КРИПТОКОМ

УТВЕРЖДЕН
СЕИУ.00009-05 32 09 - ЛУ

СРЕДСТВО КРИПТОГРАФИЧЕСКОЙ ЗАЩИТЫ ИНФОРМАЦИИ
«МагПро КриптоПакет» 4.0

Виртуальная частная сеть «OpenVPN-ГОСТ»

Руководство по использованию

СЕИУ.00009-05 32 09
Листов 153

Литера О

2023

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

Аннотация

Настоящий документ содержит руководство по использованию виртуальной частной сети «OpenVPN-ГОСТ», которая представляет собой исполнение 7 (соответствует классу КС1) и исполнение 8 (соответствует классу КС2) СКЗИ «МагПро КриптоПакет» 4.0.

Содержание

1 Назначение 7

2 Условия работы программы 8

3 Перечень функций 9

4 Обеспечение информационной безопасности при использовании «МагПро КриптоПакет» 4.0 10

5 Установка 11
5.1 Установка в Linux . . . 11
5.2 Установка в FreeBSD и Solaris . . . 11
5.3 Установка в macOS . . . 11
5.4 Установка в Windows . . . 11
5.4.1 Установка для запуска пользователями, не имеющими административных прав . . . 11
5.4.2 Контроль над сервисом «OpenVPN-ГОСТ» из графического интерфейса «OpenVPN-ГОСТ» . . . 12
5.4.3 Windows «Run as» . . . 12
5.4.4 Создание иконки Run As в ОС Windows . . . 12
5.4.5 «Молчаливая» установка . . . 13

6 Настройка 14
6.1 Общие замечания . . . 14
6.2 Выбор типа VPN . . . 14
6.2.1 Маршрутизированная VPN и VPN типа «мост» . . . 14
6.2.2 Нумерация частных подсетей . . . 15
6.3 Инфраструктура открытых ключей . . . 16
6.3.1 Введение . . . 16
6.3.2 Генерация самоподписанного сертификата и ключа удостоверяющего центра . . . 17
6.3.3 Отзыв сертификатов . . . 17
6.4 Конфигурационные файлы . . . 18
6.4.1 Получение примеров конфигурационных файлов . . . 18
6.4.2 Редактирование серверного конфигурационного файла . . . 18
6.4.3 Редактирование клиентских конфигурационных файлов . . . 19
6.4.4 Расширение области действия VPN с включением дополнительных машин в клиентскую или серверную продсеть . . . 19
6.4.4.1 Включение нескольких машин со стороны сервера при использовании маршрутизированной VPN . . . 19
6.4.4.2 Включение нескольких машин на серверной стороне при использовании VPN типа «мост» . . . 20
6.4.4.3 Включение нескольких машин на клиентской стороне при использовании маршрутизированной VPN . . . 20
6.4.4.4 Включение нескольких машин на клиентской стороне при использовании VPN типа «мост» . . . 21

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

6.4.5 Как включить форвардинг IP-пакетов . . . 21
6.4.5.1 Windows . . . 22
6.4.5.2 Linux . . . 22
6.4.5.3 macOS . . . 22
6.4.5.4 FreeBSD . . . 22
6.4.5.5 Solaris . . . 23
6.4.6 Передача опций DHCP клиентам . . . 23
6.4.7 Назначение клиентам фиксированных IP-адресов . . . 23
6.4.8 Конфигурирование клиент-специфичных правил и политик доступа . . . 25
6.4.9 Использование альтернативных способов аутентификации . . . 26
6.4.9.1 Использование скриптовых плагинов . . . 26
6.4.9.2 Использование динамических библиотек в качестве плагинов 27
6.4.9.3 Использование аутентификации по логину и паролю как единственной формы клиентской аутентификации . . . 27
6.4.10 Добавление к конфигурации «OpenVPN-ГОСТ» двухфакторной аутентификации с использованием токенов на клиентской стороне . . . 28
6.4.10.1 О двухфакторной аутентификации . . . 28
6.4.10.2 Конфигурирование «OpenVPN-ГОСТ» для работы с токенами Рутокен . . . 28
6.4.10.3 Конфигурирование «OpenVPN-ГОСТ» для работы с токенами JaCarta и другими токенами, предоставляющими интерфейс PKCS#11 . . . 29
6.4.10.4 Конфигурирование «OpenVPN-ГОСТ» для работы с токенами Вьюга . . . 29
6.4.11 Маршрутизация всего клиентского трафика (включая веб-трафик) через VPN . . . 29
6.4.11.1 Введение . . . 29
6.4.11.2 Реализация . . . 30
6.4.11.3 Предупреждения . . . 30
6.4.12 Работа сервера «OpenVPN-ГОСТ» на динамическом IP-адресе . . . 31
6.4.13 Подключение к серверу «OpenVPN-ГОСТ» через HTTP-прокси . . . 31
6.4.14 Соединение с совместно используемым ресурсом Samba через «OpenVPN-ГОСТ» . . . 32
6.4.15 Реализация конфигурации балансировки нагрузки/восстановления после сбоя . . . 33
6.4.15.1 Клиент . . . 33
6.4.15.2 Список серверов . . . 33
6.4.15.3 Сервер . . . 33
6.4.16 Конфигурирование работы по IPv6 . . . 34
6.4.17 Конфигурирование «OpenVPN-ГОСТ» для автоматического запуска при старте системы . . . 34
6.4.17.1 Linux . . . 34
6.4.17.2 Windows . . . 34
6.4.18 Запуск VPN и тест на начальную подключаемость . . . 34
6.4.18.1 Запуск сервера . . . 34
6.4.18.2 Запуск клиента . . . 35
6.4.18.3 Поиск ошибок . . . 35

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

7 Использование **37**
7.1 Запуск на ОС семейства Windows . . . 37
7.2 Запуск на ОС Linux . . . 37
7.3 Запуск на macOS . . . 37
7.4 Запуск на ОС FreeBSD . . . 38
7.5 Запуск на ОС Solaris . . . 38
7.6 Запуск на ОС OpenWRT . . . 38
7.7 Управление запущенным процессом «OpenVPN-ГОСТ» . . . 38
7.7.1 Работа на Linux/BSD/Unix . . . 38
7.7.2 Работа в Windows в графическом интерфейсе . . . 38
7.7.3 Работа в окне командной строки Windows . . . 38
7.7.4 Работа в качестве сервиса Windows . . . 39
7.7.5 Модификация конфигурации запущенного сервера . . . 39
7.7.6 Файл статуса . . . 39
7.7.7 Использование интерфейса управления . . . 39
7.7.7.1 Команда echo . . . 40
7.7.7.2 Команда exit, quit . . . 41
7.7.7.3 Команда help . . . 41
7.7.7.4 Команда hold . . . 41
7.7.7.5 Команда kill . . . 41
7.7.7.6 Команда log . . . 42
7.7.7.7 Команда mute . . . 42
7.7.7.8 Команда net . . . 42
7.7.7.9 Команда password и username . . . 42
7.7.7.10 Команда signal . . . 43
7.7.7.11 Команда state . . . 43
7.7.7.12 Команда status . . . 44
7.7.7.13 Команда username . . . 44
7.7.7.14 Команда verb . . . 44
7.7.7.15 Команда version . . . 44
7.7.7.16 Команда auth-retry . . . 44
7.7.7.17 Формат сообщений в реальном времени . . . 45
7.7.7.18 Разбор команд . . . 45
7.7.8 Управление процессом «OpenVPN-ГОСТ» с помощью интерфейса управления . . . 45
7.8 Усиление безопасности «OpenVPN-ГОСТ» . . . 47
7.8.1 tls-auth . . . 47
7.8.2 user/group (кроме ОС Windows) . . . 48
7.8.3 chroot (кроме ОС Windows) . . . 48
7.8.4 Хранение корневого ключа (ca.key) на отдельной машине без сетевого соединения . . . 48

8 Приложение. Список опций команды openvpn-gost **49**
8.1 Общие опции . . . 49
8.2 Туннельные опции . . . 49
8.3 Серверный режим . . . 90
8.4 Клиентский режим . . . 106

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

8.5 Опции зашифрования канала данных . . . 110
8.6 Опции протокола TLS . . . 117
8.7 Информация по библиотеке SSL . . . 133
8.8 Создание случайного ключа . . . 134
8.9 Режим конфигурации сохраняемого туннеля TUN/TAP . . . 134
8.10 Опции, специфичные для Windows . . . 136
8.11 Автономные опции отладки . . . 142
8.12 Опции, относящиеся к IPv6 . . . 142
8.13 Скриптование и переменные среды . . . 143
8.13.1 Порядок выполнения скриптов . . . 143
8.13.2 Типы и преобразование строк . . . 144
8.13.3 Переменные среды . . . 145
8.14 Поддержка встроенных файлов . . . 152
8.15 Сигналы . . . 152

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

1 Назначение

«МагПро КриптоПакет» 4.0 в исполнении «OpenVPN-ГОСТ»— полноценная виртуальная частная сеть (VPN) на основе TLS, которая реализует расширение сетевой безопасности слоя OSI 2 или 3 с использованием промышленного стандарта — протокола TLS, поддерживает гибкие способы криптографической аутентификации участников друг другом и позволяет использовать пользовательские или групповые политики контроля доступа с использованием правил межсетевого экрана, применимые к виртуальному интерфейсу VPN. «OpenVPN-ГОСТ» не является сетевым прокси-сервером и не работает через веб-браузер.

«OpenVPN-ГОСТ» предлагает масштабируемый режим клиент/сервер, позволяет нескольким клиентам подключаться к одному и тому же серверному процессу «OpenVPN-ГОСТ» через один TCP-порт.

«OpenVPN-ГОСТ» — это составная часть СКЗИ «МагПро КриптоПакет» 4.0, а именно исполнение 7 (соответствует классу КС1) и исполнение 8 (соответствует классу КС2) указанного СКЗИ.

«OpenVPN-ГОСТ» является функционально законченным изделием.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

2 Условия работы программы

«МагПро КриптоПакет» 4.0 в исполнении «OpenVPN-ГОСТ» предназначен для работы в следующих операционных системах:

Windows 8.1 / 10;
Windows Server 2012 / 2016 / 2019;
Debian GNU/Linux 9(stretch) / 10(buster) / 11(bullseye);
Ubuntu 14.04, 16.04, 18.04, 20.04;
Linux Mint 19.x, 20.x, Linux Mint Debian Edition 4;
RedHat Enterprise Linux 7, 8;
CentOS 7, 8;
SUSE Linux 12, 15;
OpenSUSE 15.1, 15.2;
EMIAS OS 1.0;
Дистрибутивы Альт на базе платформ 8 и 9, включая Альт Сервер, Альт Рабочая станция, Альт Рабочая станция К, Альт Образование, Альт 8 СП, Simply Linux;
МСВСфера Сервер 7.3, МСВСфера АРМ 7.3;
Гослинукс ІС6;
РЕД ОС 7.2, 7.3;
Rosa Enterprise Desktop (RED) X4;
Rosa Enterprise Linux Server (RELS) 7.3;
Rosa Enterprise Linux Desktop (RELD) 7.3;
РОСА КОБАЛЬТ;
Astra Linux Special Edition Смоленск 1.6 aka исп.1, 1.7;
Astra Linux Special Edition Новороссийск;
Astra Linux Common Edition 2.12;
Numa Edge 1.0;
FreeBSD 12.x, 13.x;
MacOS 10.15, 11;
Sun Solaris 10, 11;
OpenWRT 19.07, 21.02.

Для хранения закрыты ключей могу использоваться

- файловая система компьютера;
- любой аппаратный ключевой носитель, предоставляющий интерфейс PKCS#11 («Рутокен ЭЦП», «JaCarta» и им подобные);
- устройство «Рутокен» с хранением ключей в файловой системе токена;
- устройсво «Вьюга».

В будущем может быть добавлена поддержка и других устройств.

Из-за ошибки в системных библиотеках возможны проблемы при работе с ключами на аппаратных токенах в операционных системах SUSE Linux, ROSA RED и Альт, а в операционной системе macOS работа с токенами возможна только при запуске программы без использования опции –daemon.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

3 Перечень функций

«МагПро КриптоПакет» 4.0 в исполнении «OpenVPN-ГОСТ» реализует виртуальную частную сеть (VPN) с шифрованием и имитозащитой данных, передаваемых по сети, в соответствии с ГОСТ Р 34.12-2015, ГОСТ Р 34.13-2015 и ГОСТ 28147-89 (только для взаимодействия с ПО, не поддерживающим ГОСТ Р 34.12-2015).

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

4 Обеспечение информационной безопасности при использовании «МагПро КриптоПакет» 4.0

Надежная криптографическая защита данных при использовании «МагПро КриптоПакет» 4.0 обеспечивается только в том случае, если эксплуатация «МагПро КриптоПакет» 4.0 осуществляется в строгом соответствии с требованиями документа «Средство криптографической защиты информации «МагПро КриптоПакет» 4.0. Правила пользования» (СЕИУ.00009–05 94).

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

5 Установка

Исполняемые файлы «OpenVPN-ГОСТ» следует установить и на клиентской, и на серверной машинах, потому что одни и те же исполняемые файлы предоставляют и клиентские, и серверные функции.

5.1 Установка в Linux

Установка «OpenVPN-ГОСТ» в ОС линукс выполняется из deb- или rpm-пакета `openvpn-gost`, специфичного для используемой вами версии Linux.

Пакет `openvpn-gost` имеет ряд зависимостей, самой важной из них является зависимость от пакета `openssl-r`, содержащего базовые компоненты СКЗИ «МагПро КриптоПакет», этот пакет также входит в комплект поставки.

Остальные пакеты, от которых зависит openvpn-gost, необходимо установить из штатного репозитория вашего дистрибутива Linux.

5.2 Установка в FreeBSD и Solaris

Установка «OpenVPN-ГОСТ» на эти операционные системы осуществляется путем разворачивания дерева установки из tar-архивов.

5.3 Установка в macOS

Установка «OpenVPN-ГОСТ» на macOS осуществляется из пакетов openvpn_gost и openssl_r с помощью системной утилиты installer.

5.4 Установка в Windows

Установка «OpenVPN-ГОСТ» в ОС Windows осуществляется путём запуска установочной программы.

«OpenVPN-ГОСТ» должен быть установлен и запущен пользователем, имеющим административные привилегии, однако после запуска даже не имеющие прав администратора пользователи смогут иметь доступ к VPN.

5.4.1 Установка для запуска пользователями, не имеющими административных прав

В установочный пакет «OpenVPN-ГОСТ» включена небольшая сервисная оболочка. Этот сервис просто запускает все конфигурационные файлы, которые находит в каталоге C:\cryptopack4\config. Если вы хотите, чтобы ваш туннель «OpenVPN-ГОСТ» работал всегда, вне зависимости от того, вошли вы в систему или нет, вы можете просто сконфигурировать сервис «OpenVPN-ГОСТ» автоматически запускаться при запуске Windows. Но может быть, удобнее запускать и выключать туннель по желанию, что можно делать, запуская и выключая сервис.

Крупный недостаток этого способа заключается в том, что нет возможности предоставить службе «OpenVPN-ГОСТ» пароль, который был применен для зашифрования вашего закрытого ключа. Это значит, что вы должны использовать незашифрованный закрытый ключ.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

Как правило, запуск и остановка сервиса требуют прав администратора, но вы можете дать обычному пользователю право контролировать индивдиуальный сервис. Это делается с помощью утилиты subinacl.exe, включенной в Windows Resource Kit.

Чтобы предоставить пользователю John право запускать и выключать сервис «OpenVPN-ГОСТ», войдите в систему с правами администратора и выполните следующую команду:

```
subinacl /SERVICE "OpenVPN-GOSTService" /GRANT=john=TO
```

5.4.2 Контроль над сервисом «OpenVPN-ГОСТ» из графического интерфейса «OpenVPN-ГОСТ»

Существует специальный режим «Только сервис», подходящий для пользователей, работающих без прав администратора. Этот режим меняет поведение действий Connect и Disconnect таким образом, что они запускают и останавливают сервис «OpenVPN-ГОСТ» вместо прямого запуска openvpn-gost.exe, как обычно. Он также прячет меню «Настройка прокси», поскольку оно не влияет на сервис. Чтобы запустить этот режим, установите следующее значение регистра в 1:

HKEY_LOCAL_MACHINE\SOFTWARE\Cryptocom\OpenVPN-GOST GUI\service_only

5.4.3 Windows «Run as»

В Windows существует возможность запустить приложение от лица другого пользователя, нежели вошедший в систему. Лучший способ использовать эту возможность в данном случае — запустив графический интерфейс «OpenVPN-ГОСТ». Поскольку графический интерфейс работает от лица администратора, то пока он работает, может быть открыто и закрыто любое количество туннелей «OpenVPN-ГОСТ».

Помните, что, используя эту возможность, вы даете пользователям возможность расширить свои права до административных. Если вы не хотите, чтобы компьютер работал под администратором, для того, чтобы защититься от вредоносного кода из сети, выполняющегося с правами администратора, то это может быть хороший способ, но если вашим пользователям ни при каких обстоятельствах нельзя запускать приложения с правами администратора, вы НЕ должны использовать этот способ работы с графическим интерфейсом «OpenVPN-ГОСТ»!

Устанавливая графический интерфейс, обязательно отключите опцию AutoStart OpenVPN GUI, поскольку вам понадобится создавать иконку для запуска вручную.

5.4.4 Создание иконки Run As в ОС Windows

- Создайте обычную иконку для openvpn-gui.exe (C:\cryptopack4\openvpn-gost-gui.exe) на рабочем столе.
- Щелкните правой клавишей мыши по иконке и выберите Свойства.
- Щелкните на Advanced...
- Поставьте галочку Run with different credentials.

Когда вы щелкаете двойным щелчком мыши по этой иконке, появится окно, в котором нужно будет ввести логин и пароль для пользователя, от лица которого вы хотите запустить графический интерфейс. Если вы хотите, чтобы он запускался автоматически, когда вы входите в систему, перенесите эту иконку в каталог Startup в меню Start-Programs. Тогда у вас будут запрашивать логин и пароль непосредственно каждый раз при входе в систему.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

5.4.5 «Молчаливая» установка

Установщик «OpenVPN-ГОСТ» может работать в «молчаливом» режиме без взамодействия с пользователем. Для включения «молчаливого» режима при запуске установщика необходимо указать в командной строке ключ /S.

При использовании «молчаливого» режима параметрами установки можно управлять с помощью ключей командной строки. Инсталлятор базовых компоненов поддерживает следующие ключи

`/KEY=ключ` — лицензионный ключ (обязателен);
`/RNG=тип` — тип ДСЧ (по умолчанию PROGRAM);
`/PHOST=хост` — хост прокси (требуется только при работе через прокси);
`/PPORT=порт` — порт прокси (требуется только при работе через прокси);
`/PUSER=польз` — пользователь прокси (требуется только при работе через прокси с авторизацией);
`/PPASS=пароль` — пароль пользователя прокси (требуется только при работе через прокси с авторизацией);
`/SELECT_RUTOKEN=1` — предписывает устанавливать модули работы с Рутокенами;
`/D=путь` — папка, в которую будет выполнена установка (по умолчанию `C:\cryptopack4`). Этот ключ должен быть последним, так как все, что написано после него, включая пробелы, трактуется как имя папки.

Инсталлятор «OpenVPN-ГОСТ» поддерживает следующие ключи:
`/SELECT_OPENVPNGUI=1` — установить GUI;
`/SELECT_TAP=1` — установить TAP-адаптер;
`/SELECT_ASSOCIATIONS=1` — ассоциировать файлы .ovpn;
`/SELECT_PATH=1` — добавить путь в PATH;
`/SELECT_SHORTCUTS=1` — создать ярлык в меню "Пуск";
`/D=путь` — папка, в которую будет выполнена установка (значение по умолчанию `C:\cryptopack4`). Этот ключ должен быть последним, так как все, что написано после него, включая пробелы, трактуется как имя папки.

По умолчанию все эти действия включены, отключить их можно, выставив соответствующему параметру значение 0.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

6 Настройка

6.1 Общие замечания

Для настройки «OpenVPN-ГОСТ» необходимо:

- Определиться с типом VPN — выбрать маршрутизированную VPN или VPN типа «мост» (см. раздел 6.2).
- Создать или воспользоваться инфраструктурой открытых ключей для создания сертификата закрытого ключа сервера и каждого клиента, корневого сертификата удостоверяющего центра и ключа, который используется для подписи каждого из серверных и клиентских сертификатов (см. раздел 6.3.1).
- Создать конфигурационные файлы для сервера и клиентов (см. раздел 6.4).

6.2 Выбор типа VPN

6.2.1 Маршрутизированная VPN И VPN типа «мост»

Маршрутизирование и связь типа «мост» (бриджинг) - два способа связи систем через VPN.

Когда клиент подключается к удаленной сети через VPN типа «мост», ему присваивается IP-адрес, являющийся частью удаленной физической подсети Ethernet, и после этого он может взаимодействовать с другими машинами удаленной сети, как будто они соединены локально. Настройки VPN типа «мост» требуют специального инструмента, зависящего от операционной системы, чтобы связать адаптер подсети Ethernet с виртуальным устройством стиля TAP. Например, для Linux такой инструмент - brctl. На ОС семейства Windows выберите адаптер TAP-Win32 и адаптер сети Ethernet в Control Panel->Network connections, затем щелкните правой клавишей мыши и выберите Bridge Connections.

Когда клиент подключается через маршрутизированную VPN, он использует собственную отдельную подсеть, и маршруты устанавливаются и на клиентской машине и на удаленном гейте, чтобы пакеты данных беспрепятственно проходили через VPN. Здесь клиент - необязательно одна машина, это может быть подсеть из нескольких машин.

