



РУКОВОДСТВО АДМИНИСТРАТОРА

СЛЕТ.10001-01 90 05

Версия 5.0. Выпуск от мая 2024 (с исправлениями
от 28.06.2024)

Настройка компонента «Шлюз»

ОГЛАВЛЕНИЕ

1 . ОБЩИЕ СВЕДЕНИЯ.....	3
1.1 . О документе.....	3
1.2 . Назначение компонента «Шлюз»	3
1.3 . Требования к программному и аппаратному обеспечению	3
1.4 . Типографские соглашения	3
2 . УСТАНОВКА И УДАЛЕНИЕ КОМПОНЕНТА	5
2.1 . Получение пакетов установки в ОС Astra Linux Special Edition	5
2.2 . Установка Шлюза.....	6
2.3 . Удаление Шлюза	7
2.4 . Обновление Шлюза	8
3 . НАСТРОЙКА КОМПОНЕНТА	9
3.1 . Общие сведения по настройке и функционированию Шлюза	9
3.2 . Регистрация компонента в системе управления и мониторинга Termidesk	9
3.3 . Параметры конфигурирования компонента	10
3.4 . Журналирование	15
4 . ПРИНЯТЫЕ ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ.....	16
5 . ПЕРЕЧЕНЬ СОКРАЩЕНИЙ	17

1 . ОБЩИЕ СВЕДЕНИЯ

1.1 . О документе

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

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

1.2 . Назначение компонента «Шлюз»

Компонент «Шлюз» (далее - Шлюз) входит в состав Termidesk. Шлюз отвечает за туннелирование протоколов доставки, использующих транспортный протокол TCP, обеспечивая отделение инфраструктуры виртуальных рабочих мест (ВРМ), находящихся во внутренней локальной сети, от внешних локальных сетей или глобальных сетей.

Шлюз является компонентом Termidesk и может устанавливаться как совместно с компонентами «Универсальный диспетчер», «Менеджер рабочих мест», так и отдельно при необходимости обеспечить распределенную конфигурацию.

1.3 . Требования к программному и аппаратному обеспечению

Для установки Шлюза минимальные аппаратные требования узла должны соответствовать следующим:

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

В среде функционирования Шлюза должна быть предварительно установлена операционная система (ОС) Astra Linux Special Edition версии 1.7 (и выше).

1.4 . Типографские соглашения

В настоящем документе 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.

2. УСТАНОВКА И УДАЛЕНИЕ КОМПОНЕНТА

2.1. Получение пакетов установки в ОС Astra Linux Special Edition

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

Получить iso-образ можно двумя способами:

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

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

- скопировать в домашний каталог пользователя образ диска `termidesk-⟨версия⟩.iso`;
- подключить образ диска к файловой системе в каталог `/mnt`:

```
~$ sudo mount -o loop termidesk-⟨версия⟩.iso /mnt
```

где:

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

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

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

где:

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

- отключить подключенный ранее образ диска от узла:

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

- установить пакет `lsb-release`:

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

где:

`-y` - ключ для пропуска подтверждения установки;

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

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

где:

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

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

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

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

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

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

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

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

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

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

При успешном выполнении всех шагов команда выведет содержимое ключа в формате Base64.

- обновить данные пакетного менеджера:

```
:~$ sudo apt update
```

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

2.2 . Установка Шлюза

Для установки Шлюза необходимо:

- остановить и отключить службу Шлюза termidesk-wsproxy, если он ранее использовался:

```
:~$ sudo systemctl stop termidesk-wsproxy
:~$ sudo systemctl disable termidesk-wsproxy
```

- выполнить установку `termidesk-gateway` из подключенного репозитория Termidesk (см. подраздел **Получение пакетов установки в ОС Astra Linux Special Edition**):

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

⚠ Для обслуживания API-запросов по состоянию Шлюза необходимо после установки Шлюза задать параметр `urlCheckToken`.
 Для регистрации Шлюза в системе управления и мониторинга Termidesk и отслеживания его актуального статуса из портала администратора Termidesk, необходимо после установки Шлюза задать параметры `coordinatorUrl`, `coordinatorUser`, `coordinatorPass` (см. подраздел **Параметры конфигурирования компонента**).

Зависимости пакета `termidesk-gateway`:

- `libc6 (>= 2.14)`;
- `libgcc1 (>= 1:3.3.1)`;
- `libssl1.1 (>= 1.1.0)`;
- `libstdc++6 (>= 5.2)`.

