УТВЕРЖДЕНО
RU.ЛСАФ.00013-01 34 01–ЛУ

Программный комплекс «LirVPN»

Руководство оператора

RU.ЛСАФ.00013-01 34 01

Листов 64

Инв. № подл.	Подп. и дата	Взам инв. №	Инв. № дубл.	Подп. и дата

2013

АННОТАЦИЯ

Целью данного документа является ознакомление оператора с архитектурой ПК «LirVPN» и предоставление полной информации по его использованию.

СОДЕРЖАНИЕ

1 Список сокращений

АС - автоматизированная система.

АРМ - автоматизированное рабочее место.

ЗВКС - защищенная виртуальная корпоративная сеть.

ЛВС - локальная вычислительная сеть.

ОС - операционная система.

РД - руководящий документ.

МЭ - межсетевой экран.

ПК - программный комплекс.

СКЗИ - средство криптографической защиты информации.

НСД - несанкционированный доступ.

2 Общие сведения

ПК «LirVPN» предназначен для создания защищенных виртуальных корпоративных сетей (ЗВКС) путем объединения территориально-удаленных локальных вычислительных сетей/автоматизированных систем (далее сегмент ЗВКС) с использованием в качестве транспортной среды публичных сетей (включая сеть Интернет, ведомственные сети), поддерживающих IP протокол, в том числе для автоматизированных систем органов государственного управления и организаций Российской Федерации, систем управления транспортом, связью, энергетикой и др.

ПК «LirVPN» создает ЗВКС на базе протоколов SSLv3/TLSv1 с использованием российских криптографических алгоритмов. В качестве криптографического ядра ПК «LirVPN» использует сертифицированное ФСБ России по классам КС1 и КС2 ПБЗИ «СКЗИ «ЛИРССЛ» (сертификаты соответствия №СФ/111-1978 и №СФ/111-1979).

ПК «LirVPN» функционирует под управлением операционных систем MS Windows, Linux.

3 Функциональное назначение

ПК «LirVPN» обеспечивает:

- построение ЗВКС различных типов:
  - ЗВКС канального уровня ("bridging mode");
  - ЗВКС сетевого уровня ("routed mode");
- использование в качестве транспорта протокол TCP или UDP;
- поддержку балансировки нагрузки;
- возможность работы в режиме VPN-концентратора (т.е. узла, который может принимать множество VPN-соединений);
- возможность передачи сетевых настроек от сервера клиентам;
- возможность использования библиотеки компрессии LZO, для сжатия потока данных;
- аутентификацию узлов ЗВКС на базе цифровых сертификатов или на базе разделяемого секрета;
- защиту передаваемой между сегментами ЗВКС информации от НСД;

4 Принципы работы

4.1 Понятие ЗВКС

Защищенная виртуальная корпоративная сеть (ЗВКС) - это логическая сеть («наложенная сеть»), создаваемая поверх другой сети, например, Интернет. Несмотря на то, что коммуникации осуществляются по публичным сетям, с использованием небезопасных протоколов, за счёт шифрования создаются закрытые от посторонних каналы обмена информацией. ЗВКС позволяет объединить, например, несколько офисов организации в единую сеть с использованием для связи между ними неподконтрольных каналов.

Сегментами ЗВКС могут быть как ЛВС, так и отдельные АРМы.

4.2 Сетевая архитектура

ПК «LirVPN» использует универсальный TUN/TAP драйвер для организации сетевого взаимодействия сегментов ЗВКС.

Универсальный TUN/TAP драйвер был разработан для предоставления ядру ОС поддержки туннелирования IP/Ethernet трафика. После установки этого драйвера, в системе появляется виртуальный сетевой интерфейс. Любое приложение, которое может работать с обычным сетевым интерфейсом, может работать и с этим виртуальным интерфейсом.

TUN интерфейс может быть использован в качестве виртуального point-to-point интерфейса, как например модемное или DSL соединение. Такой режим называется routed mode, потому что маршруты устанавливаются до другой стороны туннеля.

TAP интерфейс может использоваться в качестве виртуального Ethernet адаптера. Это позволяет ПК «LirVPN» прослушивать интерфейс для перехватывата Ethernet-пакетов, что невозможно с TUN интерфейсом. Такой режим называется bridging mode – потому что, в этом режиме сети объединяются так, как будто они объединены через аппаратный коммутатор (мост).

Приложение посылает данные через виртуальное соединение TUN/TAP драйвера. ПК «LirVPN» прослушивает TUN/TAP интерфейсы, перехватывает трафик и шифрует его с помощью СКЗИ «LirSSL». Далее данные упаковываются и отправляются на другую сторону туннеля, где ПК «LirVPN» получает их, расшифровывает, и передаёт на виртуальное сетевое соединение, где приложение ожидает данные.

Данные передаются с помощью протокола UDP или TCP. Протокол UDP предпочтительней, но TCP может быть полезен в ряде случаев.

5 Управление и настройка ПК «LirVPN»

5.1 Управление

Исполняемый файл ПК «LirVPN» находится:

- в ОС семейства MS Windows - `<Каталог установки>\lirvpn.exe`
- в ОС семейства Linux - `/usr/local/sbin/lirvpn`

5.1.1 ОС семейства Unix

Запуск

Запуск производится с помощью команды:

```
#lirvpn --config <конфигурационный_файл>
```

где:

`<конфигурационный_файл>` - путь до файла конфигурации ПК «LirVPN».

Останов

Для останова необходимо передать процессу ПК «LirVPN» сигнал **SIGINT** или **SIGTERM**. Это можно сделать 2-мя способами:

- командой kill (`#kill -s SIGTERM <номер_процесса_lirvpn>`)
- ctrl+c в консоли на которой был запущен ПК «LirVPN»

Перезапуск

Для перезапуска необходимо передать процессу ПК «LirVPN» сигнал **SIGHUP** или **SIGUSR1** с помощью команд:

```
#kill -s SIGHUP <номер_процесса_lirvpn>
```

Данная команда заставляет ПК «LirVPN» закрыть все TUN/TAP соединения и сетевые соединения, используемые им, перезапуститься, перечитать все файлы конфигурации, и открыть заново TUN/TAP соединения и сетевые соединения.

```
#kill -s SIGUSR <номер_процесса_lirvpn>
```

Действует также как и **SIGHUP**, но:

- не перечитывает файлы конфигурации;
- не закрывает и не открывает заново TUN/TAP соединения (при заданной опции **persist-tun**);
- не перечитывает файлы ключей и сертификатов (при заданной опции **persist-key**);
- сохраняет локальный IP адрес/порт (при заданной опции **persist-local-ip**);
- сохраняет самый последний авторизованный IP адрес/порт удаленного компьютера (при заданной опции **persist-remote-ip**).

Этот сигнал также может быть вызван изнутри ПК «LirVPN» по таймауту соединения (опция **ping-restart**).

Сигнал **SIGHUP**, в комбинации с опцией **persist-remote-ip**, может быть автоматически вызван, когда параметры сетевого интерфейса компьютера меняются, например, когда компьютер получает новый IP адрес по DHCP. Обратитесь к описанию опции **ipchange** для дополнительной информации.

Проверка состояния

Для проверки состояния необходимо передать процессу ПК «LirVPN» сигнал **SIGUSR2** с помощью команды:

```
#kill -s SIGUSR2 <номер_процесса_lirvpn>
```

Заставляет ПК «LirVPN» отобразить текущую статистику соединений (на устройство stdout, или системный файл журнала, если используется опция **daemon**).

5.1.2 ОС семейства Windows

Запуск

Запуск производится с помощью команды:

```
lirvpn.exe --config <конфигурационный_файл>
```

где:

`<конфигурационный_файл>` - путь до файла конфигурации ПК «LirVPN».

Останов

Для остановки ПК «LirVPN» запущенного из командной строки необходимо нажать клавишу «**F4**» в окне с запущенным ПК «LirVPN».

Перезапуск

Для перезапуска ПК «LirVPN» запущенного из командной строки необходимо нажать клавишу «**F3**» (**SIGHUP**) или «**F1**» (**SIGUSR1**) в окне с запущенным ПК «LirVPN».

«**F3**» - **SIGHUP** Данная команда заставляет ПК «LirVPN» закрыть все TUN/TAP соединения и сетевые соединения, используемые им, перезапуститься, перечитать все файлы конфигурации, и открыть заново TUN/TAP соединения и сетевые соединения.

«**F1**» - **SIGUSR1** Действует также как и **SIGHUP**, но:

- не перечитывает файлы конфигурации;
- не закрывает и не открывает заново TUN/TAP соединения (при заданной опции **persist-tun**);
- не перечитывает файлы ключей и сертификатов (при заданной опции **persist-key**);
- сохраняет локальный IP адрес/порт (при заданной опции **persist-local-ip**);
- сохраняет самый последний авторизованный IP адрес/порт удаленного компьютера (при заданной опции **persist-remote-ip**).

Этот сигнал также может быть вызван изнутри ПК «LirVPN» по таймауту соединения (опция **ping-restart**).

Сигнал **SIGHUP**, в комбинации с опцией **persist-remote-ip**, может быть автоматически вызван, когда параметры сетевого интерфейса компьютера меняются, например когда компьютер получает новый IP адрес по DHCP. Обратитесь к описанию опции **ipchange** для дополнительной информации

Проверка состояния
Для проверки состояния необходимо нажать клавишу «**F2**» в окне с запущенным ПК «LirVPN». Это заставляет ПК «LirVPN» отобразить текущую статистику соединений.

Также проверить текущее состояние ПК «LirVPN», можно просмотрев файлы журналов (имя файла журнала задается опцией **log**).

5.2 Настройка

Настройка ПК «LirVPN» производится путем создания простого текстового файла. Для каждого сервера и каждого клиента необходим свой файл конфигурации. Синтаксис файла конфигурации одинаков для любой поддерживаемой операционной системы.

Далее приведен пример простого файла конфигурации:

Листинг № 1: Пример простого файла конфигурации ПК «LirVPN»

```
1 dev tun
2 port 1194
3 comp-lzo
4 keepalive 120 600
5 log-append /var/log/lirvpn/log.log
6 client
7 pkcs12 /etc/lirvpn/sample.p12
8 remote 179.121.13.10
```

6 Конфигурационные опции

6.1 Опции туннеля

mode <m>
Устанавливает режим, в котором будет работать ПК «LirVPN». По умолчанию, ПК «LirVPN» использует режим точка-точка (**«p2p»**). Режим сервер (**«server»**) устанавливает режим клиент-сервер. Клиентов может быть несколько.

local <host>
Имя узла или IP адрес компьютера. Если опция задана, ПК «LirVPN» будет работать только через интерфейс с указанным адресом. Если не задана, ПК «LirVPN» работает со всеми сетевыми интерфейсами.

remote <host> [<port>]
Имя удаленного узла или IP адрес. На клиенте, может быть задано несколько опций **remote**, каждая опция в данном случае указывает на свой сервер ПК «LirVPN».
Клиент ПК «LirVPN» будет пытаться соединиться с сервером
<host>:<port> в порядке следования опций **remote**.
Клиент будет переходить к следующему серверу в случае неудачи связи с предыдущим сервером. В любой момент времени клиент ПК «LirVPN» может быть подключен только к одному серверу.
Так как протокол UDP не требует соединения с удаленным модулем UDP («бессвязный» протокол), в случае его использования, условия неудачи связи задаются опциями **ping** и **ping-restart**.
В случае если ваша конфигурация отвечает следующим 3-м пунктам:

- Если вы используете несколько опций **remote**;
- Если вы сбрасываете привилегии суперпользователя на клиенте с помощью опций **user** и/или **group**;
- Если клиент запущен на ОС, отличной от Windows.

если клиенту понадобиться переключиться на другой сервер, и сервер посылает клиенту другие настройки TUN/TAP или настройки маршрутизации, клиенту может не доставать привилегий на закрытие и открытие TUN/TAP интерфейса. В этом случае возможна остановка ПК «LirVPN» на компьютере клиента с сообщением об ошибке.
Если опция **remote** не задана, ПК «LirVPN» будет ждать соединения от любого IP адреса, но не будет реагировать на те пакеты, которые не прошли все процедуры аутентификации. Эти процедуры также применяется ко всем возможным узлам, даже к тем которые считаются известными и доверенными (IP адрес отправителя в UDP пакете очень легко подделать).
Когда опция **remote** используется в режиме TCP, она работает как фильтр, отсеивая соединения от любого узла который не совпадает с **<host>**.
Если **<host>** это DNS имя, которое разрешается в несколько IP адресов, один будет выбран случайно, что предоставляет возможность простой балансировки нагрузки и отказоустойчивости.

remote-random

Если задано несколько опций **remote**, случайный порядок следования серверов в списке предоставляет простую возможность для простой балансировки нагрузки.

proto <p>

Опция задает использование протокола <**p**> для коммуникации с удаленным узлом.

<**p**> может принимать значения **«udp»**, **«tcp-client»**, или **«tcp-server»**.

Если опция **proto** не задана, по умолчанию используется **«udp»**.

Для использования UDP, опция **proto udp** должна быть задана на каждом узле.

Для использования TCP, один из узлов должен использовать опцию **proto tcp-server**, а все остальные **proto tcp-client**. Узел, запущенный с опцией **proto tcp-server** будет бесконечно ждать входящего соединения. Узел, запущенный с опцией **proto tcp-client** попытается установить соединение с удаленным узлом, и если это невозможно, перейдет в режим сна на 5 секунд (это время регулируется опцией **connect-retry**) после чего повторит попытку. Как TCP сервер, так и TCP клиент формируют сигнал **SIGUSR1**, если одна из сторон прервала соединение.

ПК «LirVPN» спроектирован для оптимальной работы через UDP, но поддержка TCP предоставляется для ситуаций, в которых UDP не может использоваться. В сравнении с UDP, TCP обычно менее эффективен и менее надежен, когда используется в ненадежных или перегруженных сетях.

connect-retry <n>

Если задана опция **proto tcp-client**, <**n**> - время в секундах, которое должно пройти между попытками соединения (по умолчанию, 5 секунд).

http-proxy <server> <port> [<authfile>] [<auth-method>]

Установить соединение с удаленным узлом через HTTP прокси с адресом <**server**> и портом <**port**>. Если HTTP прокси требует аутентификации, параметр <**authfile**>, это имя файла, который содержит имя пользователя и пароль в 2 строки, или **«stdin»** для их ввода с консоли.

Параметр <**auth-method**> должен содержать **«none»**, **«basic»**, или **«ntlm»**.

http-proxy-retry

Постоянно повторять попытки соединения в случае ошибок HTTP прокси. Если возникает ошибка HTTP прокси, ПК «LirVPN» передается сигнал **SIGUSR1**.

http-proxy-timeout <n>

Установка времени ожидания ответа от прокси сервера в <**n**> секунд (по-умолчанию - 5).

http-proxy-option <type> [<parm>]

Установка дополнительных опций передаваемых HTTP прокси. Задайте опцию несколько раз для нескольких значений.

VERSION version - Установить номер версии HTTP - **version** (по умолчанию - 1.0).

AGENT user-agent – Установить строку HTTP "User-Agent" - **user-agent**.

socks-proxy <server> [<port>]

Установить соединение с удаленным узлом через Socks5 прокси с адресом <**server**> и портом <**port**> (по умолчанию - 1080).

socks-proxy-retry

Постоянно повторять попытки соединения в случае ошибок Socks прокси. Если возникает ошибка HTTP прокси, ПК «LirVPN» передается сигнал **SIGUSR1**.

resolv-retry <n>

Если разрешение имени узла указанного в опции **remote** в IP адрес заканчивается неудачей, повторять попытки <**n**> секунд, перед тем как остановить попытки.

Если выставить значение <**n**> в «**infinite**» то попытки будут повторяться бесконечно.

По умолчанию, включена опция **resolv-retry infinite**. Можно отключить попытки повторения, выставив <**n**> равным **0**.

float

Разрешить удаленному узлу менять IP адрес и/или порт, например, если они изменились из-за изменения настроек DHCP (если опция **remote** не указана, опция **float** по умолчанию включена). Если опция **float** задается вместе с опцией **remote**, ПК «LirVPN» сначала подключается к удаленному узлу по адресу указанному в опции **remote**, после этого, если пакеты приходят с нового адреса и проходят все этапы аутентификации, новый адрес становиться основным. Это удобно, когда ПК «LirVPN» подключается к узлу с динамическим адресом, например, если этот узел использует DHCP или узел подключается к Интернет по коммутируемому каналу.

В общем случае, опция **float** говорит ПК «LirVPN» принимать пакеты, прошедшие аутентификацию от любого адреса, а не только от адреса, который был указан в опции **remote**.

ipchange <cmd>

Выполнить командный файл <**cmd**>, когда IP адрес ПК «LirVPN» клиента прошел аутентификацию или изменился.

Выполняется как:

```
cmd ip_address port_number
```

Не используйте опцию **ipchange** совместно с опцией **mode server**. Вместо этого используйте опцию **client-connect**.

Обратитесь к секции «**Переменные окружения**», чтобы узнать о дополнительных параметрах передающихся как переменные окружения.

Параметр <**cmd**> также может быть командным файлом с множеством аргументов, в этом случае аргументы которые передает ПК «LirVPN», будут добавлены к <**cmd**>, чтобы сформировать параметры которые будут переданы скрипту.

Если вы работаете в окружении, где используются динамические IP адреса и адреса узлов могут меняться без уведомления, вы можете использовать этот скрипт, например, для добавления в файл /etc/hosts текущего адреса узла. Командный файл будет вызываться каждый раз, когда у удаленного узла будет меняться IP адрес.

Также, если локальный IP адрес изменяется через DHCP, необходимо изменить скрипт изменения IP адреса (смотрите документацию к dhcpcd) для передачи сигнала **SIGHUP** или **SIGUSR1** в ПК «LirVPN». ПК «LirVPN» восстановит соединение с самым последним узлом прошедшим аутентификацию, используя новый IP адрес.

port <port>

Номер TCP/UDP порта для локального и удаленного узла. Текущий номер порта по умолчанию - **1194**.

lport <port>

Номер порта TCP/UDP для локального узла.

rport <port>
Номер порта TCP/UDP для удаленного узла.

nobind
Не привязываться к локальному адресу и порту. IP стек ОС сам выделит динамический порт для возвращаемых пакетов. Так как номер динамического порта не известен узлу заранее, эта опция подходит только для узлов, в конфигурации которых указана опция **remote**.

dev <tunX | tapX | null>
TUN/TAP виртуальное сетевой интерфейс (**X** может быть опущен для устройств создающихся динамически).