Маршрутизация и связь типа «мост» с функциональной точки зрения очень похоже, с важным различием в том, что маршрутизированная VPN не пропускает широковещательные рассылки IP-пакетов, а VPN типа «мост» пропускает.

Эти разновидности VPN связаны с использованием различных адаптеров. Устройство TAP - виртуальный адаптер сети Ethernet, устройство TUN - виртуальная IP-связь точка-в-точку.

Когда вы используете VPN типа «мост», вы должны всегда использовать -dev tap на обоих концах соединения. Если вы используете маршрутизированную VPN, вы можете использовать и -dev tap, и -dev tun, но вы должны использовать одно и то же на обоих концах соединения. Мы рекомендуем использовать -dev tun, так как некоторые возможности «OpenVPN-ГОСТ», включая использование конфигурации шлюза филиала, невозможно использовать с tap-адаптером.

Следует иметь ввиду, что tap-адарперы не поддерживаются на macOS начиная с версии 10.5 (catalina).

Преимущества VPN типа «мост»:

- Широковещательные рассылки IP-пакетов проходят через VPN - это позволяет программному обеспечению, зависящему от рассылок локальной сети, например, Windows NetBIOS, обмениваться файлами и просматривать сетевое окружение.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

– Не нужно конфигурировать маршрутизационных утверждений.
– Решение, относительно легкое для конфигурирования.

Недостатки VPN типа «мост»:

– Менее эффективна, чем маршрутизирование, и плохо масштабируется.

Преимущества маршрутизированной VPN:

– Эффективность и масштабируемость
– Позволяет лучшую настройку MTU для эффективности.

Недостатки маршрутизированной VPN:

– Клиенты должны использовать сервер WINS (например, Samba), чтобы позволить работать просмотру сети через VPN.
– Необходимо настроить маршруты, связывающие каждую подсеть.
– Программное обеспечение, зависящее от броадкастов, не будет «видеть» машины на другой стороне VPN.

В общем, маршрутизирование, вероятно, лучший выбор для большинства пользователей, потому что оно более эффективно и его легче настроить (как и саму конфигурацию «OpenVPN-ГОСТ»). Маршрутизирование также предоставляет большую возможность избирательно контролировать права доступа на клиентски-специфичной основе.

Рекомендуется использовать маршрутизацию, если только вам не нужны специфические возможности, требующие использования «моста», такие как:

1. Вы запускаете приложения над VPN, которые полагаются на сетевую ретрансляцию (такие как сетевые игры)
2. Вы бы хотели позволить просматривать через VPN совместно используемые ресурсы, не устанавливая серверы Samba или WINS.

6.2.2 Нумерация частных подсетей

Установка VPN часто требует связывания вместе частных подсетей из разных локаций.

IANA зарезервировала следующие три блока пространства IP-адресов для частных сетей (кодифицировано в RFC 1918):

```
10.0.0.0      10.255.255.255   (префикс 10/8)

172.16.0.0    172.31.255.255   (префикс 172.16/12)

192.168.0.0   192.168.255.255 (префикс 192.168/16)
```

В то время как адреса из этих блоков следует в норме использовать в конфигурациях VPN, важно выбирать адреса, которые минимизируют возможность конфликтов IP-адресов или подсетей. Типы конфликтов, которых следует избегать:

– Конфликты разных мест в VPN, использующих одинаковую нумерацию подсетей LAN;
– Соединения удаленного доступа из мест, которые используют частные подсети, конфликтущие с вашими подсетями VPN.

Например, предположим, что вы используете популярную подсеть 192.168.0.0/24 как вашу частную подсеть LAN. Теперь вы пытаетесь соединиться с VPN из интернет-кафе, использующего ту же подсеть для своего WiFi LAN. У вас будет проблема маршрутизации, потому

что ваша машина не поймет, относится ли 192.168.0.1 к локальному WiFi-гейту или к тому же адресу на VPN.

В качестве другого примера, предположим, что вы хотите соединить вместе несколько точек с помощью VPN, но каждая точка использует 192.168.0.0/24 в качестве своей подсети LAN. Это не будет работать, если не добавить дополнительный слой трансляции NAT, потому что VPN не поймет, как маршрутизировать пакеты между несколькими точками, если эти точки не используют подсеть, которая уникально идентифицирует их.

Лучшее решение — избегать пользоваться 10.0.0.0/24 или 192.168.0.0/24 в качестве сетевых адресов LAN. Вместо этого используйте что-нибудь, что с более низкой вероятностью будет использовано в WiFi-кафе, аэропорту или отеле, откуда вы, возможно, захотите установить удаленнй доступ. Лучшие кандидаты — подсети в середине большого блока 10.0.0.0/8 (например, 10.66.77.0/24).

Чтобы избежать кросс-локационных конфликтов номеров IP, всегда используйте уникальную нумерацию для ваших подсетей LAN.

6.3 Инфраструктура открытых ключей

6.3.1 Введение

Инфраструктура открытых ключей (PKI) состоит из:

- Отдельного сертификата (также известного как открытый ключ) и закрытого ключа для сервера и каждого клиента, и - Корневого сертификата удостоверяющего центра и ключа, который используется для подписи каждого из серверных и клиентских сертификатов.

Для аутентификации необходимо, чтобы клиент мог аутентифицировать сервер по его сертификату. Дополнительно можно сделать так, чтобы сервер мог аутентифицировать клиентов по их сертификатам. «OpenVPN-ГОСТ» поддерживает такую двунаправленную аутентификацию, что означает, что клиент должен аутентифицировать серверный сертификат, и сервер должен аутентифицировать клиентский сертификат, прежде чем устанавливается взаимное доверие.

И сервер, и клиент будут аутентифицировать друг друга, сначала проверив, что представленный кандидат был подписан на сертификате удостоверяющего центра, а затем тестируя информацию в заголовке уже аутентифицированного сертификата, такую, как common name сертификата или его тип (клиентский или серверный).

Эта модель безопасности имеет ряд желательных признаков с точки зрения VPN:

- Серверу нужен только его собственный сертификат/ключ — ему не нужно знать индивидуальные сертификаты всех клиентов, которые, возможно, установят с ним соединение.
- Сервер будет принимать только клиентов, чьи сертификаты были подписаны на сертификате удостоверяющего центра (который будет генерирован ниже). И поскольку сервер может выполнять эту проверку подписи без необходимости доступа к самому закрытому ключу УЦ, ключ УЦ (самый чувствительный ключ во всей PKI) может находиться на совершенно другой машине, даже на машине без сетевого соединения.
- Если закрытый ключ скомпрометирован, он может быть дезактивирован путем добавления его сертификата в CRL (список отзыва сертификатов). CRL позволяет скомпрометированным сертификатам быть отвергнутыми избирательно, не требуя перестраивать всю PKI.
- Сервер может усилить клиентски-специфичные права доступа, основанные на внутренних полях сертификата, таких как Common Name.

Обратите внимание, что часы сервера и клиента должны быть более-менее синхронизированы, иначе сертификаты могут работать некорректно.

6.3.2 Генерация самоподписанного сертификата и ключа удостоверяющего центра

Для генерации комплекта ключей и сертификатов, необходимых для работы «OpenVPN-ГОСТ», рекомендуем использовать набор утилит easy-gost, поставляемый в составе дистрибутива.

6.3.3 Отзыв сертификатов

Отзыв сертификата означает объявление недействительным ранее подписанного сертификата, так что его больше нельзя использовать для целей аутентификации.

Типичные причины желания отозвать сертификат включают:

- Закрытый ключ, связанный с сертификатом, скомпрометирован или украден.
- Пользователь зашифрованного закрытого ключа забывает пароль к ключу.
- Вы хотите прекратить доступ пользователя к VPN.

Сначала вам нужно создать список отзыва. Хорошая практика — создать пустой список отзыва, и пусть ваши клиенты его проверяют. Таким образом, когда вам действительно придется отзывать сертификат, у вас не будет проблем с тем, чтобы заставить клиентов это заметить.

Чтобы создать список отзыва для вашего УЦ, вам прежде всего нужно создать файл index.txt. Это сначала будет пустой файл (созданный командой touch). Однако, когда вы начнете отзывать сертификаты, в него будет добавляться информация. Файл читабелен для людей и неподписан, поэтому нам нужно, чтобы OpenSSL сделала для него подписанную PEM-форму. Таким образом, когда вы получите свой пустой файл указателя, вы можете сделать из него список отзыва с помощью:

```
openssl ca -gencrl -keyfile ca.key -cert ca.crt -out crl.pem

где ca.key --- закрытый ключ CA
ca.crt --- сертификат CA
crl.pem --- требуемый файл, содержащий списки отзыва
```

Эта команда создаст для вас список отзыва, действительный в течение умолчательного промежутка времени (1 месяц). В том случае, если список отзыва используется только на серверах, которые вы контролируете, и где вы уверены, что обновите список отзыва при следующем отзыве, вы, возможно, захотите увеличить время жизни списка отзыва. Иначе через несколько месяцев ваши серверы будут жаловаться, что список отзыва устарел.

Чтобы увеличить продолжительность времени, в течение которого список отзыва действителен, добавьте опцию crldays xxx к вышеприведенной команде генерации списка отзыва (где xxx — количество дней, в течение которых список отзыва действителен)

Примечание. Если срок действия вашего списка отзыва истекает, прежде чем вы отзовете сертификат, просто создайте новый, как описано выше. Список отзыва — просто подписанная копия внутреннего списка отозванных сертификатов, имеющая срок действия и оформленная в стандартном формате. Вы можете создавать новые списки отзыва, когда захотите.

Теперь, когда начальный список отзыва готов, мы отзовем сертификат:

```
openssl ca -revoke bad.crt -keyfile ca.key -cert ca.crt

где ca.key и ca.crt --- то же самое, что и в предыдущей команде, а
```

`bad.crt --- отзываемый сертификат.`

Это автоматически обновит ваш файл index.txt, добавив новые детали об отозванном сертификате. Теперь вам нужно создать новый файл списка отзыва той же командой, которой мы пользовались выше, чтобы создать пустой файл. Когда новый список отзыва создан, его необходимо опубликовать!

Если вы хотите поиграть с продолжительностью действия списка отзыва и прочими подобными вещами, вам необходимо прочитать раздел «Опции списка отзыва» в руководстве по OpenSSL CA. Если вы хотите манипулировать со списком отзыва, просматривать его и т.д., прочитайте руководство по утилите CRL.

6.4 Конфигурационные файлы

6.4.1 Получение примеров конфигурационных файлов

Лучше всего использовать готовые примеры конфигурационных файлов «OpenVPN-ГОСТ» в качестве основы для вашей собственной конфигурации. В ОС Windows эти файлы можно найти в C:\cryptopack4\sample-config, в остальных ОС — в /etc/openvpn-gost.

Обратите внимание что в Linux, BSD или Unix-подобных операционных системах примеры конфигурационных файлов называются server.conf.sample и client.conf.sample. В Windows они называются gost-server.ovpn и gost-client.ovpn.

6.4.2 Редактирование серверного конфигурационного файла

Пример серверного конфигурационного файла — идеальная стартовая точка для серверной конфигурации «OpenVPN-ГОСТ». Он создаст VPN с использованием виртуального сетевого интерфейса, будет слушать клиентские соединения на tcp-порту 1194 (номер порта, зарезервированный для VPN), и будет распределять виртуальные адреса для подключающихся клиентов из подсети 10.9.1.0/24.

Прежде чем использовать пример конфигурационного файла, вам прежде всего следует отредактировать параметры ca, cert и key так, чтобы они указывали на файлы, которые вы сгенерировали раньше.

Теперь конфигурацию сервера можно использовать, но вы можете захотеть ее еще больше модифицировать.

1. Если вы используете VPN типа «мост», вы должны использовать server-bridge вместо server и dev tap.
2. Если вы хотите использовать другой набор виртуальных IP-адресов, нежели 10.9.1.0/24, вам следует модифицировать директиву server. Помните, что этот набор виртуальных адресов должен быть частным набором, который не используется в вашей сети.
3. Добавьте директиву client-to-client, если вы хотите, чтобы подключающиеся клиенты могли видеть друг друга через VPN. По умолчанию клиенты могут видеть только сервер. В случае использования TUN-интерфейса необходимо также включить форвардинг IP (см. 6.4.5).
4. Если вы используете Windows, необходимо закомментировать директивы user nobody и group nobody.

Если вы хотите запустить несколько экземпляров «OpenVPN-ГОСТ» на одной и той же машине, каждый со своим конфигурационным файлом, это возможно если вы:

1. используете отдельный номер порта для каждого экземпляра.

2. Если вы используете Windows, каждая конфигурация «OpenVPN-ГОСТ» должна использовать свой собственный адаптер TAP-Win32. Вы можете добавить дополнительные адаптеры в Start Menu -> All Programs -> «OpenVPN-ГОСТ» -> Add a new TAP-Win32 virtual ethernet adapter.
3. Если вы запускаете несколько экземпляров «OpenVPN-ГОСТ» из одного и того же каталога, обязательно отредактируйте директивы, которые создают выходные файлы, чтобы различные экземпляры не переписывали выходные файлы друг друга. Эти директивы включают log, log-append и status.

6.4.3 Редактирование клиентских конфигурационных файлов

Пример клиентской конфигурации (client.conf.sample на Linux/BSD/MacOS или gost-client.ovpn на Windows) отражает умолчательные директивы, установленные в примере серверного конфигурационного файла.

1. Как и с серверным конфигурационным файлом, сначала отредактируйте параметры ca, cert и key так, чтобы они указывали на файлы, которые вы создали раньше. Обратите внимание, что каждый клиент должен иметь свою собственную пару cert/key. Только файл ca универсален для сервера «OpenVPN-ГОСТ» и всех клиентов.
2. Далее, отредактируйте директиву remote так, чтобы она указывала на hostname/IP-адрес и номер порта сервера «OpenVPN-ГОСТ» (если ваш сервер «OpenVPN-ГОСТ» будет работать на машине с одним сетевым адаптером за брандмауэром/NAT-гейтом, используйте публичный IP-адрес гейта и номер порта, который вы сконфигурировали для гейта для форвардинга на сервер «OpenVPN-ГОСТ»).
3. Наконец, убедитесь, что клиентский конфигурационный файл соответствует директивам в серверном конфигурационном файле. Самое главное — проверить, что директивы dev (tun или tap) и proto (tcp или udp) совпадают. Также убедитесь, что comp-lzo и fragment, если используются, присутствуют и в серверном, и в клиентском конфигурационном файле.

6.4.4 Расширение области действия VPN с включением дополнительных машин в клиентскую или серверную продсеть

6.4.4.1 Включение нескольких машин со стороны сервера при использовании маршрутизированной VPN

Поскольку VPN работает в режиме «точка-точка» между клиентом и сервером, может быть желательно расширить область действия VPN так, чтобы клиенты могли связаться с несколькими машинами в сети сервера, а не только с самой машиной сервера.

Для целей этого примера предположим, что серверная LAN использует подсеть 10.66.0.0/24, а множество IP-адресов VPN использует 10.9.1.0/24, как показано в директиве server в конфигурационном файле сервера «OpenVPN-ГОСТ».

Сначала вы должны *объявить* подсеть 10.66.0.0/24 клиентам VPN как доступную через VPN. Это легко делается с помощь следующей директивы серверного конфигурационного файла:

```
push "route 10.66.0.0 255.255.255.0"
```

Затем вы должны установить маршрут на гейте серверной LAN, чтобы маршрутизировать подсеть клиента VPN (10.9.1.0/24) на сервер «OpenVPN-ГОСТ» (это необходимо только в том случае, если сервер «OpenVPN-ГОСТ» и гейт LAN — различные машины.)

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

Убедитесь, что вы включили форвардинг IP на серверной машине «OpenVPN-ГОСТ» (см. 6.4.5).

6.4.4.2 Включение нескольких машин на серверной стороне при использовании VPN типа «мост»

Одно из преимуществ использования VPN типа «мост» через Ethernet заключается в том, что это вы получаете без дополнительного конфигурирования.

6.4.4.3 Включение нескольких машин на клиентской стороне при использовании маршрутизированной VPN

В типичном сценарии удаленного доступа клиентская машина подключается к VPN как одиночная машина. Но предположим, что клиентская машина — гейт локальной сети (например, домашнего офиса), и вы бы хотели, чтобы все машины клиентской сети могли подключаться к VPN.

В первую очередь необходимо отметить, что сделать это можно только при работе через TUN-интерфейс. Если Вы используете -dev tap, необходимо заменить его на -dev tun как на стороне клиента, так и на стороне сервера.

Для этого примера предположим, что клиентская локальная сеть использует подсеть 192.168.4.0/24, а клиент VPN использует сертификат с Common Name client2. Наша цель — так настроить VPN, чтобы любая машина клиентской сети могла общаться с любой машиной серверной сети через VPN.

Перед настройкой необходимо обеспечить несколько основных требований:

- Клиентская локальная подсеть (в нашем примере 192.168.4.0/24) не должна экспортироваться в VPN сервером или любыми другими клиентскими локациями, которые используют ту же подсеть. Каждая подсеть, объединенная VPN через маршрутизацию, должна быть уникальной.
- Клиент должен иметь уникальное поле Common Name в своем сертификате (в нашем примере client2), и флаг duplicate-cn не должен использоваться в серверном конфигурационном файле «OpenVPN-ГОСТ».

Прежде всего убедитесь, что на клиентской машине включен форвардинг IP (см. 6.4.5).

Далее мы произведем необходимые конфигурационные изменения на серверной стороне. Если серверный конфигурационный файл не указывает на каталог клиентской конфигурации, добавьте указание сейчас:

```
client-config-dir ccd
```

В вышеуказанной директиве ccd должен быть именем подкаталога, который необходимо заранее создать в каталоге с конфигурационным файлом «OpenVPN-ГОСТ». В Unix это обычно /etc/openvpn-gost, а в Windows C:\Cryptopack4\config. Когда новый клиент связывается с сервером «OpenVPN-ГОСТ», демон просканирует этот каталог в поисках файла, который соответствует полю common name соединяющегося клиента. Если соответствующий файл найден, он будет прочитан и обработан для применения дополнительных директив конфигурационного файла к поименованному клиенту.

Следующий шаг — создать файл под названием client2 в каталоге ccd. Этот файл должен содержать строку:

```
iroute 192.168.4.0 255.255.255.0
```

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

Это скажет серверу «OpenVPN-ГОСТ», что подсеть 192.168.4.0/24 следует маршрутизировать на client2.

Далее, добавьте следующую строку к главному серверному конфигурационному файлу (а не к файлу ccd/client2):

```
route 192.168.4.0 255.255.255.0
```

Вы можете спросить, почему используются и route, и iroute? Причина в том, что route контролирует маршрутизацию из ядра на сервер «OpenVPN-ГОСТ» (через интерфейс TUN), а iroute контролирует маршрутизацию от сервера «OpenVPN-ГОСТ» к удаленным клиентам. Оба необходимы.

Далее, спросите себя, хотите ли вы допустить сетевую передачу информации между подсетью client2 (192.168.4.0/24) и другими клиентами сервера «OpenVPN-ГОСТ». Если да, добавьте к серверному конфигурационному файлу следующее:

```
client-to-client
push "route 192.168.4.0 255.255.255.0"
```

Это заставит сервер «OpenVPN-ГОСТ» *объявить* подсеть client2 остальным подключающимся клиентам.

Последний шаг, который часто забывают — добавить маршрут к гейту серверной локальной сети, который направляет 192.168.4.0/24 к серверу «OpenVPN-ГОСТ» (вам это не понадобится, если сервер «OpenVPN-ГОСТ» - гейт для сервера локальной сети). Предположим, что вы пропустили этот шаг и попытались передать сигнал ping машине (не самому серверу «OpenVPN-ГОСТ») в серверной сети из 192.168.4.8. Внешний сигнал, вероятно, достигнет машины, но она не будет знать, как маршрутизировать ответный сигнал, потому что не будет знать, как достичь 192.168.4.0/24. Основное правило пользования состоит в том, что когда вы маршрутизируете целые локальные сети через VPN (где сервер VPN — не та же машина, что гейт локальной сети), убедитесь, что гейт локальной сети маршрутизирует все подсети VPN на серверную машину VPN.

Подобным же образом, если клиентская машина, на которой работает «OpenVPN-ГОСТ», не является гейтом клиентской локальной сети, то гейт клиентской локальной сети должен иметь маршрут, который направляет все подсети, которые можно видеть через VPN, на клиенскую машину «OpenVPN-ГОСТ».

6.4.4.4 Включение нескольких машин на клиентской стороне при использовании VPN типа «мост»

Это требует более сложной настройки (может быть, не сложнее на практике, но сложнее для объяснения в деталях):

- Необходимо соединить клиентский интерфейс TAP с сетевым адаптером, соединенным с локальной сетью на клиенте, в режиме «мост».
- Необходимо вручную установить IP/маску сети на интерфейсе TAP на клиенте.
- Необходимо сконфигурировать машины на клиентской стороне так, чтобы они использовали IP/маску сети, которая находится внутри подсети, соединенной в режиме «мост», возможно, запрашивая сервер DHCP на стороне сервера «OpenVPN-ГОСТ» в VPN.

6.4.5 Как включить форвардинг IP-пакетов

Практически во всех операционных системах форвардинг IP-пакетов по умолчанию выключен. Если требуется, чтобы «OpenVPN-ГОСТ» мог работать шлюзом между VPN-сетью

и локальной сетью, форвардинг необходимо включить. В разных операционных системах и даже в разных версиях одной ОС это может делаться по-разному, в данном разделе содержатся общие рекомендации, которые будут работать в большинстве случаев, но если же они не сработают, обратитесь к руководству по настройке вашей операционной системы.

