуального Сервера.

Идентификатором маршрута являются пары «VRF – IP-адрес» (поддерживаются множественные VRF), при этом состояние маршрута определяется условиями:

- если все Виртуальные Сервера по данному маршруту находятся в режиме `PASSIVE`, то Termidesk Connect всегда будет объявлять маршрут для виртуального IP-адреса;
- если хотя бы один Виртуальный Сервер находятся в режиме `ACTIVE` и в состоянии «В работе», то Termidesk Connect будет объявлять маршрут для виртуального IP-адреса;
- в остальных случаях Termidesk Connect не будет объявлять маршрут.

```
set vs HTTP <имя> rhi-state <значение>
```

- (опционально) привязка Профиля ограничения скорости (см. подраздел **Профили ограничения скорости**):

```
set vs HTTP <имя> rl-profile-id <имя_Профиля>
```

- (опционально) задание комментария, который будет привязан к Виртуальному Серверу:

Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.

```
set vs HTTP <имя> description <комментарий>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – XML, JSON или TXT):

```
show configuration xml vs HTTP <имя>
```

- просмотр выполненных команд:

```
show configuration cli vs HTTP <имя>
```

Создание и настройка Виртуального Сервера для балансировки по протоколу RAPID-TCP

Создание и настройка Виртуального Сервера выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см.подраздел **Веб. Виртуальные Серверы**).

Для создания и настройки Виртуального Сервера для балансировки по протоколу RAPID-TCP используются команды:

Описание параметров также приведено в подразделе **Объект** vs.

- создание Виртуального Сервера:

```
set vs RAPID-TCP <имя>
```

- задание IP-адреса Виртуального Сервера:

```
set vs RAPID-TCP <имя> vip ip <IP-адрес>
```

- задание порта Виртуального Сервера:

```
set vs RAPID-TCP <имя> vip port <порт>
```

Виртуальный Сервер может принимать входящие сообщения также на диапазон портов, список портов или на все порты для решения сценариев:

- балансировки одновременных подключений по нескольким портам;

- ограничения занимаемых портов и использования только конкретного списка или диапазона;
- обработки заранее неизвестных портов.

В этих случаях команда назначения портов будет отличаться:

- для задания диапазона портов:

```
set vs RAPID-TCP <имя> vip portrange from <начало_диапазона>
set vs RAPID-TCP <имя> vip portrange to <конец_диапазона>
```

- для задания списка портов (каждый порт задается отдельной командой для добавления в список):

```
set vs RAPID-TCP <имя> vip portlist <порт>
```

- для задания всех портов:

```
set vs RAPID-TCP <имя> vip port 0
```

Если на Виртуальном Сервере настроен диапазон или список портов, а на Реальном Сервере указан конкретный порт, то трафик перенаправляется на этот порт. Если на Реальном Сервере указаны все порты или порт не задан, то порт из запроса сохраняется и передается на Реальный Сервер без изменений.

- задание алгоритма определения статуса Виртуального Сервера:

Алгоритм определения статуса может быть:

- OR – статус Виртуального Сервера будет «В работе», если работает хотя бы один ассоциированный с ним Сервер Балансировки;
- AND – статус Виртуального Сервера будет «В работе», если работают все ассоциированные с ним Серверы Балансировки;
- NONE – не использовать никакой из алгоритмов.

```
set vs RAPID-TCP <имя> check-lbs algorithm <алгоритм>
```

- задание Сервера Балансировки (можно указать несколько узлов), по состоянию которого будет определяться статус Виртуального Сервера:

Статус Виртуального Сервера зависит от статуса ассоциированного с ним Сервера Балансировки. Алгоритм определения статуса задается в команде выше.

```
set vs RAPID-TCP <имя> check-lbs lbs-ids <имя>
```

- задание правила выбора Сервера Балансировки:

Параметр `<приоритет>` задает приоритет применения правила: чем ниже число, тем выше приоритет, и тем раньше правило будет обработано Виртуальным Сервером.

```
set vs RAPID-TCP <имя> netrules <приоритет>
```

- задание сети источника в формате CIDR для правила:

В правиле задается сеть источника запроса: в зависимости от того, из какой сети подключился пользователь, будет выбран тот или иной Сервер Балансировки.

```
set vs RAPID-TCP <имя> netrules <приоритет> network <сеть><длина_префикса>
```

- задание Сервера Балансировки для правила:

```
set vs RAPID-TCP <имя> netrules <приоритет> lbs-id <имя>
```

- (опционально) привязка VRF к Виртуальному Серверу:

```
set vs RAPID-TCP <имя> vrf <имя_VRF>
```

- (опционально) активация возможности анонсирования IP-адреса (Route Health Injection – RHI), привязанного к Виртуальному Серверу, протоколам динамической маршрутизации:

Для работы RHI необходимо, чтобы динамическая маршрутизация была включена на Termidesk Connect (см. подраздел **Сеть**).

Параметру могут быть заданы значения:

- `ON` – активация RHI. При активации RHI Termidesk Connect будет анонсировать в сеть IP-адреса Виртуальных Серверов в зависимости от их режима работы;
- `OFF` – отключение RHI.

```
set vs RAPID-TCP <имя> rhi <значение>
```

- (опционально, только при активации RHI) задание режима работы RHI для Виртуального Сервера:

Параметру могут быть заданы значения:

- `ACTIVE` – активный режим RHI для Виртуального Сервера;

- PASSIVE – пассивный режим RHI для Виртуального Сервера.

Идентификатором маршрута являются пары «VRF – IP-адрес» (поддерживаются множественные VRF), при этом состояние маршрута определяется условиями:

- если все Виртуальные Серверы по данному маршруту находятся в режиме PASSIVE, то Termidesk Connect всегда будет объявлять маршрут для виртуального IP-адреса;
- если хотя бы один Виртуальный Сервер находятся в режиме ACTIVE и в состоянии «В работе», то Termidesk Connect будет объявлять маршрут для виртуального IP-адреса;
- в остальных случаях Termidesk Connect не будет объявлять маршрут.

```
set vs RAPID-TCP <имя> rhi-state <значение>
```

- (опционально) задание комментария, который будет привязан к Виртуальному Серверу:

Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.

```
set vs RAPID-TCP <имя> description <комментарий>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – XML, JSON или TXT):

```
show configuration xml vs RAPID-TCP <имя>
```

- просмотр выполненных команд:

```
show configuration cli vs RAPID-TCP <имя>
```

Создание и настройка Виртуального Сервера для балансировки по протоколу RAPID-UDP

Создание и настройка Виртуального Сервера выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см.подраздел **Веб. Виртуальные Серверы**).

Для создания и настройки Виртуального Сервера для балансировки по протоколу RAPID-UDP используются команды:

Описание параметров также приведено в подразделе **Объект** vs.

- создание Виртуального Сервера:

```
set vs RAPID-UDP <имя>
```

- задание IP-адреса Виртуального Сервера:

```
set vs RAPID-UDP <имя> vip ip <IP-адрес>
```

- задание порта Виртуального Сервера:

```
set vs RAPID-UDP <имя> vip port <порт>
```

Виртуальный Сервер может принимать входящие сообщения также на диапазон портов, список портов или на все порты для решения сценариев:

- балансировки одновременных подключений по нескольким портам;
- ограничения занимаемых портов и использования только конкретного списка или диапазона;
- обработки заранее неизвестных портов.

В этих случаях команда назначения портов будет отличаться:

- для задания диапазона портов:

```
set vs RAPID-UDP <имя> vip portrange from <начало_диапазона>
set vs RAPID-UDP <имя> vip portrange to <конец_диапазона>
```

- для задания списка портов (каждый порт задается отдельной командой для добавления в список):

```
set vs RAPID-UDP <имя> vip portlist <порт>
```

- для задания всех портов:

```
set vs RAPID-UDP <имя> vip port 0
```

Если на Виртуальном Сервере настроен диапазон или список портов, а на Реальном Сервере указан конкретный порт, то трафик перенаправляется

на этот порт. Если на Реальном Сервере указаны все порты или порт не задан, то порт из запроса сохраняется и передается на Реальный Сервер без изменений.

- задание алгоритма определения статуса Виртуального Сервера:

Алгоритм определения статуса может быть:

- OR – статус Виртуального Сервера будет «В работе», если работает хотя бы один ассоциированный с ним Сервер Балансировки;
- AND – статус Виртуального Сервера будет «В работе», если работают все ассоциированные с ним Серверы Балансировки;
- NONE – не использовать никакой из алгоритмов.

```
set vs RAPID-UDP <имя> check-lbs algorithm <алгоритм>
```

- задание Сервера Балансировки (можно указать несколько узлов), по состоянию которого будет определяться статус Виртуального Сервера:

Статус Виртуального Сервера зависит от статуса ассоциированного с ним Сервера Балансировки. Алгоритм определения статуса задается в команде выше.

```
set vs RAPID-UDP <имя> check-lbs lbs-ids <имя>
```

- задание правила выбора Сервера Балансировки:

Параметр <приоритет> задает приоритет применения правила: чем ниже число, тем выше приоритет, и тем раньше правило будет обработано Виртуальным Сервером.

```
set vs RAPID-UDP <имя> netrules <приоритет>
```

- задание сети источника в формате CIDR для правила:

В правиле задается сеть источника запроса: в зависимости от того, из какой сети подключился пользователь, будет выбран тот или иной Сервер Балансировки.

```
set vs RAPID-UDP <имя> netrules <приоритет> network <сеть><префикс>
```

- задание Сервера Балансировки для правила:

```
set vs RAPID-UDP <имя> netrules <приоритет> lbs-id <имя>
```

- (опционально) привязка VRF к Виртуальному Серверу:

```
set vs RAPID-UDP <имя> vrf <имя_VRF>
```

- (опционально) активация возможности анонсирования IP-адреса (Route Health Injection – RHI), привязанного к Виртуальному Серверу, протоколам динамической маршрутизации:

Для работы RHI необходимо, чтобы динамическая маршрутизация была включена на Termidesk Connect (см. подраздел **Сеть**).

Параметру могут быть заданы значения:

- `ON` – активация RHI. При активации RHI Termidesk Connect будет анонсировать в сеть IP-адреса Виртуальных Серверов в зависимости от их режима работы;
- `OFF` – отключение RHI.

```
set vs RAPID-UDP <имя> rhi <значение>
```

- (опционально, только при активации RHI) задание режима работы RHI для Виртуального Сервера:

Параметру могут быть заданы значения:

- `ACTIVE` – активный режим RHI для Виртуального Сервера;
- `PASSIVE` – пассивный режим RHI для Виртуального Сервера.

Идентификатором маршрута являются пары «VRF – IP-адрес» (поддерживаются множественные VRF), при этом состояние маршрута определяется условиями:

- если все Виртуальные Серверы по данному маршруту находятся в режиме `PASSIVE`, то Termidesk Connect всегда будет объявлять маршрут для виртуального IP-адреса;
- если хотя бы один Виртуальный Сервер находятся в режиме `ACTIVE` и в состоянии «В работе», то Termidesk Connect будет объявлять маршрут для виртуального IP-адреса;
- в остальных случаях Termidesk Connect не будет объявлять маршрут.

```
set vs RAPID-UDP <имя> rhi-state <значение>
```

- (опционально) задание комментария, который будет привязан к Виртуальному Серверу:

Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.

```
set vs RAPID-UDP <имя> description <комментарий>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml vs RAPID-UDP <имя>
```

- просмотр выполненных команд:

```
show configuration cli vs RAPID-UDP <имя>
```

ГЕОБАЛАНСИРОВКА

ADNS

Общие сведения об ADNS

ADNS в Termidesk Connect – это объект (абстракция), позволяющий настроить Termidesk Connect в качестве авторитетного DNS-сервера. Для ADNS настраивается IP-адрес и порт для приема входящих DNS-запросов от пользователей или вышестоящих DNS-серверов.

Создание и настройка ADNS

Создание и настройка ADNS выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. ADNS**).

Для создания и настройки ADNS используются команды:

Описание параметров также приведено в подразделе **Объект** `geolb`.

- назначение IP-адреса и порта, на котором Termidesk Connect будет принимать входящие DNS-запросы:

```
set geolb adns <IP-адрес> <порт>
```

В случае использования отказоустойчивой конфигурации Termidesk Connect необходимо указывать общий IP-адрес кластера.

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml geolb
```

- просмотр выполненных команд:

```
show configuration cli geolb
```

Геолокационные IP-базы

Общие сведения о геолокационных IP-базах

Геолокационная IP-база (GeoIP) – это база данных, содержащая информацию по пулам IP-адресов, их координатам и наименованиям локаций. Используется для определения местоположения IP-адреса пользователя: страны, города, широты. долготы и др.

В Termidesk Connect такие IP-базы используются для активации определенного алгоритма балансировки и выбора ближайшей к пользователю Площадки.

Добавление геолокационных IP-баз

Для добавления собственной геолокационной IP-базы нужно:

- конвертировать основную геолокационную IP-базу в формат `JSON`, например, с помощью утилиты `mmdbctl`:

```
mmdbctl export <путь_к_базе_.mmdb> -o <путь_к_новому_файлу_базы_.json>
```

- сформировать текстовый файл (например, `geonames.txt`) для удобного поиска `geoname_id` с помощью командной строки:

```
jq -r '. a|
"(.continent.geoname_id),(.country.geoname_id),(.subdivisions[0].geoname_id),(.city
.geoname_id),(.location a|
"(.latitude),(.longitude),(.accuracy_radius)")a|(.continent.code)a|(.country.names
a| "(.ru):(.en)")a|(.subdivisions[0].names a| "(.ru):(.en)")a|(.city.names a|
"(.ru):(.en)");"' exported-geoip.json a| sort a| uniq > geonames.txt
```

- найти необходимые географические идентификаторы в базе данных, например:

```
grep -i 'Тюмень' geonames.txt grep -i 'vladivos' geonames.txt sed -e '/Санкт/I!d'
-e '/rus/I!d' geonames.txt
```

Идентификаторы находятся в первом столбце, если разделительный знак принять за «|». В этом же списке находятся координаты в системе WGS 84 (последние два значения), а также радиус точности местоположения в километрах. Координаты можно заменить, воспользовавшись онлайн-картой.

- сформировать файл (например, `infrastructure.txt`) с содержимым IP-адресов и их местоположений:

```
10.0.1.0/24,6255148,2017370,536203,498817,59.93863,30.31413,5
10.0.3.0/25,6255148,2017370,1488747,1488754,57.138530,65.522186,1
10.0.3.128/25,6255148,2017370,1488747,1488754,57.116799,65.550740,1
```

Где:

- значения в столбце 1 соответствуют подсети, для которой определяется местоположение;
- далее в столбцах 2 – 5 следуют идентификаторы: континента, страны, региона, города;
- затем в столбцах 6 и 7 приводятся координаты (широта, долгота);
- в столбце 8 приводится радиус точности в километрах.

Необходимые столбцы: 1, 6, 7.

- конвертировать получившийся файл в формат MMDB с помощью исполняемого файла `writer.py`:

Необходимо предварительно установить библиотеку `netaddr`:

```
sudo apt install python3-netaddr
```

```
python3 writer.py --file infrastructure.txt --output infrastructure.mmdb
```

Площадки

Площадки в Termidesk Connect – это группа серверов (или ЦОД) геораспределенной балансировки.

В текущей версии Termidesk Connect функционал не влияет на работу Termidesk Connect. Приведена справочная информация.

DNS View

Общие сведения о DNS View

DNS View в Termidesk Connect – это объект (абстракция), описывающий сети, при запросе из которых должен быть выдан локальный IP-адрес Сервиса (см. подраздел **Сервисы**).

DNS View может быть привязан к Виртуальному Серверу геобалансировки (см. подраздел **Виртуальные Серверы геобалансировки**), что позволит настроить поведение, при котором на запросы из определённой сети Сервис будет всегда отвечать локальным IP-адресом.

Создание и настройка DNS View

Создание и настройка DNS View выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. DNS View**).

Для создания и настройки DNS View используются команды:

Описание параметров также приведено в подразделе **Объект** `geolb`.

- создание DNS View:

```
set geolb view <имя>
```

- задание IP-адреса сети в формате CIDR:

```
set geolb view <имя> net <сеть><длина_префикса>
```

Пример команды:

```
set geolb view ViewForLocalNet net 172.16.0.0/16
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml geolb view <имя>
```

- просмотр выполненных команд:

```
show configuration cli geolb view <имя>
```

Сервисы

Общие сведения о Сервисах

Сервис в Termidesk Connect – это объект (абстракция) с назначенным IP-адресом, доступ к которому нужно предоставить пользователю в рамках геобалансировки. IP-адрес, заданный в Сервисе, помещается в DNS-ответ.

Можно задать два типа IP-адреса:

- общедоступный – IP-адрес назначается Сервису всегда, соответственно в DNS-ответах он будет помещаться всегда;
- локальный – IP-адрес назначается Сервису только в тех случаях, когда нужно обеспечить его функционирование также во внутренней сети организации.

В случае, если назначены оба IP-адреса:

- если к Сервису запросил доступ внешний пользователь, то в DNS-ответ будет помещен общедоступный IP-адрес;
- если к Сервису запросил доступ внутренний пользователь, сеть которого попадает в DNS View (см. подраздел **DNS View**), то в DNS-ответ будет помещен локальный IP-адрес.

Создание и настройка Сервиса

Создание и настройка Сервиса выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Сервисы**).

Для создания и настройки Сервиса используются команды:

Описание параметров также приведено в подразделе **Объект** `geolb`.

- создание Сервиса:

```
set geolb service <имя>
```

- (опционально) задание Проверки, которая будет выполняться для Сервиса:

Используются те же Проверки, которые были заданы для управления трафиком (см. подраздел **Проверки**).

```
set geolb service <имя> hc-id <имя>
```

- (опционально) задание локального IP-адреса Сервиса:

Для работы функционала DNS View (см. подраздел **DNS View**) задание IP-

адреса обязательно.

```
set geolb service <имя> local-ip <IP-адрес>
```

- задание общедоступного IP-адреса Сервиса:

```
set geolb service <имя> public-ip <IP-адрес>
```

- (опционально) задание Веса Сервиса:

Вес определяет приоритет Сервиса и устанавливается в диапазоне от 1 до 255, параметр необходим при балансировке с использованием алгоритма ROUNDROBIN.

Чем больше Вес Сервиса, тем чаще на этот Сервис идет перенаправление входящих подключений.

```
set geolb service <имя> weight <значение>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – XML, JSON или TXT):

```
show configuration xml geolb service <имя>
```

- просмотр выполненных команд:

```
show configuration cli geolb service <имя>
```

Виртуальные Серверы геобалансировки

Общие сведения о Виртуальных Серверах геобалансировки

Виртуальный Сервер геобалансировки в Termidesk Connect – это объект (абстракция), описывающий алгоритмы балансировки, по которым будут выдаваться IP-адреса Сервисов в DNS-ответах.

Создание и настройка Виртуальных Серверов геобалансировки

Создание и настройка Виртуального Сервера геобалансировки выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Виртуальные Серверы геобалансировки**).

Для создания и настройки Виртуального Сервера геобалансировки используются команды:

Описание параметров также приведено в подразделе **Объект** `geolb`.

- создание Виртуального Сервера геобалансировки:

```
set geolb vs <имя>
```

- задание алгоритма геобалансировки:

Могут быть заданы алгоритмы:

- `ONLINE` – Виртуальный Сервер включит в DNS-ответ все доступные (прошедшие проверку) IP-адреса;
- `ROUNDROBIN` – Виртуальный Сервер последовательно распределит подключения между Сервисами для их равномерного распределения;
- `SOURCEIPHASH` – Виртуальный Сервер будет перенаправлять подключения к Сервисам, основываясь на значении хеша IP-адреса. Это гарантирует, что подключение одного пользователя всегда будет направляться на один и тот же Сервис, снижая вероятность перегрузки;
- `STATICPROXIMITY` – Виртуальный Сервер перенаправит подключения на ближайшую Площадку для выдачи Сервиса. В этом случае учитывается местоположение исходного источника запроса (см. подраздел **Геолокационные IP-базы**).

```
set geolb vs <имя> algorithm <алгоритм>
```

- управление опцией `esc` (EDNS Client Subnet), может принимать значения `ON` (включена) или `OFF` (выключена):

Опция `ecs` используется для хранения информации о сети исходного источника запроса. Это полезно в случае, когда запрос пришел от вышестоящего DNS-сервера, местоположение которого точно неизвестно. Termidesk Connect получит информацию о сети исходного источника из входящего DNS-запроса и сохранит ее.

```
set geolb vs <имя> ecs <значение>
```

- задание типа привязки входящего подключения к одному Сервису на время обработки его DNS-запросов:

Параметру могут быть заданы значения:

- `OFF` – привязка к Сервису отключена;
- `SRCIP` – привязка входящего подключения к Сервису на основе его IP-адреса.

```
set geolb vs <имя> persistence <значение>
```

- задание привязки Сервиса к Виртуальному Серверу геобалансировки:

```
set geolb vs <имя> service-ids <имя>
```

- задание привязки DNS View к Виртуальному Серверу геобалансировки:

Привязка DNS View к Виртуальному Серверу геобалансировки позволит настроить поведение, при котором на запросы из определённой сети Сервис будет всегда отвечать локальным IP-адресом (см. подраздел **Сервисы**).

```
set geolb vs <имя> view-id <имя>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml geolb vs <имя>
```

- просмотр выполненных команд:

```
show configuration cli geolb vs <имя>
```

Зоны

Общие сведения о Зонах

Зона в Termidesk Connect – это объект, описывающий доменные имена, для которых обеспечивается геобалансировка подключений.

Создание и настройка Зоны

Создание и настройка Зоны выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см.подраздел **Веб. Зоны**).

Для создания и настройки Зоны используются команды:

Описание параметров также приведено в подразделе **Объект** `geolb`.

- создание Зоны:

```
set geolb zone <имя_домена>
```

- задание поддомена для указанного домена:

В командах с опцией `nodes` используется символ «.» для указания корневого домена. Для применения настроек домена третьего уровня нужно вместо точки использовать имя поддомена.

Пример задания поддомена connect для домена connect.termidesk.ru:

```
set geolb zone termidesk.ru nodes connect
```

```
set geolb zone <имя_домена> nodes .
```

- задание TTL для создаваемой зоны:

```
set geolb zone <имя_домена> nodes . ttl <значение>
```

- задание порядкового номера Виртуального Сервера, который будет использоваться для работы с доменом (значения указываются в порядке возрастания):

```
set geolb zone <имя_домена> nodes . vs <значение>
```

- задание Виртуального Сервера для указанного порядкового номера `vs`:

Можно указать несколько Виртуальных Серверов в Зоне для домена. Обработка запросов будет осуществляться в порядке возрастания порядкового номера Виртуального Сервера, указанного в параметре `vs`.

```
set geolb zone <имя_домена> nodes . vs <значение> vs-id <имя>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml geolb zone <имя_домена>
```

- просмотр выполненных команд:

```
show configuration cli geolb zone <имя_домена>
```

БЕЗОПАСНОСТЬ

Списки контроля доступа

Списки контроля доступа позволяют ограничить запросы к сетевым абстракциям, настроенным в Termidesk Connect: например, можно ограничить доступ к Виртуальным Серверам. Список контроля доступа привязывается к сетевому интерфейсу устройства (см. подраздел **Сеть**).

Создание и настройка списка контроля доступа выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Списки контроля доступа**).

Для создания и настройки списка контроля доступа используются команды:

Описание параметров также приведено в подразделе **Объект** `acl`.

- создание списка контроля доступа:

```
set acl list <имя>
```

- (опционально) настройка журналирования событий:

```
set acl list <имя> logging <значение>
```

Пример:

```
set acl list ACL_LIST logging true
```

- задание приоритета правила в списке контроля доступа:

```
set acl list <имя> rules <приоритет>
```

Пример:

```
set acl list ACL_LIST rules 2
```

- задание протокола, для которого создается правило:

```
set acl list <имя> rules <приоритет> proto <протокол>
```

Пример:

```
set acl list ACL_LIST rules 2 proto TCP
```

- задание типа действия при обработке правила:

```
set acl list <имя> rules <приоритет> action <действие>
```

Пример:

```
set acl list ACL_LIST rules 2 action ALLOW
```

- задание списка адресов назначения, для которых создается правило:

```
set acl list <имя> rules <приоритет> destination network <IP-адрес><префикс>
```

Пример:

```
set acl list ACL_LIST rules 2 destination network 192.168.0.2/24
```

- задание определенного порта назначения, для которого создается правило:

```
set acl list <имя> rules <приоритет> destination oneport <порт>
```

Пример:

```
set acl list ACL_LIST rules 2 destination oneport 80
```

Команда применяется для протоколов типа TCP и UDP.

- задание условия определения списка портов назначения, для которых создается правило:

```
set acl list <имя> rules <приоритет> destination compare <параметр> <значение>
```

Пример:

```
set acl list ACL_LIST rules 2 destination compare gt 1024
```

Команда применяется для протоколов типа TCP и UDP.

- задание диапазона портов назначения, для которых создается правило:

```
set acl list <имя> rules <приоритет> destination portrange from <порт>
set acl list <имя> rules <приоритет> destination portrange to <порт>
```

Пример:

```
set acl list ACL_LIST rules 2 destination portrange from 1000
set acl list ACL_LIST rules 2 destination portrange to 2000
```

Команда применяется для протоколов типа TCP и UDP.

- задание списка адресов источника запроса, для которых создается правило:

```
set acl list <имя> rules <приоритет> source network <IP-адрес><префикс>
```

Пример:

```
set acl list ACL_LIST rules 2 source network 192.168.100.0/24
```

- задание определенного порта источника запроса, для которого создается правило:

```
set acl list <имя> rules <приоритет> source oneport <порт>
```

Пример:

```
set acl list ACL_LIST rules 2 source oneport 80
```

Команда применяется для протоколов типа TCP и UDP.

- задание условия определения списка портов источника запроса, для которых создается правило:

```
set acl list <имя> rules <приоритет> source compare <параметр> <значение>
```

Пример:

```
set acl list ACL_LIST rules 2 source compare gt 1023
```

Команда применяется для протоколов типа TCP и UDP.

- задание диапазона портов источника запроса, для которых создается правило:

```
set acl list <имя> rules <приоритет> source portrange from <порт>
set acl list <имя> rules <приоритет> source portrange to <порт>
```

Пример:

```
set acl list ACL_LIST rules 2 source portrange from 1000
set acl list ACL_LIST rules 2 source portrange to 2000
```

Команда применяется для протоколов типа TCP и UDP.

- задание идентификатора VLAN, для которого создается правило:

```
set acl list <имя> rules <приоритет> vlan-id <идентификатор_VLAN>
```

Пример:

```
set acl list ACL_LIST rules 2 vlan-id 114
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

- просмотр заданных настроек (указывается формат вывода – `XML`, `JSON` или `TXT`):

```
show configuration xml acl list <имя>
```

- просмотр выполненных команд:

```
show configuration cli acl list <имя>
```

DDoS-Профили

Для возможности использования правил доступа и защиты от различных вариантов DDoS-атак для публикуемых ресурсов настраивается DDoS-Профиль Termidesk Connect.

Создание и настройка DDoS-Профиля выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. DDoS-Профили**).

Для добавления DDoS-Профиля выполнить:

Описание параметров также приведено в подразделе **Объект** `ddos`.

- указание имени DDoS-Профиля:

```
set ddos profile <имя>
```

- настройка защиты от ICMP-атаки:
  - включение или выключение защиты от ICMP-атаки:

    ```
    set ddos profile <имя> icmp-flood enable <true/false>
    ```

  - указание порогового значения ICMP-пакетов на интерфейсе для применения правила (по умолчанию – `0`):

    ```
    set ddos profile <имя> icmp-flood interface-limit threshold
    <пороговое_значение_на интерфейсе>
    ```

  - указание гистерезиса порогового значения на интерфейсе, т.е. на сколько должен снизиться трафик для выключения защиты (по умолчанию – `0`):

    ```
    set ddos profile <имя> icmp-flood interface-limit hysteresis
    <гистерезис_на_интерфейсе>
    ```

- указание порогового значения ICMP-пакетов от одного IP-адреса для применения правила (по умолчанию – 0):

```
set ddos profile <имя> icmp-flood ip-limit threshold <пороговое_значение_IP-адреса>
```

- указание гистерезиса порогового значения от одного IP-адреса, т.е. на сколько должен снизиться трафик для выключения защиты (по умолчанию – 0):

```
set ddos profile <имя> icmp-flood ip-limit hysteresis <гистерезис_IP-адреса>
```

- настройка защиты от SYN-атаки:
  - включение или выключение защиты от SYN-атаки:

```
set ddos profile <имя> syn-flood enable <true/false>
```

  - указание порогового значения SYN-пакетов на интерфейсе для применения правила (по умолчанию – 0):

```
set ddos profile <имя> syn-flood interface-limit threshold <пороговое_значение_на интерфейсе>
```

  - указание гистерезиса порогового значения на интерфейсе, т.е. на сколько должен снизиться трафик для выключения защиты (по умолчанию – 0):

```
set ddos profile <имя> syn-flood interface-limit hysteresis <гистерезис_на_интерфейсе>
```

  - указание порогового значения SYN-пакетов от одного IP-адреса для применения правила (по умолчанию – 0):

```
set ddos profile <имя> syn-flood ip-limit threshold <пороговое_значение_IP-адреса>
```

  - указание гистерезиса порогового значения от одного IP-адреса, т.е. на сколько должен снизиться трафик для выключения защиты (по умолчанию – 0):

```
set ddos profile <имя> syn-flood ip-limit hysteresis <гистерезис_IP-адреса>
```

- настройка защиты от UDP-атаки:
  - включение или выключение защиты от UDP-атаки:

```
set ddos profile <имя> udp-flood enable <true/false>
```

- указание порогового значения UDP-пакетов на интерфейсе для применения правила (по умолчанию – 0):

```
set ddos profile <имя> udp-flood interface-limit threshold
<пороговое_значение_на интерфейсе>
```

- указание гистерезиса порогового значения на интерфейсе, т.е. на сколько должен снизиться трафик для выключения защиты (по умолчанию – 0):

```
set ddos profile <имя> udp-flood interface-limit hysteresis
<гистерезис_на_интерфейсе>
```

- указание порогового значения UDP-пакетов от одного IP-адреса для применения правила (по умолчанию – 0):

```
set ddos profile <имя> udp-flood ip-limit threshold <пороговое_значение_ip-
адреса>
```

- указание гистерезиса порогового значения от одного IP-адреса, т.е. на сколько должен снизиться трафик для выключения защиты (по умолчанию – 0):

```
set ddos profile <имя> udp-flood ip-limit hysteresis <гистерезис_IP-адреса>
```

- включение или выключение журналирования при обнаружении или завершении атаки:

```
set ddos profile <имя> logging <true/false>
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

- просмотр настроенных DDoS-Профилей:

```
show ddos profile all
```

- просмотр статистики защиты от DDoS-атак:

```
show ddos detection all
```

AAA

Общие сведения об AAA

Конфигурация AAA описывает настройку аутентификации пользователя Termidesk Connect со стороны клиента и способа передачи данных Реальному Серверу, в том числе с использованием внешних AAA-сервисов.

В конфигурацию AAA входят настройки компонентов:

- AAA-Сервер – совокупность параметров взаимодействия с сервером аутентификации и авторизации;
- AAA-Профиль – совокупность параметров аутентификации пользователей, таких как количество попыток входа, порядок выбора серверов аутентификации и другое;
- KDC-Сервер – совокупность параметров взаимодействия с сервером аутентификации и авторизации KDC (Kerberos-аутентификация);
- SSO-Профиль – совокупность параметров перенаправления учетной записи пользователя на Реальный Сервер.

Kerberos-аутентификация

Termidesk Connect поддерживает Kerberos-аутентификацию пользователя на Реальном Сервере.

При использовании Kerberos-аутентификации Termidesk Connect имперсонализирует пользователя, являясь прокси-сервисом, и запрашивает билет для подключения у сервера KDC. Это происходит по следующему алгоритму:

- пользователь подключается к Termidesk Connect;
- Termidesk Connect использует имя делегирующей службы (service principal, формат: `<протокол>/<имя службы>`:<port>@<realm>) для имперсонализации пользователя (происходит первый запрос TGS-билета у сервера KDC для делегирующей службы);
- дальнейшее подключение пользователя к Реальному Серверу происходит следующим образом:
  - Termidesk Connect запрашивает TGS-билет (второй запрос для делегирования) у сервера KDC для подключения пользователя к Реальному Серверу;
  - Termidesk Connect формирует на основе полученного от сервера KDC билета GSS-токен;
  - Termidesk Connect подключает пользователя к Реальному Серверу.

У TGS-билетов существует срок хранения (по умолчанию 24 часа) в кеше библиотеки Kerberos (тип `MEMORY` – в оперативной памяти). Таким образом первый запрос TGS-билета у сервера KDC не происходит при каждом подключении пользователя. Однако второй запрос TGS-билета выполняется при подключении пользователя, если его сессия перестала быть актуальной.

AAA-Серверы

Создание и настройка AAA-Сервера выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. AAA**).

Для добавления AAA-Сервера выполнить:

Описание параметров также приведено в подразделе **Объект** `aaa`.

- создание AAA-Сервера:

```
set aaa server <имя>
```

- указание типа службы каталогов, к которой выполняется подключение:

Возможные значения:

- `AD` – подключение к службе каталогов с поддержкой Microsoft Active Directory;
- `OpenLDAP` – подключение к службе каталогов с поддержкой реализации с открытым исходным кодом протокола LDAP – OpenLDAP.

```
set aaa server <имя> LDAP type <тип>
```

- указание адреса для подключения к службе каталогов:

```
set aaa server <имя> LDAP domain <адрес>
```

- указание порта (по умолчанию – `389`):

```
set aaa server <имя> LDAP port <порт>
```

- указание времени (в секундах) ожидания авторизации (по умолчанию – `30`):

```
set aaa server <имя> LDAP timeout <время>
```

- указание корневой точки службы каталогов:

Пример значения параметра: `DC=myserver,DC=local`.

```
set aaa server <имя> LDAP base-dn <корневая_точка>
```

- указание полного имени учетной записи администратора службы каталогов:

Пример значения параметра: `CN=Administrator,CN=Users,DC=example,DC=ru`.

Так же имя может быть получено из Хранилища секретов. Пример значения параметра:

- `kv://secret/data/my_ldap#dn` – для `kv` версии 2;
- `ad://ldap/static-cred/my_ldap#dn` – для `ldap`.

```
set aaa server <имя> LDAP administrator-bind-dn <имя_учетной_записи>
```

- указание пароля для соединения со службой каталогов:

Так же пароль может быть получен из Хранилища секретов. Пример значения параметра:

- `kv://secret/data/my_ldap#password` – для `kv` версии 2;
- `ad://ldap/static-cred/my_ldap#last_password` – для `ldap`.

```
set aaa server <имя> LDAP password <пароль>
```

- указание атрибута группы:

Пример значения параметра: `memberOf`.

```
set aaa server <имя> LDAP group-attribute <атрибут_группы>
```

- (опционально) указание Клиентского SSL-Профиля для подключения:

```
set aaa server <имя> LDAP ssl-profile-id <имя_профиля>
```

- (опционально) указание списка атрибутов, которые должны быть импортированы из службы каталогов:

```
set aaa server <имя> LDAP import-attribute-list <список_атрибутов>
```

- (опционально) указание атрибута имени пользователя (по умолчанию – `userPrincipalName`):

```
set aaa server <имя> LDAP user-name-attribute <атрибуты>
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

AAA-Профили

Создание и настройка AAA-Профиля выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. AAA**).

Для добавления AAA-Профиля выполнить:

Описание параметров также приведено в подразделе **Объект** `aaa`.

- создание AAA-Профиля:

```
set aaa profile <имя>
```

- включение или выключение аутентификации пользователя (по умолчанию – `true`):

Возможные значения:

- `true` – выполняется аутентификация пользователя с проверкой учетных данных. Пользователь должен предоставить корректное имя и пароль;
- `false` – проверка пароля не выполняется. Для запроса атрибутов пользователя у сервера аутентификации используется только его имя (например, полученное из SSL-сертификата).

```
set aaa profile <имя> authentication <true/false>
```

- указание пространства имен пользователя для аутентификации:

```
set aaa profile <имя> realm <пространство_имен>
```

- указание времени (в секундах) жизни аутентификационной cookie (значение от 1 до 86400, по умолчанию – `600`):

```
set aaa profile <имя> cookie-ttl <время_жизни>
```

- настройка параметров блокирования пользователя при неуспешной аутентификации:

Если пользователь в течение в течение времени `attempt-ttl` сделает неудачные попытки аутентификации в количестве `attempts`, то он будет заблокирован на время `blocking-ttl`.

- указание периода (в секундах), в течение которого подсчитываются неудачные попытки аутентификации (значение от 1 до 86400, по умолчанию – 600):

```
set aaa profile <имя> attempt-ttl <период>
```

- указание количества неудачных попыток аутентификации (значение от 1 до 255, по умолчанию – 3):

```
set aaa profile <имя> attempts <количество_попыток>
```

- указание времени (в секундах) до блокировки (значение от 1 до 86400, по умолчанию – 600):

```
set aaa profile <имя> blocking-ttl <время_блокировки>
```

- указание имени аутентификационной cookie (по умолчанию – TC-AAC):

Имя аутентификационной cookie – это уникальный идентификатор cookie, который Termidesk Connect использует для поиска и валидации сессии.

```
set aaa profile <имя> cookie-name <имя_cookie>
```

- указание атрибутов аутентификационной cookie:

Пример атрибутов: SameSite=Strict; HttpOnly.

```
set aaa profile <имя> cookie-attr <атрибуты_cookie>
```

- настройка привязки AAA-Сервера:

Для одного AAA-Профиля может быть задано несколько AAA-Серверов. В случае недоступности одного AAA-Сервера осуществляется переход на другой.

- указание порядкового номера привязки настроенного AAA-Сервера:

```
set aaa profile <имя> servers <порядковый_номер>
```

- указание имени настроенного AAA-Сервера:

```
set aaa profile <имя> servers <порядковый_номер> server-id <AAA-Сервер>
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

KDC-Серверы

KDC-Серверы представляют собой совокупность параметров взаимодействия с сервером аутентификации и авторизации KDC (Kerberos-аутентификация).

Создание и настройка KDC-Сервера выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. AAA**).

Для добавления KDC-Сервера выполнить:

Описание параметров также приведено в подразделе **Объект** `krb5`.

- создание KDC-Сервера:

```
set krb5 server <имя>
```

- указание IP-адреса для подключения к серверу:

```
set krb5 server <имя> domain <IP-адрес>
```

- указание порта для подключения к серверу (по умолчанию – `88`):

```
set krb5 server <имя> port <порт>
```

- (опционально) указание времени (в секундах) ожидания ответа от сервера KDC (по умолчанию – `30`):

```
set krb5 server <имя> timeout <значение>
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

SSO-Профили

Создание и настройка SSO-Профиля выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. AAA**).

Для добавления SSO-Профиля выполнить:

Описание параметров также приведено в подразделе **Объект** `aaa`.

- создание SSO-Профиля:

```
set aaa sso-profile <имя>
```

- указание типа аутентификации (т.е. как данные пользователя передаются на Реальный Сервер):

Возможные значения типа аутентификации:

- `BASIC` – базовый алгоритм аутентификации;
- `OFF` – данные аутентификация не передаются;
- `KRB5-IM` – аутентификация с использованием Kerberos версии 5.

```
set aaa sso-profile <имя> <тип_аутентификации>
```

- при выборе типа аутентификации `KRB5-IM` потребуется задать:
  - пространство имен пользователя для аутентификации:

    ```
    set aaa sso-profile <имя> KRB5-IM realm <значение>
    ```

  - имя делегирующего пользователя, обладающего правом от своего имени запрашивать билеты для обычных пользователей и делегировать их на другие сервисы:

    

    Так же имя может быть получено из Хранилища секретов. Пример значения параметра:

    - `kv://secret/kerberos#service_principal` – для `kv` версии 1;
    - `kv://secret/data/kerberos#service_principal` – для `kv` версии 2.

    ```
    set aaa sso-profile <имя> KRB5-IM service-principal <имя>
    ```

- пароль делегирующего пользователя:

Так же пароль может быть получен из Хранилища секретов. Пример значения параметра:

- `kv://secret/kerberos#password` – для `kv` версии 1;
- `kv://secret/data/kerberos#password` – для `kv` версии 2.

```
set aaa sso-profile <имя> KRB5-IM password <пароль>
```

- KDC-Сервер для аутентификации из списка добавленных в Termidesk Connect (может быть указано несколько) и его приоритет:

```
set aaa sso-profile <имя> KRB5-IM servers <приоритет> <имя>
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

Профили ограничения скорости

Общие сведения о Профилях ограничения скорости

Для возможности ограничения нагрузки на Виртуальные Серверы (TCP или HTTP), Серверы Балансировки (TCP или HTTP) или Реальные Серверы настраивается Профиль ограничения скорости.

Профиль ограничения скорости - совокупность параметров для ограничения трафика. Профиль может переиспользоваться на:

- Виртуальных серверах (см. подраздел Создание и настройка Профиля ограничения скорости для Виртуального Сервера);
- Серверах Балансировки (см. подраздел Создание и настройка Профиля ограничения скорости для Сервера Балансировки);
- Реальных Серверах (см. подраздел Создание и настройка Профиля ограничения скорости для Реального Сервера).

Для Виртуальных Серверов и Серверов Балансировки ограничивается только входящий трафик. Для Реальных Серверов ограничивается входящий и исходящий трафик, ограничения пороговых значений действуют на каждый Реальный Сервер в отдельности.

Создание и настройка Профиля ограничения скорости для Виртуального Сервера

Создание и настройка Профиля ограничения скорости для Виртуального Сервера

выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Профили ограничения скорости**).

Для создания и настройки Профиля ограничения скорости для Виртуального Сервера используются команды:

Описание параметров также приведено в подразделе **Объект** `rl-profile`.

- создание Профиля ограничения скорости для Виртуального Сервера:

```
set rl-profile vs <имя>
```

- настройка ограничения скорости передачи данных (битов в секунду) от пользователя к Виртуальному Серверу:
  - задание порога скорости:

    ```
    set rl-profile vs <имя> in bandwidth rate <значение>
    ```

  - задание порога всплеска:

    ```
    set rl-profile vs <имя> in bandwidth burst <значение>
    ```

- настройка ограничения количества пакетов в секунду от пользователя к Виртуальному Серверу:
  - задание порога скорости:

    ```
    set rl-profile vs <имя> in pps rate <значение>
    ```

  - задание порога всплеска:

    ```
    set rl-profile vs <имя> in pps burst <значение>
    ```

- настройка скорости установленных TCP-сессий (сессий в секунду) от пользователя к Виртуальному Серверу:

При превышении заданного количества TCP-сессий новые сессии будут отброшены.

- задание порога скорости:

```
set rl-profile vs <имя> in tcp-conn rate <значение>
```

- задание порога всплеска:

```
set rl-profile vs <имя> in tcp-conn burst <значение>
```

  - (опционально) задание комментария, который будет привязан к Профилю ограничения скорости для Виртуального Сервера:

Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.

```
set rl-profile vs <имя> description <комментарий>
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

Создание и настройка Профиля ограничения скорости для Сервера Балансировки

Создание и настройка Профиля ограничения скорости для Сервера Балансировки выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Профили ограничения скорости**).

Для создания и настройки Профиля ограничения скорости для Сервера Балансировки используются команды:

Описание параметров также приведено в подразделе **Объект** `rl-profile`.

- создание Профиля ограничения скорости для Сервера Балансировки:

```
set rl-profile lbs <имя>
```

- настройка ограничения скорости передачи данных (битов в секунду) от пользователя к Серверу Балансировки:
  - задание порога скорости:

```
set rl-profile lbs <имя> in bandwidth rate <значение>
```

- задание порога всплеска:

```
set rl-profile lbs <имя> in bandwidth burst <значение>
```

- настройка ограничения количества пакетов в секунду от пользователя к Серверу Балансировки:
  - задание порога скорости:

```
set rl-profile lbs <имя> in pps rate <значение>
```

  - задание порога всплеска:

```
set rl-profile lbs <имя> in pps burst <значение>
```

- (опционально) задание комментария, который будет привязан к Профилю ограничения скорости для Сервера Балансировки:

Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.

```
set rl-profile lbs <имя> description <комментарий>
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

Создание и настройка Профиля ограничения скорости для Реального Сервера

Создание и настройка Профиля ограничения скорости для Реального Сервера выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Профили ограничения скорости**).

Для создания и настройки Профиля ограничения скорости для Реального Сервера используются команды:

Описание параметров также приведено в подразделе **Объект** `rl-profile`.

- создание Профиля ограничения скорости для Реального Сервера:

```
set rl-profile rs <имя>
```

- настройка ограничения скорости получения данных (битов в секунду) от Реального Сервера:
  - задание порога скорости:

    ```
    set rl-profile rs <имя> in bandwidth rate <значение>
    ```

  - задание порога всплеска:

    ```
    set rl-profile rs <имя> in bandwidth burst <значение>
    ```

- настройка ограничения количества пакетов в секунду от Реального Сервера:
  - задание порога скорости:

    ```
    set rl-profile rs <имя> in pps rate <значение>
    ```

  - задание порога всплеска:

    ```
    set rl-profile rs <имя> in pps burst <значение>
    ```

- настройка ограничения скорости передачи данных (битов в секунду) от Termidesk Connect к Реальному Серверу:
  - задание порога скорости:

    ```
    set rl-profile rs <имя> out bandwidth rate <значение>
    ```

  - задание порога всплеска:

    ```
    set rl-profile rs <имя> out bandwidth burst <значение>
    ```

- настройка количества пакетов в секунду от Termidesk Connect к Реальному Серверу:
  - задание порога скорости:

    ```
    set rl-profile rs <имя> out pps rate <значение>
    ```

  - задание порога всплеска:

```
set rl-profile rs <имя> out pps burst <значение>
```

- (опционально) задание комментария, который будет привязан к Профилю ограничения скорости для Реального Сервера:

Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.

```
set rl-profile lbs <имя> description <комментарий>
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

Хранилище секретов

Общие сведения о Хранилище секретов

Termidesk Connect позволяет получать секреты из хранилищ на базе OpenBao или HashiCorp Vault.

Хранилище секретов представляет собой сервер со специализированным ПО, предназначенным для безопасного хранения чувствительной информации: ключей шифрования, логинов, паролей, токенов, сертификатов и т. д.

Termidesk Connect предоставляет возможность настройки параметров подключения к Хранилищу секретов, например к OpenBao, Hashicorp Vault или другому.

Termidesk Connect поддерживает следующие типы Хранилищ секретов:

- `kv` – статическое хранилище. Возвращает значение, соответствующее указанному ключу. Используется для хранения статических паролей, API-ключей, токенов и конфигурационных файлов;
- `ad` – AD/LDAP. Позволяет создавать временные учетные записи. Используется для управления динамическими учетными данными (LDAP) и данными аутентификации (AD);
- `pki` – PKI. Используется генерации TLS/SSL-сертификатов по запросу. Каждый успешный запрос на выдачу или подпись создает новый уникальный сертификат с собственным закрытым ключом. Ранее выданные сертификаты продолжают действовать до истечения их времени жизни.

Для получения данных из Хранилищ секретов используется специально сформированный URL в следующем формате:

```
<тип>://<путь>[?<параметры>]#<секрет>
```

где:

- `<тип>` – тип Хранилища секретов (`kv`, `ad` или `pki`);
- `<путь>` – путь к секрету;
- `<параметры>` – опциональные параметры запроса;
- `<секрет>` – имя запрашиваемого секрета.

Примеры формирования URL

Пример 1. Получение секрета из Хранилища секретов `kv` версии 1.

Если секрет создан командами:

```
secrets enable -version=1 -path=legacy kv
vault kv put legacy/snmp username="snm-user" auth_key="authkey" priv_key="privkey"
```

Выполнить HTTP-запрос для чтения секрета:

```
curl -H "X-Vault-Token: root-token" http://192.0.2.1:8200/v1/legacy/snmp
```

Ответ в формате `JSON`:

```
{
    "request_id": "780cb833-32b7-2dfa-1699-51027d56b542",
    "data": {
        "auth_key": "snmp-authkey",
        "priv_key": "snm-privkey",
        "username": "snm-user2"
    }
}
```

Правила формирования URL на основе запроса:

- определить схему: `kv://`;
- извлечь путь из URL HTTP-запроса `http://192.0.2.1:8200/v1/legacy/snmp`, исключив базовую часть `http://192.0.2.1:8200/v1/`: `legacy/snmp`;
- добавить к схеме `kv://` полученный путь `legacy/snmp`: `kv://legacy/snmp`;
- указать имя требуемого ключа через символ `#`:
  - для ключа `username`: `kv://legacy/snmp#username`;
  - для ключа `auth_key`: `kv://legacy/snmp#auth_key`;
  - для ключа `priv_key`: `kv://legacy/snmp#priv_key`.

Пример 2. Получение секрета из Хранилища секретов `kv` версии 2.

Если секрет создан командой:

```
vault kv put secret/snmp username="snm-user" auth_key="authkey" priv_key="privkey"
```

Выполнить HTTP-запрос для чтения секрета:

```
curl -H "X-Vault-Token: root-token" http://192.0.2.1:8200/v1/secret/data/snmp
```

Ответ:

```
{
    "request_id": "780cb833-32b7-2dfa-1699-51027d56b542",
    "data": {
        "data": {
            "auth_key": "snmp-authkey",
            "priv_key": "snm-privkey",
            "username": "snm-user2"
        }
    }
}
```

Правила формирования URL на основе запроса:

Правила аналогичны `kv` версии 1, отличие только в пути.

- определить схему: `kv://`;
- извлечь путь из URL HTTP-запроса `http://192.0.2.1:8200/v1/secret/data/snmp`, исключив базовую часть `http://192.0.2.1:8200/v1/`: `secret/data/snmp`;
- добавить к схеме `kv://` полученный путь `secret/data/snmp`: `kv://secret/data/snmp`;
- указать имя требуемого ключа через символ `#`:
  - для ключа `username`: `kv://secret/data/snmp#username`;
  - для ключа `auth_key`: `kv://secret/data/snmp#auth_key`;
  - для ключа `priv_key`: `kv://secret/data/snmp#priv_key`.

Пример 3. Получение сертификатов из Хранилища секретов `pki`.

Если PKI и роль настроены через команды:

```
# Включение PKI
vault secrets enable pki

# Создание роли для выпуска сертификатов
vault secrets enable pki
    vault write pki/roles/example-dot-com \
    allowed_domains="example.com" \
```

```
    allow_subdomains=true \
    allow_bare_domains=true \
    max_ttl=72h
```

Выполнить HTTP-запрос для генерации сертификата:

В теле HTTP-запроса передаются параметры сертификата в формате JSON: `{"common_name": "app.example.com", "ttl": "24h"}`.