Вы должны использовать либо **tun** устройство на обоих сторонах туннеля, либо **tap** устройство на обоих сторонах туннеля. Их нельзя смешивать, т.к. в их основе лежат различные протоколы.

TUN устройства инкапсулируют IPv4 в то время как **TAP** устройства инкапсулируют Ethernet 802.3.

dev-type <device-type>
Опция указывает тип устройства TUN/TAP. Параметр <**device-type**> должен быть **«tun»** или **«tap»**. Используйте эту опцию, только если имя вашего устройства TUN/TAP заданного с помощью опции **dev** не начинается с **tun** или **tap**.

tun-ipv6
Настроить tun туннель для передачи трафика IPv6. Эта опция должна быть использована вместе с опцией **dev tun** или **dev tunX**. Если в ПК «LirVPN» не была встроена поддержка IPv6 TUN для вашей операционной системы, будет показано предупреждающее сообщение.

dev-node <node>
Четко указывает имя устройства, вместо того чтобы использовать
/dev/net/tun, /dev/tun, /dev/tap, и т.д. Если ПК «LirVPN» не может определить какое именно устройство указано в параметре **node** (TUN или TAP), основываясь на его имени, в этом случае также необходимо указать опцию **dev-type tun** или **dev-type tap**.

Если вы используете ОС Windows, параметр <**node**> должен содержать название TAP-Win32 адаптера как оно отображается в окне «Сетевые подключения» в «Панели управления», или GUID адаптера заключенный в кавычки. Опция командной строки ПК «LirVPN» **show-adapters** при использовании с ОС Windows может быть использована для отображения всех доступных TAP-Win32 адаптеров. Она показывает как название, так и GUID для каждого TAP-Win32 адаптера

ifconfig <l> <rn>
Установка параметров TUN/TAP адаптера. <**l**> - локальный IP адрес туннеля. Для TUN устройств. <**rn**>, это IP адрес удаленного конца туннеля. Для TAP устройств, <**rn**>, это маска подсети виртуального сегмента Ethernet, который создается или к которому происходит подключение.

Для TUN устройств (которые обслуживают виртуальные IP соединения точка-точка), правильными параметрами для опции **ifconfig** являются 2 внутренних IP адреса, которые не принадлежат любой существующей и используемой подсети. IP адреса должны указываться в обратном порядке в конфигурации удаленного узла. После установки VPN соединения, используя команду **ping <rn>**, вы можете проверить работоспособность туннеля.

Для TAP устройств, которые предоставляют возможность создавать виртуальные Ethernet сегменты, опция **ifconfig** используется для задания IP адреса и маски подсети, также как и на аппаратном Ethernet адаптере. Если вы устанавливаете соединение с удаленным Ethernet мостом, IP адрес и маска подсети должны быть заданы в соответствии с настройками объединяемого Ethernet сегмента (также для этих целей может использоваться DHCP).

Эта опция является оболочкой для команды **ifconfig**, и разработана для упрощения конфигурирования TUN/TAP туннеля, предоставляя стандартный интерфейс к различным реализациям **ifconfig** на разных платформах.

Параметры опции **ifconfig** также могут быть заданы как имя DNS или имя, записанное в файле /etc/hosts.

Для TAP устройств, опция **ifconfig** не должна использоваться, если TAP интерфейс получает IP адрес по DHCP.

ifconfig-noexec

Не выполнять команды **ifconfig/netsh**, вместо этого передавать параметры опции **ifconfig** скриптам используя переменные окружения.

ifconfig-nowarn

Не выводить предупреждающее сообщение о соответствии опций, если опция **ifconfig** на локальном узле не совпадает с настройками удаленного узла. Это удобно, если вы хотите сохранить проверку соответствия всех опций (также смотрите опцию **disable-occ**), кроме проверки опции **ifconfig**.

Например, если вы используете конфигурацию, в которой локальный узел использует опцию **ifconfig**, а удаленный узел нет, используйте опцию **ifconfig-nowarn** на локальном узле.

Эта опция также подавляет предупреждающие сообщения о потенциальных конфликтах в адресации, которые порой раздражают опытных пользователей.

route <network/IP> [<netmask>] [<gateway>] [<metric>]

Добавить маршрут в таблицу маршрутов после установки соединения. Возможно задание нескольких маршрутов. Маршруты будут автоматически удалены из таблицы маршрутов в обратном порядке после закрытия TUN/TAP устройства.

Эта опция является оболочкой для команды **route**, в тоже время она предоставляет одинаковую структуру задания опций на любой поддерживаемой ОС.

Параметр <**netmask**> по умолчанию равен - **255.255.255.255**.

По умолчанию параметр <**gateway**> берется из значения параметра опции **route-gateway** или из второго параметра опции **ifconfig** в том случае если задана опция **dev tun**.

Чтобы присвоить параметрам значения по умолчанию их можно оставить пустыми или задать в качестве значения «**default**».

Параметры <**network**> и <**gateway**> также могут быть заданы как имя DNS или имя, записанное в файле /etc/hosts или как одно из специальных значений:

vpn_gateway – Адрес удаленного конца VPN туннеля (берется из значения параметра **route-gateway** или из второго параметра опции **ifconfig** в том случае если задана опция **dev tun**).

net_gateway – существующий IP адрес основного шлюза, берется из таблицы маршрутов (не поддерживается на всех ОС).

remote_host – параметр опции **remote**, если ПК «LirVPN» запущен в режиме клиента и не задан в режиме сервера.

route-gateway <gw>

Указывает основной шлюз <**gw**> для использования с опцией **route**.

route-delay [<n>] [<w>]

Установить задержку в <**n**> секунд (по умолчанию - 0) после установки соединения, но перед добавлением маршрутов. Если <**n**> = 0, маршруты будут добавлены сразу же после установки соединения. Если опция **route-delay** не указана, маршруты будут добавлены сразу же после открытия устройства TUN/TAP и после выполнения команды указанной в параметре опции **up**, но до понижения привилегий (если указаны опции **user**, **group** или **chroot**).

Эта опция полезна в тех случаях, когда используется DHCP для задания адресов tap адаптера. Задержка дает DHCP время для установки параметров до того, как будут добавлены маршруты.

В ОС Windows, опция **route-delay** устанавливает время ожидания <**w**> секунд перед добавлением маршрутов (по умолчанию <**w**> = 30) для ожидания включения TAP-Win32 адаптера.

route-up <cmd>

Выполнить команду <**cmd**>, после того как будут добавлены маршруты в таблицу маршрутов (учитываются настройки **route-delay**).

Обратитесь к секции «**Переменные окружения**», чтобы узнать о дополнительных параметрах передающихся как переменные окружения.

Заметьте, что <**cmd**> может быть командой с множеством аргументов.

route-noexec

Не добавлять и не удалять маршруты автоматически. Вместо этого передавать маршруты команде указанной в параметре опции **route-up** используя переменные окружения.

redirect-gateway [<local>] [<def1>]

Автоматически выполнять команды добавления маршрутов, для того чтобы весь исходящий IP трафик передавался через ЗВКС.

Эта опция выполняет три действия:

6.1 Создает статический маршрут для адреса указанного в параметре к опции **remote**, который перенаправляет пакеты на существующий адрес основного шлюза. Это делается для того, чтобы в третьем действии не создавалась маршрутная петля;

6.2 Удаляет из таблицы маршрутов запись об основном шлюзе.

6.3 Устанавливает адрес удаленного конца туннеля ЗВКС в качестве основного шлюза (берется из значения параметра **route-gateway** или из второго параметра опции **ifconfig** в том случае если задана опция **dev tun**)

Когда туннель закрывается, все вышеперечисленные шаги проделываются в реверсивном порядке, и восстанавливается изначальный основной шлюз.

Добавьте флаг <**local**>, если оба ПК «LirVPN» сервера соединены напрямую и находятся в одной подсети, например в случае беспроводной сети. При использовании флага <**local**> первое действие из списка выше не выполняется.

Добавьте флаг <**def1**> чтобы заменить основной шлюз используя 0.0.0.0/1 и 128.0.0.0/1 вместо 0.0.0.0/0. Это позволяет заменить ранее установленный основной шлюз, но не удалять его.

link-mtu <n>

Устанавливает верхний предел размера UDP пакетов, которые передаются от одного узла ПК «LirVPN» к другому. Изменяйте этот параметр только в том случае, если вы уверены в ваших действиях.

tun-mtu <n>

Задает значение MTU для TUN устройства и передает этот параметр опции **link-mtu** (по умолчанию - 1500). В большинстве случаев, следует оставить этот параметр в значении по умолчанию.

MTU (Maximum Transmission Units), это максимальный размер датаграммы в байтах которая может быть передана без фрагментации через определенный сетевой путь. ПК «LirVPN» требует, чтобы пакеты передаваемые через канал управления и через канал данных не были фрагментированы.

Проблемы с MTU обычно проявляются в соединениях, которые зависают в периоды активного использования.

Опции **fragment** и/или **mssfix** более предпочтительны для разрешения вопросов с размерами MTU.

tun-mtu-extra <n>

Допускает, что TUN/TAP устройство не может возвращать больше <**n**> байт, чем размер **tun-mtu** при чтении. По умолчанию равен 0, что достаточно для большинства TUN устройств. TAP устройства могут добавлять дополнительные накладные расходы в связи с избыточностью размера MTU, поэтому значение 32 используется по умолчании при использовании TAP устройств. Эта опция управляет только внутренней политикой задания размера буфера, поэтому увеличение накладных расходов, не будет зависеть от увеличения <**n**>.

mtu-disc <type>

Контролирует использование «Path MTU discovery» на TCP/UDP каналах. Поддерживаются ОС, в которых реализованы необходимые системные вызовы, например Linux.

- «**no**» - Никогда не посылать не фрагментированные фреймы.
- «**maybe**» - Использовать per-route hints.
- «**yes**» - Всегда посылать не фрагментированные фреймы.

mtu-test

Для эмпирического замера MTU в начале соединения, добавьте опцию **mtu-test** к своей конфигурации. ПК «LirVPN» будет посылать «**ping**» пакеты различного размера удаленному узлу и определит самый большой пакет который был успешно получен. Замер обычно длится около 3-х минут.

fragment <max>

Задействует внутренний механизм управления фрагментацией датаграмм. В этом случае UDP датаграммы которые превышают размер <**max**> (в байтах) не отправляются.

Параметр <**max**> интерпретируется также как и параметр опции **link-mtu**, то есть параметр max - это размер пакета UDP после инкапсуляции, но без включения размера самого UDP заголовка.

Опция **fragment** имеет значение только в случае использования UDP протокола (**proto udp**).

Опция **fragment** добавляет 4 байта накладных расходов на датаграмму.

Обратитесь к описанию опции **mssfix**, так как она является важным дополнением к опции **fragment**.

Эта опция не заменит механизм управления фрагментацией UDP на уровне IP стека. Она может использоваться как последнее средство при неработоспособности «Path MTU discovery». Использование этой опции менее эффективно по сравнению с восстановлением работоспособности «Path MTU discovery» для вашего IP соединения и использования родного механизма управления фрагментацией IP.

Также необходимо сказать, что бывают случаи когда использование внутреннего механизма управления фрагментацией ПК «LirVPN» может быть единственным средством, например в случае туннелирования группового (multicast) потока UDP, которому необходима фрагментация.

mssfix <max>

Информирует TCP сессии, работающие через туннель, о том, что они должны ограничить размер посылаемых пакетов сразу же после того как ПК «LirVPN» инкапсулирует их. Получаемый размер UDP пакета, который
ПК «LirVPN» посылает удаленному узлу, не будет превышать <**max**> байт.

Параметр <**max**> интерпретируется также как и параметр опции **link-mtu**, то есть параметр <**max**> - это размер пакета UDP после инкапсуляции, но без включения размера самого UDP заголовка.

Опция **mssfix** имеет значение только в случае использования UDP протокола (**proto udp**).

Опции **mssfix** и **fragment** идеальны для использования совместно, опция **mssfix** старается оградить TCP от возможной фрагментации пакетов, и если все таки большие пакеты как-то пройдут через этот барьер (в случае протоколов отличных от TCP), опция **fragment** фрагментирует их используя внутренний механизм управления фрагментацией.

Обе опции **fragment** и **mssfix** используются для обхода ситуаций, когда «Path MTU discovery» не работает на канале связи между узлами ПК «LirVPN».

Обычный симптом такой неработоспособности - зависание работающего туннеля ПК «LirVPN» в период активного использования.

Если опции **fragment** и **mssfix** используются вместе, **mssfix** берет свой параметр по умолчанию <**max**> из параметра <**max**> опции **fragment**.

Следующий набор опций понижает максимальный размер пакетов UDP до 1300 байт (хорошее начало для решения проблем связи связанных с MTU):

tun-mtu 1500
fragment 1300
mssfix

sndbuf <size> Устанавливает размер буфера отправки TCP/UDP сокета. По умолчанию 65536 байт.

rcvbuf <size> Устанавливает размер буфера приема TCP/UDP сокета. По умолчанию 65536 байт.

txqueuelen <n> (Только для Linux) Установить длину очереди TX на TUN/TAP интерфейсе. По умолчанию - 100.

shaper <n>

Ограничить пропускную способность исходящего канала в туннеле до <**n**> байт в секунду на TCP или UDP портах. Если вы хотите ограничить пропускную способность в обоих направлениях, используйте данную опцию на обоих узлах.

ПК «LirVPN» использует следующий алгоритм для управления скоростью передачи данных: если заданна скорость **n** байт в секунду, то после записи в очередь TCP/UDP порта датаграммы длиной **b** байт, происходит ожидание как минимум **(b/n)** секунд перед следующей записью в очередь.

Необходимо заметить, что ПК «LirVPN» позволяет реализовать несколько туннелей между двумя узлами. Это позволяет в одно и тоже время создавать как высокоскоростные туннели, так и туннели с ограниченной пропускной способностью, и передавать низкоприоритетную информацию через туннель с ограниченной пропускной способностью, а все остальные данные через высокоскоростной туннель.

Заметьте, что для туннелей с низкой пропускной способностью (меньше 1000 байт в секунду), вы возможно должны использовать меньшие значения MTU (смотрите выше), иначе задержки при передаче пакетов возрастут настолько, что будут достигнуты таймауты на TLS уровне и в TCP соединениях.

Параметр <**n**> может принимать значения от 100 байт в секунду, до 100 Мбайт в секунду.

inactive <n>

Останавливает ПК «LirVPN» если в течении <**n**> секунд через TUN/TAP интерфейс не прошло ни одного пакета. Время замеряется с момента приема через туннель последнего входящего пакета.

ping <n>

Посылать «**ping**» пакет удаленному узлу через управляющее соединение TCP/UDP, если в течении <**n**> секунд через туннель не было передано хотя бы одного пакета. Укажите опцию **ping** на обоих узлах для того, чтобы «**ping**» пакеты пересылались в обоих направлениях. «**ping**» пакеты ПК «LirVPN» не возвращаются как обычные ICMP пакеты. Если опция используется в конфигурации с заданными опциями **secret**, **tls-server**, или **tls-client**, «**ping**» пакеты будет пересылаться в зашифрованном виде.

Эта опция используется в двух случаях:

- Для совместимости с межсетевыми экранами с контролем состояния соединения. Периодическая передача «**ping**» пакета для уверенности, что правило межсетевого экрана с контролем состояния соединения разрешающее прохождение UDP пакетов ПК «LirVPN» не закончит свое действие по истечению определенного количества времени;

- Для использования опции **ping-exit**.

ping-exit <n>

Заставляет ПК «LirVPN» завершить работу по прошествии <**n**> секунд, если за это время не был получен «**ping**» пакет, либо любой пакет от удаленного узла через туннель. Эта опция может быть использована вместе с опциями **inactive**, **ping** и **ping-exit** для создания двухуровневых проверок на активность соединения. **Например**:

```
inactive 3600
ping 10
ping-exit 60
```

В случае если данные опции используется на обоих узлах,
ПК «LirVPN» завершает свою работу по прошествии 60 секунд после отключения удаленного узла, и после 60 минут, если через открытый туннель не было передано какой либо информации.

ping-restart <n> Сходна с опцией **ping-exit**, но в отличие от нее не завершает работу ПК «LirVPN», а инициирует сигнал **SIGUSR1**, по прошествии <**n**> секунд, если через открытый туннель не было передано какой-либо информации или не был получен хотя бы один «**ping**» пакет.

Эта опция полезна в случаях использования удаленным узлом динамического IP адреса и для присвоения этому IP DNS имени используются специальные сервисы (например, используется сервис http://dyndns.org/ совместно с клиентом ddclient)

Если удаленный узел недоступен, будет произведен перезапуск ПК «LirVPN» и DNS имя, указанное в параметре опции **remote** будет заново разрешено в IP адрес (если указана опция –resolv-retry).

В режиме сервера, действия опций **ping-restart**, **inactive**, а также любой сигнал всегда применяются к конкретному клиентскому соединению, и никогда к самому серверу целиком. Также заметьте, что в режиме сервера любой сигнал, который обычно вызывает перезапуск ПК «LirVPN», вместо этого вызывает разрыв клиентского соединения.

В режиме клиента, параметр опции **ping-restart** по умолчанию равен 120-ти секундам. Значение по умолчанию сохраняется до тех пор, пока клиент не получит от сервера новое значение, основанное на параметрах опции **keepalive** указанных на сервере. Для сброса значения по умолчанию, задайте **ping-restart** равным 0 на клиенте.

Обратитесь к секции «**Сигналы**» для подробной информации о сигнале **SIGUSR1**.

Заметьте, что действие сигнала **SIGUSR1** может быть изменено опциями **persist-tun**, **persist-key**, **persist-local-ip**, и **persist-remote-ip**.