6.4.5.1 Windows

В ОС Windows для того, чтобы включить форвардинг IP, необходимо запустить редактор реестра, найти путь `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters` и выставить параметру IPEnableRouter значение 1 (тип данных REG_DWORD).

6.4.5.2 Linux

В ОС Linux для того, чтобы включить форвардинг IP, необходимо отредактировать файл /etc/sysctl.conf. В ряде дистрибутивов (дебиан, убунту) этот файл ставится «из коробки» и в нём есть необходимые строки, только закомментированные, соответственно, их нужно найти и раскомментировать. Для IPv4 это может быть строка

```
net.ipv4.ip_forward=1
```

или строка

```
net.ipv4.conf.all.forwarding=1
```

Для IPv6 строка

```
net.ipv6.conf.all.forwarding=1
```

Если же у вас нет файла /etc/sysctl.conf, его нужно создать и запишисать в него строки

```
net.ipv4.conf.all.forwarding=1
net.ipv6.conf.all.forwarding=1
```

После того, как файл /etc/sysctl.conf отредактирован, необходимо выполнить команду

```
sudo sysctl -p /etc/sysctl.conf
```

6.4.5.3 macOS

В macOS для того, чтобы включить форвардинг IP, необходимо создать файл /etc/sysctl.conf и добавить в него строки

```
net.inet.ip.forwarding=1
net.inet6.ip6.forwarding=1
```

После этого необходимо выполнить команды

```
sudo sysctl -w net.inet.ip.forwarding=1
sudo sysctl -w net.inet6.ip6.forwarding=1
```

6.4.5.4 FreeBSD

В ОС FreeBSD для того, чтобы включить форвардинг IP, необходимо в файл /etc/rc.conf добавить строки

```
gateway_enable="YES"
ipv6_gateway_enable="YES"
```

После этого необходимо выполнить команды

```
sudo sysctl -w net.inet.ip.forwarding=1
sudo sysctl -w net.inet6.ip6.forwarding=1
```

6.4.5.5 Solaris

В ОС Solaris для того, чтобы включить форвардинг IP, необходимо в консоли администратора выполнить команды

```
routeadm -e ipv4-forwarding
routeadm -e ipv6-forwarding
```

Изменения вступят в силу после перезагрузки. Для того, чтобы они вступили в силу немедленно, необходимо дополнительно выполнить команду

```
pfexec routeadm -u
```

6.4.6 Передача опций DHCP клиентам

Сервер «OpenVPN-ГОСТ» может передавать некоторые опции DHCP, такие как адреса серверов DNS и WINS, клиентам. Клиенты Windows могут принимать переданные опции DHCP сами по себе, а клиенты дргуих операционных систем могут их принимать, используя скрипт up на клиентской стороне, который анализирует список переменных среды `foreign_option_n`.

Например, предположим, что вы бы хотели, чтобы соединяющиеся клиенты использовали внутренний DNS-сервер на 10.66.0.4 или 10.66.0.5 и сервер WINS на 10.66.0.8. Добавьте в конфигурацию сервера «OpenVPN-ГОСТ»:

```
push "dhcp-option DNS 10.66.0.4"
push "dhcp-option DNS 10.66.0.5"
push "dhcp-option WINS 10.66.0.8"
```

Чтобы протестировать эту возможность на Windows, запустите следующую команду из окна командной строки после того, как машина подключилась к серверу «OpenVPN-ГОСТ»:

```
ipconfig /all
```

Запись для адаптера TAP-Win32 должна показать опции DHCP, переданные сервером.

6.4.7 Назначение клиентам фиксированных IP-адресов

Обычно при подключении к серверу клиент получает первый свободный виртуальный IP-адрес из пула адресов, заданного командой server. Когда клиент отключается, адрес освобождается и может быть выдан другому клиенту. Когда клиент подключается заново, он также может получит другой адрес, даже если его «старый» адрес свободен.

Однако иногда (см., например, 6.4.8) бывает полезно, чтобы некоторый клиент всегда получал один и тот же виртуальный IP-адрес. В данном разделе описано, как это сделать.

Хотя общие приниципы конфигурирования сервера для выдачи клиентам фиксированных IP-адресов совпадают, детали различаются для разных топологий. Здесь описана конфигурация для топологии «subnet», которая является рекомендуемой. При работе через TUN-интерфейс для выбора этой топологии необходимо в конфигурационный файл сервера добавить строки

```
topology subnet
push "topology subnet"
```

При работе через TAP-интерфейс эта топология является единственно возможной, поэтому указывать её явным образом не требуется.

Все необходимые настройки делаются на стороне сервера.

В первую очередь следует отметить, что диапазон адресов виртуальной сети необходимо разделить на две части: пул адресов, которые сервер будет назначать клиентам автоматически, и статические адреса, закреплённые за выделенными клиентами. Если этого не сделать, то может возникнуть ситуация, когда адрес, закреплённый за конкретным клиентом, будет автоматически выдан какому-то другому клиенту.

Предположим, что мы хотим назначить виртуальной сети диапазон адресов 10.9.1.0/24, в нём выделить пул адресов с 10.9.1.10 до 10.9.1.199 для автоматического назначение адресов, а фиксированные адреса выдавать за пределами этого пула.

Важно иметь ввиду, что директиву server в этом случае использовать нельзя, так как она помещает в пул весь диапазон адресов. Вместо неё следует вписать в конфигурационный файл следующую последовательность директив:

```
mode server
tls-server
ifconfig 10.9.1.1 255.255.255.0
ifconfig-pool 10.9.1.10 10.9.1.199 255.255.255.0
push "route-gateway 10.9.1.1"
```

Кроме того, если у вас в директиве proto установлено значение tcp, следует изменить эту директиву на

```
proto            tcp-server
```

Далее следует задать каталог клиентской конфигурации

```
client-config-dir ccd
```

В вышеуказанной директиве ccd должен быть именем каталога, который необходимо заранее создать. Когда новый клиент связывается с сервером «OpenVPN-ГОСТ», демон просканирует этот каталог в поисках файла, который соответствует полю Common Name соединяющегося клиента. Если соответствующий файл найден, он будет прочитан и обработан для применения дополнительных директив конфигурационного файла к поименованному клиенту.

Следует пояснить, что единственное, что сервер «OpenVPN-ГОСТ» знает про подключающегося к нему клиента – это предъявленный клиентом сертификат, поэтому постоянный IP-адрес «привязывается» именно к сертификату, а точнее, к полю Common Name сертификата. По этой причине механизм назначения постоянных IP-адресов не будет работать при наличии директивы duplicate-cn, убедитесь, что в конфигурационном файле сервера эта директива отсутствует.

Теперь можно назначать клиентам постоянные IP-адреса. Для каждого клиента, которому требуется назначить постоянный виртуальный IP-адрес, нужно в каталоге ccd создать файл с именем, совпадающим с Common Name из сертификата клиента. Этот файл должен содержать директиву ifconfig-push, предписывающую, какой именно IP-адрес следует назначить клиенту с таким Common Name. Например, файл, содержащий строку вида

```
ifconfig-push 10.9.1.4 255.255.255.0
```

предпишет серверу назначить клиенту адрес 10.9.1.4. Обратите внимание, что адрес должен входить в диапазон, заданный директивой server, но не должен входить в пул, заданный директивой ifconfig-pool.

6.4.8 Конфигурирование клиент-специфичных правил и политик доступа

Предположим, что мы настраиваем VPN для компании, и нам бы хотелось установить различные политики доступа для 3 различных классов пользователей:

- Системные администраторы — полный доступ ко всем машинам сети
- Сотрудники — доступ только к серверам Samba/email
- Контракторы — доступ только к специальному серверу

Базовый подход, который мы предпримем:

1. выделить каждому классу собственный диапазон IP-адресов,
2. контролировать доступ к машинам, установив брандмауэрные правила, которые основываются на виртуальном IP-адресе клиента.

В нашем примере, предположим, что у нас меняется количество сотрудников, но только один системный администратор и два контрактора. Наш подход к распределению IP будет состоять в том, чтобы поместить всех сотрудников в диапазон IP-адресов, а потом выдать фиксированные IP-адреса системному администратору и контракторам.

Заметьте, что одно из требований к этому примеру заключается в том, что у вас есть программный брандмауэр, работающий на серверной машине «OpenVPN-ГОСТ», который дает вам возможность задавать конкретные брандмауэрные правила. Для нашего примера мы предположим, что этот брандмауэр - iptables в Linux.

Сначала создадим карту виртуальных IP-адресов в соответствии с каждым классом:

Класс	Диапазон виртуальных IP	Позволенный доступ к локальной сети	Поля common name
Сотрудники	10.9.0.0/24	Сервер samba/email на 10.66.4.4	[различные]
Системные администраторы	10.9.1.0/24	Вся подсеть 10.66.4.0/24	sysadmin1
Контракторы	10.9.2.0/24	Сервер контракторов на 10.66.4.12	contractor1, contractor2

Теперь переведем эту карту в серверную конфигурацию «OpenVPN-ГОСТ». Прежде всего убедитесь, что вы выполнили вышеописанные шаги, сделав подсеть 10.66.4.0/24 доступной для всех клиентов (хотя мы сконфигурируем маршрутизацию так, чтобы позволить клиентский доступ ко всей подсети 10.66.4.0/24, затем мы наложим ограничения доступа, пользуясь брандмауэрными правилами, чтобы реализовать вышеприведенную таблицу политик).

Прежде всего определим статический номер для нашего интерфейса tun, чтобы мы могли позже ссылаться на него в наших брандмауэрных правилах:

dev tun0

В серверном конфигурационном файле определим диапазон IP-адресов для сотрудников:

server 10.9.0.0 255.255.255.0

Добавим маршруты для диапазонов системного администратора и контракторов:

route 10.9.1.0 255.255.255.0 10.9.0.1

route 10.9.2.0 255.255.255.0 10.9.0.1

Назначим фиксированные IP-адреса системным администраторам и контракторам, как описано в 6.4.7: зададим каталог клиентских конфигураций

```
client-config-dir ccd
```
и поместим в него специальные конфигурационные файлы:

ccd/sysadmin1

```
ifconfig-push 10.9.1.1 255.255.255.255
push "route 10.9.0.0 255.255.255.0 10.9.1.1"
```

ccd/contractor1

```
ifconfig-push 10.9.2.1 255.255.255.255
push "route 10.9.0.0 255.255.255.0 10.9.2.1"
```

ccd/contractor2

```
ifconfig-push 10.9.2.5 255.255.255.255
push "route 10.9.0.0 255.255.255.0 10.9.2.5"
```

Это завершает конфигурацию «OpenVPN-ГОСТ». Последний шаг — добавить брадмауэрные правила, чтобы финализировать политику доступа. Для этого примера мы воспользуемся правилами в синтаксисе iptables из Linux:

```
# Employee rule
iptables -A FORWARD -i tun0 -s 10.9.0.0/24 -d 10.66.4.4 -j ACCEPT

# Sysadmin rule
iptables -A FORWARD -i tun0 -s 10.9.1.0/24 -d 10.66.4.0/24 -j ACCEPT

# Contractor rule
iptables -A FORWARD -i tun0 -s 10.9.2.0/24 -d 10.66.4.12 -j ACCEPT
```

6.4.9 Использование альтернативных способов аутентификации

«OpenVPN-ГОСТ» имеет возможность позволять своему серверу безопасно получать логин и пароль от подключающегося клиента и использовать эту информацию как базис для аутентификации клиента.

Чтобы использовать этот способ аутентификации, сначала добавьте директиву auth-user-pass в клиентскую конфигурацию. Она заставит клиент «OpenVPN-ГОСТ» запрашивать у пользователя логин/пароль, передавая его на сервер по безопасному TLS-каналу. Обратите внимание на описанные в разделе 7.2 особенности запуска OpenVPN-ГОСТ на ОС Linux в случае, если пользователь должен вводить логин и пароль.

Далее, сконфигурируйте сервер так, чтобы он использовал аутентификационный плагин, который может быть скриптом или динамической библиотекой. Сервер «OpenVPN-ГОСТ» будет вызывать этот плагин каждый раз, когда клиент VPN будет пытаться подключиться, передавая ему логин и пароль, введенные на клиенте. Аутентификационный плагин может управлять тем, позволяет ли сервер «OpenVPN-ГОСТ» клиенту соединиться, возвращая значение неудачи (1) или успеха (0).

6.4.9.1 Использование скриптовых плагинов

Скриптовые плагины можно применять, добавив директиву auth-user-pass-verify в серверный конфигурационный файл, а также разрешить программе вызывать скрипты, определяемые пользователем, добавив директиву script-security 2. Необходимость этого разрешения делает

использование скриптовых плагинов несколько менее безопасным по сравнению с использованием плагинов в форме динамических библиотек.

В комплект поставки входит плагин auth-pam.pl на языке perl, проверяющий переданные клиентом логин и пароль с использованием PAM-модуля login, соединение будет установлено только в том случае, если в операционной системе сервера есть пользователь с такими логином и паролем. Для использования этого плагина необходимо в конфигурационный файл сервера добавить строки

```
auth-user-pass-verify /opt/openvpn-gost/plugins/auth-pam.pl via-file
script-security 2
```

Для работы скрипта в системе должна быть установлена perl-библиотека Authen::PAM. На линуксах на базе deb-пакетов эта библиотека входит в состав пакета libauthen-pam-perl, на линуксах на базе rpm-пакетов – в состав пакета perl-Authen-PAM, на FreeBSD – в состав пакета p5-Authen-PAM.

Примечание: В RedHat и CentOS в основном репозитории нет пакета perl-Authen-PAM, для его установки необходимо подключить репозитоий EPEL (https://docs.fedoraproject.org/en-US/epel/), в российских дистрибутивах на базе CentOS 7 пакет можно поставить командой yum install https://mirror.yandex.ru/epel/7/x86_64/Packages/p/perl-Authen-PAM-0.16-16.el7.x86_64.rpm В SUSE и OpenSUSE нет пакета, содержащего Authen::PAM, библиотеку можно поставить из CPAN.

Этот скрипт не будет работать на ОС Windows и OpenWRT, так как в состав этих ОС не входит PAM-модуль.

6.4.9.2 Использование динамических библиотек в качестве плагинов

Плагины в форме динамичеких библиотек обычно являются скомпилированными модулями на языке C, которые загружаются сервером «OpenVPN-ГОСТ» в ходе работы. В комплект поставки входит плагин openvpn-plugin-auth-pam.so, проверяющий переданные клиентом логин и пароль с использованием PAM-модуля login, соединение будет установлено только в том случае, если в операционной системе сервера есть пользователь с такими логином и паролем. Для использования этого плагина необходимо в конфигурационный файл сервера добавить строку

```
plugin /opt/openvpn-gost/plugins/openvpn-plugin-auth-pam.so login
```

Данный плагин не поставляется для ОС Windows и OpenWRT, так как в состав этих ОС не входит PAM-модуль.

6.4.9.3 Использование аутентификации по логину и паролю как единственной формы клиентской аутентификации

По умолчанию, использование auth-user-pass-verify или проверяющего логины и пароли плагина на сервере включит двойную аутентификацию, требуя, чтобы для аутентификации клиента были успешны и аутентификация на клиентском сертификате, и логин-парольная аутентификация.

Хотя с точки зрения безопасности это не рекомендуется, возможно отключить использование клиентских сертификатов и оставить только аутентификацию по логину и паролю. На сервере:

```
client-cert-not-required
```

Такие конфигурации обычно должны иметь строку:

```
username-as-common-name
```

Которая велит серверу использовать логин для целей индексации, как он использовал бы поле Common Name для клиента, аутентифицировавшегося на клиентском сертификате.

Обратите внимание, что строка client-cert-not-required не устраняет необходимость в серверном сертификате, так что клиент, соединяющийся с сервером, который использует client-cert-not-required, может удалить директивы cert и key из клиентского конфигурационного файла, но не директиву ca, потом что клиенту необходимо проверить серверный сертификат.

6.4.10 Добавление к конфигурации «OpenVPN-ГОСТ» двухфакторной аутентификации с использованием токенов на клиентской стороне

6.4.10.1 О двухфакторной аутентификации

Двухфакторная аутентификация — это способ аутентификации, который объединяет два элемента: что-то, что у вас есть, и что-то, что вы знаете.

Что-то, что у вас есть, должно быть устройством, которое не может быть продублировано; такое устройство может быть криптографическим токеном, содержащим закрытый ключ. Этот закрытый ключ генерируется внутри устройства и никогда его не покидает. Если пользователь, обладающий этим токеном, пытается подключиться к защищенным сервисам в удаленной сети, процесс авторизации, который дает доступ к сети или отказывает в нем, может установить, с высокой степенью уверенности, что подключающийся пользователь физически обладает известным, сертифицированным токеном.

Что-то, что вы знаете, может быть паролем (пин-кодом) для доступа к криптографическому устройству. Без предоставления корректного пароля вы не можете воспользоваться закрытым ключом. Еще одна возможность криптографических устройств — запретить использование закрытого ключа, если некорректный пароль (пин-код) был предоставлен больше, чем позволенное количество раз. Это поведение гарантирует, что если пользователь потерял свое устройство, другой человек не сможет его использовать.

Таким образом, использование токенов повышает защищенность виртуальной сети, поскольку затрудняет подключение к ней неуполномоченного пользователя.

Обратите внимание, что из-за ошибки в системных библиотеках возможны проблемы при работе с ключами на аппаратных токенах в операционных системах SUSE Linux, ROSA RED и Альт, а в операционной системе macOS работа с токенами возможна только при запуске программы без использования опции –daemon.

6.4.10.2 Конфигурирование «OpenVPN-ГОСТ» для работы с токенами Рутокен

«МагПро КриптоПакет» 4.0 позволяет хранить закрытый ключ не только в файле, но и на устройсте Рутокен. Существует два способа хранения ключей на Рутокене: в файловой системе устройства или в неизвлекаемой памяти (отметим, что второй вариант доступен только для устройств Рутокен ЭЦП и Рутокен PKI).

Ключ в файловой системе устройства должен быть создан средствами самого «МагПро КриптоПакет» 4.0. Для использования такого ключа необходимо установить драйверы устройства и в файле конфигурации отредактировать параметр **key** следущим образом:

key [[ENGINE]]cryptocom:RUTOKEN:*имя_контейнера*

Имя_контейнера является числом в десятичной или шестнадцатеричной записи и должно совпадать с *именем_контейнера*, использованным при генерации ключа.

Рутокен с ключом в неизвлекаемой памяти обычно приходит из Удостоверяющеего центра. Также можно создать такой ключ самостоятельно с помощью утилит из комплекта поставки Рутокена или средствами «МагПро КриптоПакет» 4.0.

Для использования ключа в неизвлекаемой памяти устройства Рутокен необходимо установить драйверы устройства и библиотеку rtpkcs11ecp, предоставляющую интерфейс PKCS#11, после чего в файле конфигурации отредактировать параметр **key** следущим образом:

key [[ENGINE]]cryptocom:PKCS11:*метка_ключа*

Если в неизвлекаемой памяти Рутокен содержится только один ключ, *метку_ключа* можно не указывать.

Обратите внимание, что есть Рутокен защищён PIN-кодом, то на ОС Linux при запуске OpenVPN-ГОСТ необходимо выполнить дополнительные действия, описанные в разделе 7.2.

6.4.10.3 Конфигурирование «OpenVPN-ГОСТ» для работы с токенами JaCarta и другими токенами, предоставляющими интерфейс PKCS#11

«МагПро КриптоПакет» 4.0 позволяет использовать закрытый ключ, расположенный в неизвлекаемой памяти устройства JaCarta, а также иных устройство, предоставляющих интерфейс PKCS#11. Совместно с «МагПро КриптоПакет» 4.0 допускается использовать только устройства, имеющие действующий сертификат ФСБ России.

Для использования таких ключей необходимо установить драйверы устройства, а также библиотеку, предоставляющую интерфейс PKCS#11 для этого устройства, и выставить переменную окружения `PKCS11_LIBNAME`, значением которой должен быть путь к этой библиотеке.

В файле конфигурации необходимо отредактировать параметр **key** следущим образом:

key [[ENGINE]]cryptocom:PKCS11:*метка_ключа*

Если в неизвлекаемой памяти токен содержится только один ключ, *метку_ключа* можно не указывать.

Обратите внимание, что есть токен защищён PIN-кодом, то на ОС Linux при запуске OpenVPN-ГОСТ необходимо выполнить дополнительные действия, описанные в разделе 7.2.

6.4.10.4 Конфигурирование «OpenVPN-ГОСТ» для работы с токенами Вьюга

«OpenVPN-ГОСТ» может использовать закрытые ключи, записанные в файловую систему токенов Вьюга (должны быть дополнительно установлены драйвера устройства). Данная возможность реализована только для 256-битных ключей.

Для использования ключей с Вьюги параметр key файла конфигурации «OpenVPN-ГОСТ» нужно задать следующим образом:

key [[ENGINE]]cryptocom:VJUGA.X

6.4.11 Маршрутизация всего клиентского трафика (включая веб-трафик) через VPN

6.4.11.1 Введение

По умолчанию, когда клиент «OpenVPN-ГОСТ» активен, только сетевой трафик на сервер «OpenVPN-ГОСТ» и с него будет проходить через VPN. Общий просмотр интернет-сайтов, например, будет выполнен через прямые соединения, которые не проходят через VPN.

В некоторых случаях это поведение может быть нежелательным — вы можете захотеть, чтобы VPN-клиент туннелировал весь сетевой трафик через VPN, включая общий просмотр интернет-сайтов. Хотя этот тип конфигурации VPN заставит клиент работать медленнее, он дает администратору VPN больший контроль над политиками безопасности, когда клиент одновременно соединен и с публичной сетью Интернет, и с VPN.