Проверка состояния службы `termidesk-gateway` выполняется командой:

```
:~$ sudo systemctl status termidesk-gateway
```

Строка «Active» отображает состояние сервиса, где статус «active (running)» свидетельствует об успешном запуске `termidesk-gateway`.

Для просмотра установленной версии Шлюза `termidesk-gateway` нужно выполнить:

```
:~$ termidesk-gateway -v
```

⚠ Начиная с Termidesk версии 5.0 изменился API-запрос валидации токена Шлюза `termidesk-gateway` для подключения к компоненту «Универсальный диспетчер».
 При обновлении компонента «Универсальный диспетчер» и/или Шлюза `termidesk-gateway` на версию из состава программного комплекса 5.X нужно вручную обновить параметр `urlCheckToken` (см. подраздел **Параметры конфигурирования компонента**) на значение `https://<IP-адрес>/api/wsproxy/v1.1/verify` для корректной работы переключения между протоколами TCP и UDP.

2.3 . Удаление Шлюза

Для удаления Шлюза `termidesk-gateway` нужно:

- удалить без подтверждения `termidesk-gateway`:

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

- очистить оставшиеся зависимости и конфигурации:

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

2.4 . Обновление Шлюза

Обновление Шлюза `termidesk-gateway` выполняется процедурой установки новой версии. При обновлении `termidesk-gateway` файл `/lib/systemd/system/termidesk-gateway.service` будет перезаписан. Конфигурационный файл Шлюза `/etc/termidesk/termidesk-gateway.conf` будет заменен файлом `/etc/termidesk/gateway.yaml`.

⚠ Начиная с Termidesk версии 5.0 изменился API-запрос валидации токена Шлюза `termidesk-gateway` для подключения к компоненту «Универсальный диспетчер». При обновлении компонента «Универсальный диспетчер» и/или Шлюза `termidesk-gateway` на версию из состава программного комплекса 5.X нужно вручную обновить параметр `urlCheckToken` (см. подраздел **Параметры конфигурирования компонента**) на значение `https://<IP-адрес>/api/wsproxy/v1.1/verify` для корректной работы переключения между протоколами TCP и UDP.

При обновлении распределенной конфигурации необходимо учесть, что если ранее на узлах со Шлюзом был установлен `termidesk-gateway`, необходимо сначала обновить эти узлы, и только потом - узлы компонента «Универсальный диспетчер» и узлы компонента «Менеджер рабочих мест».

3. НАСТРОЙКА КОМПОНЕНТА

3.1 . Общие сведения по настройке и функционированию Шлюза

Для проверки состояния службы Шлюза используется команда:

```
1 :~$ sudo systemctl status termidesk-gateway
```

Шлюз может быть вынесен в демилитаризованную зону сетевой инфраструктуры предприятия. Для работы подключения пользователей к ВРМ через Шлюз необходимо обеспечить доступность следующих сетевых портов:

- 80 (HTTP);
- 443 (HTTPS).

По умолчанию Шлюз прослушивает порт TCP:5099 на интерфейсе localhost (127.0.0.1). Для изменения порта прослушивания службы необходимо внести изменение в конфигурационный файл `termidesk-gateway.yaml` (см. подраздел **Параметры конфигурирования компонента**).

На новой установке Шлюза `termidesk-gateway` механизм взаимодействия веб-сервера со Шлюзом удален.

Однако если ранее на узле был установлен Шлюз `termidesk-wsproxy` и конфигурация веб-сервера была сохранена, то взаимодействие выглядит следующим образом:

- запросы на подключения принимает веб-сервер `apache` по портам 80 или 443;
- веб-сервер `apache` перенаправляет запросы Шлюзу на указанные IP-адрес на порт;
- далее Шлюз направляет запросы либо на поставщик ресурсов, либо в виртуальную машину.

3.2 . Регистрация компонента в системе управления и мониторинга Termidesk

Для централизованного управления объектами инфраструктуры в веб-интерфейсе Termidesk реализована система управления и мониторинга состояния компонентов.

Регистрация компонента в системе происходит через подключение к серверу RabbitMQ, который хранит информацию об узле компонента и передает ее компоненту «Менеджер рабочих мест» (`termidesk-taskman`). Для регистрации Шлюза в системе необходимо задать обязательные параметры `coordinatorUrl`, `coordinatorUser`, `coordinatorPass` (см. подраздел **Параметры конфигурирования компонента**).