Также заметьте, что опции **ping-exit** и **ping-restart** не могут использоваться совместно.

keepalive <n> <m>

Опция для упрощения задания опций **ping** и **ping-restart** в режиме сервера. Например, **keepalive 10 60** раскрывается как: **В режиме сервера**:

```
ping 10
ping-restart 120
push "ping 10"
push "ping-restart 60"
```

во всех остальных режимах:

```
ping 10
ping-restart 60
```

ping-timer-rem

Запускать таймер опций **ping-exit** и **ping-restart** только тогда, когда установлено соединение с удаленным узлом. Используйте эту опцию, если вы запускаете ПК «LirVPN» в ждущем режиме (например, без указания опции **remote**), и вы не хотите начинать отсчет до тех пор, пока удаленный узел не установит соединение.

persist-tun

Не закрывать и не открывать заново TUN/TAP устройство и не запускать команды указанные в параметрах опций **up** и **down** в случае получения сигнала **SIGUSR1** или в результате действия опции **ping-restart**. **SIGUSR1**, это сигнал перезапуска ПК «LirVPN» сходный с сигналом **SIGHUP**, но предоставляющий более тонкое управление над процессом перезапуска.

persist-key

Не считывать каждый раз файлы ключей при получении сигнала **SIGUSR1** или действия опции **ping-restart**.

Эта опция может использоваться с опцией **user nobody**, чтобы дать возможность для перезапуска ПК «LirVPN» по сигналу **SIGUSR1**. Обычно если вы сбрасываете привилегии суперпользователя, ПК «LirVPN» не может перезапуститься, так как он не может считать заново файлы закрытых ключей.

Эта опция решает проблему путем сохранения ключей между перезапусками ПК «LirVPN» вызванными сигналом **SIGUSR1**, поэтому не появляется надобности считывать их каждый раз заново.

persist-local-ip

Сохранять изначально присвоенный IP адрес и номер порта локального узла между перезапусками ПК «LirVPN», в случае получения сигнала **SIGUSR1** или в результате действия опции **ping-restart**.

persist-remote-ip

Сохранять последний прошедший аутентификацию IP адрес и номер порта удаленного узла, в случае получения сигнала **SIGUSR1** или в результате действия опции **ping-restart**.

mlock

Отключает сброс страниц оперативной памяти в своп-файл путем вызова функции POSIX mlockall. Требует запуск ПК «LirVPN» с правами суперпользователя (впрочем ПК «LirVPN» может изменить UID используя опцию **user**).

Использование этой опции дает гарантию, что временные ключи и данные туннеля никогда не будут записаны на жесткий диск, в связи с операциями с виртуальной памятью, которые используются в большинстве современных ОС. Это дает гарантию, что даже если злоумышленник смог получить контроль над компьютером с запущенным ПК «LirVPN», он не сможет просканировать системный своп-файл на предмет получения используемых ранее временных ключей, которые были использованы в период времени управляемом опцией **reneg**, и после этого отброшены.

Обратной стороной использования опции **mlock** является то, что она уменьшает количество оперативной памяти доступной другим приложениям.

up <cmd>

Команда оболочки, которая исполняется после успешного открытия
TUN/TAP устройства (до изменения UID). Эта команда полезна для задания маршрутов, которые перенаправляют IP трафик, предназначенный для внутренних подсетей, находящихся на стороне удаленного узла ПК «LirVPN» через туннель.

Если указана опция **dev tun** выполняется как:

```
cmd tun_dev tun_mtu link_mtu ifconfig_local_ip ifconfig_remote_ip [ init | restart ]
```

Если указана опция **dev tap** выполняется как:

```
cmd tap_dev tap_mtu link_mtu ifconfig_local_ip ifconfig_netmask [ init | restart ]
```

Обратитесь к секции «**Переменные окружения**», чтобы узнать о дополнительных параметрах передающихся как переменные окружения.

Параметр <**cmd**> также может быть командным файлом с множеством аргументов, в этом случае аргументы которые передает ПК «LirVPN», будут добавлены к <**cmd**>, чтобы сформировать параметры которые будут переданы скрипту.

Обычно, <**cmd**> выполняет скрипт для добавления маршрутов перенаправляющих трафик через туннель.

Обычно <**cmd**> вызывается после открытия TUN/TAP устройства. В этом случае параметр **init** будет последним переданным в командую строку. Если задана в конфигурации ПК «LirVPN» также задана опция **up-restart**, <**cmd**> скрипт будет вызываться и при перезапуске ПК «LirVPN», в котором состояние TUN/TAP интерфейса сохраняется (такое поведение задает опция **persist-tun**). Также перезапуск может быть инициирован сигналом **SIGUSR1**, действием опции **ping-restart**, или обрывом соединения, если для связи используется протокол TCP (опция **proto**). Если происходит перезапуск ПК «LirVPN», и задана опция **up-restart**, <**cmd**> скрипт вызывается с **restart** в качестве последнего параметра.

Заметьте, что ПК «LirVPN» также предоставляет опцию **ifconfig** для автоматического задания параметров TUN устройству и добавления маршрута до подсети находящейся за удаленным концом VPN туннеля, при этом опцию **up** указывать не надо, кроме тех случаев, когда вы хотите задать маршруты именно через скрипт <**cmd**>.

Если опция **ifconfig** указана, ПК «LirVPN» передает адреса локального и удаленного концов VPN туннеля в командную строку скрипта, заданного в параметре опции **up**, для их использования в конфигурации таблицы маршрутов, например:

route add -net 10.0.0.0 netmask 255.255.255.0 gw $5 –up-delay

Задержать открытие TUN/TAP и выполнение скрипта указанного в параметре опции **up** до того момента, пока TCP/UDP соединение с удаленным узлом не будет установлено.

В режиме **proto udp**, эта опция обычно требует задания опции **ping** для распознавания того, что соединение установлено, в отсутствии данных переданных через туннель, так как UDP это «бессвязный» протокол.

В ОС Windows, эта опция задерживает переход TAP-Win32 интерфейса в состояние «подключен» до установки фактической установки соединения, например, до получения от удаленного узла первого пакета прошедшего аутентификацию.

down <cmd>

Команда оболочки, которую необходимо запустить после закрытия
TUN/TAP устройства (после изменения UID и/или **chroot**). Вызывается с теми же параметрами и переменными окружения, как и в опции **up**. Заметьте, если вы понижаете привилегии, используя опции **user** и **group**, скрипт <**cmd**> тоже будет запущен с пониженными привилегиями.

down-pre Выполнить команду, указанную в параметре опции **down** до закрытия TUN/TAP устройства, а не после.

up-restart

Запускать команды указанные в параметрах опций **up** и **down** при перезагрузках ПК «LirVPN» также как и при первом старте ПК «LirVPN». Эта опция описана подробнее в описании опции **up**.

setenv <name> <value>

Задать собственную переменную окружения <**name**>=<**value**> для передачи скриптам.

disable-occ

Не выводить предупреждающее сообщение, если обнаружено несоответствие опций между узлами. Примером несоответствия может служить задание опции **dev tun** на одном узле и

опции **dev tap** на другом. Использование этой опции может показаться странным, но предоставляет возможность, в качестве временного решения в ситуациях, когда актуальная версия ПК «LirVPN» должна установить связь с более старой версией.

user <user>

Изменяет ID пользователя процесса ПК «LirVPN» на <**user**> после запуска, и понижает привилегии процесса. Эта опция полезна для защиты системы в случаях получения злоумышленником контроля над процессом ПК «LirVPN». Несмотря на то, что это маловероятно из-за функциональных особенностей защиты ПК «LirVPN», эта опция обеспечивает второю линию обороны.

Установка «**nobody**» в качестве параметра <**user**> или выбор любого другого пользователя с аналогичными привилегиями, ограничивает злоумышленника в действиях по причинению вреда, который он может нанести. Очевидно, что если привилегии были понижены, они не могут быть возвращены процессу ПК «LirVPN». В этом случае если вы хотите перезапустить ПК «LirVPN» с помощью сигнала **SIGUSR1** (например, при обновлении IP адреса через DHCP), вы должны использовать одну из **persist** опций, для уверенности в том, что процессу ПК «LirVPN» не потребуется дополнительных привилегий для корректного перезапуска (например, для перечитывания ключей или задания свойств TUN устройства).

group <group>

Аналогично опции **user**, эта опция изменяет ID группы для процесса ПК «LirVPN» на <**group**> после запуска.

cd <dir>

Изменить директорию на <**dir**> перед чтением каких-либо файлов, таких как файлы конфигурации, файлы ключей, скрипты и так далее. Параметр <**dir**> должен являться абсолютным путем, начинаться с «/» и не должен содержать каких либо ссылок на текущую директорию, таких как «.» или «..».

Эта опция полезна в случае запуска ПК «LirVPN» в режиме **daemon**, и вы хотите собрать все файлы управления ПК «LirVPN» в одном месте.

chroot <dir>

Делает директорию <**dir**> корневой для ПК «LirVPN» после запуска. После этого ПК «LirVPN» будут недоступны любые файлы за пределами этой директории. Это может быть полезно с точки зрения безопасности.

Так как выполнение этой операции задерживается до фактического запуска ПК «LirVPN», большинство опций ПК «LirVPN» указывающих на файлы будут работать правильно до изменения корневой директории.

В большинстве случаев, параметр <**dir**> может указывать на пустую директорию, но могут возникнуть сложности при запуске скриптов и перезапуске самого ПК «LirVPN» после изменения корневой директории.

daemon [<progname>]

Перейти в режим «демона» после окончания инициализации всех функций. При задании этой опции все сообщения и сообщения об ошибках будут перенаправляться в системный журнал (например, /var/log/messages), кроме сообщений от скриптов и команд ifconfig, которые будут перенаправлены в /dev/null, если они каким-либо образом не перенаправлены в другое место. Перенаправление вступает в силу сразу же, как опция **daemon** будет распознана в командной строке, даже если сам переход в режим «демона» будет произведен позже. Перенаправлении в системный журнал не будет выполнятся, если указана опция **log** или **log-append**.

Опциональный параметр <**progname**> передает имя программы, от которой передаются сообщения в системный журнал. Это может быть удобно для отделения сообщений от разных туннелей ПК «LirVPN» в системном журнале. Если параметр не задан, параметр <**progname**> по умолчанию равен «**lirvpn**».

Когда ПК «LirVPN» запущен с опцией **daemon**, он старается задержать переход в режим «демона» до тех пор пока все функции в которых возможны критические ошибки не закончат инициализацию. Это означает, что стартовые скрипты могут проверить статус команды lirvpn, для сравнительно честной индикации того, что ПК «LirVPN» удачно стартовал и перешел в режим передачи пакетов в туннель.

В ПК «LirVPN», сравнительно большое количество сообщений об ошибках, которые появляются после инициализации, не критичны.

syslog [<progname>]

Перенаправлять вывод сообщений о работе в системный журнал сообщений, но не переходить в режим **daemon**. Смотрите выше описание опции **daemon** для описания параметра <**progname**>.

passtos

Изменить поле TOS в пакете, отправляемом через туннель на такой же, как и TOS в оригинальном пакете.

log <file>

Записывать сообщения о работе в <**file**>, включая вывод в stdout/stderr, который создается вызываемыми скриптами. Если <**file**> уже существует, его содержимое будет потеряно. Опция вступает в силу сразу же после того как будет распознана ПК «LirVPN» и перенаправляет сообщения в файл, даже если заданы опции **daemon** или **inetd**. Эта опция остается действовать в течение всего времени работы ПК «LirVPN», и продолжает действовать после получения сигналов **SIGHUP**, **SIGUSR1**, или действия опции **ping-restart**.

log-append <file>

Добавлять сообщения о работе в <**file**>. Если <**file**> не существует, он будет создан. Эта опция работает также как и опция **log**, за исключением того, что сообщения добавляются в файл журнала работы, вместо его предварительной очистки.

suppress-timestamps

Не добавлять временные метки в сообщения журнала работы. Это также относится к сообщениям журнала работы, которые посылаются на стандартный вывод.

writepid <file>

Записать ID основного процесса ПК «LirVPN» в <**file**>.

nice <n>

Изменить приоритет выполнения процесса ПК «LirVPN» после инициализации (<**n**> больше 0 - понижение приоритета, <**n**> меньше 0 - повышение приоритета).

Оптимизирует операции Ввода/Вывода для TUN/TAP/UDP путем исключения вызовов poll/epoll/select перед операциями записи. Назначение этих вызовов состоит в блокировании устройств или сокетов до тех пор, пока они не будут готовы принять данные. Такие блокировки не нужны на некоторых платформах, которые не поддерживают блокировку перед записью на UDP сокетах или TUN/TAP устройствах. В этих случаях, данная опция может оптимизировать

цикл обработки событий путем исключения poll/epoll/select вызовов, поднимая эффективность использования ЦП на 5-10 процентов.

Эта опция может быть использована только на ОС отличных от Windows, когда задана опция **proto udp**, и когда опция **shaper** не задана.

echo [<parms...>]

Передать <**parms**> в вывод сообщений о работе. Используется для передачи сообщений управляющему приложению, которое получает сообщения о работе ПК «LirVPN».

remap-usr1 <signal>

Контролирует переназначение сигналов **SIGUSR1** полученных изнутри или снаружи в **SIGHUP** (перезапуск без сохранения состояния) или в **SIGTERM** (выход).

Параметр <**signal**> может принимать значения «**SIGHUP**» или «**SIGTERM**». По умолчанию, переназначения не производятся.

verb <n>

Установить детальность выводимых сообщений - <**n**> (по умолчанию - 1). Каждый уровень выводит всю информацию из предыдущего уровня. Уровень 3 рекомендуется, если вы хотите получить достаточно полную информацию и не запутаться в слишком большом количестве сообщений.

0 - Без вывода, только критические ошибки. **от 1 до 4** - Интервал для обычного использования. **5** - Выводит **R** и **W** символы на консоль для каждого полученного или переданного пакета, верхний регистр используется для TCP/UDP пакетов, а нижний для TUN/TAP пакетов. **от 6 до 11** - Уровни отладочной информации.

status <file> [<n>]

Записывать текущее состояние ПК «LirVPN» в <**file**> каждые <**n**> секунд. Статус также может быть записан в **syslog** передачей ПК «LirVPN» сигнала **SIGUSR2**.

status-version [<n>]

Выбор номера версии формата для файла статуса. <**n**> может быть 1 или 2. По умолчанию 1.

mute <n>

Записывать в журнал работы не более <**n**> повторяющихся сообщений из одной категории. Это полезно для ограничения записи в журнал работы повторяющихся сообщений одного типа.

comp-lzo

Использовать быстрое LZO сжатие. Данная опция может добавлять 1 байт на пакет для несжимаемых данных.

comp-noadapt

При использовании совместно с опцией **comp-lzo**, эта опция отключает алгоритм адаптивного сжатия. Обычно адаптивное сжатие включено, если задана опция **comp-lzo**.

Адаптивное сжатие пытается оптимизировать процесс передачи данных, в случае если сжатие данных включено, но через туннель пересылаются преимущественно несжимаемые (или уже сжатые) пакеты, например, такие как при передаче через FTP или rsync больших сжатых файлов. С включенным адаптивным сжатием, ПК «LirVPN» периодически проверяет эффективность сжатия и если она слишком мала, отключает сжатие до следующей проверки.

management <IP> <port> [<pw-file>]

Открыть TCP сервер на <**IP**>:<**port**> для управляющего канала. <**pw-file**> (если параметр задан), это файл с паролем (пароль на первой строке) или «**stdin**» для запроса пароля со стандартного ввода. Полученный пароль будет использован для проверки полномочий TCP клиентов к функциям управляющего канала.

Управляющий канал предоставляет специальный режим, в котором соединение с управляющим каналом может производиться через сам VPN туннель. Для включения этого режима установите параметр <**IP**> = «**tunnel**». В этом режиме управляющий канал будет ждать соединения TCP соединения на локальном VPN адресе TUN/TAP интерфейса.

Основная задача управляющего канала - дать доступ к управлению ПК «LirVPN» сторонним программам, также возможно установить telnet соединение с управляющим каналом, используя telnet-клиент в режиме «raw». После соединения передайте команду «**help**», для вывода списка возможных команд.

Настойчиво рекомендуется указывать адрес **127.0.0.1(localhost)** в качестве параметра <**IP**>, для разрешения доступа к управляющему каналу только локальным клиентам.

management-query-passwords

Запрашивать через управляющий канал пароль на закрытый ключ, а также имя пользователя и пароль для опции **auth-user-pass**. Запрашивать ввод через управляющий канал только в тех случаях, в которых обычно запрос происходит через консоль.

management-hold

Запускать ПК «LirVPN» в режиме ожидания, и ждать до тех пор, пока через управляющий интерфейс не поступит команда **«hold release»**.

management-log-cache <n>

Кэшировать <**n**> последних строк истории файла журнала для использования управляющим каналом.

plugin <module-pathname> [<init-string>]

Загрузить подключаемый модуль из файла <**module-pathname**>, передавая <**init-string**> в качестве аргумента к функции инициализации модуля. Может быть задано несколько подключаемых модулей.

Несколько подключаемых модулей могут использоваться одновременно, также модули могут использоваться вместе со скриптами. Модули будут вызваны из ПК «LirVPN» в том порядке, в каком они указаны в файле конфигурации. Если и подключаемый модуль и скрипт настроены на один и тот же вызов, скрипт вызывается последним. Если код возврата модуля/скрипта управляет функциями аутентификации (такими как **tls-verify**, **auth-user-pass-verify** или **client-connect**), в этом случае каждый модуль или скрипт должны возвращать 0 для того чтобы соединение прошло аутентификацию.