6.4.11.2 Реализация

Добавьте следующую директиву к серверному конфигурационному файлу:

```
push "redirect-gateway def1"
```

Если ваша VPN настроена в беспроводной сети, где все клиенты и сервер находятся в одной и той же беспроводной подсети, поставьте флаг local:

```
push "redirect-gateway local def1"
```

Передача опции redirect-gateway клиентам заставит весь IP-сетевой трафик, исходящий с клиентских машин, проходить через сервер «OpenVPN-ГОСТ» Сервер нужно будет как-то сконфигурировать, чтобы он мог работать с этим трафиком, например, соединить его с сетью Интернет через NAT или маршрутизировать через HTTP-прокси сервера.

В Linux вы можете воспользоваться вот такой командой, чтобы подключить клиентский трафик к сети Интернет через NAT:

```
iptables -t nat -A POSTROUTING -s 10.9.1.0/24 -o eth0 -j MASQUERADE
```

Эта команда предполагает, что подсеть VPN — 10.9.1.0/24 (взятая из директивы server в серверной конфигурации «OpenVPN-ГОСТ»), и что локальный интерфейс сети Ethernet — eth0.

Когда используется redirect-gateway, клиенты «OpenVPN-ГОСТ» будут маршрутизировать запросы DNS через VPN, и серверу VPN нужно будет их обрабатывать. Это может быть достигнуто передачей адреса сервера DNS подключающимся клиентам, который заместит их обычные настройки сервера DNS на время, пока VPN активна. Например:

```
push "dhcp-option DNS 10.9.1.1"
```

сконфигурирует клиенты Windows (или других операционных систем с некоторыми дополнительными скриптами) использовать 10.9.1.1 в качестве их DNS-сервера. Любой адрес, который виден с клиентов, может быть использован как адрес сервера DNS.

6.4.11.3 Предупреждения

Перенаправление всего сетевого трафика через VPN — не совсем беспроблемное предложение. Вот несколько типичных проблем, которые следует иметь в виду:

- Многие машины-клиенты «OpenVPN-ГОСТ», соединяющиеся с Интернетом, будут периодически взаимодействовать с сервером DHCP, чтобы обновить свою аренду IP-адресов. Опция redirect-gateway может не дать клиенту связаться с локальным DHCP0-сервером (потому что сообщения DHCP будут маршрутизироваться через VPN), что заставит их потерять свою аренду IP-адресов.
- Существуют проблемы, связанные с передачей DNS-адресов на клиенты Windows.
- Просмотр интернет-сайтов на клиенте будет заметно медленнее.

6.4.12 Работа сервера «OpenVPN-ГОСТ» на динамическом IP-адресе

В то время как клиенты «OpenVPN-ГОСТ» могут легко достигать сервера через динамический IP-адрес без какого-либо специального конфигурирования, возникают определенные проблемы, когда сам сервер имеет динамический адрес. Хотя «OpenVPN-ГОСТ» без проблем справляется с ситуацией на динамическом сервере, необходимо некоторое дополнительное конфигурирование.

Первый шаг — получить динамический DNS-адрес, который может быть сконфигурирован так, чтобы «следовать» за сервером каждый раз, как меняется IP-адрес сервера. Существуют несколько доступных провайдеров динамических DNS-услуг, например, dyndns.org.

Следующий шаг — настроить механизм, чтобы каждый раз, когда IP-адрес сервера меняется, имя динамического DNS быстро получало бы новый IP-адрес, позволяя клиентам найти сервер по его новому IP-адресу. Есть два основных способа это сделать:

- Использовать оборудование маршрутизатора NAT с поддержкой динамического DNS (например, Linksys BEFSR41). Большинство недорогих и широкодоступных маршрутизаторов NAT обладают возможностью обновлять динамическое имя DNS каждый раз, как новая аренда DHCP получена с ISP. Эта настройка идеальна, когда сервер «OpenVPN-ГОСТ» является компьютером с одним сетевым адаптером внутри брандмауэра.
- Использовать динамическое приложение DNS-клиента, например, ddclient (http://sourceforge.net/apps/trac/ddclient) для обновления динамического DNS-адреса, как только поменяется IP-адрес сервера. Эта настройка идеальна, когда машина, на которой работает «OpenVPN-ГОСТ», имеет несколько сетевых адаптеров и работает как брандмауэр/гейт. Чтобы реализовать эту настройку, вам необходимо настроить скрипт, который будет выполняться вашим программным приложением DHCP-клиента каждый раз, как происходит смена IP-адреса. Этот скрипт должен а) выполнять ddclient, чтобы уведомить ваш провайдер динамического DNS о вашем новом IP-адресе и б) перезапускать серверный демон «OpenVPN-ГОСТ».

Клиент «OpenVPN-ГОСТ» по умолчанию почувствует, когда сменится IP-адрес сервера, если клиентская конфигурация использует директиву remote, которая соотносится с именем динамического DNS. Обычная цепь событий такова: а) клиенту «OpenVPN-ГОСТ» не удается получить своевременные поддерживающие сообщения со старого IP-адреса сервера, что запускает перезагрузку, и б) перезагрузка заставляет пересмотреть имя DNS в директиве remote, позволяя клиенту переподключиться к серверу по его новому IP-адресу.

6.4.13 Подключение к серверу «OpenVPN-ГОСТ» через HTTP-прокси

«OpenVPN-ГОСТ» поддерживает соединения через HTTP-прокси, со следующими моделями аутентификации:

- Нет аутентификации на прокси
- Basic аутентификация на прокси
- NTLM аутентификация на прокси

Добавьте директиву http-proxy в клиентский конфигурационный файл.

Например, предположим, что у вас HTTP-прокси сервер на клиентской локальной сети на 192.168.4.1, который слушает соединения на порту 1080. Добавьте следующую строку к клиентской конфигурации:

```
http-proxy 192.168.4.1 1080
```

Предположим, что HTTP-прокси требует аутентификации Basic:

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

```
http-proxy 192.168.4.1 1080 stdin basic
```

Предположим, что HTTP-прокси требует аутентификации NTLM:

```
http-proxy 192.168.4.1 1080 stdin ntlm
```

Два вышеприведенных аутентификационных примера заставят «OpenVPN-ГОСТ» запросить логин и пароль со стандартного ввода. Если вы вместо этого предпочитаете поместить логин и пароль в файл, замените stdin именем файла и поместите логин в первую строку этого файла, а пароль во вторую.

6.4.14 Соединение с совместно используемым ресурсом Samba через «OpenVPN-ГОСТ»

Этот пример предназначен для того, чтобы показать, как клиенты «OpenVPN-ГОСТ» могут соединяться с совместно используемым ресурсом Samba через маршрутизированный туннель dev tun. Если вы используете «OpenVPN-ГОСТ» типа «мост» (dev tap), вам, вероятно, не нужно следовать этим инструкциям, поскольку клиенты «OpenVPN-ГОСТ» должны видеть машины серверной стороны в своем сетевом окружении.

Для этого примера предположим, что:

- Серверная локальная сеть использует подсеть 10.66.0.0/24,
- Диапазон IP-адресов VPN использует 10.9.1.0/24 (как указано в директиве server в серверном конфигурационном файле «OpenVPN-ГОСТ».)
- Сервер Samba имеет IP-адрес 10.66.0.4 и
- Сервер Samba уже был сконфигурирован и доступен из местной локальной сети.

Если серверы Samba и «OpenVPN-ГОСТ» работают на различных машинах, удостоверьтесь, что вы выполнили указания раздела «Расширение области действия VPN с включением дополнительных машин в клиентскую или серверную подсеть».

Далее отредактируйте конфигурационный файл сервера Samba (smb.conf). Удостоверьтесь, что директива hosts allow позволит клиентам «OpenVPN-ГОСТ», приходящим из подсети 10.9.1.0/24, устанавливать соединение. Например:

```
hosts allow = 10.66.0.0/24 10.9.1.0/24 127.0.0.1
```

Если серверы Samba и «OpenVPN-ГОСТ» работают на одной и той же машине, вы, возможно, захотите отредактировать директиву interfaces в файле smb.conf так, чтобы также слушать на подсети 10.9.1.0/24 интерфейса TUN:

```
interfaces  = 10.66.0.0/24 10.9.1.0/24
```

Если серверы Samba и «OpenVPN-ГОСТ» работают на одной и той же машине, подключитесь с клиента «OpenVPN-ГОСТ» к совместно используемому ресурсу Samba с использованием имени каталога:

```
\\10.9.1.1\\sharename
```

Если серверы Samba и «OpenVPN-ГОСТ» работают на разных машинах, используйте имя каталога:

```
\\10.66.0.4\sharename
```

Например, из окна командной строки:

```
net use z: \\10.66.0.4\sharename /USER:myusername
```

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

6.4.15 Реализация конфигурации балансировки нагрузки/восстановления после сбоя

6.4.15.1 Клиент

Клиентская конфигурация «OpenVPN-ГОСТ» может указывать на несколько серверов для балансировки нагрузки и восстановления после отказа. Например:

```
remote server1.mydomain
remote server2.mydomain
remote server3.mydomain
```

прикажет клиенту «OpenVPN-ГОСТ» пытаться устанавливать соединение с server1, server2 и server3 в этом порядке. Если существующее соединение разорвано, клиент OpenVPN попытается восстановить соединение с последним сервером, с которым соединение было установлено, а если это не удастся, перейдет на следующий сервер в списке. Вы также можете указать клиенту «OpenVPN-ГОСТ» рандомизовать свой список серверов при загрузке, чтобы клиентская нагрузка была вероятностно распределена по диапазону серверов.

```
remote-random
```

Если вы также хотите, чтобы неудачи разрешения DNS заставляли клиент «OpenVPN-ГОСТ» переходить на следующий сервер в списке, добавьте следующее:

```
resolv-retry 60
```

Параметр 60 велит клиенту «OpenVPN-ГОСТ» пытаться разрешить каждое удаленное имя DNS в течение 60 секунд, прежде чем переходить на следующий сервер в списке.

Список серверов может также указывать на несколько серверных демонов «OpenVPN-ГОСТ», работающих на одной и той же машине, каждый из которых слушает соединения на собственном порту, например:

```
remote smp-server1.mydomain 8000
remote smp-server1.mydomain 8001
remote smp-server2.mydomain 8000
remote smp-server2.mydomain 8001
```

6.4.15.2 Список серверов

Если ваши сервера — многопроцессорные машины, запуск нескольких демонов «OpenVPN-ГОСТ» на каждом сервере может иметь преимущества с точки зрения скорости выполнения.

«OpenVPN-ГОСТ» также поддерживает директиву remote, указывающую на имя DNS, которое обладает несколькими записями A в зонной конфигурации для домена. В этом случае клиент «OpenVPN-ГОСТ» будет случайным образом выбирать одну из записей A каждый раз, как домен разрешается.

6.4.15.3 Сервер

Самый простой подход к конфигурации балансировки нагрузки и восстановления после отказа на сервере — использовать эквивалентные конфигурационные файлы на каждом сервере в кластере, за исключением использования различных диапазонов виртуальных IP-адресов для каждого сервера. Например:

server1
server 10.9.0.0 255.255.255.0
server2

server 10.9.1.0 255.255.255.0
server3
server 10.9.2.0 255.255.255.0

6.4.16 Конфигурирование работы по IPv6

«OpenVPN-ГОСТ» может как создавать защищенное соединение, используя IPv6 в качестве транспорта, так и передавать IPv6-трафик внутри защищенного соединения.

Для того, чтобы создать защищенное соединение поверх IPv6, нужно заменить параметр proto tcp на proto tcp6. При этом на стороне клиента адрес сервера можно указывать как в IPv6-формате, так и (при наличии работающего DNSv6-сервера) в формате доменного имени.

Для того, чтобы включить передачу IPv6 внутри защищенного соединения, нужно указать серверу параметр server-ipv6. Обратите внимание, что параметр server, задающий адрес IPv4, тоже должен присутствовать, т.е. pure IPv6 VPN создать невозможно.

6.4.17 Конфигурирование «OpenVPN-ГОСТ» для автоматического запуска при старте системы

Недостаток стандартов в этой области означает, что большинство операционных систем имеет собственный способ конфигурирования демонов/сервисов для запуска при запуске системы.

6.4.17.1 Linux

При установке «OpenVPN-ГОСТ» на ОС Linux из пакета ставятся файлы, необходимые для его запуска через systemd, однако автоматический запуск при старте системы не настраивается. Для того, чтобы «OpenVPN-ГОСТ» автоматически запускался при старте системы, нужно выполнить команду

```
sudo systemctl enable openvpn-gost
```

Для того, чтобы «OpenVPN-ГОСТ» автоматически запускался при старте системы с неумолчательным файлом конфигурации, нужно выполнить команду

```
sudo systemctl enable openvpn-gost@имя-конфига-без-расширения
```

6.4.17.2 Windows

Инсталлятор Windows установит сервисную оболочку, но оставит ее отключенной по умолчанию. Чтобы активировать ее, идите в Control Panel / Administrative Tools / Services, выберите сервис «OpenVPN-ГОСТ», щелкните правой клавишей на свойствах и установите Startup Type в Automatic. Это сконфигурирует сервис для автоматического старта при следующей перезагрузке.

При запуске сервисная оболочка «OpenVPN-ГОСТ» просканирует каталог C:\Cryptopack4\config в поисках конфигурационных файлов .ovpn и зарустит отдельный процесс «OpenVPN-ГОСТ» для каждого файла.

6.4.18 Запуск VPN и тест на начальную подключаемость

6.4.18.1 Запуск сервера

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

Сначала убедитесь, что сервер «OpenVPN-ГОСТ» будет доступен из интернета. Это значит:

– Открыть TCP-порт 1194 на брандмауэре (или любой другой TCP-порт, который вы сконфигурировали) или
– Установить правило форвардинга порта, чтобы произвести форвардинг TCP-порта 1194 с брандмауэра/гейта на машину, где работает сервер «OpenVPN-ГОСТ».

Далее, убедитесь, что интерфейс TUN/TAP не за брандмауэром.

Чтобы упростить поиск ошибок, лучше сначала запустить сервер «OpenVPN-ГОСТ» из командной строки (или щелкнуть правой клавишей по файлу .ovpn в Windows), а не запускать ее как демон или сервис:

```
openvpn [server config file]
```

6.4.18.2 Запуск клиента

Как и в случае серверной конфигурации, лучше запустить клиент «OpenVPN-ГОСТ» из командной строки (или в Windows щелкнуть правой клавишей мыши по файлу client.ovpn), а не запускать ее как демон или сервис:

```
openvpn [client config file]
```

Нормальное начало работы клиента на Windows будет похоже на серверный вывод, приведенный выше, и должен закончиться сообщением Initialization Sequence Completed.

Теперь попробуйте отправить сигнал ping через VPN от клиента. Если вы используете маршрутизацию (т.е. dev tun в серверном конфигурационном файле), попробуйте:

```
ping 10.9.1.0
```

Если вы используете VPN типа «мост» (т.е. dev tap в серверном конфигурационном файле), попытайтесь отправить сигнал ping на IP-адрес машины на подсети серверной сети Ethernet.

Если сигнал проходит успешно, поздравляем! У вас теперь действующая VPN.

6.4.18.3 Поиск ошибок

Если сигнал ping не прошел или инициализация клиента «OpenVPN-ГОСТ» не закончилась, вот список обычных симптомов и их решений:

1. Вы получаете сигнал ошибки: TLS Error: TLS key negotiation failed to occur within 60 seconds (check your network connectivity). Эта ошибка указывает, что клиент не сумел установить сетевую связь с сервером.
   Решения:
   – Убедитесь, что клиент использует корректный hostname/IP-адрес и номер порта, который позволит ему соединиться с сервером «OpenVPN-ГОСТ».
   – Если машина сервера «OpenVPN-ГОСТ» является компьютером с одним сетевым адаптером внутри защищенной LAN, убедитесь, что вы пользуетесь корректным правилом форвардинга порта на брандмауэр сервера. Например, предположим, что ваш сервер «OpenVPN-ГОСТ» находится на 192.168.4.4 внутри брандмауэра, слушая клиентские подключения на TCP-порту 1194. Гейт NAT, обслуживающий подсеть 192.168.4.x должен иметь правило форвардинга порта, которое говорит «переадресовать TCP-порт 1194 с моего публичного IP-адреса на 192.168.4.4.»
   – Откройте брандмауэр сервера, чтобы разрешить входящие соединения на TCP-порт 1194 (или любой TCP-порт, который вы сконфигурировали в серверном конфигурационном файле).

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

2. Вы получаете сообщение об ошибке: Initialization Sequence Completed with errors — эта ошибка может произойти на Windows, если а) у вас нет запущенного доступного сервиса DHCP или б) вы используете персональный брандмауэр, отличный от Windows Defender. Решение: запустите сервер для клиента DHCP, если проблема вызвана брандмауэром, обратитесь за консультацией к его производителю.
3. Вы получаете сообщение Initialization Sequence Completed, но сигнал ping не проходит — это обычно связано с политикой настройки брандмауэра на серверной или клиентской стороне.
   Решение: разрешите посылку эхо-ответов через расширенные настройки брандмауэра или объявите сеть VPN частной и выключите брандмауэр для частных сетей.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

7 Использование

7.1 Запуск на ОС семейства Windows

После того, как вы выполните установку, «OpenVPN-ГОСТ» готова к использованию и ассоциирована с файлами с расширением .ovpn.

Чтобы запустить «OpenVPN-ГОСТ», вы можете:

- Щелкнуть правой клавишей на конфигурационном файле «OpenVPN-ГОСТ» (.ovpn) и выбрать Start OpenVPN-GOST с этим конфигурационным файлом. Чтобы выйти, можно воспользоваться клавишей F4.
- Запустить «OpenVPN-ГОСТ» из командной строки Windows такой командой, как:

  ```
  openvpn-gost [config file]
  ```

  Запущенную из окна командной строки «OpenVPN-ГОСТ» можно отключить клавишей F4.
- Запустить «OpenVPN-ГОСТ» как сервис, поместив один или больше конфигурационных файлов .ovpn в каталог C:\Cryptopack4\\config и запустив сервис «OpenVPN-ГОСТ», который может управляться из Start Menu -> Control Panel -> Administrative Tools -> Services.

7.2 Запуск на ОС Linux

Запуск «OpenVPN-ГОСТ» на ОС Linux выполняется через systemd командой

```
sudo -H systemctl start openvpn-gost
```

По этой комнанде программа стартует с умолчательным файлом конфигурации /etc/openvpn-got/openvpn-gost.conf, для запуска с другим файлом конфигурации можно использовать команду

```
sudo -H systemctl start openvpn-gost@имя-конфига-без-расширения
```

Если используются ключи на токене, требующем ввода PIN-кода, а также если используется ключ, защищённый паролем, либо парольная аутентификация пользователей (см. 6.4.9), после запуска OpenVPN-ГОСТ необходимо дать команду

```
sudo systemd-tty-ask-password-agent --query
```

и по запросу программы ввести PIN-код или пароль.

7.3 Запуск на macOS

Для запуска «OpenVPN-ГОСТ» на macOC используйте команду

```
sudo -H /opt/openvpn-gost/sbin/start_stop start
```

Если используется ключ, защищённый паролем, то в команде запуска необходимо дополнительно указать опцию --askpass:

```
sudo -H /opt/openvpn-gost/sbin/start_stop start --askpass
```

и по запросу программы ввести пароль.

Если же ключи у вас хранятся на токене, то для запуска «OpenVPN-ГОСТ» необходимо использовать команду

```
sudo -H /opt/openvpn-gost/sbin/start_stop_token start
```

При этом окно командной строки, в которм запущена программа, нельзя закрывать до окончания её работы.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

7.4 Запуск на ОС FreeBSD

Для запуска «OpenVPN-ГОСТ» на ОС FreeBSD используйте команду
`sudo -H /etc/rc.d/openvpngost start`

7.5 Запуск на ОС Solaris

Для запуска «OpenVPN-ГОСТ» на ОС Solaris используйте команду
`sudo -H /etc/init.d/openvpn-gost start`

7.6 Запуск на ОС OpenWRT

Для запуска «OpenVPN-ГОСТ» на ОС OpenWRT используйте команду
`/etc/rc.d/openvpngost start`

7.7 Управление запущенным процессом «OpenVPN-ГОСТ»

7.7.1 Работа на Linux/BSD/Unix

«OpenVPN-ГОСТ» принимает несколько сигналов:

SIGUSR1 — условный перезапуск, предназначенный для перезапуска без привилегий суперпользователя
SIGHUP — жесткий перезапуск
SIGUSR2 — статистика исходящего соединения в файл журнала или системный журнал
SIGTERM, SIGINT — выход

Используйте директиву writepid, чтобы записать PID демона «OpenVPN-ГОСТ» в файл, чтобы вы знали, куда отправить сигнал (если вы запускаете openvpn с помощь initscript, скрипт может уже передавать директиву –writepid в командную строку openvpn).

7.7.2 Работа в Windows в графическом интерфейсе

Хотя «OpenVPN-ГОСТ» может быть запущена как демон, сервис или из командной строки, возможно управлять «OpenVPN-ГОСТ» через графический интерфейс.

7.7.3 Работа в окне командной строки Windows

В Windows вы можете запустить «OpenVPN-ГОСТ», кликнув правой клавишей мыши на конфигурационном файле «OpenVPN-ГОСТ» (файл .ovpn) и выбрав Start OpenVPN on this config file.