Если параметр `coordinatorUrl` задан, то после запуска Шлюз инициирует подключение к серверу RabbitMQ. Все запросы к серверу RabbitMQ ожидают ответа в течение времени, заданного в параметре `coordinatorTimeout`, по истечении которого фиксируется ошибка подключения. При успешном подключении Шлюз передает свой статус с URL проверки состояния (`healthcheck`) в

формате JSON и периодически обновляет его. Таймаут обновления статуса задается в параметре `coordinatorRefreshTime`.

3.3 . Параметры конфигурирования компонента

Параметры конфигурирования Шлюза задаются в конфигурационном файле `/etc/termidesk/gateway.yaml`.

⚠ При обновлении компонента «Универсальный диспетчер» и/или Шлюза `termidesk-gateway` на версию из состава программного комплекса 5.X вместо значения `https://127.0.0.1/api/wsproxy/v1/verify` для параметра `urlCheckToken` нужно использовать значение `https://127.0.0.1/api/wsproxy/v1.1/verify`.

ℹ Реализована обратная совместимость с прошлым файлом конфигурации: файл `/etc/termidesk/gateway.yaml` реализует подстановку значений параметров из переменных окружения, которые были заданы в `/etc/termidesk/termidesk-gateway.conf`.

Для задания параметров конфигурирования Шлюза из файла необходимо:

- отредактировать файл `gateway.yaml`, указав необходимые значения для параметров. Пример приведен в файле `/etc/termidesk/gateway.yaml.example`:

```

1  # Объявляем переменную для SSL (потом ее используем в качестве ссылки)
2  _val0: &sslprof0                # Шаблон для SSL
3  key: "/etc/key"                 # SSL Key file name (default "")
4  cert: "/etc/cert"              # SSL Cert file name (default "")
5  ca: "/etc/ca"                  # SSL CA file name (default "")
6  passphrase: "phrase"           # SSL PassPhrase (default "")
7  dhparams: "/etc/dhparams"      # SSL DhParams file name (default
  "")
8  ciphers: "ciphers"             # SSL Cipher (default "")
9
10 # Gateway серверы (это список, по умолчанию он пустой)
11 gwserver:
12 - listen: "127.0.0.1:9200"     # IP + port, для IPv6
  "[::1]:80" (default ~)
13   ssl: *sslprof0               # SSL настройки (default {})
14   websocket:                   # Настройка WebSocket
15     pingtimeout: 35            # Время пинга «Клиента» в секундах,
  диапазон [0, 100000], если 0, то пинги отключены (default 30)
16   checktoken:                  # Настройка проверки токенов
17     url: "http://127.0.0.1:80/api/verify" # URL для проверки (default "http://
  127.0.0.1:8080/api/wsproxy/v1/verify")
18   # ... Другие шлюзы
19
20 # MGT сервер
21 mgtserver:
22   listen: "127.0.0.1:8181"     # IP + port, для IPv6
  "[::1]:80" (default ~)
    
```

```

23   path: /api/health                               # Path для запроса информации
      (default "/api/health")
24   token: "abcde12345"                             # Токен валидации (default "")
25   ssl: *sslprof0                                  # SSL настройки (default {})
26
27   # RabbitMQ клиент
28   rabbitmq:
29     url: "amqp://USER:PASS@localhost/"            # URL, где USER и PASS - это
      необязательные поля (default ~)
30     user: "user"                                  # user - пользователь, если задан,
      то изменяет url.USER (default "")
31     pass: "pass"                                  # RabbitMQ pass - пароль, если
      задан, то изменяет url.PASS (default "")
32     timeout: 40                                   # RabbitMQ timeout(sec), диапазон
      [1, 60] (default 10)
33     refreshtime: 80                               # Период (sec) обновления информация
      на сервере RabbitMQ, диапазон [10, 100'000] (default 60)
34     single: true                                  # Если true - то данные передаются
      до первого подтверждения,
35                                                    # если false - то данные передаются
      по циклу (бесконечно) (default false)
36     exchange: "exchange"                         # RabbitMQ exchange (default "")
37     routingkey: "routingkey"                     # RabbitMQ routing key (default
      "termidesk_appnode")
38
39   loglevel:                                       # Логирование сообщений
40     info: true                                    # Уровень INFO (default true)
41     debug: true                                  # Уровень DEBUG (default false)
    
```

- выполнить перезагрузку службы:

```

:~$ systemctl restart termidesk-gateway.service
    
```

Для получения информации по доступным аргументам командной строки нужно выполнить команду:

```

:~$ termidesk-gateway --help
    
```

Параметры, используемые в файле `/etc/termidesk/gateway.yaml` после установки приведены в таблице (см. Таблица 1).