6.2 Опции режима «Сервер»

Режим «**сервер**» включается заданием опции **mode server** в конфигурационном файле. В этом режиме, ПК «LirVPN» ждать входящих соединений на одном сетевом порту. Все соединения будут перенаправляться через один tun или tap интерфейс. Этот режим может поддерживать сотни и даже тысячи клиентов на достаточно мощном оборудовании. В этом режиме должна использоваться SSL/TLS аутентификация.

server <network> <netmask>

Опция упрощает настройку ПК «LirVPN» в режиме «**сервер**». Данная опция настраивает сервер ПК «LirVPN», который выделяет адреса клиентам из заданной подсети. Сам сервер берет «.1» адрес заданной подсети для использования в качестве адреса локального TUN/TAP интерфейса. Например, **server 10.8.0.0 255.255.255.0** раскрывается так:

mode server
tls-server

если **dev tun**:

ifconfig 10.8.0.1 10.8.0.2
ifconfig-pool 10.8.0.4 10.8.0.251
route 10.8.0.0 255.255.255.0

если **client-to-client**:

push "route 10.8.0.0 255.255.255.0"

иначе:

push "route 10.8.0.1"

если **dev tap**:

ifconfig 10.8.0.1 255.255.255.0
ifconfig-pool 10.8.0.2 10.8.0.254 255.255.255.0
push "route-gateway 10.8.0.1"

Не используйте опцию **server** для настройки туннелей типа «**мост**». Вместо этого используйте опцию **server-bridge**.

server-bridge <gateway> <netmask> <pool-start-IP> <pool-end-IP>

Опция упрощает настройку ПК «LirVPN» в режиме «**сервер**» при использовании туннелей типа «**мост**».

Для настройки туннеля типа «**мост**», вы должны использовать инструменты вашей ОС для объединения TAP интерфейса с интерфейсом Ethernet. Например, на ОС Linux это выполняется с помощью утилиты brctl, а в ОС Windows XP это выполняется в окне «Сетевые Подключения».

Далее необходимо вручную задать IP адрес и маску подсети на «объединенном» интерфейсе. Параметры <**gateway**> и <**netmask**> могут быть установлены как IP адрес и маска подсети «объединенного» интерфейса, или как IP адрес и маска подсети шлюза по молчанию в объединяемой подсети.

Выделите диапазон IP адресов для объединяемой подсети, используя параметры <**pool-start-IP**> и <**pool-end-IP**>. ПК «LirVPN» будет использовать его для выделения адресов подключающимся клиентам.

Например, **server-bridge 10.8.0.4 255.255.255.0 10.8.0.128 10.8.0.254** раскрывается так:

mode server
tls-server
ifconfig-pool 10.8.0.128 10.8.0.254 255.255.255.0
push "route-gateway 10.8.0.4"

push <option> Передает опцию конфигурационного файла клиенту. Параметр <**option**> должен быть заключен в кавычки (""). В конфигурационном файле клиента должна присутствовать опция **pull**. Набор опций, которые могут быть переданы клиентам, ограничен как возможностью реализации, так и соображениями безопасности. Некоторые опции, например те которые вызывают выполнение скриптов запрещены, так как в результате их действия может быть выполнен произвольный код на клиентской машине. Другие опции, такие как опции

TLS или MTU не могут быть переданы, потому что клиенту нужно знать их, перед тем как установить соединение с сервером.

Список опций, которые могут быть переданы клиентам: **route**, **route-gateway**, **route-delay**, **redirect-gateway**, **ip-win32**, **dhcp-option**, **inactive**, **ping**, **ping-exit**, **ping-restart**, **setenv**, **persist-key**, **persist-tun**, **echo**

push-reset Не передавать опции заданные в конфигурационном файле сервера (с помощью опции **push**) определенным клиентам. Эта опция задается в контексте клиента, например в индивидуальном файле конфигурации (опция **client-config-dir**). Эта опция игнорирует опции передаваемые с помощью опции **push** на уровне конфигурационного файла сервера.

disable

Запретить соединения для определенного клиента (основываясь на common name). Не используйте эту опцию для запрета доступа в случае компрометации ключа или пароля. Вместо этого используйте СОС (список отозванных сертификатов, обратитесь к опции **crl-verify**).

Эта опция должна быть задана либо в индивидуальном конфигурационном файле клиента (опция **client-config-dir**), либо создана динамически, используя скрипт заданный в параметре опции **client-connect**.

ifconfig-pool <start-IP> <end-IP> [<netmask>]

Выделение пула подсетей для динамического распределения между клиентами, устанавливающими соединение, подобно серверу DHCP. Для tun туннелей, каждому клиенту будет выделена /30 подсеть (для функциональной совместимости с Windows). Для tap туннелей, каждому клиенту будет выделен отдельный адрес, и опциональный параметр <**netmask**> также будет передан клиентам.

ifconfig-pool-persist <file> [<seconds>]

Записывать данные о выделенных IP адресах или подсетях в файл <**file**>, каждые <**seconds**> секунд (по умолчанию - 600), а также при старте и закрытии ПК «LirVPN».

Задача этой опции в предоставлении длительной ассоциации между клиентами (определяемых по их common name) и виртуальными IP адресами выделенными им из **ifconfig-pool**. Поддержание долгосрочных ассоциаций позволяет клиентам более эффективно использовать опцию **persist-tun**.

Параметр <**file**>, это ASCII файл с запятой в качестве разделителя, имеющий формат <**Common-Name**>,<**IP-адрес**>.

Если параметр <**seconds**> равен 0, **file** будет иметь атрибут доступа «**только для чтения**». Это удобно если вы хотите использовать **file** в качестве конфигурационного файла.

Записи в этом файле расцениваются ПК «LirVPN» только в качестве предложения, основывающимся на последних ассоциациях между common name и IP адресом. Они не гарантируют, что клиент с определенным common name всегда будет получать определенный IP адрес. Если вы хотите получить гарантию при задании адреса, используйте опцию **ifconfig-push**.

ifconfig-pool-linear

Заставляет опцию **ifconfig-pool** выделять отдельный адрес TUN интерфейсу клиентов вместо /30 подсетей. Эта опция не совместима с клиентами, использующими ОС Windows.

ifconfig-push <local> <remote-netmask>

Передавать конкретный виртуальный IP адрес для TUN/TAP интерфейса клиента, вместо динамического выделения с помощью опции **ifconfig-pool**.

Параметры <**local**> и <**remote-netmask**> задаются также как параметры опции **ifconfig**, если бы вы задавали эту опцию на стороне клиента. Заметьте, что параметры <**local**> и <**remote-netmask**> задаются для клиента, а не для сервера. Эти параметры могут быть заданы как DNS имена, в этом случае они будут разрешены в IP адреса на сервере в момент установки соединения клиентом.

Эта опция должна задаваться в файле конфигурации конкретного клиента (опция **client-config-dir**) или динамически создаваться, используя скрипт указанный в параметре опции **client-connect**.

Также не забудьте включить опцию **route** в конфигурационный файл клиента ПК «LirVPN» с указанием в качестве параметра <**local**>, для того чтобы ядро ОС знало, как перенаправлять трафик на TUN/TAP интерфейс сервера.

Алгоритм присвоения виртуального IP адреса клиенту работает в следующем порядке:

6.1 Используется созданный опцией **client-connect** файл для задания статического IP адреса.

6.2 Используется опция **client-config-dir** для задания статического IP адреса.

6.3 Используется опция **ifconfig-pool** для задания динамического IP адреса.

iroute <network> [<netmask>]

Создать внутренний сетевой маршрут до подсети определенного клиента. Если параметр <**netmask**> не задан, то он по умолчанию принимает значение - 255.255.255.255.

Эта опция может использоваться для предоставления доступа к подсети клиента всем подключенным к серверу ПК «LirVPN» клиентам, независимо от того, откуда они устанавливают соединение. Имейте в виду, что вы также должны добавить маршрут в таблицу маршрутов ОС клиентов (например, используя опцию **route**). Необходимостью в использовании двух маршрутов является то, что опция **route** перенаправляет пакеты из ядра ОС системы в ПК «LirVPN», а опция **iroute** перенаправляет пакеты в подсеть клиента, к которой необходимо получить доступ.

Эта опция должна быть задана либо в индивидуальном конфигурационном файле клиента (опция **client-config-dir**), либо создана динамически, используя скрипт заданный в параметре опции **client-connect**.

Опция **iroute** также имеет важное взаимодействие с опцией **push route**. Опция **iroute** задает подсеть, в которой находиться определенный клиент (назовем его клиент A). Если вам необходимо чтобы остальные клиенты имели доступ к подсети клиента A, вы можете использовать опцию **push route** совместно с опцией **client-to-client**. Для того чтобы все клиенты имели доступ к подсети клиента A, ПК «LirVPN» должен передать этот маршрут всем клиентам , кроме клиента A, так как клиент A уже находится в этой подсети. ПК «LirVPN» не передает маршрут клиенту на подсеть, если она задана в его опции **iroute**.

client-to-client

Так как ПК «LirVPN» в режиме «**сервер**» обслуживает множество клиентов через один tun или tap интерфейс, его можно назвать маршрутизатором. Опция **client-to-client** маршрутизирует трафик, который идет от клиента к клиенту, вместо передачи всего трафика от клиентов на TUN/TAP интерфейс сервера.

Когда используется эта опция, каждый клиент установивший соединение с сервером ПК «LirVPN» может напрямую обратиться к другому клиенту также установившему соединение с сервером ПК «LirVPN». Иначе каждый клиент может обмениваться информацией только с сервером. Не используйте эту опцию, если вы хотите фильтровать трафик, проходящий через

туннель с помощью межсетевого экрана, используя специальные (сформированные для определенных клиентов) правила межсетевого экрана.

duplicate-cn

Разрешить одновременное подключение множества клиентов с одинаковым common name. Если эта опция не задана, ПК «LirVPN» завершит соединение с клиентом при подключении нового клиента с тем же самым common name.

client-connect <script>

Запустить <**script**> при подключении клиента. Скрипт принимает common name и IP адрес клиента только что прошедшего аутентификацию в качестве переменных окружения (обратитесь к главе «**Переменные окружения**»). Скрипт также принимает путь до еще не созданного временного файла как $1 (первый аргумент командной строки), который будет использоваться скриптом, для передачи динамически сформированного конфигурационного файла обратно ПК «LirVPN».

Если скрипту необходимо сформировать динамический файл конфигурации, который будет использоваться на сервере в момент подключения клиента, он должен записать его в файл с именем $1.

Обратитесь к описанию опции **client-config-dir** за списком опций, которые могут быть использованы в динамически создаваемом файле конфигурации.

Заметьте, что значение, которое возвращает <**script**> важно. Если <**script**> возвращает значение отличное от 0, это приведет к отключению клиента.

client-disconnect

Опция аналогична опции **client-connect**, но вызывает скрипт в случае закрытия соединения клиентом. Скрипт не будет вызван, если **client-connect** скрипт и плагины (если заданы), которые были вызваны до этого для этого соединения, не возвратят код завершения - 0.

Исключение к этому правилу: если **client-disconnect** скрипт или плагины расположены друг за другом, и по крайней мере одна функция **client-connect** успешно завершилась, тогда все функции **client-disconnect** для скриптов и плагинов будут вызваны при завершении соединения клиентом, даже в тех случаях в которых функции относящиеся к **client-connect** возвратили код завершения отличный от 0.

client-config-dir <dir>

Задает директорию <**dir**> для пользовательских конфигурационных файлов. После соединения клиент проходит аутентификацию, ПК «LirVPN» просматривает эту директорию на предмет файла, имеющего такое же имя, как и X509 common name клиента. Если такой файл найден, он будет открыт и проанализирован на предмет наличия опций конфигурации специфичных для клиента. Если такого файла найдено не будет, ПК «LirVPN» попытается открыть и обработать файл по умолчанию с названием «**DEFAULT**», наличие которого не обязательно.

В этом файле можно задать определенный IP адрес для конкретного клиента, используя опцию **ifconfig-push**, а также определенные подсети, принадлежащие клиенту, с помощью опции **iroute**.

Одним из полезных свойств данной опции является то, что она дает возможность легко создавать, изменять или удалять клиентские файлы конфигурации в процессе работы сервера, без необходимости его перезапуска.

Следующие опции можно задавать в контексте клиента: **push**, **push-reset**, **iroute**, **ifconfig-push**, и **config**.

ccd-exclusive

Использовать наличие файла конфигурации (опция **client-config-dir**) в качестве условия аутентификации, для клиентов, устанавливающих соединение.

tmp-dir <dir> Задает директорию для временных файлов. Эта директория будет использоваться скриптами, заданными в опции **client-connect**, для динамического создания конфигурационных файлов для определенных клиентов.

hash-size <r> <v>
Установить размер хеш-таблицы реальных адресов равной **<r>** и размер таблицы виртуальных адресов равной **<v>**. По умолчанию, обе таблицы имею размер 256 записей.

bcast-buffers <n>
Выделить **<n>** буферов для широковещательных датаграмм (по умолчанию - 256).

tcp-queue-limit <n>
Максимальное количество TCP пакетов, которые могут быть поставлены в очередь (по умолчанию - 64).

Когда ПК «LirVPN» туннелирует данные из TUN/TAP устройства к удаленному клиенту через TCP соединение, существует возможность, что TUN/TAP устройство может передавать данные на большей скорости, чем поддерживает TCP соединение. Когда количество исходящих TCP пакетов поставленных в очередь достигает предела для соединения клиента, ПК «LirVPN» начинает сбрасывать исходящие пакеты направляемые клиенту.

max-clients <n>
Ограничить максимальное количество клиентов, которые могут одновременно подключиться к серверу.

max-routes-per-client <n>
Задает максимальное количество внутренних маршрутов на одного клиента равным **<n>** (по-умолчанию - 256). Эта опция предназначена для противодействия DoS-аттаке, в процессе которой аутентифицированный клиент отправляет серверу множество пакетов с поддельными уникальными MAC-адресами, вынуждая сервер каждый раз выделять память для расширения внутренней таблицы маршрутизации. Эта опция может быть использована в индивидуальном файле конфигурации клиента (обратитесь к описанию опции **client-config-dir**) или может быть автоматически создана с помощью скрипта **client-connect** для замещения глобального значения для конкретного клиента.

Эта опция затрагивает внутреннюю таблицу маршрутов ПК «LirVPN», а не таблицу маршрутов ОС.

connect-freq <n> <sec>
Разрешать максимум **<n>** новых соединений в **<sec>** секунд от клиентов. Эта опция используется для предотвращения DoS атак, в течение которых серверу посылается множество запросов на соединение, используя сертификаты, которые 100% не пройдут проверку.

Однако это не совершенное решение, потому что в случае реальной DoS атаки, клиенты, которые имеют право на подключение, также могут быть отброшены.

Для лучшей защиты от DoS атак в режиме сервера используйте опции **proto udp** и **tls-auth**.

learn-address <cmd>
Запустить скрипт или команду оболочки **<cmd>** для проверки достоверности виртуального адреса клиента или маршрутов.

<cmd> принимает на вход 3 параметра:

6.1 **operation** - «**add**», «**update**», или «**delete**»основываясь на том, был ли адрес добавлен, изменен или удален из внутренней таблицы маршрутов ПК «LirVPN».

6.2 **address** - Адрес который был узнан или наоборот. Может быть адресом IPv4 («198.162.10.14»), подсетью IPv4 («198.162.10.0/24»), или Ethernet MAC адресом (когда используется опция **dev tap**) - «00:FF:01:02:03:04».

6.3 **common name** - common name сертификата ассоциированного с клиентом, использующим этот адрес. Присутствует только при операциях «**add**» и «**update**», но не при «**delete**».

При использовании методов «**add**» или «**update**», если скрипт возвращает код ошибки отличный от 0, ПК «LirVPN» отклонит адрес, и не будет изменять внутреннюю таблицу маршрутов.

Обычно, <**cmd**> скрипт будет использовать предоставляемую информацию для создания соответствующих правил межсетевого экрана на интерфейсе TUN/TAP. Так как ПК «LirVPN» предоставляет ассоциацию между виртуальным IP адресом или MAC адресом и прошедшим проверку common name клиента, это делает возможным управлять правилами межсетевого экрана с помощью определяемого пользователем скрипта, относительно common name клиента, вместо виртуального адреса клиента.

auth-user-pass-verify <**script**> <**method**>

Данная опция требует от клиента предоставления имени пользователя/пароля (возможно в дополнении к сертификату клиента) для аутентификации.

ПК «LirVPN» выполнит <**script**> в качестве команды оболочки для проверки достоверности имени пользователя/пароля переданного клиентом.

Если в качестве параметра <**method**> указано «**via-env**», ПК «LirVPN» вызовет <**script**> с переменными окружения username и password, содержащими имя пользователя/пароль переданные клиентом. Обратите внимание, что этот метод небезопасен на некоторых платформах, которые делают доступными переменные окружения процесса другим непривилегированным процессам.

Если в качестве параметра <**method**> указано «**via-file**», ПК «LirVPN» запишет имя пользователя и пароль в первые две строчки временного файла. Имя файла будет передано <**script**> в качестве аргумента, и файл будет автоматически удален ПК «LirVPN» после выполнения скрипта. Местонахождение временного файла определяется опцией **tmp-dir**, и по умолчанию это текущая директория, если опция **tmp-dir** не задана. Для обеспечения безопасности, выберите в качестве параметра опции **tmp-dir** временное устройство хранения, например /dev/shm (если доступно), для предотвращения записи имени пользователя/пароля на жесткий диск.

Скрипт должен проверить имя пользователя и пароль, и возвратить код завершения 0, если аутентификация клиента прошла успешно, или код 1 для отклонения соединения.

Эта опция позволяет расширять возможности аутентификации ПК «LirVPN» в стиле плагинов.

Для защиты от специально сформированных имен пользователя и паролей (которые могут нести деструктивные функции), имя пользователя может содержать только следующие символы: буквы, цифры, подчерк («_»), тире («-»), точка («.»), или «собаку» («@»). Пароль может содержать из любых печатных символов кроме CR или LF. Любой недопустимый символ, как в имени пользователя, так и в пароле будет заменен на подчерк («_»).

Необходимо уделить особое внимание скриптам, для избегания проблем с безопасностью при использовании этих строк. Никогда не используйте эти строки таким образом, что они могут быть выполнены интерпретатором оболочки.

client-cert-not-required

Не использовать сертификат клиента для аутентификации, клиент будет проходить аутентификацию, используя только имя пользователя/пароль. Имейте в виду, что использование этой опции более опасно, чем запрос сертификатов от всех клиентов.

Если вы используете эту опцию, вся ответственность за аутентификацию клиентов ложится на скрипт проверки имени пользователя/пароля (опция **auth-user-pass-verify**), поэтому имейте в виду, что ошибки в этом скрипте потенциально могут скомпрометировать защиту вашей ЗВКС.