Если «OpenVPN-ГОСТ» запущена таким образом, доступны несколько клавишных команд:

F1 — условный перезапуск (не закрывает/переоткрывает адаптер TAP)
F2 — показать статистику соединения
F3 — жесткий перезапуск
F4 — выход

7.7.4 Работа в качестве сервиса Windows

Когда «OpenVPN-ГОСТ» запущена в качестве сервиса на Windows, управлять ей можно только:

- Через менеджер управления сервисами (Control Panel / Administrative Tools / Services), который дает возможность запускать и выключать сервис;
- Через интерфейс управления (см. раздел 7.7.7)

7.7.5 Модификация конфигурации запущенного сервера

Хотя большинство изменений конфигурации требуют перезапуска сервера, есть две директивы, связанные с файлами, которые могут быть динамически изменены в ходе работы, и которые немедленно повлияют на сервер без необходимости перезапускать процесс.

client-config-dir — эта директива устанавливает каталог клиентской конфигурации, которую «OpenVPN-ГОСТ» будет сканировать при каждом входящем соединении в поисках клиент-специфичного конфигурационного файла. Файлы в этом каталоге могут быть изменены в ходе работы, без перезапуска сервера. Обратите внимание, что изменения в этом каталоге повлияют только на новые соединения, не на существующие соединения. Если вы хотите, чтобы изменение клиент-специфичного конфигурационного файла немедленно подействовало на уже соединившегося клиента (или на клиента, который уже отключился, но сервер еще не уничтожил его экземпляр), уничтожьте экземпляр клиента, воспользовавшись интерфейсом управления (см. раздел 7.7.7). Это заставит клиента переподключиться и использовать новый файл client-config-dir.

crl-verify — эта директива именует список отзыва сертификатов, описанный ниже в разделе «Отзыв сертификатов». Файл списка отзыва сертификатов можно менять в ходе работы, и изменения немедленно повлияют на новые соединения, или на существующие соединения, которые обновляют свой канал SSL/TLS (по умолчанию это происходит раз в час). Если вы хотите уничтожить подключенного клиента, чей сертификат только что добавили в список отзыва сертификатов, воспользуйтесь интерфейсом управления (см. раздел 7.7.7).

7.7.6 Файл статуса

Умолчательный файл server.conf имеет строку

status openvpn-status.log

которая будет выводить список текущих клиентских соединений в файл openvpn-status.log раз в минуту.

7.7.7 Использование интерфейса управления

Интерфейс управления «OpenVPN-ГОСТ» позволяет административно управлять «OpenVPN-ГОСТ» из внешней программы через TCP-сокет.

Интерфейс был специально разработан для разработчиков графического интерфейса и для тех, кто хотел бы программно или удаленно управлять демоном «OpenVPN-ГОСТ».

Интерфейс управления реализован с использованием клиент-серверного TCP-соединения, где «OpenVPN-ГОСТ» будет слушать на предоставленном IP-адресе и порту входящие управляющие клиентские соединения.

Протокол управления сейчас текстовый без сложного защитного слоя. По этой причине рекомендуется, чтобы интерфейс управления слушал или на localhost (127.0.0.1) или на локаль-

ном адресе VPN. Возможно удаленное соединение с интерфейсом управления поверх самой VPN, хотя некоторые возможности в этом режиме будут ограничены, такие, как способность предоставлять пароли от закрытых ключей.

Интерфейс управления включается в конфигурационном файле «OpenVPN-ГОСТ» с помощью следующх директив:

–management

–management-query-passwords

–management-log-cache

Когда «OpenVPN-ГОСТ» запущена с подключенным слоем управления, вы можете подключиться через telnet к порту управления (обязательно используйте клиент telnet, который понимает «грубый» режим).

Подключившись к порту управления, вы можете использовать команду help для вывода списка всех команд.

7.7.7.1 Команда echo

Эта команда используется, чтобы дать возможность или вписать в конфигурационный файл «OpenVPN-ГОСТ», или передать на клиент «OpenVPN-ГОСТ» с сервера параметры, специфичные для графического интерфейса.

Примеры команды:

echo on — включает нотификацию сообщений echo в реальном времени

echo all — выводит текущий список истории echo

echo off — отключает нотификацию сообщений echo в реальном времени

echo on all — атомно включает нотификацию в реальном времени, плюс показывает все сообщения в буфере истории

Например, предположим, что вы разрабатываете графический интерфейс «OpenVPN-ГОСТ» и хотите дать серверу «OpenVPN-ГОСТ» способность просить интерфейс забыть все сохраненные пароли.

В серверном конфигурационном файле «OpenVPN-ГОСТ» добавьте:

```
push "echo forget-passwords"
```

Когда клиент «OpenVPN-ГОСТ» получает свой список директив с сервера, директива echo forget-passwords будет в списке и заставит интерфейс управления сохранить строку forget-passwords в своем списке параметров echo.

Клиент управления может использовать echo all для вывода полного списка параметров echo, echo on для включения нотификации этих параметров в реальном времени через префикс >ECHO:, или echo off для отключения нотификации в реальном времени.

Когда графический интерфейс подключается к сокету управления «OpenVPN-ГОСТ», он может отдать команду echo all, которая даст вывод типа:

```
1101519562,forget-passwords
END
```

По сути команда echo позволила нам передать параметры с сервера «OpenVPN-ГОСТ» на клиент, а затем на клиент управления (такой как графический интерфейс). Большое целое число — юниксовая дата/время, когда был получен параметр echo.

Если клиент управления отдал команду echo on, она включит нотификацию параметров echo в реальном времени. В этом случае наше сообщение forget-passwords будет иметь вывод типа:

```
>ECHO:1101519562,forget-passwords
```

Как и команда log, команда echo может атомно показывать историю, одновременно активируя обновления в реальном времени:

```
echo on all
```

Размер буфера echo в настоящее время жестко ограничен 100 сообщениями.

7.7.7.2 Команда exit, quit

Закрывает сеанс управления и восстанавливает слушание соединений от других клиентов на порту управления. В настоящее время демон «OpenVPN-ГОСТ» может поддерживать самое большее один клиент управления в любой момент времени.

7.7.7.3 Команда help

Выводит краткий список команд.

7.7.7.4 Команда hold

Команда hold может быть использована для манипуляций с флагом hold или освобождения «OpenVPN-ГОСТ» из состояния hold.

Если флаг hold установлен при первоначальном запуске или перезагрузке, «OpenVPN-ГОСТ» будет находиться в замершем состоянии перед инициализацией туннеля, пока интерфейс управления не получит команду hold release.

Директива «OpenVPN-ГОСТ» –management-hold может быть использована, чтобы запустить «OpenVPN-ГОСТ» с установленным флагом hold.

Установка флага hold постоянна и не будет переключена перезагрузками.

«OpenVPN-ГОСТ» укажет, что она в состоянии hold, отправив нотификацию в реальном времени клиенту управления:

```
>HOLD:Waiting for hold release
```

Примеры команды:

hold — показать текущий флаг hold, 0=отключен, 1=включен.

hold on — включить флаг hold, чтобы будущие перезагрузки приводили в состояние hold.

hold off — отключить флаг hold, чтобы будущие перезагрузки не приводили в состояние hold.

hold release — выйти из состояния hold и запустить «OpenVPN-ГОСТ», но не изменить текущее состояние флага hold.

7.7.7.5 Команда kill

В режиме сервера уничтожить конкретный экземпляр клиента.

Примеры команды:

kill Test-Client — уничтожить экземпляр клиента с Common Name «Test-Client».

kill 1.2.3.4.4000 — уничтожить экземпляр клиента с исходящим адресом и портом 1.2.3.4.4000.

Используйте команду «status», чтобы посмотреть, какие клиенты подключены.

7.7.7.6 Команда log

Показывает файл журнала «OpenVPN-ГОСТ». Интерфейсом управления кэшируются только последние n строк файла журнала, где n управляется «OpenVPN-ГОСТ» — директивой management-log-cache.

Примеры команды:

log on — Включить вывод журнальных сообщений в реальном времени

log all — Показать текущую кэшированную историю файла журнала

log on all — Атомно показать всю текущую кэшированную историю файла журнала, потом включить вывод журнальных сообщений в реальном времени

log off — Отключить нотификацию сообщений журнала в реальном времени

log 20 — показать последние 20 строк истории файла журнала

Формат нотификации в реальном времени:

Сообщения журнала в реальном времени начинаются с префикса >LOG:, за которым следуют разделенные запятой поля:

1. юниксовое целое дата/время
2. ноль или больше флагов сообщений в одной строке:
   - I — информационное
   - F — фатальная ошибка
   - N — не фатальная ошибка
   - W — предупреждение
   - D — отладка, и
3. текст сообщения.

7.7.7.7 Команда mute

Меняет параметр «OpenVPN-ГОСТ» –mute. Этот параметр используется, чтобы не выводить повторяющиеся сообщения одной и той же категории сообщений.

Примеры команды:

mute 40 — установить параметр mute в 40

mute — показать текущее значение mute

7.7.7.8 Команда net

(Только для Windows) Дает эквивалент вывода директивы «OpenVPN-ГОСТ» –show-net. Вывод включает взгляд «OpenVPN-ГОСТ» на список системных сетевых адаптеров и таблицу маршрутизации, основанный на информации, возвращенной Windows IP helper API.

7.7.7.9 Команда password и username

Команда password используется, чтобы передавать пароли в «OpenVPN-ГОСТ».

Если «OpenVPN-ГОСТ» запущена с директивой –management-query-passwords, она будет запрашивать у интерфейса управления пароли закрытых ключей и пароль/логин –auth-user-pass.

Когда «OpenVPN-ГОСТ» нуждается в пароле с интерфейса управления, она выводит сообщение >PASSWORD: в реальном времени.

Пример 1:

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

```
>PASSWORD:Need 'Private Key' password
```

«OpenVPN-ГОСТ» указывает, что она нуждается в пароле типа «закрытый ключ».

Клиент управления должен отвечать на этот запрос так:

```
password "Private Key" foo
```

Пример 2:

```
>PASSWORD:Need 'Auth' username/password
```

«OpenVPN-ГОСТ» нуждается в пароле –auth-user-pass. Клиент управления должен ответить:

```
username "Auth" foo
password "Auth" bar
```

Сами логин и пароль могут быть в кавычках, и специальные символы, такие как двойные кавычки или обратный слэш, должны быть под escape-последовательностью, например:

```
password "Private Key" "foo\"bar"
```

Правила составления escape-последовательностей такие же, как для конфигурационного файла.

Тип сообщений PASSWORD в реальном времени может также использоваться, чтобы указать некорректность пароля или невыполнение других видов аутентификации:

Пример 3: пароль закрытого ключа некорректен и «OpenVPN-ГОСТ» завершает работу:

```
>PASSWORD:Verification Failed: 'Private Key'
```

Пример 4: логин/пароль –auth-user-pass некорректен, и «OpenVPN-ГОСТ» завершает работу:

```
>PASSWORD:Verification Failed: 'Auth'
```

7.7.7.10 Команда signal

Команда signal посылает сигнал демону «OpenVPN-ГОСТ». Сигнал может быть одним из SIGHUP, SIGTERM, SIGUSR1 или SIGUSR2.

Пример команды:

signal SIGUSR1 — посылает демону сигнал SIGUSR1

7.7.7.11 Команда state

Показывает текущее состояние «OpenVPN-ГОСТ», показывает историю состояний или включает нотификацию перемен состояния в реальном времени.

Существуют состояния «OpenVPN-ГОСТ»:

- CONNECTING — исходное состояние «OpenVPN-ГОСТ»
- WAIT — (только клиент) ждет первого ответа от сервера
- AUTH — (только клиент) аутентифицируется на сервере
- GET_CONFIG — (только клиент) загружает конфигурационные опции с сервера.
- ASSIGN_IP — присваивает IP-адрес виртуальному сетевому интерфейсу.
- ADD_ROUTES — добавляет к системе маршрутизацию
- CONNECTED — инициализационная последовательность завершена.
- RECONNECTING — произошла перезагрузка
- EXITING — в процессе аккуратного выхода.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

Примеры команды:
state — Вывести текущее состояние «OpenVPN-ГОСТ».
state on — включить нотификацию изменений состояния в реальном времени
state off — выключить нотификацию изменений состояния в реальном времени
state all — вывести текущую историю состояния
state 3 — вывести 3 последних перехода состояний
state on all — атомно показать историю состояний и в то же время включить нотификацию будущих переходов состояний в реальном времени.

Формат вывода состоит из 4 разделенных запятыми параметров:

1. целое юниксовое дата/время
2. наименование состояния
3. опциональная описательная строка (используется в основном при RECONNECTING или EXITING, чтобы показать причину разрыва связи)
4. опциональный локальный IP-адрес TUN/TAP (показывается для ASSIGN_IP и CONNECTED).

Сообщения о состояниях в реальном времени будут иметь префикс >STATE:.

7.7.7.12 Команда status

Показывает текущую информацию о статусе демона, в том же формате, какой используется директивой «OpenVPN-ГОСТ» –status.

Примеры команды:
status — показать информацию о статусе, используя умолчательную версию формата статуса.
status 2 — показать информацию о статусе, используя версию формата статуса 2.

7.7.7.13 Команда username

См. раздел 7.7.7.9.

7.7.7.14 Команда verb

Изменяет параметр «OpenVPN-ГОСТ» –verb. Параметр verb контролирует подробность вывода и может варьироваться от 0 (нет вывода) до 15 (максимальный вывод).

Примеры команды:
verb 4 — изменяет параметр verb в 4
verb — показать текущую установку параметра verb

7.7.7.15 Команда version

Показывает текущие версии «OpenVPN-ГОСТ» и интерфейса управления.

7.7.7.16 Команда auth-retry

Устанавливает параметр auth-retry, контролирующий, как «OpenVPN-ГОСТ» отвечает на аутентификационные ошибки логина/пароля.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

Примеры команды:

auth-retry interact — не завершать работу, если введены некорректные логин и пароль. Запросить новый ввод и повторить попытку.

7.7.7.17 Формат сообщений в реальном времени

Интерфейс управления «OpenVPN-ГОСТ» производит два вида вывода:

1. ответ на команду
2. асинхронный вывод в реальном времени, который может быть сгенерирован в любое время.

Сообщения в реальном времени начинаются с символа > в первой колонке, за которым немедленно следует ключевое слово, указывающее тип сообщения. Сейчас определены следующие типы:

ECHO — сообщения echo, подобные контролируемым командой echo

FATAL — фатальная ошибка, сообщение выводится в журнал сразу перед завершением работы «OpenVPN-ГОСТ».

HOLD — используется для указания, что «OpenVPN-ГОСТ» находится в замершем состоянии и не начнет работать, пока не получит команду hold release.

INFO — информационные сообщения, такие, как приветственное сообщение.

LOG — вывод журнального сообщения, подобный контролируемым командой log

PASSWORD — используется, чтобы сообщить клиенту управления, что «OpenVPN-ГОСТ» нуждается в пароле, а также чтобы сообщить о неудаче проверки пароля.

STATE — показывает текущее состояние «OpenVPN-ГОСТ», подобное контролируемому командой state.

7.7.7.18 Разбор команд

«OpenVPN-ГОСТ» использует тот же лексический анализатор командных строк, что используется парсером конфигурационного файла «OpenVPN-ГОСТ».

Параметры разделяются пробелом.

Двойные кавычки `("")` можно использовать, чтобы ограничить параметры, содержащие пробел

Используются escape-последовательности, основанные на обратном слэше, со следующей символикой:

`\\` — означает один символ обратного слэша (\) `\"` — передает буквальный символ двойных кавычек ("), не интерпретируя его как ограничение параметра `\[SPACE]` — передает буквальный символ пробела или табуляции, не интерпретируя его как разделитель параметров.

7.7.8 Управление процессом «OpenVPN-ГОСТ» с помощью интерфейса управления

Интерфейс управления «OpenVPN-ГОСТ» дает большие возможности управления процессом «OpenVPN-ГОСТ». Вы можете использовать интерфейс управления непосредственно, подключаясь через telnet к порту интерфейса управления, или косвенно, используя графический интерфейс «OpenVPN-ГОСТ», который сам подключается к интерфейсу управления.

Чтобы включить интерфейс управления на сервере или клиенте «OpenVPN-ГОСТ», добавьте к конфигурационному файлу следующую строку:

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

```
management localhost 7505
```

Эта строка прикажет «OpenVPN-ГОСТ» слушать на TCP-порту 7505 клиенты интерфейса управления (порт 7505 — произвольный выбор, вы можете выбрать любой свободный порт)

Когда «OpenVPN-ГОСТ» запущен, вы можете подключиться к интерфейсу управления, воспользовавшись клиентом telnet. Например:

```
ai:~ # telnet localhost 7505
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
>INFO:OpenVPN Management Interface Version 1 -- type 'help' for more info
help
Management Interface for OpenVPN 2.4.11 i686-pc-linux-gnu [SSL (OpenSSL)]
[EPOLL] [MH] [IPv6] built on Jan 16 2017
Commands:
auth-retry t           : Auth failure retry mode (none,interact,nointeract).
bytecount n            : Show bytes in/out, update every n secs (0=off).
echo [on|off] [N|all]  : Like log, but only show messages in echo buffer.
exit|quit              : Close management session.
forget-passwords       : Forget passwords entered so far.
help                   : Print this message.
hold [on|off|release]  : Set/show hold flag to on/off state, or
                         release current hold and start tunnel.
kill cn                : Kill the client instance(s) having common name cn.
kill IP:port           : Kill the client instance connecting from IP:port.
load-stats             : Show global server load stats.
log [on|off] [N|all]   : Turn on/off realtime log display
                         + show last N lines or 'all' for entire history.
mute [n]               : Set log mute level to n, or show level
                         if n is absent.
needok type action     : Enter confirmation for NEED-OK request of 'type',
                         where action = 'ok' or 'cancel'.
needstr type action    : Enter confirmation for NEED-STR request of 'type',
                         where action is reply string.
net                    : (Windows only) Show network info and routing table.
password type p        : Enter password p for a queried OpenVPN password.
remote type [host port] : Override remote directive, type=ACCEPT|MOD|SKIP.
proxy type [host port flags] : Enter dynamic proxy server info.
pid                    : Show process ID of the current OpenVPN process.
client-auth CID KID    : Authenticate client-id/key-id CID/KID (MULTILINE)
client-auth-nt CID KID : Authenticate client-id/key-id CID/KID
client-deny CID KID R [CR] : Deny auth client-id/key-id CID/KID with
                             log reason text R and optional client
                             reason text CR
client-kill CID [M]    : Kill client instance CID with message M
                         (def=RESTART)
env-filter [level]     : Set env-var filter level
```

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

```
client-pf CID           : Define packet filter for client CID (MULTILINE)
rsa-sig                 : Enter an RSA signature in response to >RSA_SIGN
                          challenge
                          Enter signature base64 on subsequent lines followed
signal s                : Send signal s to daemon,
                          s = SIGHUP|SIGTERM|SIGUSR1|SIGUSR2.
state [on|off] [N|all] : Like log, but show state history.
status [n]              : Show current daemon status info using format #n.
test n                  : Produce n lines of output for testing/debugging.
username type u         : Enter username u for a queried OpenVPN username.
verb [n]                : Set log verbosity level to n,
                          or show if n is absent.
version                 : Show current version number.
END
exit
Connection closed by foreign host.
ai:~ #
```

7.8 Усиление безопасности «OpenVPN-ГОСТ»

Одна из часто повторяемых максим сетевой безопасности состоит в том, что никогда не следует целиком полагаться на один компонент защиты, потому что его отказ вызывает катастрофический провал в безопасности. «OpenVPN-ГОСТ» предоставляет несколько механизмов, предоставляющих возможность добавить дополнительные слои защиты, чтобы отгородиться от подобного исхода.

7.8.1 tls-auth

Директива tls-auth добавляет дополнительную подпись HMAC ко всем пакетам хендшейка SSL/TLS для проверки целостности. Эта подпись HMAC предоставляет дополнительный слой защиты над и за тем, что предоставляется SSL/TLS. Она может защитить от:

- DoS-атак
- Уязвимостей переполнения буфера в реализации SSL/TLS
- Инициаций хэндшейка SSL/TLS с неавторизованных машин (хотя такие хэндшейки в конце концов не будут аутентифицированы, tls-auth может их отсечь намного раньше)

Использование tls-auth требует, чтобы вы сгенерировали закрытый ключ общего пользования, который используется дополнительно к сертификатам/ключам ГОСТ:

```
openvpn --genkey --secret ta.key
```

Эта команда сгенерирует статический ключ «OpenVPN-ГОСТ» и запишет его в файл ta.key. Этот ключ следует скопировать через уже существующий безопасный канал на сервер и все клиентские машины. Он может быть помещен в тот же каталог, что и файлы ГОСТ .key и .crt.

В серверной конфигурации добавьте:

```
tls-auth ta.key 0
```

В клиентской конфигурации добавьте:

```
tls-auth ta.key 1
```

Внимание! Алгоритмы magma-mac, kuznyechik-mac и gost-mac не могут быть использованы для выработки имитовставки пакетов контрольного канала, поэтому если вы хотите использовать опцию --tls-auth, в параметре --auth необходимо указать md_gost12_256 или md_gost12_512.

7.8.2 user/group (кроме ОС Windows)

«OpenVPN-ГОСТ» была очень тщательно создана так, чтобы позволить отказаться от прав привилегированного пользователя после инициализации, и эту возможность следует всегда использовать в Linux/BSD/Solaris. Без прав привилегированного пользователя, работающий серверный демон «OpenVPN-ГОСТ» представляет собой значительно менее привлекательную цель для атакующего.

Чтобы отключить права привилегированного пользователя после инициализации, добавьте в серверную конфигурацию директивы:

```
user nobody
group nobody
```

7.8.3 chroot (кроме ОС Windows)

Директива chroot позволяет вам запереть демон «OpenVPN-ГОСТ» в так называемую тюрьму chroot, где демон не сможет работать ни с какими частями файловой системы в операционной системе, за исключением специального каталога, указанного как параметр к директиве. Например

```
chroot jail
```

заставит демон «OpenVPN-ГОСТ» перейти в подкаталог jail при инициализации, а затем переориентирует его корневую файловую систему в этот каталог, так что затем демону будет невозможно увидеть никаких файлов снаружи каталога jail и его подкаталогов. Это важно с точки зрения безопасности, потому что даже если атакующий сможет скомпрометировать сервер с помощью вредоносного кода, этот код будет заперт от большей части файловой системы сервера.