Задание параметров может производиться и переменными окружения в формате: `${ENV}` или `{ENV:DEFAULT}`, где `DEFAULT` - значение для подстановки, если указанная переменная окружения не найдена.

⚠ Если переменная окружения существует с пустым значением, то пустое значение будет подставлено в `/etc/termidesk/gateway.yaml` вместо переменной, а не значение по умолчанию.

Например, если нужно задать через переменные окружения адрес IP_ADDR и порт прослушивания службы PORT, то в конфигурационном файле /etc/termidesk/ gateway.yaml можно изменить значение следующим образом:

```
1 listen: ${IP_ADDR}:${PORT:8000}
```

Либо эти значения можно задать напрямую в файле, без использования переменных окружения, тогда запись изменится к виду:

```
1 listen: "127.0.0.1:8181"
```

Таблица 1 – Доступные переменные Шлюза

Параметр	Описание
Секция val0: &sslprof0	
key: \${sslKey}	Путь к файлу ключа для соединения SSL. Параметр не имеет значения по умолчанию. При использовании сертификатов и ключей на файлы .key и .pem необходимо выдать права на чтение командой chmod 644: <pre>~\$ sudo chmod 644 /etc/termidesk/ssl-cert-snakeoil.key ~\$ sudo chmod 644 /etc/termidesk/ssl-cert-snakeoil.pem</pre>
cert: \${sslCert}	Путь к файлу сертификата для соединения SSL. Параметр не имеет значения по умолчанию
ca: \${sslCa}	Путь к корневому сертификату, на котором выпущен сертификат, указанный в параметре cert. Параметр не имеет значения по умолчанию. Используется для проверки подлинности указанного в параметре cert сертификата
passphrase: \${sslPassPhrase}	Парольная фраза для соединения SSL. Параметр не имеет значения по умолчанию
dhparams: \${sslDhParams}	Путь к файлу с ключами Диффи-Хеллмана для соединения SSL. Параметр не имеет значения по умолчанию
ciphers: \${sslCiphers}	Используемый алгоритм преобразований для соединения SSL. Параметр не имеет значения по умолчанию
Секция gwserver:	
listen: \${wsServerIP:0.0.0.0}:\${wsServerPort:5099}	Назначение параметров узла Шлюза: <ul style="list-style-type: none"> ▪ wsServerIP - адрес прослушивания входящих подключений; ▪ wsServerPort - порт прослушивания службы для входящих подключений. Значения по умолчанию: <ul style="list-style-type: none"> ▪ wsServerIP:0.0.0.0; ▪ wsServerPort:5099. Для протокола SSL нужно использовать параметры wssServerIP и wssServerPort и значение 10000 для номера порта: listen:\${wssServerIP:0.0.0.0}:\${wssServerPort:10000}

Параметр	Описание
websocket: pingtimeout: \${wsIdleTimeout:30}	Таймаут соединения (в секундах) к компоненту «Клиент» Termidesk. Может принимать значения в интервале с 8 до 960, а также 0 (отключен). Значение по умолчанию: 30
checktoken: url: \${urlCheckToken:https://127.0.0.1/api/wsproxy/v1/verify}	Назначение IP-адреса или FQDN узла компонента «Универсальный диспетчер» с ролью «Портал пользователя» для обслуживания API-запросов по состоянию Шлюза. При обновлении компонента «Универсальный диспетчер» и/или Шлюза termidesk-gateway на версию из состава программного комплекса 5.X следует изменить значения параметра в связи с переходом на новую версию API: urlCheckToken:https://disp.termidesk.local/api/wsproxy/v1.1/verify, где: disp.termidesk.local - FQDN узла компонента «Универсальный диспетчер» с ролью «Портал пользователя». При комплексной установке Termidesk значение disp.termidesk.local может быть заменено на 127.0.0.1. Если программный комплекс Termidesk установлен в распределенном варианте, следует изменить значение disp.termidesk.local на внешний IP-адрес или FQDN узла балансировщика нагрузки. Значение по умолчанию: urlCheckToken: https://127.0.0.1/api/wsproxy/v1/verify
Секция mgtserver:	
listen: \${mgtServerIP:127.0.0.1}:\${mgtServerPort:8102}	Задание параметров для подключения к API Шлюза: <ul style="list-style-type: none"> ▪ mgtServerIP - IP-адрес или FQDN доступа к API Шлюза. Код 200 в ответе на API-запрос свидетельствует о работоспособности Шлюза. При распределенной установке следует задать значение 0.0.0.0 для активации приема запросов с внешних систем; ▪ mgtServerPort - порт доступа к API Шлюза. Значения по умолчанию: <ul style="list-style-type: none"> ▪ mgtServerIP:127.0.0.1; ▪ mgtServerPort:8102. Запрос со значениями по умолчанию успешно выполнится только с узла, на котором установлен Шлюз: <pre style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">~\$ wget http://localhost:8102/api/health</pre>
path: \${healthCheckURL:/api/health}	Адрес для запросов проверки состояния Шлюза. Значение по умолчанию: healthCheckURL:/api/health
token: \${healthCheckAccessKey}	Задание ключа доступа для аутентификации запросов к API Шлюза. Параметр не имеет значения по умолчанию. При задании значения ключа следует руководствоваться правилом, что: <ul style="list-style-type: none"> ▪ размер ключа должен составлять от 0 до 64 символа; ▪ должны использоваться символы в шестнадцатеричной системе (0-9, a-f)
ssl: *sslprof0	Настройка SSL