Если эта опции не задана, но вы используете опцию **auth-user-pass-verify**, тогда ПК «LirVPN» будет выполнять двойную аутентификацию. Проверка клиентского сертификата и скрипт проверки имени пользователя/пароля (опция **auth-user-pass-verify**) должны завершится успехом, для того чтобы клиент прошел аутентификацию и получил доступ к ЗВКС.

username-as-common-name

Если для аутентификации используется опция **auth-user-pass-verify**, использовать имя пользователя в качестве **common name**, а не **common name** из сертификата клиента.

6.3 Опции режима «Клиент»

Используйте режим «**клиент**» при подключении к серверу ПК «LirVPN», в конфигурации которого указаны опции **server**, **server-bridge**, или **mode server**.

client

Опция упрощает настройку ПК «LirVPN» в режиме «**клиент**». Задание данной опции эквивалентно заданию следующих опций:

pull
tls-client

pull

Эта опция должны быть задана на узле, который подключается к серверу в режиме концентратора. Она информирует ПК «LirVPN», что он должен принимать опции, переданные ему сервером, учитывая то, что они входят в список опций, которые можно передавать таким способом (заметьте, что опция **client** включает в себя опцию **pull**).

В частности, опция **pull** дает серверу возможность передавать сетевые маршруты клиентам, поэтому вы не должны использовать опции **pull** или **client** в ситуациях, когда вы не доверяете серверу управление таблицами маршрутизации клиентов.

auth-user-pass [<up>]

Пройти аутентификацию на сервере, используя имя пользователя/пароль. Опциональный параметр <**up**> - имя файла содержащего имя пользователя/пароль на 2-х строках.

Если параметр <**up**> не указан, имя пользователя/пароль будут запрошены с консоли. В настройках сервера необходимо указать опцию **auth-user-pass-verify** для проверки имени пользователя/пароля переданного клиентом.

auth-retry <type>

Определяет, как ПК «LirVPN» реагирует на ошибки проверки имени пользователя/пароля, в случаях получения сообщения **AUTH_FAILED** от сервера или ошибки проверки пароля на закрытый ключ.

Обычно используется для предотвращения возникновения критической ошибки на стороне клиента, и для разрешения запросов на имя пользователя/пароль в случае ошибки.

Сообщение **AUTH_FAILED** генерируется сервером, если аутентификация клиента с использованием опции **auth-user-pass** не удалась, или если скрипт **client-connect** (на стороне сервера) возвращает ошибку при попытке подключении клиента.

Параметр <**type**> может принимать следующие значения:

- **none** - Процесс ПК «LirVPN» завершиться с сообщением о критической ошибке (действие по умолчанию)
- **nointeract** - клиент попытается установить соединение заново без запроса имени пользователя/пароля (опция **auth-user-pass**). Используйте эту опцию для необслуживаемых клиентов.
- **interact** - клиенту необходимо ввести имя пользователя/пароль (опция **auth-user-pass**) и/или пароль на закрытый ключ перед попыткой установки соединения.

Заметьте, что эта опция не может быть передана клиенту, в тоже время её можно изменять с помощью управляющего интерфейса.

explicit-exit-notify [<n>]

В режиме «**клиент**» с использованием протокола UDP или в режиме точка-точка, посылать серверу/клиенту уведомление о выходе, если туннель находится в процессе перезапуска или процесс ПК «LirVPN» завершился. В режиме «**клиент**», в случае выхода/перезапуска, эта опция говорит серверу немедленно закрыть сеанс клиента, вместо того, чтобы ожидать таймаута соединения. Параметр <**n**> (по умолчанию - 1) задает максимальное количество попыток, которые предпримет клиент для отправки сообщения о выходе.

6.4 Опции шифрования

Эти опции применимы как для туннелей с симметричным шифрованием, так и для режима TLS (настройки узлов должны быть совместимы друг с другом).

secret <file> [<direction>]

Включить режим симметричного шифрования (не-TLS). Используется файл разделяемого секрета <**file**>, который формируется с помощью опции командной строки **genkey**.

Опциональный параметр <**direction**> включает использование 4 различных ключей (HMAC-send, cipher-encrypt, HMAC-receive, cipher-decrypt), вследствие чего каждый поток данных имеет различный набор HMAC и ключей шифрования. Это дает некоторые преимущества с точки зрения безопасности, включая блокирование некоторых типов DoS атак и атак с повторением пакетов.

Если параметр <**direction**> не задан, 2 ключа используются в двух направлениях, один для HMAC, а другой для шифрования/расшифровки.

Параметр <**direction**> должен всегда быть согласован с каждой стороной туннеля, то есть одна сторона должна использовать «**0**», а другая «**1**», или параметр не должен быть задан ни на одной стороне.

Параметру <**direction**> необходимо, чтобы <**file**> содержал 2048-ми битный ключ.

Режим симметричного шифрования имеет некоторые преимущества и основной из них, это простота настройки.

В это режиме не используются сертификаты, не нужен УЦ, или сложные процедуры установки соединения и протоколы. Единственное требование, это наличие защищенного канал передачи данных с удаленным узлом (например, ssh) для передачи секретного ключа. Это требование, а также тот факт, что этот секретный ключ никогда не меняется (если только вы не сформируете новый), делает этот метод менее защищенным в отличие от режима TLS. Если злоумышленнику удастся получить доступ к вашему секретному ключу, все данные, которые когда-либо были зашифрованы с его использованием, будут скомпрометированы. В режиме TLS наоборот: даже если злоумышленнику удастся получить доступ к закрытому ключу, он не получит информации помогающей ему расшифровать данные предыдущих сессий.

Другое преимущество режима симметричного шифрования - этот протокол с очень простой процедурой установки соединения, не содержащей никаких признаков того, что это пакеты относящиеся к ЗВКС и созданные ПК «LirVPN». Любой, кто будет прослушивать линию связи, не увидит ничего, кроме непонятных наборов байт.

auth <alg>

Аутентифицировать пакеты с помощью HMAC, используя алгоритм ХЭШ-функции <**alg**>. HMAC это широко используемый механизм контроля целостности который использует данные, надежную ХЭШ-функцию и ключ, для вычисления контрольной суммы сообщения.

ПК «LirVPN» сначала шифрует пакет, а потом вычисляет HMAC от зашифрованного пакета.

В режиме симметричного шифрования, ключ HMAC хранится в файле, сформированном с помощью опции **genkey**. В режиме TLS, ключ HMAC формируется динамически и передается узлам с помощью канала управления TLS. Если ПК «LirVPN» получает неправильный пакет HMAC, он его отбрасывает. HMAC обычно добавляет 16 или 20 байт на пакет. Установите параметр <**alg**> равным «**none**» для отключения аутентификации.

cipher <alg>

Шифровать пакеты с помощью алгоритма шифрования <**alg**>.

Для вывода информации об алгоритмах доступных ПК «LirVPN», используйте опцию командной строки **show-ciphers**.

ПК «LirVPN» поддерживает режимы шифрования CBC, CFB, и OFB.

Установите параметр <**alg**> равным **«none»** для создания туннеля без шифрования.

keysize <n>

Длина ключа шифрования в битах (не обязательная опция). Если опция не используется, длина равна размеру ключа по умолчанию, для выбранного алгоритма. Опция **show-ciphers** показывает все доступные алгоритмы шифрования СКЗИ «LirSSL», длины ключей по умолчанию, и алгоритмы, для которых может быть изменена длина ключа.

engine [<engine-name>]

Использовать сторонний модуль реализации криптографических функций поддерживаемый библиотекой СКЗИ «LirSSL».

Если задан параметр <**engine-name**>, использовать определенный модуль. Используйте опцию **show-engines** для вывода списка доступных на данный момент сторонних модулей реализации криптографических функций поддерживаемых библиотекой СКЗИ «LirSSL».

no-replay

Отключить в ПК «LirVPN» защиту от атак на протокол с повторной передачей («replay attacks»). Не используйте эту опцию если вы не согласны получить более высокую скорость работы туннеля в обмен на уменьшение защищенности.

ПК «LirVPN» предоставляет защиту от атаки на протокол с повторной передачей («replay attacks») в передаваемых датаграммах по умолчанию.

Защита от атак с повторной передачей осуществляется путем добавления к каждой исходящей датаграмме уникального идентификатора. Уникальность идентификатора гарантируется для каждого используемого ключа. Узел получивший датаграмму проверяет уникальность идентификатора. Если такой идентификатор уже был получен в предыдущей датаграмме, ПК «LirVPN» отбрасывает этот пакет. Защита от атак с повторной передачей является важной составляющей защиты от атак типа «SYN flood». В процессе этой атаки производится перехват злоумышленником TCP SYN пакетов и затем производится множественная повторная отправка этого пакета получающей стороне.

Реализация защиты от атак с повторной передачей отличается для разных режимов управления ключами.

В режиме с разделяемого секрета или при использовании режимов шифрования CFB или OFB, ПК «LirVPN» использует 64-х битный уникальный идентификатор который сочетает в себе метку времени и инкрементальный порядковый номер.

При использовании протокола TLS для обмена ключами и при использовании режима шифрования CBC, ПК «LirVPN» использует только 32-х битную последовательность чисел без метки времени, так как ПК «LirVPN» может гарантировать уникальность этого идентификатора для каждого ключа. Если номер последовательности приближается к обнулению, ПК «LirVPN» инициирует смену ключей, так же как в протоколе IPSec.

replay-window <n> [<t>]

Использовать в защите от атак с повторной передачей скользящее окно размером <**n**> и временное окно длительностью <**t**> секунд.

По-умолчанию параметр <**n**> равен 64, а <**t**> - 15 секунд.

Эта опция имеет значение только при использовании протокола UDP.

Когда ПК «LirVPN» использует в качетсве транспортного протокола UDP, существует вероятность, что UDP-пакет будет потерян или доставлен вне очереди. Так как ПК «LirVPN» эмулирует физический сетевой уровень (как IPSec), он принимает пакеты идущие вне очереди и доставляет их в той последовательности, в которой они попадали в TCP/IP стэк, при условии что пакеты удовлетворяют следующим условиям:

6.1 Пакет не является повторением (только если не задана опция **no-replay**, которая отключает защиту от атак повторения)

6.2 Если пакет пришел вне очереди, он будет принят только в том случае, если разница между его последовательным номером и самым большим номером принятого пакета не превышает **n**.

6.3 Если пакет пришел вне очереди, он будет принят только в том случае, если он пришел не позже чем через **t** секунд после любого пакета с большим номером.

Если используется сетевое соединение с большой пропускной способностью, но с большими задержками, то рекомендуется использовать большие значения **n**. Для спутниковой связи это часто необходимо.

Если ПК «LirVPN» запущен с опцией **verb 4**, в сообщениях журнала будут такого вида записи: "Replay-window backtrack occurred [x] каждый раз когда разница между последовательным номером пакета и самым большим номером принятого пакета превышает **n**. Эти сообщения можно использовать для калибровки значения **n**.

mute-replay-warnings

Подавлять вывод предупреждающих сообщений о повторных пакетах. Обычно в сетях Wi-Fi такие сообщения являются ложной тревогой. Эта опция сохраняет действие функции защиты от атак с повторной передачей пакетов, без вывода предупреждений о дублирующихся пакетах.

replay-persist <file>

Сохранять состояние защиты от атак с повторной передачей пакетов между перезапусками ПК «LirVPN», используя файл <**file**> для сохранения и загрузки состояния.

Эта опция усиливает защиту против атак с повторной передачей пакетов, особенно если ПК «LirVPN» часто останавливается и запускается снова (например, используется опция **inetd**).

При использовании этой опции в <**file**> записывается текущее состояние защиты от атак с повторной передачей пакетов (то есть самая последняя метка времени и порядковый номер полученного пакета от удаленного узла), поэтому, если ПК «LirVPN» будет остановлен и перезапущен, он не примет повторные пакеты, которые уже были получены в предыдущей сессии.

Эта опция имеет значение только при включенной защите от атак с повторной передачей пакетов (по умолчанию), и вы используете опцию **secret** (режим симметричного шифрования) или режим TLS с опцией **tls-auth**.

no-iv

Отключить использование IV (вектор инициализации) ПК «LirVPN». Не используйте эту опцию если вы не согласны получить более высокую скорость работы туннеля в обмен на уменьшение защищенности.

ПК «LirVPN» использует IV по умолчанию, и требует его для режимов шифрования CFB и OFB (которые абсолютно не защищены без него). Использование IV необходимо когда несколько сообщений шифруются/расшифровываются на одном ключе.

Значение IV вырабатывается по разному в зависимости от режима шифрование.

В режиме CBC, ПК «LirVPN» использует псевдослучайный IV для каждого пакета.

В режимах CFB/OFB, ПК «LirVPN» использует порядковый номер последовательности и метку времени в качестве IV. Дело в том, что в режимах CFB/OFB, ПК «LirVPN» использует оптимизацию использования размера датаграммы которая подразумевает использование уникального идентификатора датаграммы системы защиты от атак с повторной передачей пакетов в качестве IV.

6.5 Опции режима TLS

Режим TLS это наиболее мощный режим шифрования в ПК «LirVPN» как с точки зрения безопасности, так и удобства использования.

Режим TLS функционирует посредством создания канала управления и канала данных, которые объединяются и используют один TCP/UDP порт. ПК «LirVPN» начинает TLS сессию через управляющий канал и использует его для обмена ключами шифрования и HMAC ключами, для защиты передаваемых по туннелю данных.

Режим TLS использует надежный и устойчивый к ошибкам канал использующий UDP соединение, для всех данных передаваемых через канал управления, в то время как канал данных,

через который передаются зашифрованные данные, не использует посредников. В результате: быстрый канал передачи данных, который пересылается через UDP протокол, только с накладными расходами на шифрование, расшифровку и функции HMAC, и канал управления, который предоставляет все функции безопасности TLS, включая аутентификацию, основанную на сертификатах и распределении ключей Diffie-Hellman.

Для использования режима TLS, каждый узел ПК «LirVPN» должен иметь свой сертификат/ключевую пару (опции **cert** и **key**), подписанный корневым сертификатом удостоверяющего центра (опция **ca**).

Когда происходит соединение двух узлов использующих ПК «LirVPN», каждый предоставляет свой локальный сертификат другому узлу. Далее каждый узел проверяет, предоставил ли удаленный узел сертификат, подписанный корневым сертификатом удостоверяющего центра, указанным в опции **ca**.

Если эта проверка успешно завершилась на обоих узлах, устанавливается TLS соединение, далее оба узла ПК «LirVPN» обмениваются временными сессионными ключами, и шифрованный туннель начинает функционировать.

tls-server

Включить режим TLS и выбрать роль сервера в момент TLS соединения. Выбор роли нужен только для установки TLS соединения.

tls-client Включить TLS и выбрать режим клиента в момент установки TLS соединения.

ca <file>

Опция задает имя файла сертификата Удостоверяющего центра (УЦ) в формате PEM, также называемом корневым сертификатом. Этот файл может содержать несколько сертификатов в формате PEM.

dh <file> Опция задает имя файла, содержащего параметры Diffie Hellman в формате PEM (необходим только в случае использовании опции **tls-server**).

Используйте файл dh1024.pem включенный в дистрибутив ПК «LirVPN». Параметры Diffie Hellman могут быть доступны публично без угрозы безопасности.

cert <file>

Подписанный сертификат узла в формате PEM - должен быть подписан УЦ, чей сертификат указан в опции **ca**.

Каждый узел, устанавливающий соединение в режиме TLS должен иметь свой собственный сертификат и закрытый ключ. В дополнении к этому, каждый сертификат должен быть подписан ключом УЦ, чей публичный ключ хранится в файле указанном в опции **ca**.

key file

Закрытый ключ узла в формате PEM.

pkcs12 <file>

Опция задает путь до контейнера PKCS#12 содержащего закрытый ключ, сертификат и корневой сертификат УЦ. Эта опции может быть использована вместо опций **ca**, **cert**, и **key**.

cryptoapicert

Загружает сертификат и закрытый ключ из Windows Certificate System Store (работает только на ОС Windows).

Используйте эту опцию вместо опций **cert** и **key**.

Это позволяет использование любой смарт-карты, поддерживаемой ОС Windows.

key-method <m>

Использовать метод <**m**> для согласования ключей канала данных. Методы должны совпадать на обоих сторонах туннеля.

После того как ПК «LirVPN» устанавливает TLS соединение, формируется новый набор ключей для защиты данных в туннеле и пересылается через это TLS соединение.

Метод 1 - обе стороны формируют случайные ключ шифрования и HMAC-send ключ которые передаются удаленному узлу через канал TLS.

Метод 2 - клиент формирует случайных ключ. И клиент и сервер также формируют случайную последовательность. Весь ключевой материал передается по TLS-соединению. Сами ключи формируются используя функцию TLS PRF, используя источники энтропии как от клиента так и от сервера. Метод 2 очень похож на процесс генерации ключей, используемый в протоколе TLS 1.0.

Заметьте, что в режиме TLS различаются 2 уровня манипуляции с ключами:

- Устанавливается TLS соединение, обе стороны предоставляют свои сертификаты, и происходит их проверка (также могут проверяться другие данные, например, пароли). Параметр опции **key-method** не имеет влияния на этот процесс.

- После установки TLS соединения, происходит обмен ключами канала данных через это соединение. Здесь, опция **key-method** определяет метод согласования ключей канала данных.

tls-cipher <l>

В параметре <l>, задается перечень всех разрешенных алгоритмов шифрования TLS разделенных двоеточиями («**:**»). Если вам необходим высокий уровень безопасности, вы можете установить это параметр вручную, для предотвращения атаки «понижение версии»(«version rollback»). В процессе этой атаки «человек-посередине»(«man-in-the-middle») пытается заставить узлы перейти на алгоритм с самой низкой криптостойкостью из тех, которые они поддерживают. Используйте опцию командной строки **show-tls** для вывода списка всех поддерживаемых алгоритмов шифрования TLS.

tls-timeout <n>

Время ожидания передачи пакета через канал управления TLS, если не было получено подтверждение от удаленного узла (по умолчанию - 2 секунды).

Когда узел ПК «LirVPN» посылает управляющий пакет удаленному узлу, он ожидает получения подтверждения в течении <**n**> секунд, иначе происходит повторная передача пакета.