Предупреждения: поскольку chroot переориентирует файловую систему (только с точки зрения демона), необходимо поместить все файлы, которые могут потребоваться «OpenVPN-ГОСТ» после инициализации, в каталог jail, например, файл crl-verify или каталог client-config-dir.

7.8.4 Хранение корневого ключа (ca.key) на отдельной машине без сетевого соединения

Одно из преимуществ использования PKI по стандарту X.509 с точки зрения безопасности состоит в том, что корневой ключ удостоверяющего центра (ca.key) не должен присутствовать на серверной машине «OpenVPN-ГОСТ». В среде высокой безопасности вы можете захотеть специально выделить машину для целей подписания ключей, держать эту машину физически хорошо защищенной и отключить ее от всех сетей. Для переноса ключей можно использовать съемные диски. Такие меры делают кражу корневого ключа исключительно трудной для атакующего, если только он не украдет физически машину для подписывания ключей.

8 Приложение. Список опций команды openvpn-gost

8.1 Общие опции

«OpenVPN-ГОСТ» позволяет помещать любую опцию в командную строку или в конфигурационный файл. Хотя все опции командной строки имеют префикс в виде двойного дефиса, этот префикс может быть опущен, когда опцию помещают в конфигурационный файл.

--help	Показать опции
--config file	Загрузить дополнительные конфигурационные опции из файла file, где каждая строка соответствует одной опции командной строки, но с опущенным двойным дефисом. Если --config file — единственная опция в команде openvpn-gost, --config можно опустить и записать команду как openvpn-gost file Заметьте, что конфигурационные файлы могут быть вложенными до осмысленной глубины. Можно использовать двойные или одинарные кавычки для ограничения параметров, включающих пробел, # или ; символы в первой колонке можно использовать для обозначения комментариев. Заметьте, что «OpenVPN-ГОСТ» обрабатывает escape-последовательности на основе обратной косой черты для символов, не заключенных в одинарные кавычки, так что следует иметь в виду следующие обозначения: \\ Обозначает одну обратную косую черту (\) \" Передает буквально двойные кавычки, не интерпретируя их как ограничение параметра \[SPACE] передает буквально пробел или табуляцию, не интерпретируя их как разделение параметров Например, в Windows используйте двойные обратные косые черты для представления путей к файлам: `secret "c:\\cryptopack4\\config\\secret.key"`

8.2 Туннельные опции

--mode m	Устанавливает режим работы «OpenVPN-ГОСТ». По умолчанию «OpenVPN-ГОСТ» работает в режиме точка-в-точку (p2p). В «OpenVPN-ГОСТ» присутствует режим «сервер», реализующий многоклиентную способность сервера.
--local host	Имя или IP-адрес локального хоста для связывания. Если указан, «OpenVPN-ГОСТ» будет связываться только с этим адресом. Если не указан, «OpenVPN-ГОСТ» будет связываться со всеми интерфейсами.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--remote host [port] [proto]	Имя или IP-адрес удаленного хоста. IP-адрес может быть как в формате IPv4, так и в формате IPv6. На клиенте задаёт адрес сервера и порт, к которому требуется подключиться. Если значение параметра -- DNS-имя, которое разрешается в несколько IP-адресов, один из них будет выбран случайно, предоставляя своего рода возможность уравновешивания нагрузки. Для надежности могут быть указаны несколько опций --remote, каждая относится к отдельному серверу «OpenVPN-ГОСТ». Клиент будет пытаться соединиться с серверами в порядке, указанном в списке опций --remote. Клиент перейдет к следующему хосту в списке, если соединение с предыдущим установить не удастся. Обратите внимание, что в любой момент времени клиент «OpenVPN-ГОСТ» может быть соединен только с одним сервером. Указание нескольких опций --remote -– отдельный случай более общей функциональности профиля соединения. См. ниже описание <connection>. [proto] обозначает протокол, который будет использован при соединении с сервером. Может иметь значение “tcp” или “udp ”. Для принудительного подключения по IPv4 или IPv6 используйте суффикс 4/6 в конце “tcp”/“udp”. К примеру, udp4/udp6/tcp4/tcp6. Клиент перейдет к следующему в списке хосту по уведомлению об ошибке соединения. Обратите внимание, что в любой момент времени, «OpenVPN-ГОСТ» клиент будет подключен не более чем к одному серверу. Учтите, что поскольку UPD не требует установления соединения, ошибка подключения определяется с помощью --ping и --ping-restart опций. Если хост – DNS-имя, которое разрешает несколько IP-адресов, «OpenVPN-ГОСТ» будет пробовать подключиться к ним в порядке, в котором их предоставит система getaddrinfo(). Поэтому приоритизация и рандомизация DNS выполняется с помощью системной библиотеки. Если IP-версия не указана в спецификации протокола (суффикс 4/6), «OpenVPN-ГОСТ» переберет оба IPv4 и IPv6 адреса в том порядке, в какой getaddrinfo() вернет их. Обратите внимание на следующий предельный случай. Если вы используете опции --remote и отказываетесь от прав привилегированного пользователя с помощью --user и/или --group, и клиент работает не в Windows, если клиенту нужно переподключиться к другому серверу, и этот сервер передает различные настройки TUN/TAP или маршрутизации, клиенту может не хватить необходимых прав, чтобы закрыть и снова открыть интерфейс TUN/TAP. Это может привести к прекращению работы клиента с фатальной ошибкой.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

	Когда используется TCP режим, на сервере --remote будет работать как фильтр, отвергая соединения со всех хостов, не соответствующих значению параметра. Если опция --remote не указана, сервер будет слушать пакеты с любого IP-адреса, но не будет обрабатывать эти пакеты, если они не пройдут все аутентификационные тесты. Это требование к аутентификации относится ко всем потенциальным партнерам, даже приходящим с известных и предположительно доверенных IP-адресов (очень легко подделать исходный IP-адрес в UDP пакете).
--remote-random-hostname	Добавляет случайную строку (6 байт, 12 шестнадцатиричных символов) к имени хоста, чтобы предотвратить DNS кэширование. К примеру, "foo.bar.gov"будет преобразовано в «random-chars >.foo.bar.gov ".

<connection>	Определяет профиль клиентского соединения. Профили клиентского соединения — группы опций «OpenVPN-ГОСТ», описывающие, как связаться с конкретным сервером «OpenVPN-ГОСТ». Профили клиентского соединения указываются в конфигурационном файле «OpenVPN-ГОСТ», и каждый профиль начинается с <connection> и заканчивается </connection>.
	Клиент «OpenVPN-ГОСТ» будет пробовать каждый профиль соединения последовательно, пока не достигнет успешного соединения.
	Опция --remote-random может использоваться для того, чтобы с самого начала «смешать» список соединений.
	Вот пример использования профилей соединений:
	`client`
	`dev tun`
	
	`<connection>`
	`remote 198.19.34.56 1194 udp`
	`</connection>`
	
	`<connection>`
	`remote 198.19.34.56 443 tcp`
	`</connection>`
	
	`<connection>`
	`remote 198.19.34.56 443 tcp`
	`http-proxy 192.168.0.8 8080`
	`</connection>`
	
	`<connection>`
	`remote 198.19.36.99 443 tcp`
	`http-proxy 192.168.0.8 8080`
	`</connection>`
	
	`persist-key`
	`persist-tun`
	`pkcs12 client.p12`
	`remote-cert-type server`
	`verb 3`
	Сначала мы пытаемся соединиться с сервером на 198.19.34.56:1194, используя UDP. Если это не удается, мы пытаемся соединиться с 198.19.34.56:443, используя TCP, и далее по порядку.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

	Внутри блока <connection> могут быть следующие опции «OpenVPN-ГОСТ»: bind, connect-retry, connect-retry-max, connect-timeout, explicit-exit-notify, float, http-proxy, http-proxy-option, link-mtu, local, lport, mssfix, mtu-disc, nobind, port, proto, remote, rport, socks-proxy, tun-mtu and tun-mtu-extra. Существует механизм умолчания для указания опций, применимых ко всем профилям <connection>. Если любая из вышеуказанных опций (за исключением remote) появляется вне блока <connection>, но в конфигурационном файле, содержащем хотя бы один блок <connection>, эта опция будет использована по умолчанию для всех блоков <connection>, которые следуют за ней в конфигурационном файле. Например, предположим, что опция nobind помещена в начало приведенного выше конфигурационного файла, перед первым блоком <connection>. Эффект будет такой же, как если бы nobind была объявлена во всех блоках <connection> под ней.
--proto-force p	При переборе профилей подключения учитываются только профили, использующие протокол p ('tcp'|'udp ').
--remote-random	Когда указаны несколько адресов/портов remote, или если используются профили соединения, данная опция изначально рандомизирует порядок списка в качестве своего рода простейшей меры равномерного распределения нагрузки.

--proto p	Использовать протокол p для связи с удаленным хостом. p может быть udp, tpc, tcp-client или tcp-server. В качестве протокола по умолчанию используется udp, когда --proto не задан. Чтобы соединение было успешным, соединающиеся узлы должны использовать совместимые протоколы, то есть нельзя указывать на одном конце --proto udp, а на другом --proto tcp. Стабильная работа по протоколу UDP гарантируется только при использовании TLS версии 1.3 и алгоритмов шифрования magma-mgm и kuznyechik-mgm. Значения tcp-server и tcp-clent позволяют указать, что узел будет узлом, принимающим соединения, или узлом, устанавливающим соединения, соответственно. Партнер, начавший с tcp-server, будет бесконечно ждать входящего соединения. Партнер, начавший с tcp-client, попытается подключиться, и если не удастся, будет спать 5 секунд (это время можно изменить с помощью опции --connect-retry) и будет опять пытаться подключиться бесконечное количество раз или до N повторений (количество повторений можно изменить с помощью опции --connect-retry-max). И клиент, и сервер будут симулировать сигнал рестарта SIGUSR1, если какая-либо из сторон будет пересоединяться.
--connect-retry n [max]	Для --proto tcp-client, ждать n секунд между попытками переприсоединиться (по умолчанию 5). Повторяющиеся попытки переприсоединения замедляются после 5 попыток на каждый удаленный узел, удваивая время ожидания после каждой неудачной попытки. Опциональный аргумент max определяет максимальное значение времени ожидания в секундах (по умолчанию = 300).
--connect-retry-max n	Для --proto tcp-client, пытаться переподключиться n раз (бесконечное количество раз по умолчанию). n определяет количество попыток вызовов --remote или <connection> (бесконечное количество раз по умолчанию). Если установить n = 1, то это позволит попробовать выполнить каждый вызов ровно один раз. Успешное соединение обнуляет счетчик.
--show-proxy-settings	Показать обнаруженные настройки HTTP или SOCKS-прокси. На данный момент только клиенты Windows поддерживают эту опцию.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--http-proxy server port [authfile|'auto'| 'auto-nct'][auth-method]	Соединиться с удаленным хостом через HTTP-прокси с адресом server и портом port. Если требуется HTTP Proxy-Autenticate, authfile – файл, содержащий логин и пароль на 2 строках, или «stdin» для запроса с консоли. Его содержимое можно так же указать в конфигурационном файле с помощью опции --http-proxy-user-pass. auth-method должен быть «none», «basic» или «ntlm». HTTP-дайджест-аутентификация также поддерживается, но только через флаги auto или auto-nct (см. ниже). Флаг auto заставляет «OpenVPN-ГОСТ» автоматически определять auth-method и запрашивать со стандартного входа или интерфейса управления логин и пароль, если требуются. Флаг auto-nct (no clear-text auth) велит «OpenVPN-ГОСТ» автоматически определять метод аутентификации, но отвергать слабые аутентификационные протоколы, такие как HTTP-аутентификация Basic.
--http-proxy-option type[parm]	Установить расширенные опции HTTP-прокси. Можно указывать этот параметр несклько раз для, чтобы установить несколько опций. Поддерживаемые опции: VERSION version – установить номер версии HTTP в version (по умолчанию 1.0); AGENT user-agent – установить HTTP-строку "User-Agent"в user-agent; CUSTOM-HEADER name content – добавить настраиваемый заголовок с именем name и с content в качестве содержимого настраиваемого HTTP-заголовка.
--socks-proxy server [port] [authfile]	Подключиться к удаленному хосту через Socks-прокси с адресом server и портом port (по умолчанию 1080). authfile (опциональный) – это файл, содержащий имя пользователя и пароль в две строчки, либо “stdin” для запроса с консоли.
--resolv-retry n	Если разрешение имени хоста для --remote оказывается ошибочным, попытаться разрешить в течение n секунд перед ошибкой. Установить n в infinite, чтобы пытаться разрешить бесконечно. По умолчанию установлена --resolve-retry infinite. Можно отключить эту опцию, установив n=0.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--float	Позволить удаленному партнеру менять свой IP-адрес и/или номер порта, например, из-за DHCP (это умолчание, если --remote не используется.) --float, указанный вместе с --remote, позволяет сеансу «OpenVPN-ГОСТ» изначально подключаться к партнеру с известным адресом, однако если пакеты приходят с нового адреса и проходят аутентификационные тесты, новый адрес примет управление сеансом. Это полезно, когда вы соединяетесь с партнером с динамическим адресом, например, пользователем с телефона или клиентом DHCP. В сущности, --float велит «OpenVPN-ГОСТ» принимать аутентифицированные пакеты с любого адреса, а не только с того, который указан в опции --remote.
--ipchange cmd	Выполнить команду оболочки cmd, когда наш удаленный ip-адрес с самого начала аутентифицирован или изменяется. cmd содержит путь к скрипту (или исполняемой программе), опционально используется с аргументами. Путь и аргументы могут быть заключены в одинарные или двойные кавычки и/или отделены обратной косой чертой, а также должны быть разделены пробелами. Выполнять как: cmd ip_address port_number Не используйте -ipchange в режиме --mode server. Вместо этого воспользуйтесь скриптом --client-connect. См. в разделе «Переменные среды» ниже дополнительные параметры, передаваемые как переменные среды. Обратите внимание, что cmd может быть командой оболочки с несколькими аргументами, в этом случае все аргументы, генерированные «OpenVPN-ГОСТ», будут добавлены в конец строки cmd, чтобы создать командную строку, которая будет передана в скрипт. Если вы работаете в среде с переменным IP-адресом, где IP-адреса каждого партнера могут меняться без предупреждения, вы можете использовать этот скрипт, например, чтобы редактировать файл /etc/hosts с текущим адресом партнера. Скрипт будет выполняться каждый раз, когда удаленный партнер меняет свой IP-адрес. Подобным же образом, если *наш* IP-адрес меняется из-за DHCP, нам следует сконфигурировать наш скрипт перемены IP-адреса так, чтобы «OpenVPN-ГОСТ» получала сигнал SIGHUP или SIGUSR1. «OpenVPN-ГОСТ» затем переустановит соединение со своим последним аутентифицированным партнером на его новом IP-адресе.

--port port	Номер TCP/UDP-порта для локального и удаленного хостов (устанавливает обе опции --lport и --rport для заданного порта). Текущий номер по умолчанию 1194 представляет официальное назначение порта IANA для OpenVPN.
--lport port	Номер TCP/UDP-порта для связывания. Не может быть использован вместе с --nobind опцией.
-rport port	Номер TCP/UDP-порта для удаленного хоста. Порт может быть также установлен непосредственно с помощью --remote опции.
--bind [ipv6only]	Связаться с локальным адресом и портом. Это умолчание, если не используются какие-либо из опций -proto tcp-client, --http-proxy или --socks-proxy.
--nobind	Не связываться с локальным адресом и портом. IP-стек назначит динамический порт для возвращающихся пакетов. Поскольку значение динамического порта партнеру неизвестно заранее, эта опция подходит только для партнеров, которые будут инициировать соединение, используя опцию --remote.
--dev tunX|tapX|null	Виртуальное сетевое устройство TUN/TAP (для динамического устройства X можно опустить.) Вы должны использовать на обоих концах соединения либо устройства tun, либо tap. Вы не можете указывать на одном конце tap, а на другом tun, поскольку они представляют различные слои сети. Устройства tun инкапсулируют IPv4 или IPv6 (слой 3 OSI), а устройства tap инкапсулируют Ethernet 802.3 (слой 2 OSI).
--dev-type device-type	Какой тип устройства мы используем? device-type должен иметь значение tun (слой 3 OSI) или tap (слой 2 OSI). Используйте эту опцию только в том случае, если устройство TUN/TAP, использованное в опции --dev, не начинается с tun или tap.

--topology mode	Сконфигурировать виртуальную адресную топологию, работая в режиме --dev tun. Эта директива не имеет значения в режиме --dev tap, который всегда использует топологию subnet. Если вы настраиваете эту директиву на сервере, директивы --server и --server-bridge автоматически передадут выбранные вами настройки топологии клиентам. Эту директиву также можно передать клиентам вручную. Как и директива --dev, эта директива должна всегда быть совместимой между клиентом и сервером. mode может иметь значение: net30 - использовать топологию точка-в-точку, назначая одну подсеть /30 каждому клиенту. Это предназначено для того, чтобы разрешить семантику точка-в-точку, когда несколько или все подключающиеся клиенты могут быть ОС Windows. Это умолчание. p2p - использовать топологию точка-в-точку, где удаленная конечная точка интерфейса tun клиента всегда указывает на локальную конечную точку интерфейса tun сервера. Этот режим назначает один IP-адрес для каждого соединяющегося клиента. Используйте только тогда, когда ни один из подключающихся клиентов не является ОС Windows. subnet - используйте подсеть вместо топологии точка-в-точку, сконфигурировав интерфейс tun с локальным IP-адресом и маской подсети, подобным топологии, используемой в --dev tap и режиме «мост» Ethernet. Этот режим назначает один IP-адрес каждому соединяющемуся клиенту и работает в Windows и в Unix-подобных системах. При использовании в Windows требует версию драйвера TAP-WIN32 8.2 или выше. При использовании в Unix-подобных системах, требует, чтобы драйвер tun поддерживал команду ifconfig, которая устанавливает подсеть вместо удаленного конечного IP-адреса. Примечание: Использование --topology subnet меняет интерпретацию аргументов опции --ifconfig, означая “address netmask”, а не “local remote”.

--tun-ipv6	Построить tun-связь, способную передавать трафик по протоколу IPv6. Следует использовать совместно с --dev tun или --dev tunX. Если никакой специальной поддержки IPv6 в TUN для вашей операционной системы не скомпилировано в «OpenVPN-ГОСТ», будет выводиться предупреждение. Явно устанавливает узел устройства вместо использования /dev/net/tun, /dev/tun, /dev/tap и т.д. Если «OpenVPN-ГОСТ» не может определить является ли устройство TUN или TAP на основании имени, то нужно также задать --dev-type tun или --dev-type tap. В операционной системе Mac OS X эта опция может быть использована, чтоб задать значение по умолчанию для tun реализации. Использование --dev-node utun вынуждает использовать исходную поддержку ядра Darwin tun. Используйте --dev-node utunN, чтоб выбрать конкретный utun. Чтоб использовать tun.kext (/dev/tunX) используйте --dev-node tun. Если не задать опцию --dev-node, «OpenVPN-ГОСТ» попробует открыть utun в первую очередь, а если не получится, то вернется к tun.kext. В Windows выберете TAP-Win32 адаптер с именем узла в Панели Управления Сетевыми Подключениями или исходный GUID адаптера, заключенный в скобки. Опция --show-adapters в Windows также может быть использована для перечисления всех доступных TAP-Win32 адаптеров и для отображения и имени панели управления сетевыми подключениями, и GUID для каждого из TAP-Win32 адаптеров.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

-dev-node node	Явным образом установить узел устройства, а не использовать /dev/net/tun, /dev/tun, /dev/tap и т.д. Если «OpenVPN-ГОСТ» не может определить, является ли node устройством TUN или TAP, на основании этого имени, вы должны также указать --dev type tun или --dev type tap. В операционной системе Mac OS X эта опция может быть использована, чтоб задать значение по умолчанию для tun реализации. Использование --dev-node utun предписывает использовать исходную поддержку ядра Darwin tun. Используйте --dev-node utunN, чтобы выбрать конкретный utun. Чтоб использовать tun.kext (/dev/tunX), используйте --dev-node tun. Если не задать опцию --dev-node, «OpenVPN-ГОСТ» сначала попытается использовать utun, затем tun.kext. В системах Windows выберите адаптер TAP-Win32, который называется node, в Network Connections Control Panel или GUID адаптера в скобках. В Windows можно также использовать опцию --show-adapters, чтобы пронумеровать все доступные адаптеры TAP-Win32, эта опция покажет название контрольной панели сетевых соединений и GUID для каждого адаптера TAP-Win32.
--lladdr address	Указать адрес слоя связи, более известный как MAC-адрес. Применяется только к устройствам TAP.
iproute cmd	Установить альтернативную команду для выполнения вместо умолчательной команды iproute2. Может быть использована, чтобы выполнять «OpenVPN-ГОСТ» в непривилегированной среде.