Параметр	Описание
Секция rabbitmq:	
url: \${coordinatorUrl}	<p>URL, содержащий IP-адрес (или FQDN) и порт (по умолчанию 5672) для подключения к серверу RabbitMQ, а также очередь - termidesk. Параметр не имеет значения по умолчанию.</p> <p>Используемый формат: amqp(s)://USER:PASS@<IP-адрес>:<порт>/termidesk.</p> <p>USER и PASS необязательные параметры, они могут быть заданы в параметрах coordinatorPass и coordinatorPass.</p> <p>Пример: coordinatorUrl:amqp://termidesk:ksedimret@disp.termidesk.local:5672/termidesk.</p> <p>На данный момент поддерживается задание только одного URL для подключения к серверу RabbitMQ.</p> <p>Подключение может использоваться как незащищенное (в этом случае указывается amqp), так и защищенное (amqps).</p> <p>В случае, если RabbitMQ установлен на том же узле (локально), нужно указать:</p> <p>coordinatorUrl: amqp://127.0.0.1:5672/termidesk</p>
user: \${coordinatorUser}	<p>Имя пользователя для подключения к серверу RabbitMQ. Параметр не имеет значения по умолчанию.</p> <p>Если параметр задан, то он меняет значение USER в параметре coordinatorUrl.</p> <p>При стандартной установке компонента «Универсальный диспетчер» строго по документации в части настроек файла /etc/rabbitmq/definitions.json используется coordinatorUser:termidesk</p>
pass: \${coordinatorPass}	<p>Пароль для подключения серверу RabbitMQ. Параметр не имеет значения по умолчанию.</p> <p>При стандартной установке компонента «Универсальный диспетчер» строго по документации в части настроек файла /etc/rabbitmq/definitions.json используется coordinatorPass:ksedimret</p>
timeout: \${coordinatorTimeout}	<p>Интервал (в секундах) ожидания ответа от сервера RabbitMQ. Параметр не имеет значения по умолчанию</p>
refreshTime: \${coordinatorRefreshTime}	<p>Интервал (в секундах) обновления (перепубликации) регистрационной информации (URL и другие данные) Шлюза. Параметр не имеет значения по умолчанию.</p> <p>Значение «0» отключает автоматическую отправку состояния Шлюза на сервер RabbitMQ</p>
single: \${coordinatorSingle:true}	<p>Способ передачи данных.</p> <p>Параметр принимает значения:</p> <ul style="list-style-type: none"> ▪ true - данные передаются до первого подтверждения; ▪ false - данные передаются по циклу (бесконечно). <p>Значение по умолчанию: coordinatorSingle:true</p>
exchange: \${coordinatorExchange}	<p>Координатор маршрутизации сообщений, определенный в RabbitMQ. Отвечает за маршрутизацию сообщений в разные очереди. Параметр не имеет значения по умолчанию</p>
routingkey: \${coordinatorRoutingKey}	<p>Ключ маршрутизации RabbitMQ, используется для маршрутизации задачи в очереди. Параметр не имеет значения по умолчанию</p>
Секция loglevel:	
info: \${logInfo:true}	Активация режима журналирования уровня INFO