Эта опция применима только к пакетам, передающимся через канал управления. ПК «LirVPN» никогда не подтверждает получения, не передает повторно и не ставит в очередь пакеты, передающиеся через канал данных (который отвечает за шифрование данных в туннеле) потому, что это обеспечивают сетевые протоколы через которые функционирует туннель, например, TCP.

reneg-bytes <n>

Провести замену ключа канала после получения или отправки <**n**> байт данных (по умолчанию отключено). ПК «LirVPN» позволяет задать время действия ключа в качестве критериев могут выступать:

- количество зашифрованных/расшифрованных байт;

- количество полученных и принятых пакетов;
- время в секундах с момента последней замены ключей.

Замена ключа будет произведена если любой из этих критериев вступает в действие на каком либо узле.

reneg-pkts <n>
Провести замену ключа канала данных после <**n**> полученных и принятых пакетов (по умолчанию отключено).

reneg-sec <n>
Провести замену ключа канала данных после <**n**> секунд (по умолчанию - 3600).

hand-window <n>
Handshake Window - обмен ключами по протоколу TLS должен завершиться в течении <**n**> секунд после инициализации соединения одним из узлов (по-умолчанию - 60 секунд). Если установить соединение не удалось - ПК «LirVPN» пробует сбросить соединение с удаленной стороной и пробуем еще раз. Даже с учетом неудачных соединений ПК «LirVPN» будет использовать ключ не дольше количества секунд, указанных опцией **tran-window**, для того чтобы поддержать непрерывность передачи данных по туннелю.

tran-window <n>
Transition window - старый ключ может существовать еще <**n**> секунд после того, как был выработан новый ключ (по-умолчанию 3600 секунд). Это свойство позволяет осуществить «мягкий» переход от старого ключа к новому и исключает проблемы, которые могут возникнуть при передаче по туннелю данных в процессе смены ключа.

single-session
После установки соединения с удаленным узлом, запретить любые новые соединения. Использование этой опции означает, что удаленный узел не сможет установить соединение, закрыть его, и после этого установить соединение заново.

Если служба (демон) ПК «LirVPN» перезапустится по сигналу или в результате действия опции **ping-restart**, будет разрешено одно новое подключение.

Опция **single-session** может быть использована совместно опцией **ping-exit** или **inactive** для создания динамического туннеля ПК «LirVPN», который будет закрыт после окончания передачи данных.

tls-exit
Выйти в случае неудачи установки TLS соединения.

tls-auth <file> [<direction>]
Добавляет дополнительный уровень HMAC аутентификации поверх управляющего канала TLS для защиты от DoS атак.

Опция **tls-auth** задействует так называемый «HMAC firewall» на TCP/UDP порту ПК «LirVPN», после чего пакеты канал управления TLS несущие неправильную HMAC сигнатуру могут быть немедленно отброшены без ответной реакции.

Обязательный параметр <**file**>, это файл ключа в одном из двух форматов:

- Секретный ключ ПК «LirVPN» формируемый опцией командной строки **genkey** (если используется параметр <**direction**>, данный формат файла является обязательным).

- Произвольный файл. В этом случае ключ HMAC будет сформирован путем вычисления значения ХЭШ-функции от этого файла.

ПК «LirVPN» сначала попробует использовать первый формат, и если произойдет ошибка обработки файла в качестве секретного ключа, будет использован формат 2.

Обратитесь к описанию опции **secret** для подробной информации об использовании необязательного параметра **direction**.

Использование опции **tls-auth** рекомендуется, когда ПК «LirVPN» принимает пакеты от любого IP адреса, например, если опция **remote** не задана, или опция **remote** используется совместно с опцией **float**.

В качестве дополнительной защиты, ПК «LirVPN» предоставляет специальный уровень аутентификации поверх канала управления TLS. Каждый пакет, поступающий на канал управления проходит проверку на HMAC сигнатуру и на уникальный номер для «replay protection». Сигнатура также помогает в защите от DoS (Отказ в Обслуживании) атак. Главным правилом в предотвращении DoS атак является минимизирование количества ресурсов, которые выделяются клиенту, устанавливающему соединение, который еще не прошел аутентификацию.

Опция **tls-auth** реализует это, подписывая каждый пакет, передающийся через канал управления TLS подписью HMAC, включая пакеты которые еще не прошли аутентификацию TLS. В результате пакеты без корректной подписи отбрасываются сразу же при получении, до того как на их обработку будут выделены ресурсы системы. Действие опции **tls-auth** может быть усилено добавлением в конфигурацию опции **replay-persist**, которая сохраняет состояние «replay protection» ПК «LirVPN» в файле, и эта информация не теряется в случае перезапуска ПК «LirVPN».

Следует отметить, что эта опция является опциональной и не дает узлу ничего, кроме как возможность установить TLS соединение. Эта опция не используется для шифрования или аутентификации каких-либо данных проходящих через туннель.

askpass [<file>]

Взять пароль на сертификат с консоли или из файла <**file**> до перехода в режим службы (демона).

Для дополнительной защиты, возможна защита закрытого ключа паролем. Это значит что каждый раз при старте ПК «LirVPN» в качестве службы (демона) будет необходим ввод пароля. Опция **askpass** позволяет запустить ПК «LirVPN» из командной строки. Она запросит пароль перед переходом ПК «LirVPN» в режим службы (демона).

Если задан параметр файл <**file**>, пароль будет взят из первой строчки файла <**file**>. Имейте в виду, что хранение пароля в файле в известной степени аннулирует дополнительную безопасность, предоставляемую зашифрованными ключами.

auth-nocache

Не кэшировать **askpass** или **auth-user-pass** имена пользователей/пароли в виртуальной памяти.

Если опция указана, ПК «LirVPN» сразу же удаляет из памяти введенные имя пользователя и пароль после их использования. В этом случае, когда ПК «LirVPN» потребуется имя пользователя и пароль, они будут запрошены со стандартного ввода, что может случаться множество раз в течение работы ПК «LirVPN».

Эта опция не действует на имена пользователей и пароли, задаваемые с помощью опции **http-proxy**. Они всегда кэшируются.

tls-verify <cmd>

Выполнить команду оболочки <**cmd**> для проверки X509 name, ожидающего TLS соединения, которое уже прошло все остальные проверки (кроме проверки на отзыв с помощью опции **crl-verify**; проверка на отзыв выполняется после проверки **tls-verify**).

Скрипт <**cmd**> должен возвращать 0 для продолжения TLS соединения, или 1 в случае неудачи. <**cmd**> выполняется как:

cmd certificate_depth X509_NAME_oneline

Опция полезна, если сертификат узла, который вы хотите проверить, подписан УЦ, который подписал уже много других сертификатов, и есть необходимость доверять только определенным сертификатам. Данная опция позволяет создать скрипт проверки имени X509 сертификата, и принять решение о его допуске.

Обратитесь к секции «**Переменные окружения**», чтобы узнать о дополнительных параметрах передающихся как переменные окружения.

Заметьте, что <**cmd**> может быть командой с множеством аргументов, в этом случае все аргументы сформированные ПК «LirVPN» будут добавлены к <**cmd**>, для составления командной строки, которая будет передана скрипту.

tls-remote <name>

Принимать запросы на подключение только от узлов с X509 name или common name равными параметру <**name**>. Удаленный узел также должен пройти все остальные проверки.

Параметр name также может быть префиксом к common name, например, если вы хотите, чтобы клиент принимал подключения только от «Server-1», «Server-2» и так далее, вы можете использовать опцию **tls-remote Server**

Использование префикса common name это удобная альтернатива ведению COC (списка отозванных сертификатов) на клиенте, так как это позволяет клиенту отклонять все сертификаты кроме тех, которые ассоциированы с определенными серверами.

Опция **tls-remote** является полезной заменой опции **tls-verify** для проверки удаленного узла, потому что опция **tls-remote** работает также с опцией **chroot**.

ns-cert-type <client|server>

Требует, чтобы сертификат узла был подписан с явным указанием значения **nsCertType** «**client**» или «**server**».

Это полезная опция безопасности для клиентов, для обеспечения гарантии того, что узел, к которому они подключаются, действительно является сервером.

Если поле **nsCertType** в сертификате сервера имеет значение «server», тогда клиенты могут проверить это с помощью опции **ns-cert-type server**.

Это важная мера предосторожности для защиты от атак
«человек-посередине» («man-in-the-middle»), при которых авторизованные клиенты пытаются установить подключение к другим клиентам, представляясь им сервером. Атака легко предотвращается проверкой клиентами серверного сертификата (опции **ns-cert-type**, **tls-remote**, или **tls-verify**).

crl-verify <crl>

Проверить сертификат узла используя файл <**crl**> в формате PEM.

COC (список отозванных сертификатов) используется в случае компрометации определенного ключа, но вся ИОК остается в рабочем состоянии.

Допустим, вы используете ИОК, состоящую из УЦ, корневого сертификата, и нескольких клиентских сертификатов. Если был украден ноутбук, содержащий закрытый ключи сертификат, путем добавления украденного сертификата в файл COC, вы запрещаете любые соединения с использованием украденного сертификата, сохраняя работоспособность ИОК.

Создание ИОК заново требуется только в том случае, если был скомпрометирован закрытый ключ корневой сертификата УЦ.

6.6 Информация LirSSL

show-ciphers
(Опция командной строки) Показывает все возможные алгоритмы шифрования для использования совместно с опцией **cipher**.

show-digests
(Опция командной строки) Показывает все ХЭШ-алгоритмы для использования совместно с опцией **auth**.

show-tls
(Опция командной строки) Показывает все алгоритмы шифрования TLS
(TLS используется только в управляющем канале). Алгоритмы шифрования TLS будут выведены на экран, начиная от самого защищенного, заканчивая менее защищенным.

show-engines
(Опция командной строки) Показывает доступные на данный момент сторонние модули реализации криптографических функций поддерживаемые библиотекой СКЗИ «LirSSL».

6.7 Опции управления постоянными туннелями TUN/TAP

mktun
(Опция командной строки) Создать постоянный туннель (ОС Linux). Обычно TUN/TAP туннели существуют только на период времени работы приложения.

Одним из преимуществ постоянных туннелей, это то, что их использование делает ненужным использование скриптов **up** и **down** для выполнения соответствующих команд **ifconfig** и **route**. Эти команды могут быть добавлены в тот же самый скрипт командной оболочки, который запускает или останавливает ПК «LirVPN».

Другое преимущество состоит в том, что открытые соединения, работающие через постоянный TUN/TAP туннель, не будут обрываться в случае перезапуска ПК «LirVPN». Это может быть полезно для предоставления беспрерывной связи через туннель в случае изменения публичного IP адреса в результате работы DHCP (обратитесь к опции **ipchange**).

Недостаток постоянных туннелей состоит в том, что сложнее автоматически конфигурировать их значения MTU (обратитесь к опциям **link-mtu** и **tun-mtu**).

На некоторых платформах, например ОС Windows, TAP-Win32 туннели постоянны по умолчанию.

rmtun
(Опция командной строки) Удалить постоянный туннель.

dev <tunX | tapX>
Задать TUN/TAP устройство.

6.8 Опции ПК «LirVPN» работающего под управлением ОС Windows

ip-win32 <method>

При использовании опции **ifconfig** на ОС Windows, установить адрес IP адрес и маску сети TAP-Win32 адаптера, используя метод <**method**>. Не используйте эту опцию, если опция **ifconfig** не задана.

Доступные методы:

- **manual** - Не устанавливать IP адрес и маску сети автоматически. Вместо этого вывести сообщение на консоль говорящее о том, что пользователь должен сам задать настройки TUN/TAP адаптера и показывает IP адрес/маску сети, которые ПК «LirVPN» ожидает увидеть в настройках TUN/TAP адаптера.

- **dynamic [<offset>] [<lease-time>]** - (По умолчанию) Автоматически назначить IP адрес и маску подсети путем ответа на DHCP запрос сформированный ядром ОС. Этот метод, вероятно, является самым «правильным» для задания TCP/IP настроек, так как он использует широко известный протокол DHCP. Для использования этого метода необходимо выполнение двух условий:

  – В настройках протокола TCP/IP для TAP-Win32 адаптера должен быть выбран пункт «Получить IP адрес автоматически»;

  – ПК «LirVPN» должен получить IP адрес в подсети для использования в качестве адреса виртуального сервера DHCP.

  По умолчанию в режиме **dev tap**, ПК «LirVPN» использует обычно не используемый первый адрес в подсети. Например, в подсети 192.168.4.0 с маской подсети 255.255.255.0, ПК «LirVPN» будет использовать IP адрес 192.168.4.0 в качестве адреса виртуального DHCP сервера. В режиме **dev tun**, ПК «LirVPN» будет использовать адрес удаленного узла в качестве адреса DHCP сервера.

  Необязательный параметр <**offset**> это целое число > -256 и < 256 и по умолчанию принимает значение равное 0. Если <**offset**> положительный, адрес виртуального DHCP сервера будет задан как IP адрес сети + <**offset**>. Если <**offset**> отрицательный, адрес виртуального DHCP сервера будет задан как широковещательный IP адрес сети + <**offset**>.

  Команда ОС Windows «**ipconfig /all**» может быть использована для вывода информации о том, какой адрес DHCP сервера используется в данный момент. Убедитесь, что ПК «LirVPN» будет использовать другой свободный адрес. Стоит сказать, что разные процессы ПК «LirVPN», включая разные узлы одного и того же соединения, могут делать один и тот же адрес виртуального DHCP сервера.

  Параметр <**lease-time**> - время аренды DHCP назначения для TAP-Win32 адаптера, и задается в секундах. Обычно используется очень большое время, потому что это предотвращает потерю маршрутов для TAP-Win32 адаптера в случаях долгого неиспользования системы. Время аренды по умолчанию - 1 год.

- **netsh** - автоматически установить IP адрес и маску сети используя команду Windows «**netsh**». Этот метод работает корректно на Windows XP, но не на Windows 2000.

- **ipapi** - автоматически установить IP адрес и маску сети используя Windows IP Helper API. Если вы используете этот метод, настройки TCP/IP для TAP-Win32 адаптера лучше оставить по умолчанию, то есть «Получить IP адрес автоматически».

route-method <m>

Опция задает метод <**m**>, который будет использоваться для добавления сетевых маршрутов на ОС Windows.

Доступные методы:

- **ipapi** (по умолчанию) - Используя IP helper API;
- **exe** - Используя команду **route.exe**.

dhcp-option type [<parm>]

Устанавливает дополнительные опции TCP/IP для TAP-Win32 адаптера, должна использоваться совместно с опцией **ip-win32 dynamic**. Эта опция может быть использована для задания дополнительных опций TCP/IP для TAP-Win32 адаптера, и особенно полезна для настройки доступа узла ПК «LirVPN» к Samba серверам через ЗВКС.

- **DOMAIN name** - Установить специфичный DNS суффикс для соединения.
- **DNS addr** - Установить адрес первичного DNS сервера. Повторите эту опцию, если необходимо задать адрес вторичного DNS сервера.
- **WINS addr** - Установить адрес первичного WINS сервера (NetBIOS over TCP/IP Name Server). Повторите эту опцию, если необходимо задать адрес вторичного WINS сервера.
- **NBDD addr** - Установить адрес первичного NBDD сервера (NetBIOS over TCP/IP Datagram Distribution Server). Повторите эту опцию, если необходимо задать адрес вторичного NBDD сервера.
- **NTP addr** - Установить адрес первичного NTP сервера (Network Time Protocol). Повторите эту опцию, если необходимо задать адрес вторичного NTP сервера.
- **NBT type** - Установить типа узла NetBIOS over TCP/IP. Возможные параметры типы:
  - **1 = b-node** - Использует только широковещательные сообщения для регистрации и разрешения имен
  - **2 = p-node** - Используют только сервер имен NetBIOS (WINS) для регистрации и разрешения имен. Широковещательные сообщения не используются, следовательно сеть не перегружается. Так как запрос адресуется непосредственно серверу имен, то он проходит через маршрутизатор. Основной проблемой является невозможность связи даже в локальной сети при выключенном сервере имен.
  - **4 = m-node** - Комбинация B и P узлов. Сначала используется широковещательное сообщение, в случае неудачи - обращение к серверу имен.
  - **8 = h-node** - Комбинация B и P узлов. Сначала идет обращение к WINS серверу, в случае неудачи используется широковещательное сообщение.
- **NBS scope-id** - Установить Идентификатор группы имен NetBIOS over TCP/IP. Идентификатор группы имен расширенный сервис имен для NetBIOS over TCP/IP. Идентификатор группы имен NetBIOS (NetBIOS scope ID) представляет собой строку символов (с учетом регистра), добавленную к имени NetBIOS. При этом общая длина имени не должна превышать 16 символов. Ресурсы NetBIOS внутри группы доступны только ее членам и недоступны извне. То есть для того, чтобы два хоста могли установить связь друг с другом по NBT, необходимо совпадение идентификаторов групп у этих хостов.

- **DISABLE-NBT** - Отключить поддержку Netbios через TCP/IP. Заметьте, если опция **dhcp-option** передается через опцию **push** клиентам, работающим под ОС отличной от Windows, эта опция будет сохранена в переменной окружения «**foreign_option_{n}**» перед вызовом скрипта **up**.

tap-sleep <n>

Заставляет ПК «LirVPN» ожидать <**n**> секунд сразу же после установки TAP-Win32 адаптера в состояние «**Подключен**».

Эта опция предназначена для решения проблем с опциями **ifconfig** и **ip-win32**, и используется для ожидания включения TAP-Win32 адаптера, перед тем как операции Windows IP Helper API будут применены к нему.

show-net-up

Выводит таблицу маршрутов системы и список сетевых адаптеров с точки зрения ПК «LirVPN» в системный файл журнала или просто в файл журнала, после того как TUN/TAP адаптер перешел в рабочее состояние и были добавлены сетевые маршруты.

dhcp-renew

Ask Windows to renew the TAP adapter lease on startup. This option is normally unnecessary, as Windows automatically triggers a DHCP renegotiation on the TAP adapter when it comes up, however if you set the TAP-Win32 adapter Media Status property to «Always Connected», you may need this flag.

dhcp-release

Ask Windows to release the TAP adapter lease on shutdown. This option has the same caveats as **dhcp-renew** above.

pause-exit

Выводить сообщение «press any key to continue» на консоль перед завершением ПК «LirVPN». Эта опция автоматически используется в случае запуска ПК «LirVPN» с помощью правого щелчка мыши на конфигурационном файле ПК «LirVPN» в программе «Проводник».

service <exit-event> [<0|1>]

Эта опция должна использоваться, когда ПК «LirVPN» автоматически запускается с помощью другой программы и соответственно интерактивное взаимодействие с пользователем через монитор и клавиатуру не возможно. В общем случае, у конечных пользователей никогда не должна возникнуть потребность в явном указании этой опции, так как она автоматически добавляется службой ПК «LirVPN», когда конкретная конфигурация ПК «LirVPN» выполняется в качестве службы.