```
curl -H "X-Vault-Token: token" \
-d '{"common_name": "app.example.com", "ttl": "24h"}' \
http://192.0.2.1:8200/v1/pki/issue/example-dot-com
```

Ответ:

```
{
    "request_id": "e86ed1f5-f7bb-1f6a-a864-6a5540c92191",
    "lease_id": "",
    "renewable": false,
    "lease_duration": 0,
        "data": {
        "certificate": "-----BEGIN CERTIFICATE....",
        "issuing_ca": "-----BEGIN CERTIFICATE-----...",
        "private_key": "-----BEGIN RSA PRIVATE KEY-----..."
    },
    "wrap_info": null,
    "warnings": null,
    "auth": null,
    "mount_type": "pki"
}
```

Правила формирования URL на основе запроса:

- определить схему: `pki://`;
- извлечь путь из URL HTTP-запроса `http://192.0.2.1:8200/v1/pki/issue/example-dot-com`, исключив базовую часть `http://192.0.2.1:8200/v1/`: `pki/issue/example-dot-com`;
- добавить к схеме `pki://` полученный путь `pki/issue/example-dot-com`: `kv://secret/data/snmp`;
- добавить параметры из тела запроса после символа `?`: `pki://pki/issue/example-dot-com?common_name="app.example.com"&ttl="24h"`;

Из JSON-формата `{"common_name": "app.example.com", "ttl": "24h"}` формируется строка параметров: `common_name="app.example.com"&ttl="24h"`.

- указать конкретный сертификат или ключ для получения через символ `#`:
  - для сертификата:

```
pki://pki/issue/example-dot-
com?common_name="app.example.com"&ttl="24h"#certificate
```

- для закрытого ключа:

```
pki://pki/issue/example-dot-
com?common_name="app.example.com"&ttl="24h"#private_key
```

- для сертификата УЦ:

```
pki://pki/issue/example-dot-
com?common_name="app.example.com"&ttl="24h"#issuing_ca
```

Настройка подключения к Хранилищу секретов

Настройка подключения к Хранилищу секретов выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Хранилище секретов**).

Для настройки Хранилища секретов выполнить:

Описание параметров также приведено в подразделе **Объект** `openbao`.

- указание URL-адреса для Хранилища секретов:

Пример значения параметра: `http://192.0.2.10:8200`

```
set openbao url <URL-адрес>
```

- указание VRF для подключения к Хранилищу секретов (по умолчанию – `default`):

```
set openbao vrf <имя_VRF>
```

- задание максимально допустимого времени (в секундах) ожидания ответа HTTP-запроса от API-сервера (по умолчанию – `60`):

```
set openbao request-timeout <значение>
```

- задание интервала (в секундах) повторной отправки HTTP-запроса при возникновении ошибки (по умолчанию – `10`):

```
set openbao retry-interval <значение>
```

- задание интервала (в секундах) обновления токена авторизации (по умолчанию – `3600`):

```
set openbao renew-token-interval <значение>
```

- задание интервала (в секундах) обновления секретов (по умолчанию – `1800`):

Termidesk Connect с заданной периодичностью будет отправлять запросы в Хранилище секретов для получения новых значений секретов.

```
set openbao renew-secret-interval <значение>
```

- задание интервала (в секундах) обновления сертификатов (по умолчанию – `86400`):

Termidesk Connect с заданной периодичностью будет отправлять запросы в Хранилище секретов для получения новых значений сертификатов.

```
set openbao renew-cert-interval <значение>
```

- (опционально) указание пространства имен (Namespace), заданного в Хранилище секретов:

```
set openbao ns <значение>
```

- настройка параметров аутентификации Termidesk Connect в Хранилище секретов:
  - указание пути к методу аутентификации (auth method) в Хранилище секретов:

    ```
    set openbao auth mount <значение>
    ```

  - задание идентификатора роли (RoleID) для метода аутентификации AppRole:

    ```
    set openbao auth approle <значение>
    ```

  - указание пути к файлу, содержащему идентификатор секрета (SecretID) или токен для его раскрытия (unwrapping):

    ```
    set openbao auth secret-file <значение>
    ```

  - указание флага использования упакованного токена (wrapped-token) для идентификатора секрета (SecretID) (по умолчанию – `true`):

Возможные значения:

- `true` – содержимое файла будет предварительно раскрыто (unwrapped) для получения действительного идентификатора

секрета (SecretID);

- `false` – содержимое файла будет упаковано (wrapped) для получения действительного идентификатора секрета (SecretID).

```
set openbao auth wrapping-token <true/false>
```

- (опционально) настройка параметров SSL-подключения к Хранилищу конфигурации:
  - задание файла сертификата УЦ (указывается полный путь и имя файла с расширением):

Пример значения параметра: `/dev/shm/server.cert`.

```
set openbao ssl ca-cert <имя_файла>
```

  - задание файла сертификата (указывается полный путь и имя файла с расширением):

```
set openbao ssl cert <имя_файла>
```

  - задание файла ключа (указывается полный путь и имя файла с расширением):

```
set openbao ssl key <имя_файла>
```

- применение заданных настроек:

```
commit
```

- сохранение настроек:

```
write
```

ШЛЮЗ

Точки подключений

Настройка точек подключений позволяет регистрировать Шлюз на разных Фермах Termidesk (Termidesk VDI).

Настройка точек подключения для Шлюза выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Точки подключений**).

При работе Шлюза совместно с Виртуальным Сервером (т.е. одновременным использованием и функционала Шлюза, и функционала балансировки) следует учесть особенности настройки:

- на Шлюзе должен отсутствовать Серверный SSL-Профиль. В то же время у Виртуального Сервера он должен быть задан. Это нужно, поскольку Виртуальный Сервер является точкой входа для пользовательских подключений, и при такой конфигурации он будет принимать входящие зашифрованные соединения, расшифровывать их и передавать уже открытый трафик на Шлюз;
- рекомендуется исключить возможность прямого обращения к Шлюзу из внешней сети, так как он обрабатывает незашифрованный трафик. Для этого в настройках Шлюза необходимо указать, чтобы он прослушивал подключения только на локальном интерфейсе (например, 127.0.0.1 или 127.0.0.2 и т.д.).

Для добавления точки подключения выполнить:

Описание параметров также приведено в подразделе **Объект** `gw`.

- указание имени Шлюза:

```
set gw server <имя>
```

- указание IP-адреса Виртуального Сервера:

```
set gw server <имя> vip ip <IP-адрес>
```

- указание порта Виртуального Сервера:

```
set gw server <имя> vip port <порт>
```

- указание существующего Серверного SSL-Профиля:

```
set gw server <имя> ssl-profile-id <SSL-Профиль>
```

- указание существующего VRF (по умолчанию – `default`):

```
set gw server <имя> vrf <имя_VRF>
```

- (опционально) настройка поддержки протокола UDP для протокола TERA на Шлюзе:
  - указание IP-адреса Виртуального Сервера для TERA:

IP-адрес Виртуального Сервера для TERA может принимать любое значение, т.е. IP-адрес может отличаться от IP-адреса Шлюза или может быть таким же. Указывается IP-адрес именно Виртуального Сервера, если Шлюз скрывается за Виртуальным Сервером.

```
set gw server <имя> tera-udp vip ip <IP-адрес_TERA>
```

  - указание порта Виртуального Сервера для TERA (по умолчанию – 443):

```
set gw server <имя> tera-udp vip port <порт_TERA>
```

- указание VRF для TERA (по умолчанию – `default`):

```
set gw server <имя> tera-udp vrf <имя_VRF_TERA>
```

- указание IP-адреса источника, т.е. адреса, с которого Шлюз устанавливает UDP-соединение с рабочим местом Termidesk:

Значение `0.0.0.0` указывает на использование всех IP-адресов.

```
set gw server <имя> tera-udp src ip <IP-адрес_источника>
```

- указание порта источника (по умолчанию – `0`):

```
set gw server <имя> tera-udp src port <порт_источника>
```

- указание VRF источника (по умолчанию – `default`):

```
set gw server <имя> tera-udp src vrf <имя_VRF_источника>
```

- настройка взаимодействия с компонентом «Универсальный диспетчер» Termidesk:
  - указание URL, поддерживаемого компонентом «Универсальный диспетчер» Termidesk с ролью «Портал пользователя» для обслуживания API-запросов по валидации подключений, запрашиваемых через Шлюз:

Следует использовать актуальные значения версий API для соответствующей версии компонента «Универсальный диспетчер» Termidesk.

```
set gw server <имя> checktoken url <URL>
```

  - указание существующего IP-Фонда, используемого для взаимодействия с компонентом «Универсальный диспетчер» Termidesk:

```
set gw server <имя> checktoken ipset-id <IP-Фонд>
```

  - указание существующего Клиентского SSL-Профиля (по умолчанию – `backend-default`):

```
set gw server <имя> checktoken ssl-profile-id <SSL-Профиль>
```

- настройка переподключения Шлюза к рабочим местам Termidesk:
  - указание количества разрешенных переподключений Шлюза к рабочим местам Termidesk (значение от 0 до 10, по умолчанию – `0`);

Разные протоколы удаленного доступа могут требовать разных значений параметра:

- для RDP нужно использовать ненулевое значение, поскольку некоторые функции (например, перенаправление USB в гостевую ОС рабочего места по протоколу RDP) могут требовать переподключения;
- для TERA и SPICE нужно использовать значение по умолчанию (`0`).

```
set gw server <имя> tcp-downstream reconnect <количество_переподключений>
```

  - указание существующего IP-фонда для взаимодействия с рабочими местами Termidesk:

```
set gw server <имя> tcp-downstream ipset-id <IP-Фонд>
```

  - указание существующего Клиентского SSL-Профиля для взаимодействия с рабочими местами Termidesk (по умолчанию – `backend-default`):

```
set gw server <имя> tcp-downstream ssl-profile-id <SSL-Профиль>
```

- настройка клиента RabbitMQ:
  - указание URL-адреса для подключения к серверу RabbitMQ:

```
set gw server <имя> rabbitmq url <URL-адрес>
```

  - указание имени пользователя для подключения к серверу RabbitMQ:

Так же имя пользователя может быть получено из Хранилища секретов. Пример значения параметра:

- `kv://secret/rabbit#username` – для `kv` версии 1;
- `kv://secret/data/rabbit#username` – для `kv` версии 2.

```
set gw server <имя> rabbitmq user <имя>
```

  - указание пароля пользователя для подключения к серверу RabbitMQ:

Так же пароль пользователя может быть получен из Хранилища секретов. Пример значения параметра:

- `kv://secret/rabbit#password` – для `kv` версии 1;

- `kv://secret/data/rabbit#password` – для `kv` версии 2.

```
set gw server <имя> rabbitmq pass <пароль>
```

- указание существующего IP-Фонда для взаимодействия с RabbitMQ:

```
set gw server <имя> rabbitmq ipset-id <IP-Фонд>
```

- указание времени (в секундах) ожидания ответа от сервера RabbitMQ (значение от 1 до 60, по умолчанию – `10`):

```
set gw server <имя> rabbitmq timeout <время>
```

- указание интервала (в секундах) обновления регистрационной информации (URL и другие данные) Шлюза (значение от 10 до 100000, по умолчанию – `60`):

```
set gw server <имя> rabbitmq refreshtime <интервал>
```

- указание существующего Клиентского SSL-Профиля (по умолчанию – `backend-default`):

```
set gw server <имя> rabbitmq ssl-profile-id <SSL-Профиль>
```

- указание координатора маршрутизации сообщений, определенного в RabbitMQ:

```
set gw server <имя> rabbitmq exchange <координатор>
```

- указание ключа маршрутизации RabbitMQ, используемого для маршрутизации задачи в очереди:

```
set gw server <имя> rabbitmq routingkey <ключ_маршрутизации>
```

- указание способа передачи данных (по умолчанию – `false`):

Возможные значения:

- `true` – передача данных до первого подтверждения;
- `false` – передача данных по циклу (бесконечно).

```
set gw server <имя> rabbitmq single <false/true>
```

- (опционально) указание комментария для сервера:

```
set description <комментарий>
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

Пример конфигурации Шлюза:

```
set gw
set gw server GW-Farm-01
set gw server GW-Farm-01 ssl-profile-id
set gw server GW-Farm-01 vip ip 192.0.2.100
set gw server GW-Farm-01 vip port 1000
set gw server GW-Farm-01 vrf default
set gw server GW-Farm-01 websocket pingtimeout 30
set gw server GW-Farm-01 checktoken url https://192.0.2.20/api/wsproxy/v6.1/verify
set gw server GW-Farm-01 checktoken ssl-profile-id backend-default
set gw server GW-Farm-01 tcp-downstream reconnect 0
set gw server GW-Farm-01 tera-udp
set gw server GW-Farm-01 tera-udp vip ip 198.51.100.100
set gw server GW-Farm-01 tera-udp vip port 443
set gw server GW-Farm-01 tera-udp vrf default
set gw server GW-Farm-01 tera-udp src ip 203.0.113.49
set gw server GW-Farm-01 tera-udp src port 0
set gw server GW-Farm-01 tera-udp src vrf default
```

Координатор

Координатор маршрутизации сообщений отвечает за маршрутизацию сообщений в разные очереди. Координатор маршрутизации сообщений назначается в настройках клиента RabbitMQ для Шлюза Termidesk Connect. Настройка клиента RabbitMQ позволяет регистрировать Шлюз на одной Ферме или Агрегаторе Termidesk (Termidesk VDI).

Настройка клиента RabbitMQ выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Координатор**).

Для настройки клиента RabbitMQ выполнить:

Описание параметров также приведено в подразделе **Объект** `gw`.

- указание URL-адреса для подключения к серверу RabbitMQ:

```
set gw rabbitmq url <URL-адрес>
```

- указание имени пользователя для подключения к серверу RabbitMQ:

Так же имя пользователя может быть получено из Хранилища секретов. Пример значения параметра:

- `kv://secret/rabbit#username` – для `kv` версии 1;
- `kv://secret/data/rabbit#username` – для `kv` версии 2.

```
set gw rabbitmq user <имя>
```

- указание пароля пользователя для подключения к серверу RabbitMQ:

Так же пароль пользователя может быть получен из Хранилища секретов. Пример значения параметра:

- `kv://secret/rabbit#password` – для `kv` версии 1;
- `kv://secret/data/rabbit#password` – для `kv` версии 2.

```
set gw rabbitmq pass <пароль>
```

- указание существующего IP-Фонда для взаимодействия с RabbitMQ:

```
set gw rabbitmq ipset-id <IP-Фонд>
```

- указание времени (в секундах) ожидания ответа от сервера RabbitMQ (значение от 1 до 60, по умолчанию – `10`):

```
set gw rabbitmq timeout <время>
```

- указание интервала (в секундах) обновления регистрационной информации (URL и другие данные) Шлюза (значение от 10 до 100000, по умолчанию – `60`):

```
set gw rabbitmq refreshtime <интервал>
```

- указание существующего Клиентского SSL-Профиля (по умолчанию – `backend-default`):

```
set gw rabbitmq ssl-profile-id <SSL-Профиль>
```

- указание координатора маршрутизации сообщений, определенного в RabbitMQ:

```
set gw rabbitmq exchange <координатор>
```

- указание ключа маршрутизации RabbitMQ, используемого для маршрутизации задачи в очереди:

```
set gw rabbitmq routingkey <ключ_маршрутизации>
```

- указание способа передачи данных (по умолчанию – `false`):

Возможные значения:

- `true` – передача данных до первого подтверждения;
- `false` – передача данных по циклу (бесконечно).

```
set gw rabbitmq single <false/true>
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

Сбор статистики

Для сбора статистики и получения метрик Шлюза настраивается управляющий сервер Шлюза Termidesk Connect.

Настройка управляющего сервера Шлюза выполняется одним из способов:

- из интерфейса командной строки Termidesk Connect;
- из веб-интерфейса Termidesk Connect (см. подраздел **Веб. Сбор статистики**).

Для настройки управляющего сервера Шлюза выполнить:

Описание параметров также приведено в подразделе **Объект** `gw`.

- указание существующего IP-адреса для получения статистики Шлюза:

```
set gw mgt listen ip <IP-адрес>
```

- указание номера порта для получения статистики Шлюза:

```
set gw mgt listen port <порт>
```

- указание существующего Серверного SSL-Профиля:

```
set gw mgt ssl-profile-id <SSL-Профиль>
```

- указание пути для получения статистики Шлюза (по умолчанию – `/api/health`):

```
set gw mgt path <путь>
```

- указание токена доступа для получения статистики Шлюза:

Для интеграции с Termidesk (Termidesk VDI) значение может быть получено из параметра `HEALTH_CHECK_ACCESS_KEY`, определенного в конфигурационном файле `/etc/opt/termidesk-vdi/termidesk.conf` узла компонента «Универсальный диспетчер» Termidesk.

Так же токен может быть получен из Хранилища секретов. Пример значения параметра:

- `kv://secret/gw#token` – для `kv` версии 1;
- `kv://secret/data/gw#token` – для `kv` версии 2.

```
set gw mgt token <токен>
```

- указание пути для получения метрик Шлюза (по умолчанию – `/api/health/metrics`):

```
set gw mgt metrics path <путь>
```

- указание токена доступа для получения метрик Шлюза:

Для интеграции с Termidesk значение может быть получено из параметра `METRICS_ACCESS_KEY`, определенного в конфигурационном файле `/etc/opt/termidesk-vdi/termidesk.conf` узла компонента «Универсальный диспетчер» Termidesk.

Так же токен может быть получен из Хранилища секретов. Пример значения параметра:

- `kv://secret/gw#token` – для `kv` версии 1;
- `kv://secret/data/gw#token` – для `kv` версии 2.

```
set gw mgt metrics token <токен>
```

- применение конфигурации:

```
commit
```

- сохранение конфигурации:

```
write
```

ВОССТАНОВЛЕНИЕ РАБОТОСПОСОБНОСТИ

Восстановление базы данных

При обновлении Termidesk Connect могут возникать ошибки в конфигурации базы данных. В этом случае загрузка выполняется с пустой конфигурацией базы данных для успешного запуска.

Для восстановления базы данных:

- загрузить резервную копию, игнорируя неконсистентные параметры (эти параметры будут удалены):

```
restore_broken_config
```

- восстановить параметры до работоспособного состояния, т.е. внести изменения в базе данных;
- (опционально) просмотреть различия между предварительной и рабочей конфигурациями:

```
show difference
```

- проверить корректность изменений в предварительной конфигурации:

```
validate
```

- применить изменения в рабочую конфигурации:

```
commit
```

- сохранить изменения рабочей конфигурации:

```
write
```

- (опционально) просмотреть конфигурацию в формате `XML`:

```
show configuration xml
```

ИНТЕГРАЦИЯ С СИСТЕМАМИ МОНИТОРИНГА

Просмотр статистики по протоколу HTTP

Для просмотра статистической информации Termidesk Connect возможно запрашивать данные по протоколу HTTP.

Вывод статистики доступен в форматах:

- `OpenMetrics` – метрики систем мониторинга, например, для сбора статистики в системе мониторинга Prometheus. Формат запроса: `https://<IP-адрес_управления>/metrics`;
- `JSON` – метрики в формате `JSON`. Формат запроса: `https://<IP-адрес_управления>/metrics?format=json`.

ИНТЕРФЕЙС РАСШИРЕННОГО МЕНЮ

Общие сведения по работе с интерфейсом расширенного меню

Интерфейс расширенного меню предназначен для настройки и управления Termidesk Connect и предоставляет базовый набор функций для работы с Termidesk Connect после его запуска.

Переход в интерфейс расширенного меню выполняется клавишей **<F2>** из главного меню Termidesk Connect. Для перехода потребуется ввести текущий пароль администратора (по умолчанию после установки – `tdadmin`).

Смена пароля администратора

После установки Termidesk Connect по умолчанию используется логин `tdadmin` с паролем `tdadmin` для доступа к ряду функций управления.

Для смены пароля:

- в главном меню Termidesk Connect нажать клавишу **<F2>**, ввести текущий пароль администратора;
- далее выбрать пункт «Пароль» и нажать клавишу **<ENTER>**;
- в появившемся окне (см. Ввод текущего пароля администратора) ввести текущий пароль и нажать экранную кнопку **[OK]**;
- затем (см. Ввод нового пароля администратора) ввести новый пароль, переключиться на строку «Повтор пароля» при помощи клавиши **<↓>** (**<СТРЕЛКА ВНИЗ>**) и повторить ввод пароля. Подтвердить данные, нажав экранную кнопку **[OK]**.

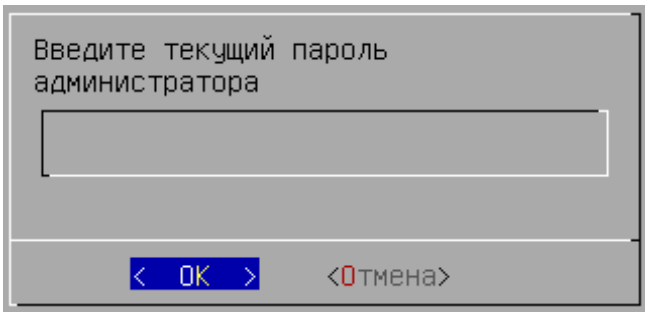

Рисунок 66. Ввод текущего пароля администратора

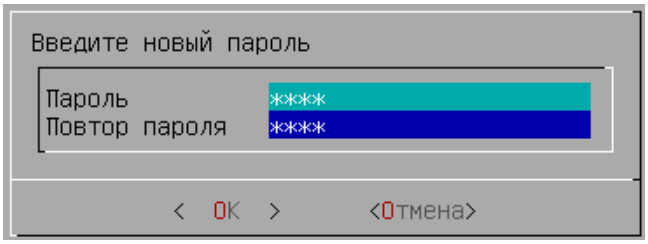

Рисунок 67. Ввод нового пароля администратора

Перезагрузка

Для перезагрузки Termidesk Connect:

- в главном меню Termidesk Connect нажать клавишу **<F2>**, ввести текущий пароль администратора;
- далее выбрать пункт «Перезагрузка» и нажать клавишу **<ENTER>**;
- подтвердить действие, нажав экранную кнопку **[Да]**.

Выключение

Для завершения работы Termidesk Connect и выключения ВМ:

- в главном меню Termidesk Connect нажать клавишу **<F2>**, ввести текущий пароль администратора;
- далее выбрать пункт «Выключение» и нажать клавишу **<ENTER>**;
- подтвердить действие (см. Подтверждение выключения Termidesk Connect), нажав экранную кнопку **[Да]**.

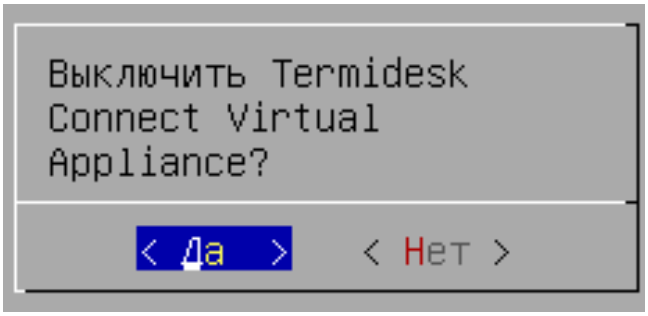

Рисунок 68. Подтверждение выключения Termidesk Connect

Переход в интерфейс CLI

Для перехода в интерфейс CLI (командной строки):

- в главном меню Termidesk Connect нажать клавишу **<F2>**, ввести текущий пароль администратора;
- далее выбрать пункт «CLI» и нажать клавишу **<ENTER>**;
- отобразится строка приглашения `termidesk-connect#`, свидетельствующая об успешном переходе в интерфейс командной строки.

ИНТЕРФЕЙС CLI

Общие сведения по работе с CLI

CLI предназначен для настройки и управления Termidesk Connect и предоставляет набор

команд с возможностью их автоматического дополнения.

CLI является основным способом настройки Termidesk Connect.

Переход в CLI осуществляется из главного меню Termidesk (см. подраздел **Переход в интерфейс CLI**).

Для просмотра списка доступных команд или просмотра справки по ним можно воспользоваться клавишами:

- **<TAB>** – выводит возможные варианты продолжения команды:
  - если вариантов больше одного, то выводит список доступных опций;
  - если вариант один, то автоматически дописывает окончание команды;
- **<?>** – выводит список доступных команд с краткой справкой к ним. Ввод клавиши не будет отображен в CLI.

Команда aaa

Выполняет управление пользователями, авторизующимися через AAA.

Формат:

```
aaa user <действие>
```

или:

```
aaa user <действие> | <утилита>
```

Доступные утилиты приведены в таблице (см. Доступные утилиты команды aaa).

Таблица 29. Доступные утилиты команды aaa

Утилита	Описание
grep	Вызов стандартной утилиты grep для поиска по заданному выражению. Формат: `aaa user <действие> | grep <выражение>`
less	Вызов стандартной утилиты less для постраничного просмотра и поиска. Формат: `aaa user <действие> | less`

Доступные действия приведены в таблице (см. Доступные действия команды aaa).

Таблица 30. Доступные действия команды aaa

Действие	Описание
`block <время> <имя_пользователя> <пространство_имен_пользователя>`	Блокирует пользователя AAA на период указанного времени (в секундах)
`kill-session <имя_пользователя> <пространство_имен_пользователя>`	Останавливает текущие сессии пользователя AAA
`show-sessions <имя_пользователя> <пространство_имен_пользователя>`	Показывает сессии пользователя AAA
`unblock <имя_пользователя> <пространство_имен_пользователя>`	Разблокирует пользователя AAA

Команда `bash`

Выполняет переход в стандартный интерфейс командной строки Linux Shell.

Для некоторых команд, выполняемых через стандартный интерфейс командной строки Linux Shell, требуется повышение привилегий. В этом случае следует вызвать команду через `sudo` и ввести текущий пароль администратора:

```
sudo <команда>
```

Для того чтобы вернуться в интерфейс CLI Termidesk Connect, нужно выполнить в Linux Shell:

```
exit
```

Команда `clear`

Очищает CLI от результатов выполнения предыдущих команд.

Команда `commit`

Применяет изменения, сделанные в предварительной конфигурации, в рабочую конфигурацию. Без сохранения (команда `write`) изменения сбросятся после перезагрузки Termidesk Connect.

Команда `cookieinsert`

Работает с привязкой сессии пользователя по cookie (COOKIEINSERT).

Формат:

```
cookieinsert <действие>
```

Доступные действия приведены в таблице (см. таблицу Доступные действия для команды `cookieinsert`).

Таблица 31. Доступные действия для команды `cookieinsert`

Действие	Описание
`decode <значение>`	Декодирует значение COOKIEINSERT и выводит его на экран. Применяется для получения информации о Сервере Балансировки, закодированном в COOKIEINSERT

Команда `debug`

Устанавливает уровень отладки и вывода отладочной информации. Позволяет включать или отключать отладочные сообщения, что полезно для диагностики и анализа работы системы.

Формат:

```
debug <подсистема> <уровень>
```

Доступные подсистемы для отладки приведены в таблице (см. Доступные подсистемы).

Таблица 32. Доступные подсистемы

Подсистема	Описание
`backend`	Подсистема настройки и управления внутренней конфигурацией Termidesk Connect
`cli`	Подсистема интерфейса командной строки
`restconf`	Подсистема интерфейса управления, основанного на протоколе REST

Доступные уровни отладки приведены в таблице (см. Доступные уровни отладки).

Таблица 33. Доступные уровни отладки

Уровень	Описание
`0`	Отладочный режим отключен. Отладочные сообщения выводиться не будут
`1`	Вывод минимальных отладочных сообщений
`2` и выше	Чем выше уровень, тем больше подробностей отображается в отладочных сообщениях

Команда `delete`

Удаляет существующие объекты или их параметры из конфигурации (изменения попадают в предварительную конфигурацию).

Формат:

```
delete <объект> <параметр> <значение>
```

Доступные для удаления объекты приведены в таблице (см. таблицу Доступные для удаления объекты).

Таблица 34. Доступные для удаления объекты

Объект	Описание
`all`	Вся рабочая конфигурация. Не требует указания параметров объекта, поскольку удаляет все параметры. Пример команды: `delete all`
`aaa`	Конфигурация, описывающая настройку аутентификации пользователя со стороны клиента и способа передачи данных Реальному Серверу, в том числе с использованием внешних AAA-сервисов
`acl`	Конфигурация, описывающая настройку списка контроля доступа
`aggregation`	Конфигурация, описывающая настройку агрегации каналов
`arp`	Конфигурация, описывающая статическую запись ARP
`audit`	Конфигурация, описывающая настройку отправки событий Termidesk Connect на syslog-сервер
`ddos`	Конфигурация, описывающая настройку защиты от DDoS-атак
`dns`	Конфигурация, описывающая настройку разрешения доменных имен в IP-адреса
`ethernet`	Конфигурация, описывающая настройку сетевых интерфейсов
`geolb`	Конфигурация, описывающая настройку геораспределенной балансировки
`groups`	Конфигурация, описывающая группы пользователей
`gw`	Конфигурация, описывающая настройку Шлюза
`ha`	Конфигурация, описывающая настройку в отказоустойчивом (высокодоступном) исполнении Termidesk Connect
`health-check`	Конфигурация, описывающая настройку Проверок
`http-profile`	Конфигурация, описывающая настройку HTTP-Профиля
`ip`	Конфигурация, описывающая настройку IP-адресов
`ipset`	Конфигурация, описывающая настройку IP-Фондов
`lbs`	Конфигурация, описывающая настройку Серверов Балансировки
`ldap`	Конфигурация, описывающая настройку подключения к службе каталогов
`logging`	Конфигурация, описывающая уровень журналирования
`nacm`	Конфигурация, описывающая настройку протокола NETCONF
`ntp`	Конфигурация, описывающая настройку серверов точного времени
`openbao`	Конфигурация, описывающая настройку подключения к Хранилищу секретов
`persistence-profile`	Конфигурация, описывающая настройку Профиля сохранения сессий
`restconf`	Конфигурация, описывающая настройку протокола RESTCONF
`rl-profile`	Конфигурация, описывающая настройку параметров для ограничения трафика
`rs-pools`	Конфигурация, описывающая настройку Групп Реальных Серверов
`snmp`	Конфигурация, описывающая настройку протокола SNMP версий 1 и 2c
`snmpv3`	Конфигурация, описывающая настройку протокола SNMP версии 3
`ssl-policy`	Конфигурация, описывающая настройку SSL-Политик
`ssl-profile`	Конфигурация, описывающая настройку SSL-Профилей для функционала SSL Offload
`system`	Конфигурация, описывающая системные настройки

Объект	Описание
`tcp-profile`	Конфигурация, описывающая настройку TCP-Профиля
`user`	Конфигурация, описывающая настройку пользователей
`vlan`	Конфигурация, описывающая настройку VLAN
`vrf`	Конфигурация, описывающая настройку VRF
`vs`	Конфигурация, описывающая настройку Виртуальных Серверов

Команда `discard`

Отменяет несохраненные изменения предварительной конфигурации и возвращает ее к состоянию, соответствующему текущей рабочей конфигурации.

Команда `ha`

Управляет состоянием узла в отказоустойчивой конфигурации.

Формат:

```
ha <действие>
```

Доступные действия приведены в таблице (см. Доступные действия для команды `ha`).

Таблица 35. Доступные действия для команды `ha`

Действие	Описание
`ha set-active`	Перевести текущий узел в активное состояние (сделатьмастер-узлом)
`ha set-active <идентификатор>`	Перевести узел с идентификатором `<идентификатор>` в активное состояние (сделать мастер-узлом)

Команда `load`

Загружает конфигурацию из XML-файла в предварительную конфигурацию.

Формат:

```
load <XML-файл> <действие>
```

Доступные действия приведены в таблице (см. Доступные действия для команды `load`).

Таблица 36. Доступные действия для команды `load`

Действие	Описание
`merge`	Объединяет параметры из указанного файла с текущей предварительной конфигурацией. Если объекты из файла совпадают с уже существующими, то они будут обновлены. Если какие-то объекты отсутствуют, то они будут добавлены

Действие	Описание
`replace`	Заменяет текущую предварительную конфигурацию параметрами из указанного файла. Все существующие в конфигурации объекты и параметры будут удалены перед загрузкой новых

Команда `mode`

Выбирает поддерево конфигурации.

Команда `quit`

Выполняет выход из CLI и переход в расширенное меню Termidesk Connect.

Команда `restore-broken-config`

Восстанавливает неконсистентную конфигурацию после обновления, т.е. рабочая конфигурация копируется в предварительную с удалением ошибочных параметров.

Команда `save`

Сохраняет текущую предварительную конфигурацию Termidesk Connect в XML-файл.

Формат:

```
save <XML-файл>
```

Чтобы позже загрузить сохраненную конфигурацию из файла используется команда `load`.

Команда `secret`

Выполняет принудительное обновление секретов и сертификатов из Хранилища секретов.

Формат:

```
secret force-sync
```

Команда `set`

Общие сведения по команде `set`

Устанавливает новый объект в конфигурации. Команда позволяет как создать объект без параметров, так и создать объект и назначить ему параметры одной командой.

Формат:

```
set <объект>
```

или:

```
set <объект> <параметр объекта> <ключ> <значение>
```

Доступные для создания объекты приведены в таблице (см. таблицу Доступные объекты для команды set).

Таблица 37. Доступные объекты для команды set

Объект	Описание
aaa	Конфигурация, описывающая настройку аутентификации пользователя со стороны клиента и способа передачи данных Реальному Серверу, в том числе с использованием внешних AAA-сервисов
acl	Конфигурация, описывающая настройку списка контроля доступа
aggregation	Конфигурация, описывающая настройку агрегации каналов
arp	Конфигурация, описывающая статическую запись ARP
audit	Конфигурация, описывающая настройку отправки событий Termidesk Connect на syslog-сервер
ddos	Конфигурация, описывающая настройку защиты от DDoS-атак
dns	Конфигурация, описывающая настройку разрешения доменных имен в IP-адреса
ethernet	Конфигурация, описывающая настройку сетевых интерфейсов
geolb	Конфигурация, описывающая настройку геораспределенной балансировки
groups	Конфигурация, описывающая группы пользователей
gw	Конфигурация, описывающая настройку Шлюза
ha	Конфигурация, описывающая настройку в отказоустойчивом (высокодоступном) исполнении Termidesk Connect
health-check	Конфигурация, описывающая настройку Проверок
http-profile	Конфигурация, описывающая настройку HTTP-Профиля
ip	Конфигурация, описывающая настройку IP-адресов
ipset	Конфигурация, описывающая настройку IP-Фондов
krb5	Конфигурация, описывающая настройку KDC-Сервера
lbs	Конфигурация, описывающая настройку Серверов Балансировки
ldap	Конфигурация, описывающая настройку подключения к службе каталогов
logging	Конфигурация, описывающая уровень журналирования
nacm	Конфигурация, описывающая настройку протокола NETCONF
ntp	Конфигурация, описывающая настройку серверов точного времени
openbao	Конфигурация, описывающая настройку подключения к Хранилищу секретов
persistence-profile	Конфигурация, описывающая настройку Профиля сохранения сессий
restconf	Конфигурация, описывающая настройку протокола RESTCONF
rl-profile	Конфигурация, описывающая настройку параметров для ограничения трафика
rs-pools	Конфигурация, описывающая настройку Групп Реальных Серверов
snmp	Конфигурация, описывающая настройку протокола SNMP версий 1 и 2c
snmpv3	Конфигурация, описывающая настройку протокола SNMP версии 3
ssl-policy	Конфигурация, описывающая настройку SSL-Политик
ssl-profile	Конфигурация, описывающая настройку SSL-Профилей для функционала SSL Offload

Объект	Описание
system	Конфигурация, описывающая системные настройки
tcp-profile	Конфигурация, описывающая настройку TCP-Профиля
user	Конфигурация, описывающая настройку пользователей
vlan	Конфигурация, описывающая настройку VLAN
vrf	Конфигурация, описывающая настройку VRF
vs	Конфигурация, описывающая настройку Виртуальных Серверов

Объект aaa

Доступные команды объекта aaa приведены в таблице (см. Доступные команды объекта aaa).

Таблица 38. Доступные команды объекта aaa

Команда	Описание
set aaa profile <имя>	Создание AAA-Профиля
Настройка AAA-Профиля	
set aaa profile <имя> authentication <значение>	Включение или выключение аутентификации пользователя (по умолчанию – true)
set aaa profile <имя> attempt-ttl <значение>	Период (в секундах), в течение которого подсчитываются неудачные попытки аутентификации (значение от 1 до 86400, по умолчанию – 600)
set aaa profile <имя> attempts <значение>	Количество неудачных попыток аутентификации (значение от 1 до 255, по умолчанию – 3)
set aaa profile <имя> blocking-ttl <значение>	Время (в секундах) блокировки (значение от 1 до 86400, по умолчанию – 600)
set aaa profile <имя> cookie-ttl <значение>	Время (в секундах) жизни cookie аутентификации (значение от 1 до 86400, по умолчанию – 600)
set aaa profile <имя> cookie-name <значение>	Название аутентификационной cookie (по умолчанию – TC-AAC)
set aaa profile <имя> cookie-attr <значение>	Атрибуты аутентификационной cookie
set aaa profile <имя> realm <значение>	Пространство имен пользователя для аутентификации
set aaa profile <имя> servers <значение>	Порядковый номер привязки настроенного AAA-Сервера (минимальное значение – 1) Может быть задано несколько параметров servers, т.е. для одного AAA-Профиля может быть задано несколько настроенных AAA-Серверов. В случае недоступности одного AAA-Сервера осуществляется переход на другой. Для параметра servers устанавливается дополнительный параметр: • server-id – имя настроенного AAA-Сервера
set aaa server <имя>	Создание AAA-Сервера
Настройка AAA-Сервера	

Команда	Описание
`set aaa server <имя> LDAP type <значение>`	Тип службы каталогов. Для параметра `type` возможны значения: • `AD` – подключение к службе каталогов с поддержкой Microsoft Active Directory; • `OpenLDAP` – подключение к службе каталогов с поддержкой реализации с открытым исходным кодом протокола LDAP – OpenLDAP
`set aaa server <имя> LDAP administrator-bind-dn <значение>`	Полное имя учетной записи администратора LDAP-каталога. Так же имя может быть получено из Хранилища секретов
`set aaa server <имя> LDAP base-dn <значение>`	Корневая точка LDAP-каталога
`set aaa server <имя> LDAP domain <значение>`	Адрес для подключения к LDAP-каталогу
`set aaa server <имя> LDAP group-attribute <значение>`	Атрибут группы
`set aaa server <имя> LDAP import-attribute-list <значение>`	Список атрибутов, которые должны быть импортированы из службы каталогов
`set aaa server <имя> LDAP password <значение>`	Пароль для соединения с LDAP-каталогом. Так же пароль может быть получен из Хранилища секретов
`set aaa server <имя> LDAP port <значение>`	Номер порта (по умолчанию – `389`)
`set aaa server <имя> LDAP timeout <значение>`	Время (в секундах) ожидания авторизации (по умолчанию – `30`)
`set aaa server <имя> LDAP ssl-profile-id <значение>`	Клиентский SSL-Профиль
`set aaa server <имя> LDAP user-name-attribute <значение>`	Атрибут имени пользователя (по умолчанию – `userPrincipalName`)
`set aaa sso-profile <имя>`	Создание SSO-Профиля
Настройка SSO-Профиля	

Команда	Описание
`set aaa sso-profile <имя> <тип>`	Тип SSO-Профиля. Для `<тип>` возможны значения: • `BASIC` – базовый алгоритм аутентификации; • `OFF` – данные аутентификация не передаются; • `KRB5-IM` – аутентификация с использованием Kerberos версии 5. При выборе типа аутентификации `KRB5-IM` задаются: • `realm` – пространство имен пользователя для аутентификации; • `service-principal` – имя делегирующего пользователя, обладающего правом от своего имени запрашивать билеты для обычных пользователей и делегировать их на другие сервисы (так же имя может быть получено из Хранилища секретов); • `password` – пароль делегирующего пользователя (так же пароль может быть получен из Хранилища секретов); • `servers` – KDC-Сервер для аутентификации из списка добавленных в Termidesk Connect (может быть указано несколько) и его приоритет

Объект `acl`

Доступные команды объекта `acl` приведены в таблице (см. Доступные команды объекта `acl`).

Таблица 39. Доступные команды объекта `acl`

Команда	Описание
`set acl list <имя>`	Создание списка контроля доступа
`set acl list <имя> logging <значение>`	Настройка журналирования событий. Значение может быть: • `true` – разрешить журналирование; • `false` – запретить журналирование
`set acl list <имя> rules <приоритет>`	Приоритет правила в списке контроля доступа
`set acl list <имя> rules <приоритет> action <значение>`	Тип действия при обработке правила. Значение может быть: • `ALLOW` – разрешить обработку пакета; • `DENY` – запретить обработку пакета

Команда	Описание
`set acl list <имя> rules <приоритет> destination compare <параметр> <значение>`	Условие определения списка портов назначения, для которых создается правило. Параметром может быть: • `eq` – применять правило к указанному значению; • `gt` – применять правило к значению больше указанного; • `lt` – применять правило к значению меньше указанного; • `not` – применять правило ко всем значениям, не соответствующим указанному. Параметр применяется для протоколов типа `TCP` и `UDP`
`set acl list <имя> rules <приоритет> destination network <значение>`	Адреса назначения в формате CIDR, для которых создается правило
`set acl list <имя> rules <приоритет> destination oneport <значение>`	Определяет один порт назначения, для которого создается правило. Параметр применяется для протоколов типа `TCP` и `UDP`
`set acl list <имя> rules <приоритет> destination portrange <параметр> <значение>`	Диапазон портов назначения, для которых создается правило. Параметром может быть: • `from` – начальный порт диапазона; • `to` – конечный порт диапазона. Параметр применяется для протоколов типа `TCP` и `UDP`
`set acl list <имя> rules <приоритет> proto <значение>`	Протокол, для которого создается правило. Значение может быть: • `ICMP`; • `IP`; • `TCP`; • `UDP`
`set acl list <имя> rules <приоритет> source compare <параметр> <значение>`	Условие определения списка портов источника запроса, для которых создается правило. Параметром может быть: • `eq` – применять правило к указанному значению; • `gt` – применять правило к значению больше указанного; • `lt` – применять правило к значению меньше указанного; • `not` – применять правило ко всем значениям, не соответствующим указанному. Параметр применяется для протоколов типа `TCP` и `UDP`
`set acl list <имя> rules <приоритет> source network <значение>`	Адреса источника запроса в формате CIDR, для которых создается правило
`set acl list <имя> rules <приоритет> source oneport <значение>`	Определяет один порт источника запроса, для которого создается правило. Параметр применяется для протоколов типа `TCP` и `UDP`

Команда	Описание
`set acl list <имя> rules <приоритет> source portrange <параметр> <значение>`	Диапазон портов источника запроса, для которых создается правило. Параметром может быть: • `from` – начальный порт диапазона; • `to` – конечный порт диапазона. Параметр применяется для протоколов типа `TCP` и `UDP`
`set acl list <имя> rules <приоритет> vlan-id <значение>`	Идентификатор VLAN, для которого создается правило

Объект `aggregation`

Доступные команды объекта `aggregation` приведены в таблице (см. Доступные команды объекта `aggregation`).

Таблица 40. Доступные команды объекта `aggregation`

Команда	Описание
`set aggregation name <имя>`	Создание LAG-интерфейса
`set aggregation name <имя> type <тип>`	Назначение типа LAG-интерфейсу. Не допускается изменение типа для уже созданного и настроенного интерфейса. Для использования типа LACP он должен поддерживаться сетевым коммутатором. Возможные значения: • `LACP` – будет использоваться протокол LACP. После того как агрегированный канал сформирован, за поддержание статуса канала отвечает LACP; • `Active-backup` – будет использоваться только активный интерфейс из объединенных. При отказе активного интерфейса выполняется автоматическое переключение на резервный интерфейс
`set aggregation name <имя> lacp-rate <режим>`	Назначение режима работы LAG-интерфейса (по умолчанию – `slow`). Возможные значения: • `fast` – режим работы, при котором частота отправки LACPDU-пакетов составляет один раз в секунду для более быстрого обнаружения изменений в сети; • `slow` – режим работы, при котором частота отправки LACPDU-пакетов составляет один раз в 30 секунд
`set aggregation name <имя> miimon <значение>`	Назначение частоты (в миллисекундах) мониторинга MII (Media Independent Interface) (по умолчанию – `100`)
`set aggregation name <имя> vrf <имя_VRF>`	Назначение VRF для LAG-интерфейса

Объект `arp`

Доступные команды объекта `arp` приведены в таблице (см. Доступные команды объекта `arp`).

Таблица 41. Доступные команды объекта `arp`

Команда	Описание
`set arp static <IP-адрес> mac <MAC-адрес>`	Создание статической записи ARP

Объект `audit`

Доступные команды объекта `audit` приведены в таблице (см. Доступные команды объекта audit).

Таблица 42. Доступные команды объекта audit

Команда	Описание
`set audit syslog <имя>`	Создание syslog-сервера
Настройка syslog-сервера	
`set audit syslog <имя> ca-cert <значение>`	Путь к файлу корневого сертификата для защищенного соединения SSL/TLS. Так же файл может быть получен из Хранилища секретов
`set audit syslog <имя> ca-key <значение>`	Путь к файлу ключа для защищенного соединения SSL/TLS. Так же файл может быть получен из Хранилища секретов
`set audit syslog <имя> facility <значение>`	Категория событий для записи в журнал. Значение может быть: • `local0`; • `local1`; • `local2`; • `local3`; • `local4`; • `local5`; • `local6`; • `local7`
`set audit syslog <имя> facility <значение>`	Уровень событий для записи в журнал. Значение может быть: • `alert`; • `crit`; • `debug`; • `emerg`; • `error`; • `info` (по умолчанию); • `notice`; • `warning`
`set audit syslog <имя> local-address <значение>`	IP-адрес, используемый для отправки событий на syslog-сервер
`set audit syslog <имя> peer-verify <значение>`	Использование проверки подлинности сертификата при запросах к syslog-серверу. Значение может быть: • `no`; • `yes`

Команда	Описание
`set audit syslog <имя> port <значение>`	Порт для подключения к syslog-серверу
`set audit syslog <имя> service <значение>`	Служба, события которой будут записаны в журнал. Значение может быть: • `GSLB` – запись событий службы глобальной балансировки; • `LocalLB` – запись событий службы локальной балансировки; • `System` – запись системных событий; • `kernel` – запись событий ядра ОС; • `termidesk-clixon` – запись событий службы управления конфигурацией Clixon (по умолчанию)
`set audit syslog <имя> transport <значение>`	Используемый протокол для передачи событий. Значение может быть: • `tcp`; • `tls`; • `udp`  При использовании значения `tls` нужно задать значения параметрам `ca-cert`, `ca-key` и `peer-verify`.

Объект ddos

Доступные команды объекта `ddos` приведены в таблице (см. Доступные команды объекта `ddos`).

Таблица 43. Доступные команды объекта `ddos`

Команда	Описание
`set ddos profile <имя>`	Создание DDoS-Профиля
	Настройка DDoS-Профиля
`set ddos profile <имя> <тип>`	Тип DDoS-Профиля. Указывает на определенную DDoS-атаку, может быть: • `icmp-flood` – ICMP-атака; • `syn-flood` – SYN-атака; • `udp-flood` – UDP-атака
`set ddos profile <имя> <тип> enable <значение>`	Включение или выключение защиты от определенной DDoS-атаки. Для параметра `enable` могут устанавливаться следующие значения: • `true` – включение защиты; • `false` – выключение защиты

Команда	Описание
`set ddos profile <имя> <тип> interface-limit <значение>`	Ограничение по всему интерфейсу при защите от определенной DDoS-атаки. Для параметра `interface-limit` могут устанавливаться дополнительные параметры: • `threshold` – пороговое значение на интерфейсе; • `hysteresis` – гистерезис порогового значения на интерфейсе
`set ddos profile <имя> <тип> ip-limit <значение>`	Ограничение для одного IP-адреса при защите от определенной DDoS-атаки. Для параметра `ip-limit` могут устанавливаться дополнительные параметры: • `threshold` – пороговое значение от одного IP-адреса; • `hysteresis` – гистерезис порогового значения от одного IP-адреса
`set ddos profile <имя> logging <значение>`	Настройка журналирования при обнаружении или завершении атаки. Значение может быть: • `true` – включение журналирования; • `false` – выключение журналирования

Объект dns

Доступные команды объекта `dns` приведены в таблице (см. Доступные команды объекта `dns`).

Таблица 44. Доступные команды объекта `dns`

Команда	Описание
`set dns ip-address <IP-адрес>`	Добавление DNS-сервера. Конфигурация сохраняется в файле ОС `/etc/resolv.conf`
`set dns search-domain <имя_домена>`	Добавление домена поиска. Конфигурация сохраняется в файле ОС `/etc/resolv.conf`

Объект ethernet

Доступные команды объекта `ethernet` приведены в таблице (см. Доступные команды объекта `ethernet`).

Таблица 45. Доступные команды объекта `ethernet`

Команда	Описание
`set ethernet name <имя> acl-id <имя_списка>`	Назначение списка контроля доступа сетевому интерфейсу
`set ethernet name <имя> agg <имя_агрегированного_интерфейса>`	Привязка интерфейса к агрегированному каналу
`set ethernet name <имя> ddos-profile-id <имя_DDoS-Профиля>`	Назначение DDoS-Профиля сетевому интерфейсу

Команда	Описание
`set ethernet name <имя> vrf <имя_vrf>`	Назначение VRF сетевому интерфейсу
`set ethernet name <имя> mtu <значение>`	Задание параметра MTU (значение от 0 до 9216, по умолчанию – `1500`)
`set ethernet name <имя> state <ENABLED/DISABLED>`	Активация или отключение сетевого интерфейса (по умолчанию – `ENABLED`)
`set ethernet name <имя> ha-monitor <true/false>`	Включение или отключение отслеживания состояния интерфейса для готовности узла к переходу в состояние `ACTIVE` (по умолчанию – `false`)

Объект `geolb`

Доступные команды объекта `geolb` приведены в таблице (см. Доступные команды объекта `geolb`).

Таблица 46. Доступные команды объекта `geolb`

Команда	Описание
`set geolb adns <IP-адрес> <порт>`	Создание конфигурации ADNS-сервера.  В случае использования отказоустойчивой конфигурации Termidesk Connect необходимо указывать общий IP-адрес кластера. Пример команды: `set geolb adns ip 192.168.1.1 53`
`set geolb geodb <имя>`	Указание файла геолокационной IP-базы
Настройка конфигурации Сервиса	
`set geolb service <имя> hc-id <значение>`	Проверка, относящаяся к георасределенной балансировке
`set geolb service <имя> local-ip <значение>`	Внутренний IP-адрес Сервиса
`set geolb service <имя> public-ip <значение>`	Внешний IP-адрес Сервиса
`set geolb service <имя> site <значение>`	Площадка
`set geolb service <имя> weight <значение>`	Вес Сервиса
`set geolb site <имя>`	Создание конфигурации удаленной Площадки  Функциональность команды в разработке и приведена для ознакомления.
`set geolb view <имя> <сеть>`	Создание конфигурации DNS View. Параметр `<сеть>` задается в формате CIDR. Пример команды: `set geolb view ViewForLocalNet net 172.16.0.0/16`

Команда	Описание
`set geolb vs <имя>`	Создание конфигурации Виртуального Сервера геобалансировки
Настройка конфигурации Виртуального Сервера геобалансировки	
`set geolb vs <имя> algorithm <значение>`	Алгоритм выбора Виртуального Сервера. Для параметра `algorithm` могут устанавливаться следующие значения: • `ONLINE` – Виртуальный Сервер перенаправит подключения только на доступные Сервисы; • `ROUNDROBIN` – Виртуальный Сервер последовательно распределит подключения между Сервисами для их равномерного распределения; • `SOURCEIPHASH` – Виртуальный Сервер будет перенаправлять подключения к Сервисам, основываясь на значении хеша IP-адреса. Это гарантирует, что подключение одного пользователя всегда будет направляться на один и тот же Сервис, снижая вероятность перегрузки; • `STATICPROXIMITY` – Виртуальный Сервер перенаправит подключения на ближайшую Площадку для выдачи Сервиса. В этом случае учитывается местоположение исходного источника запроса (см. подраздел **Геолокационные IP-базы**)
`set geolb vs <имя> ecs <значение>`	Использование опции EDNS Client Subnet. Для параметра `ecs` могут устанавливаться значения `ON` (включен) или `OFF` (отключен). Опция `ecs` используется для хранения информации о сети исходного источника запроса. Это полезно в случае, когда запрос пришел от вышестоящего DNS-сервера, местоположение которого точно неизвестно. Termidesk Connect получит информацию о сети исходного источника из входящего DNS-запроса и сохранит ее
`set geolb vs <имя> persistence <значение>`	Использование привязки сессии пользователя. Для параметра `persistence` могут устанавливаться следующие значения: • `OFF` – привязка входящего подключения к Сервису отключена; • `SRCIP` – привязка входящего подключения к Сервису на основе его IP-адреса
`set geolb vs <имя> service-ids <значение>`	Привязка Виртуального Сервера к Сервису
`set geolb vs <имя> view-id <значение>`	Привязка Виртуального Сервера к идентификатору DNS View
`set geolb zone <имя домена> <параметр> <значение>`	Создание конфигурации сетевой зоны. Параметр может принимать значение `nodes`, который является частью географически распределенной балансировки нагрузки для конкретного домена

Объект `groups`

Доступные команды объекта `groups` приведены в таблице (см. Доступные команды объекта `groups`).

Таблица 47. Доступные команды объекта groups

Команда	Описание
set groups <имя>	Создание группы

Объект gw

Доступные команды объекта gw приведены в таблице (см. Доступные команды объекта gw).

Таблица 48. Доступные команды объекта gw

Команда	Описание
	Настройка управляющего сервера
set gw mgt listen <значение>	Настройка прослушивания входящих подключений. Для параметра listen могут устанавливаться дополнительные параметры: • ip – IP-адрес прослушивания входящих подключений; • port – порт прослушивания входящих подключений
set gw mgt metrics <значение>	Настройка получения метрик Шлюза. Для параметра metrics могут устанавливаться дополнительные параметры: • path – путь для получения метрик Шлюза (по умолчанию – /api/health/metrics); • token – токен доступа для получения метрик Шлюза (так же токен может быть получен из Хранилища секретов) Для интеграции с Termidesk значение может быть получено из параметра METRICS_ACCESS_KEY, определенного в конфигурационном файле /etc/opt/termidesk-vdi/termidesk.conf узла компонента «Универсальный диспетчер» Termidesk.
set gw mgt path <значение>	Путь для получения статистики Шлюза (по умолчанию – /api/health)
set gw mgt ssl-profile-id <значение>	Серверный SSL-Профиль
set gw mgt token <значение>	Токен доступа для получения статистики Шлюза. Для интеграции с Termidesk значение может быть получено из параметра HEALTH_CHECK_ACCESS_KEY, определенного в конфигурационном файле /etc/opt/termidesk-vdi/termidesk.conf узла компонента «Универсальный диспетчер» Termidesk. Так же токен может быть получен из Хранилища секретов
	Настройка клиента RabbitMQ
set gw rabbitmq exchange <значение>	Координатор маршрутизации сообщений, определенный в RabbitMQ. Отвечает за маршрутизацию сообщений в разные очереди

Команда	Описание
`set gw rabbitmq ipset-id <значение>`	IP-Фонд для взаимодействия с RabbitMQ
`set gw rabbitmq pass <значение>`	Пароль пользователя для подключения к серверу RabbitMQ. Так же пароль пользователя может быть получен из Хранилища секретов
`set gw rabbitmq refreshtime <значение>`	Интервал (в секундах) обновления регистрационной информации (URL и другие данные) Шлюза (значение от 10 до 100000, по умолчанию – `60`)
`set gw rabbitmq routingkey <значение>`	Ключ маршрутизации RabbitMQ, используемый для маршрутизации задачи в очереди (по умолчанию – `termidesk_appnode`)
`set gw rabbitmq single <значение>`	Способ передачи данных (по умолчанию – `false`). Для параметра `single` возможны значения: • `true` – передача данных до первого подтверждения; • `false` – передача данных по циклу (бесконечно)
`set gw rabbitmq ssl-profile-id <значение>`	Клиентский SSL-Профиль (по умолчанию – `backend-default`)
`set gw rabbitmq timeout <значение>`	Время (в секундах) ожидания ответа от сервера RabbitMQ (значение от 1 до 60, по умолчанию – `10`)
`set gw rabbitmq url <значение>`	URL-адрес для подключения к серверу RabbitMQ
`set gw rabbitmq user <значение>`	Имя пользователя для подключения к серверу RabbitMQ. Так же имя пользователя может быть получено из Хранилища секретов
`set gw server <имя>`	Создание Шлюза
Настройка параметров Шлюза	
`set gw server <имя> checktoken <значение>`	Настройка взаимодействия с компонентом «Универсальный диспетчер» Termidesk. Для параметра `checktoken` могут устанавливаться дополнительные параметры: • `ipset-id` – IP-Фонд, используемый для взаимодействия с компонентом «Универсальный диспетчер» Termidesk; • `ssl-profile-id` – Клиентский SSL-Профиль (по умолчанию – `backend-default`); • `url` – URL, поддерживаемый компонентом «Универсальный диспетчер» Termidesk с ролью «Портал пользователя» для обслуживания API-запросов по валидации подключений, запрашиваемых через Шлюз Следует использовать актуальные значения версий API для соответствующей версии компонента «Универсальный диспетчер» Termidesk.
`set gw server <имя> description <значение>`	Комментарий для сервера

Команда	Описание
`set gw server <имя> rabbitmq <значение>`	Настройка клиента RabbitMQ. Для параметра `rabbitmq` могут устанавливаться дополнительные параметры: • `exchange` – координатор маршрутизации сообщений, определенный в RabbitMQ. Отвечает за маршрутизацию сообщений в разные очереди; • `ipset-id` – IP-Фонд для взаимодействия с RabbitMQ; • `pass` – пароль пользователя для подключения к серверу RabbitMQ; • `refreshtime` – интервал (в секундах) обновления регистрационной информации (URL и другие данные) Шлюза (значение от 10 до 100000, по умолчанию – `60`); • `routingkey` – ключ маршрутизации RabbitMQ, используемый для маршрутизации задачи в очереди (по умолчанию – `termidesk_appnode`); • `single` – способ передачи данных (по умолчанию – `false`). Возможные значения: `true` – передача данных до первого подтверждения, `false` – передача данных по циклу (бесконечно); • `ssl-profile-id` – Клиентский SSL-Профиль (по умолчанию – `backend-default`); • `timeout` – время (в секундах) ожидания ответа от сервера RabbitMQ (значение от 1 до 60, по умолчанию – `10`); • `url` – URL-адрес для подключения к серверу RabbitMQ; • `user` – имя пользователя для подключения к серверу RabbitMQ
`set gw server <имя> ssl-profile-id <значение>`	Серверный SSL-Профиль
`set gw server <имя> tcp-downstream <значение>`	Настройка переподключения Шлюза к рабочим местам Termidesk. Для параметра `tcp-downstream` могут устанавливаться дополнительные параметры: • `ipset-id` – IP-Фонд для взаимодействия с рабочим местом Termidesk; • `reconnect` – количество разрешенных переподключений сессии пользователя (значение от 0 до 10, по умолчанию – `0`);  Разные протоколы удаленного доступа могут требовать разных значений параметра: • для RDP нужно использовать ненулевое значение, поскольку некоторые функции (например, перенаправление USB в гостевую ОС рабочего места по протоколу RDP) могут требовать переподключения; • для TERA и SPICE нужно использовать значение по умолчанию (`0`). • `ssl-profile-id` – Клиентский SSL-Профиль (по умолчанию – `backend-default`)

Команда	Описание
`set gw server <имя> tera-udp <значение>`	Настройка поддержки протокола UDP для протокола TERA на Шлюзе. Для параметра `tera-udp` могут устанавливаться дополнительные параметры: • `src ip` – IP-адрес источника, т.е. адрес, с которого Шлюз устанавливает UDP-соединение с рабочим местом Termidesk; Для параметра `src ip` значение `0.0.0.0` указывает на использование всех IP-адресов. • `src port` – порт источника (по умолчанию – `0`); • `src vrf` – VRF источника (по умолчанию – `default`); • `vip ip` – IP-адрес Виртуального Сервера для TERA; Параметр `vip ip` может принимать любое значение, т.е. IP-адрес может отличаться от IP-адреса Шлюза или может быть таким же. Указывается IP-адрес именно Виртуального Сервера, если Шлюз скрывается за Виртуальным Сервером. • `vip port` – порт Виртуального Сервера для TERA (по умолчанию – `443`); • `vrf` – VRF для TERA (по умолчанию – `default`)
`set gw server <имя> vip <значение>`	Настройка Виртуального Сервера. Для параметра `vip` могут устанавливаться дополнительные параметры: • `ip` – IP-адрес Виртуального Сервера; • `port` – порт Виртуального Сервера
`set gw server <имя> vrf <значение>`	Имя VRF (по умолчанию – `default`)
`set gw server <имя> websocket <значение>`	Настройка подключения к компоненту «Клиент» Termidesk. Для параметра `websocket` могут устанавливаться дополнительные параметры: • `pingtimeout` – время ожидания (в секундах) соединения (значение от 0 до 100000, по умолчанию – `30`)

Объект ha

Доступные команды объекта `ha` приведены в таблице (см. таблицу Доступные команды объекта ha).

Таблица 49. Доступные команды объекта `ha`

Команда	Описание
`set ha`	Создание отказоустойчивой (высокодоступной) конфигурации со значениями по умолчанию

Команда	Описание
`set ha id <значение>`	Задание уникального числового идентификатора узлу.  Уникальные числовые идентификаторы не должны повторяться в одноранговом кластере. Возможные значения: от 0 до 255
`set ha interval <значение>`	Задание интервала (в миллисекундах) периодических запросов. Интервал должен быть одинаковым на всех устройствах. Значение по умолчанию: `500`
`set ha ip <IP-адрес>`	Указание IP-адреса, используемого для взаимодействия узлов кластера. Выбранный IP-адрес должен соответствовать типу «ha-type LOCAL»
`set ha cluster-ip <IP-адрес>`	Общий IP-адрес управления кластером, указывается на мастер-узле. Первоначально должен быть создан и настроен на мастер-узле. Выбранный IP-адрес должен соответствовать типу «ha-type SHARED»
`set ha port <порт>`	Задание UDP-порта, используемого для взаимодействия. Значение по умолчанию: `322`
`set ha ip-monitor ip <IP-адрес>`	Отслеживаемый IP-адрес для готовности узла к переходу в состояние `ACTIVE`. Для параметра `ip-monitor ip` может быть задан дополнительный параметр `vrf` (VRF по умолчанию – `default`)
`set ha remote <имя_узла>`	Задание имени соседнего узла, участвующего в отказоустойчивой (высокодоступной) конфигурации. Имя может быть любым
`set ha remote <имя_узла> <параметр> <значение>`	Задание параметров подключения к соседнему узлу, участвующему в отказоустойчивой (высокодоступной) конфигурации. Параметром может быть: • `id` – уникальный идентификатор соседнего узла; • `ip` – IP-адрес соседнего узла; • `port` – порт соседнего узла (по умолчанию – `322`); • `seckey` – пароль пользователя соседнего узла (так же пароль может быть получен из Хранилища секретов)

Объект `health-check`

Доступные команды объекта `health-check` приведены в таблице (см. таблицу Доступные команды объекта `health-check`).

Таблица 50. Доступные команды объекта `health-check`

Команда	Описание
`set health-check id <имя>`	Создание Проверки

Команда	Описание
`set health-check id <имя> <тип>`	Тип Проверки. Тип указывает на использование определенного протокола или способа создания Проверки, может быть: • `ICMP` – ICMP-Проверка; • `TCP` – TCP-Проверка; • `HTTP` – HTTP-Проверка; • `USER` – Пользовательская Проверка; • `COMBO` – Комбинированная Проверка
`set health-check id <имя> ICMP`	Создание ICMP-Проверки
Настройка ICMP-Проверки	
`set health-check id <имя> ICMP interval <значение>`	Время (в секундах), через которое будут отправляться запросы (по умолчанию – `10`)
`set health-check id <имя> ICMP source-ip <значение>`	IP-адрес источника, с которого отправляются запросы
`set health-check id <имя> ICMP success-try <значение>`	Количество успешных попыток, необходимых для Проверки (по умолчанию – `1`)
`set health-check id <имя> ICMP target-ip <значение>`	IP-адрес цели, к которой будут отправляться запросы
`set health-check id <имя> ICMP timeout <значение>`	Время ожидания (в секундах) ответа на запрос (по умолчанию – `5`)
`set health-check id <имя> ICMP try <значение>`	Количество повторных Проверок в случае отсутствия ответа (по умолчанию – `1`)
`set health-check id <имя> ICMP vrf <значение>`	Имя VRF (по умолчанию – `default`)
`set health-check id <имя> ICMP ha-monitor <значение>`	Включение или отключение отслеживания Проверки для готовности узла к переходу в состояние `ACTIVE` (по умолчанию – `false`)
`set health-check id <имя> TCP`	Создание TCP-Проверки. Для `TCP` могут устанавливаться параметры, аналогичные `ICMP`, а также дополнительные
Настройка TCP-Проверки	
`set health-check id <имя> TCP target-port <значение>`	Порт, на который будет отправлен запрос
`set health-check id <имя> HTTP`	Создание HTTP-Проверки. Для `HTTP` могут устанавливаться параметры, аналогичные `TCP`, а также дополнительные
`set health-check id <имя> HTTP headers <значение>`	Заголовок запроса, по которому выполняется Проверка
`set health-check id <имя> HTTP maintenance-codes <значение>`	Коды ответов для перевода Реального Сервера в режим технического обслуживания
`set health-check id <имя> HTTP method <значение>`	Метод запроса, по которому выполняется Проверка (по умолчанию – `HEAD`). Возможные значения: • `GET`– запрос с получением тела ответа; • `HEAD`– запрос с получением заголовка ответа; • `POST`– отправка данных с телом запроса
`set health-check id <имя> HTTP response-string <значение>`	Строка в ответе, по которой выполняется Проверка

Команда	Описание
`set health-check id <имя> HTTP reverse <значение>`	Инверсия результата Проверки (по умолчанию – `false`)
`set health-check id <имя> HTTP ssl-profile-id <значение>`	Профиль защищенного соединения, который будет использоваться при выполнении Проверки  Для установки Профиля защищенного соединения HTTPS-Проверки подходит только Клиентский SSL-Профиль.
`set health-check id <имя> HTTP status-codes <значение>`	Ожидаемые коды ответов (по умолчанию – `200`)
`set health-check id <имя> HTTP uri <значение>`	Путь, по которому выполняется Проверка (по умолчанию – `/`)
`set health-check id <имя> USER`	Создание Пользовательской Проверки. Для `USER` могут устанавливаться параметры, аналогичные `TCP`, а также дополнительные
Настройка Пользовательской Проверки	
`set health-check id <имя> USER script <значение>`	Имя файла скрипта, согласно которому будет выполняться Проверка
`set health-check id <имя> COMBO`	Создание Комбинированной Проверки
Настройка Комбинированной Проверки	
`set health-check id <имя> COMBO hc-ids <значение>`	Идентификатор базовой Проверки. Для параметра `hc-ids` может устанавливаться дополнительный параметр: • `weight` – Вес базовой Проверки
`set health-check id <имя> COMBO threshold <значение>`	Максимальный Вес суммы Проверок

Объект `http-profile`

Доступные команды объекта `http-profile` приведены в таблице (см. Доступные команды объекта `http-profile`).

Таблица 51. Доступные команды объекта `http-profile`

Команда	Описание
`set http-profile server <имя>`	Создание Серверного HTTP-Профиля. Серверный HTTP-Профиль определяет взаимодействие между пользователем и Termidesk Connect, где Termidesk Connect является сервером
Настройка Серверного HTTP-Профиля	
`set http-profile server <имя> error-reply <значение>`	Файл Сценария ошибок (по умолчанию – `error-reply.lua`)
`set http-profile server <имя> keep-alive <значение>`	Настройка проверки активности соединения
`set http-profile server <имя> max-header-count <значение>`	Максимальное число заголовков в запросе, при котором он считается валидным (по умолчанию – `128`)
`set http-profile server <имя> max-header-length <значение>`	Максимальный размер (в байтах) одного заголовка (по умолчанию – `24820`)
`set http-profile server <имя> max-headers-size <значение>`	Максимальный размер (в байтах) всех заголовков (по умолчанию – `32768`)

Команда	Описание
`set http-profile server <имя> read-timeout <значение>`	Время ожидания (в секундах) чтения из сокета (по умолчанию – `60`)
`set http-profile server <имя> body-size-limit <значение>`	Задание максимального размера (в байтах) тела HTTP-запроса (по умолчанию – `0`)
`set http-profile server <имя> description <значение>`	Комментарий, который будет привязан к Серверному HTTP-Профилю Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.
`set http-profile client <имя>`	Создание Клиентского HTTP-Профиля. Клиентский HTTP-Профиль определяет взаимодействие между Termidesk Connect и Реальным Сервером, где Termidesk Connect является клиентом
Настройка Клиентского HTTP-Профиля	
`set http-profile client <имя> allow-connect <значение>`	Активация или отключение проксирования метода CONNECT с переключением на TCP (по умолчанию – `false`)
`set http-profile client <имя> max-header-count <значение>`	Максимальное число заголовков в запросе, при котором он считается валидным (по умолчанию – `128`)
`set http-profile client <имя> max-header-length <значение>`	Максимальный размер (в байтах) одного заголовка (по умолчанию – `24820`)
`set http-profile client <имя> max-headers-size <значение>`	Максимальный размер (в байтах) всех заголовков (по умолчанию – `32768`)
`set http-profile client <имя> proxy-100 <значение>`	Активация или отключение обработки заголовка `Expect: 100-continue` (по умолчанию – `false`)
`set http-profile client <имя> read-timeout <значение>`	Время ожидания (в секундах) чтения из сокета (по умолчанию – `60`)
`set http-profile client <имя> upgrade-types <значение>`	Значения (может быть несколько) из заголовка Upgrade, для которых разрешена смена протокола (по умолчанию – `websocket`)
`set http-profile client <имя> body-size-limit <значение>`	Задание максимального размера (в байтах) тела HTTP-ответа (по умолчанию – `0`)
`set http-profile client <имя> description <значение>`	Комментарий, который будет привязан к Клиентскому HTTP-Профилю Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.

Объект `interfaces`

Объект удален, начиная с Termidesk Connect версии 1.1. Настройка интерфейсов осуществляется через объект `ethernet` (см. подраздел **Объект `ethernet`**).

Объект `ip`

Доступные команды объекта `ip` приведены в таблице (см. Доступные команды объекта `ip`).

Таблица 52. Доступные команды объекта ip

Команда	Описание
set ip address <IP-адрес> <длина_префикса>	Добавление IP-адреса в формате CIDR
set ip address <IP-адрес> <длина_префикса> if-ethernet <имя_интерфейса>	Назначение IP-адреса существующему сетевому интерфейсу
set ip address <IP-адрес> <длина_префикса> ha-type <тип>	Назначение режима использования IP-адреса при отказоустойчивой конфигурации, где <тип> может быть: • LOCAL – IP-адрес используется в локальной конфигурации, не синхронизируется для отказоустойчивой конфигурации; • SHARED – IP-адрес синхронизируется для отказоустойчивой конфигурации
set ip address <IP-адрес> <длина_префикса> if-vlan <имя>	Назначение IP-адреса существующему интерфейсу VLAN
set ip route <имя_VRF> <сеть><длина_префикса> <IP-адрес шлюза>	Создание маршрута. По умолчанию существует VRF default – создание маршрута по умолчанию
set ip route <имя_VRF> <сеть><длина_префикса> <IP-адрес_шлюза> ha-type <тип>	Назначение режима использования IP-адреса при отказоустойчивой конфигурации, где <тип> может быть: • LOCAL – маршрут используется в локальной конфигурации, не синхронизируется для отказоустойчивой конфигурации; • SHARED – маршрут синхронизируется для отказоустойчивой конфигурации

Объект ipset

Доступные команды объекта ipset приведены в таблице (см. таблицу Доступные команды объекта ipset).

Таблица 53. Доступные команды объекта ipset

Команда	Описание
set ipset id <имя>	Создание IP-Фонда с IP-адресами WARNING: Имя создаваемого объекта не должно содержать дефисов, нижнее подчеркивание использовать разрешено.
set ipset id <имя> ips <IP-адреса>	Добавление IP-адресов в IP-Фонд. Команда позволяет за один раз добавить только один IP-адрес
set ipset id <имя> vrf <имя VRF>	Добавление VRF в IP-Фонд (по умолчанию – default)

Объект krb5

Доступные команды объекта krb5 приведены в таблице (см. Доступные команды объекта krb5).

Таблица 54. Доступные команды объекта krb5

Команда	Описание
set krb5 server <имя>	Создание KDC-Сервера

Команда	Описание
`set krb5 server <имя> domain <IP-адрес>`	Настройка IP-адреса для подключения к серверу
`set krb5 server <имя> port <порт>`	Настройка порта для подключения к серверу (по умолчанию – 88)
`set krb5 server <имя> timeout <значение>`	Настройка времени (в секундах) ожидания ответа от сервера KDC (по умолчанию – 30)

Объект lbs

Доступные команды объекта lbs приведены в таблице (см. таблицу Доступные команды объекта lbs).

Таблица 55. Доступные команды объекта lbs

Команда	Описание
`set lbs TCP <имя>`	Создание Сервера Балансировки для протокола TCP
Настройка Сервера Балансировки для протокола TCP	
`set lbs TCP <имя> rs-pool-id <значение>`	Задание Группы Реальных Серверов, к которой будет привязан Сервер Балансировки
`set lbs TCP <имя> min-rs <значение>`	Задание минимального количества действующих Реальных Серверов, необходимых для обработки запросов (по умолчанию – 1)

Команда	Описание
`set lbs TCP <имя> algorithm <значение>`	Задание алгоритма балансировки. Для параметра `algorithm` могут устанавливаться значения (по умолчанию – `LEASTCONN`): • `ROUNDROBIN` – подключения пользователей в этом случае распределяются последовательно между Реальными Серверами, что обеспечивает их равномерное распределение; • `LEASTCONN` – подключения пользователей в этом случае распределяются оптимизировано, с учетом количества текущих активных соединений на каждом Реальном Сервере. Для подключения пользователя выбирается Реальный Сервер с наименьшим количеством текущих активных соединений, что обеспечивает более равномерное распределение нагрузки и помогает избежать перегрузки отдельных Реальных Серверов; • `WEIGHTEDROUNDROBIN` – подключения пользователей в этом случае распределяются последовательно между Реальными Серверами пропорционально их Весу, что обеспечивает их равномерное распределение; • `WEIGHTEDLEASTCONN` – подключения пользователей в этом случае распределяются оптимизировано, с учетом соотношения количества текущих активных соединений к Весу для каждого Реального Сервера, что помогает избежать перегрузки отдельных Реальных Серверов; • `WEIGHTEDLEASTCONNECTTIME` – подключения пользователей в этом случае распределяются оптимизировано, с учетом соотношения наименьшего среднего времени соединения и количества текущих сессий к Весу для каждого Реального Сервера; • `RANDOM` – для подключения пользователей в этом случае выбирается случайный Реальный Сервер; • `POWEROFTWORANDOM` – для подключения пользователей в этом случае выбирается Реальный Сервер с наименьшим числом соединений из двух Реальных Серверов, выбранных случайным образом
`set lbs TCP <имя> leastconn-param starttime <значение>`	Задание времени (в секундах) смены алгоритма с `LEASTCONN` на `ROUNDROBIN` или с `WEIGHTEDLEASTCONN` на `WEIGHTEDROUNDROBIN` при изменении количества серверов с состоянием «В работе» в привязанной Группе Реальных Серверов (по умолчанию – `0`)

Команда	Описание

`set lbs TCP <имя> persistence <значение>`	Задание параметра, определяющего постоянство подключения пользователя к Реальному Серверу.
	
	Для параметра `persistence` могут устанавливаться дополнительные параметры:
	
	• `algorithm` – алгоритм привязки пользователя к Реальному Серверу (по умолчанию – `NONE`). Может принимать значения:
	  ◦ `NONE` – привязка не используется;
	  ◦ `IPSOURCE` – привязка по IP-адресу источника запроса, т.е. IP-адресу пользователя;
	  ◦ `SSLSESSION` – привязка пользователя к Реальному Серверу по идентификатору SSL-сессии, являющегося частью процесса установления соединения с выбранным сервером. Последующие запросы пользователя с данным идентификатором отправляются на ранее выбранный Реальный Сервер;
	• `ipsource-param timeout` – время ожидания (в секундах) для алгоритма `IPSOURCE`, в течение которого повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер (по умолчанию – `60`);
	• `sslsession-param timeout` – время ожидания (в секундах) для алгоритма `SSLSESSION`, в течение которого повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер (по умолчанию – `60`)

`set lbs TCP <имя> persistence-profile <значение>`	Задание Профиля сохранения сессий
`set lbs TCP <имя> ssl-profile-id <значение>`	Задание Клиентского SSL-Профиля, который будет использоваться для функционала SSL Offload
`set lbs TCP <имя> tcp-profile-id <значение>`	Задание Клиентского TCP-Профиля (по умолчанию – `tcpp-client-default`)
`set lbs TCP <имя> use-cip <значение>`	Управление режимом подмены IP-адреса клиента (по умолчанию – `false`)
`set lbs TCP <имя> re-balancing <значение>`	Настройка функции перебалансировки. Для параметра `re-balancing` могут устанавливаться дополнительные параметры: • `enable` – разрешение или запрет перебалансировки. Перебалансировка (попытка выбрать другой Реальный Сервер) осуществляется в случае ошибки подключения к Реальному Серверу (по умолчанию – `false`); • `max-attempts` – максимальное количество попыток перебалансировки (значение от 1 до 10, по умолчанию – `1`)
`set lbs TCP <имя> ha-monitor <значение>`	Включение или отключение отслеживания состояния Сервера Балансировки для готовности узла к переходу в состояние `ACTIVE` (по умолчанию – `false`)
`set lbs TCP <имя> rl-profile-id <значение>`	Задание Профиля ограничения скорости для привязки к Серверу Балансировки

Команда	Описание
`set lbs TCP <имя> description <значение>`	Задание комментария, который будет привязан к Серверу Балансировки Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.
`set lbs HTTP <имя>`	Создание Сервера Балансировки для протокола HTTP. Параметры идентичны протоколу TCP, включая дополнительные
Настройка Сервера Балансировки для протокола HTTP	
`set lbs HTTP <имя> algorithm <значение>`	Задание алгоритма балансировки. Для параметра `algorithm` могут устанавливаться значения (по умолчанию – `LEASTCONN`): • `ROUNDROBIN` – подключения пользователей в этом случае распределяются последовательно между Реальными Серверами, что обеспечивает их равномерное распределение; • `LEASTCONN` – подключения пользователей в этом случае распределяются оптимизировано, с учетом количества текущих активных соединений на каждом Реальном Сервере. Для подключения пользователя выбирается Реальный Сервер с наименьшим количеством текущих активных соединений, что обеспечивает более равномерное распределение нагрузки и помогает избежать перегрузки отдельных Реальных Серверов; • `WEIGHTEDROUNDROBIN` – подключения пользователей в этом случае распределяются последовательно между Реальными Серверами пропорционально их Весу, что обеспечивает их равномерное распределение; • `WEIGHTEDLEASTCONN` – подключения пользователей в этом случае распределяются оптимизировано, с учетом соотношения количества текущих активных соединений к Весу для каждого Реального Сервера, что помогает избежать перегрузки отдельных Реальных Серверов; • `WEIGHTEDLEASTCONNECTTIME` – подключения пользователей в этом случае распределяются оптимизировано, с учетом соотношения наименьшего среднего времени соединения и количества текущих сессий к Весу для каждого Реального Сервера; • `WEIGHTEDLEASTRESPONSETIME` – подключения пользователей в этом случае распределяются оптимизировано, с учетом соотношения наименьшего среднего времени соединения, наименьшим средним временем получения первого байта ответа и количества текущих сессий к Весу для каждого Реального Сервера; • `RANDOM` – для подключения пользователей в этом случае выбирается случайный Реальный Сервер; • `POWEROFTWORANDOM` – для подключения пользователей в этом случае выбирается Реальный Сервер с наименьшим числом соединений из двух Реальных Серверов, выбранных случайным образом

Команда	Описание

`set lbs HTTP <имя> persistence <значение>`	Задание параметра, определяющего постоянство подключения пользователя к Реальному Серверу.
	
	Для параметра `persistence` могут устанавливаться дополнительные параметры:
	
	• `algorithm` – может принимать дополнительные значения:
	  ▫ `COOKIEINSERT` – привязка по cookie, который помещается в HTTP-ответ, направляемый пользователю. Обеспечивает постоянство выбора Реального Сервера путем автоматической вставки cookie в HTTP-ответ. Последующие запросы пользователя с этим cookie перенаправляются на тот же Реальный Сервер. В случае, когда пользователь не сохраняет cookie в HTTP, его запросы не будут содержать cookie для отправки Termidesk Connect. Для такого пользователя данный метод привязки не подходит, требуется настройка альтернативного метода;
	  ▫ `HEADER` – привязка по значению заголовка, указанного в конфигурации. Этот алгоритм независим от TCP/IP параметров подключения;
	  ▫ `COOKIE` – привязка по cookie, который получен в ответе Реального Сервера. Значение cookie, указанного в конфигурации, записывается в персистентную таблицу. Последующие запросы пользователя с данным cookie отправляются на этот Реальный Сервер;
	• `ci-param timeout` – время ожидания (в секундах) для алгоритма `COOKIEINSERT`, в течение которого повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер (по умолчанию – `60`);
	• `header-param header-name` – заголовок для алгоритма `HEADER`, по которому повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер;
	• `header-param timeout` – время ожидания (в секундах) для алгоритма `HEADER`, в течение которого повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер (по умолчанию – `60`);
	• `cookie-param cookie-name` – имя cookie для алгоритма `COOKIE`, который ожидается в ответе Реального Сервера для повторного подключения пользователя на этот Реальный Сервер;
	• `cookie-param timeout` – время ожидания (в секундах) для алгоритма `COOKIE`, в течение которого повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер (по умолчанию – `60`)

`set lbs HTTP <имя> http-profile-id <значение>`	Задание Клиентского HTTP-Профиля (по умолчанию – `httpp-client-default`)
`set lbs HTTP <имя> luarules <значение>`	Настройка приоритета Сценария и файла Сценария
`set lbs RAPID-TCP <имя>`	Создание Сервера Балансировки для протокола RAPID-TCP. Параметры идентичны протоколу TCP за исключением `persistence-profile` и `rl-profile-id` (не задается для RAPID TCP), а так же включая дополнительные и измененный `persistence`
Настройка Сервера Балансировки для протокола RAPID-TCP	
`set lbs RAPID-TCP <имя> connection-idle <значение>`	Задание времени жизни сессии (в секундах) при бездействии, т.е. отсутствии пакетов (по умолчанию – `60`)

Команда	Описание
`set lbs RAPID-TCP <имя> fin-timeout <значение>`	Задание времени жизни сессии (в секундах) после получения управляющего флага FIN в заголовке пакета (по умолчанию – `2`)
`set lbs RAPID-TCP <имя> dsr-mode <значение>`	Управление режимом работы Сервера Балансировки. Для параметра `dsr-mode` могут устанавливаться значения (по умолчанию – `OFF`): • `OFF` – режим работы, при котором DSR отключен; • `MAC` – режим с подменой MAC-адресов (L2 DSR). В этом режиме IP-адреса во входящем пакете остаются неизменными, Termidesk Connect подменяет в пакете только MAC-адреса (MAC-адрес источника – Termidesk Connect, MAC-адрес назначения – Реальный Сервер) и отправляет этот пакет по MAC-адресам на сервер; • `IPIP` – режим (L3 DSR), при котором входящий пакет инкапсулируется в IPIP-туннель и направляется на Реальный Сервер. Далее Реальный Сервер декапсулирует IP-пакет и видит IP-адрес пользователя и IP-адрес Виртуального Сервера

`set lbs RAPID-TCP <имя> persistence <значение>`	Задание параметра, определяющего постоянство подключения пользователя к Реальному Серверу.
	
	Для параметра `persistence` устанавливаются параметры:
	
	• `algorithm` – алгоритм привязки пользователя к Реальному Серверу (по умолчанию – `NONE`). Может принимать значения:
	  ◦ `NONE` – привязка не используется;
	  ◦ `IPSOURCE` – привязка по IP-адресу источника запроса, т.е. IP-адресу пользователя;
	• `ipsource-param timeout` – время ожидания (в секундах) для `IPSOURCE`, в течение которого повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер (по умолчанию – `60`);
	• `ipsource-param ipset-src-persist` – управление режимом сохранения IP-адреса из IP-Фонда для взаимодействия с Реальным Сервером для `IPSOURCE` (по умолчанию – `false`)

`set lbs RAPID-TCP <имя> ttl <значение>`	Задание параметров TTL IP-пакета в заголовке. Для параметра `ttl` устанавливаются: • `to-rs` – TTL пакета к Реальному Серверу (по умолчанию – `0`); • `to-client` – TTL пакета к источнику запроса (по умолчанию – `0`)
`set lbs RAPID-UDP <имя>`	Создание Сервера Балансировки для протокола RAPID-UDP. Параметры идентичны протоколу RAPID-TCP, за исключением `fin-timeout` (не задается для RAPID-UDP)

Объект `ldap`

Доступные команды объекта `ldap` приведены в таблице (см. таблицу Доступные команды объекта `ldap`).

Таблица 56. Доступные команды объекта `ldap`

Команда	Описание
`set ldap type <тип_службы_каталогов>`	Указание типа подключаемой службы каталогов. Возможные значения: • `AD` – служба каталогов Active Directory Domain Services; • `FreeIPA` – служба каталогов FreeIPA; • `OpenLDAP` – служба каталогов OpenLDAP Directory Services
`set ldap domain <доменное_имя_сервера_или_IP>`	Указание доменного имени сервера службы каталогов или его IP-адреса
`set ldap timeout <значение>`	Указание времени ожидания (в секундах) ответа от службы каталогов (по умолчанию – `30`)
`set ldap security <тип_безопасности>`	Указание типа безопасности для подключения к службе каталогов (по умолчанию – `TEXT`). Возможные значения: • `TEXT` – незащищенное подключение; • `SSL` – защищенное подключение. Перед обменом данными будет установлена TLS-сессия
`set ldap ca-cert <имя_файла>`	Указание файла сертификата УЦ – опционально, если используется защищенное подключение к службе каталогов. Файл сертификата УЦ должен быть предварительно загружен на Termidesk Connect (см. подраздел **TLS**). Так же файл может быть получен из Хранилища секретов
`set ldap port <значение>`	Указание порта для подключения к службе каталогов (по умолчанию – `389`). Возможные стандартные значения: • `389` – используется, если доступ к службе каталогов осуществляется по протоколу LDAP (незащищенное подключение); • `636` – используется, если доступ к службе каталогов осуществляется по протоколу LDAPS (защищенное подключение); • `3268` – альтернативный порт. Используется, если доступ к службе каталогов осуществляется по протоколу LDAP (незащищенное подключение); • `3269` – альтернативный порт. Используется, если доступ к службе каталогов осуществляется по протоколу LDAPS (защищенное подключение)

Команда	Описание
`set ldap base-dn <значение>`	Указание корня поиска в службе каталогов (Base DN). Вводимое значение не должно содержать пробелов: • в начале и конце строки; • рядом с разделителями (запятыми); • в элементах пути (например, «DC=company name,DC=de»). Примеры: • для поиска в домене example.loc: `DC=example,DC=loc`; • для поиска только в подразделении Users домена example.loc: `OU=Users,DC=example,DC=loc`
`set ldap administrator-bind-dn <значение>`	Указание учетной записи (с правами на чтение) в формате DN, используемой для подключения к службе каталогов. Вводимое значение не должно содержать пробелов: • в начале и конце строки; • рядом с разделителями (запятыми); • в элементах пути (например, «DC=company name,DC=de»). Примеры: • Active Directory Domain Services: `CN=Administrator,OU=Users,DC=example,DC=loc;` • FreeIPA: `UID=admin,CN=users,CN=accounts,DC=example,DC=loc`; • OpenLDAP Directory Services: `CN=admin,DC=example,DC=loc`. Так же учетная запись может быть получена из Хранилища секретов
`set ldap password <пароль>`	Указание пароля учетной записи. Так же пароль может быть получен из Хранилища секретов
`set ldap user-name-attribute <атрибут_имени_пользователя>`	Указание атрибута уникального имени или идентификатора пользователя в службе каталогов (по умолчанию – `userPrincipalName`). Возможные значения: • `uid` – уникальный идентификатор учетной записи; • `userPrincipalName` – логин пользователя в формате `user@example.loc`; • `SamAccountName` – короткое имя пользователя; • `cn` – отображаемое имя пользователя (иногда – имя для входа).

Команда	Описание
`set ldap group-attribute <атрибут_группы>`	Указание атрибута группы (по умолчанию – `memberOf`). Возможные значения: • `memberOf` – список участников группы, в котором каждый участник указывается в виде полного DN. Универсальный атрибут; • `uniqueMember` – аналог атрибута выше, но используется в некоторых реализациях LDAP, например, OpenLDAP Directory Services

Объект `logging`

Доступные команды объекта `logging` приведены в таблице (см. Доступные команды объекта `logging`).

Таблица 57. Доступные команды объекта `logging`

Команда	Описание
`set logging level <уровень>`	Переключение уровня журналирования. Возможные значения: • `INFO` – журналируется общая информация о работе; • `DEBUG` – журналируется отладочная информация о работе

Объект `nacm`

Доступные команды объекта `nacm` приведены в таблице (см. Доступные команды объекта `nacm`).

Таблица 58. Доступные команды объекта `nacm`

Команда	Описание
`set nacm`	Создание конфигурации NETCONF со значениями по умолчанию
Настройка конфигурации NETCONF	
`set nacm enable-nacm <значение>`	Проверка активации контроля доступа (по умолчанию – `true`)
`set nacm exec-default <значение>`	Управление действием по умолчанию для выполнения команд, если явное правило отсутствует (по умолчанию – `permit`)
`set nacm read-default <значение>`	Управление доступом к чтению данных, если явное правило отсутствует (по умолчанию – `permit`)
`set nacm write-default <значение>`	Управление доступом к записи данных, если явное правило отсутствует (по умолчанию – `deny`)

Команда	Описание

`set nacm rule-list <значение>`	Управление набором правил доступа групп пользователей
	
	Конфигурация параметра `rule-list` может содержать объекты для определенного списка правил:
	
	• `group` – группа пользователей, к которой применяется набор правил доступа;
	• `rule` – непосредственно правило доступа для набора, в котором задаются:
	  ◦ `name` – имя правила доступа;
	  ◦ `path` – путь к элементам конфигурации Termidesk Connect (по умолчанию – `/`). Предполагает выражения для указания конкретных путей обхода XML-дерева, расположенного в хранилище конфигурации. Осуществляется на языке запросов XPath, определенного в RFC 5261. По заданному выражению определяются права доступа к конкретным элементам конфигурации. Значение `/` указывает на все возможное содержимое хранилища данных;
	  ◦ `access-operations` – операции доступа, которые будут связаны с правилом (по умолчанию – `*`);
	  ◦ `action` – действия по доступу, которые будут связаны с правилом;
	  ◦ `comment` – текстовое описание правила доступа

Объект `ntp`

Доступные команды объекта `ntp` приведены в таблице (см. Доступные команды объекта `ntp`).

Таблица 59. Доступные команды объекта `ntp`

Команда	Описание
`set ntp server <IP-адрес_или_FQDN>`	Добавление NTP-сервера

Объект `openbao`

Доступные команды объекта `openbao` приведены в таблице (см. таблицу Доступные команды объекта `openbao`).

Таблица 60. Доступные команды объекта `openbao`

Команда	Описание
`set openbao auth <параметр> <значение>`	Настройка параметров аутентификации Termidesk Connect в Хранилище секретов, параметром может быть: • `approle` – идентификатор роли (RoleID) для метода аутентификации AppRole; • `mount` – путь к методу аутентификации (auth method) в Хранилище секретов; • `secret-file` – путь к файлу, содержащему идентификатор секрета (SecretID) или токен для его раскрытия (unwrapping); • `wrapping-token` – флаг использования упакованного токена (wrapped-token) для идентификатора секрета (SecretID) (по умолчанию – `true`). Возможные значения: • `true` – содержимое файла будет предварительно раскрыто (unwrapped) для получения действительного идентификатора секрета (SecretID); • `false` – содержимое файла будет упаковано (wrapped) для получения действительного идентификатора секрета (SecretID)
`set openbao ns <значение>`	Пространство имен (Namespace), заданное в Хранилище секретов
`set openbao renew-cert-interval <значение>`	Интервал (в секундах) обновления сертификатов (по умолчанию – `86400`). Termidesk Connect с заданной периодичностью будет отправлять запросы в Хранилище секретов для получения новых сертификатов
`set openbao renew-secret-interval <значение>`	Интервал (в секундах) обновления секретов (по умолчанию – `1800`). Termidesk Connect с заданной периодичностью будет отправлять запросы в Хранилище секретов для получения новых значений секретов
`set openbao renew-token-interval <значение>`	Интервал (в секундах) обновления токена авторизации (по умолчанию – `3600`)
`set openbao request-timeout <значение>`	Максимально допустимое время (в секундах) ожидания ответа HTTP-запроса от API-сервера (по умолчанию – `60`)
`set openbao retry-interval <значение>`	Интервал (в секундах) повторной отправки HTTP-запроса при возникновении ошибки (по умолчанию – `10`)
`set openbao ssl <параметр> <значение>`	Настройка параметров SSL-подключения к Хранилищу конфигурации, параметром может быть: • `ca-cert` – файл сертификата УЦ (указывается полный путь и имя файла с расширением). Пример значения параметра: `/dev/shm/server.cert`; • `cert` – файл сертификата (указывается полный путь и имя файла с расширением); • `key` – файл ключа (указывается полный путь и имя файла с расширением)
`set openbao url <значение>`	URL-адрес Хранилища секретов. Пример значения параметра: `\hhttp://192.0.2.10:8200`
`set openbao vrf <значение>`	Имя VRF для подключения к Хранилищу секретов (по умолчанию – `default`)

Объект `persistence-profile`

Доступные команды объекта `persistence-profile` приведены в таблице (см. таблицу Доступные команды объекта `persistence-profile`).

Таблица 61. Доступные команды объекта `persistence-profile`

Команда	Описание
`set persistence-profile id <имя>`	Создание Профиля сохранения сессий
`set persistence-profile id <имя> algorithm <значение>`	Тип Профиля сохранения сессий (по умолчанию – `IPSOURCE`), может принимать значения: • `COOKIE` – привязка по cookie, который получен в ответе Реального Сервера. Значение cookie, указанного в конфигурации, записывается в персистентную таблицу. Последующие запросы пользователя с данным cookie отправляются на этот Реальный Сервер; • `HEADER` – привязка по значению заголовка, указанного в конфигурации. Этот алгоритм независим от TCP(IP) параметров подключения; • `IPSOURCE` – привязка по IP-адресу источника запроса, т.е. IP-адресу пользователя; • `SSLSESSION` – привязка пользователя к Реальному Серверу по идентификатору SSL-сессии, являющегося частью процесса установления соединения с выбранным сервером. Последующие запросы пользователя с данным идентификатором отправляются на ранее выбранный Реальный Сервер
`set persistence-profile id <имя> cookie-param <параметр> <значение>`	Настройка параметров для Профиля сохранения сессий с типом `COOKIE`. Параметром может быть: • `cookie-name` – имя cookie, который ожидается в ответе Реального Сервера для повторного подключения пользователя на этот Реальный Сервер; • `timeout` – время ожидания (в секундах) для типа `COOKIE`, в течение которого повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер (по умолчанию – `60`)
`set persistence-profile id <имя> header-param <параметр> <значение>`	Настройка параметров для Профиля сохранения сессий с типом `HEADER`. Параметром может быть: • `header-name` – имя HTTP-заголовка, по которому повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер; • `timeout` – время ожидания (в секундах) для типа `HEADER`, в течение которого повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер (по умолчанию – `60`)
`set persistence-profile id <имя> ipsource-param timeout <значение>`	Время ожидания (в секундах) для типа `IPSOURCE`, в течение которого повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер (по умолчанию – `60`)
`set persistence-profile id <имя> sslsession-param timeout <значение>`	Время ожидания (в секундах) для типа `SSLSESSION`, в течение которого повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер (по умолчанию – `60`)

Объект `restconf`

Доступные команды объекта `restconf` приведены в таблице (см. Доступные команды объекта `restconf`).

Таблица 62. Доступные команды объекта `restconf`

Команда	Описание
`set restconf`	Создание конфигурации RESTCONF со значениями по умолчанию
`set restconf auth-type <тип>`	Назначение типа аутентификации, где `<тип>` может принимать значения: • `client-certificate` – тип аутентификации с использованием клиентского сертификата для идентификации пользователя; • `user` – тип аутентификации с использованием имени пользователя и пароля для доступа к API. Значение по умолчанию: `user`
`set restconf debug <значение>`	Управление отладкой и мониторингом работы RESTCONF, где `<debug>` – уровень детализации отладочных сообщений, может принимать значения: • `0` – отключение отладки (по умолчанию); • `1` – базовая информация о работе системы и основных событиях; • `2` – более подробная информация, включая некоторые внутренние события и предупреждения; • `3` – полная отладочная информация, включая детализированные логи о внутреннем состоянии, вызовах функций и т.п. Значение по умолчанию: `0`
`set restconf enable <false/true>`	Управление режимом RESTCONF. Значение по умолчанию: `false`
`set restconf enable-core-dump <false/true>`	Управление режимом создания снимка (дампа) состояния памяти ядра. Значение по умолчанию: `false`
`set restconf log-destination <тип>`	Указание формата записи журнала, где `<тип>` может принимать значения: • `syslog` – запись в службу `syslog`; • `file` – запись в файл. Значение по умолчанию: `syslog`
`set restconf pretty <false/true>`	Управление режимом форматирования заголовков в формате `XML` или `JSON` Значение по умолчанию: `true`
`set restconf server-ca-cert-path <путь>`	Указание файла сертификата корневого удостоверяющего центра (CA)
`set restconf server-cert-path <путь>`	Указание файла сертификата сервера
`set restconf server-key-path <путь>`	Указание закрытого ключа, который соответствует сертификату сервера

Команда	Описание
`set restconf socket <идентификатор> <IP-адрес> <порт>`	Настройка соединения к API в программном пространстве: • `<идентификатор>` – уникальный идентификатор, который определяет область применения API; • `<IP-адрес>` – IP-адрес сервера, на котором располагается API; • `<порт>` – номер порта, через который осуществляется доступ к API на указанном сервере. Пример команды: `set restconf socket default 127.0.0.1 8080`
`set restconf timeout <значение>`	Длительность (в секундах) сессии пользователя. Значение по умолчанию: `0` (длительность не ограничена)

Объект `rl-profile`

Доступные команды объекта `rl-profile` приведены в таблице (см. таблицу Доступные команды объекта `rl-profile`).

Таблица 63. Доступные команды объекта `rl-profile`

Команда	Описание
`set rl-profile lbs <имя>`	Создание Профиля ограничения скорости для Сервера Балансировки
`set rl-profile lbs <имя> in <параметр> <значение>`	Настройка входящего трафика Профиля ограничения скорости для Сервера Балансировки, параметром может быть: • `bandwidth` – ограничение скорости передачи данных (битов в секунду) от пользователя к Серверу Балансировки; • `pps` – ограничение количества пакетов в секунду от пользователя к Серверу Балансировки. Для параметров устанавливаются дополнительные параметры: • `burst` – порог всплеска; • `rate`- порог скорости
`set rl-profile lbs <имя> description <значение>`	Комментарий, который будет привязан к Профилю ограничения скорости для Сервера Балансировки Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.
`set rl-profile rs <имя>`	Создание Профиля ограничения скорости для Реального Сервера

Команда	Описание
`set rl-profile rs <имя> in <параметр> <значение>`	Настройка входящего трафика Профиля ограничения скорости для Реального Сервера, параметром может быть: • `bandwidth` – ограничение скорости получения данных (битов в секунду) от Реального Сервера; • `pps` – ограничение количества пакетов в секунду от Реального Сервера. Для параметров устанавливаются дополнительные параметры: • `burst` – порог всплеска; • `rate`- порог скорости
`set rl-profile rs <имя> out <параметр> <значение>`	Настройка исходящего трафика Профиля ограничения скорости для Реального Сервера, параметром может быть: • `bandwidth` – ограничение скорости передачи данных (битов в секунду) от Termidesk Connect к Реальному Серверу; • `pps` – ограничение количества пакетов в секунду от Termidesk Connect к Реальному Серверу. Для параметров устанавливаются дополнительные параметры: • `burst` – порог всплеска; • `rate`- порог скорости
`set rl-profile rs <имя> description <значение>`	Комментарий, который будет привязан к Профилю ограничения скорости для Реального Сервера Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.
`set rl-profile vs <имя>`	Создание Профиля ограничения скорости для Виртуального Сервера
`set rl-profile vs <имя> in <параметр> <значение>`	Настройка входящего трафика Профиля ограничения скорости для Виртуального Сервера, параметром может быть: • `bandwidth` – ограничение скорости передачи данных (битов в секунду) от пользователя к Виртуальному Серверу; • `pps` – ограничение количества пакетов в секунду от пользователя к Виртуальному Серверу; • `tcp-conn` – ограничение скорости установленных TCP-сессий (сессий в секунду) от пользователя к Виртуальному Серверу. При превышении заданного количества TCP-сессий новые сессии будут отброшены. Для параметров устанавливаются дополнительные параметры: • `burst` – порог всплеска; • `rate`- порог скорости

Команда	Описание
`set rl-profile vs <имя> description <значение>`	Комментарий, который будет привязан к Профилю ограничения скорости для Виртуального Сервера  Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.

Объект `rs-pool`

Доступные команды объекта `rs-pool` приведены в таблице (см. Доступные команды объекта `rs-pool`).

Таблица 64. Доступные команды объекта `rs-pool`

Команда	Описание
`set rs-pool id <имя>`	Создание Группы Реальных Серверов  Имя создаваемого объекта не должно содержать дефисов, нижнее подчеркивание использовать разрешено.
Настройка Группы Реальных Серверов	
`set rs-pool id <имя> hc-id <значение>`	Проверка, которая будет использоваться для Группы Реальных Серверов
`set rs-pool id <имя> ipset-id <значение>`	IP-Фонд
`set rs-pool id <имя> rs <значение>`	Узел Реального Сервера, добавляемый по IP-адресу и порту, для привязки к Группе Реальных Серверов. При этом: • может быть задано несколько узлов; • если порт из запроса должен быть сохранен и передаться на Реальный Сервер без изменений, то указывается `0`. Для параметра `rs` дополнительно могут быть заданы: • `weight` – Вес Реального Сервера (по умолчанию – `1`); • `state` – состояние Реального Сервера

Команда	Описание
`set rs-pool id <имя> rs-domain <значение>`	Узел Реального Сервера, добавляемый по доменному имени и порту, для привязки к Группе Реальных Серверов. При этом: • может быть задано несколько узлов; • если порт из запроса должен быть сохранен и передаться на Реальный Сервер без изменений, то указывается `0`. Для параметра `rs-domain` дополнительно могут быть заданы: • `TTL` – время жизни (в секундах) информации о доменном имени (по умолчанию – `60`); • `autoscale` – автоматическое масштабирование Группы Реальных Серверов (по умолчанию – `false`); • `weight` – Вес Реального Сервера (по умолчанию – `1`); • `state` – состояние Реального Сервера
`set rs-pool id <имя> rl-profile-id <значение>`	Профиль ограничения скорости
`set rs-pool id <имя> description <значение>`	Комментарий, который будет привязан к Группе Реальных Серверов. Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.
`set rs-pool id <имя> maintenance-timeout <значение>`	Время (в секундах), по истечении которого все сессии (если они остались), сбросятся, а записи привязки сессии пользователя удалятся

Объект `snmp`

Доступные команды объекта `snmp` приведены в таблице (см. Доступные команды объекта `snmp`).

Таблица 65. Доступные команды объекта `snmp`

Команда	Описание
`set snmp`	Настройка протокола SNMP для версий 1 и 2c
`set snmp community <имя> <исходный_узел>`	Сообщество для доступа к данным SNMP: • `<имя>` – имя сообщества; • `<исходный_узел>` – IP-адрес или подсеть, с которых будет осуществлен доступ к сообществу (по умолчанию – `0.0.0.0/0`)
`set snmp community <имя> <исходный_узел> access <доступ>`	Уровень доступа сообщества (по умолчанию – `RO`). Для `<доступ>` могут устанавливаться значения: • `RO` – доступ на чтение; • `RW` – доступ на чтение и запись

Команда	Описание
`set snmp description <значение>`	Описание SNMP-сервиса (по умолчанию – `Termidesk Connect`) Для описания допустимы только латинские буквы. В случае использования пробелов следует заключать текст описания в двойные кавычки.
`set snmp listener <ip-адрес> <порт> <протокол>`	Интерфейс, на котором SNMP будет ожидать запросы: • `<ip-адрес>` – IP-адрес, на котором будет работать SNMP; Указывается настроенный IP-адрес с типом `Shared`. • `<порт>` – порт, на котором будет работать SNMP (по умолчанию – `161`); • `<протокол>` – протокол, по которому будет работать SNMP (по умолчанию – `UDP`). Для `<протокол>` могут устанавливаться значения: • `TCP` – протокол TCP; • `UDP` – протокол UDP
`set snmp location <значение>`	Условное описание физического расположения устройства (по умолчанию – `DC`) Для описания допустимы только латинские буквы. В случае использования пробелов следует заключать текст описания в двойные кавычки.

Объект `snmpv3`

Доступные команды объекта `snmpv3` приведены в таблице (см. Доступные команды объекта `snmpv3`).

Таблица 66. Доступные команды объекта `snmpv3`

Команда	Описание
`set snmpv3`	Настройка протокола SNMPv3
`set snmpv3 engine-id-type <идентификатор>`	Идентификатор агента SNMP для сценариев, где требуется его уникальность. Возможные значения: • `1` – IPv4-адрес; • `2` – IPv6-адрес; • `3` – MAC-адрес Обратите внимание, что изменение IP-адреса (или замена сетевой карты) может вызвать проблемы.
`set snmpv3 user <имя>`	Добавление пользователя для SNMPv3 (доступно значение от 1 до 64 символов). Так же имя пользователя может быть получено из Хранилища секретов

Команда	Описание
Настройка пользователя для SNMPv3	
`set snmpv3 user <имя> auth-protocol <значение>`	Протокол аутентификации (по умолчанию – `NONE`). Для параметра `auth-protocol` могут устанавливаться следующие значения: • `NONE`; • `MD5`; • `SHA`; • `SHA224`; • `SHA256`; • `SHA384`; • `SHA512`
`set snmpv3 user <имя> auth-key <значение>`	Пароль аутентификации. Так же пароль может быть получен из Хранилища секретов
`set snmpv3 user <имя> priv-protocol <значение>`	Протокол конфиденциальности (по умолчанию – `NONE`). Для параметра `priv-protocol` могут устанавливаться следующие значения: • `NONE`; • `DES`; • `AES`; • `AES192`; • `AES256`
`set snmpv3 user <имя> priv-key <значение>`	Пароль конфиденциальности. Так же пароль может быть получен из Хранилища секретов
`set snmpv3 user <имя> permissions <значение>`	Уровень доступа пользователя. Для параметра `permissions` могут устанавливаться следующие значения: • `RO` – доступ на чтение; • `RW` – доступ на чтение и запись

Объект `ssl-policy`

Доступные команды объекта `ssl-policy` приведены в таблице (см. Доступные команды объекта `ssl-policy`).

Таблица 67. Доступные команды объекта `ssl-policy`

Команда	Описание
`set ssl-policy policy <имя>`	Создание SSL-Политики
`set ssl-policy policy <имя> rules <номер>`	Порядковый номер для применения правила SSL-Политики. Чем ниже номер, тем выше приоритет, и тем раньше правило будет обработано SSL-Политикой

Команда	Описание

`set ssl-policy policy <имя> rules <номер> <тип> <условие> <значение>`	Условие для правила SSL-Политики.
	
	Возможные типы:
	
	• `ALPN` – проверка списка ALPN из TLS Hello. Условием может быть:
	  ◦ `eq` – точное соответствие `ALPN` и <значение>;
	  ◦ `not-eq` – точное несоответствие `ALPN` и <значение>;
	• `CIPHER` – проверка списка алгоритмов шифрования из TLS Hello. Проверяется последовательность в любом месте на соответствие списку. Условием может быть:
	  ◦ `contains` – содержит последовательность в <значение>;
	  ◦ `not-contains` – не содержит последовательность в <значение>;
	• `SNI` – проверка поля SNI из TLS Hello. Условием может быть:
	  ◦ `contains` – содержит последовательность в <значение>;
	  ◦ `not-contains` – не содержит последовательность в <значение>;
	  ◦ `eq` – точное соответствие `SNI` и <значение>;
	  ◦ `not-eq` – точное несоответствие `SNI` и <значение>;
	  ◦ `ends-with` – заканчивается на <значение>;
	  ◦ `starts-with` – начинается с <значение>;
	• `DEFAULT` – действие при невыполнении условия (не содержит условие)

`set ssl-policy policy <имя> rules <номер> action <параметр> <значение>`	Действие при выполнении условия. Параметром может быть: • `lbs-id` – выбор Сервера Балансировки; • `ssl-profile-id` – выбор SSL-Профиля; • `drop` – разрыв соединения. Параметр не имеет значения

Объект `ssl-profile`

Доступные команды объекта `ssl-profile` приведены в таблице (см. Доступные команды объекта `ssl-profile`).

Таблица 68. Доступные команды объекта `ssl-profile`

Команда	Описание
`set ssl-profile server <имя>`	Создание Серверного SSL-Профиля для функционала SSL Offload. Серверный SSL-Профиль определяет взаимодействие между пользователем и Termidesk Connect, где Termidesk Connect является сервером
Настройка Серверного SSL-Профиля для функционала SSL Offload	
`set ssl-profile server <имя> handshake-timeout <значение>`	Задание времени ожидания (в секундах) установления соединения (значение от 1 до 60, по умолчанию – 5)

Команда	Описание
`set ssl-profile server <имя> host <имя>`	Задание значения поля SNI из TLS Hello, для которого требуются особые настройки обработки. Поддерживается задание шаблонов SNI по формату: `*.<домен>`. При этом: • астериск может быть расположен только слева и должен быть разделен от домена точкой; • астериск означает, что любой хост домена (не включая оригинальный домен) удовлетворяет шаблону. Для профиля оригинального домена требуется отдельный хост SNI; • при выборе SNI приоритетным будет тот, у которого совпадет больше уровней доменов или имеется полное совпадение по шаблону
`set ssl-profile server <имя> host <имя> setting <значение>`	Задание конфигурации SNI для особых настроек обработки. Список параметров команды идентичен параметрам `setting-default`
`set ssl-profile server <имя> setting-default <параметр> <значение>`	Задание конфигурации по умолчанию. Используется, если пришедший SNI пуст, либо не соответствует SNI, указанному в параметре `host`. Для определения конфигурации, используемой для SNI из параметра `host` или для конфигурации по умолчанию `setting-default`, применяются команды ниже
`set ssl-profile server <имя> setting-default ca-certs <значение>`	Файлы сертификатов УЦ для проверки подлинности клиентского сертификата. Указывается полное имя файлов с расширением. Так же файлы могут быть получены из Хранилища секретов. Используется, если активирована взаимная аутентификация по протоколу mTLS
`set ssl-profile server <имя> setting-default cert <значение>`	Файл сертификата сервера для аутентификации. Указывается полное имя файла с расширением. Так же файл может быть получен из Хранилища секретов
`set ssl-profile server <имя> setting-default key <значение>`	Файл закрытого ключа, соответствующий сертификату сервера. Указывается полное имя файла с расширением. Так же файл может быть получен из Хранилища секретов
`set ssl-profile server <имя> setting-default password <значение>`	Пароль к файлу закрытого ключа, если он необходим. Так же пароль может быть получен из Хранилища секретов
`set ssl-profile server <имя> setting-default ciphers <значение>`	Список используемых алгоритмов преобразований. Полный перечень приведен после таблицы

Команда	Описание
`set ssl-profile server <имя> setting-default crl-param file <значение>`	Задание CRL-файла для проверки клиентских сертификатов. Указывается полное имя файла с расширением. Так же файл может быть получен из Хранилища секретов
`set ssl-profile server <имя> setting-default dh-params <значение>`	Файл с параметрами Диффи-Хеллмана. Указывается полное имя файла с расширением
`set ssl-profile server <имя> setting-default mtls <значение>`	Управление режимом взаимной аутентификации по протоколу mTLS (по умолчанию – `false`). Может быть: • `true` (также `enable`) – взаимная аутентификация включена; • `false` (также `disable`) – взаимная аутентификация отключена
`set ssl-profile server <имя> setting-default mtls-check <значение>`	Метод проверки отзыва клиентских сертификатов (по умолчанию – `NONE`). Может быть: • `NONE` – отсутствует проверка на отзыв клиентского сертификата; • `CRL` – проверка методом CRL, отозванные сертификаты перечислены в CRL-файле. Если клиент предоставляет сертификат из этого списка, то он считается отозванным и подключение сбрасывается; • `OCSP` – проверка методом OCSP, когда на сервер (OCSP Responder) отправляется OCSP-запрос для получения актуального статуса сертификата

Команда	Описание

`set ssl-profile server <имя> setting-default ocsp-param <значение>`	Задание параметров проверки клиентских сертификатов методом OCSP.
	
	Для параметра `ocsp-param` указываются дополнительные свойства:
	
	• `cache` – кеширование OCSP-ответов (по умолчанию – `true`);
	• `method` – метод HTTP-запроса (по умолчанию – `POST`);
	• `nonce` – поддержка расширения Nonce (по умолчанию – `true`);
	• `strict` – строгий режим проверки (по умолчанию – `true`). Значение может быть:
	  ◦ `true` – допуск только клиентов, чьи сертификаты разрешены;
	  ◦ `false` – допуск клиентов, чьи сертификаты разрешены или чьих сертификатов нет в БД;
	• `timeout` – время ожидания (в секундах) ответа от сервера (OCSP Responder) (по умолчанию – `10`);
	• `url` – URL-адрес сервера (OCSP Responder)
	
	В текущей версии Termidesk Connect для сервера (OCSP Responder) поддерживается только протокол HTTP. Пример значения параметра: `http://myocspserver.local/`.

`set ssl-profile server <имя> setting-default stapling <значение>`	Поддержка OCSP Stapling с фоновым обновлением подписи серверного сертификата через OCSP Responder (по умолчанию – `false`)
`set ssl-profile server <имя> setting-default versions <значение>`	Задание версий протокола TLS, которые должны поддерживаться при установлении соединения. Для параметра `versions` могут устанавливаться следующие значения: • `ssl-v3` – не рекомендуется использовать из-за уязвимостей; • `tls-v1` – первая версия TLS, которая улучшает безопасность по сравнению с SSLv3, но также содержит некоторые уязвимости; • `tls-v11` – улучшенная версия TLS 1.0, которая исправляет некоторые недостатки, но все еще считается устаревшей; • `tls-v12` – широко используемая версия, обеспечивающая улучшенные механизмы шифрования и безопасности; • `tls-v13` – последняя версия протокола TLS, предлагающая значительные улучшения в производительности и безопасности

Команда	Описание

`set ssl-profile server <имя> setting-default ssl-reneg <параметр> <значение>`	Задание параметров механизма Secure Renegotiation (применяется для протокола TLS версии 1.2 или старше).
	
	Для параметра `ssl-reneg` указываются:
	
	• `state` – управление состоянием использования Secure Renegotiation (по умолчанию – `ENABLE`):
	  ◦ `ENABLE` – механизм Secure Renegotiation включен;
	  ◦ `DISABLE` – механизм Secure Renegotiation отключен;
	• `rate` – количество допустимых запросов в минуту на повторное согласование в рамках одного SSL-подключения (по умолчанию – `10`). При превышении заданного лимита для всех последующих попыток будет возвращено предупреждение о невозможности согласования на уровне протокола, пока период не обновится. При значении `0` ограничение на количество допустимых запросов отсутствует

`set ssl-profile server <имя> setting-default session-reuse <значение>`	Задание параметров механизма Session Reuse (применяется для протокола TLS версии 1.2 или старше).
	
	Для параметра `session-reuse` указываются:
	
	• `type` – тип Session Reuse (по умолчанию – `STATEFUL`):
	  ◦ `NONE` – повторное использование SSL-сессии не выполняется;
	  ◦ `STATEFUL` – выполняется повторное использование SSL-сессии;
	• `stateful-param session-timeout` – время хранения (в секундах) SSL-сессии в кеше (по умолчанию – `7200`)

`set ssl-profile server <имя> description <значение>`	Задание комментария, который будет привязан к Серверному SSL-Профилю Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.
`set ssl-profile client <имя>`	Создание Клиентского SSL-Профиля для функционала SSL Offload. Клиентский SSL-Профиль определяет взаимодействие между Termidesk Connect и Реальным Сервером, где Termidesk Connect является клиентом
Настройка Клиентского SSL-Профиля для функционала SSL Offload	
`set ssl-profile client <имя> ca-cert <значение>`	Задание файла сертификата УЦ для проверки подлинности Реального Сервера. Указывается полное имя файла с расширением. Так же файл может быть получен из Хранилища секретов

Команда	Описание
`set ssl-profile client <имя> cert <значение>`	Задание файла клиентского сертификата для аутентификации Termidesk Connect на Реальном Сервере. Указывается полное имя файла с расширением. Так же файл может быть получен из Хранилища секретов
`set ssl-profile client <имя> ciphers <значение>`	Задание списка используемых алгоритмов преобразования. Полный перечень приведен после таблицы
`set ssl-profile client <имя> dh-params <значение>`	Задание файла с параметрами Диффи-Хеллмана. Указывается полное имя файла с расширением
`set ssl-profile client <имя> handshake-timeout <значение>`	Задание времени ожидания (в секундах) установления соединения (значение от 1 до 60, по умолчанию – 5)
`set ssl-profile client <имя> key <значение>`	Задание файла закрытого ключа для аутентификации Termidesk Connect и Реального Сервера. Указывается полное имя файла с расширением. Так же файл может быть получен из Хранилища секретов
`set ssl-profile client <имя> password <значение>`	Задание пароля к файлу закрытого ключа, если он необходим. Так же пароль может быть получен из Хранилища секретов
`set ssl-profile client <имя> sni-default <значение>`	Задание значения по умолчанию для SNI
`set ssl-profile client <имя> versions <значение>`	Задание версий протокола TLS, которые должны поддерживаться при установлении соединения. Для параметра `versions` могут устанавливаться следующие значения: • `ssl-v3` – не рекомендуется использовать из-за уязвимостей; • `tls-v1` – первая версия TLS, которая улучшает безопасность по сравнению с SSLv3, но также содержит некоторые уязвимости; • `tls-v11` – улучшенная версия TLS 1.0, которая исправляет некоторые недостатки, но все еще считается устаревшей; • `tls-v12` – широко используемая версия, обеспечивающая улучшенные механизмы шифрования и безопасности; • `tls-v13` – последняя версия протокола TLS, предлагающая значительные улучшения в производительности и безопасности

Команда	Описание
`set ssl-profile client <имя> setting-default ssl-reneg <значение>`	Управление состоянием использования Secure Renegotiation (применяется для протокола TLS версии 1.2 или старше). Возможные значения (по умолчанию – `ENABLE`): • `ENABLE` – механизм Secure Renegotiation включен; • `DISABLE` – механизм Secure Renegotiation отключен
`set ssl-profile client <имя> description <значение>`	Задание комментария, который будет привязан к Клиентскому SSL-Профилю  Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.

Список поддерживаемых алгоритмов преобразования для параметра `ciphers`:

- AES128-GCM-SHA256;
- AES256-GCM-SHA384;
- AES128-SHA AES256-SHA;
- AES128-SHA256;
- AES256-SHA256;
- DHE-PSK-AES128-CBC-SHA;
- DHE-PSK-AES128-CBC-SHA256;
- DHE-PSK-AES128-GCM-SHA256;
- DHE-PSK-AES256-CBC-SHA;
- DHE-PSK-AES256-CBC-SHA384;
- DHE-PSK-AES256-GCM-SHA384;
- DHE-PSK-CHACHA20-POLY1305;
- DHE-RSA-AES128-GCM-SHA256;
- DHE-RSA-AES128-SHA;
- DHE-RSA-AES128-SHA256;
- DHE-RSA-AES256-GCM-SHA384;
- DHE-RSA-AES256-SHA;
- DHE-RSA-AES256-SHA256;
- DHE-RSA-CHACHA20-POLY1305;
- ECDHE-ECDSA-AES128-GCM-SHA256;
- ECDHE-ECDSA-AES128-SHA;
- ECDHE-ECDSA-AES128-SHA256;
- ECDHE-ECDSA-AES256-GCM-SHA384;
- ECDHE-ECDSA-AES256-SHA;
- ECDHE-ECDSA-AES256-SHA384;
- ECDHE-ECDSA-CHACHA20-POLY1305;

- ECDHE-PSK-AES128-CBC-SHA;
- ECDHE-PSK-AES128-CBC-SHA256;
- ECDHE-PSK-AES256-CBC-SHA;
- ECDHE-PSK-AES256-CBC-SHA384;
- ECDHE-PSK-CHACHA20-POLY1305;
- ECDHE-RSA-AES128-GCM-SHA256;
- ECDHE-RSA-AES128-SHA;
- ECDHE-RSA-AES128-SHA256;
- ECDHE-RSA-AES256-GCM-SHA384;
- ECDHE-RSA-AES256-SHA;
- ECDHE-RSA-AES256-SHA384;
- ECDHE-RSA-CHACHA20-POLY1305;
- GOST2001-GOST89-GOST89;
- GOST2012-GOST8912-GOST8912;
- PSK-AES128-CBC-SHA;
- PSK-AES128-CBC-SHA256;
- PSK-AES128-GCM-SHA256;
- PSK-AES256-CBC-SHA;
- PSK-AES256-CBC-SHA384;
- PSK-AES256-GCM-SHA384;
- PSK-CHACHA20-POLY1305;
- RSA-PSK-AES128-CBC-SHA;
- RSA-PSK-AES128-CBC-SHA256;
- RSA-PSK-AES128-GCM-SHA256;
- RSA-PSK-AES256-CBC-SHA;
- RSA-PSK-CHACHA20-POLY1305;
- SRP-RSA-AES-128-CBC-SHA;
- TLS_AES_256_GCM_SHA384;
- SRP-RSA-AES-256-CBC-SHA;
- TLS_CHACHA20_POLY1305_SHA256;
- SRP-AES-256-CBC-SHA;
- RSA-PSK-AES256-CBC-SHA384;
- SRP-AES-128-CBC-SHA;
- RSA-PSK-AES256-GCM-SHA384;
- TLS_AES_128_GCM_SHA256.

Объект `system`

Доступные команды объекта `system` приведены в таблице (см. Доступные команды объекта `system`).

Таблица 69. Доступные команды объекта `system`

Команда	Описание
`set system hostname <имя>`	Задание имени хоста
`set system license <файл>`	Загрузка лицензии Termidesk Connect из файла формата `.zip`
`set system timezone <часовой_пояс>`	Задание часового пояса из списка доступных
`set system mgmt ip <IP-адрес>`	Задание IP-адреса из списка заданных адресов, на котором будет доступен веб-интерфейс управления
`set system mgmt ip webui-port <порт>`	Задание порта, на котором будет доступен веб-интерфейс управления (по умолчанию – `443`)
`set system mgmt webui-cert <файл>`	Задание сертификата для подключения к веб-интерфейсу по протоколу HTTPS (по умолчанию – `tdc_ss_public_key.pem`). Так же файл может быть получен из Хранилища секретов
`set system mgmt webui-key <файл>`	Задание закрытого ключа к сертификату для подключения к веб-интерфейсу по протоколу HTTPS (по умолчанию – `tdc_ss_private_key.key`). Так же файл может быть получен из Хранилища секретов
`set system mgmt webui-timeout <значение>`	Задание времени жизни (в секундах) сессии пользователя (по умолчанию – `3600`)

Объект `tcp-profile`

Доступные команды объекта `tcp-profile` приведены в таблице (см. Доступные команды объекта `tcp-profile`).

Таблица 70. Доступные команды объекта `tcp-profile`

Команда	Описание
`set tcp-profile server <имя>`	Создание Серверного TCP-Профиля. Серверный TCP-Профиль определяет взаимодействие между пользователем и Termidesk Connect, где Termidesk Connect является сервером
Настройка Серверного TCP-Профиля	
`set tcp-profile server <имя> buffer-size <значение>`	Размер буфера (в байтах) для чтения (по умолчанию – `32768`)
`set tcp-profile server <имя> cc <значение>`	Алгоритм предотвращения перегрузок (по умолчанию – `CUBIC`)
`set tcp-profile server <имя> keep-alive <значение>`	Настройка проверки активности соединения. Для параметра `keep-alive` могут быть установлены дополнительные параметры: • `enable` – активация или отключение проверки активности соединения; • `interval` – интервал (в секундах) отправки пакетов проверки (по умолчанию – `75`); • `probe` – количество пакетов проверки, которые следует отправить при отсутствии подтверждения от узла (по умолчанию – `3`); • `timeout` – время бездействия (в секундах) соединения перед отправкой пакетов проверки (по умолчанию – `900`)

Команда	Описание
`set tcp-profile server <имя> proxy-protocol <значение>`	Настройка PROXY-протокола для входящего соединения. Для параметра `proxy-protocol` могут быть установлены дополнительные параметры: • `allow-con-without-proxy-protocol` – разрешение подключения пользователя без передачи заголовка PROXY-протокола (по умолчанию – `false`); • `timeout` – время ожидания (в секундах) получения заголовка PROXY-протокола от пользователя (по умолчанию – `10`); • `v1` – включение поддержки PROXY-протокола версии 1; • `v2` – включение поддержки PROXY-протокола версии 2. Для параметра `v1` может быть установлен дополнительный параметр: • `allow-unknown` – разрешение приема заголовка `PROXY UNKNOWN` (по умолчанию – `false`). Для параметра `v2` могут быть установлены дополнительные параметры: • `allow-af-unix` – разрешение приема заголовка, содержащего адрес `AF_UNIX` (по умолчанию – `false`); • `allow-af-unspec` – разрешение приема заголовка, содержащего адрес `UNSPEC` (по умолчанию – `false`); • `allow-local` – разрешение приема заголовка, содержащего команду `LOCAL` (по умолчанию – `false`)
`set tcp-profile server <имя> tcp-nodelay <значение>`	Активация или отключение использования алгоритма Нейгла (по умолчанию – `true`)
`set tcp-profile server <имя> write-timeout <значение>`	Время ожидания (в секундах) записи в сокет (по умолчанию – `60`)
`set tcp-profile server <имя> description <значение>`	Комментарий, который будет привязан к Серверному TCP-Профилю Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.
`set tcp-profile client <имя>`	Создание Клиентского TCP-Профиля. Клиентский TCP-Профиль определяет взаимодействие между Termidesk Connect и Реальным Сервером, где Termidesk Connect является клиентом
Настройка Клиентского TCP-Профиля	
`set TCP-profile client <имя> buffer-size <значение>`	Размер буфера (в байтах) для чтения (по умолчанию – `32768`)
`set TCP-profile client <имя> cc <значение>`	Алгоритм предотвращения перегрузок (по умолчанию – `CUBIC`)
`set TCP-profile client <имя> connect-timeout <значение>`	Время ожидания (в секундах) соединения с Реальным Сервером (по умолчанию – `10`)

Команда	Описание
`set TCP-profile client <имя> idle-timeout <значение>`	Время (в секундах) отсутствия данных в сессии (по умолчанию – `120`)
`set TCP-profile client <имя> keep-alive <значение>`	Настройка проверки активности соединения. Для параметра `keep-alive` могут быть установлены дополнительные параметры: • `enable` – активация или отключение проверки активности соединения; • `interval` – интервал (в секундах) отправки пакетов проверки (по умолчанию – `75`); • `probe` – количество пакетов проверки, которые следует отправить при отсутствии подтверждения от узла (по умолчанию – `3`); • `timeout` – время бездействия (в секундах) соединения перед отправкой пакетов проверки (по умолчанию – `900`)

`set TCP-profile client <имя> proxy-protocol <значение>`	Настройка PROXY-протокола для соединения с Реальным Сервером.
	
	Для параметра `proxy-protocol` могут быть установлены дополнительные параметры:
	
	• `pass-through-addr` – формирование адреса для заголовка PROXY-протокола (по умолчанию – `true`). Значение может быть:
	  ◦ `true` – Termidesk Connect передаст на Реальный Сервер адрес источника и адрес назначения, полученные из заголовка PROXY-протокола пользовательского подключения (настроенного на Серверном TCP-Профиле);
	  ◦ `false` – Termidesk Connect сформирует адреса по соответствию:
	  ◦ адрес источника — адрес пользователя;
	  ◦ адрес назначения — адрес Реального Сервера;
	• `v1` – включение поддержки PROXY-протокола версии 1;
	• `v2` – включение поддержки PROXY-протокола версии 2.
	
	Для параметра `v2` могут быть установлены дополнительные параметры:
	
	• `pass-through-tlvs` – тип TLV, который будет передан из пользовательского подключения в сторону Реального Сервера;
	• `added-tlv` – тип TLV для добавления в заголовок PROXY-протокола

Команда	Описание

`set TCP-profile client <имя> proxy-protocol v2 added-tlv <значение>`	Тип TLV для добавления в заголовок PROXY-протокола.
	
	Для параметра `added-tlv` могут быть установлены дополнительные параметры:
	
	• `value` – значение TLV;
	• `format` – формат TLV (по умолчанию – `STR`). Значение может быть:
	  ◦ `STR` – строка;
	  ◦ `BASE64` – бинарные данные закодированные в формате `Base64`

`set TCP-profile client <имя> tcp-nodelay <значение>`	Активация или отключение использования алгоритма Нейгла (по умолчанию – `true`)
`set TCP-profile client <имя> write-timeout <значение>`	Время ожидания (в секундах) записи в сокет (по умолчанию – `60`)
`set TCP-profile client <имя> description <значение>`	Комментарий, который будет привязан к Клиентскому TCP-Профилю Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.

Объект `user`

Доступные команды объекта `user` приведены в таблице (см. Доступные команды объекта `user`).

Таблица 71. Доступные команды объекта `user`

Команда	Описание
`set user <имя>`	Создание пользователя
`set user <имя> <параметр> <значение>`	Настройка пользователя. Параметром может быть: • `password` – пароль пользователя; • `groups` – группа пользователя. Группа должна быть предварительно создана (см. подраздел **Объект `groups`**)

Объект `vlan`

Доступные команды объекта `vlan` приведены в таблице (см. Доступные команды объекта `vlan`).

Таблица 72. Доступные команды объекта `vlan`

Команда	Описание
`set vlan name <имя>`	Создание VLAN
`set vlan name <имя> vlan-id <идентификатор>`	Назначение идентификатора VLAN
`set vlan name <имя> if-ethernet <имя интерфейса>`	Назначение VLAN сетевому интерфейсу (физическому)

Команда	Описание
`set vlan name <имя> if-aggregation <имя_агрегированного_интерфейса>`	Назначение VLAN агрегированному интерфейсу (логическому)
`set vlan name <имя> vrf <имя_VRF>`	Назначение VRF для VLAN
`set vlan name <имя> ha-type <тип>`	Назначение режима использования VLAN при отказоустойчивой конфигурации, где `<тип>` может быть: • `LOCAL` – VLAN используется в локальной конфигурации, не синхронизируется для отказоустойчивой конфигурации; • `SHARED` – VLAN синхронизируется для отказоустойчивой конфигурации

Объект `vrf`

Доступные команды объекта `vrf` приведены в таблице (см. Доступные команды объекта `vrf`).

Таблица 73. Доступные команды объекта `vrf`

Команда	Описание
`set vrf name <имя>`	Создание VRF
`set vrf name <имя> table-id <номер таблицы>`	Назначение идентификатора таблицы маршрутизации, которая будет ассоциирована с этим VRF Номер таблицы должен быть уникальным.

Объект `vs`

Доступные команды объекта `vs` приведены в таблице (см. Доступные команды объекта `vs`).

Таблица 74. Доступные команды объекта `vs`

Команда	Описание
`set vs HTTP <имя>`	Создание Виртуального Сервера для балансировки по протоколу HTTP
Настройка Виртуального Сервера для балансировки по протоколу HTTP	

Команда	Описание

`set vs HTTP <имя> check-lbs <параметр> <значение>`	Настройка проверки доступности Серверов Балансировки.
	
	Параметром может быть:
	
	• `algorithm` – алгоритм определения статуса Виртуального Сервера. Возможные значения:
	  ◦ `OR` – статус Виртуального Сервера будет «В работе», если работает хотя бы один ассоциированный с ним Сервер Балансировки;
	  ◦ `AND` – статус Виртуального Сервера будет «В работе», если работают все ассоциированные с ним Серверы Балансировки;
	  ◦ `NONE` – не использовать никакой из алгоритмов;
	• `lbs-ids` – Серверы Балансировки, влияющие на статус Виртуального Сервера. Статус Виртуального Сервера зависит от статуса ассоциированного с ним Сервера Балансировки
	
	Может быть задано несколько Серверов Балансировки. Команда позволяет за один раз добавить только один Сервер Балансировки.

`set vs HTTP <имя> http-profile-id <значение>`	Серверный HTTP-Профиль, который будет использоваться для HTTP-запросов
`set vs HTTP <имя> luarules <номер> script <имя_файла>`	Настройка применения Сценария (исполняемого файла формата LUA), который определяет правила обработки трафика, проходящего через Виртуальный Сервер. Параметр `<номер>` задает порядковый номер применения Сценария: чем ниже номер, тем выше приоритет, и тем раньше Сценарий будет обработан Виртуальным Сервером. Описание работы со Сценариями приведено в подразделе **Сценарии**
`set vs HTTP <имя> rhi <значение>`	Настройка возможности анонсирования IP-адреса, привязанного к Виртуальному Серверу, протоколам динамической маршрутизации (по умолчанию – `OFF`). Возможные значения: • `ON` – активация RHI. При активации RHI Termidesk Connect будет анонсировать в сеть IP-адреса Виртуальных Серверов в зависимости от их режима работы (`rhi-state`); • `OFF`– отключение RHI

Команда	Описание
`set vs HTTP <имя> rhi-state <значение>`	Режим работы RHI для Виртуального Сервера (настраивается, если активирован `rhi`) (по умолчанию – `ACTIVE`). Возможные значения: • `ACTIVE` – активный режим RHI для Виртуального Сервера; • `PASSIVE` – пассивный режим RHI для Виртуального Сервера Идентификатором маршрута для функционала RHI являются пары «VRF – IP-адрес» (поддерживаются множественные VRF), при этом состояние маршрута определяется условиями: • если все Виртуальные Сервера по данному маршруту находятся в режиме `PASSIVE`, то Termidesk Connect всегда будет объявлять маршрут для виртуального IP-адреса; • если хотя бы один Виртуальный Сервер находятся в режиме `ACTIVE` и в состоянии «В работе», то Termidesk Connect будет объявлять маршрут для виртуального IP-адреса; • в остальных случаях Termidesk Connect не будет объявлять маршрут. Состояние RHI можно проверить командой: `show status rhi`
`set vs HTTP <имя> rl-profile-id <значение>`	Профиль ограничения скорости для привязки к Виртуальному Серверу
`set vs HTTP <имя> ssl-profile-id <значение>`	Серверный SSL-Профиль, который будет использоваться для соединений
`set vs HTTP <имя> sslrules <номер> script <имя_файла>`	Настройка применения Сценария SSL (исполняемого файла формата LUA), который определяет выбор SSL-Профиля или Сервера Балансировки на основании данных, полученных из сообщения TLS Hello. Параметр `<номер>` задает порядковый номер применения Сценария: чем ниже номер, тем выше приоритет, и тем раньше Сценарий будет обработан Виртуальным Сервером. Описание работы со Сценариями приведено в подразделе **Сценарии**
`set vs HTTP <имя> tcp-profile-id <значение>`	Серверный TCP-Профиль, который будет использоваться для TCP-соединений
`set vs HTTP <имя> vip <параметр> <значение>`	IP-адрес и порт, который будет присвоен Виртуальному Серверу. Для задания любого порта используется значение `0`. Параметром может быть: • `ip` – IP-адрес Виртуального Сервера; • `port` – конкретный порт Виртуального Сервера; • `portrange` – диапазон портов Виртуального Сервера. Начальное значение задается в `portrange from`, конечное – в `portrange to`; • `portlist` – список портов Виртуального Сервера. Каждый порт задается отдельной командой для добавления в список
`set vs HTTP <имя> vrf <значение>`	Имя VRF для привязки к Виртуальному Серверу

Команда	Описание
`set vs HTTP <имя> description <значение>`	Комментарий, который будет привязан к Виртуальному Серверу. Для написания комментария допустимы только латинские буквы. В случае использования пробелов следует заключать текст комментария в двойные кавычки.
`set vs TCP <имя>`	Создание Виртуального Сервера для балансировки по протоколу TCP. Настройка идентична настройке по протоколу HTTP, за исключением: • `http-profile-id` (неприменимо для этого типа Виртуального Сервера); • `luarules` (неприменимо для Виртуального Сервера уровня L4); • `sslrules`. Дополнительные настройки представлены ниже
Настройка Виртуального Сервера для балансировки по протоколу TCP.	
`set vs TCP <имя> ssl-policy-id <SSL-Политика>`	SSL-Политика для привязки к Виртуальному Серверу. Описание работы с SSL-Политиками приведено в подразделе **SSL-Политики**
`set vs TCP <имя> netrules <номер> <параметр> <значение>`	Настройка применения правил на основе сети источника. Чем ниже порядковый номер, тем выше приоритет, и тем раньше правило будет обработано Виртуальным Сервером. Параметром может быть: • `network` – сеть источника запроса. В зависимости от того, из какой сети подключился пользователь, будет выбран тот или иной Сервер Балансировки; • `lbs-id` – Сервер Балансировки для правила. Пример команды для параметра `network`: `set vs TCP VSName netrules 10 network 0.0.0.0/0`
`set vs RAPID-TCP <имя>`	Создание Виртуального Сервера для балансировки по протоколу RAPID TCP. Настройка идентична TCP, за исключением (не задается для RAPID TCP): • `rl-profile-id`; • `ssl-policy-id`; • `ssl-profile-id`; • `tcp-profile-id`
`set vs RAPID-UDP <имя>`	Создание Виртуального Сервера для балансировки по протоколу RAPID UDP. Настройка идентична RAPID-TCP

Команда show

Отображает различные параметры Termidesk Connect: конфигурацию, состояние, версию и др.

Формат:

```
show <опция> <параметры>
```

или:

```
show <опция> <параметры> | <утилита>
```

Доступные утилиты приведены в таблице (см. Доступные утилиты команды show).

Таблица 75. Доступные утилиты команды show

Утилита	Описание
grep	Вызов стандартной утилиты grep для поиска по заданному выражению. Формат: `show <опция> <параметры> | grep <выражение>`
less	Вызов стандартной утилиты less для постраничного просмотра и поиска. Формат: `show <опция> <параметры> | less`

Доступные опции приведены в таблице (см. Доступные опции команды show).

Таблица 76. Доступные опции команды show

Опция	Описание
bandwidth	Вывод пропускной способности сети
cert	Вывод информации о сертификате или файле сертификатов УЦ. При указании другого файла выводится ошибка Файл не является сертификатом, при указании несуществующего — Сертификат не найден. Формат: `show cert <файл_сертификата>`

Опция	Описание
`configuration`	Вывод рабочей конфигурации Termidesk Connect. Может использоваться вывод конкретного элемента или параметра рабочей конфигурации в определенном формате: `show configuration <формат> <объект>` Формат может быть: • `cli` – вывод будет представлен командами CLI; • `json` – вывод будет представлен в формате `JSON`; • `netconf` – вывод будет представлен в формате операций протокола управления конфигурациями сетевых устройств NETCONF; • `text` – вывод будет представлен простым текстом; • `xml` – вывод будет представлен в формате `XML`
`connectiontable`	Вывод таблицы открытых соединений. Формат: `show connectiontable <фильтр>` Фильтр может быть: • `all` – записи не будут отфильтрованы, в выводе будут все записи; • `client-ip <IP-адрес>` – записи будут отфильтрованы по IP-адресу клиента, который открыл соединение; • `lbs <тип> <имя>` – записи будут отфильтрованы по имени Сервера Балансировки заданного типа (HTTP, TCP, RAPID-TCP или RAPID-UDP); • `vs <тип> <имя>` – записи будут отфильтрованы по имени Виртуального Сервера заданного типа (HTTP, TCP, RAPID-TCP или RAPID-UDP)
`cpu`	Вывод информации о загрузке процессора
`ddos`	Вывод информации о состоянии защиты от DDoS-атак. Может использоваться вывод всех DDoS-Профилей (значение имени `all`) или одного с указанием его имени: `show ddos <параметр> <имя>` Параметр может быть: • `detection` – статистика защиты от DDoS-атак; • `profile` – DDoS-Профиль
`difference`	Вывод различий между предварительной и рабочей конфигурациями
`disk`	Вывод информации по использованию дискового пространства

Опция	Описание
`entities`	Вывод текущего количества существующих серверов. Формат команды: `show entities <объект>` Объект может быть: • `lbs` – Сервер Балансировки; • `rs-pool` – Группа Реальных Серверов; • `vs` – Виртуальный Сервер
`interfaces`	Вывод информации о сетевых интерфейсах. Формат команды: `show interfaces`
`ip address`	Вывод списка IP-адресов. Формат команды: `show ip address`
`ip route`	Вывод списка маршрутов. Формат команды: `show ip route`
`ip route vrf`	Вывод списка маршрутов для выбранного VRF. Формат команды: `show ip route vrf <имя_VRF>`
`lbs`	Вывод статистики Серверов Балансировки. Может использоваться вывод всех типов (значение `all`) или одного выбранного типа с указанием его имени: `show lbs <тип> <имя>`
`license attributes`	Вывод информации о параметрах используемой лицензии
`logging level`	Вывод текущего уровня журналирования
`memory`	Вывод информации по использованию оперативной памяти

Опция	Описание
`rs-pool`	Вывод статистики Групп Реальных Серверов. Может использоваться вывод всех групп (значение имени `all`) или одной с указанием ее имени: `show rs-pool <имя>`
`secret`	Вывод информации о состоянии синхронизации секретов
`status`	Вывод информации о состоянии объектов. Формат команды: `show status <объект>`
`system-time`	Вывод информации о системном времени Termidesk Connect
`tcp`	Вывод информации о состоянии TCP-сессий
`version`	Вывод информации о версии Termidesk Connect
`vs`	Вывод статистики Виртуальных Серверов. Может использоваться вывод всех типов (значение `all`) или одного выбранного типа с указанием его имени: `show vs <тип> <имя>`
`xpath`	Поиск по выражению в предварительной конфигурации. Формат команды: `show xpath <запрос_элемента_конфигурации>`

Примеры вывода команд приведены в таблице (см. Примеры вывода команд).

Таблица 77. Примеры вывода команд

Команда	Вывод

`show bandwidth`	`Интерфейс  Прием, Мбит/с  Отправка, Мбит/с  Потерь исх., п/с  Потерь вх., п/с  Ошибок вх., п/с  Ошибок исх. п/с`
	
	`eth00  0.003  0.002  0 0  0  0`
	
	`Всего  0.003  0.002  0 0  0  0`

`show configuration cli ip | grep ha-type LOCAL`	`set ip address 192.0.2.120 /24 ha-type LOCAL` `set ip route vrf_custom 198.51.100.1/24 203.0.113.10 ha-type LOCAL`

Команда	Вывод
show connectiontable all	 Может возникнуть ситуация, при которой подключение к Виртуальному Серверу, Серверу Балансировки или Реальному Серверу еще не установлено. В этом случае в соответствующих полях вывода (ВС, СБ, Источник, РС) будут отображены прочерки (–). Клиент (IP:порт) ВС (имя) ВС (IP:порт) СБ (имя) Источник (IP:порт) РС (IP:порт) 10.198.16.138:50772 vs_http 10.101.34.112:8888 lbs1 127.0.0.1:9092 127.0.0.1:9092 10.198.16.138:53504 vs_http 10.101.34.112:8888 lbs2 127.0.0.1:9095 127.0.0.1:9095 Показано сессий: 2 (лимит: 5000)
show connectiontable client-ip 10.198.16.138	Клиент (IP:порт) ВС (имя) ВС (IP:порт) СБ (имя) Источник (IP:порт) РС (IP:порт) 10.198.16.138:50772 vs_http 10.101.34.112:8888 lbs1 127.0.0.1:9092 127.0.0.1:9092 10.198.16.138:53504 vs_http 10.101.34.112:8888 lbs2 127.0.0.1:9095 127.0.0.1:9095 Показано сессий: 2 (лимит: 5000)
show connectiontable lbs HTTP lbs2	Клиент (IP:порт) ВС (имя) ВС (IP:порт) СБ (имя) Источник (IP:порт) РС (IP:порт) 10.198.16.138:53504 vs_http 10.101.34.112:8888 lbs2 127.0.0.1:9095 127.0.0.1:9095 Показано сессий: 1 (лимит: 5000)
show connectiontable vs HTTP vs_http	Клиент (IP:порт) ВС (имя) ВС (IP:порт) СБ (имя) Источник (IP:порт) РС (IP:порт) 10.198.16.138:50772 vs_http 10.101.34.112:8888 lbs1 127.0.0.1:9092 127.0.0.1:9092 10.198.16.138:53504 vs_http 10.101.34.112:8888 lbs2 127.0.0.1:9095 127.0.0.1:9095 Показано сессий: 2 (лимит: 5000)
show cpu	Общая загрузка ЦП: 0.6% Загрузка по ядрам: 0.6% 0.6%

Команда	Вывод
show ddos detection all	ID профиля: ddos0 I Интерфейс SYN-flood ICMP-flood UDP-flood ens224 Обнаружен (IP) Не обнаружен Не обнаружен
show difference	<system xmlns="urn: termidesk:system"> - <hostname>vatb</hostname> + <hostname>vate</hostname> - <timezone>Europe/Moscow</timezone> + <timezone>UTC</timezone> </system>
show disk	Файловая система Размер Использовано Доступно Смонтировано в udev 1.4 ГБ 0.0 байт 1.4 ГБ /dev tmpfs 297.6 МБ 30.4 МБ 267.3 МБ /run /dev/tcida5d502a/root 4.3 ГБ 1.6 ГБ 2.5 ГБ / tmpfs 1.5 ГБ 8.0 КБ 1.5 ГБ /dev/shm
show interfaces	Номер Имя интерфейса Тип VRF MAC-адрес VLAN Родитель Тип HA 1 eth110 Ethernet default 00:50:56:a7:af:f8 - - -
show ip address	IP-адрес Префикс Имя интерфейса VRF 10.101.32.100 /24 eth110 default
show ip route	VRF Сеть назначения Адрес шлюза Тип HA default 0.0.0.0/0 10.101.32.1 SHARED vrf_custom 1.1.1.1/32 10.10.10.10 LOCAL

Команда	Вывод
show ip route vrf default	VRF Сеть назначения Адрес шлюза Тип НА default 0.0.0.0/0 10.101.32.1 SHARED
show lbs all	ID Тип rs-pool-id Статус TCP сес TCP с/сек TCP(TLS) сес TCP(TLS) с/сек UDP п/сек r_tcp RAPID-TCP TCP ONLINE 0 0 - - - r_udp RAPID-UDP UDP ONLINE - - - - 0 tcp_lb TCP HTTP ONLINE 0 0 0 0 -
show license attributes	HWID: 1AE02A267D277A5D48DB22F49DDFBF90 Customer: test-termidesk-connect-integration Redaction: termidesk_connect Since: 2025-11-12T00:00:00+0000 Until: 2026-11-11T23:59:59+0000 End version: 1.2 CPU: 1
show logging level	INFO
show memory	Всего: 2.9 ГБ Использовано: 229.5 МБ Свободно: 2.0 ГБ Доступно: 2.5 ГБ
show rs-pool all	rs-pool-id Адрес Статус TCP сес TCP с/сек TCP(TLS) сес TCP(TLS) с/сек UDP п/сек HTTP 10.130.255.197:80 ONLINE 0 0 0 0 0 HTTPS 10.130.255.197:443 ONLINE 0 0 0 0 0 TCP 10.130.255.197:22 ONLINE 0 0 0 0 0 UDP 10.130.255.197:5001 ONLINE 0 0 0 0 0 mTLS 10.130.255.197:8443 ONLINE 0 0 0 0 0

Команда	Вывод

`show tcp`	`ESTABLISHED:  21`
	`LISTEN:  19`
	`SYN_SENT:  0`
	`SYN_RECV:  0`
	`FIN_WAIT_1:  0`
	`FIN_WAIT_2:  0`
	`TIME_WAIT:  0`
	`CLOSE:  0`
	`CLOSE_WAIT:  0`
	`LAST_ACK:  0`
	`CLOSING:  0`
	
	`Всего:  40`

`show vs all`	`ID  Тип  VIP  Статус  TCP cec`
	`TCP с/сек  TCP(TLS) cec  TCP(TLS) с/сек UDP п/сек`
	
	`http_vs  HTTP  10.100.70.220:80  ONLINE  0`
	`0  0  0  -`
	`r_tcp_vs  RAPID-TCP 10.100.70.220:22  ONLINE  0`
	`0  -  -  -`
	`r_udp_vs  RAPID-UDP 10.100.70.220:5005  ONLINE  -`
	`-  -  -  0`

Команда `top`

Возвращает в корень дерева конфигурации. Обычно используется, если конфигурация редактируется через команду `mode`, для удобства навигации по элементам конфигурации.

Команда `up`

Возвращает на уровень выше в дереве конфигурации. Обычно используется, если конфигурация редактируется через команду `mode`, для удобства навигации по элементам конфигурации.

Команда `validate`

Проверяет корректность изменений в предварительной конфигурации.

Команда `write`

Сохраняет сделанные изменения конфигурации. Команда необходима для сохранения изменений после перезагрузки Termidesk Connect.

ИНТЕРФЕЙС VTYSH

Общие сведения по работе с VTYSH

VTYSH предназначен для настройки и управления службами маршрутизации, входящими в состав ПО FRR. Подробная информация по работе с интерфейсом VTYSH приведена в

документации ПО FRR: https://docs.frrouting.org/en/latest/vtysh.html.

Переход в VTYSH осуществляется из интерфейса CLI, для этого нужно последовательно выполнить:

```
bash
sudo vtysh
```

Для просмотра списка доступных команд или просмотра справки по ним можно воспользоваться клавишами:

- **<TAB>** – выводит возможные варианты продолжения команды:
  - если вариантов больше одного, то выводит список доступных опций;
  - если вариант один, то автоматически дописывает окончание команды;
- **<?>** – выводит список доступных команд с краткой справкой к ним.

ВЕБ-ИНТЕРФЕЙС

Доступ к веб-интерфейсу

Доступ к веб-интерфейсу управления Termidesk Connect осуществляется из веб-браузера по протоколу HTTPS с указанием URL-адреса: `https://<IP-адрес>:<порт>/`.

Для доступа к веб-интерфейсу по протоколу HTTPS на этапе установки автоматически генерируется самоподписанный сертификат и закрытый ключ к нему. IP-адрес и порт, на котором будет доступен веб-интерфейс, задается при первоначальной настройке Termidesk Connect (см. подраздел **Первоначальная настройка Termidesk Connect** документа СЛЕТ.10101-01 90 01 «Руководство администратора. Установка Termidesk Connect»).

Для подключения должны использоваться веб-браузеры с поддержкой спецификации W3C HTML5: Яндекс.Браузер версии 15.9 и выше, Google Chrome версии 46 и выше, Mozilla Firefox версии 41 и выше.

Минимально необходимое разрешение экрана монитора для работы с веб-интерфейсом – 1366x768 пикселей.

На странице подключения нужно заполнить экранные поля:

- «Username» – идентификатор субъекта с ролью «Администратор»;
- «Password» – набор символов, подтверждающий назначение полномочий.

По умолчанию после установки используется логин `tdadmin` с паролем `tdadmin`.

Обзор доступных функций веб-интерфейса

Веб-интерфейс Termidesk Connect позволяет администратору выполнить ряд задач по настройке и управлению, а именно:

- отслеживать статистику производительности, доступности узлов и сетевой активности Termidesk Connect;
- выполнять настройки системы, управлять трафиком, балансировкой запросов, задавать Сценарии работы узлов.

Сразу после авторизации в левой части веб-интерфейса администратору доступна панель,

содержащая список основных функций, перечисленных в таблице (см. таблицу Описание разделов веб-интерфейса).

В правой верхней части веб-интерфейса доступна возможность сохранить текущую конфигурацию Termidesk Connect. Для этого следует нажать экранную кнопку **[Сохранить]**.

Параметры конфигурации будут сохранены в БД Termidesk Connect. Для применения параметров конфигурации следует выполнить перезапуск Termidesk Connect.

Таблица 78. Описание разделов веб-интерфейса

Наименование раздела или функции	Описание
«Инфо»	Предоставляет сведения о нагрузке на узел, пропускной способности узла, сетевой активности и доступности узлов сетевой инфраструктуры
«Настройки»	Предоставляет доступ к параметрам конфигурирования Termidesk Connect
«Система»	Предоставляет доступ к настройкам системных параметров
«Лицензия»	Предоставляет возможность ознакомления с лицензионным соглашением
«DNS»	Позволяет настроить разрешение доменных имен
«NTP»	Позволяет настроить синхронизацию времени системы с заданным сервером точного времени
«Сеть»	Позволяет задать параметры для маршрутизации и взаимодействия узлов в IP-сетях для интеграции Termidesk Connect в существующую сеть организации
«Отказоустойчивость»	Позволяет задать параметры отказоустойчивой конфигурации
«Конфигурация кластера»	Позволяет настроить отказоустойчивую конфигурацию
«Отслеживание IP»	Позволяет задать отслеживаемые IP-адреса для готовности узла к переходу в статус «ACTIVE»
«Управление»	Позволяет задать параметры доступа к управлению устройством Termidesk Connect
«Аудит»	Позволяет задать параметры отправки событий Termidesk Connect на syslog-сервер
«SNMP»	Позволяет задать параметры для SNMP
«Настройки»	Позволяет управлять настройками SNMP версий 1 и 2c
«Пользователи»	Позволяет управлять настройками пользователей SNMPv3
«Обновление»	Позволяет выполнить обновление Termidesk Connect на новую версию
«Управление трафиком»	Предоставляет доступ к настройкам локальной балансировки подключений, управления доступом и проверки доступности Реальных Серверов
«Проверки»	Позволяет задать параметры периодических проверок доступности Реальных Серверов для балансировки подключений
«Группы Реальных Серверов»	Позволяет задать адреса и порты узлов с опубликованными приложениями, а также параметры Проверки их доступности
«Профили»	Позволяет настроить Профили для обработки трафика
«TCP»	Позволяет настроить параметры взаимодействия между пользователем и Termidesk Connect
«HTTP»	Позволяет настроить параметры взаимодействия между Termidesk Connect и Реальным Сервером

Наименование раздела или функции	Описание
«Сохранение Сессий»	Позволяет настроить параметры для привязки пользователя к Реальному Серверу
«Серверы Балансировки»	Определяет логику распределения подключений между группой Реальных Серверов
«TLS»	Позволяет настроить параметры для функционала SSL Offload
«Файлы»	Позволяет загружать файлы закрытого ключа и публичного сертификата для защищенного соединения по протоколу TLS
«Профили»	Позволяет задать настройки SSL-Профилей для управления параметрами обработки TLS-соединений
«Политики»	Позволяет задать настройки SSL-Политик
«Сценарии»	Позволяет задать параметры обработки запросов с использованием файлов Сценария, которые определяют логику взаимодействия с реальными серверами и маршрутизации трафика
«Виртуальные Серверы»	Определяет правила обработки пользовательских запросов и их последующую передачу на Сервер Балансировки в соответствии с заданными настройками
«ГеоБН»	Предоставляет доступ к настройкам геораспределенной балансировки подключений, позволяющим перенаправлять пользовательские запросы на наиболее подходящий ЦОД с учетом текущих параметров доступности и производительности узлов на основе DNS-запросов
«ADNS»	Позволяет задать настройку Termidesk Connect в качестве ADNS-сервера
«Площадки»	Позволяет задать имена логических групп серверов или ЦОД, которые используются в геораспределенной балансировке
«DNS View»	Позволяет задать настройку сетей, для запросов из которых будет выдаваться локальный IP-адрес, настроенный в Сервисах
«Сервисы»	Позволяет задать настройку общедоступного и локального IP-адреса, который будет помещаться в DNS-ответы
«Виртуальные Серверы»	Позволяет определить правила балансировки, по которым в DNS-ответах будут выдаваться IP-адреса Сервисов
«Зоны»	Позволяет задать настройку DNS-зоны, для которой будет применяться балансировка
«Безопасность»	Предоставляет доступ к настройкам безопасности
«DDoS»	Позволяет настроить DDoS-Профиль
«Списки Контроля Доступа»	Позволяет настроить правила фильтрации сетевого трафика
«AAA»	Позволяет задать настройки аутентификации пользователя со стороны клиента и способа передачи данных Реальному Серверу, в том числе с использованием внешних AAA-сервисов
«Серверы»	Позволяет задать настройку параметров взаимодействия с сервером аутентификации и авторизации
«Профили»	Позволяет задать настройку параметров аутентификации пользователей, таких как количество попыток входа, порядок выбора серверов аутентификации и другое
«KDC Серверы»	Позволяет задать настройку параметров взаимодействия с сервером аутентификации и авторизации KDC
«SSO Профили»	Позволяет задать настройку параметров перенаправления учетной записи пользователя на Реальный Сервер
«Ограничение скорости»	Позволяет задать настройки Профилей ограничения скорости
«Хранилище секретов»	Позволяет задать настройки подключения к Хранилищу секретов

Наименование раздела или функции	Описание
«Шлюз»	Предоставляет доступ к настройкам Шлюза
«Точки подключений»	Позволяет настроить точки подключений для Шлюза
«Координатор»	Позволяет задать общие настройки клиента RabbitMQ для Шлюза
«Сбор статистики»	Позволяет задать общие настройки сбора статистики для Шлюза
«Производительность»	Предоставляет сведения о производительности системы, Виртуальных Серверов и Серверов Балансировки
«Документация»	Предоставляет доступ к актуальной документации Termidesk Connect
«Настройка языка»	Определяет язык отображения элементов веб-интерфейса
«Переключение темы»	Определяет тему веб-интерфейса
«Выход»	Завершает текущую сессию пользователя в веб-интерфейсе и возвращает экран авторизации

Панель мониторинга и управления

Для просмотра панели мониторинга и управления в левой части веб-интерфейса следует нажать экранную кнопку **[Инфо]**.

В графических блоках панели мониторинга и управления визуализируется краткое представление следующей статистики:

- «TCP, сессий/сек» – отображает количество новых TCP-сессий в секунду;
- «UDP, пакетов/сек» – отображает количество новых UDP-пакетов в секунду;
- «Полоса пропускания, Мбит/сек» – отображает входящую и исходящую скорость передачи данных;
- «ЦПУ» – отображает процент загруженности центрального процессора;
- «Память» – отображает процент загруженности оперативной памяти;
- «Отказоустойчивость» – отображает доступное и общее количество узлов отказоустойчивой конфигурации;
- «Виртуальные Серверы» – отображает доступное и общее количество Виртуальных Серверов;
- «Серверы Балансировки» – отображает доступное и общее количество Серверов Балансировки;
- «ГеоБН Площадки» – отображает количество площадок геораспределенной балансировки подключений;

В текущей версии Termidesk Connect параметр «ГеоБН площадки» не влияет на работу Termidesk Connect. Приведена справочная информация.

- «Версия» – отображает информацию о текущей версии Termidesk Connect;
- «Uptime» – отображает время непрерывной работы Termidesk Connect с момента последнего запуска или перезагрузки.

Панель производительности

Производительность

Для просмотра производительности в левой части веб-интерфейса следует нажать экранную кнопку **[Производительность]**.

Параметры выборки для просмотра производительности приведены в таблице (см. таблицу Параметры выборки для просмотра производительности).

Таблица 79. Параметры выборки для просмотра производительности

Параметр	Описание
«Секция»	Выбор элементов для просмотра. Возможные значения: • «Система» (см. подраздел Производительность системы); • «Виртуальные Серверы» (см. подраздел Производительность Виртуальных Серверов); • «Серверы Балансировки» (см. подраздел Производительность Серверов Балансировки). Значение по умолчанию: «Система»
«Период»	Период отображения данных статистики. Возможные значения: • «5 минут»; • «1 час»; • «1 день»; • «7 дней». Значение по умолчанию: «1 час»
«Интервал обновления»	Интервал обновления информации в графических блоках. Возможные значения: • «5 секунд»; • «1 минута». Значение по умолчанию: «5 секунд»

Для сохранения информации в файл `.pdf` отображаемых данных следует нажать экранную кнопку **[Скачать в PDF]**.

Производительность системы

Для просмотра сведений об утилизации системы в поле «Секция» следует выбрать значение «Система».

В графических блоках доступно переключение видимости графиков, если представлено несколько графиков в одном блоке. Для скрытия или отображения графика следует выполнить нажатие на его обозначение в нижней левой части графического блока.

В графических блоках секции «Система» отобразится следующая информация:

- «CPU» – график загруженности центрального процессора (в процентах);
- «Полоса пропускания» – график входящего и исходящего трафика (Кбит/сек);
- «TCP соединений/с» – график активности новых TCP-сессий;
- «HTTP запросов/с» – график запросов по протоколу HTTP;
- «ВС» – общее количество Виртуальных Серверов;

- «СБ» – общее количество Серверов Балансировки;
- «ГРС» – общее количество Групп Реальных Серверов.

Производительность Виртуальных Серверов

Для просмотра параметров производительности Виртуальных Серверов в поле «Секция» следует выбрать значение «Виртуальные Серверы». Отобразится список Виртуальных Серверов, основные параметры списка приведены в таблице (см. таблицу Список параметров Виртуальных Серверов).

Таблица 80. Список параметров Виртуальных Серверов

Параметр	Описание
«Имя»	Наименование Виртуального Сервера
«Состояние»	Статус Виртуального Сервера
«TCP сессий/с»	Количество новых TCP-сессий
«HTTP запросов/с»	Количество запросов по протоколу HTTP

Для просмотра параметров производительности определенного Виртуального Сервера следует выбрать его из списка нажатием левой кнопкой мыши.

В графических блоках Виртуального Сервера отобразится следующая информация:

- «TCP соединений/с» – график активности новых TCP-сессий (для Виртуального Сервера с типом TCP);
- «HTTP запросов/с» – график запросов по протоколу HTTP (для Виртуального Сервера с типом HTTP);
- «CPU» – график загруженности центрального процессора (в процентах);
- «Полоса пропускания» – график входящего и исходящего трафика (Кбит/сек).

Производительность Серверов Балансировки

Для просмотра параметров производительности Серверов Балансировки в поле «Секция» следует выбрать значение «Серверы Балансировки». Отобразится список Серверов Балансировки, основные параметры списка приведены в таблице (см. таблицу Список параметров Серверов Балансировки).

Таблица 81. Список параметров Серверов Балансировки

Параметр	Описание
«Имя»	Наименование Сервера Балансировки
«Состояние»	Статус Сервера Балансировки
«TCP сессий/с»	Количество новых TCP-сессий
«HTTP запросов/с»	Количество запросов по протоколу HTTP

Для просмотра параметров производительности определенного Сервера Балансировки следует выбрать его из списка нажатием левой кнопкой мыши.

В графических блоках Сервера Балансировки отобразится следующая информация:

- «TCP соединений/с» – график активности новых TCP-сессий;
- «Состояние» – диаграмма состояний Сервера Балансировки.

В области «Группа РС» Сервера Балансировки отобразится список Реальных Серверов, привязанных к данному Серверу Балансировки. Основные параметры списка приведены в таблице (см. таблицу Список параметров Реальных Серверов).

Таблица 82. Список параметров Реальных Серверов

Параметр	Описание
«IP»	IP-адрес Реального Сервера
«Порт»	Порт Реального Сервера
«Состояние»	Статус Реального Сервера
«TCP сессий/с»	Количество новых TCP-сессий
«HTTP запросов/с»	Количество HTTP-запросов

Функция «Система»

Веб. Лицензионное соглашение

Для просмотра лицензионного соглашения следует перейти «Настройки – Система – Лицензия».

Основные параметры идентификационных данных лицензионного соглашения приведены в таблице (см. Основные параметры идентификационных данных).

Таблица 83. Основные параметры идентификационных данных

Параметр	Описание
«HWID»	Идентификатор устройства для активации лицензии
«Customer»	Имя заказчика
«Redaction»	Редакция лицензии
«End Date»	Дата окончания действия лицензии
«CPU»	Количество доступных процессоров

Для загрузки лицензии в области «Файл лицензии» нажать экранную кнопку **[□]** и указать путь к файлу в формате `.zip`.

Веб. DNS

DNS. Общие сведения

Раздел «DNS» предназначен для настройки параметров, обеспечивающих разрешение доменных имен в IP-сетях.

Раздел предоставляет доступ к следующим настройкам:

- «DNS серверы» – позволяет задать список серверов, к которым система будет обращаться для преобразования доменных имен в IP-адреса;
- «Домены поиска» – позволяет настроить автоматическое добавление доменных суффиксов при попытке разрешения неполных доменных имен.

Добавление DNS-сервера

Для отображения списка DNS-серверов следует перейти «Настройки – Система – DNS – DNS серверы».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Сервер».

Основные параметры списка приведены в таблице (см. Основные параметры списка DNS-серверов).

Таблица 84. Основные параметры списка DNS-серверов

Параметр	Описание
«Сервер»	IP-адрес DNS-сервера

Для добавления DNS-сервера следует перейти «Настройки – Система – DNS – DNS серверы» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления DNS-сервера).

Таблица 85. Данные для добавления DNS-сервера

Параметр	Описание
«IP»	IP-адрес DNS-сервера

Добавление домена поиска

Для отображения списка доменов поиска следует перейти «Настройки – Система – DNS – Домены поиска».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Домен».

Основные параметры списка приведены в таблице (см. Основные параметры списка доменов поиска).

Таблица 86. Основные параметры списка доменов поиска

Параметр	Описание
«Домен»	Имя домена поиска

Для добавления DNS-сервера следует перейти «Настройки – Система – DNS – Домены поиска» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления домена поиска).

Таблица 87. Данные для добавления домена поиска

Параметр	Описание
«Имя»	Имя домена поиска. Пример: «example.com»

Веб. NTP

Для отображения списка NTP-серверов следует перейти «Настройки – Система – NTP».

По умолчанию записи представлены табличном виде и упорядочены согласно столбцу «Сервер».

Основные параметры списка приведены в таблице (см. Основные параметры списка NTP-серверов).

Таблица 88. Основные параметры списка NTP-серверов

Параметр	Описание
«Сервер»	IP-адрес или доменное имя NTP-сервера
«Тип сервера»	Приоритет использования указанного NTP-сервера

Для добавления NTP-сервера следует перейти «Настройки – Система – NTP» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в следующей таблице (см. Данные для добавления NTP-сервера).

Таблица 89. Данные для добавления NTP-сервера

Параметр	Описание
«IP / Доменное имя»	IP-адрес или доменное имя NTP-сервера
«Основной»	Приоритет использования указанного NTP-сервера

Для настройки часового пояса в Termidesk Connect нужно выбрать соответствующее значение в раскрывающемся списке «Timezone».

Веб. Отказоустойчивость

Конфигурация кластера

Конфигурация кластера. Общие сведения

Раздел предназначен для настройки параметров отказоустойчивой конфигурации узлов Termidesk Connect. Основная информация об особенностях отказоустойчивой конфигурации приведена в подразделе **Отказоустойчивость**.

Раздел предоставляет доступ к следующим настройкам:

- «Локальные настройки» – позволяет задать локальные настройки узла, которые будут использоваться для взаимодействия с ним;
- «Общий IP управления– позволяет задать общий IP-адрес управления кластером;
- «Узлы» – позволяет добавить информацию о других узлах, которые существуют в отказоустойчивой конфигурации Termidesk Connect. Информация о каждом удаленном узле должна быть доступна с каждого узла.

Добавление локальных настроек

Для отображения локальных настроек, связанных с отказоустойчивой конфигурацией, следует перейти «Настройки – Система – Отказоустойчивость – Конфигурация кластера».

Основные параметры списка области «Локальные настройки» приведены в таблице (см. Основные параметры списка локальных настроек).

Таблица 90. Основные параметры списка локальных настроек

Параметр	Описание
«IP адрес»	Локальный IP-адрес узла
«ID»	Уникальный числовой идентификатор узла

Параметр	Описание
«Статус»	Статус узла в отказоустойчивой конфигурации: • «STANDBY» – узел в данный момент является резервным и не обрабатывает трафик пользователей; • «ACTIVE» – узел в данный момент является мастер-узлом и обрабатывает трафик пользователей. Для принудительного перевода узла в активное состояние нужно нажать левой клавишей мыши на статус узла

Для добавления локальных настроек следует перейти «Настройки – Система – Отказоустойчивость – Конфигурация кластера» и в области «Локальные настройки» нажать экранную кнопку **[Изменить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (Данные для добавления локальных настроек).

Таблица 91. Данные для добавления локальных настроек

Параметр	Описание
«IP»	Локальный IP-адрес узла, используемый для взаимодействия удаленного узла Termidesk Connect с ним
«Порт»	UDP-порт, используемый для взаимодействия удаленного узла Termidesk Connect с ним. Значение по умолчанию: «322»
«ID»	Уникальный числовой идентификатор узла. Значения не должны повторяться в одноранговом кластере отказоустойчивой конфигурации Termidesk Connect. Узел с минимальным идентификатором будет мастер-узлом. Возможные значения: от 0 до 255
«Интервал»	Интервал (в миллисекундах) периодических запросов к узлу. Интервал должен быть одинаковым на всех узлах отказоустойчивой конфигурации Termidesk Connect. Значение по умолчанию: «500»

Добавление общего IP-адреса управления

Для отображения общего IP-адреса управления следует перейти «Настройки – Система – Отказоустойчивость – Конфигурация кластера».

Для изменения IP-адреса нужно нажать экранную кнопку **[Изменить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления IP-адреса управления).

Таблица 92. Данные для добавления IP-адреса управления

Параметр	Описание
«Общий IP управления»	Выбор общего IP-адреса управления кластером. IP-адрес должен быть создан и настроен на мастер-узле, и должен соответствовать типу «ha-type SHARED» Указывается только на мастер-узле.

Добавление узлов отказоустойчивой конфигурации

Для отображения узлов, связанных с отказоустойчивой конфигурацией, следует перейти «Настройки – Система – Отказоустойчивость – Конфигурация кластера».

По умолчанию записи в области «Узлы» представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены в таблице (Основные параметры списка узлов).

Таблица 93. Основные параметры списка узлов

Параметр	Описание
«Имя»	Наименование узла
«IP адрес»	Локальный IP-адрес узла
«ID»	Уникальный числовой идентификатор узла
«Состояние»	Состояние узла
«Статус»	Статус узла в отказоустойчивой конфигурации: • «STANDBY» – узел в данный момент является резервным и не обрабатывает трафик пользователей; • «ACTIVE» – узел в данный момент является мастер-узлом и обрабатывает трафик пользователей
«Синхронизация»	Статус синхронизации конфигурации между узлами кластера: • «X» – обнаружены ошибки в конфигурации при синхронизации. Необходимо подключиться к узлу с ролью «STANDBY» и выполнить поиск причин ошибки в журналах; • «V» – конфигурация синхронизирована успешно

Для добавления узла следует перейти «Настройки – Система – Отказоустойчивость – Конфигурация кластера» и в области «Узлы» нажать экранную кнопку **[Добавить]**.

Экранная кнопка **[Добавить]** будет доступна, если заданы локальные настройки.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. таблицу Данные для добавления узла отказоустойчивой конфигурации).

Таблица 94. Данные для добавления узла отказоустойчивой конфигурации

Параметр	Описание
«Имя»	Наименование узла
«IP узла»	IP-адрес удаленного узла
«Порт»	UDP-порт удаленного узла Значение по умолчанию: «322»
«ID»	Уникальный числовой идентификатор удаленного узла. Значения не должны повторяться в одноранговом кластере. Возможные значения: от 0 до 255

Параметр	Описание
«Пароль»	Пароль для подключения к удаленному узлу.  Должен быть указан пароль пользователя `tdadmin`. Так же пароль может быть получен из Хранилища секретов. Пример значения параметра: • «kv://secret/ha#password» – для `kv` версии 1; • «kv://secret/data/na#password» – для `kv` версии 2

Отслеживание IP-адресов

Для отображения списка отслеживаемых IP-адресов (для готовности узла к переходу в статус «ACTIVE») перейти «Настройки – Система – Отказоустойчивость – Отслеживание IP».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «IP».

Основные параметры списка приведены в таблице (см. таблицу Основные параметры списка IP-адресов).

Таблица 95. Основные параметры списка IP-адресов

Параметр	Описание
«IP»	IP-адрес
«VRF»	Имя VRF

Для добавления IP-адреса перейти «Настройки – Система – Отказоустойчивость – Отслеживание IP» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. таблицу Данные для добавления IP-адреса).

Таблица 96. Данные для добавления IP-адреса

Параметр	Описание
«IP»	IP-адрес
«VRF»	Выбор VRF для добавления IP-адреса. Значение по умолчанию: «default»

Веб. Сеть

Сеть. Общие сведения

Раздел «Сеть» предназначен для настройки сетевых параметров, обеспечивающих корректную маршрутизацию и взаимодействие узлов в IP-сетях.

Раздел предоставляет доступ к настройке следующих сетевых параметров:

- «VRF» – позволяет создавать виртуальные таблицы маршрутизации для изоляции сетевого трафика и разделения маршрутов между различными сетевыми сегментами;
- «Интерфейсы» – позволяет управлять физическими и виртуальными сетевыми интерфейсами, включая их параметры и привязку к VLAN;

- «Агрегация каналов» – позволяет объединять несколько физических каналов в один логический для увеличения общей пропускной способности и повышения надежности соединения;
- «VLAN» – позволяет сегментировать сеть на виртуальные локальные сети для изоляции трафика между группами устройств, подключенными к одной физической сети;
- «IP» – позволяет управлять IP-адресами, связанными с интерфейсами;
- «Маршруты» – позволяет задавать статические маршруты для управления передачей данных между сетевыми сегментами;
- «Фонды IP» – используется для управления пулами IP-адресов, которые могут быть назначены устройствам или интерфейсам.

Добавление VRF

Для отображения списка VRF следует перейти «Настройки – Система – Сеть – VRF».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены в таблице (см. Основные параметры списка VRF).

Таблица 97. Основные параметры списка VRF

Параметр	Описание
«Имя»	Наименование VRF
«Номер таблицы VRF»	Идентификатор VRF

Для добавления VRF следует перейти «Настройки – Система – Сеть – VRF» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления VRF).

Таблица 98. Данные для добавления VRF

Параметр	Описание
«Имя»	Наименование VRF
«Номер таблицы»	Идентификатор VRF

Просмотр списка сетевых интерфейсов

Для отображения списка сетевых интерфейсов следует перейти «Настройки – Система – Сеть – Интерфейсы».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены в таблице (см. Основные параметры списка сетевых интерфейсов).

Таблица 99. Основные параметры списка сетевых интерфейсов

Параметр	Описание
«Имя»	Наименование сетевого интерфейса
«VRF»	Наименование VRF, назначенный сетевому интерфейсу

Параметр	Описание
«Агрегированный интерфейс»	Наименование агрегированного интерфейса, к которому привязан сетевой интерфейс
«MAC-адрес»	MAC-адрес сетевого интерфейса
«ACL»	Список контроля доступа, назначенный сетевому интерфейсу
«DDoS Профиль»	DDoS-Профиль, назначенный сетевому интерфейсу

Для изменения сетевого интерфейса следует выбрать интерфейс и нажать экранную кнопу **[Изменить]**.

Данные, доступные для изменения, перечислены в столбце «Параметр» следующей таблицы (см. Основные параметры изменения сетевого интерфейса).

Таблица 100. Основные параметры изменения сетевого интерфейса

Параметр	Описание
«Имя»	Наименование сетевого интерфейса
«VRF»	Выбор VRF, который будет назначен сетевому интерфейсу
«Агрегированный интерфейс»	Выбор агрегированного интерфейса, к которому будет привязан сетевой интерфейс
«ACL»	Выбор списка контроля доступа, который будет назначен сетевому интерфейсу
«DDoS Профиль»	Выбор DDoS-Профиля, который будет назначен сетевому интерфейсу
«MTU»	MTU для интерфейса
«Состояние»	Активация или отключение сетевого интерфейса
«Отслеживание в HA»	Активация параметра включает отслеживание состояния интерфейса для готовности узла к переходу в статус «ACTIVE»

Агрегация каналов

Для отображения списка агрегированных интерфейсов следует перейти «Настройки – Система – Сеть – Агрегация каналов».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены в таблице (см. Основные параметры списка агрегированных интерфейсов).

Таблица 101. Основные параметры списка агрегированных интерфейсов

Параметр	Описание
«Имя»	Наименование агрегированного интерфейса
«VRF»	Наименование VRF, который назначен агрегированному интерфейсу

Для добавления агрегированного интерфейса следует перейти «Настройки – Система – Сеть – Агрегация каналов» и нажать экранную кнопу **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Основные параметры для добавления агрегированного интерфейса).

Таблица 102. Основные параметры для добавления агрегированного интерфейса

Параметр	Описание
«Имя»	Наименование агрегированного интерфейса

Параметр	Описание
«Тип»	Назначение типа агрегированного интерфейса. Возможные значения: • «LACP» – будет использоваться протокол LACP. После того как агрегированный канал сформирован, за поддержание статуса канала отвечает LACP; • «Active-backup» – будет использоваться только активный интерфейс из объединенных. При отказе активного интерфейса выполняется автоматическое переключение на резервный интерфейс
«LACP»	Назначение режима работы агрегированного интерфейса. Не допускается изменение типа для уже созданного и настроенного интерфейса. Для использования типа LACP он должен поддерживаться сетевым коммутатором. Возможные значения: • «fast» – режим работы, при котором частота отправки LACPDU-пакетов составляет один раз в секунду для более быстрого обнаружения изменений в сети; • «slow» – режим работы, при котором частота отправки LACPDU-пакетов составляет один раз в 30 секунд. Значение по умолчанию: «slow» Параметр доступен при добавлении агрегированного интерфейса с типом «LACP».
«Интервал мониторинга Active Backup»	Назначение частоты (в миллисекундах) мониторинга MII (Media Independent Interface). Значение по умолчанию: «100». Параметр доступен при добавлении агрегированного интерфейса с типом «Active-backup»
«VRF»	Выбор VRF, который будет назначен агрегированному интерфейсу. Значение по умолчанию: «default»

Добавление VLAN

Для отображения списка VLAN следует перейти «Настройки – Система – Сеть – VLAN».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены в таблице (см. Основные параметры списка VLAN).

Таблица 103. Основные параметры списка VLAN

Параметр	Описание
«Имя»	Наименование VLAN
«ID»	Идентификатор VLAN

Параметр	Описание
«Физический интерфейс»	Наименование физического сетевого интерфейса,которому назначен VLAN
«Агрегированный интерфейс»	Наименование агрегированного интерфейса (логического), которому назначен VLAN
«VRF»	Наименование VRF, которому назначен VLAN
«Тип HA»	Режим синхронизации VLAN в отказоустойчивой конфигурации. Может быть: • «Локальный» – локальный VLAN; • «Общий» – VLAN используется в отказоустойчивой конфигурации

Для добавления VLAN следует перейти «Настройки – Система – Сеть – VLAN» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления VLAN).

Таблица 104. Данные для добавления VLAN

Параметр	Описание
«Имя»	Наименование VLAN
«ID»	Идентификатор VLAN
«Физический интерфейс»	Выбор физического сетевого интерфейса, которому будет назначен VLAN
«Агрегированный интерфейс»	Выбор агрегированного интерфейса (логического), которому будет назначен VLAN
«Локальный»	Определяет режим использования VLAN в отказоустойчивой конфигурации. При активации параметра используется только локальная конфигурация. Отключение параметра позволяет синхронизировать конфигурацию между устройствами
«VRF»	Выбор VRF, которой будет назначен VLAN. Значение по умолчанию: «default»

Добавление IP-адресации

Для отображения списка IP-адресов следует перейти «Настройки – Система – Сеть – IP».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Адрес».

Основные параметры списка приведены в таблице (см. Основные параметры списка IP-адресов).

Таблица 105. Основные параметры списка IP-адресов

Параметр	Описание
«Адрес»	IP-адрес, который назначен интерфейсу
«Префикс»	Размер подсети в формате CIDR
«Интерфейс»	Сетевой интерфейс, которому назначен IP-адрес

Параметр	Описание
«Тип HA»	Режим синхронизации IP-адреса в отказоустойчивой конфигурации. Может быть: • «Локальный» – локальный IP-адрес; • «Общий» – IP-адрес используется в отказоустойчивой конфигурации

Для добавления IP-адреса следует перейти «Настройки – Система – Сеть – IP» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления IP-адреса).

Таблица 106. Данные для добавления IP-адреса

Параметр	Описание
«Адрес»	IP-адрес, который будет назначен интерфейсу
«Префикс»	Размер подсети в формате CIDR
«Интерфейс»	Выбор сетевого интерфейса, которому будет назначенIP-адрес. Доступные типы сетевых интерфейсов: «VLAN» или «Ethernet»
«Локальный»	Определяет режим использования IP-адреса в отказоустойчивой конфигурации. При активации параметра используется только локальная конфигурация. Отключение параметра позволяет синхронизировать конфигурацию между устройствами

Добавление параметров маршрутизации

Для отображения списка параметров маршрутизации следует перейти «Настройки – Система – Сеть – Маршруты».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Сеть».

Основные параметры списка приведены в таблице (см. Основные параметры списка параметров маршрутизации).

Таблица 107. Основные параметры списка параметров маршрутизации

Параметр	Описание
«Сеть»	Адрес и маска сети назначения в формате CIDR
«Адрес шлюза»	IP-адрес шлюза, через который будет выполнена маршрутизация
«VRF»	Наименование VRF, назначенного маршруту
«Тип HA»	ежим маршрута в отказоустойчивой конфигурации. Может быть: • «Локальный» – локальный маршрут; • «Общий» – маршрут используется в отказоустойчивой конфигурации

Для добавления параметров маршрутизации следует перейти «Настройки – Система – Сеть – Маршруты» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления параметров маршрутизации).

Таблица 108. Данные для добавления параметров маршрутизации

Параметр	Описание
«Префикс»	Адрес и маска сети назначения в формате CIDR
«Адрес шлюза»	IP-адрес шлюза, через который будет выполнена маршрутизация
«VRF»	Выбор VRF для добавления маршрута. Значение по умолчанию: «default»
«Локальный»	Определяет режим использования маршрута в отказоустойчивой конфигурации. При активации параметра используется только локальная конфигурация. Отключение параметра позволяет синхронизировать конфигурацию между устройствами

Добавление IP-Фондов

Для отображения списка IP-Фондов следует перейти «Настройки – Система – Сеть – Фонды IP».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены в таблице (см. таблицу Основные параметры списка IP-Фондов).

Таблица 109. Основные параметры списка IP-Фондов

Параметр	Описание
«ID»	Наименование IP-Фонда
«VRF»	Имя VRF

Для добавления IP-Фонда следует перейти «Настройки – Система – Сеть – Фонды IP» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. таблицу Данные для добавления IP-Фонда).

Таблица 110. Данные для добавления IP-Фонда

Параметр	Описание
«ID»	Наименование IP-Фонда
«VRF»	Выбор VRF для добавления списка IP-адресов. Значение по умолчанию: «default»
«Список IP адресов»	Список IP-адресов, которым будет разрешено устанавливать соединение с Реальным Сервером. Над списком IP-адресов можно выполнять следующие действия: • добавить, для этого нажать экранную кнопку **[+ Добавить]** и указать соответствующий IP-адрес в открывшемся поле; • удалить, для этого нажать экранную кнопку **[-]** в строке с IP-адресом в списке

Веб. Управление

Управление. Общие сведения

Раздел «Управление» предназначен для настройки доступа к веб-интерфейсу, параметров аутентификации, пользователей и групп Termidesk Connect.

Раздел предоставляет доступ к настройке следующих параметров:

- «Общее» – позволяет настроить общие параметры доступа к веб-интерфейсу;
- «Серверы Аутентификации» – позволяет подключить службу каталогов, существующую в инфраструктуре организации, в качестве средства аутентификации и авторизации администраторов Termidesk Connect;
- «Локальные Пользователи» – позволяет создавать пользователей Termidesk Connect и управлять их настройками;
- «Локальные Группы» – позволяет создавать группы пользователей Termidesk Connect;
- «Правила доступа» – позволяет разграничивать доступ пользователей к настройке и управлению Termidesk Connect.

Управление общими параметрами доступа к веб-интерфейсу

Для изменения настроек доступа к веб-интерфейсу Termidesk Connect следует перейти «Настройки – Система – Управление – Общее» и заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные службы каталогов).

Таблица 111. Данные службы каталогов

Параметр	Описание
«Сертификат пользовательского интерфейса»	Сертификат для подключения к веб-интерфейсу по протоколу HTTPS. При активированном параметре «PKI» для сертификата указывается путь Хранилища секретов. Пример значения параметра: «pki://pki/issue/example-dot-com?common_name=\"app.example.com\"&ttl=\"24h\"#certificate». Значение по умолчанию: «tdc_ss_public_key.pem»
«PKI»	Активация возможности получения файла из Хранилища секретов для сертификата
«Ключ пользовательского интерфейса»	Закрытый ключ к сертификату для подключения к веб-интерфейсу по протоколу HTTPS. При активированном параметре «PKI» для закрытого ключа указывается путь Хранилища секретов. Пример значения параметра: «pki://pki/issue/example-dot-com?common_name=\"app.example.com\"&ttl=\"24h\"#private_key». Значение по умолчанию: «tdc_ss_private_key.key»
«PKI»	Активация возможности получения файла из Хранилища секретов для закрытого ключа
«Порт»	Порт, на котором будет доступен веб-интерфейс. Значение по умолчанию: «443»
«Таймаут, с»	Время жизни (в секундах) сессии пользователя. Значение по умолчанию: «3600»

Для применения параметров следует нажать экранную кнопку **[Применить]**.

Управление аутентификацией

Для подключения службы каталогов в качестве средства аутентификации и авторизации пользователей следует перейти «Настройки – Система – Управление – Серверы Аутентификации» и заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные службы каталогов).

Таблица 112. Данные службы каталогов

Параметр	Описание
«Тип»	Тип подключаемой службы каталогов. Возможные значения: • «AD» – служба каталогов Active Directory Domain Services; • «FreeIPA» – служба каталогов FreeIPA; • «OpenLDAP» – служба каталогов OpenLDAP Directory Services
«Доменное имя сервера»	IP-адрес или доменное имя сервера службы каталогов, являющегося источником сведений о пользователях и их полномочиях
«Время ожидания»	Время ожидания (в секундах) ответа от службы каталогов. Значение по умолчанию: «30»
«Тип безопасности»	Выбор типа безопасности подключения к службе каталогов. Возможные значения: • «TEXT» – незащищенное подключение; • «SSL» – защищенное подключение. Перед обменом данными будет установлена TLS-сессия. Значение по умолчанию: «TEXT»
«Порт»	TCP-порт, на котором запущена служба каталогов. Значение по умолчанию: «389». Возможные стандартные значения: • «389» – используется, если доступ к службе каталогов осуществляется по протоколу LDAP (незащищенное подключение); • «636» – используется, если доступ к службе каталогов осуществляется по протоколу LDAPS (защищенное подключение); • «3268» – альтернативный порт. Используется, если доступ к службе каталогов осуществляется по протоколу LDAP (незащищенное подключение); • «3269» – альтернативный порт. Используется, если доступ к службе каталогов осуществляется по протоколу LDAPS (защищенное подключение)

Параметр	Описание
«Base DN»	Корень поиска в службе каталогов, с которого начинается поиск объектов. Указывается в формате Distinguished Name (DN). Параметру следует задавать значение, соответствующее записи верхнего уровня в иерархии службы каталогов (без указания OU). Вводимое значение не должно содержать пробелов: • в начале и конце строки; • рядом с разделителями (запятыми); • в элементах пути (например, «DC=company name,DC=de»). Примеры (вводятся без кавычек): • для поиска в домене example.loc: «DC=example,DC=loc»; • для поиска только в подразделении Users домена example.loc: «OU=Users,DC=example,DC=loc»
«Administrator Bind DN»	Учетная запись в формате DN, используемая для подключения к службе каталогов. Должна указываться административная учетная запись с правами чтения объектов службы каталогов. Вводимое значение не должно содержать пробелов: • в начале и конце строки; • рядом с разделителями (запятыми); • в элементах пути (например, «DC=company name,DC=de»). Примеры (вводятся без кавычек): • Active Directory Domain Services: «CN=Administrator,OU=Users,DC=example,DC=loc»; • FreeIPA: «UID=admin,CN=users,CN=accounts,DC=example,DC=loc»; • OpenLDAP Directory Services: «CN=admin,DC=example,DC=loc». Так же учетная запись может быть получена из Хранилища секретов. Пример значения параметра: • «kv://secret/data/my_ldap#dn» – для `kv` версии 2; • «ad://ldap/static-cred/my_ldap#dn» – для `ldap`
«Пароль»	Набор символов, подтверждающий полномочия объекта для подключения к службе каталогов. Так же пароль может быть получен из Хранилища секретов. Пример значения параметра: • «kv://secret/data/my_ldap#last_password» – для `kv` версии 2; • «ad://ldap/static-cred/my_ldap#last_password» – для `ldap`

Параметр	Описание
«Атрибут имени пользователя»	Атрибут уникального имени или идентификатора пользователя в службе каталогов. Возможные атрибуты имени пользователя в службе каталогов: • «uid» – уникальный идентификатор учетной записи; • «userPrincipalName» – логин пользователя в формате «user@example.loc»; • «SamAccountName» – короткое имя пользователя; • «cn» – отображаемое имя пользователя (иногда – имя для входа)
«Атрибут группы»	Атрибут группы в службе каталогов, содержащий список участников группы. Возможные атрибуты группы в службе каталогов: • «memberOf» – список участников группы, в котором каждый участник указывается в виде полного DN. Универсальный атрибут; • «uniqueMember» – аналог атрибута выше, но используется в некоторых реализациях LDAP, например, OpenLDAP Directory Services
«Имя сертификата УЦ»	Имя файла сертификата УЦ, который будет использоваться для защищенного подключения к службе каталогов. Файл сертификата должен быть предварительно загружен на Termidesk Connect (см. подраздел **TLS**). При активированном параметре «PKI» для сертификата УЦ указывается путь Хранилища секретов. Пример значения параметра: «pki://pki/issue/example-dot-com?common_name=\"app.example.com\"&ttl=\"24h\"#issuing_ca». Параметр доступен при выборе типа безопасности «SSL»
«PKI»	Активация возможности получения файла из Хранилища секретов. Параметр доступен при выборе типа безопасности «SSL»

Для применения параметров следует нажать экранную кнопку **[Применить]**.

Управление пользователями

Для отображения списка пользователей следует перейти «Настройки – Система – Управление – Локальные Пользователи».

По умолчанию после установки доступны пользователи `tdadmin` и `tdoperator`. Запрещается удалять пользователя `tdadmin`, поскольку это приведет к неработоспособности системы.

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены в таблице (см. Основные параметры списка пользователей).

Таблица 113. Основные параметры списка пользователей

Параметр	Описание
«Имя»	Имя пользователя
«Группы»	Группы пользователя

Для добавления пользователя следует перейти «Настройки – Система – Управление – Локальные Пользователи» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления пользователя).

Таблица 114. Данные для добавления пользователя

Параметр	Описание
«Имя»	Имя пользователя
«Пароль»	Пароль пользователя  В пароле нельзя использовать символы: ", ', \, `, $, !, <, >, ;, {, }, (,), [,], |, *, ?, ~, &, ^, а также управляющие символы – табуляции,переноса строки и возврата каретки.
«Подтвердите пароль»	Повтор ввода пароля пользователя
«Группа»	Группы пользователя

Управление группами

Для отображения списка групп следует перейти Настройки – Система – Управление – Локальные Группы».

Для добавления группы следует нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления группы).

По умолчанию после установки доступны группы `tdadmin` и `tdoperator`. Запрещается удалять группу `tdadmin`, поскольку это приведет к неработоспособности системы.

Таблица 115. Данные для добавления группы

Параметр	Описание
«Имя группы»	Имя группы  Параметр должен соответствовать имени группы в службе каталогов, если настроено подключение к ней.

Управление правилами контроля доступа

Для отображения списка правил контроля доступа следует перейти «Настройки – Система – Управление – Правила доступа».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя списка».

Основные параметры списка приведены в таблице (см. Основные параметры списка правил).

Таблица 116. Основные параметры списка правил

Параметр	Описание
«Имя списка»	Имя списка правил контроля доступа

Параметр	Описание
«Группы»	Список групп, к которым применяются права доступа
«Правила»	Список правил контроля доступа

По умолчанию есть преднастроенные списки правил контроля доступа:

- «admin» – разрешены все действия по настройке Termidesk Connect;
- «operator» – запрещены все действия по настройке Termidesk Connect, разрешен только просмотр.

Для добавления списка правил контроля доступа следует перейти «Настройки – Система – Управление – Правила доступа» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления правил).

Таблица 117. Данные для добавления правил

Параметр	Описание
«Имя списка»	Имя списка правил контроля доступа
«Группы»	Область добавления групп пользователей, которым будут назначены права доступа. Для добавления группы нажать экранную кнопку **[+ Добавить]**. Над группой можно выполнять следующие действия: • добавить новую группу, для этого нажать экранную кнопку **[+]**; • удалить текущую группу, для этого нажать экранную кнопку **[-]**. Значение «*» указывает, что список правил применяется ко всем группам пользователей
«Правила»	Список правил контроля доступа. Правила будут выполняться по порядку. Над списком правил можно выполнять следующие действия: • добавить, для этого нажать экранную кнопку **[Добавить]**; • изменить, для этого выбрать правило в списке и нажать экранную кнопку **[Изменить]** или нажать на имя правила в списке; • удалить, для этого выбрать правило в списке и нажать экранную кнопку **[Удалить]**

Доступные параметры для добавления правила перечислены в столбце «Параметр» следующей таблицы (см. Данные для добавления правила).

Таблица 118. Данные для добавления правила

Параметр	Описание
«Имя»	Имя правила контроля доступа
«xpath»	Путь к элементам конфигурации Termidesk Connect. Предполагает выражения для указания конкретных путей обхода XML-дерева, расположенного в хранилище конфигурации. Осуществляется на языке запросов XPath, определенного в RFC 5261. По заданному выражению определяются права доступа к конкретным элементам конфигурации. Значение по умолчанию: «/». Значение «/» указывает на все возможное содержимое хранилища данных

Параметр	Описание
«Операции»	Операции доступа, которые будут связаны с правилом. Возможные значения: • «*» – все операции и команды; • «edit» – изменение конфигурации Termidesk Connect; • «read» – чтение конфигурации Termidesk Connect; • «exec» – выполнение определенных операций. Например: `commit`, `write`, `restore-broken-config` и т.д. Значение по умолчанию: «*»
«Действие»	Выбор действия по управлению доступом, связанного с правилом. Возможные значения: • «deny» – запретить; • «permit» – разрешить
«Комментарий»	Текстовое описание правила контроля доступа

Веб. Аудит

Для отображения списка syslog-серверов следует перейти «Настройки – Система – Аудит».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя сервера».

Основные параметры списка приведены в таблице (см. Основные параметры списка syslog-серверов).

Таблица 119. Основные параметры списка syslog-серверов

Параметр	Описание
«Имя сервера»	Наименование сервера
«IP»	IP-адрес syslog-сервера
«Порт»	Порт для подключения к syslog-серверу
«Протокол»	Используемый протокол для передачи событий
«LogLevel»	Уровень журналирования событий

Для добавления syslog-сервера следует перейти «Настройки – Система – Аудит» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления syslog-сервера).

Таблица 120. Данные для добавления syslog-сервера

Параметр	Описание
«Имя сервера»	Наименование сервера
«IP сервера»	IP-адрес syslog-сервера
«IP источника»	Выбор IP-адреса источника для передачи событий в журнал

Параметр	Описание
«Порт»	Порт для подключения к syslog-серверу
«Уровень сообщений»	Выбор уровня событий для записи в журнал. Возможные значения: • «Emergency»; • «Critical»; • «Warning»; • «Info»; • «Alert»; • «Error»; • «Notice»; • «Debug»
«Категория»	Выбор категории событий для записи в журнал. Возможные значения: • «local0»; • «local1»; • «local2»; • «local3»; • «local4»; • «local5»; • «local6»; • «local7»
«Сервис»	Выбор службы, события которой будут записаны в журнал. Возможные значения: • «LocalLB» – запись событий службы локальной балансировки; • «Kernel» – запись событий ядра ОС; • «GSLB» – запись событий службы глобальной балансировки; • «System» – запись системных событий
«Протокол»	Используемый протокол для передачи событий. Возможные значения: • «TCP»; • «UDP»; • «TLS»
«Имя файла ключа УЦ»	Имя файла закрытого ключа для защищенного соединения SSL/TLS. При активированном параметре «PKI» для закрытого ключа указывается путь Хранилища секретов. Пример значения параметра: «pki://pki/issue/example-dot-com?common_name=\"app.example.com\"&ttl=\"24h\"#private_key» Параметр доступен при использовании протокола «TLS»

Параметр	Описание
«PKI»	Активация возможности получения файла из Хранилища секретов для закрытого ключа. Параметр доступен при использовании протокола «TLS»
«Имя файла сертификата УЦ»	Имя файла сертификата УЦ для защищенного соединения SSL/TLS. При активированном параметре «PKI» для сертификата УЦ указывается путь Хранилища секретов. Пример значения параметра: «pki://pki/issue/example-dot-com?common_name=\"app.example.com\"&ttl=\"24h\"#certificate». Параметр доступен при использовании протокола «TLS»
«PKI»	Активация возможности получения файла из Хранилища секретов для сертификата УЦ. Параметр доступен при использовании протокола «TLS»
«Проверка верификации»	Использование проверки подлинности сертификата для защищенного соединения SSL/TLS. Значение по умолчанию: «Нет». Параметр доступен при использовании протокола «TLS»

Веб. SNMP

SNMP. Настройка

Общие сведения о настройке SNMP

Раздел «SNMP – Настройки» предназначен для управления настройками SNMP версий 1 и 2c в Termidesk Connect.

Раздел предоставляет доступ к настройке следующих параметров:

- «Адреса агентов» – позволяет задать адреса агентов, на которых SNMP-сервис будет ожидать запросы и управлять их настройками;
- «SNMP Community» – позволяет задать сообщества для доступа к данным SNMP-сервиса.

Добавление адреса агента

Для отображения адресов агентов следует перейти «Настройки – Система – SNMP – Настройки».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «IP».

Основные параметры списка области «Адреса агентов» приведены в таблице (см. Основные параметры списка адресов агентов).

Таблица 121. Основные параметры списка адресов агентов

Параметр	Описание
«IP»	IP-адрес, на котором работает SNMP
«Порт»	Порт, на котором работает SNMP
«Протокол»	Протокол, по которому работает SNMP

Для добавления адреса агента следует перейти «Настройки – Система – SNMP – Настройки» и в области «Адреса агентов» нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления адреса агента).

Таблица 122. Данные для добавления адреса агента

Параметр	Описание
«IP»	IP-адрес, на котором будет работать SNMP
«Порт»	Порт, на котором будет работать SNMP. Значение по умолчанию: «161»
«Протокол»	Протокол, по которому будет работать SNMP. Возможные значения: • «UDP»; • «TCP». Значение по умолчанию: «UDP»

Добавление сообщества SNMP

Для отображения сообществ SNMP следует перейти «Настройки – Система – SNMP – Настройки».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка области «SNMP Community» приведены в таблице (см. Основные параметры списка сообществ SNMP).

Таблица 123. Основные параметры списка сообществ SNMP

Параметр	Описание
«Имя»	Имя сообщества
«Уровень доступа»	Уровень доступа сообщества
«Источник»	Сеть, из которой осуществлен доступ к сообществу

Для добавления сообщества SNMP следует перейти «Настройки – Система – SNMP – Настройки» и в области «SNMP Community» нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления сообщества SNMP).

Таблица 124. Данные для добавления сообщества SNMP

Параметр	Описание
«Имя»	Имя сообщества

Параметр	Описание
«Уровень доступа»	Уровень доступа сообщества к данным SNMP. Возможные значения: • «RO»; • «RW». Значение по умолчанию: «RO»
«Разрешённый источник»	Сеть, из которой разрешены запросы для сообщества SNMP. Значение по умолчанию: «0.0.0.0/0»

SNMP. Пользователи

Общие сведения о пользователях SNMP

Раздел «SNMP – Пользователи» предназначен для настройки пользователей SNMPv3 в Termidesk Connect.

Раздел предоставляет доступ к настройке следующих параметров:

- «Пользователи» – позволяет управлять настройками пользователя SNMPv3;
- «Настроить EngineID» – позволяет настроить идентификатор агента SNMP для сценариев, где требуется его уникальность.

Добавление пользователя

Для отображения пользователей следует перейти «Настройки – Система – SNMP – Пользователи».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Пользователь».

Основные параметры списка области «Пользователи» приведены в таблице (см. Основные параметры списка пользователей).

Таблица 125. Основные параметры списка пользователей

Параметр	Описание
«Пользователь»	Имя пользователя
«Тип аутентификации»	Протокол аутентификации пользователя
«Протокол»	Протокол конфиденциальности пользователя

Для добавления пользователя следует перейти «Настройки – Система – SNMP – Пользователи» и в области «Пользователи» нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления пользователя).

Таблица 126. Данные для добавления пользователя

Параметр	Описание
«Имя пользователя»	Имя пользователя (доступно значение от 1 до 64 символов). Так же имя пользователя может быть получено из Хранилища секретов. Пример значения параметра: • «kv://secret/snmp#username» – для kv версии 1; • «kv://secret/data/snmp#username» – для kv версии 2
«Протокол аутентификации»	Протокол аутентификации пользователя. Возможные значения: • «NONE»; • «MD5»; • «SHA»; • «SHA224»; • «SHA256»; • «SHA384»; • «SHA512». Значение по умолчанию: «NONE»
«Пароль аутентификации»	Пароль аутентификации пользователя. Так же пароль может быть получен из Хранилища секретов. Пример значения параметра: • «kv://secret/snmp#auth_key» – для kv версии 1; • «kv://secret/data/snmp#auth_key» – для kv версии 2. Параметр доступен при указании протокола аутентификации со значением, отличным от «NONE»
«Протокол конфиденциальности»	Протокол конфиденциальности пользователя. Возможные значения: • «NONE»; • «DES»; • «AES»; • «AES192»; • «AES256». Значение по умолчанию: «NONE»
«Пароль конфиденциальности»	Пароль конфиденциальности пользователя. Так же пароль может быть получен из Хранилища секретов. Пример значения параметра: • «kv://secret/snmp#priv_key» – для kv версии 1; • «kv://secret/data/snmp#priv_key» – для kv версии 2. Параметр доступен при указании протокола конфиденциальности со значением, отличным от «NONE»

Параметр	Описание
«Уровень доступа»	Уровень доступа пользователя. Возможные значения: • «RO»; • «RW»

Настройка идентификатора агента SNMP

Для настройки идентификатора агента SNMP следует перейти «Настройки – Система – SNMP – Пользователи» и в области «Пользователи» нажать экранную кнопку **[Настроить EngineID]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для настройки идентификатора агента SNMP).

Таблица 127. Данные для настройки идентификатора агента SNMP

Параметр	Описание
«Тип EngineID»	ип идентификатора агента SNMP. Возможные значения: • «1» – IPv4-адрес; • «2» – IPv6-адрес; • «3» – MAC-адрес Обратите внимание, что изменение IP-адреса (или замена сетевой карты) может вызвать проблемы.

Веб. Обновление

Для обновления Termidesk Connect следует перейти «Настройки – Система – Обновление», нажать экранную кнопку **[Загрузить]** и указать путь к файлу обновления в формате `.img`.

Функция «Управление трафиком»

Веб. Проверки

Добавление Проверки

Раздел «Проверки» предоставляет доступ к следующим возможностям:

- создание файлов скриптов Проверок;
- создание и управление Проверками.

Для отображения списка файлов скриптов Проверок следует перейти «Настройки – Управление трафиком – Проверки – Скрипты Проверок». Подробные сведения о работе с файлами скриптов приведены в подразделе **Скрипты Проверок**.

Для отображения списка Проверок следует перейти «Настройки – Управление трафиком – Проверки».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены в таблице (см. Основные параметры списка Проверок).

Таблица 128. Основные параметры списка Проверок

Параметр	Описание
«Имя»	Наименование Проверки
«Тип»	Тип Проверки. Возможные значения: • «TCP»; • «ICMP»; • «HTTP»; • «USER»; • «COMBO»
«VRF»	Имя VRF

Для добавления Проверки следует перейти «Настройки – Управление трафиком – Проверки» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления Проверки доступности сервера).

Таблица 129. Данные для добавления Проверки доступности сервера

Параметр	Описание
«Имя»	Наименование Проверки
«Тип»	Тип Проверки. Возможные значения: • «TCP» – проверяет доступность узла, выполняя попытку подключения к указанному TCP-порту; • «ICMP» – проверяет доступность узла с помощью ICMP-запросов (ping); • «HTTP» – проверяет доступность и работоспособность Сервиса на основании ответов на HTTP-запросы; • «USER» – проверяет доступность на основании логики, определенной пользовательским скриптом; • «COMBO» – проверяет доступность на основании комбинации ранее созданных Проверок
«Интервал, с»	Период (в секундах) между Проверками доступности узла. Значение по умолчанию: «10». Параметр доступен при добавлении Проверки с типом «ICMP», «TCP», «HTTP» или «USER»
«Время ожидания ответа, с»	Время (в секундах) ожидания ответа от узла. Значение по умолчанию: «5». Параметр доступен при добавлении Проверки с типом «ICMP», «TCP», «HTTP» или «USER»

Параметр	Описание
«Кол-во попыток, шт»	Максимальное количество попыток определения состояния узла, необходимое для признания его недоступным. Значение по умолчанию: «1». Параметр доступен при добавлении Проверки с типом «ICMP», «TCP», «HTTP» или «USER»
«Успешное кол-во попыток, шт»	Минимальное количество успешных попыток определения состояния узла, необходимое для признания его доступным. Значение по умолчанию: «1». Параметр доступен при добавлении Проверки с типом «ICMP», «TCP», «HTTP» или «USER»
«IP источника запроса»	IP-адрес, с которого инициируется запрос или устанавливается соединение. Параметр доступен при добавлении Проверки с типом «ICMP», «TCP», «HTTP» или «USER»
«IP назначения»	IP-адрес назначения, к которому будет направлен запрос или установлено соединение. Если сервис на указанном IP-адресе недоступен, этот адрес исключается из списка балансировки. Параметр доступен при добавлении Проверки с типом «ICMP», «TCP», «HTTP» или «USER»
«VRF»	Выбор VRF для добавления Проверки. Значение по умолчанию: «default». Параметр доступен при добавлении Проверки с типом «ICMP», «TCP», «HTTP» или «USER»
«Отслеживание в HA»	Активация параметра включает отслеживание Проверки для готовности узла к переходу в статус «ACTIVE». При активации данного параметра станет недоступен параметр «IP источника запроса». Параметр доступен при добавлении Проверки с типом «TCP», «ICMP», «HTTP» или «USER»
«Порт назначения»	Порт назначения, к которому будет направлен запрос или установлено соединение. Если служба на указанном порту недоступна, то узел будет исключен из списка балансировки. Параметр доступен при добавлении Проверки с типом «TCP», «HTTP» или «USER»
«Скрипт проверки»	Выбор файла скрипта, согласно которому будет выполняться Проверка. Параметр доступен при добавлении Проверки с типом «USER»

Параметр	Описание
«Метод»	Метод запроса, по которому выполняется Проверка. Возможные значения: • «GET» – запрос с получением тела ответа; • «HEAD» – запрос с получением заголовка ответа; • «POST» – отправка данных с телом запроса. Значение по умолчанию: «HEAD». Параметр доступен при добавлении Проверки с типом «HTTP»
«URL»	Путь, по которому выполняется Проверка. Значение по умолчанию: «/». Параметр доступен при добавлении Проверки с типом «HTTP»
«Ожидаемые коды ответа»	Код ответа, по которому выполняется Проверка. Для добавления кода ответа нужно нажать экранную кнопку **[+]** и указать код, ожидаемый в ответе на HTTP-запрос. Значение по умолчанию: «200». Параметр доступен при добавлении Проверки с типом «HTTP»
«Ожидаемая строка в ответе»	Строка в ответе, по которой выполняется Проверка. Параметр доступен при добавлении Проверки с типом «HTTP»
«Заголовки запроса»	Заголовок запроса, по которому выполняется Проверка. Для добавления заголовка запроса нужно нажать экранную кнопку **[Добавить]** и заполнить поля: • «Имя» – ключ заголовка HTTP-запроса; • «Значение» – значение, соответствующее ключу заголовка. Параметр доступен при добавлении Проверки с типом «HTTP»
«TLS»	Использование защищенного SSL-соединения. Параметр доступен при добавлении Проверки с типом «HTTP»
«TLS профиль»	Выбор Профиля для защищенного SSL-соединения. Параметр доступен при добавлении Проверки с типом «HTTP» и активированном параметре «TLS»
«Reverse»	Параметр инверсии результата Проверки. Параметр доступен при добавлении Проверки с типом «HTTP»
«Максимальный вес суммы проверок»	Пороговое значение успеха для суммы Весов базовых Проверок. Параметр доступен при добавлении Проверки с типом «COMBO»
«ID проверки»	Идентификатор базовой Проверки. Параметр доступен при добавлении Проверки с типом «COMBO»

Параметр	Описание
«Вес»	Вес базовой Проверки. Возможные значения: от 1 до 255. Параметр доступен при добавлении Проверки с типом «COMBO»

Добавление файлов скриптов проверок

Для загрузки файлов скриптов проверок следует перейти «Настройки – Управление трафиком – Проверки – Скрипты Проверок» и нажать экранную кнопку:

- **[Загрузить]** для загрузки файла в Termidesk Connect. Допускается загрузка только файлов в формате `.py` и `.sh`;
- **[Добавить]** для создания файла скрипта Проверки и его заполнения в интерактивном режиме. Для создания файла потребуется заполнить параметры (см. Данные для добавления файла скрипта Проверки) и нажать экранную кнопку **[Применить]**. Экранная кнопка **[Закрыть]** закроет интерактивный режим без сохранения изменений.

После загрузки или создания файла он будет расположен в каталоге `/var/lib/tdc/lbscripts/hc/`.

Для удаления файла нужно:

- отметить его и нажать экранную кнопку **[Удалить]**;
- подтвердить удаление, нажав экранную кнопку **[Удалить]** на запрос об удалении.

Для изменения файла нужно отметить его и нажать экранную кнопку **[Изменить]**.

Таблица 130. Данные для добавления файла скрипта Проверки

Параметр	Описание
«Имя»	Наименование файла на латинице с указанием формата: • `.py` – для python-скрипта; • `.sh` – для bash-скрипта  При задании наименования файла, уже имеющегося в Termidesk Connect, исходный файл будет перезаписан.
«Синтаксис»	Язык интерпретатора, используемого при выполнении скрипта. Возможные значения: • «python»; • «bash»
«Содержимое файла»	Содержимое файла. В содержимом можно определять условия выполнения того или иного действия логическими выражениями. Редактор поддерживает функцию автоматического дополнения выражений по мере их ввода.  Автоматическая проверка синтаксиса при работе в интерактивном режиме не производится. Примеры скриптов приведены в подразделе: **Скрипты Проверок**

Веб. Группы Реальных Серверов

Для отображения списка Групп Реальных Серверов следует перейти «Настройки – Управление трафиком – Группы Реальных Серверов».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены в таблице (см. таблицу Основные параметры списка Групп Реальных Серверов).

Таблица 131. Основные параметры списка Групп Реальных Серверов

Параметр	Описание
«Имя»	Наименование Группы Реальных Серверов
«Статус»	Статус Группы Реальных Серверов
«Проверка»	Проверка, назначенная Группе Реальных Серверов
«Фонд IP»	IP-Фонд, назначенный Группе Реальных Серверов

Для добавления Группы Реальных Серверов следует перейти «Настройки – Управление трафиком – Группы Реальных Серверов» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. таблицу Данные для добавления Группы Реальных Серверов).

Таблица 132. Данные для добавления Группы Реальных Серверов

Параметр	Описание
«Имя»	Наименование Группы Реальных Серверов
«Используемая Проверка»	Выбор Проверки доступности узла
«Фонд IP»	Выбор IP-Фонда, которому будет разрешено устанавливать соединение со списком Реальных Серверов
«Время обслуживания, с»	Время (в секундах), по истечении которого все сессии (если они остались), сбросятся, а записи привязки сессии пользователя удалятся
«Список Реальных Серверов»	Над списком Реальных Серверов можно выполнять следующие действия: • добавить, для этого нажать экранную кнопку **[Добавить]**. Выбрать тип: «IP» или «Domain»; • удалить, для этого выбрать нужные серверы в списке и нажать экранную кнопку **[Удалить]**; • перевести во включенное состояние, для этого выбрать нужный сервер в списке и нажать экранную кнопку **[Включить]**; • перевести в выключенное состояние, для этого выбрать нужный сервер в списке и нажать экранную кнопку **[Выключить]**; • перевести в режим техобслуживания, для этого выбрать нужный сервер в списке и нажать экранную кнопку **[Обслуживание]**
«Профиль ограничения потока данных»	Выбор Профиля ограничения скорости
«Комментарий»	Комментарий, который будет привязан к Группе Реальных Серверов

Форма «Список Реальных Серверов» позволяет задать список Реальных Серверов.

Доступные параметры для добавления Реального Сервера с типом «IP» перечислены в столбце «Параметр» следующей таблицы (см. таблицу Данные для добавления Реального Сервера с типом «IP»).

Таблица 133. Данные для добавления Реального Сервера с типом «IP»

Параметр	Описание
«IP»	IP-адрес Реального Сервера
«Порт»	Порт Реального Сервера
«Любой»	Если параметр активирован, то порт из запроса будет сохранен и передастся на Реальный Сервер без изменений
«Вес»	Вес Реального Сервера. Возможные значения: от 1 до 100. Значение по умолчанию: «1»

Доступные параметры для добавления Реального Сервера с типом «Domain» перечислены в столбце «Параметр» следующей таблицы (см. таблицу Данные для добавления Реального Сервера с типом «Domain»).

Таблица 134. Данные для добавления Реального Сервера с типом «Domain»

Параметр	Описание
«Доменное имя»	Доменное имя Реального Сервера
«Порт»	Порт Реального Сервера
«Любой»	Если параметр активирован, то порт из запроса будет сохранен и передастся на Реальный Сервер без изменений
«Вес»	Вес Реального Сервера. Возможные значения: от 1 до 100. Значение по умолчанию: «1»
«TTL»	Время жизни информации о доменном имени (в секундах). Значение по умолчанию: «60»
«Масштабирование»	Автоматическое масштабирование Группы Реальных Серверов. При включении параметра будут использоваться все разрешенные IP-адреса в DNS-ответе (если их несколько), при выключении будет использоваться первый разрешенный IP-адрес в DNS-ответе (если их несколько)

Веб. Профили

TCP-Профили

Для отображения списка TCP-Профилей следует перейти «Настройки – Управление трафиком – Профили – TCP Профили».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены таблице (см. Основные параметры списка TCP-Профилей).

Таблица 135. Основные параметры списка TCP-Профилей

Параметр	Описание
«Имя»	Наименование TCP-Профиля
«Тип»	Тип TCP-Профиля

Для добавления TCP-Профиля следует перейти «Настройки – Управление трафиком – Профили – TCP Профили» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления TCP-Профиля).

Таблица 136. Данные для добавления TCP-Профиля

Параметр	Описание
«Имя»	Наименование TCP-Профиля
«Тип»	Выбор типа TCP-Профиля. Возможные значения: • «Клиентский»; • «Серверный»
«Таймаут записи в сокет, с»	Время ожидания (в секундах) записи в сокет. Значение по умолчанию: «60»
«TCP No Delay»	Активация использования алгоритма Нейгла
«Размер буфера»	Размер буфера для чтения. Значение по умолчанию: «32». Доступен выбор единиц измерения для указанного значения: • Б – байт; • КиБ – килобайт

Параметр	Описание
«Алгоритм предотвращения перегрузок»	Выбор алгоритма предотвращения перегрузок. Алгоритм может быть: • «BBR»; • «BIC»; • «CDG»; • «CUBIC»; • «DCTCP»; • «HIGHSPEED»; • «HTCP»; • «HYBLA»; • «ILLINOIS»; • «LP»; • «NV»; • «RENO»; • «VEGAS»; • «VENO»; • «WESTWOOD»; • «YEAH». Значение по умолчанию: «CUBIC»
«Таймаут соединения с Реальным Сервером, с»	Время ожидания (в секундах) соединения с Реальным Сервером. Значение по умолчанию: «10». Параметр доступен для TCP-Профиля типа «Клиентский»
«Ограничение времени отсутствия данных в сессии, с»	Время (в секундах) отсутствия данных в сессии. Значение по умолчанию: «120». Параметр доступен для TCP-Профиля типа «Клиентский»
«Proxy Protocol»	Активация PROXY-протокола
«v1»	Активация версии 1. Параметр доступен при активации «Proxy Protocol»
«PROXY UNKNOWN»	Разрешение приема заголовка `PROXY UNKNOWN`. Параметр доступен при активации «Proxy Protocol» и «v1» для TCP-Профиля типа «Серверный»
«v2»	Активация версии 2. Параметр доступен при активации «Proxy Protocol»

Параметр	Описание
«TLV из клиентского подключения»	Область добавления TLV, который будет передан из клиентского подключения в сторону Реального Сервера. Для доступа к параметрам области нажать экранную кнопку **[+ Добавить]**. Может быть добавлено несколько TLV. Параметр доступен при активации «Proxy Protocol» и «v2» для TCP-Профиля типа «Клиентский»
«Добавить TLV»	Область добавления TLV, который добавляется в заголовок PROXY-протокола. Для доступа к параметрам области нажать экранную кнопку **[+ Добавить]**. Может быть добавлено несколько TLV. Параметр доступен при активации «Proxy Protocol» и «v2» для TCP-Профиля типа «Клиентский»
«LOCAL»	Разрешение приема заголовка, содержащего адрес `AF_UNIX`. Параметр доступен при активации «Proxy Protocol» и «v2» для TCP-Профиля типа «Серверный»
«AF_UNIX»	Разрешение приема заголовка, содержащего адрес `UNSPEC`. Параметр доступен при активации «Proxy Protocol» и «v2» для TCP-Профиля типа «Серверный»
«UNSPEC»	Разрешение приема заголовка, содержащего команду `LOCAL`. Параметр доступен при активации «Proxy Protocol» и «v2» для TCP-Профиля типа «Серверный»
«Формирование адреса для заголовка Proxy Protocol»	Активация формирования адреса для заголовка PROXY-протокола. Параметр доступен при активации «Proxy Protocol» для TCP-Профиля типа «Клиентский»
«Подключение без Proxy Protocol»	Разрешение подключения пользователя без передачи заголовка PROXY-протокола. Параметр доступен при активации «Proxy Protocol» для TCP-Профиля типа «Серверный»
«Время ожидания получения заголовка Proxy Protocol»	Время ожидания (в секундах) получения заголовка PROXY-протокола от пользователя. Значение по умолчанию: «10». Параметр доступен при активации «Proxy Protocol» для TCP-Профиля типа «Серверный»
«TCP Keep-Alive»	Активация проверки активности соединения
«Время до отправки пробы KA, с»	Время бездействия (в секундах) соединения перед отправкой пакетов проверки. Значение по умолчанию: «900». Параметр доступен при активации «TCP Keep-Alive»

Параметр	Описание
«Интервал отправки проб KA, с»	Интервал (в секундах) отправки пакетов проверки. Значение по умолчанию: «75». Параметр доступен при активации «TCP Keep-Alive»
«Количество проб KA»	Количество пакетов проверки, которые следует отправить при отсутствии подтверждения от узла. Значение по умолчанию: «3». Параметр доступен при активации «TCP Keep-Alive»
«Комментарий»	Комментарий, который будет привязан к TCP-Профилю

HTTP-Профили

Для отображения списка HTTP-Профилей следует перейти «Настройки – Управление трафиком – Профили – HTTP Профили».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены таблице (см. Основные параметры списка HTTP-Профилей).

Таблица 137. Основные параметры списка HTTP-Профилей

Параметр	Описание
«Имя»	Наименование HTTP-Профиля
«Тип»	Тип HTTP-Профиля

Для добавления HTTP-Профиля следует перейти «Настройки – Управление трафиком – Профили – HTTP Профили» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления HTTP-Профиля).

Таблица 138. Данные для добавления HTTP-Профиля

Параметр	Описание
«Имя»	Наименование HTTP-Профиля
«Тип»	Выбор типа HTTP-Профиля. Возможные значения: • «Клиентский»; • «Серверный»
«Значения из заголовка Upgrade»	Значения (может быть несколько) из заголовка Upgrade, для которых разрешена смена протокола. Для добавления значения следует нажать экранную кнопку **[+]**. Параметр доступен только для HTTP-Профиля типа «Клиентский»
«Таймаут чтения из сокета, с»	Время ожидания (в секундах) чтения из сокета
«Максимальное число заголовков»	Максимальное число заголовков в запросе, при котором он считается валидным

Параметр	Описание
«Максимальный размер всех заголовков»	Максимальный размер всех заголовков. Доступен выбор единиц измерения для указанного значения: • Б – байт; • КиБ – килобайт
«Максимальный размер одного заголовка»	Максимальный размер одного заголовка. Доступен выбор единиц измерения для указанного значения: • Б – байт; • КиБ – килобайт
«Максимальный размер тела запроса»	Максимальный размер тела HTTP-запроса. Доступен выбор единиц измерения для указанного значения: • Б – байт; • КиБ – килобайт; • МиБ – мебибайт. Параметр доступен только для HTTP-Профиля типа «Серверный». Значение по умолчанию: «0»
«Максимальный размер тела ответа»	Максимальный размер тела HTTP-ответа. Доступен выбор единиц измерения для указанного значения: • Б – байт; • КиБ – килобайт; • МиБ – мебибайт. Параметр доступен только для HTTP-Профиля типа «Клиентский». Значение по умолчанию: «0»
«CONNECT»	Активация или отключение проксирования метода CONNECT с переключением на TCP. Параметр доступен только для HTTP-Профиля типа «Клиентский»
«Proxy 100»	Активация или отключение обработки заголовка `Expect: 100-continue`. Параметр доступен только для HTTP-Профиля типа «Клиентский»
«HTTP Keep-Alive»	Активация или отключение проверки активности соединения. Параметр доступен только для HTTP-Профиля типа «Серверный»
«Время ожидания следующего запроса, с»	Время ожидания (в секундах) следующего запроса в соединении. Параметр доступен только для HTTP-Профиля типа «Серверный»
«Максимальное число запросов»	Максимальное количество запросов, которое может быть отправлено в этом соединении перед его закрытием. Параметр доступен только для HTTP-Профиля типа «Серверный»

Параметр	Описание
«Сценарий ошибок»	Файл Сценария ошибок. Параметр доступен только для HTTP-Профиля типа «Серверный»
«Комментарий»	Комментарий, который будет привязан к HTTP-Профилю

Профили сохранения сессий

Для отображения списка Профилей сохранения сессий следует перейти «Настройки – Управление трафиком – Профили – Сохранение Сессий».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены таблице (см. таблицу Основные параметры списка Профилей сохранения сессий).

Таблица 139. Основные параметры списка Профилей сохранения сессий

Параметр	Описание
«Имя»	Наименование Профиля сохранения сессий
«Тип»	Тип Профиля сохранения сессий

Для добавления Профиля сохранения сессий следует перейти «Настройки – Управление трафиком – Профили – Сохранение Сессий» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. таблицу Данные для добавления Профиля сохранения сессий).

Таблица 140. Данные для добавления Профиля сохранения сессий

Параметр	Описание
«Имя»	Наименование Профиля сохранения сессий