Параметр	Описание
debug: \${logDebug:false}	Активация подробного режима журналирования уровня DEBUG

Пример команды проверки состояния компонента через утилиту `curl` для Шлюза `termidesk-gateway`:

```

:~$ curl -v -s -X 'GET' "${HOSTNAME}:8102/api/health" -H 'accept: application/json' -H
"Authorization: Token ${HEALTH_CHECK_ACCESS_KEY}" --fail -w "\n%{http_code}\n"
    
```

3.4 . Журналирование

Для просмотра журнала Шлюза `termidesk-gateway` можно выполнить:

```

:~$ sudo journalctl -f -u termidesk-gateway.service
    
```

или:

```

:~$ sudo less /var/log/syslog
    
```

4. ПРИНЯТЫЕ ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ

Термин	Определение
Компонент «Агент виртуального рабочего места»	Компонент Termidesk. Устанавливается в гостевую ОС при подготовке базового ВРМ. Выполняет взаимодействие с «Универсальным диспетчером», конфигурирует ВРМ, фиксирует действия пользователя, реализует передачу управляющих сообщений
Балансировщик нагрузки	Самостоятельный компонент, отвечающий за распределение нагрузки на множество «Универсальных диспетчеров» и «Шлюзов»
Виртуальное рабочее место	Также: ВРМ. Развернутая на ВМ ОС с установленным «Агентом виртуального рабочего места» и необходимым прикладным ПО. Подключение к ВРМ происходит при помощи протоколов удаленного доступа, чаще всего называемыми протоколами доставки
Гостевая ОС	ОС, функционирующая на ВМ
Компонент «Менеджер рабочих мест»	Компонент Termidesk. Отделяемый компонент программного комплекса, отвечающий за взаимодействие с поставщиком ресурсов и управления жизненным циклом ВРМ, включая создание, настройку, запуск, отключение и удаление. Является обработчиком фоновых задач. Устанавливается из пакета <code>termidesk-vdi</code> . Наименование службы после установки: <code>termidesk-taskman.service</code>
Поставщик ресурсов	ОС, платформа виртуализации или терминальный сервер (MS RDS/STAL), предоставляющие вычислительные мощности, ресурсы хранения данных, а также сетевые ресурсы для размещения фондов ВРМ
Протокол доставки	Поддерживаемый в Termidesk протокол удаленного доступа к ВРМ
Компонент «Универсальный диспетчер»	Компонент Termidesk. Отделяемый компонент программного комплекса, отвечающий за идентификацию пользователей, назначение им ВРМ и контроля доставки ВРМ. Устанавливается из пакета <code>termidesk-vdi</code> . Наименование службы после установки: <code>termidesk-vdi.service</code>
Компонент «Шлюз»	Компонент Termidesk. Самостоятельный компонент, отвечающий за туннелирование протоколов доставки, использующих транспортный протокол TCP. В более старой реализации устанавливается из пакета <code>termidesk-vdi</code> . Наименование службы после установки: <code>termidesk-wsproxy.service</code> . В новой реализации устанавливается из пакета <code>termidesk-gateway</code> , поддержка старой реализации удалена, начиная с Termidesk версии 5.0. Наименование службы после установки: <code>termidesk-gateway.service</code> .
Ключ	Применяется в контексте файла, не опции в команде. Последовательность псевдослучайных чисел, сгенерированная особым образом
Сертификат	Артефакт, содержащий информацию о владельце ключа и подтверждающий принадлежность ключа владельцу

5. ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

Сокращение	Пояснение
ВРМ	Виртуальное рабочее место
API	Application Programming Interface (интерфейс прикладного программирования)
HTTP	HyperText Transfer Protocol (протокол передачи гипертекста)
HTTPS	Hypertext Transfer Protocol Secure (расширение протокола HTTP для поддержки шифрования)
IP	Internet Protocol (межсетевой протокол)
TCP	Transmission Control Protocol (протокол управления передачей данных)
Termidesk	Программный комплекс «Диспетчер подключений виртуальных рабочих мест Termidesk»



© ООО «УВЕОН»

119571, г. Москва, Ленинский проспект,
д. 119А, помещ. 9Н
<https://termidesk.ru/>
Телефон: +7 (495) 975-1-975

Общий e-mail: info@uveon.ru
Отдел продаж: sales@uveon.ru
Техническая поддержка: support@uveon.ru