<**exit-event**>, это имя объекта Windows global event object, и ПК «LirVPN» будет постоянно контролировать активность этого события и завершит свою работу при появлении данного сигнала.

Второй параметр задает начальное состояние <**exit-event**> и обычно равен 0.

Множество процессов ПК «LirVPN» могут быть одновременно запущены с одним и тем же параметром <**exit-event**>. В любом случае, управляющий процесс может послать сигнал <**exit-event**>, в результате которого все такие процессы ПК «LirVPN» завершатся.

Когда процесс ПК «LirVPN» стартует с заданной опцией **service**, окно вывода сообщений о работе и об ошибках не появляется, поэтому в таких случаях полезно использовать опции **log** или **log-append** для записи этих сообщений в файл.

show-adapters

(Опция командной строки) Показывает список доступных TAP-Win32 адаптеров, которые могут быть использованы совместно с опцией **dev-node**. На ОС отличных от Windows, команда **ifconfig** предоставляет схожую функциональность.

show-valid-subnets

(Опция командной строки) Показывает допустимые подсети для эмуляции **dev tun**. Так как TAP-Win32 драйвер представляется в ОС Windows в качестве Ethernet интерфейса, и так как TUN устройства являются устройствами «точка-точка» по своей сути, драйвер устройства TAP-Win32 должен соблюдать некоторые ограничения при выборе конечных адресов в TUN туннеле.

А именно, адреса, используемые в эмуляции TUN устройств должны быть двумя средними адресами в /30 подсети (маска подсети 255.255.255.252).

show-net

(Опция командной строки) Показывает таблицу маршрутов системы и список сетевых адаптеров с точки зрения ПК «LirVPN».

6.9 Скрипты и переменные окружения

ПК «LirVPN» экспортирует ряд переменных окружения, для использования в заданных пользователем скриптах.

6.9.1 Порядок выполнения скриптов

up

Выполняется после открытия TCP/UDP сокета и открытия TUN/TAP интерфейса.

tls-verify

Выполняется, когда клиент еще не прошел аутентификацию.

ipchange

Выполняется после аутентификации соединения, или после изменения IP адреса.

client-connect

Выполняется в режиме **mode server** сразу же после того как клиент прошел аутентификацию.

route-up

Выполняется после аутентификации соединения, либо сразу же, либо по истечении времени в секундах (опция **route-delay**).

client-disconnect

Выполняется в режиме **mode server** при закрытии клиентского соединения.

down

Выполняется после закрытия TCP/UDP сокетов и TUN/TAP интерфейсов.

learn-address

Выполняется в режиме **mode server** когда IPv4 адрес/маршрут или MAC адрес добавляется во внутреннюю таблицу маршрутов ПК «LirVPN».

auth-user-pass-verify

Выполняется в режиме **mode server** при установке соединения клиентом, когда клиент еще не прошел аутентификацию.

6.9.2 Типы строк и их преобразование

В определенных случаях, ПК «LirVPN» производит преобразование символов в строках. Любые символы, которые не входят в набор допустимых символов для конкретного типа строк будут заменены на подчерк («_»).

Это важная особенность (с точки зрения безопасности) используется для предотвращения передачи специально сформированных вредоносных строк от ненадежных источников в качестве параметров скриптам, а также сохранения в окружении, использования в качестве common name, использования в качестве имени файла, и так далее.

Короткое описание типов строк ПК «LirVPN» и набор допустимых символов для каждого типа приводится ниже:

X509 Names: буквенно-цифровые, подчерк («_»), тире («-»), точка («.»), «собака» («@»), двоеточие («:»), слеш («/») и равно («=»). Буквенно-цифровыми символами считаются те, для которых функция библиотеки C isalnum() возвращает положительный результат.

Common Names: буквенно-цифровые, подчерк («_»), тире («-»), точка («.») и «собака» («@»).

Имя пользователя проверяемого в опции auth-user-pass: Аналогично Common Name, с одной оговоркой: имя пользователя передается плагину **OPENVPN_PLUGIN_AUTH_USER_PASS_VERIFY** в необработанном виде.

Пароль используемый в опции auth-user-pass: Любые печатаемые символы кроме CR и LF. Печатаемыми символами считаются те, для которых функция библиотеки C isprint() возвращает положительный результат.

Имя файла получаемого из common name или имени пользователя для опции client-config-dir: буквенно-цифровые, подчерк («_»), тире («-»), и точка («.»), кроме «.» или «..» в случае если в строке присутствуют только эти символы. Символ «собака» («@») был добавлен для совместимости с классом символов common name.

Имена переменных окружения: буквенно-цифровые или подчерк («_»).

Значения переменных окружения: Любые печатаемые символы.

Для всех случаев, символы в строке, которые не отвечают требованиям этой строки, будут заменены на подчерк («_»).

6.9.3 Переменные окружения

Однажды установленная, переменная не меняет своего значения, пока не будет задано новое значение или не произойдет перезапуск программы.

В режиме «**сервер**», переменные окружения, устанавливаемые ПК «LirVPN», могут быть использованы только с теми клиентскими соединениями, с которыми они ассоциированы. Поэтому не должно быть ситуаций, при которых скрипты будут иметь доступ к устаревшим переменным, которые были установлены для других клиентских соединений.

bytes_received

Общее количество байт полученных от клиента за время его работы. Устанавливается до выполнения скрипта **client-disconnect**.

bytes_sent
Общее количество байт переданных клиенту за время его работы. Устанавливается до выполнения скрипта **client-disconnect**.

common_name
X509 common name клиента прошедшего аутентификацию. Устанавливается до выполнения скриптов **client-connect**, **client-disconnect**, и **auth-user-pass-verify**.

config
Имя первого конфигурационного файла указанного в параметре опции **config**. Устанавливается при старте ПК «LirVPN» и сбрасывается при поступлении сигнала **SIGHUP**.

daemon
Устанавливается в «**1**» если задана опция **daemon**, иначе в «**0**». Устанавливается при старте ПК «LirVPN» и сбрасывается при поступлении сигнала **SIGHUP**.

daemon_log_redirect
Устанавливается в «**1**» если заданы опции **log** или **log-append**, иначе в «**0**». Устанавливается при старте ПК «LirVPN» и сбрасывается при поступлении сигнала **SIGHUP**.

dev
Фактическое имя TUN/TAP устройства, включая unit number, если он существует. Устанавливается перед выполнением скриптов **up** или **down**.

foreign_option_{n}
Опция, переданная с помощью опции **push** клиенту её не поддерживающему, например опция **dhcp-option** на ОС отличных от Windows, будет записана в эту переменную окружения до выполнения скрипта **up**.

ifconfig_broadcast
Широковещательный адрес виртуального Ethernet сегмента, который извлекается из опции **ifconfig**, если используется опция **dev tap**. Устанавливается до выполнения ПК «LirVPN» команд **ifconfig** или **netsh** (ОС Windows), которые обычно вызываются до выполнения скрипта **up**.

ifconfig_local
Локальный IP адрес туннеля ЗВКС, задаваемый в опции **ifconfig** (первый параметр). Устанавливается до выполнения ПК «LirVPN» команд **ifconfig** или **netsh** (ОС Windows), которые обычно вызываются до выполнения скрипта **up**.

ifconfig_remote
IP адрес удаленного конца туннеля ЗВКС, задаваемый в опции **ifconfig** (второй параметр), когда используется опция **dev tun**. Устанавливается до выполнения ПК «LirVPN» команд **ifconfig** или **netsh** (ОС Windows), которые обычно вызываются до выполнения скрипта **up**.

ifconfig_netmask
Маска подсети виртуального Ethernet сегмента, которая указана в качестве второго параметра опции **ifconfig**, если используется опция **dev tap**. Устанавливается до выполнения ПК

«LirVPN» команд **ifconfig** или **netsh** (ОС Windows), которые обычно вызываются до выполнения скрипта **up**.

ifconfig_pool_local_ip
Локальный виртуальный IP адрес TUN/TAP туннеля установленный с помощью опции **ifconfig-push**, если она была указана, иначе берется из пула адресов (задается опцией **ifconfig-pool**). Устанавливается только в случае использования опции **dev tun**. Устанавливается на сервере до выполнения скриптов **client-connect** и **client-disconnect**.

ifconfig_pool_netmask
Виртуальная маска сети TUN/TAP туннеля установленная с помощью опции **ifconfig-push**, если она была указана, иначе берется из пула адресов (задается опцией **ifconfig-pool**). Устанавливается только в случае использования опции **dev tap**. Устанавливается на сервере до выполнения скриптов **client-connect** и **client-disconnect**.

ifconfig_pool_remote_ip
Удаленный виртуальный IP адрес TUN/TAP туннеля установленный с помощью опции **ifconfig-push**, если она была указана, иначе берется из пула адресов (задается опцией **ifconfig-pool**). Устанавливается на сервере до выполнения скриптов **client-connect** и **client-disconnect**.

link_mtu
Максимальный размер пакета (не включая IP заголовок) данных в туннеле использующего протокол UDP. Устанавливается до выполнения скриптов **up** или **down**.

local
Параметр опции **local**. Устанавливается при старте ПК «LirVPN» и сбрасывается при поступлении сигнала **SIGHUP**.

local_port
Номер локального сетевого порта, указанного в опциях **port** или **lport**. Устанавливается при старте ПК «LirVPN» и сбрасывается при поступлении сигнала **SIGHUP**.

password
Пароль предоставляемый клиентом устанавливающим соединения. Устанавливается до выполнения скрипта **auth-user-pass-verify** только тогда, когда задан метод «**via-env**», и удаляется из окружения после завершения работы скрипта.

proto
Параметр опции **proto**. Устанавливается при старте ПК «LirVPN» и сбрасывается при поступлении сигнала **SIGHUP**.

remote_{n} Параметр опции **remote**. Устанавливается при старте ПК «LirVPN» и сбрасывается при поступлении сигнала **SIGHUP**.

remote_port_{n}
Номер удаленного порта, указанного в параметре опций **port** или **rport**. Устанавливается при старте ПК «LirVPN» и сбрасывается при поступлении сигнала **SIGHUP**.

route_net_gateway

Существующий шлюз по умолчанию в системной таблице маршрутов. Устанавливается до выполнения скрипта **up**.

route_vpn_gateway

Шлюз по умолчанию используемый опцией **route**, указанный либо в параметре опции **route-gateway**, либо во втором параметре опции **ifconfig** если присутствует опция **dev tun**. Устанавливается до выполнения скрипта **up**.

route_{parm}_{n}

Набор переменных, которые определяют каждый маршрут, предназначенный для добавления, и устанавливаются до выполнения скрипта **up**.

parm может принимать следующие значения - «**network**», «**netmask**», «**gateway**», «**metric**».

n, это номер маршрута ПК «LirVPN», начиная с 1.

Если сеть или шлюз, это DNS имена, в переменную будут записаны их IP адреса, вместо тех имен, которые были указаны в командной строке или конфигурационном файле.

script_context

Устанавливается в «**init**» или «**restart**»до выполнения **up** или **down** скриптов. Для дополнительной информации обратитесь к описанию опции **up**.

script_type

Тип скрипта. Один из **up**, **down**, **ipchange**, **route-up**, **tls-verify**, **auth-user-pass-verify**, **client-connect**, **client-disconnect**, или **learn-address**. Устанавливается до выполнения любого скрипта.

signal

Причина выхода или перезапуска. Может быть одним из **sigusr1**, **sighup**, **sigterm**, **sigint**, **inactive** (управляется опцией **inactive**), **ping-exit** (управляется опцией **ping-exit**), **ping-restart** (управляется опцией **ping-restart**), **connection-reset** (в случае обрыва TCP соединения), **error**, или **unknown** (неизвестный сигнал). Эта переменная устанавливается перед выполнением скрипта **down**.

tls_id_{n}

Набор полей сертификата от удаленного узла, где **n**, это уровень проверки. Устанавливается только для TLS соединений. Устанавливается до выполнения скрипта **tls-verify**.

tls_serial_{n}

Серийный номер сертификата удаленного узла, где **n**, это уровень проверки. Устанавливается только для TLS соединений. Устанавливается до выполнения скрипта **tls-verify**.

tun_mtu

Значение MTU для TUN/TAP устройства. Устанавливается до выполнения скриптов **up** или **down**.

trusted_ip

IP адрес клиента устанавливающего соединение или узла, который уже прошел аутентификацию Устанавливается до выполнения скриптов **ipchange**, **client-connect**, и **client-disconnect**.

trusted_port

Номер сетевого порта клиента устанавливающего соединение или узла, который уже прошел аутентификацию. Устанавливается до выполнения скриптов **ipchange**, **client-connect**, и **client-disconnect**.

untrusted_ip
IP адрес клиента устанавливающего соединение или узла, который еще не прошел аутентификацию. Иногда используется для проверки подключающегося узла с помощью nmap в скрипте **tls-verify** для проверки того, что он правильно защищен межсетевым экраном. Устанавливается до выполнения скриптов **tls-verify** и **auth-user-pass-verify**.

untrusted_port
Номер сетевого порта клиента устанавливающего соединение или узла, который еще не прошел аутентификацию. Устанавливается до выполнения скриптов **tls-verify** и **auth-user-pass-verify**.

username
Имя пользователя, предоставляемое клиентом который устанавливает соединение. Задается до выполнения скрипта **auth-user-pass-verify** только тогда, когда задан метод **«via-env»**.

6.10 Сигналы

SIGHUP
Заставляет ПК «LirVPN» закрыть все TUN/TAP устройства и сетевые соединения, перезапуститься, перечить конфигурационные файлы (если есть), и открыть заново TUN/TAP устройства и сетевые соединения.

SIGUSR1
Действует также как и **SIGHUP**, но:

- не перечитывает файлы конфигурации
- не закрывает и не открывает заново TUN/TAP соединения (при заданной опции **persist-tun**)
- не перечитывает файлы ключей и сертификатов (при заданной опции **persist-key**)
- сохраняет локальный IP адрес/порт (при заданной опции **persist-local-ip**)
- сохраняет самый последний авторизованный IP адрес/порт удаленного компьютера (при заданной опции **persist-remote-ip**)

Этот сигнал также может быть вызван изнутри ПК «LirVPN» по таймауту соединения (опция **ping-restart**).
Сигнал **SIGHUP**, в комбинации с опцией **persist-remote-ip**, может быть автоматически вызван, когда параметры сетевого интерфейса компьютера меняются, например, когда компьютер получает новый IP адрес по DHCP. Обратитесь к описанию опции **ipchange** для дополнительной информации.

SIGUSR2
Заставляет ПК «LirVPN» отобразить текущую статистику (в стандартный вывод или в системный журнал, если используется опция **daemon**).

SIGINT, SIGTERM
Заставляет ПК «LirVPN» корректно завершить свою работу.

7 Типовые сценарии применения

7.1 Маршрутизируемый туннель «точка-точка» без использования шифрования

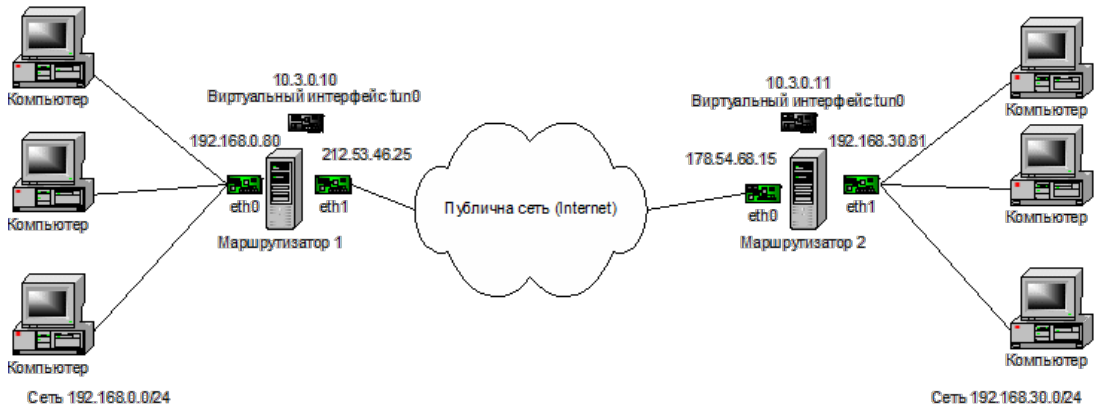

Рис. 1 – Маршрутизируемый туннель «точка-точка» без использования шифрования

Листинг № 2: Файл конфигурации Маршрутизатора 1

```
#тип виртуального устройства туннеля
dev tun

#порт, используемый для передачи данных
port 1194

#протокол, на котором нужно ждать соединений
proto udp

#в случае если по каналу ЗВКС не передается ни каких данных,
#приказывается отправлять ping каждые 10 секунд, чтобы не позволить
#соединению разорваться из-за простоя. Перезапускает LirVPN если ping
#не пришел от удаленной машины в течении 120-ти секунд
keepalive 10 120

#уровень подробности выводимых сообщений
verb 3

#назначает виртуальному tun/tap интерфейсу IP-адрес используемый в
#виртуальной сети, и указывает адрес удаленной машины в виртуальной
ifconfig 10.3.0.9 10.3.0.10

#описывает маршрут, который должны пройти пакеты, чтобы попасть в
#удаленную сеть
route 192.168.30.0 255.255.255.0 10.3.0.10

#Не использовать шифрование данных, и не проверять пакеты
```

```
28 auth none
29 cipher none
```