-ifconfig l rn	Установить параметры адаптера TUN/TAP. l – IP-адрес локальной конечной точки VPN. Для устройств TUN в режиме точка-точка, rn – IP-адрес удаленной конечной точки VPN. Для устройств TAP или TUN устройств, используемых с --topology subnet, rn – маска подсети виртуального сегмента, который создается или с которым связываются. Для устройств TUN, которые упрощают виртуальные IP-соединения точка-в-точку (когда используется в --topology net30 или p2p режиме), правильное использование --ifconfig – использовать два частных IP-адреса, которые не являются членами какой-либо используемой существующей подсети. IP-адреса могут идти подряд, и их порядок у удаленного партнера должен быть обратным. После того, как VPN установлена, вы будете отправлять сигнал ping на rn через VPN. Для устройств TAP, которые предоставляют возможность создавать виртуальные сегменты Ethernet, или TUN устройств в режиме --topology subnet (который создает виртуальную “многоточечную сеть”), --ifconfig используется, чтобы настроить IP-адрес и маску подсети так же, как был бы подобным образом сконфигурирован физический адаптер Ethernet. Если вы пытаетесь соединиться с удаленным Ethernet-«мостом», IP-адрес и подсеть следует установить в величины, которые были бы действительны в сегменте Ethernet, с которым вы соединяетесь (обратите также внимание, что для этой же цели можно использовать DHCP). Эта опция, изначально прокси для команды ifconfig, предназначена, чтобы упростить туннельную конфигурацию TUN/TAP, предоставляя стандартный интерфейс различным реализациям ifconfig на различных платформах. Параметры --ifconfig, которые являются IP-адресами, также могут быть указаны как имя, разрешимое в DNS или в файл /etc/hosts. Для устройств TAP, --ifconfig не следует использовать, если интерфейс TAP будет получать аренду IP-адреса с сервера DHCP.
--ifconfig-noexec	Не выполнять команды ifconfig/netsh, вместо этого передавать параметры -ifconfig в скрипты, используя переменные среды.

--ifconfig-nowarn	Не выводить предупреждение о проверке корректности опций, если опция --ifconfig на этой стороне соединения не соответствует удаленной стороне. Это полезно, когда вы хотите сохранить общие преимущества проверки корректности опций (см. также опцию --disable-occ), отключая только компонент проверки ifconfig. Например, если у вас есть конфигурация, где локальный хост использует -ifconfig, но удаленный хост ее не использует, примените --ifconfig-nowarn на локальном хосте. Эта опция также отключит предупреждения о потенциальных конфликтах адресов, которые иногда раздражают более опытных пользователей включением «ложных положительных» предупреждений.
--route network/IP [netmask] [gateway] [metric]	Добавить маршрут в таблицу маршрутов после того, как установлено соединение. Можно указать несколько маршрутов. Маршруты будут автоматически отключены в обратном порядке перед отключением устройства TUN/TAP. Эта опция предназначена как прокси для удобства пользования командой оболочки route, в то же время предоставляя семантику, переносимую через пространство платформы «OpenVPN-ГОСТ». netmask по умолчанию – 255.255.255.255 gateway по умолчанию – берется из --route-gateway или второго параметра --ifconfig, когда указан --dev tun. metric по умолчанию – берется из --route-metric, в остальных случаях 0. Умолчание можно указать, оставив опцию пустой или указав значение «default». Параметры network и gateway могут также быть указаны как имя, разрешаемое в DNS или файл /etc/hosts, или как одно из трех специальных ключевых слов. vpn_gateway – адрес удаленной конечной точки VPN (берется или из --route-gateway или из второго параметра --ifconfig, если указана опция --dev tun) net_gateway – ранее существовавший гейт с умолчательным IP, читается из таблицы маршрутизации (поддерживается не во всех ОС) remote_host – адрес --remote, если «OpenVPN-ГОСТ» работает в клиентском режиме и не определена в серверном режиме.
--route-gateway gw|'dhcp'	Указать умолчательный гейт gw для использования с --route. Если в качестве параметра указано dhcp, адрес гейта будет взят из взаимодействия DHCP с серверной локальной сетью «OpenVPN-ГОСТ».
--route-metric m	Указать умолчательную метрику m для использования с --route.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--route-delay [n] [w]	Подождать n секунд (по умолчанию 0) после установки соединения, прежде чем добавлять маршруты. Если n равно 0, маршруты будут добавлены немедленно после установления соединения. Если оппция --route-delay опущена, маршруты будут добавлены сразу после открытия устройства TUN/TAP и выпонения скрипта --up, перед отключением любых прав привилегированного пользователя директивами --user или --group (или выполнения --chroot). Эта опция полезна в сценариях, где используется DHCP для установки адресов адаптера tap. Задержка даст хендшейку DHCP время завершиться, прежде чем будут добавлены маршруты. В Windows опция --route-delay пытается быть более интеллектуальной, ожидая w секунд (по умолчанию w=30) включения адаптера TAP-Win32 перед добавлением маршрутов.
--route-up cmd	Выполнить команду оболочки cmd после того, как маршруты добавлены в соответствии с опцией --route-delay. cmd состоит из пути до скрипта (или исполняемой программы) и опционально используется с аргументами. Путь и аргументы могут быть заключены в одинарные или двойные кавычки и/или отделены обратной косой чертой, а также обязательно разделены одним или более пробелом. См. в разделе «Переменные среды» ниже дополнительные параметры, которые передаются как переменные среды.
--route-pre-down cmd	Выполнить команду cmd до того, как маршруты будут удалены в процессе отключения. cmd содержит путь к скрипту (или исполняемой программе), опционально используется с аргументами. Путь и аргументы могут быть заключены в одинарные или двойные кавычки и/или отделены обратной косой чертой, а также должны быть разделены пробелами. См. в разделе «Переменные среды» ниже дополнительные параметры, которые передаются как переменные среды.
--route-noexec	Не добавлять и не удалять маршрутов автоматически. Вместо этого передавать маршруты в скрипт --route-up с помощью переменных среды.

--route-nopull	При использовании с --client или --pull, принять опции, переданные сервером, КРОМЕ маршрутов. Блокирует внешние dns и dhcp опции, такие как DNS сервера. При использовании на клиенте эта опция эффективно запрещает серверу добавлять маршруты в таблицу маршрутизации клиента, однако обратите внимание, что эта опция все же позвояет серверу устанавливать свойства TCP/IP на клиентском интерфейсе TUN/TAP.
--allow-pull-fqdn	Позволяет клиенту брать DNS-имена с сервера (вместо того, чтобы ограничиться IP-адресом) для --ifconfig, --route и --route-gateway.
--client-nat snat|dnat network netmask alias	Эта передаваемая клиентская опция устанавливает правило NAT “один-к-одному” без сохранения состояния на пакетных адресах (не портах) и очень полезно, когда route или ifconfig, передаваемые клиенту, создают конфликт нумерации IP. network/netmask (к примеру, 192.168.0.0/255.255.0.0) определяет локальное представление ресурса с точки зрения клиента, в то время как alias/netmask (к примеру, 10.64.0.0/255.255.0.0) определяет удаленное преставление с точки зрения сервера. Используйте snat (исходный NAT) для ресурсов, принадлежащих клиенту и dnat (конечный NAT) для удаленных ресурсов. Установите --verb 6 для для отладочной информации, показывающей трансформацию src/dest адресов в пакетах.

--redirect-gateway flags...	(Экспериментальная) Автоматически выполняет команды маршрутизации, чтобы заставить весь исходящий трафик проходить через VPN. Эта опция выполняет три шага: 1) Создает статический маршрут для адреса --remote, который форвардится на ранее существовавший умолчательный гейт. Это делается для того, чтобы третий шаг не создал маршрутизационной петли. 2) Удаляет маршрут умолчательного гейта. 3) Устанавливает новый умолчательный гейт в качестве конечной точки VPN (берется либо из --route-gateway, либо из второго параметра --ifconfig, если указана опция --dev tun). Когда туннель разрывается, все вышеизложенные шаги выполняются в обратном порядке, чтобы восстановить исходный умолчательный маршрут. Флаги опции: local - добавьте флаг local, если оба сервера «OpenVPN-ГОСТ» прямо соединены через общую подсеть, например, беспроводную. Флаг local заставит опустить вышеописанный шаг 1. autolocal - пытается автоматически определить, следует ли включить локальный флаг выше. def1 - используйте этот флаг, чтобы переопределить умолчательный гейт, используя 0.0.0.0/1 и 128.0.0.0/1 вместо 0.0.0.0/0. Это имеет то преимущество, что исходный умолчательный гейт переопределяется, но не уничтожается. bypass-dhcp - добавляет прямой маршрут к серверу DHCP (если он не локальный), который обходит туннель (доступно на клиентах Windows, может быть недоступно на других клиентах) bypass-dns - добавляет прямой маршрут к серверу (серверам) DNS (если они не локальные), который обходит туннель (доступно на клиентах Windows, может быть недоступно на других клиентах). block-local - блокирует доступ к локальной сети, когда туннель активен, за исключением самого шлюза локальной сети. Это сделано путем маршрутизации локальной сети (кроме адреса шлюза локальной сети) в туннель. ipv6 – переадресует маршрутизацию IPv6 в туннель. Это работает схоже с def1 флагом, то есть добавляются более конкретные IPv6 маршруты (2000::/4, 3000::/4), покрывающие все пространство одноадресной передачи IPv6. !ipv – не переадресовывать IPv4 траффик – обычо используется в паре флагов ipv6 !ipv4, чтоб переадресовывать только IPv6.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--link-mtu n	Устанавливает верхнюю границу размера UDP пакетов, которые пересылаются между узлами «OpenVPN-ГОСТ». Желательно не устанавливать этот параметр, если вы не уверены в том, что вы делаете.
--redirect-private [flags]	Аналогично с --redirect-gateway, но без фактического изменения гейта по умолчанию. Полезно при передаче в частные подсети.
--tun-mtu n	Принять MTU устройства TUN за n и вычислить из него MTU связи (умолчание=1500). В большинстве случаев вы, вероятно, захотите оставить этот параметр установленным в умолчательное значение. MTU (Maximum Transmission Unit) – максимальный размер пакета в байтах, который может быть отправлен без фрагментирования по конкретному пути в сети. «OpenVPN-ГОСТ» требует, чтобы пакеты по контрольным каналам или каналам данных пересылались без фрагментирования. Проблемы с MTU часто проявляются как соединения, которые обрываются во время периодов активного использования. Лучше использовать опции --fragment и/или --mssfix, чтобы справиться с проблемами размеров MTU.
--tun-mtu-extra n	Предположить, что устройство TUN/TAP может вернуть пакет на n байтов больше, чем размер, указанный в --tun-mtu. Этот параметр по умолчанию установлен в 0, что достаточно для большинства устройств TUN. Устройства TAP могут ввести дополнительное увеличение размера MTU, поэтому когда используются устройства TAP, используется умолчательное значение 32. Этот параметр контролирует только размеры внутреннего буфера «OpenVPN-ГОСТ», так что никакого увеличения передачи, связанного с использованием большей величины, не происходит.
--mtu-disc type	Следует ли нам выполнять Path MTU discovery на TCP-канале? Поддерживается только в таких ОС как Linux, которые поддерживают необходимый системный вызов для настройки. 'no' - Никогда не посылать фреймы DF (Don't Fragment) 'maybe' - использовать инструкции, связанные с маршрутом 'yes' - Всегда DF (Don't Fragment)

--mtu-test	Чтобы эмпирически замерить MTU в начале соединения, добавьте опцию --mtu-test к вашей конфигурации. «OpenVPN-ГОСТ» отправит пинговые пакеты разных размеров удаленному партнеру и измерит самые большие пакеты, которые были успешно получены. Процесс --mtu-test обычно занимает около 3 минут.
--fragment max	Включает внутреннюю фрагментацию датаграмм, так чтобы не посылались UDP-датаграммы размером больше, чем max байтов. Параметр max интерпретируется так же, как параметр --link-mtu, т.е. был добавлен размер UDP-пакета после инкапсуляции, но без самого UDP-заголовка. Опция fragment имеет смысл только в том случае, если вы пользуетесь протоколом UDP (--proto udp). --fragment добавляет 4 байта к каждой датаграмме. См. в описании опции mssfix внизу описание важной опции, связанной с --fragment. Следует также иметь в виду, что эта опция не предназначена для того, чтобы заменить UDP-фрагментацию на уровне IP-стека. Она введена как последнее средство, когда path MTU discovery недоступно. Использование этой опции менее эффективно, чем восстановление открытия MTU-пути для вашей IP-связи и использования собственной IP-фрагментации. Но бывают обстоятельства, когда использование возможностей «OpenVPN-ГОСТ» по внутренней фрагментации могут быть вашей единственной возможностью, например, передача по туннелю мультикастового потока UDP, требующего фрагментации.

--mssfix max	Объявить сеансам TCP, работающим через туннель, что им следует ограничить размеры отправляемых пакетов так, чтобы после того, как «OpenVPN-ГОСТ» их инкапсулировала, размер результирующих UDP-пакетов, которые «OpenVPN-ГОСТ» посылает своему партнеру, не будет превосходить max байтов. Значение по умолчанию – 1450. Параметр max интерпретируется так же, как параметр --link-mtu, т.е. был добавлен размер UDP-пакета после инкапсуляции, но без самого UDP-заголовка. Результирующий пакет будет не более чем на 28 байт больше для IPv4 и на 48 байтов больше для IPv6 (20/40 байт для IP заголовка и 8 байт для UDP заголовка). Значение по умолчанию – 1450 – позволяет передавать IPv4 пакеты по каналу с MTU 1473 или выше без фрагментации на уровне IP. Опция -mssfix имеет смысл только тогда, когда вы используете протокол UPD, т.е. --proto udp. --mssfix и --fragment в идеале следует использовать вместе, где --mssfix будет прежде всего пытаться сделать так, чтобы TCP не нуждалось в фрагментации пакетов, и если большие пакеты все равно проходят (с протоколов, отличных от TCP), --fragment их фрагментирует внутри VPN. И --fragment, и --mssfix предназначены для того, чтобы обходить случаи, когда Path MTU discovery недоступно на сетевом пути между партнерами, общающимися через «OpenVPN-ГОСТ». Обычный симптом подобной проблемы – соединение «OpenVPN-ГОСТ», которое успешно начинается, но потом замирает во время активного использования. Если опции --fragment и --mssfix используются вместе, --mssfix берет умолчательное значение параметра max из опции --fragment max. Таким образом, можно уменьшить максимальный размер UDP-пакета до 1300 (хорошая начальная попытка для решения проблем соединения, связанных с MTU) со следующими опциями: --tun-mtu 1500 --fragment 1300 --mssfix
--sndbuf size	Устанавливает размер буфера отправления TCP/UDP-сокета. Значение по умолчанию устанавливается в соответствии со значением по умолчанию в ОС.
--rcvbuf size	Устанавливает размер буфера получения TCP/UDP-сокета. Значение по умолчанию устанавливается в соответствии со значением по умолчанию в ОС.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--mark value	Пометить отправляемые зашифрованные пакеты значением value. Метку value может соответствовать политике маршрутизации и правилам фильтрации пакетов. Эта опция поддерживается только в Linux и ничего не делает в других операционных системах.
--socket-flags flags...	Применить данные флаги к транспортному сокету «OpenVPN-ГОСТ». В настоящее время поддерживается только флаг TCP_NODELAY. Флаг TCP_NODELAY полезен в режиме TCP и заставляет ядро сразу же посылать туннельные пакеты через TCP-соединение, не пытаясь группировать несколько маленьких пакетов в один большой пакет. Это может привести к существенному уменьшению времени задержки. Эта опция может передаваться с сервера на клиент, и для максимального эффекта ее следует использовать и на сервере, и на клиенте.
--txqueuelen n	(Только для Linux) Установить длину очереди TX на интерфейсе TUN/TAP. В настоящее время по умолчанию равна 100.

--shaper n	Ограничивает ширину полосы пропускания исходящих туннельных данных в n байт в секунду на TCP/UDP порту. Учтите, что это сработает, только в режиме точка-точка. Если вы хотите ограничить ширину полосы пропускания в обе стороны, используйте эту опцию у обоих партнеров. «OpenVPN-ГОСТ» использует следующий алгоритм для реализации ограничения полосы пропускания трафика. При величине полосы пропускания в n байт в секунду, после того, как запись датаграммы в b байтов поставлена в очередь на TCP/UDP порту, подождать минимум (b/n) секунд, прежде чем ставить в очередь следующую запись. Следует заметить, что «OpenVPN-ГОСТ» поддерживает несколько туннелей между одними и теми же двумя партнерами, позволяя вам конструировать одновременно полноскоростные туннели и туннели с уменьшенной полосой пропускания, маршрутизируя низкоприоритетные данные по туннелям с уменьшенной полосой пропускания, а остальные данные по туннелю с полной скоростью. Заметьте также, что для туннелей с низкой полосой пропускания (меньше 1000 байтов в секунду) вам, вероятно, следует использовать более низкие значения MTU (см. выше), иначе время задержки пакетов так вырастет, что будет вызывать таймауты в слое TLS и в TCP-соединениях, работающих по туннелю. В «OpenVPN-ГОСТ» n разрешено находиться в пределах от 100 байт в секунду до 100 мегабайт в секунду.
--inactive n [bytes]	Заставляет «OpenVPN-ГОСТ» отключаться после n секунд неактивности устройства TUN/TAP. Время бездействия измеряется с последнего входящего туннельного пакета. Если включен опциональный параметр bytes, отключение после n секунд бездействия на устройстве tun/tap производит комбинированный подсчет байтов туда/обратно, который меньше, чем bytes. В любом случае, внутренние ping-пакеты «OpenVPN-ГОСТ» (которые просто являются проверкой активности) и управляющие пакеты TLS не считаются «активностью» и не считаются трафиком, поскольку используются внутри «OpenVPN-ГОСТ» и не являются показателем фактической активности пользователя.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

-ping n	Отправляет сигнал ping на удаленный хост через контрольный TCP/UDP-канал, если никаких пакетов не было отправлено по крайней мере n секунд (укажите ping для обоих партнеров, чтобы пакеты ping отправлялись в обоих направлениях, поскольку пакеты ping в «OpenVPN-ГОСТ» не отражаются, как пакеты ping в IP). При использовании в одном из защищенных режимов «OpenVPN-ГОСТ» (где указаны --secret, --tls-server или --tls-client) пакет ping будет криптографически защищен. Эта опция предназначена для двух вариантов использования: 1) Совместимость с брандмауэрами, работающими на сеансовом уровне. Периодический сигнал ping обеспечит то, что такой брандмауэр, который позволяет UDP-пакетам проходить, не оборвет сеанс по таймауту. 2) Предоставить удаленному хосту основу для тестирования существования его партнера с использованием опции -ping-exit.
--ping-exit n	Заставляет «OpenVPN-ГОСТ» завершать работу после того, как пройдут n секунд без получения пакета ping или другого пакета от удаленного хоста. Эта опция может использоваться совместно с --inactive, --ping и --ping-exit, чтобы создать двухярусное завершение работы в результате неактивности. Например openvpn-gost [options...] --inactive 3600 --ping 10 --ping-exit 60 использованное для обоих партнеров заставит «OpenVPN-ГОСТ» завершить работу в течение 60 секунд, если партнер отсоединяется, и в течение часа, если не будет обмена никакими реальными туннельными данными.

--ping-restart n	Похоже на --ping-exit, но начинает перезапуск с сигналом SIGUSR1 после того, как пройдут n секунд без принятия пакета ping или другого пакета от удаленного хоста. Эта опция полезна в тех случаях, когда удаленный партнер имеет динамический IP-адрес, и для следования за IP-адресом используется DNS-имя с низким TTL с применением сервиса типа http://dyndns.org и динамического DNS-клиента, такого как ddclient. Если партнера невозможно достичь, начнется перезапуск, заставляющий имя хоста, использованное с --remote, переразрешиться (если также указана опция --resolv-retry). В серверном режиме, --ping-restart, --inactive или любой другой вид внутреннего сигнала, всегда будет применяться к объектам индивидуальных клиентских экземпляров, а не к целому серверу. Также обратите внимание, что в серверном режиме любой внутренний сигнал, который в обычном случае вызовет перезапуск, вместо этого вызовет удаление объекта клиентского экземпляра. В клиентском режиме, параметр --ping-restart по умолчанию установлен в 120 секунд. Это умолчание будет действовать, пока клиент не получит другое значение с сервера, основанное на настройке --keepalive в серверной конфигурации. Чтобы отключить это умолчательное значение, установите на клиенте --ping-restart 0. См. в разделе сигналов внизу дополнительную информацию по SIGUSR1. Обратите внимание, что поведение SIGUSR1 может быть изменено опциями --persist-tun, --persist-key, --persist-local-ip и --persist-remote-ip. Также обратите внимание, что --ping-exit и --ping-restart - взаимоисключающие опции и не могут применяться совместно.