«Тип»	Выбор типа Профиля сохранения сессий. Возможные значения: • «COOKIE» – привязка по cookie, который получен в ответе Реального Сервера; • «HEADER» – привязка по значению заголовка, указанного в конфигурации; • «IPSOURCE» – привязка по IP-адресу источника запроса, т.е. IP-адресу пользователя; • «SSLSESSION» – привязка пользователя к Реальному Серверу по идентификатору SSL-сессии. Значение по умолчанию: «IPSOURCE»
«Название cookie»	Имя cookie, который ожидается в ответе Реального Сервера для повторного подключения пользователя на ранее выбранный Реальный Сервер. Параметр доступен при выборе Профиля сохранения сессий с типом «COOKIE»

Параметр	Описание
«Название HTTP заголовка»	Заголовок, по которому повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер. Параметр доступен при выборе Профиля сохранения сессий с типом «HEADER»
«Таймаут»	Время ожидания (в секундах), в течение которого повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер. Значение по умолчанию: «60»

Веб. Серверы Балансировки

Для отображения списка Серверов Балансировки следует перейти «Настройки – Управление трафиком – Серверы Балансировки».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены в таблице (см. таблицу Основные параметры списка Серверов Балансировки).

Таблица 141. Основные параметры списка Серверов Балансировки

Параметр	Описание
«Имя»	Наименование Сервера Балансировки
«Тип»	Используемый протокол для балансировки входящих подключений
«Состояние»	Статус Сервера Балансировки
«Алгоритм»	Используемый алгоритм балансировки
«TLS Профиль»	Используемый Клиентский SSL-Профиль для аутентификации на Реальном Сервере

Для добавления Сервера Балансировки следует перейти «Настройки – Управление трафиком – Серверы Балансировки» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. таблицу Данные для добавления Сервера Балансировки).

Таблица 142. Данные для добавления Сервера Балансировки

Параметр	Описание
«Имя»	Наименование Сервера Балансировки
«Тип»	Выбор протокола, который будет использован для балансировки подключений. Возможные значения: • «TCP»; • «HTTP»; • «RAPID-TCP»; • «RAPID-UDP»
«Группа РС»	 В текущей версии Termidesk Connect доступна возможность выбора только одной Группы Реальных Серверов. Выбор Группы Реальных Серверов

Параметр	Описание
«Минимальное количество РС, шт»	Минимально необходимое количество доступных Реальных Серверов в группе для перевода Сервера Балансировки в статус «В работе». Значение по умолчанию: «1»
«Отслеживание в HA»	Активация параметра включает отслеживание состояния Сервера Балансировки для готовности узла к переходу в статус «ACTIVE»
«Алгоритм»	Выбор алгоритма, который будет использован для балансировки подключений. Возможные значения: • «LEASTCONN» – направление запроса на наименее загруженный Реальный Сервер; • «WEIGHTEDLEASTCONN» – направление запроса на Реальный Сервер с наименьшим соотношением числа соединений к его Весу; • «WEIGHTEDLEASTCONNECTTIME» – направление запроса на Реальный Сервер с наименьшим соотношением среднего времени соединения и количества текущих соединений к его Весу. Этот алгоритм не используется для Сервера Балансировки с типом «RAPID-UDP»; • «ROUNDROBIN» – поочередное распределение запросов между Реальными Серверами в группе; • «WEIGHTEDROUNDROBIN» – поочередное распределение запросов между Реальными Серверами в группе пропорционально их Весу; • «RANDOM» – выбор случайного Реального Сервера; • «POWEROFTWORANDOM» – выбор Реального Сервера с наименьшим числом соединений из двух Реальных Серверов, выбранных случайным образом; • «WEIGHTEDLEASTRESPONSETIME» – направление запроса на Реальный Сервер с наименьшим соотношением среднего времени соединения с сервером, наименьшим средним временем получения первого байта ответа и количества текущих сессий к Весу. Этот алгоритм доступен только для Сервера Балансировки с типом «HTTP». Значение по умолчанию: «LEASTCONN»
«Время старта, с»	Период (в секундах), в течение которого будет запущено выполнение алгоритма «ROUNDROBIN», «WEIGHTEDROUNDROBIN». Параметр позволяет выполнить равномерное распределение запросов между Реальными Серверами с момента, когда один или несколько серверов станут доступны. Параметр доступен при выборе алгоритмов «LEASTCONN» (будет запущено выполнение алгоритма «ROUNDROBIN»), «WEIGHTEDLEASTCONN», «WEIGHTEDLEASTCONNECTTIME» или «WEIGHTEDLEASTRESPONSETIME» (будет запущено выполнение алгоритма «WEIGHTEDROUNDROBIN»). Значение по умолчанию: «0»

Параметр	Описание
«Профиль»	Активация параметра позволяет выбрать Профиль сохранения сессий. Все Реальные Серверы в группе должны иметь разные IP-адреса (это не касается портов) для успешного сохранения сессий. Параметр доступен при выборе протокола типа «TCP» или «HTTP»
«Постоянство выбора РС (persistence)»	Выбор типа привязки пользователя к одному Реальному Серверу на время активной сессии или выбор Профиля сохранения сессий (при активированном параметре «Профиль»). Возможные значения (при отключенном параметре «Профиль»): • «NONE» – привязка не используется; • «IPSOURCE» – привязка по IP-адресу источника запроса, т.е. IP-адресу пользователя; • «COOKIEINSERT» – привязка по cookie, который помещается в HTTP-ответ, направляемый пользователю. Значение доступно при выборе типа балансировки «HTTP»; • «HEADER» – привязка по значению заголовка, указанного в конфигурации. Значение доступно при выборе типа балансировки «HTTP»; • «COOKIE» – привязка по cookie, который получен в ответе Реального Сервера. Значение доступно при выборе типа балансировки «HTTP»; • «SSLSESSION» – привязка пользователя к Реальному Серверу по идентификатору SSL-сессии. Значение доступно при выборе типа балансировки «TCP» или «HTTP». Значение по умолчанию (при отключенном параметре «Профиль»): «NONE»
«Название HTTP заголовка»	Заголовок, по которому повторное подключение пользователя будет направлено на ранее выбранный Реальный Сервер. Параметр доступен при выборе «Постоянство выбора РС (persistence)» со значением «HEADER»
«Название cookie»	Имя cookie, который ожидается в ответе Реального Сервера для повторного подключения пользователя на ранее выбранный Реальный Сервер. Параметр доступен при выборе «Постоянство выбора РС (persistence)» со значением «COOKIE»
«Время жизни, с»	Период (в секундах), в течение которого сохраняется привязка пользователя к Реальному Серверу при значении параметра «Постоянство выбора РС (persistence)», отличного от «NONE». Таймер параметра сбрасывается к заданному значению при каждом новом действии пользователя. Значение по умолчанию: «60»

Параметр	Описание
«Сохранять SRC IP»	Параметр позволяет управлять подменой IP-адреса источника запроса при активированном параметре «Постоянство выбора РС (persistence)». Активация параметра сохраняет исходный IP-адрес источника запроса при подключении к Реальному Серверу. Отключение параметра подменяет IP-адрес источника запроса при подключении к Реальному Серверу. Параметр доступен при выборе протокола типа «RAPID-TCP» или «RAPID-UDP»
«Таймаут закрытия по FIN, с»	Период (в секундах) времени хранения записи о сессии после получения FIN-пакета. Параметр доступен при выборе протокола типа «RAPID-TCP». Значение по умолчанию: «2»
«Сохранять клиентский IP»	Параметр позволяет управлять подменой IP-адреса клиента. Активация параметра сохраняет исходный IP-адрес клиента при подключении к Реальному Серверу. Отключение параметра подменяет IP-адрес клиента при подключении к Реальному Серверу.
«Продолжительность неактивной сессии, с»	Период (в секундах), в течение которого соединение остается открытым при отсутствии активной сессии. Параметр доступен при выборе протокола типа «RAPID-TCP» или «RAPID-UDP». Значение по умолчанию: «60»
«Режим DSR»	Выбор режима работы Сервера Балансировки. Возможные значения: • «OFF» – режим работы, при котором DSR отключен; • «MAC» – режим с подменой MAC-адресов (L2 DSR). В этом режиме IP-адреса во входящем пакете остаются неизменными, Termidesk Connect подменяет в пакете только MAC-адреса (MAC-адрес источника – Termidesk Connect, MAC-адрес назначения – Реальный Сервер) и отправляет этот пакет по MAC-адресам на сервер; • «IPIP» – режим (L3 DSR), при котором входящий пакет инкапсулируется в IPIP-туннель и направляется на Реальный Сервер. Далее Реальный Сервер декапсулирует IP-пакет и видит IP-адрес пользователя и IP-адрес Виртуального Сервера. Параметр доступен при выборе протокола типа «RAPID-TCP» или «RAPID-UDP». Значение по умолчанию: «OFF»
«IP TTL к РС»	TTL IP-пакета к Реальному Серверу. Параметр доступен при выборе протокола типа «RAPID-TCP» или «RAPID-UDP». Значение по умолчанию: «0»

Параметр	Описание
«IP TTL к источнику запроса»	TTL IP-пакета к источнику запроса. Параметр доступен при выборе протокола типа «RAPID-TCP» или «RAPID-UDP». Значение по умолчанию: «0»
«Клиентский TLS Профиль»	Выбор Клиентского SSL-Профиля для аутентификации на Реальном Сервере. Параметр доступен при выборе типа балансировки «TCP» или «HTTP»
«Клиентский TCP Профиль»	Выбор Клиентского TCP-Профиля. Параметр доступен при выборе типа балансировки «TCP» или «HTTP». Значение по умолчанию: «tcpp-client-default»
«Клиентский HTTP Профиль»	Выбор Клиентского HTTP-Профиля. Параметр доступен при выборе типа балансировки «HTTP». Значение по умолчанию: «httpp-client-default»
«Профиль ограничения потока данных»	Выбор Профиля ограничения скорости. Параметр доступен при выборе типа балансировки «TCP» или «HTTP»
«Правила модификации ответов»	Список правил для балансировки подключений. Над списком правил можно выполнять следующие действия: • добавить, для этого нажать экранную кнопку **[Добавить]**; • удалить, для этого выбрать нужный Сценарий в списке и нажать экранную кнопку **[Удалить]**
«Перебалансировка»	Настройка функции перебалансировки. Перебалансировка (попытка выбрать другой Реальный Сервер) осуществляется в случае ошибки подключения к Реальному Серверу.  В случае выбора другого Реального Сервера в результате перебалансировки, если была запись в персистентной таблице, эта запись будет удалена (заменена) на другой Реальный Сервер. Активация параметра разрешает перебалансировку, отключение параметра запрещает перебалансировку. Параметр доступен при выборе типа балансировки «TCP» и «HTTP»
«Максимальное количество попыток»	Максимальное количество попыток перебалансировки. Возможные значения: от 1 до 10. Значение по умолчанию: «1». Параметр доступен при активации параметра «Перебалансировка»
«Комментарий»	Комментарий, который будет привязан к Серверу Балансировки

Доступные параметры для добавления Сценария перечислены в столбце «Параметр» следующей таблицы (см. таблицу Данные для добавления Сценария балансировки подключений по протоколу HTTP).

Таблица 143. Данные для добавления Сценария балансировки подключений по протоколу HTTP

Параметр	Описание
«Приоритет»	Приоритет обработки Сценария
«Сценарий»	Выбор исполняемого файла в раскрывающемся списке. Файл должен располагаться в каталоге `/var/lib/tdc/lbscripts/response-modifying/`

Веб. TLS

Просмотр и добавление сертификата через веб-интерфейс Termidesk Connect

Для доступа к веб-интерфейсу Termidesk Connect по протоколу HTTPS, а также для других подключений, требующих защищенного соединения, на этапе первичной настройки генерируется самоподписанный сертификат.

Для просмотра списка сертификатов и других файлов следует перейти «Настройки – Управление трафиком – TLS – Файлы». Файлы располагаются в каталоге `/etc/ssl/tdc/`.

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены таблице (см. Основные параметры списка TLS-файлов).

Таблица 144. Основные параметры списка TLS-файлов

Параметр	Описание
«Имя»	Имя файла
«Тип»	Тип файла. Возможные значения: • «SRVR_CERT» – сертификат; • «UNKNOWN» – другой файл
«Common Name»	Общее имя
«Sig Alg»	Алгоритм электронной подписи сертификата
«Действителен до»	Дата окончания действия сертификата Если срок действия сертификата истекает менее чем через неделю, текст выделяется оранжевым цветом.
«Статус»	Статус сертификата. Возможные значения по цветовым маркерам: • зеленый – действительный сертификат; • красный – недействительный сертификат; • серый – другой файл (не имеет статус)

Для загрузки сертификата или другого файла нажать экранную кнопку **[Загрузить]** и выбрать соответствующий файл:

В списке приведены основные типы файлов. Также возможна загрузка дополнительных файлов.

- файл сертификата;
- файл закрытого ключа;
- файл сертификатов УЦ;
- CRL-файл;
- файл с параметрами Диффи-Хеллмана.

Для удаления сертификата или другого файла выбрать соответствующий файл в списке и нажать экранную кнопку **[Удалить]**.

SSL-Профили

Для отображения списка Профилей следует перейти «Настройки – Управление трафиком – TLS – Профили».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены таблице (см. Основные параметры списка SSL-Профилей).

Таблица 145. Основные параметры списка SSL-Профилей

Параметр	Описание
«Имя»	Наименование SSL-Профиля
«Тип»	Тип SSL-Профиля

Для добавления SSL-Профиля следует перейти «Настройки – Управление трафиком – TLS – Профили» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления SSL-Профиля).

Таблица 146. Данные для добавления SSL-Профиля

Параметр	Описание
«Имя»	Наименование SSL-Профиля
«Таймаут»	Время ожидания (в секундах) установки соединения. Значение по умолчанию: «5»
«Тип»	Выбор типа SSL-Профиля. Возможные значения: • «Клиентский»; • «Серверный»
«Конфигурация по-умолчанию»	Форма настройки защищенного соединения для SSL-Профиля типа «Серверный». Для настройки защищенного соединения нужно раскрыть форму «Конфигурация по-умолчанию» и нажать экранную кнопку **[Создать]**. Конфигурация используется, когда пришедший SNI не соответствует ни одному заданному значению. Статус «Активна» свидетельствует об использовании заданных параметров защищенного соединения

Параметр	Описание
«Файл сертификата»	Выбор файла клиентского сертификата для аутентификации Termidesk Connect. При активированном параметре «PKI» для клиентского сертификата указывается путь Хранилища секретов. Пример значения параметра: «pki://pki/issue/example-dot-com?common_name=\"app.example.com\"&ttl=\"24h\"#certificate»
«PKI»	Активация возможности получения файла из Хранилища секретов для клиентского сертификата
«Файл ключа»	Выбор файла закрытого ключа, соответствующего клиентскому сертификату. При активированном параметре «PKI» для закрытого ключа указывается путь Хранилища секретов. Пример значения параметра: «pki://pki/issue/example-dot-com?common_name=\"app.example.com\"&ttl=\"24h\"#private_key»
«PKI»	Активация возможности получения файла из Хранилища секретов для закрытого ключа
«Пароль»	Пароль для расшифровки файла закрытого ключа, если ключ зашифрован. Так же пароль может быть получен из Хранилища секретов. Пример значения параметра: • «kv://secret/cert#password» – для kv версии 1; • «kv://secret/data/snmp#password» – для kv версии 2
«SNI по-умолчанию»	Наименование сервера по умолчанию для расширения SNI. Параметр доступен для SSL-Профиля типа «Клиентский»
«Файл сертификата УЦ»	Выбор файла сертификата УЦ для проверки подлинности сертификата Реального Сервера. При активированном параметре «PKI» для сертификата УЦ указывается путь Хранилища секретов. Пример значения параметра: «pki://pki/issue/example-dot-com?common_name=\"app.example.com\"&ttl=\"24h\"#issuing_ca» Параметр доступен для SSL-Профиля типа «Клиентский»
«PKI»	Активация возможности получения файла из Хранилища секретов для сертификата УЦ. Параметр доступен для SSL-Профиля типа «Клиентский»
«Файлы сертификата УЦ»	Выбор файлов сертификата УЦ для проверки подлинности клиентского сертификата пользователя.  Может быть выбрано несколько файлов последовательным нажатием на файлы в раскрывающемся списке. При активированном параметре «PKI» для сертификатов УЦ указывается путь Хранилища секретов. Пример значения параметра: «pki://pki/issue/example-dot-com?common_name=\"app.example.com\"&ttl=\"24h\"#issuing_ca» Параметр доступен для SSL-Профиля типа «Серверный»

Параметр	Описание
«PKI»	Активация возможности получения файлов из Хранилища секретов для сертификатов УЦ. Параметр доступен для SSL-Профиля типа «Серверный»
«Файл параметров Diffie-Hellman KE»	Выбор файла с параметрами Диффи-Хеллмана для обеспечения безопасного соединения
«Stapling»	Активация поддержки OCSP Stapling с фоновым обновлением подписи серверного сертификата через OCSP Responder. Для OCSP Stapling не задается URL-адрес, т.к он извлекается из серверного сертификата. В файле серверного сертификата необходимо указывать всю цепочку сертификатов, ведущую к УЦ, включая все промежуточные сертификаты. Параметр доступен при настройке конфигурации по умолчанию для SSL-Профиля типа «Серверный»
«mTLS»	Активация использования протокола mTLS. Параметр доступен при настройке конфигурации по умолчанию для SSL-Профиля типа «Серверный»
«Режим проверки mTLS»	Выбор метода проверки отзыва для клиентских сертификатов. Возможные значения: • «NONE»; • «CRL»; • «OCSP». Параметр доступен при активации параметра «mTLS» в конфигурации по умолчанию для SSL-Профиля типа «Серверный»
«Файл CRL»	Выбор CRL-файла для проверки клиентских сертификатов. При активированном параметре «PKI» для CRL-файла указывается путь Хранилища секретов. Пример значения параметра: «pki://v1/pki/cert/crl#certificate» Параметр доступен при выборе метода проверки mTLS «CRL» в конфигурации по умолчанию для SSL-Профиля типа «Серверный»
«PKI»	Активация возможности получения файла из Хранилища секретов для CRL-файла. Параметр доступен при выборе метода проверки mTLS «CRL» в конфигурации по умолчанию для SSL-Профиля типа «Серверный»
«URL»	URL-адрес сервера (OCSP Responder). В текущей версии Termidesk Connect для сервера (OCSP Responder) поддерживается только протокол HTTP. Параметр доступен при выборе метода проверки mTLS «OCSP» в конфигурации по умолчанию для SSL-Профиля типа «Серверный»

Параметр	Описание
«HTTP-метод»	Метод HTTP-запроса. В текущей версии Termidesk Connect поддерживается только метод «POST». Параметр доступен при выборе метода проверки mTLS «OCSP» в конфигурации по умолчанию для SSL-Профиля типа «Серверный»
«Таймаут, с»	Время ожидания (в секундах) ответа от сервера (OCSP Responder). Значение по умолчанию: «10». Параметр доступен при выборе метода проверки mTLS «OCSP» в конфигурации по умолчанию для SSL-Профиля типа «Серверный»
«Cache»	Активация кеширования OCSP-ответов. Параметр доступен при выборе метода проверки mTLS «OCSP» в конфигурации по умолчанию для SSL-Профиля типа «Серверный»
«Nonce»	Активация поддержки расширения Nonce. Параметр доступен при выборе метода проверки mTLS «OCSP» в конфигурации по умолчанию для SSL-Профиля типа «Серверный»
«Strict»	Активация строгого режима проверки. При активации параметра разрешается допуск только клиентов, чьи сертификаты разрешены. При отключении разрешается допуск клиентов, чьи сертификаты разрешены или чьих сертификатов нет в БД. Параметр доступен при выборе метода проверки mTLS «OCSP» в конфигурации по умолчанию для SSL-Профиля типа «Серверный»
«Secure Renegotiation»	Активация механизма Secure Renegotiation (применяется для протокола TLS версии 1.2 или старше)
«Лимит запросов в минуту на SSL-соединение»	Количество допустимых запросов в минуту на повторное согласование в рамках одного SSL-подключения. При превышении заданного лимита для всех последующих попыток будет возвращено предупреждение о невозможности согласования на уровне протокола, пока период не обновится. При значении «0» ограничение на количество допустимых запросов отсутствует. Значение по умолчанию: «10». Параметр доступен в конфигурации по умолчанию для SSL-Профиля типа «Серверный»
«Тип Session Reuse»	Тип Session Reuse (применяется для протокола TLS версии 1.2 или старше). Значение по умолчанию: «STATEFUL». Параметр доступен в конфигурации по умолчанию для SSL-Профиля типа «Серверный»

Параметр	Описание
«Время хранения сессии в кеше, с»	Время хранения (в секундах) SSL-сессии в кеше. Значение по умолчанию: «7200». Параметр доступен при выборе типа Session Reuse «STATEFUL» в конфигурации по умолчанию для SSL-Профиля типа «Серверный»
«Поддерживаемые версии TLS (SSL)»	Выбор поддерживаемых версий протокола TLS для установления соединения. Для использования защищенного соединения нужно отметить галочкой соответствующую версию протокола в списке. Для выбора всех версий протокола следует нажать экранную кнопку **[Выбрать всё]**. Для очистки списка нужно нажать экранную кнопку **[Очистить]**
«Наборы криптографических алгоритмов»	Выбор поддерживаемых алгоритмов преобразования данных. Для использования преобразования данных нужно отметить галочкой соответствующий алгоритм в списке. Для выбора всех алгоритмов следует нажать экранную кнопку **[Выбрать всё]**. Для очистки списка нужно нажать экранную кнопку **[Очистить]**
«Серверы»	Список SNI. Параметр доступен для SSL-Профиля типа «Серверный» Над списком можно выполнять следующие действия: • добавить, для этого нажать экранную кнопку **[Добавить]**; • изменить, для этого выбрать нужное правило в списке и нажать экранную кнопку **[Изменить]**; • удалить, для этого выбрать нужные правила в списке и нажать экранную кнопку **[Удалить]**
«Комментарий»	Комментарий, который будет привязан к SSL-Профилю

Доступные параметры для добавления SNI перечислены в столбце «Параметр» следующей таблицы (см. Данные для добавления SNI).

Таблица 147. Данные для добавления SNI

Параметр	Описание
«SNI»	Имя (hostname) узла, полученное во время установки соединения. Поддерживается задание шаблонов SNI по формату: `*.<домен>`. При этом: • астериск может быть расположен только слева и должен быть разделен от домена точкой; • астериск означает, что любой хост домена (не включая оригинальный домен) удовлетворяет шаблону. Для профиля оригинального домена требуется отдельный хост SNI; • при выборе SNI приоритетным будет тот, у которого совпадет больше уровней доменов или имеется полное совпадение по шаблону

Параметр	Описание
«Файл сертификата»	Выбор файла серверного сертификата, который будет передан для заданного SNI. При активированном параметре «PKI» для серверного сертификата указывается путь Хранилища секретов. Пример значения параметра: «pki://pki/issue/example-dot-com?common_name=\"app.example.com\"&ttl=\"24h\"#certificate»
«PKI»	Активация возможности получения файла из Хранилища секретов для серверного сертификата
«Stapling»	Активация поддержки OCSP Stapling с фоновым обновлением подписи серверного сертификата через OCSP Responder
«Файл ключа»	Выбор файла закрытого ключа, соответствующего серверному сертификату. При активированном параметре «PKI» для закрытого ключа указывается путь Хранилища секретов. Пример значения параметра: «pki://pki/issue/example-dot-com?common_name=\"app.example.com\"&ttl=\"24h\"#private_key»
«PKI»	Активация возможности получения файла из Хранилища секретов для закрытого ключа
«Файлы сертификата УЦ»	Выбор файлов сертификата УЦ, которые используются для проверки подлинности клиентского сертификата. Может быть выбрано несколько файлов последовательным нажатием на файлы в раскрывающемся списке. При активированном параметре «PKI» для сертификатов УЦ указывается путь Хранилища секретов. Пример значения параметра: «pki://pki/issue/example-dot-com?common_name=\"app.example.com\"&ttl=\"24h\"#issuing_ca»
«PKI»	Активация возможности получения файлов из Хранилища секретов для сертификатов УЦ
«mTLS»	Активация использования протокола mTLS
«Режим проверки mTLS»	Выбор метода проверки отзыва для клиентских сертификатов. Возможные значения: • «NONE»; • «CRL»; • «OCSP». Параметр доступен при активации параметра «mTLS»
«Файл CRL»	Выбор CRL-файла для проверки клиентских сертификатов. При активированном параметре «PKI» для CRL-файла указывается путь Хранилища секретов. Пример значения параметра: «pki://pki/issue/example-dot-com?common_name=\"app.example.com\"&ttl=\"24h\"#issuing_ca». Параметр доступен при выборе метода проверки mTLS «CRL»

Параметр	Описание
«PKI»	Активация возможности получения файла из Хранилища секретов для сертификатов УЦ. Параметр доступен при выборе метода проверки mTLS «CRL»
«URL»	URL-адрес сервера (OCSP Responder) В текущей версии Termidesk Connect для сервера (OCSP Responder) поддерживается только протокол HTTP. Параметр доступен при выборе метода проверки mTLS «OCSP»
«HTTP-метод»	Метод HTTP-запроса. В текущей версии Termidesk Connect поддерживается только метод «POST». Параметр доступен при выборе метода проверки mTLS «OCSP»
«Таймаут, с»	Время ожидания (в секундах) ответа от сервера (OCSP Responder). Значение по умолчанию: «10». Параметр доступен при выборе метода проверки mTLS «OCSP»
«Cache»	Активация кеширования OCSP-ответов. Параметр доступен при выборе метода проверки mTLS «OCSP»
«Nonce»	Активация поддержки расширения Nonce. Параметр доступен при выборе метода проверки mTLS «OCSP»
«Strict»	Активация строгого режима проверки. При активации параметра разрешается допуск только клиентов, чьи сертификаты разрешены. При отключении разрешается допуск клиентов, чьи сертификаты разрешены или чьих сертификатов нет в БД. Параметр доступен при выборе метода проверки mTLS «OCSP»
«Файл параметров Diffie-Hellman KE»	Выбор файла с параметрами Диффи-Хеллмана для обеспечения безопасного соединения
«Пароль»	Пароль для расшифровки файла закрытого ключа, если ключ зашифрован. Так же пароль может быть получен из Хранилища секретов. Пример значения параметра: • «kv://secret/cert#password» – для kv версии 1; • «kv://secret/data/snmp#password» – для kv версии 2
«Secure Renegotiation»	Активация механизма Secure Renegotiation (применяется для протокола TLS версии 1.2 или старше)

Параметр	Описание
«Лимит запросов в минуту на SSL-соединение»	Количество допустимых запросов в минуту на повторное согласование в рамках одного SSL-подключения. При превышении заданного лимита для всех последующих попыток будет возвращено предупреждение о невозможности согласования на уровне протокола, пока период не обновится. При значении «0» ограничение на количество допустимых запросов отсутствует. Значение по умолчанию: «10»
«Тип Session Reuse»	Тип Session Reuse (применяется для протокола TLS версии 1.2 или старше). Значение по умолчанию: «STATEFUL»
«Время хранения сессии в кеше, с»	Время хранения (в секундах) SSL-сессии в кеше. Значение по умолчанию: «7200»
«Поддерживаемые версии TLS (SSL)»	Выбор поддерживаемых версий протокола TLS для установления соединения. Для использования защищенного соединения нужно отметить галочкой соответствующую версию протокола в списке. Для выбора всех версий протокола следует нажать экранную кнопку **[Выбрать всё]**. Для очистки списка нужно нажать экранную кнопку **[Очистить]**
«Наборы криптографических алгоритмов»	Выбор поддерживаемых алгоритмов преобразования данных. Для использования преобразования данных нужно отметить галочкой соответствующий алгоритм в списке. Для выбора всех алгоритмов следует нажать экранную кнопку **[Выбрать всё]**. Для очистки списка нужно нажать экранную кнопку **[Очистить]**

SSL-Политики

Для отображения списка Политик следует перейти «Настройки – Управление трафиком – TLS – Политики».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены таблице (см. Основные параметры списка SSL-Политик).

Таблица 148. Основные параметры списка SSL-Политик

Параметр	Описание
«Имя»	Наименование SSL-Политики
«Количество условий»	Количество условий, привязанных к SSL-Политике
«Используется»	Список Серверов Балансировки и SSL-Профилей, указанных в условиях SSL-Политики

Для добавления SSL-Политики следует перейти «Настройки – Управление трафиком – TLS – Политики» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления SSL-Политики).

Таблица 149. Данные для добавления SSL-Политики

Параметр	Описание
«Имя»	Наименование SSL-Политики
«Условия»	Список условий для SSL-Политики. Над списком можно выполнять следующие действия: • добавить, для этого нажать экранную кнопку **[Добавить]**; • изменить, для этого выбрать нужное условие в списке и нажать экранную кнопку **[Изменить]**; • удалить, для этого выбрать нужные условие в списке и нажать экранную кнопку **[Удалить]**

Доступные параметры для добавления условия перечислены в столбце «Параметр» следующей таблицы (см. Данные для добавления условия).

Таблица 150. Данные для добавления условия

Параметр	Описание
«Номер»	Порядковый номер обработки условия
«Тип»	Тип условия. Возможные значения: • «SNI» – проверка поля SNI из TLS Hello; • «CIPHER» – проверка списка алгоритмов шифрования из TLS Hello. Проверяется последовательность в любом месте на соответствие списку; • «ALPN» – проверка списка ALPN из TLS Hello; • «DEFAULT» – действие при невыполнении условия
«Условие»	Условие для выполнения действия. Возможные значения для «SNI»: • «EQ» – точное соответствие «SNI» и значения; • «NOTEQ» – точное несоответствие «SNI» и значения; • «CONTAINS» – содержит последовательность в значении; • «NOTCONTAINS» – не содержит последовательность в значении; • «STARTSWITH» – начинается с значения; • «ENDWITH» – заканчивается значением. Возможные значения для «CIPHER»: • «CONTAINS» – содержит последовательность в значении; • «NOTCONTAINS» – не содержит последовательность в значении. Возможные значения для «ALPN»: • «EQ» – точное соответствие «ALPN» и значения; • «NOTEQ» – точное несоответствие «ALPN» и значения

Параметр	Описание
«Действие»	Действие при успешном выполнении условия. Возможные значения: • «LBS» – выбор Сервера Балансировки; • «SSL» – выбор SSL-Профиля; • «DROP» – разрыв соединения

Веб. Сценарии

Для загрузки файла следует перейти «Настройки – Управление трафиком – Сценарии» и выбрать нужную вкладку:

- «Сценарии Балансировки»;
- «Сценарии модификации ответов»;
- «Сценарии ошибок»;
- «Сценарии SSL».

Затем нажать экранную кнопку:

- **[Загрузить]** для загрузки файла в Termidesk Connect. Допускается загрузка только файлов в формате `.lua`;
- **[Добавить]** для создания файла Сценария и его заполнения в интерактивном режиме. Для создания файла потребуется заполнить параметры (см. Параметры для создания файла Сценария работы с соединением) и нажать экранную кнопку **[Применить]**. Экранная кнопка **[Закрыть]** закроет интерактивный режим без сохранения изменений.

После загрузки или создания файлов они будут расположены:

- Сценарии балансировки в каталоге `/var/lib/tdc/lbscripts/content-switching/`;
- Сценарии модификации ответов в каталоге `/var/lib/tdc/lbscripts/response-modifying/`;
- Сценарии ошибок в каталоге `/var/lib/tdc/lbscripts/error-reply/`;
- Сценарии SSL в каталоге `/var/lib/tdc/lbscripts/ssl`.

Для удаления файла нужно:

- отметить его и нажать экранную кнопку **[Удалить]**;
- подтвердить удаление, нажав экранную кнопку **[Удалить]** на запрос об удалении.

Для изменения файла нужно:

- либо отметить его и нажать экранную кнопку **[Изменить]**;
- либо нажать левой кнопкой мыши на его имя.

Таблица 151. Параметры для создания файла Сценария работы с соединением

Параметр	Описание
«Имя»	Наименование файла на латинице с указанием формата `.lua`  При задании наименования файла, уже имеющегося в Termidesk Connect, исходный файл будет перезаписан.
«Содержимое файла»	Содержимое файла. В содержимом можно определять условия выполнения того или иного действия логическими выражениями (см. подраздел **Сценарии**). Редактор поддерживает функцию автоматического дополнения выражений по мере их ввода.  Автоматическая проверка синтаксиса при работе в интерактивном режиме не производится. Примеры Сценариев приведены в подразделе **Сценарии**

Веб. Виртуальные Серверы

Для отображения списка Виртуальных Серверов следует перейти «Настройки – Управление трафиком – Виртуальные серверы».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены в таблице (см. таблицу Основные параметры списка Виртуальных Серверов).

Таблица 152. Основные параметры списка Виртуальных Серверов

Параметр	Описание
«Имя»	Наименование Виртуального Сервера
«IP Адрес»	IP-адрес Виртуального Сервера
«Порт»	Порт Виртуального Сервера
«Тип»	Тип протокола для балансировки подключения
«Статус»	Статус Виртуального Сервера
«VRF»	Имя VRF
«TLS Профиль»	Используемый Серверный SSL-Профиль

Над Виртуальными Серверами можно выполнять следующие действия:

- включить, для этого выбрать нужные серверы в списке и нажать экранную кнопку **[Включить]**;
- выключить, для этого выбрать нужные серверы в списке и нажать экранную кнопку **[Выключить]**.

Для добавления Виртуального Сервера следует перейти «Настройки – Управление трафиком – Виртуальные серверы» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. таблицу Данные для добавления Виртуального Сервера).

Таблица 153. Данные для добавления Виртуального Сервера

Параметр	Описание
«Имя»	Наименование Виртуального Сервера

Параметр	Описание
«Тип»	Выбор типа протокола для балансировки подключения. Возможные значения: • «TCP»; • «HTTP»; • «RAPID-TCP»; • «RAPID-UDP»
«IP Адрес»	IP-адрес Виртуального Сервера
«Диапазон»	Диапазон портов Виртуального Сервера
«Порт начало»	Начальное значение диапазона портов Виртуального Сервера
«Порт конец»	Конечное значение диапазона портов Виртуального Сервера
«Порт»	Порт Виртуального Сервера. Для добавления порта: • нажать экранную кнопку **[Добавить]**; • ввести порт и нажать экранную кнопку **[+]**. Поддерживается задание списка портов. Для удаления порта из списка нажать экранную кнопку **[-]**
«Любой»	Любое значение порта Виртуального Сервера.При переключении станут недоступны параметры «Диапазон» и «Порт». Параметр доступен при выборе «Тип» со значением «RAPID-TCP» или «RAPID-UDP»
«Зависимость состояния»	Параметр позволяет задать правило изменения статуса Виртуального Сервера в зависимости от состояния Серверов Балансировки в списке. Для добавления Сервера Балансировки нужно нажать экранную кнопку **[Добавить]** и выбрать нужный сервер в списке. Возможные значения: • «NONE» – Виртуальный Сервер всегда находится в статусе «В работе»; • «OR» – Виртуальный Сервер находится в статусе «В работе», если хотя бы один Сервер Балансировки доступен; • «AND» – Виртуальный Сервер переходит в статус «Отключен», если хотя бы один Сервер Балансировки становится недоступным
«VRF»	Выбор VRF для добавления Виртуального Сервера
«Серверный TLS Профиль»	Выбор Серверного SSL-Профиля
«Серверный TCP Профиль»	Выбор Серверного TCP-Профиля. Параметр доступен при выборе типа Виртуального Сервера «TCP» или «HTTP»
«Серверный HTTP Профиль»	Выбор Серверного HTTP-Профиля. Параметр доступен при выборе типа Виртуального Сервера «HTTP»

Параметр	Описание
«Профиль ограничения потока данных»	Выбор Профиля ограничения скорости. Параметр доступен при выборе типа Виртуального Сервера «TCP» или «HTTP»
«TLS Проверка»	Выбор SSL-Политики. Параметр доступен при выборе типа Виртуального Сервера «TCP»
«Правила»	Список правил для балансировки подключений. Над списком правил можно выполнять следующие действия: • добавить, для этого нажать экранную кнопку **[Добавить]**; • удалить, для этого выбрать нужные правила в списке и нажать экранную кнопку **[Удалить]**. Параметр доступен при выборе типа Виртуального Сервера «HTTP»
«TLS Правила»	Список правил для выбора SSL-Политики или Сервера Балансировки на основании данных, полученных из сообщения TLS Hello. Над списком правил можно выполнять следующие действия: • добавить, для этого нажать экранную кнопку **[Добавить]**; • удалить, для этого выбрать нужные правила в списке и нажать экранную кнопку **[Удалить]**. Параметр доступен при выборе типа Виртуального Сервера «HTTP»
«RHI»	Параметр позволяет управлять возможностью анонсирования IP-адреса, привязанного к Виртуальному Серверу, протоколам динамической маршрутизации. Для работы RHI необходимо, чтобы динамическая маршрутизация была включена на Termidesk Connect (см. настройку динамической маршрутизации в подразделе **Сеть**). При активации параметра Termidesk Connect будет анонсировать в сеть IP-адреса Виртуальных Серверов в зависимости от их режима работы, определенного в параметре «Состояние»
«Состояние»	Параметр позволяет управлять режимом работы RHI для Виртуального Сервера. Возможные значения: • «ACTIVE» – активный режим RHI для Виртуального Сервера; • «PASSIVE» – пассивный режим RHI для Виртуального Сервера. Состояние маршрута определяется условиями: • если все Виртуальные Сервера по данному маршруту находятся в режиме «PASSIVE», то Termidesk Connect всегда будет объявлять маршрут для виртуального IP-адреса; • если хотя бы один Виртуальный Сервер находятся в режиме «ACTIVE» и в состоянии «В работе», то Termidesk Connect будет объявлять маршрут для виртуального IP-адреса; • в остальных случаях Termidesk Connect не будет объявлять маршрут
«Комментарий»	Комментарий, который будет привязан к Виртуальному Серверу

Доступные правила балансировки подключений по протоколам TCP, RAPID-TCP, RAPID-UDP перечислены в столбце «Параметр» следующей таблицы (см. таблицу Данные для добавления правил балансировки подключений по протоколам TCP, RAPID-TCP, RAPID-UDP).

Таблица 154. Данные для добавления правил балансировки подключений по протоколам TCP, RAPID-TCP, RAPID-UDP

Параметр	Описание
«Приоритет»	Порядковый номер обработки правила
«Сервер балансировки»	Выбор Сервера Балансировки
«Сеть источника запроса»	IP-адрес сети в формате CIDR

Доступные параметры для добавления правил перечислены в столбце «Параметр» следующей таблицы (см. таблицу Данные для добавления правил по протоколу HTTP).

Таблица 155. Данные для добавления правил по протоколу HTTP

Параметр	Описание
«Приоритет»	Порядковый номер обработки правила
«Сценарий»	Выбор исполняемого файла в раскрывающемся списке. Файл Сценария балансировки должен располагаться в каталоге `/var/lib/tdc/lbscripts/content-switching/`. Файл Сценария SSL должен располагаться в каталоге `/var/lib/tdc/lbscripts/ssl`

Функция «ГеоБН»

Веб. ADNS

Для отображения списка ADNS следует перейти «Настройки – ГеоБН – ADNS».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «IP Адрес».

Основные параметры списка приведены в таблице (см. Основные параметры списка ADNS).

Таблица 156. Основные параметры списка ADNS

Параметр	Описание
«IP Адрес»	IP-адрес Termidesk Connect для обработки DNS-запросов
«Порт»	Порт для обработки DNS-запросов

Для добавления ADNS следует перейти «Настройки – ГеоБН – ADNS» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления ADNS).

Таблица 157. Данные для добавления ADNS

Параметр	Описание
«IP Адрес»	IP-адрес Termidesk Connect для обработки DNS-запросов  В случае использования отказоустойчивой конфигурации Termidesk Connect необходимо указывать общий IP-адрес кластера.

Параметр	Описание
«Порт»	Порт для обработки DNS-запросов

Веб. Площадки

Для отображения списка Площадок следует перейти «Настройки – ГеоБН – Площадки».

В текущей версии Termidesk Connect функционал не влияет на работу Termidesk Connect. Приведена справочная информация.

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены в таблице (см. Основные параметры списка Площадок).

Таблица 158. Основные параметры списка Площадок

Параметр	Описание
«Имя»	Наименование Площадки

Для добавления Площадки следует перейти «Настройки – ГеоБН – Площадки» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления Площадки).

Таблица 159. Данные для добавления Площадки

Параметр	Описание
«Имя»	Наименование группы серверов или ЦОД георасределенной балансировки

Веб. DNS View

Для отображения списка DNS View следует перейти «Настройки – ГеоБН – DNS View».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены в таблице (см. Основные параметры списка DNS View).

Таблица 160. Основные параметры списка DNS View

Параметр	Описание
«Имя»	Наименование DNS View

Для добавления DNS View следует перейти «Настройки – ГеоБН – DNS View» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления DNS View).

Таблица 161. Данные для добавления DNS View

Параметр	Описание
«Имя»	Наименование DNS View

Параметр	Описание
«Сети»	Список сетей, при запросе из которых Termidesk Connect будет выдавать локальные IP-адреса Сервисов. Над списком сетей можно выполнять следующие действия: • добавить, для этого нажать экранную кнопку **[Добавить]** и указать IP-адрес сети в формате CIDR в открывшемся поле; • удалить, для этого выбрать нужные сети в списке и нажать экранную кнопку **[Удалить]**

Веб. Сервисы

Для отображения списка сервисов следует перейти «Настройки – ГеоБН – Сервисы».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены в таблице (см. Основные параметры списка Сервисов).

Таблица 162. Основные параметры списка Сервисов

Параметр	Описание
«Имя»	Наименование Сервиса
«Состояние»	Статус Сервиса
«Публичный IP»	IP-адрес Сервиса, используемый для доступа из внешних сетей
«Внутренний IP»	IP-адрес Сервиса, используемый для доступа внутри локальной сети

Над Сервисами можно выполнять следующие действия:

- включить, для этого выбрать нужные Сервисы в списке и нажать экранную кнопку **[Включить]**;
- выключить, для этого выбрать нужные Сервисы в списке и нажать экранную кнопку **[Выключить]**.

Для добавления Сервиса следует перейти «Настройки – ГеоБН – Сервисы» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления Сервиса).

Таблица 163. Данные для добавления Сервиса

Параметр	Описание
«Имя»	Наименование Сервиса
«Вес»	Определяет приоритет распределения запросов на Сервис. Распределение запросов на Сервис выполняется при активации алгоритма «ROUNDROBIN» на Виртуальном Сервере геобалансировки (см. подраздел **Виртуальные Серверы геобалансировки**)
«Публичный IP»	IP-адрес Сервиса, используемый для доступа из внешних сетей
«Внутренний IP»	IP-адрес Сервиса, используемый для доступа внутри локальной сети

Параметр	Описание
«Проверка»	Выбор типа Проверки доступности Сервиса
«Площадка»	В текущей версии Termidesk Connect параметр представлен в демонстрационном режиме.

Веб. Виртуальные Серверы геобалансировки

Для отображения списка Виртуальных Серверов геобалансировки следует перейти «Настройки – ГеоБН – Виртуальные серверы».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены в таблице (см. Основные параметры списка Виртуальных Серверов геобалансировки).

Таблица 164. Основные параметры списка Виртуальных Серверов геобалансировки

Параметр	Описание
«Имя»	Наименование Виртуального Сервера геобалансировки
«Состояние DNS View»	В текущей версии Termidesk Connect состояние отображается в демонстрационном режиме. Статус DNS View
«Алгоритм»	Используемый алгоритм балансировки

Над Виртуальными Серверами геобалансировки можно выполнять следующие действия:

- включить, для этого выбрать нужные серверы в списке и нажать экранную кнопку **[Включить]**;
- выключить, для этого выбрать нужные серверы в списке и нажать экранную кнопку **[Выключить]**.

Для добавления Виртуального Сервера геобалансировки следует перейти «Настройки – ГеоБН – Виртуальные серверы» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления Виртуального Сервера геобалансировки).

Таблица 165. Данные для добавления Виртуального Сервера геобалансировки

Параметр	Описание
«Имя»	Наименование Виртуального Сервера геобалансировки
«ECS»	Активация перенаправления запросов на ближайший Сервис на основе информации о подсети источника запроса

Параметр	Описание
«Алгоритм»	Выбор метода балансировки подключений. Возможные значения: • «ONLINE» – перенаправление запросов только на доступные Сервисы в списке балансировки; • «ROUNDROBIN» – поочередное распределение запросов между Сервисами в списке балансировки; • «SOURCEIPHASH» – привязка запросов к Сервисам на основе хеша IP-адреса; • «STATICPROXIMITY» – перенаправление запросов на ближайшую Площадку для выдачи Сервиса
«Постоянство выбора сервиса (persistence)»	Выбор типа привязки источника запроса к одному Сервису на время обработки его DNS-запросов. Возможные значения: • «OFF» – привязка к Сервису отключена; • «SRCIP» – привязка источника запроса к Сервису на основе его IP-адреса
«DNS View»	Выбор DNS View
«Сервисы»	Список Сервисов для балансировки подключений. Над списком Сервисов можно выполнять следующие действия: • добавить, для этого нажать экранную кнопку **[Добавить]**; • удалить, для этого выбрать нужные правила в списке и нажать экранную кнопку **[Удалить]**

По умолчанию список Сервисов представлен в табличном виде и упорядочен согласно столбцу «Имя».

Основные параметры списка приведены в таблице (см. Основные параметры списка Сервисов).

Таблица 166. Основные параметры списка Сервисов

Параметр	Описание
«Имя»	Наименование Сервиса
«Состояние»	В текущей версии Termidesk Connect состояние сервиса отображается в демонстрационном режиме. Статус Сервиса
«Вес»	Приоритет распределения запросов на Сервис

Веб. Зоны

Для отображения списка Зон следует перейти «Настройки – ГеоБН – Зоны».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены в таблице (см. Основные параметры списка Зон).

Таблица 167. Основные параметры списка Зон

Параметр	Описание
«Имя»	Наименование Зоны

Для добавления Зоны следует перейти «Настройки – ГеоБН – Зоны» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления Зоны).

Таблица 168. Данные для добавления Зоны

Параметр	Описание
«Имя»	Наименование Зоны
«Поддомены»	Список поддоменов, для которых задаются параметры обработки запросов. Над списком поддоменов можно выполнять следующие действия: • добавить, для этого нажать экранную кнопку **[Добавить]**; • изменить, для этого выбрать нужные поддомены в списке и нажать экранную кнопку **[Изменить]**; • удалить, для этого выбрать нужные поддомены в списке и нажать экранную кнопку **[Удалить]**

По умолчанию список поддоменов представлен табличном виде и упорядочен согласно столбцу «Имена».

Основные параметры списка приведены в таблице (см. Основные параметры списка поддоменов).

Таблица 169. Основные параметры списка поддоменов

Параметр	Описание
«Имена»	Наименование поддомена
«TTL»	Время жизни DNS-записи (в секундах)
«Виртуальные серверы»	Список назначенных Виртуальных Серверов для обработки запросов

Данные, необходимые для добавления поддомена, перечислены в столбце «Параметр» следующей таблицы (см. Данные для добавления поддомена).

Таблица 170. Данные для добавления поддомена

Параметр	Описание
«Поддомен»	Наименование поддомена
«TTL»	Время жизни DNS-записи (в секундах)
«Виртуальные серверы»	Список назначенных Виртуальных Серверов для обработки запросов. Над списком Виртуальных Серверов можно выполнять следующие действия: • добавить, для этого нажать экранную кнопку **[Добавить]**; • удалить, для этого выбрать нужные серверы в списке и нажать экранную кнопку **[Удалить]**

Данные, необходимые для добавления списка виртуальных серверов перечислены в столбце «Параметр» следующей таблицы (см. Данные для добавления списка Виртуальных Серверов).

Таблица 171. Данные для добавления списка Виртуальных Серверов

Параметр	Описание
«Порядковый номер»	Приоритет обработки запросов Виртуальным Сервером. Обработка запросов осуществляется в порядке возрастания порядкового номера Виртуального Сервера. В случае недоступности Виртуального Сервера, запросы будут обработаны следующим по порядку Виртуальным Сервером
«Виртуальный сервер»	Выбор Виртуального Сервера в списке

Функция «Безопасность»

Веб. Списки контроля доступа

Для отображения списка контроля доступа следует перейти «Настройки – Безопасность – Списки Контроля Доступа».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены в таблице (см. Основные параметры списка контроля доступа).

Таблица 172. Основные параметры списка контроля доступа

Параметр	Описание
«Имя»	Наименование списка контроля доступа
«Используется»	Статус использования списка контроля доступа
«Количество правил»	Количество правил в списке контроля доступа

Для добавления списка контроля доступа следует перейти «Настройки – Безопасность – Списки Контроля Доступа» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления списка контроля доступа).

Таблица 173. Данные для добавления списка контроля доступа

Параметр	Описание
«Имя»	Наименование списка контроля доступа
«Журналировать»	Использование журналирования событий
«Правила»	Список правил контроля доступа. Над списком правил можно выполнять следующие действия: • добавить, для этого нажать экранную кнопку **[Добавить]**; • изменить, для этого выбрать нужное правило в списке и нажать экранную кнопку **[Изменить]**; • удалить, для этого выбрать нужные правила в списке и нажать экранную кнопку **[Удалить]**

Форма «Добавление списка контроля доступа» позволяет задать список правил контроля доступа.

Доступные параметры для добавления правила перечислены в столбце «Параметр» следующей таблицы (см. Данные для добавления правила).

Таблица 174. Данные для добавления правила

Параметр	Описание
«Номер»	Приоритет правила в списке контроля доступа
«Протокол»	Протокол, для которого создается правило. Возможные значения: • «ICMP»; • «IP»; • «TCP»; • «UDP»
«Действие»	Тип действия при обработке правила. Возможные значения: • «Allow» – разрешить обработку пакета; • «Deny» – запретить обработку пакета
«VLAN»	Выбор VLAN, для которой создается правило
Секция «Источник»	
«Сеть»	Адреса источника запроса в формате CIDR, для которых создается правило
«Диапазон»	Использование диапазона портов источника запроса Параметр доступен для протоколов типа «TCP» и «UDP».

«Порт»	Порт или диапазон портов источника запроса:
	
	• при отключенном параметре «Диапазон» в раскрывающемся списке нужно выбрать условие сравнения и указать порт:
	  – «=» – применять правило к указанному значению;
	  – «>» – применять правило к значению больше указанного;
	  – «<» – применять правило к значению меньше указанного;
	  – «≠» – применять правило ко всем значениям, не соответствующим указанному.
	• при включенном параметре «Диапазон» указываются начальный и конечный порт диапазона
	
	Параметр доступен для протоколов типа «TCP» и «UDP».

Секция «Назначение»	
«Сеть»	Адреса назначения в формате CIDR, для которых создается правило
«Диапазон»	Использование диапазона портов назначения Параметр доступен для протоколов типа «TCP» и «UDP».

Параметр	Описание
«Порт»	Порт или диапазон портов назначения: • при отключенном параметре «Диапазон» в раскрывающемся списке нужно выбрать условие сравнения и указать порт: ◦ «=» – применять правило к указанному значению; ◦ «>» – применять правило к значению больше указанного; ◦ «<» – применять правило к значению меньше указанного; ◦ «≠» – применять правило ко всем значениям, не соответствующим указанному. • при включенном параметре «Диапазон» указываются начальный и конечный порт диапазона Параметр доступен для протоколов типа «TCP» и «UDP»

Веб. AAA

Веб. AAA-Серверы

Для отображения списка AAA-Серверов следует перейти «Безопасность– AAA – Серверы».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены в таблице (см. Основные параметры списка AAA-Серверов).

Таблица 175. Основные параметры списка AAA-Серверов

Параметр	Описание
«Имя»	Наименование AAA-Сервера

Для добавления AAA-Сервера следует перейти «Безопасность – AAA – Серверы» и нажать экранную кнопку **[Добавить]**.

Далее заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления AAA–Сервера).

Таблица 176. Данные для добавления AAA–Сервера

Параметр	Описание
«Имя»	Наименование AAA-Сервера
«Тип Сервера»	Тип службы каталогов для подключения. Возможные значения: • «AD» – подключение к службе каталогов с поддержкой Microsoft Active Directory; • «OpenLDAP» – подключение к службе каталогов с поддержкой реализации с открытым исходным кодом протокола LDAP – OpenLDAP
«IP-адрес»	Адрес для подключения к службе каталогов
«Порт»	Номер порта. Значение по умолчанию: «389»

Параметр	Описание
«Время ожидания»	Время (в секундах) ожидания авторизации. Значение по умолчанию: «30»
«TLS Профиль»	Выбор Клиентского SSL-Профиля для подключения к службе каталогов
«Base DN»	Корневая точка службы каталогов
«Administrator Bind DN»	Полное имя учетной записи администратора службы каталогов. Так же имя может быть получено из Хранилища секретов. Пример значения параметра: • «kv://secret/data/my_ldap#dn» – для `kv` версии 2; • «ad://ldap/static-cred/my_ldap#dn» – для `ldap`
«Пароль»	Пароль для соединения. Так же пароль может быть получен из Хранилища секретов. Пример значения параметра: • «kv://secret/data/my_ldap#password» – для `kv` версии 2; • «ad://ldap/static-cred/my_ldap#last_password» – для `ldap`
«Атрибут имени пользователя»	Атрибут имени пользователя. Значение по умолчанию: «userPrincipalName»
«Импортируемые атрибуты»	Список атрибутов, которые должны быть импортированы из службы каталогов
«Атрибут Группы»	Атрибут группы. Значение по умолчанию: «memberOf»

Веб. AAA-Профили

Для отображения списка AAA-Профилей следует перейти «Безопасность – AAA – Профили».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены в таблице (см. Основные параметры списка AAA-Профилей).

Таблица 177. Основные параметры списка AAA-Профилей

Параметр	Описание
«Имя»	Наименование AAA-Профиля

Для добавления AAA-Профиля следует перейти «Безопасность – AAA – Профили» и нажать экранную кнопку **[Добавить]**.

Далее заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления AAA–Профиля).

Таблица 178. Данные для добавления AAA–Профиля

Параметр	Описание
«Имя»	Наименование AAA-Профиля

Параметр	Описание
«Аутентификация»	Режим аутентификации пользователя. При активации параметра выполняется аутентификация пользователя с проверкой учетных данных. Пользователь должен предоставить корректное имя и пароль. При отключении параметра проверка пароля не выполняется. Для запроса атрибутов пользователя у сервера аутентификации используется только его имя (например, полученное из SSL-сертификата)
«REALM»	Пространство имен пользователя для аутентификации
«Время жизни сессии, с»	Время (в секундах) жизни cookie аутентификации. Возможные значения: от 1 до 8640. Значение по умолчанию: «600»
«Время до блокировки, с»	Время (в секундах) блокировки. Возможные значения: от 1 до 8640. Значение по умолчанию: «600»
«Количество попыток аутентификации»	Количество неудачных попыток аутентификации. Возможные значения: от 1 до 255. Значение по умолчанию: «3»
«Таймаут попытки, с»	Период (в секундах), в течение которого подсчитываются неудачные попытки аутентификации. Возможные значения: от 1 до 8640. Значение по умолчанию: «600»
«Имя HTTP cookie»	Имя аутентификационной cookie. Значение по умолчанию: «TC-AAC»
«Атрибуты HTTP cookie»	Атрибуты аутентификационной cookie. Пример: «SameSite=Strict; HttpOnly»
«Серверы аутентификации»	Область добавления AAA-Сервера. Для доступа к параметрам области нажать экранную кнопку **[+ Добавить]** Может быть задано несколько настроенных AAA-Серверов. В случае недоступности одного AAA-Сервера осуществляется переход на другой.
«Порядковый номер»	Порядковый номер привязки настроенного AAA-Сервера
«Имя»	Имя настроенного AAA-Сервера

Веб. KDC-Серверы

Для отображения списка KDC-Серверов следует перейти «Безопасность– AAA – KDC Серверы».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены в таблице (см. Основные параметры списка KDC-

Серверов).

Таблица 179. Основные параметры списка KDC-Серверов

Параметр	Описание
«Имя»	Наименование KDC-Сервера

Для добавления KDC-Сервера следует перейти «Безопасность – AAA – KDC Серверы» и нажать экранную кнопку **[Добавить]**.

Далее заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления KDC–Сервера).

Таблица 180. Данные для добавления KDC–Сервера

Параметр	Описание
«Имя»	Наименование KDC-Сервера
«IP-адрес»	IP-адрес для подключения к серверу KDC
«Порт»	Номер порта для подключения к серверу KDC. Значение по умолчанию: «88»
«Время ожидания»	Время (в секундах) ожидания ответа от сервера KDC. Значение по умолчанию: «30»

Веб. SSO-Профили

Для отображения списка SSO-Профилей следует перейти «Безопасность – AAA – SSO Профили».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены в таблице (см. Основные параметры списка SSO-Профилей).

Таблица 181. Основные параметры списка SSO-Профилей

Параметр	Описание
«Имя»	Наименование SSO-Профиля
«Тип»	Тип аутентификации

Для добавления SSO-Профиля следует перейти «Безопасность – AAA – SSO Профили» и нажать экранную кнопку **[Добавить]**.

Далее заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления SSO-Профиля).

Таблица 182. Данные для добавления SSO-Профиля

Параметр	Описание
«Имя»	Наименование SSO-Профиля

Параметр	Описание
«Тип»	Тип аутентификации, т.е. как данные пользователя передаются на Реальный Сервер. Возможные значения: • «BASIC» – базовый алгоритм аутентификации; • «OFF» – данные аутентификация не передаются; • «KRB5-IM» – аутентификация с использованием Kerberos версии 5
«realm»	Пространство имен пользователя для аутентификации. Параметр доступен при выборе типа аутентификации «KRB5-IM»
«Имя пользователя»	Имя пользователя. Так же имя может быть получено из Хранилища секретов. Пример значения параметра: • «kv://secret/kerberos#service_principal» – для kv версии 1; • «kv://secret/data/kerberos#service_principal» – для kv версии 2. Параметр доступен при выборе типа аутентификации «KRB5-IM»
«Пароль»	Пароль. Так же пароль может быть получен из Хранилища секретов. Пример значения параметра: • «kv://secret/kerberos#password» – для kv версии 1; • «kv://secret/data/kerberos#password» – для kv версии 2. Параметр доступен при выборе типа аутентификации «KRB5-IM»
«KDC-Серверы»	Область добавления KDC-Сервера. Для доступа к параметрам области нажать экранную кнопку **[+ Добавить]**. Параметр доступен при выборе типа аутентификации «KRB5-IM» Может быть задано несколько настроенных KDC-Серверов. В случае недоступности одного KDC-Сервера осуществляется переход на другой.
«Порядковый номер»	Приоритет настроенного KDC-Сервера
«Сервер»	Имя настроенного KDC-Сервера

Веб. DDoS-Профили

Для отображения списка DDoS-Профилей следует перейти «Настройки – Безопасность – DDoS».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены в таблице (см. Основные параметры списка DDoS-Профилей).

Таблица 183. Основные параметры списка DDoS-Профилей

Параметр	Описание
«Имя»	Имя DDoS-Профиля
«SYN Flood»	Статус защиты от SYN-атаки. Может быть: • «Не обнаружен» – защита от SYN-атаки не активна; • «Обнаружен Интерфейс» – обнаружена SYN-атака на связанном интерфейсе (если много пользователей отправляют мало пакетов данных); • «Обнаружен IP» – обнаружена SYN-атака с определенного IP-адреса (если один пользователь отправляет много пакетов данных)
«ICMP Flood»	Статус защиты от ICMP-атаки. Может быть: • «Не обнаружен» – защита от ICMP-атаки не активна; • «Обнаружен Интерфейс» – обнаружена ICMP-атака на связанном интерфейсе (если много пользователей отправляют мало пакетов данных); • «Обнаружен IP» – обнаружена ICMP-атака с определенного IP-адреса (если один пользователь отправляет много пакетов данных)
«UDP Flood»	Статус защиты от UDP-атаки. Может быть: • «Не обнаружен» – защита от UDP-атаки не активна; • «Обнаружен Интерфейс» – обнаружена UDP-атака на связанном интерфейсе (если много пользователей отправляют мало пакетов данных); • «Обнаружен IP» – обнаружена UDP-атака с определенного IP-адреса (если один пользователь отправляет много пакетов данных)

Для добавления DDoS-Профиля следует перейти «Настройки – Безопасность – DDoS» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления DDoS-Профиля).

Таблица 184. Данные для добавления DDoS-Профиля

Параметр	Описание
«Имя»	Имя DDoS-Профиля
«Журналировать»	Настройка журналирования при обнаружении или завершении DDoS-атаки. При активации параметра включает журналирование DDoS-атак, при отключении параметра выключает журналирование
«SYN Flood, syn/c»	Настройка защиты от SYN-атак. При активации параметра включает защиту от SYN-атаки, при отключении параметра выключает защиту
«ICMP Flood, пак/с»	Настройка защиты от ICMP-атак. При активации параметра включает защиту от ICMP-атаки, при отключении параметра выключает защиту
«UDP Flood, пак/с»	Настройка защиты от UDP-атак. При активации параметра включает защиту от UDP-атаки, при отключении параметра выключает защиту

Параметр	Описание
«Пороговое значение на Интерфейсе»	Порог срабатывания ограничения по всему интерфейсу. Значение по умолчанию: «0». Параметр доступен при активации параметров «SYN Flood», «ICMP Flood» или «UDP Flood»
«Гистерезис на Интерфейсе»	Гистерезис порогового значения по всему интерфейсу, т.е. на сколько должен снизиться трафик для выключения защиты. Значение по умолчанию: «0». Параметр доступен при активации параметров «SYN Flood», «ICMP Flood» или «UDP Flood»
«Пороговое значение от одного IP-адреса»	Порог срабатывания ограничения для одного IP-адреса. Значение по умолчанию: «0». Параметр доступен при активации параметров «SYN Flood», «ICMP Flood» или «UDP Flood»
«Гистерезис от одного IP-адреса»	Гистерезис порогового значения для одного IP-адреса, т.е. на сколько должен снизиться трафик для выключения защиты. Значение по умолчанию: «0». Параметр доступен при активации параметров «SYN Flood», «ICMP Flood» или «UDP Flood»

Веб. Профили ограничения скорости

Для отображения списка Профилей ограничения скорости следует перейти «Настройки – Безопасность – Ограничение скорости».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены в таблице (см. таблицу Основные параметры списка Профилей ограничения скорости).

Таблица 185. Основные параметры списка Профилей ограничения скорости

Параметр	Описание
«Имя»	Имя Профиля ограничения скорости
«Тип»	Тип Профиля ограничения скорости

Для добавления Профиля ограничения скорости следует перейти «Настройки – Безопасность – Ограничение скорости» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. таблицу Данные для добавления Профиля ограничения скорости).

Таблица 186. Данные для добавления Профиля ограничения скорости

Параметр	Описание
Общие настройки	
«Имя»	Имя Профиля ограничения скорости

Параметр	Описание
«Тип»	Тип Профиля ограничения скорости. Возможные значения: • «Виртуальный Сервер»; • «Сервер Балансировки»; • «Группа РС»
«Комментарий»	Комментарий, который будет привязан к Профилю ограничения скорости
Настройки ограничения скорости в секции «Входящий трафик»	
«Порог скорости»	Скорость передачи данных. Доступен выбор единиц измерения для указанного значения: • бит/с – бит в секунду; • Кбит/с – килобит в секунду; • Мбит/с – мегабит в секунду; • Гбит/с – гигабит в секунду
«пакетов/с»	Количество пакетов в секунду
«сессий/с»	Количество установленных TCP-сессий. Параметр доступен при выборе типа Профиля ограничения скорости «Виртуальный Сервер»
Настройки ограничения размера всплеска в секции «Входящий трафик»	
«Порог всплеска»	Размера всплеска. Доступен выбор единиц измерения для указанного значения: • бит/с – бит в секунду; • Кбит/с – килобит в секунду; • Мбит/с – мегабит в секунду; • Гбит/с – гигабит в секунду
«пакетов/с»	Количество пакетов в секунду
«сессий/с»	Количество установленных TCP-сессий. Параметр доступен при выборе типа Профиля ограничения скорости «Виртуальный Сервер»
Настройки ограничения скорости в секции «Исходящий трафик»	
«Порог скорости»	Скорость передачи данных. Доступен выбор единиц измерения для указанного значения: • бит/с – бит в секунду; • Кбит/с – килобит в секунду; • Мбит/с – мегабит в секунду; • Гбит/с – гигабит в секунду. Параметр доступен при выборе типа Профиля ограничения скорости «Группа РС»
«пакетов/с»	Количество пакетов в секунду. Параметр доступен при выборе типа Профиля ограничения скорости «Группа РС»

Параметр	Описание
Настройки ограничения размера всплеска в секции «Исходящий трафик»	
«Порог всплеска»	Размера всплеска. Доступен выбор единиц измерения для указанного значения: • бит/с – бит в секунду; • Кбит/с – килобит в секунду; • Мбит/с – мегабит в секунду; • Гбит/с – гигабит в секунду. Параметр доступен при выборе типа Профиля ограничения скорости «Группа PC»
«пакетов/с»	Количество пакетов в секунду. Параметр доступен при выборе типа Профиля ограничения скорости «Группа PC»

Веб. Хранилище секретов

Для отображения настроек Хранилища секретов следует перейти «Настройки – Безопасность – Хранилище секретов.

Основные параметры настроек Хранилища секретов приведены в таблице (см. таблицу Основные параметры настройки Хранилища секретов).

Таблица 187. Основные параметры настройки Хранилища секретов

Параметр	Описание
Общие настройки	
«URL»	URL-адрес Хранилища секретов. Пример значения параметра: `http://192.0.2.10:8200`
«VRF»	Имя VRF для подключения к Хранилищу секретов. Значение по умолчанию: «default»
«Время ожидания ответа HTTP-запроса, с»	Максимально допустимое время (в секундах) ожидания ответа HTTP-запроса от API-сервера. Значение по умолчанию: «60»
«Интервал повторной отправки HTTP-запроса, с»	Интервал (в секундах) повторной отправки HTTP-запроса при возникновении ошибки. Значение по умолчанию: «10»
«Интервал обновления токена авторизации, с»	Интервал (в секундах) обновления токена авторизации. Значение по умолчанию: «3600»
«Интервал обновления секретов, с»	Интервал (в секундах) обновления секретов. Termidesk Connect с заданной периодичностью будет отправлять запросы в Хранилище секретов для получения новых значений секретов. Значение по умолчанию: «1800»

Параметр	Описание
«Интервал обновления сертификатов, с»	Интервал (в секундах) обновления сертификатов. Termidesk Connect с заданной периодичностью будет отправлять запросы в Хранилище секретов для получения новых сертификатов. Значение по умолчанию: «86400»
«Пространство имён в Vault Enterprise»	Пространство имен (Namespace), заданное в Хранилище секретов
Секция «Проверка токена»	
«Путь к методу аутентификации в OpenBao»	Путь к методу аутентификации (auth method) в Хранилище секретов
«Идентификатор роли для метода аутентификации AppRole»	Идентификатор роли (RoleID) для метода аутентификации AppRole
«Полный путь к файлу с Secret ID или токеном для unwrapping»	Путь к файлу, содержащему идентификатор секрета (SecretID) или токен для его раскрытия (unwrapping)
«Wrapping Token»	Настройка использования упакованного токена (wrapped-token) для идентификатора секрета (SecretID). При активации параметра содержимое файла будет предварительно раскрыто (unwrapped) для получения действительного идентификатора секрета (SecretID). При отключении параметра содержимое файла будет упаковано (wrapped) для получения действительного идентификатора секрета (SecretID)
Секция «SSL/TLS соединение»	
«Полный путь к файлу ключа»	Полный путь к файлу сертификата УЦ (указывается полный путь и имя файла с расширением)
«Полный путь к файлу сертификата»	Полный путь к файлу сертификата (указывается полный путь и имя файла с расширением)
«Полный путь к файлу сертификата УЦ»	Полный путь к файлу ключа (указывается полный путь и имя файла с расширением)

Функция «Шлюз»

Веб. Точки подключений

Для отображения списка точек подключения Шлюза следует перейти «Настройки – Шлюз – Точки подключений».

По умолчанию записи представлены в табличном виде и упорядочены согласно столбцу «Имя».

Основные параметры списка приведены в таблице (см. Основные параметры списка точек подключения Шлюза).

Таблица 188. Основные параметры списка точек подключения Шлюза

Параметр	Описание
«Имя»	Наименование Шлюза
«IP»	IP-адрес Виртуального Сервера для Шлюза
«Порт»	Порт Виртуального Сервера для Шлюза
«VRF»	VRF для Шлюза

Для добавления Шлюза следует перейти «Настройки – Шлюз – Точки подключений» и нажать экранную кнопку **[Добавить]**.

Затем заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Данные для добавления Шлюза).

Таблица 189. Данные для добавления Шлюза

Параметр	Описание
Общие настройки Шлюза	
«Имя»	Наименование Шлюза
«IP»	IP-адрес Виртуального Сервера
«Порт»	Порт Виртуального Сервера
«TLS Профиль Сервера»	Серверный SSL-Профиль
«VRF»	Выбор VRF. Значение по умолчанию: «default»
«Комментарий»	Комментарий для сервера
Настройки TERA	
«TERA UDP»	Настройка поддержки протокола UDP для протокола TERA на Шлюзе. Активация параметра открывает доступ к настройкам протокола TERA, отключение параметра закрывает доступ к настройкам протокола TERA
«TERA IP»	IP-адрес Виртуального Сервера для TERA. Параметр «TERA IP» может принимать любое значение, т.е. IP-адрес может отличаться от IP-адреса Шлюза или может быть таким же. Указывается IP-адрес именно Виртуального Сервера, если Шлюз скрывается за Виртуальным Сервером. Доступно при активированном параметре «TERA UDP»
«TERA Порт»	Порт Виртуального Сервера для TERA. Значение по умолчанию: «443». Доступно при активированном параметре «TERA UDP»
«VRF»	VRF для TERA. Значение по умолчанию: «default». Доступно при активированном параметре «TERA UDP»
«IP Источника»	Выбор IP-адреса источника, т.е. адрес, с которого Шлюз устанавливает UDP-соединение с рабочим местом Termidesk. Значение «0.0.0.0» указывает на использование всех IP-адресов. Доступно при активированном параметре «TERA UDP»
«Порт Источника»	Порт источника. Значение по умолчанию: «0». Доступно при активированном параметре «TERA UDP»

Параметр	Описание
«VRF»	VRF источника. Значение по умолчанию: «default». Доступно при активированном параметре «TERA UDP»
Область «Проверка Токена»	
«URL»	URL, поддерживаемый компонентом «Универсальный диспетчер» Termidesk с ролью «Портал пользователя» для обслуживания API-запросов по валидации подключений, запрашиваемых через Шлюз.  Следует использовать актуальные значения версий API для соответствующей версии компонента «Универсальный диспетчер» Termidesk.
«Фонд IP»	Выбор IP-Фонда, используемого для взаимодействия с компонентом «Универсальный диспетчер» Termidesk
«TLS Профиль»	Выбор Клиентского SSL-Профиля. Значение по умолчанию: «backend-default»
Область «Подключение к рабочим местам»	
«Количество переподключений»	Количество разрешенных переподключений Шлюза к рабочим местам Termidesk. Возможные значения: от 0 до 10. Значение по умолчанию: «0»
«Фонд IP»	Выбор IP-Фонда для взаимодействия с рабочими местам Termidesk
«TLS Профиль»	Выбор Клиентского SSL-Профиля. Значение по умолчанию: «backend-default»
Область «Координатор»	
«URL»	URL-адрес для подключения к серверу RabbitMQ
«Пользователь»	Имя пользователя для подключения к серверу RabbitMQ. Так же имя пользователя может быть получено из Хранилища секретов. Пример значения параметра: • «kv://secret/rabbit#username» – для kv версии 1; • «kv://secret/data/rabbit#username» – для kv версии 2
«Пароль»	Пароль пользователя для подключения к серверу RabbitMQ. Так же пароль пользователя может быть получен из Хранилища секретов. Пример значения параметра: • «kv://secret/rabbit#password» – для kv версии 1; • «kv://secret/data/rabbit#password» – для kv версии 2
«Фонд IP»	Выбор IP-Фонда для взаимодействия с RabbitMQ
«Timeout»	Время (в секундах) ожидания ответа от сервера RabbitMQ. Возможные значения: от 1 до 60. Значение по умолчанию: «10»

Параметр	Описание
«Интервал ожидания»	Интервал (в секундах) обновления регистрационной информации (URL и другие данные) Шлюза. Возможные значения: от 10 до 100000. Значение по умолчанию: «60»
«TLS Профиль»	Выбор Клиентского SSL-Профиля
«Exchange»	Координатор маршрутизации сообщений, определенный в RabbitMQ. Отвечает за маршрутизацию сообщений в разные очереди
«Ключ»	Ключ маршрутизации RabbitMQ, используемый для маршрутизации задачи в очереди. Значение по умолчанию: «termidesk_appnode»
«Single»	Настройка способа передачи данных. При активации параметра данные будут передаваться до первого подтверждения, при отключении параметра данные будут передаваться по циклу (бесконечно)

Веб. Координатор

Для отображения адреса подключения к серверу RabbitMQ следует перейти «Настройки – Шлюз – Координатор».

Для изменения настроек клиента RabbitMQ следует нажать экранную кнопу **[Изменить]**.

Далее заполнить данные, перечисленные в столбце «Параметр» следующей таблицы (см. Основные параметры настройки клиента RabbitMQ).

Таблица 190. Основные параметры настройки клиента RabbitMQ

Параметр	Описание
«URL»	URL-адрес для подключения к серверу RabbitMQ
«Пользователь»	Имя пользователя для подключения к серверу RabbitMQ. Так же имя пользователя может быть получено из Хранилища секретов. Пример значения параметра: • «kv://secret/rabbit#username» – для kv версии 1; • «kv://secret/data/rabbit#username» – для kv версии 2
«Пароль»	Пароль пользователя для подключения к серверу RabbitMQ. Так же пароль пользователя может быть получен из Хранилища секретов. Пример значения параметра: • «kv://secret/rabbit#password» – для kv версии 1; • «kv://secret/data/rabbit#password» – для kv версии 2
«Фонд IP»	Выбор IP-Фонда для взаимодействия с RabbitMQ
«Timeout»	Время (в секундах) ожидания ответа от сервера RabbitMQ. Возможные значения: от 1 до 60
«Интервал ожидания»	Интервал (в секундах) обновления регистрационной информации (URL и другие данные) Шлюза. Возможные значения: от 10 до 100000

Параметр	Описание
«TLS Профиль»	Выбор Клиентского SSL-Профиля. Значение по умолчанию: «backend-default»
«Exchange»	Координатор маршрутизации сообщений, определенный в RabbitMQ. Отвечает за маршрутизацию сообщений в разные очереди
«Ключ»	Ключ маршрутизации RabbitMQ, используемый для маршрутизации задачи в очереди
«Single»	Настройка способа передачи данных. При активации параметра данные будут передаваться до первого подтверждения, при отключении параметра данные будут передаваться по циклу (бесконечно)

Веб. Сбор статистики

Для отображения настроек сбора статистики Шлюза следует перейти «Настройки – Шлюз – Сбор статистики».

Основные параметры настроек сбора статистики приведены в таблице (см. Основные параметры настроек сбора статистики).

Таблица 191. Основные параметры настроек сбора статистики

Параметр	Описание
«Path»	Путь для получения статистики Шлюза
«IP»	IP-адрес для получения статистики Шлюза

Для изменения настроек сбора статистики следует нажать экранную кнопу **[Изменить]**.

Далее заполнить данные, перечисленные в столбце «Параметр» следующей таблицы(см. Основные параметры изменения настройки сбора статистики).

Таблица 192. Основные параметры изменения настройки сбора статистики

Параметр	Описание
«IP»	Выбор IP-адреса для получения статистики Шлюза
«Порт»	Порт для получения статистики Шлюза
«TLS Профиль»	Серверный SSL-Профиль
«Метрики»	Путь для получения статистики Шлюза. Значение по умолчанию: «/api/health»
«OpenMetrics»	Путь для получения метрик Шлюза. Значение по умолчанию: «/api/health/metrics»
«Токен получения статистики Шлюза»	Токен доступа для получения статистики Шлюза. Так же токен может быть получен из Хранилища секретов. Пример значения параметра: • «kv://secret/gw#token» – для kv версии 1; • «kv://secret/data/gw#token» – для kv версии 2

Параметр	Описание
«Токен получения OpenMetrics»	Токен доступа для получения метрик Шлюза. Так же токен может быть получен из Хранилища секретов. Пример значения параметра: • «kv://secret/gw#token» – для `kv` версии 1; • «kv://secret/data/gw#token» – для `kv` версии 2

ИНТЕРФЕЙС API

Общие сведения по работе с API

Обращение к Termidesk Connect осуществляется через IP-адрес и порт управления, которые задаются при первоначальной настройке по HTTPS-протоколу (см. подраздел **Первоначальная настройка Termidesk Connect** документа СЛЕТ.10101-01 90 01 «Руководство администратора. Установка Termidesk Connect»).

Пример команд:

```
set system mgmt ip 192.0.2.5
set system mgmt webui-port 443
```

Аутентификация и авторизация осуществляется одним из способов:

- через пользователей, настроенных в Termidesk Connect;
- с помощью внешних серверов аутентификации и авторизации (аналогично CLI или WUI).

Ролевая модель управления доступом распространяется и на API-интерфейс. Для взаимодействия с API используется путь обращения: `https://<IP_управления>/xtern/data/⋯`.

Обмен данными осуществляется в формате `JSON`. Для формирования корректного `JSON`, `YANG` -файл должен содержать:

- модуль с определением префикса;
- структуру (`container` или `list`) для группировки данных;
- `leaf` с конкретными типами;
- ключи списков (если есть `list`), которые будут идентифицировать элементы массива.

Базовые соответствия типов:

- `container` – `JSON`-объект `{ }`;
- `list` – `JSON`-массив `[ ]`;
- `leaf` – `JSON`-скаляр (`string`, `number`, `boolean`, `null`);
- `leaf-list` – `JSON`-массив скаляров.

Запрос должен содержать:

- заголовок `Authorization` с строкой `base64`, содержащей имя пользователя и пароль;
- заголовок `Content-Type` со значением `application/yang-data+json` или `application/json`.

Типы HTTP-запросов представлены в таблице (см. Типы HTTP-запросов).

Таблица 193. Типы HTTP-запросов

Тип запроса	Назначение
DELETE	Удаление конфигурации
GET	Получение данных (только чтение данных)
POST	Добавление конфигурации
PUT	Обновление конфигурации

Рассмотрим примеры запросов:

- команда `curl`:

```
curl -vvk -H "Authorization: Basic dGRhZG1pbjp0ZGFkbWluCg==" -H "Content-Type:
application/json" https://192.0.2.5/xtern/data/termidesk-monitors:health-check
```

- сервис Postman (см. Postman).

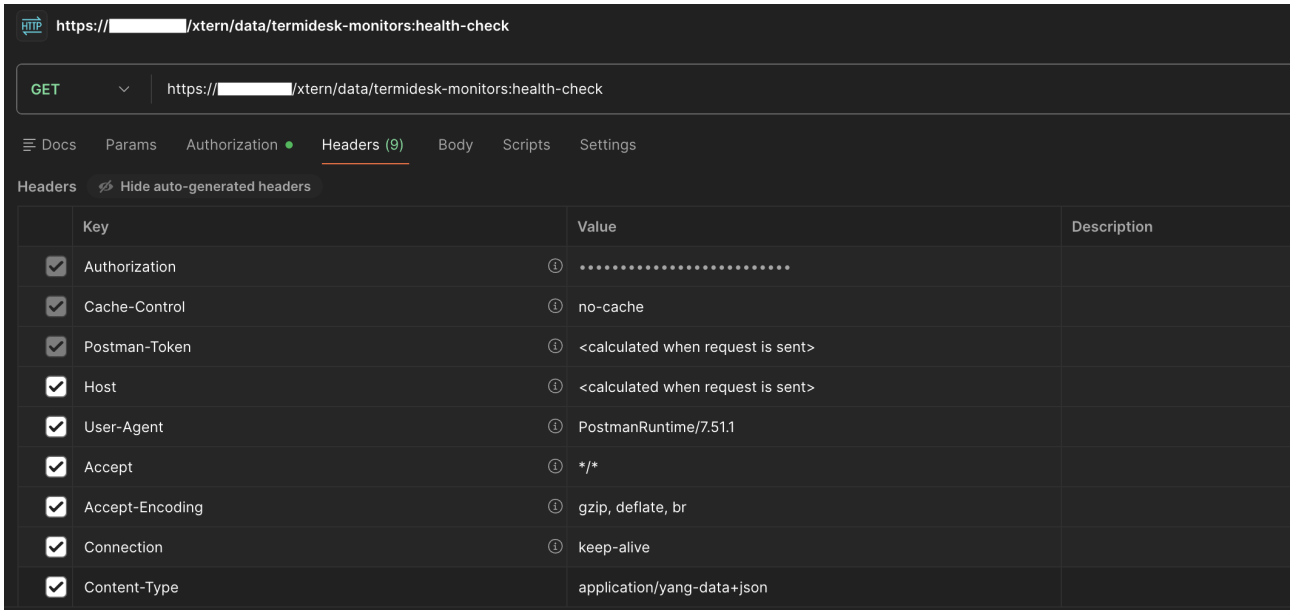

Рисунок 69. Postman

Примеры конфигурации API

В качестве примеров конфигурации API представлены:

- Создание Проверки;
- Создание Группы Реальных Серверов;
- Создание Сервера Балансировки;
- Создание Виртуального Сервера.

Создание Проверки

Рассмотрим Проверку со следующими параметрами:

- имя `APIEXP`;

- тип HTTP;
- ожидаемые ответы `200` и `302`.

Пример команды:

```
curl -k -X PATCH -H "Content-Type: application/yang-data+json" --user
tdadmin:tdadmin https://192.0.2.7/xtern/data/termidesk-monitors:health-check/ -d '{
  "termidesk-monitors:health-check": {"id": [{"id": "APIEXP","HTTP": {"interval":
2,"timeout": 1,"try": 1,"vrf": "default","success-try": 1,"ha-monitor":
false,"method": "GET","uri": "/","status-codes": [200,302]}}]}}'
```

Создание Группы Реальных Серверов

Рассмотрим Группу Реальных Серверов со следующими параметрами:

- имя `apiRSp`;
- Реальные Сервера: `10.101.34.101:80` и `10.101.34.100:80`;
- HTTP-Проверка `APIEXP`.

Пример команды:

```
curl -kvv -X PATCH -H "Content-Type: application/yang-data+json" --user
tdadmin:tdadmin https://192.0.2.7/xtern/data/termidesk-rspool:rs-pool -d
'{"termidesk-rspool:rs-pool": {"id": [[{"id": "apiRSp","rs": [{"ip":
"10.101.34.100","port": 80,"state": "ENABLE","weight": 1},{"ip":
"10.101.34.101","port": 80,"state": "ENABLE","weight": 1}],"hc-id":
"APIEXP","maintenance-timeout": 0}]]}}'
```

Создание Сервера Балансировки

Рассмотрим Сервер Балансировки со следующими параметрами:

- имя `lbAPI`;
- тип TCP;
- алгоритм балансировки `WEIGHTEDROUNDROBIN`;
- Группа Реальных Серверов `apiRSp`.

Пример команды:

```
curl -kvv -X PATCH -H "Content-Type: application/yang-data+json" --user
tdadmin:tdadmin https://192.0.2.7/xtern/data/termidesk-lbs:lbs -d '{"termidesk-
lbs:lbs":{"TCP":[{"id":"lbAPI","rs-pool-id":"apiRSp","min-
rs":1,"persistence":{"algorithm":"SSLSESSION","ipsource-
param":{"timeout":60},"sslsession-
param":{"timeout":60}},"algorithm":"WEIGHTEDROUNDROBIN","tcp-profile-id":"tcpp-
client-default","re-balancing":{"enable":false,"max-attempts":1},"ha-
monitor":false}]}}'
```

Создание Виртуального Сервера

Рассмотрим Виртуальный Сервер со следующими параметрами:

- имя `vsAPI`;
- тип TCP;
- правило: все запросы отправляются на Сервер Балансировки `lbAPI`.

Пример команды:

```
curl -kvv -X PUT -H "Content-Type: application/yang-data+json" --user
tdadmin:tdadmin https://192.0.2.7/xtern/data/termidesk-vs:vs/TCP=vsAPI -d
'{"termidesk-
vs:TCP":[{"id":"vsAPI","vip":{"ip":"1.2.3.4","port":4433},"vrf":"default","check-
lbs":{"algorithm":"NONE"},"rhi":"OFF","netrules":[{"order":0,"network":"0.0.0.0/0",
"lbs-id":"lbAPI"}],"tcp-profile-id":"tcpp-server-default"}]}'
```

ТЕРМИНЫ

AAA-Профиль	Совокупность параметров аутентификации пользователей, таких как количество попыток входа, порядок выбора серверов аутентификации и другое
AAA-Сервер	Совокупность параметров взаимодействия с сервером аутентификации и авторизации
DDoS-Профиль	Настройки и параметры, используемые для защиты от различных типов DDoS-атак
DNS View	Объект (абстракция), описывающий сети, при запросе из которых должен быть выдан локальный IP-адрес Сервиса
Full Proxy	Режим, при котором Termidesk Connect является посредником между клиентом и Реальным Сервером и полностью обрабатывает входящие и исходящие соединения
HTTP-Проверка	Проверка, основанная на использовании протокола HTTP
HTTP-Профиль	Настройки и параметры для работы с HTTP-запросами
HTTPS-Проверка	Проверка, основанная на использовании защищенного протокола HTTPS
ICMP-Проверка	Проверка, основанная на использовании протокола ICMP для определения состояния сетевого соединения
IP-Фонд	Группа IP-адресов, которые может использовать Termidesk Connect для взаимодействия с Реальными Серверами

KDC-Сервер	Совокупность параметров взаимодействия с сервером аутентификации и авторизации KDC
Linux Shell	Программный интерфейс для взаимодействия пользователя с ОС
PROXY-протокол	Сетевой протокол, который позволяет передавать информацию о пользователе (его IP-адрес и порт) через цепочку PROXY
SSL Offload	Процесс, в котором преобразование трафика SSL/TLS выполняется не на серверах приложений (Реальных Серверах), а на Termidesk Connect
SSL-Профиль	Набор настроек протокола SSL (сертификат, ключ, используемые алгоритмы)
TCP-Проверка	Проверка, основанная на использовании протокола TCP
TCP-Профиль	Настройки и параметры, используемые для TCP-соединения
Termidesk (Termidesk VDI)	Программный комплекс «Диспетчер подключений виртуальных рабочих мест»
Termidesk Connect	Программа для электронной вычислительной машины «Балансировщик нагрузки Термидеск Коннект»
TLS Hello	Начальный этап установки соединения по протоколу TLS
Вес	Весовой коэффициент, показывающий во сколько раз нагрузка на данный сервер должна быть выше, чем на другие
Виртуальный Сервер	Объект (абстракция), терминирующий на себя трафик от пользователя
Виртуальный Сервер геобалансировки	Объект (абстракция) в рамках геобалансировки, терминирующий на себя трафик от пользователя
Группа Реальных Серверов	Объединение нескольких Реальных Серверов и их периодических Проверок
Закрытый ключ	Сохраняемый в тайне ключ из ключевой пары, принадлежащий владельцу и не подлежащий распространению
Зона	Объект, описывающий доменные имена, для которых обеспечивается геобалансировка подключений
Клиентский SSL-Профиль	Профиль для обеспечения безопасного соединения между Termidesk Connect и Реальным Сервером

Ключ	Параметр в виде последовательности псевдослучайных чисел
Ключевая пара	Упорядоченная пара математически однозначно связанных ключей, определяющих взаимосвязанные защитные преобразования
Комбинированная Проверка	Проверка, основанная на использовании нескольких Проверок с заданием их приоритетов (через логические выражения `AND/OR`)
Мастер-узел («Active»)	Устройство в рамках отказоустойчивой (высокодоступной) конфигурации, обрабатывающее трафик пользователей в настоящий момент
Открытый ключ	Ключ из ключевой пары, который может быть сделан общедоступным
Площадка	Группа серверов (или ЦОД) геораспределенной балансировки
Пользовательская Проверка	Проверка, основанная на использовании пользовательских скриптов
Предварительная конфигурация («candidate»)	Временное хранилище конфигурации, где вносятся изменения перед их применением. Изменения в такой конфигурации можно редактировать, проверять и отменять без влияния на рабочую конфигурацию Termidesk Connect. Чтобы изменения вступили в силу, их нужно записать командой `commit`
Проверка	Набор правил для проверки балансируемых (Реальных) Серверов
Профиль ограничения скорости	Совокупность параметров для ограничения трафика
Профиль сохранения сессий	Совокупность параметров для привязки пользователя к Реальному Серверу
Рабочая конфигурация («running»)	Конфигурация в данный момент использующуюся Termidesk Connect. Рабочая конфигурация будет обновлена, если для предварительной конфигурации выполнить команду `commit`. Если Termidesk Connect перезагрузится, то изменения в рабочей конфигурации будут сброшены, если не выполнена команда `write`
Реальный Сервер	Узел с установленным приложением, доставку которого обеспечивает Termidesk Connect
Резервный узел	Устройство в рамках отказоустойчивой (высокодоступной) конфигурации, не обрабатывающее трафик пользователей в

	настоящий момент
Сервер Балансировки	Объект (абстракция), с заданным алгоритмом балансировки и другими параметрами, реализующий перенаправление подключения пользователя на один из Реальных Серверов
Серверный SSL-Профиль	Профиль для обеспечения безопасного соединения между пользователем и Termidesk Connect
Сервис	Объект (абстракция), представляющий собой IP-адрес, который может быть в DNS-ответе
Сертификат	Артефакт, содержащий открытый ключ, информацию о владельце ключа и подтверждающий принадлежность открытого ключа владельцу, защищенный с применением закрытого ключа
Скрипт Проверки	Файл, содержащий пользовательскую логику Проверки работоспособности сервиса или узла
Сохраненная конфигурация («startup»)	Представляет собой конфигурацию, загружаемую при старте Termidesk Connect. После перезагрузки Termidesk Connect сохраненная конфигурация копируется в рабочую. Чтобы рабочая конфигурация была загружена при следующем запуске (стала сохраненной конфигурацией), нужно выполнить команду `write`
Ферма	Логическое объединение узлов, взаимодействующих с единой БД
Хеш	Строка бит, являющаяся выходным результатом функции хеширования
Хранилище секретов	Сервер со специализированным ПО, предназначенным для хранения секретных ключей и другой чувствительной информации
Центр сертификации	Программный компонент, реализующий возможность подтверждения подлинности ключей с помощью сертификатов
Шлюз	Компонент для безопасного доступа к опубликованным ресурсам Termidesk VDI и Termidesk Terminal

СОКРАЩЕНИЯ

ВМ	Виртуальная машина
ОС	Операционная система
УЦ	Удостоверяющий центр

ЦОД	Центр обработки данных
AAA	Authentication, Authorization, and Accounting (система аутентификации авторизации и учета событий)
AD	Active Directory (активный каталог)
ADNS	Authoritative Domain Name System (авторитетная система доменных имен)
API	Application Programming Interface (программный интерфейс приложения)
ARP	Address Resolution Protocol (протокол разрешения адресов)
BGP	Border Gateway Protocol (протокол динамической маршрутизации для управления трафиком между автономными системами)
CIDR	Classless Inter-Domain Routing (бесклассовая адресация)
CLI	Command Line Interface (интерфейс командной строки)
CRL	Certificate Revocation List (список отозванных сертификатов)
DNS	Domain Name System (система доменных имен)
DSR	Direct Server Return (метод балансировки нагрузки)
EDNS	Extension Mechanisms for DNS (спецификация расширения DNS)
EFI	Unified Extensible Firmware Interface (унифицированный расширяемый микропрограммный интерфейс)
FIN	Finish (флаг протокола TCP для завершения соединения)
FRR	Free Range Routing (набор служб маршрутизации)
FQDN	Fully Qualified Domain Name (полностью определенное имя домена)
HTTP	HyperText Transfer Protocol (протокол передачи гипертекста)
HTTPS	Hypertext Transfer Protocol Secure (расширение протокола HTTP для поддержки шифрования)
ICMP	Internet Control Message Protocol (протокол межсетевых управляющих сообщений)
IEEE	Institute of Electrical and Electronics Engineers (институт инженеров электротехники и электроники)
IP	Internet Protocol (межсетевой протокол)
IPIP	IP-over-IP (метод сетевого туннелирования)
JSON	JavaScript Object Notation (стандартный текстовый формат для структурированных данных)
KDC	Key Distribution Center (центр распределения ключей)
L2	Второй (канальный) уровень сетевой модели OSI

L3	Третий (сетевой) уровень сетевой модели OSI
L4	Четвертый (транспортный) уровень сетевой модели OSI
L7	Седьмой (транспортный) уровень сетевой модели OSI
LACP	Link Aggregation Control Protocol (протокол агрегирования каналов)
LACPDU	Link Aggregation Control Protocol Data Unit (пакет данных протокола LACP для обмена служебной информацией)
LAG	Link Aggregation Group (виртуальный сетевой интерфейс,объединяющий физические порты в один логический)
LDAP	Lightweight Directory Access Protocol (легковесный протокол доступа к каталогам)
MII	Media Independent Interface (интерфейс, не зависящий от среды передачи)
mTLS	Multiplexed Transport Layer Security (протокол, основанный на TLS с усиленной безопасностью)
MTU	Maximum Transmission Unit (максимальный размер пакета данных в байтах, передаваемый по сети)
NAT	Network Address Translation (трансляция сетевых адресов)
NETCONF	Протокол конфигурации сетевых устройств
OCSP	Online Certificate Status Protocol (онлайн-протокол определения статуса сертификата)
OSI	The Open Systems Interconnection model (модель стека сетевых протоколов)
OSPF	Open Shortest Path First (протокол динамической маршрутизации,основанный на технологии отслеживания состояния канала)
PKI	Public Key Infrastructure (инфраструктура открытых ключей)
RAPID-TCP	Rapid Transmission Control Protocol (оптимизированный протокол управления передачей)
RESTCONF	Протокол, предоставляющий программный интерфейс для доступа к данным, определенным в моделях YANG
RHI	Route Health Injection (технология объявления маршрутов в сети)
SNI	Server Name Indication (расширение протокола TLS для идентификации имени сервера)
SSH	Secure Shell Protocol (протокол защищенной передачи информации)
SSL	Secure Sockets Layer (криптографический протокол)
SSO	Single Sign-On (технология единого входа)
SYN	Запрос на подключение по протоколу TCP

TCP	Transmission Control Protocol (протокол управления передачей)
TLS	Transport Layer Security (протокол защиты транспортного уровня)
TLV	Tag Length Value (метод кодирования двоичных данных, состоящий из трех полей: тег, длина и значение)
TTL	Time To Live (время жизни IP-пакета)
TXT	Формат текстового представления
UDP	User Datagram Protocol (протокол пользовательских дейтаграмм)
vCPU	Virtual Central Processing Unit (виртуальный центральный процессор)
VLAN	Virtual Local Area Network (виртуальная локальная сеть)
VRF	Virtual Routing and Forwarding (технология реализации нескольких виртуальных маршрутизаторов на базе одного физического)
WS	WebSocket (двунаправленный протокол, позволяющий клиенту установить связь с сервером)
WUI	Web User Interface (веб-ориентированный пользовательский интерфейс)
XML	Xtensible Markup Language (расширяемый язык разметки)
YANG	Yet Another Next Generation (язык моделирования данных)

РАБОТА СО СЦЕНАРИЯМИ (LUA-СКРИПТАМИ)

Основные возможности

С помощью Сценариев (lua-скриптов) можно:

- выбирать Сервер Балансировки (Content Switching (cs));
- изменять содержимое запросов/ответов (Rewrite);
- отвечать на HTTP-запрос (Responder);
- прерывать нежелательные соединения.

lua-скрипты применяются к:

- Виртуальному Серверу для:
  - выбора Сервера Балансировки (Content Switching (cs));
  - ответа на HTTP-запрос (Responder);
  - изменения содержимого запросов (Rewrite);
  - прерывания нежелательного соединения;
- Серверу балансировки для:
  - ответа на HTTP-запрос (Responder) (например, при условии получении ответа);
  - изменения содержимого ответов (Rewrite);
  - прерывания нежелательного соединения (например, при условии получении ответа).

Основные положения

Lua-скрипты привязываются к объекту с порядковым номером, определяющим очередность выполнения скрипта.

Пример:

```
set vs HTTP Vs01 luarules 10 script first.lua
set vs HTTP Vs01 luarules 15 script second.lua
```

Важно учитывать:

- правила в lua-скриптах обрабатываются по порядку. В примере выше сначала обрабатываются правила из скрипта first.lua, потом (если не было совпадений) правила из скрипта second.lua;
- правила могут располагаться как в одном файле, так и в нескольких;
- обработка правил останавливается после выражения: client.action = "xxx".

Выражение 'client.action'

Выражение 'client.action' определяет следующие варианты действий:

- client.action = "bs" – возвращает Сервер Балансировки, определенный в client.bs = "<Имя_Сервера_Балансировки>";

- `client.action = "respond"` – возвращает ответ пользователю, не передавая запрос на Реальный Сервер.Содержимое ответа описывается в `client.respond.status`;
- `client.action = "drop"` – сбрасывает соединение;
- `client.action = "pass"` – используется в скриптах для модификации ответов, после этого ответ передается клиенту;
- `client.action = "auth"` – останавливает обработку для аутентификации.

Условия выполнения

Условия выполнения того или иного действия определяются:

- логическими выражениями `if`, `elseif`, `else`, `end`;
- любыми данными, полученными из запроса, а также их комбинацией. Например:
  - `if client.http_req.host == "abc.ru" then` – если имя хоста из запроса эквивалентно «abc.ru», то выполнить какое-либо действие;
  - `if (client.http_req.host == "abc.ru" and client.remote_p.ip == "10.140.0.200") then` – если имя хоста из запроса эквивалентно «abc.ru», и запрос пришел с IP-адреса 10.140.0.200, то выполнить какое-либо действие.

Можно использовать возможности языка для работы с полученными данными, поскольку правила с четким соответствием не всегда применимы.

Пример для `curl qwe.123.ru/abczxc` (см. Пример выражений для работы с данными):

```
> GET /abczxc HTTP/1.1
> Host: qwe.123.ru
> User-Agent: curl/7.81.0
```

Таблица 194. Пример выражений для работы с данными

Выражение	Результат	Комментарий
`if client.http_req.path == "/abcZxc"`	`true`	Эквивалентно
`if client.http_req.path == "/abc"`	`false`	
`if (client.http_req.path:find("zx"))`	`true`	Содержит значение
`client.http_req.host:match("(.*).123.ru")`	`qwe`	Вернет строку, значение ДО « .123.ru»

Примеры для Content Switching

Пример 1. Передача любого запроса на Сервер Балансировки «lb1»:

```
client.bs = "lb1"
client.action = "bs"
```

Пример 2. Если имя хоста в запросе точно соответствует «abc.domain.ru», то направить запрос на Сервер Балансировки «lb1». В противном случае вернуть код ответа `403` (по

умолчанию вернется ошибка 503):

```
if client.http_req.host == "abc.domain.ru" then
        client.bs = "lb1"
        client.action = "bs"
else
        client.respond.status = 403
        client.action = "respond"
end
```

Пример 3. Если имя хоста содержит текст «abc» и запрос пришел из сети 192.0.2.0/24, то вернуть Сервер Балансировки «lb1». В противном случае вернуть код ответа 403 (по умолчанию вернется ошибка 503):

```
if (client.http_req.host:find("abc") and
client.remote_p:is_network("192.0.2.0/24"))
then
        client.bs = "lb1"
        client.action = "bs"
else
        client.respond.status = 403
        client.action = "respond"
end
```

Пример 4. Если путь содержит:

- «red», то вернуть Сервер Балансировки «lb1»;
- «green», то вернуть Сервер Балансировки «lb2».

Если не сработало ни одно из правил, то вернуть Сервер Балансировки «lb-default».

Код:

```
if client.http_req.path:find("red") then
        client.bs = "lb1"
        client.action = "bs"
elseif client.http_req.path:find("green") then
        client.bs = "lb2"
        client.action = "bs"
else
        client.bs = "lb-default"
        client.action = "bs"
end
```

Примеры для Rewrite (запросы)

Пример 1. Если имя хоста содержит текст «abc» И запрос пришел из сети 192.0.2.0/24, то добавить заголовок XFF: <IP-адрес_пользователя_при_подключении>, и перенаправить запрос на

Сервер Балансировки «lb1».

Код:

```
if (client.http_req.host:find("abc") and
client.remote_p:is_network("192.0.2.0/24")) then
        client.http_req.header:field_set("XFF", client.remote_p.ip)
        client.bs = "lb1"
        client.action = "bs"
end
```

Результат:

```
WEB -- 03<p>Method GET</p><p>URL on server: /</p><p>REQ
Headers: </p>Host: abc.domain.ru
User-Agent: curl/7.81.0
Accept: */*
XFF: 192.0.2.5
```

Пример 2. Если имя хоста начинается с «abc», то добавить заголовок `XFF: <IP-адрес_пользователя_при_подключении>` и `Remote-port: <порт-источник_на_клиенте>`, и перенаправить запрос на Сервер Балансировки «lb1».

Код:

```
function startswith(text, prefix)
  return text:find(prefix, 1, true) == 1
end

if startswith(client.http_req.host, "abc") then
  client.http_req.header:field_set("Remote-port", client.remote_p.port)
  client.http_req.header:field_set("XFF", client.remote_p.ip)
  client.bs = "lb1"
  client.action = "bs"
end
```

Пример 3. Если:

- путь запроса содержит «app», то при наличии заголовка X-Forwarded-For добавить IP-адрес клиента в конец запроса;
- заголовка не было в запросе, то добавить новый.

Перенаправить запрос на Сервер Балансировки «lb1».

Код:

```
if client.http_req.path:find("abc") then
        if client.http_req.header:field_count("X-Forwarded-For") > 0 then
```

```
                xff, xffip = client.http_req.header:field_get("X-Forwarded-For")
                client.http_req.header:field_set("X-Forwarded-For", xffip ..",
"..client.remote_p.ip)
        else
                client.http_req.header:field_set("X-Forwarded-For",
client.remote_p.ip)
        end
client.bs = "lb1"
client.action = "bs"

end
```

Пример 4. Подменить домен «ru» на домен «local» (например, если запрос идет к app.domain.ru, то в сторону Реального Сервера должен прийти app.domain.local). В начале пути запроса добавить «/external» и перенаправить запрос на Сервер Балансировки «lb1».

Код:

```
client.http_req.host =  client.http_req.host:match("(.*).ru") .. ".local"
client.http_req.path = "/external" .. client.http_req.path
client.bs = "lb1"
client.action = "bs"
```

Пример 5. Вставить сертификат (закодированный в формат `Base64`) в заголовок `X-Cert` при использовании mTLS.

Код:

```
client.http_req.header:field_set("X-Cert", client.tls:certificate)
```

Примеры для Rewrite (ответы)

Пример 1. Удаление заголовка с именем «ETag» из ответа Реального Сервера.

Код:

```
client.http_resp.header["ETag"] = nil
client.action = "pass"
```

Пример 2. Добавление заголовка со временем и именем «RESTIME».

Код:

```
client.http_resp.header["RESTIME"] = os.date()
client.action = "pass"
```

Пример 3. Добавление заголовка с Реальным Сервером (именем «RS») со значением IP-

адреса и порта Реального Сервера, от которого получен ответ.

Код:

```
client.http_resp.header["RS"] = tostring(client.rs)
client.action = "pass"
```

Пример 4. Генерация ответа с условием. Если Реальный Сервер ответил с кодом 500, то Termidesk Connect сгенерирует ответ с кодом 201.

Код:

```
if client.http_resp.status == 500 then
        client.respond.header["Connection"] = "close"
        client.respond.header["Content-Type"] = "text/html"
        client.respond.status = 201
        client.action = "respond"
end
```

Пример 5. Замена заголовка «Server» в ответе.

Код:

```
client.http_resp.header["Server"] = "MYServer"
```

Примеры для Responder

Пример 1. Если метод не GET или не HEAD, то сбросить соединение.

Код:

```
if (client.http_req.method ~= "GET" and client.http_req.method ~= "HEAD") then
    client.action = "drop"
end
```

Пример 2. Перенаправить соединение с HTTP на HTTPS.

Код:

```
client.respond.status = 302
client.respond.header["Location"] = "https://" .. client.http_req.host ..
client.http_req.path
client.respond.header["Connection"] = "close"
client.action = "respond"
```

Пример 3. Если запрос из сети 192.0.2.0/24, то ответить HTML-страницей и параметрами запроса.

Код:

```
if client.remote_p:is_network("192.0.2.0/24") then
    client.respond.header["Connection"] = "close"
    client.respond.header["Content-type"] = 'text/html
    client.respond.body = [[<html>
    <body>
    <meta charset="UTF-8">
    <h1>Lets goodbye!</h1>
    <p>Доступ запрещен</p>
    </body>
    </html>]] .. "IP: " .. client.remote_p.ip .. "\n TRY: " .. client.http_req.host
.. client.http_req.path .. "\n" ..      " Vserver: " .. client.local_p.ip .. ":" ..
client.local_p.port
client.action = 'respond'
```

Примеры для Ratio (F5)

Пример процентного разделение запросов:

- примерно 80% запросов попадет на Сервер Балансировки «lb01»;
- примерно 20% запросов попадет на Сервер Балансировки «lb02».

За каждым Сервером Балансировки может находиться свой пул серверов или один Реальный Сервер.

Код:

```
if math.random(1, 100) <= 20 then
  client.action = "bs"
  client.bs = "lb01"
else
  client.action = "bs"
  client.bs = "lb02"
end
```

Примеры для страницы обслуживания («Sorry page»)

Пример 1. Если Сервер Балансировки «lb01» находится в состоянии «Online», то он будет выбран для подключения. В противном случае будет показана страница sorry.html.

Файл `sorry.html` должен располагаться в директории `/var/lib/tdc/www`.

Код:

```
if client:bs_state("lb01") == "ONLINE" then
    client.bs = "lb01"
    client.action = "bs"
else client.respond.header["Content-Type"] = "text/html"
```

```
    client.action = 'respond'
    local filename = storage:sub_path('www', 'sorry.html')
    local file, err = io.open(filename)
    client.respond.body = file:read("*a")
    file:close()
end
```

Пример 2. Страница обслуживания:

- отправляется в интервале времени с 19:00 до 20:00, событие записывается в журнал;
- в остальное время принимаются и балансируются запросы.

Код:

```
if client:bs_state("lb01") == "ONLINE" then
    client.bs = "lb01"
    client.action = "bs"
else client.respond.header["Content-Type"] = "text/html"
    client.action = 'respond'
    local filename = storage:sub_path('www', 'sorry.html')
    local file, err = io.open(filename)
    client.respond.body = file:read("*a")
    file:close()
end
```

Примеры для AAA

Пример 1. Если пользователь не предоставлен (не получен из `cookie`, заголовка `Authorization` или клиентского сертификата), то аутентифицировать его согласно настройкам AAA-Профиля «prof01».

Код:

```
local function isempty(s)
return s == nil or s == ''
end
if (isempty(client.aaa:user_id()) and client.http_req.host:find("test2.ru")) then
    client.action = "auth"
    client.aaa.profile = "prof01"
end
```

Пример 2. Проверить пользователя на членство в группе «GR2», и, если он состоит в группе, выполнить балансировку на Сервер Балансировки «lb_http01». В противном случае ответить кодом `408`.

Код:

```
for i = 1, client.aaa:field_count("memberOf") do
```

```
    grp = client.aaa:field_get("memberOf", i)
    print("ALL GROUPS " .. grp)
    if grp:find("GR2") then
        client.bs = "lb_http01"
        client.action = "bs"
        print("OK " .. client.aaa:user_id() .. " ".. grp .. "Кол-во попыток: " ..
client.aaa:attempts() )
       break
    else
client.respond.status = 408
client.action = "respond"
print(grp)
    end
end
```

Примеры для Шлюза

Пример. При обновлении (заголовок `Upgrade`) выбрать Шлюз из настроенных точек подключений, для остального трафика выполнить балансировку на Сервер Балансировки «DISPATCHER LB».

Код:

```
if client.http_req.header["Upgrade"][1] then
client.gw = "GW-Name"
client.action = "gw"
else
client.bs = "DISPATCHER LB"
client.action = "bs"
end
```

UVEON

Телефон: +7 495 975 19 75
8 800 222 07 00

Электронная почта для связи: info@uveon.ru

Отдел продаж: info@astralinux.ru
Техподдержка: termidesk@astralinux.ru

Сайт: https://termidesk.ru/

Главный офис 119571, г. Москва,
Ленинский проспект, д. 119А