Листинг № 3: Файл конфигурации Маршрутизатора 2

```
1  #тип виртуального устройства туннеля
2  dev tun
3
4  #порт, на котором нужно ждать соединений
5  port 1194
6
7  #протокол, используемый для передачи данных
8  proto udp
9
10 #в случае если по каналу ЗВКС не передается ни каких данных,
11 #приказывается отправлять ping каждые 10 секунд, чтобы не позволить
12 #соединению разорваться из-за простоя. Перезапускает LirVPN если ping
13 #не пришел от удаленной машины в течении 120-ти секунд
14 keepalive 10 120
15
16 #уровень подробности выводимых сообщений
17 verb 3
18
19 #назначает виртуальному tun/tap интерфейсу IP-адрес используемый в
20 #виртуальной сети, и указывает адрес удаленной машины в виртуальной
21 #сети
22 ifconfig 10.3.0.10 10.3.0.9
23
24 #описывает маршрут, который должны пройти пакеты, чтобы попасть в
25 #удаленную сеть
26 route 192.168.0.0 255.255.255.0 10.3.0.9
27
28 #Не использовать шифрование данных, и не проверять пакеты
29 auth none
30 cipher none
31
32 #IP-адрес удаленного узла, являющегося сервером
33 remote 212.53.46.25
```

7.2 Маршрутизируемый туннель «точка-точка» с использованием симметричного шифрования

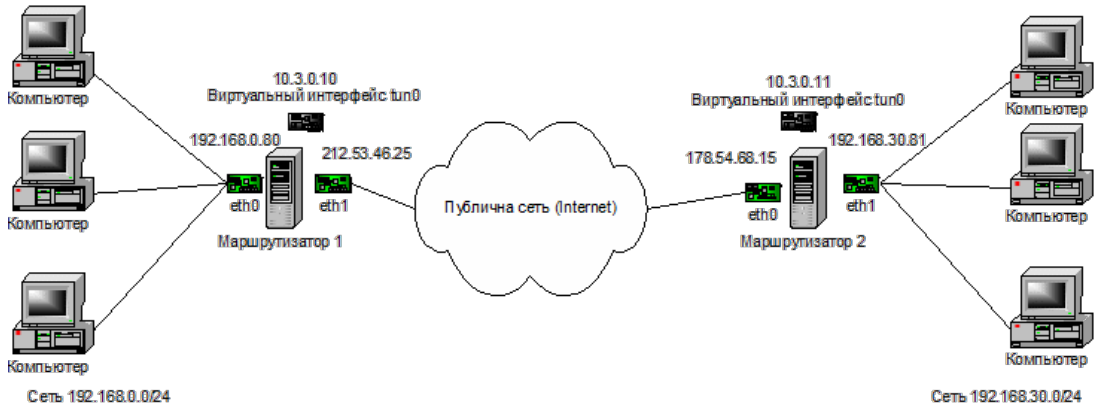

Рис. 2 – Маршрутизируемый туннель «точка-точка» с использованием симметричного шифрования

Листинг № 4: Файл конфигурации Маршрутизатора 1

```
#тип виртуального устройства туннеля
dev tun

#порт, используемый для передачи данных
port 1194

#протокол, на котором нужно ждать соединений
proto udp

#в случае если по каналу ЗВКС не передается ни каких данных,
#приказывается отправлять ping каждые 10 секунд, чтобы не позволить
#соединению разорваться из-за простоя. Перезапускает LirVPN если ping
#не пришел от удаленной машины в течении 120-ти секунд
keepalive 10 120

#уровень подробности выводимых сообщений
verb 3

#назначает виртуальному tun/tap интерфейсу IP-адрес используемый в
#виртуальной сети, и указывает адрес удаленной машины в виртуальной
ifconfig 10.3.0.9 10.3.0.10

#описывает маршрут, который должны пройти пакеты, чтобы попасть в
#удаленную сеть
route 192.168.30.0 255.255.255.0 10.3.0.10

#указывает имя файла, в котором хранится секретный ключ, используемый
#для шифрования потока
secret secret.key

#наименование алгоритма, используемого для аутентификации приходящих
#пакетов
auth GOSTHASH

#алгоритм, используемый для шифрования пакетов
cipher GOST-CBC
```

Листинг № 5: Файл конфигурации Маршрутизатора 2

```
#тип виртуального устройства туннеля
dev tun

#порт, используемый для передачи данных
port 1194

#протокол, на котором нужно ждать соединений
proto udp

#в случае если по каналу ЗВКС не передается ни каких данных,
#приказывается отправлять ping каждые 10 секунд, чтобы не позволить
#соединению разорваться из-за простоя. Перезапускает LirVPN если ping
#не пришел от удаленной машины в течении 120-ти секунд
keepalive 10 120

#уровень подробности выводимых сообщений
verb 3

#назначает виртуальному tun/tap интерфейсу IP-адрес используемый в
#виртуальной сети, и указывает адрес удаленной машины в виртуальной
ifconfig 10.3.0.10 10.3.0.9

#описывает маршрут, который должны пройти пакеты, чтобы попасть в
#удаленную сеть
route 192.168.0.0 255.255.255.0 10.3.0.9

#указывает имя файла, в котором хранится секретный ключ, используемый
#для шифрования потока
secret secret.key

#IP-адрес удаленного узла, являющегося сервером
remote 212.53.46.25

#наименование алгоритма, используемого для аутентификации приходящих
#пакетов
auth GOSTHASH

#алгоритм, используемый для шифрования пакетов
cipher GOST-CBC
```

7.3 Маршрутизируемый туннель «клиент-сервер» с использованием ассиметричного шифрования

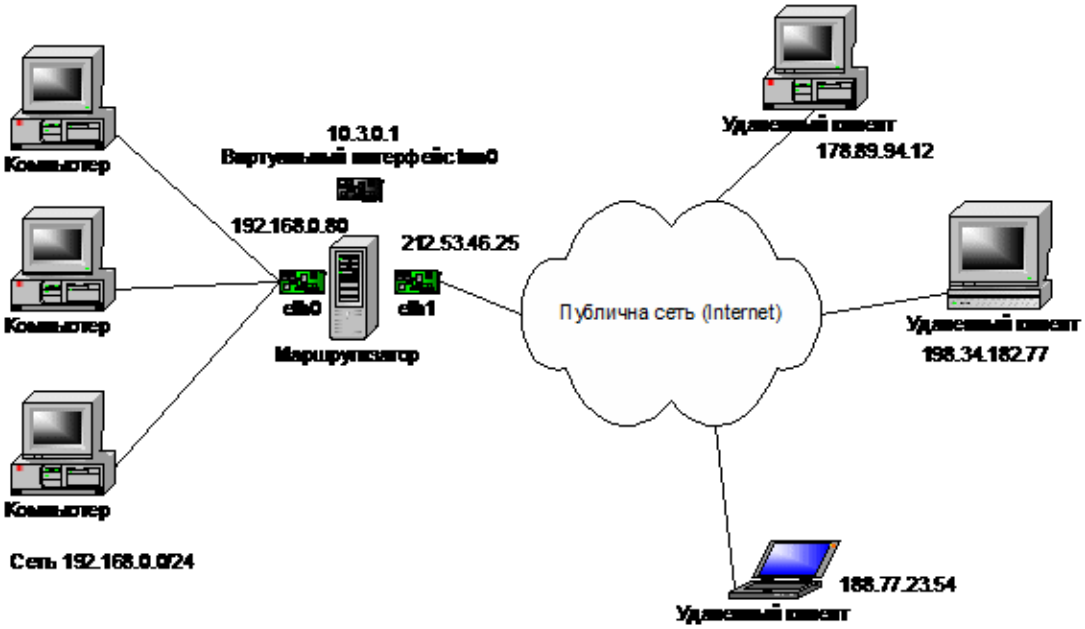

Рис. 3 – Маршрутизируемый туннель «клиент-сервер» с использованием ассиметричного шифрования

Листинг № 6: Файл конфигурации Сервера

```
#тип виртуального устройства туннеля
dev tun

#порт, используемый для передачи данных
port 1194

#протокол, на котором нужно ждать соединений
proto udp

#в случае если по каналу ЗВКС не передается ни каких данных,
#приказывается отправлять ping каждые 10 секунд, чтобы не позволить
#соединению разорваться из-за простоя. Перезапускает LirVPN если ping
#не пришел от удаленной машины в течении 120-ти секунд
keepalive 10 120

#уровень подробности выводимых сообщений
verb 3

#опция указывающая, что компьютер является сервером и задающая адресное
#пространство из которого будут назначаться адреса серверу и клиентам
server 10.3.0.0 255.255.255.0

#контейнер pkcs#12 содержащий сертификат удостоверяющего центра,
#сертификат сервера и закрытый ключ сервера
pkcs12 server.p12

#файл с параметрами Diffie-Hellman
dh dh1024.pem

#наименование алгоритма, используемого для аутентификации приходящих
#пакетов
auth GOSTHASH
```

```
33
34 #алгоритм, используемый для шифрования пакетов
35 cipher GOST-CBC
36
37 #передача маршрута на сеть 192.168.0.0/24 клиентам
38 push "route 192.168.0.0 255.255.255.0"
```

Чтобы клиенты видели друг друга, необходимо добавить в файл конфигурации Сервера опцию – **client-to-client** тем самым клиентам будет передан маршрут на подсеть 10.3.0.0/24.

Листинг № 7: Файл конфигурации Клиента

```
1  #тип виртуального устройства туннеля
2  dev tun
3
4  #порт, используемый для передачи данных
5  port 1194
6
7  #протокол, на котором нужно ждать соединений
8  proto udp
9
10 #в случае если по каналу ЗВКС не передается ни каких данных,
11 #приказывается отправлять ping каждые 10 секунд, чтобы не позволить
12 #соединению разорваться из-за простоя. Перезапускает LirVPN если ping
13 #не пришел от удаленной машины в течении 120-ти секунд
14 keepalive 10 120
15
16 #уровень подробности выводимых сообщений
17 verb 3
18
19 #наименование алгоритма, используемого для аутентификации приходящих
20 #пакетов
21 auth GOSTHASH
22
23 #алгоритм, используемый для шифрования пакетов
24 cipher GOST-CBC
25
26 #контейнер pkcs#12 содержащий сертификат удостоверяющего центра,
27 #сертификат сервера и закрытый ключ сервера
28 pkcs12 client.p12
29
30 #опция указывающая, что компьютер является клиентом
31 client
32
33 #IP-адрес удаленного узла, являющегося сервером
34 remote 212.53.46.25 1194
```

7.4 Маршрутизируемый туннель «клиент-сервер» с использованием ассиметричного шифрования, и клиентских конфигурационных файлов

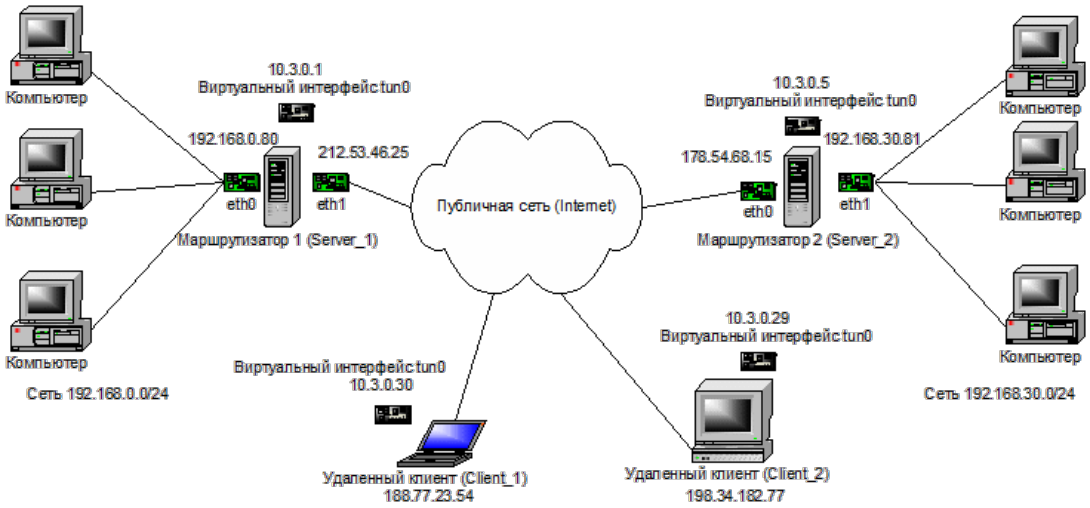

Рис. 4 – Маршрутизируемый туннель «клиент-сервер» с использованием ассиметричного шифрования, и клиентских конфигурационных файлов

Листинг № 8: Файл конфигурации Маршрутизатора 1

```
#протокол, на котором нужно ждать соединений
proto udp

#тип виртуального устройства туннеля
dev tun

#порт, используемый для передачи данных
port 1194

#уровень подробности выводимых сообщений
verb 3

#передача маршрута на сеть 192.168.0.0/24 клиентам
push "route 192.168.0.0 255.255.255.0"

#контейнер pkcs#12 содержащий сертификат удостоверяющего центра,
#сертификат сервера и закрытый ключ сервера
pkcs12 server_1.p12

#наименование алгоритма, используемого для аутентификации приходящих
#пакетов
auth GOSTHASH

#алгоритм, используемый для шифрования пакетов
cipher GOST-CBC

#опция указывающая, что компьютер является сервером и задающая адресное
#пространство из которого будут назначаться адреса серверу и клиентам
server 10.3.0.0 255.255.255.0

#файл с параметрами Diffie-Hellman
dh dh1024.pem

#Передает клиентам маршрут на подсеть 10.3.0.0/24
client-to-client
```

```
36
37 #Установка имени директории которая содержит клиентские файлы
38 #конфигурации
39 client-config-dir ccd
40
41 #Добавление сетевого машрута до подсети 192.168.30.0/24
42 #в таблицу маршрутов Маршрутизатора 1.
43 route 192.168.30.0 255.255.255.0
44
45 #в случае если по каналу ЗВКС не передается ни каких данных,
46 #приказывается отправлять ping каждые 10 секунд, чтобы не позволить
47 #соединению разорваться из-за простоя. Перезапускает LirVPN если ping
48 #не пришел от удаленной машины в течении 120-ти секунд
49 keepalive 10 120
```

Листинг № 9: Файл конфигурации Маршрутизатора 2

```
1 #тип виртуального устройства туннеля
2 dev tun
3
4 #порт, используемый для передачи данных
5 port 1194
6
7 #протокол, на котором нужно ждать соединений
8 proto udp
9
10 #уровень подробности выводимых сообщений
11 verb 3
12
13 #наименование алгоритма, используемого для аутентификации приходящих
14 #пакетов
15 auth GOSTHASH
16
17 #алгоритм, используемый для шифрования пакетов
18 cipher GOST-CBC
19
20 #контейнер pkcs#12 содержащий сертификат удостоверяющего центра,
21 #сертификат сервера и закрытый ключ сервера
22 pkcs12 server_2.p12
23
24 #опция указывающая, что компьютер является клиентом
25 client
26
27 #IP-адрес удаленного узла, являющегося сервером
28 remote 212.53.46.25 1194
```

Листинг № 10: Файл конфигурации Удаленного клиента 1

```
1 #тип виртуального устройства туннеля
2 dev tun
3
4 #порт, используемый для передачи данных
5 port 1194
6
7 #протокол, на котором нужно ждать соединений
8 proto udp
9
10 #уровень подробности выводимых сообщений
```

```
verb 3

#наименование алгоритма, используемого для аутентификации приходящих
#пакетов
auth GOSTHASH

#алгоритм, используемый для шифрования пакетов
cipher GOST-CBC

#контейнер pkcs#12 содержащий сертификат удостоверяющего центра,
#сертификат сервера и закрытый ключ сервера
pkcs12 client_1.p12

#опция указывающая, что компьютер является клиентом
client

#IP-адрес удаленного узла, являющегося сервером
remote 212.53.46.25 1194
```

Листинг № 11: Файл конфигурации Удаленного клиента 2

```
#тип виртуального устройства туннеля
dev tun

#порт, используемый для передачи данных
port 1194

#протокол, на котором нужно ждать соединений
proto udp

#уровень подробности выводимых сообщений
verb 3

#наименование алгоритма, используемого для аутентификации приходящих
#пакетов
auth GOSTHASH

#алгоритм, используемый для шифрования пакетов
cipher GOST-CBC

#контейнер pkcs#12 содержащий сертификат удостоверяющего центра,
#сертификат сервера и закрытый ключ сервера
pkcs12 client_2.p12

#опция указывающая, что компьютер является клиентом
client

#IP-адрес удаленного узла, являющегося сервером
remote 212.53.46.25 1194
```

Конфигурационные файлы клиентов должны находиться в директории **ccd**, в директории с конфигурационным файлом Маршрутизатора 1 (задается опцией **client-config-dir** в файле конфигурации Маршрутизатора 1).

Имена конфигурационных файлов клиентов должны соответствовать Common Name, указанному в их сертификатах (обратитесь к описанию опции **client-config-dir** за подробной информацией).

Листинг № 12: Конфигурационный файл клиента (Маршрутизатор 2)

```
1 #Передача определенного IP Адреса в виртуальной сети клиенту
2 ifconfig-push 10.8.0.5 10.8.0.6
3 #создание внутреннего сетевого маршрута LirVPN до сети 192.168.30.0/24
4 iroute 192.168.30.0 255.255.255.0
```

Листинг № 13: Конфигурационный файл клиента (Удаленный клиент 1)

```
1 #Передача определенного IP Адреса в виртуальной сети клиенту
2 ifconfig-push 10.8.0.17 10.8.0.18
3 #передача маршрута на сеть 192.168.30.0/24 клиентам
4 push "route 192.168.30.0 255.255.255.0"
```

Листинг № 14: Конфигурационный файл клиента (Удаленный клиент 2)

```
1 #Передача определенного IP Адреса в виртуальной сети клиенту
2 ifconfig-push 10.8.0.29 10.8.0.30
3 #передача маршрута на сеть 192.168.30.0/24 клиентам
4 push "route 192.168.30.0 255.255.255.0"
```

Лист регистрации изменений

Номера листов (страниц)					Всего листов (страниц) в докум.	№ документа	Входящий № сопроводительного докум. и дата	Подп.	Дата
Изм.	изменённых	заменённых	новых	аннулированных					