--keepalive interval timeout	Директива, предназначенная для того, чтобы упростить выражение --ping и --ping-restart в конфигурациях серверного режима.
	Эта опция может быть использована и на стороне клиента, и на стороне сервера, но достаточно использовать ее только на стороне сервера, поскольку она отправит соответствующий --ping и --ping-restart клиенту. Если использовать и на стороне сервера, и на стороне клиента, значения, направленные сервером, перепишут локальные значения клиента.
	Аргумент timeout будет в два раза длиннее на стороне сервера. Это обеспечит то, что тайм-аут будет обнаружен на стороне клиента до того, как сервер разорвет соединение.
	Например, --keepalive 10 60 расшифровывается так:
	`if mode server:`
	`ping 10 #Аргумент: interval`
	`ping-restart 120 #Аргумент: timeout*2`
	`push "ping 10" #Аргумент: interval`
	`push "ping-restart 60" #Аргумент: timeout`
	`else`
	`ping 10 #Аргумент: interval`
	`ping-restart 60 #Аргумент: timeout`

--ping-timer-rem	Запускать --ping-exit/--ping-restart только если у нас есть удаленный адрес. Используйте эту опцию, если вы запускаете демон в слушающем режиме (т.е. без определенного партнера --remote) и не хотите начинать подавать сигналы таймаутов, пока не соединится удаленный партнер.
--persist-tun	Не закрывать и не переоткрывать устройство TUN/TAP и не запускать скрипты при перезапусках SIGUSR1 или --ping-restart. SIGUSR1 – сигнал перезапуска, близкий к SIGHUP, но предлагающий более тонкий контроль над опциями перезапуска.
--persist-key	Не перечитывать ключевые файлы при SIGUSR1 или --ping-restart. Эту опцию можно совместить с --user nobody, чтобы позволять инициировать перезапуски сигналом SIGUSR1. Как правило, если вы отказываетесь от прав привилегированного пользователя в «OpenVPN-ГОСТ», демон не может быть перезапущен, потому что он теперь не сможет перечитать защищенные ключевые файлы. Эта опция решает проблему, запоминая прочитанные ключи при перезапусках SIGUSR1, так что их не нужно перечитывать.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--persist-local-ip	Сохранить изначально разрешенный местный IP-адрес и номер порта при перезапусках SIGUSR1 или --ping-restart.
--persist-remote-ip	Сохранить последний аутентифицированный удаленный IP-адрес и номер порта при перезапусках SIGUSR1 или --ping-restart.
--mlock	Отключает пейджинг, вызывая функцию mlockall из POSIX. Требует, чтобы «OpenVPN-ГОСТ» изначально была запущена под привилегированным пользователем (хотя потом можно отказаться от прав привилегированного пользователя с помощью опции --user). Использование этой опции обеспечивает то, что ключевые и туннельные данные никогда не записываются на диск из-за операций пейджинга виртуальной памяти, которые происходят в большинстве современных операционных систем. Она обеспечивает то, что даже если атакующий смог взломать компьютер, на котором работает «OpenVPN-ГОСТ», он не сможет просканировать системный swap-файл, чтобы получить ранее использованные эфемерные ключи, которые используются в течение периода времени, определенного опциями --reneg (см. ниже), затем от них отказываются. Обратная сторона использования --mlock – то, что она уменьшит количество физической памяти, доступной другим приложениям.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--up cmd	Команда оболочки, которую следует выполнить после успешного открытия устройства TUN/TAP (до изменения UID --user). cmd содержит путь к скрипту (или исполняемой программе), опционально используется с аргументами. Путь и аргументы могут быть заключены в одинарные или двойные кавычки и/или отделены обратной косой чертой, а также должны быть разделены одним или более пробелом. Скрипт up полезен для указания команд маршрутизации, которые маршрутизируют IP-трафик, предназначенный для частных подсетей, существующих на другом конце VPN-соединения, в туннель. Для --dev tun выполнять как: cmd tun_dev tun_mtu link_mtu ifconfig_local_ip ifconfig_remote_ip [init | restart] Для --dev tap выполнять как: cmd tap_dev tap_mtu link_mtu ifconfig_local_ip ifconfig_netmask [init | restart] См. в разделе «Переменные среды» ниже дополнительные параметры, которые передаются как переменные среды. Обратите внимание, что cmd может быть командой оболочки с несколькими аргументами, в этом случае аргументы, сгенерированные «OpenVPN-ГОСТ», будут добавлены в конец строки cmd, чтобы получить командную строку, которая будет передана в оболочку. Как правило, cmd запустит скрипт, чтобы добавить маршруты к туннелю. Как правило, после открытия устройства TUN/TAP вызывается скрипт up. В этом контексте последняя команда, которую параметр передал в скрипт, будет init. Если опция --up-restart также используется, скрипт up будет вызываться и для перезапусков. Перезапуск считается частичной переинициализацией «OpenVPN-ГОСТ», где сохраняется экземпляр TUN/TAP (это сохранение обеспечивает опция --persist-tun). Перезапуск может быть сгенерирован сигналом SIGUSR1, таймаутом --ping-restart или перезапуском соединения, когда опцией --proto запускается протокол TCP. Если происходит перезапуск, и была указана опция --up-restart, скрипт up будет вызван с restart в качестве последнего параметра. Обратите внимание: при перезапуске «OpenVPN-ГОСТ» не передает скрипту полный набор переменных среды. А именно все, что связано с маршрутизацией и шлюзами, не будет передано, так как ничего не нужно делать – вся настройка маршрутизации уже произведена. Кроме того каждый up-перезапуск скрипта будет запускаться с пониженными UID/GID настройками (если они заданы).

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

	Следующий отдельный пример показывает, как скрипт --up может быть вызван в контексте и инициализации, и перезапуска (ПРИМЕЧАНИЕ: в целях безопасности, не запускайте этот пример, если UDP-порт 9999 не заблокирован вашим брандмауэром. Кроме того, пример будет выполняться бесконечно, так что его следует прервать с помощью control-c). openvpn-gost --dev tun --port 9999 --verb 4 --ping-restart 10 --up 'echo up' --down 'echo down' --persist-tun --up-restart Обратите внимание, что «OpenVPN-ГОСТ» также предоставляет опцию --ifconfig, чтобы автоматически применить ifconfig к устройству TUN, исключая необходимость определять скрипт --up, если только вы не хотите также сконфигурировать маршруты в скрипте --up.
	Если --ifconfig также указана, «OpenVPN-ГОСТ» передаст локальный и удаленный конечные пункты команды ifconfig через командную строку в скрипт --up, так что они могут быть использованы, чтобы сконфигурировать маршруты, такие как: route add -net 10.0.0.0 netmask 255.255.255.0 gw $5
--up-delay	Отложить открытие TUN/TAP и возможное выполнение скрипта --up до тех пор, пока не установится TCP/UDP-соединение. В режиме --proto udp эта опция, как правило, требует использование --ping, чтобы дать возможность почувствовать инициацию соединения в отсутствие туннельных данных, поскольку UDP - «бессоединительный» протокол. В Windows эта опция отложит переход состояния TAP-Win32 в «подключенный», пока не установится соединение, т.е. пока не будет получен первый аутентифицированный пакет от партнера.
--down cmd	Команда оболочки, выполняемая после закрытия устройства TUN/TAP (после изменения UID --user и/или chroot). cmd содержит путь к скрипту (или исполняемой программе), опционально используется с аргументами. Путь и аргументы могут быть заключены в одинарные или двойные кавычки и/или отделены обратной косой чертой, а также должны быть разделены одним или более пробелом. Вызывается с теми же параметрами и переменными среди, что и опция --up, описанная выше. Обратите внимание, что если вы снижаете права, пользуясь опциями --user и/или group, ваш скрипт --down также будет выполняться с пониженными правами.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--down-pre	Вызывать команду/скрипт --down до, а не после закрытия TUN/TAP.
--up-restart	Обеспечить возможность вызывать скрипты --up и --down для перезапусков, как и для изначального старта программы. Эта опция более подробно описана выше в описании опции --up.
--setenv name value	Настраивает измененную переменную среды name=value для передачи в скрипт.
--setenv FORWARD_COMPATIBLE 1	Ослабить проверку синтаксиса конфигурационного файла так, чтобы неизвестные директивы вызывали предупреждение, а не фатальную ошибку, в предположении, что данная неизвестная директива может быть действительной в будущих версиях «OpenVPN-ГОСТ». Этой опцией следует пользоваться с осторожностью, потому что есть серьезные причины, связанные с безопасностью, заставлять «OpenVPN-ГОСТ» прекращать работу при проблемах в конфигурационном файле. Тем не менее есть причины желать, чтобы новые программные возможности не вызывали проблем при встрече со старыми программными версиями.
--setenv-safe name value	Установить измененную переменную среды OPENVPN_name=value для передачи в скрипт. Эта директива предназначена для передачи с сервера клиентам, и добавление префикса «OPENVPN_» к переменной среды – мера предосторожности, чтобы предотвратить атаку в стиле LD_PRELOAD от злонамеренного или скомпрометированного сервера.
--ignore-unknown-option opt1 opt2 opt3 ... optN	Когда в конфигурационном файле встречается одна из опций opt1 ... optN, анализ конфигурационного файла не падает с ошибкой, если «OpenVPN-ГОСТ» не поддерживает эту опцию. Несколько --ignore-unknown-option опций могут быть использованы, чтоб поддержать большее количество игнорируемых опций. Эту опцию следует использовать осторожно, так как есть ситуации, когда «OpenVPN-ГОСТ» падает из соображений безопасности, если обнаруживает проблемы в конфигурационном файле. Логично ожидать, что новые функции будут хуже работать на старых версиях программного обеспечения.

--script-security level	Эта директива предоставляет контроль над тем, как «OpenVPN-ГОСТ» использует внешние программы и скрипты, с политиками, определенными через уровни. Более низкие значения level накладывают больше ограничений, высокие – меньше. Настройки для level: 0 - строго не вызывать внешних программ 1 (по умолчанию) - вызывать только встроенные исполняемые файлы, такие как ifconfig, ip, route или netsh. 2 - позволить вызывать встроенные исполняемые файлы, а также скрипты, определенные пользователем. 3 - позволить передавать пароли в скрипты через переменные среды (вероятно небезопасно). Некоторые директивы, такие как --up, позволяют передавать опции внешним скриптам. В таких случаях убедитесь, что имя скрипта не содержит пробелов или конфигурационный анализатор остановится, так как не сможет определить, где заканчивается имя скрипта и начинаются опции. Для запуска скриптов в Windows в более ранних версиях «OpenVPN-ГОСТ» вам надо было либо добавить полный путь к интерпретатору скриптов, который может проанализировать скрипт, либо использовать системный флаг, чтоб запустить эти скрипты. Сейчас «OpenVPN-ГОСТ» строго требует иметь полный путь к интерпретатору скриптов при запуске неисполняемых файлов. Это не требуется для исполняемых файлов, таких как .exe, .com, .bat или .cmd. К примеру, если у вас есть Visual Basic скрипт, вы должны использовать теперь следующий синтаксис: --up 'C:\\Windows\\System32\\wscript.exe C:\\CryptoPack4\\config\\my-up-script.vbs' Обратите внимание на одинарные кавычки и экранирование обратной косой черты (\) и пробела. Причина, по которой поддержка системного флага была удалена, связана с последствиями для безопасности, когда выполняются скрипты через вызов system().
--disable-occ	Не выводить предупреждающее сообщение, если между партнерами обнаружены несоответствия опций. Пример несоответствия опций - если один из партнеров использует --dev tun, а второй --dev tap. Использовать эту опцию не рекомендуется, но она предоставляется в качестве временного решения в ситуациях, где «OpenVPN-ГОСТ» соединяется с «OpenVPN-ГОСТ» более старой версии.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--user user	Изменяет ID пользователя в процессе «OpenVPN-ГОСТ» в user после инициализации, понижая права. Эта опция полезна, чтобы защитить систему в том случае, когда кто-то враждебный смог получить контроль над сеансом «OpenVPN-ГОСТ». Хотя благодаря защите «OpenVPN-ГОСТ» это маловероятно, эта опция предоставляет дополнительное средство безопасности. Установив user в nobody или что-то такое же непривилегированное, вы ограничите вред, который может быть причинен враждебным агентом. Конечно, отняв права, вы не сможете их вернуть в сеансе «OpenVPN-ГОСТ». Это значит, например, что если вы хотите перезапустить демон «OpenVPN-ГОСТ» с помощью сигнала SIGUSR1 (например, в ответ на перезапуск DHCP), вам следует воспользоваться одной или несколькими опциями --persist, чтобы обеспечить «OpenVPN-ГОСТ» отсутствие необходимости пользоваться какими-либо привилегированными операциями, чтобы перезапуститься (такими, как перечитывание ключевых файлов или запуск ifconfig на устройстве TUN).
--group group	Подобно опции --user, эта опция меняет ID группы в процессе «OpenVPN-ГОСТ» в group после инициализации.
-cd dir	Изменить каталог в dir перед чтением любых файлов, таких как конфигурационные файлы, ключевые файлы, скрипты и т.д. dir должен быть абсолютным путем, начинающимся с «/», и без каких-либо ссылок на текущий каталог, таких как «.» или «..». Эта опция полезна, когда вы запускаете «OpenVPN-ГОСТ» в режиме --daemon и хотите свести все ваши контрольные файлы «OpenVPN-ГОСТ» в одном месте.
--chroot dir	Перейти в dir как в chroot после инициализации. --chroot, в сущности, переопределяет dir как верхний уровень каталога в дереве (/). «OpenVPN-ГОСТ» таким образом не сможет достичь никаких файлов вне этого дерева. Это может быть желательно с точки зрения безопасности. Поскольку операция chroot выполняется после инициализации, большинство опций «OpenVPN-ГОСТ», связанные с файлами, будут работать в контексте, предваряющем chroot. Во многих случаях параметр dir может указывать на пустой каталог, но могут появиться проблемы, когда после операции chroot выполняются скрипты или перезапуски.

--setcon context	Применить SELinux context после инициализации. Это, в сущности, предоставляет возможность ограничить права «OpenVPN-ГОСТ» до сетевых операций входа/выхода, благодаря SELinux. Эта опция идет дальше, чем --user и --chroot, потому что эти две опции, будучи прекрасными возможностями защиты, к сожалению, не защищают от повышения привилегий с помощью использования уязвимого системного вызова. Вы можете, конечно, использовать все три опции сразу, но пожалуйста имейте в виду, что поскольку setcon требует доступа в /proc, вам придется предоставлять его внутри каталога chroot (например, с помощью mount --bind). Поскольку операция setcon выполняется после инициализации, «OpenVPN-ГОСТ» может быть ограничена только системными вызовами, связанными с сетью, в то время как при использовании контекста перед запуском (таким, как предлагаемый в SELinux Reference Policies) вам придется позволить много вещей, требуемых только во время инициализации. Как и с chroot, могут возникнуть проблемы, когда после операции setcon выполняются скрипты или перезапуски, поэтому следует призадуматься над использованием опций --persist-key и --persist-tun.

--daemon [progname]	Стать демоном после завершения всех функций инициализации. Эта опция заставит все сообщения отправляться в файл syslog (такой как /var/log/messages) кроме вывода скриптов оболочки и команд ifconfig, которые будут отправляться в /dev/null, если их не перенаправить в другое место. Перенаправление в syslog происходит немедленно в момент, когда опция -daemon прочитывается в командной строке, хотя момент демонизации наступает позже. Если присутствует одна из опций --log, она заменит перенаправление в syslog. Опциональный параметр progname заставит «OpenVPN-ГОСТ» сообщить свое программное имя системному логгеру как progname. Это может быть полезно для связывания сообщений «OpenVPN-ГОСТ» в файле syslog со специальными туннелями. Если progname не указано, по умолчанию оно openvpn-gost. Когда «OpenVPN-ГОСТ» запускается с опцией --daemon, она попытается отложить демонизацию до того момента, как завершится большинство инициализационных функций, способных к генерации фатальных ошибок. Это означает, что инициализационные скрипты могут протестировать возвращенный статус команды openvpn-gost с достаточно надежным указанием того, была ли команда корректно инициализирована и вошла ли в петлю событий форвардинга пакетов. В «OpenVPN-ГОСТ» большая часть ошибок, происходящих после инициализации, не фатальны. Учтите, как только «OpenVPN-ГОСТ» станет демоном, он больше не сможет запрашивать имена пользователей, пароли, pin-коды или ключевые фразы. Это имеет определенные последствия, а именно то, что использование защищенного паролем закрытого ключа или защищённого pin-кодом токена не сработает, если только не используетсяопция --askpass или --askpin. Кроме того, использование --daemon вместе с --auth-user-pass (введенные с консоли) и --auth-nocache завершится с ошибкой, как только произойдет повторное согласование ключа (и повторная аутентификация).
--syslog [progname]	Направлять вывод журнала в системный логгер, но не становиться демоном. См. в описании директивы --daemon, приведенном выше, описание параметра progname.
--errors-to-stderr	Выводить ошибки в ыевукк вместо ыевщге, если журнала вывода не перенаправляется одной из опций --log.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--passtos	Установить поле TOS туннельного пакета в то, чем является TOS информационного наполнения пакета.
--inetd [wait|nowait] [progname]	Используйте эту опцию, когда «OpenVPN-ГОСТ» работает на сервере inetd или xinetd(8). Опция wait/nowait должна соответствовать тому, что указано в конфигурационном файле ietd/xinetd. Режим nowait может быть использован только с --proto tcp-server. По умолчанию wait. Режим nowait может быть использован, чтобы представить демон «OpenVPN-ГОСТ» как классический TCP-сервер, где запросы клиентских соединений обслуживаются на одном номере порта. Эта опция исключает использование опций --daemon, --local или --remote. Обратите внимание, что эта опция заставляет обрабатывать сообщения, в том числе об ошибках, так же, как опция --daemon. Опциональный параметр progname также обрабатывается точно так же, как в --daemon. Также обратите внимание, что в режиме wait каждый тунель «OpenVPN-ГОСТ» требует отдельного TCP/UDP-порта и отдельного входа inetd или xinetd.
--log file	Выводить журнальные сообщения в file, включая вывод на стандартный вывод/стандартную ошибку, который генерируется вызванными скриптами. Если file уже существует, он будет обрезан. Эта опция активизируется немедленно, когда она прочитывается в командной строке, и заменит вывод в syslog, если также указаны --daemon или --inetd. Эта опция сохраняется в течение всей работы «OpenVPN-ГОСТ» и не перезагружается SIGHUP, SIGUSR1 или --ping-restart. Обратите внимание, что в Windows, где «OpenVPN-ГОСТ» запускается как сервис, журналирование присходит по умолчанию без необходимости указывать эту опцию.
--log-append file	Добавить журнальные сообщения к концу file. Если file не существует, он будет создан. Эта опция ведет себя точно также, как --log, за исключением того, что она приписывает сообщения к концу файла, а не обрезает его.
--suppress-timestamps	Избегать записи меток времени в журнальные сообщения, даже тогда, когда иначе они будут добавлены в начало. В особенности это прилагается к сообщениям журнала, направляемым в стандартный вывод.

--machine-readable-output	Всегда записывать метки времени и флаги сообщений в журнальные сообщения, даже когда в противном случае они не были бы записаны. В частности, это применяется к сообщениям журнала, отправляемым в стандартный вывод.
--writepid file	Записать ID главного процесса «OpenVPN-ГОСТ» в file.
--nice n	Изменяет приоритет процесса после инициализации (n больше 0 - понижение приоритета, n меньше нуля – повышение приоритета).
--fast-io	(Экспериментальная) Оптимизировать записи ввода-вывода TUN/TAP/UDP, избегая вызова в poll/epoll/select перед операцией записи. Целью такого вызова, как правило, является блокировка, пока устройство или сокет не будут готовы принять запись. Такая блокировка не является необходимой на некоторых платформах, которые не поддерживают блокировку записей на UDP-сокетах или устройствах TUN/TAP. В таких случаях можно оптимизировать петлю событий, избегая вызова в poll/epoll/select, тем самым улучшая эффективность CPU на 5-10 процентов. Эта опция может быть использована только в системах, отличных от Windows, когда указана опция --proto udp, и когда опция --shaper НЕ указана.
--multihome	Сконфигурировать многодомный UDP-сервер. Эта опция должна быть использована, когда сервер имеет более одного IP-адреса (например, несколько интерфейсов или вторичных IP-адресов) и не использует --local для принудительной привязки только к одному определенному адресу. Эта опция может быть использована, когда «OpenVPN-ГОСТ» была сконфигурирована так, чтобы слушать на всех интерфейсах, и будет пытаться привязать клиентские сеансы к интерфейсу, на который приходят пакеты, чтобы исходящие пакеты шли через этот же интерфейс. Обратите внимание, что эта опция релевантна только для UDP-серверов. Так же обратите внимание, что если вы выполняете привязку двойного стека IPv6+IPv4 на Linux машине с несколькими IPv4 адресами, подключения к IPv4 адресам не будут работать правильно на ядрах до версии 3.15 из-за отсутствия поддержки ядра в случае сопоставления IPv4 (некоторые дистрибутивы, однако, портировали это на более ранние версии ядра).

--echo [parms...]	Копирует parms в вывод журнала. Предназначено для того, чтобы отправлять сообщения контролируемому приложению, которое получает вывод журнала «OpenVPN-ГОСТ».
--remap-usr1 signal	Контролировать, внутренние или внешние сигналы SIGUSR1 превращаются в SIGHUP (перезапуск без сохраненного состояния) или SIGTERM (выход). signal может принимать значения SIGHUP или SIGTERM. По умолчанию, никакого превращения не происходит.
--verb n	Устанавливает подробность вывода в n (по умолчанию 1). Каждый уровень показывает всю информацию с предыдущих уровней. Уровень 3 рекомендуется, если вы хотите хороший отчет о происходящем без излишней детализации. 0 - никакого вывода, кроме фатальных ошибок. 1-4 - диапазон нормального использования. 5 - выводит символы R и W на консоль для каждого прочтенного и записанного пакета, заглавные для TCP/UDP-пакетов и строчные для TUN/TAP-пакетов. 6-11 - диапазон отладочных уровней.
--status file [n]	Записывать операционный статус в file каждые n секунд. Статус также может записываться в syslog отправлением сигнала SIGUSR2. Когда на сервере включена возможность работать с несколькими клиентами, статус файл включает список клиентов и таблицу маршрутизации. В этом случае формат вывода может контролироваться с помощью --status-version. Для клиентов или экземпляров, работающих в режиме точка-в-точку, он будет содержать статистику трафика.
--status-version[n]	Выбрать номер версии формата файла статуса. Это касается только статус файла на серверах со включенной возможностью работы с несколькими клиентами. 1 – традиционный формат (по умолчанию). Список клиентов содержит следующие поля через запятую: Common Name, Real Address, Bytes Received, Bytes Sent, Connected Since. 2 – более надежный формат для внешней обработки. По сравнению с версией 1 список клиентов содержит некоторые дополнительные поля: Virtual Address, Virtual IPv6 Address, Username, Client ID, Peer ID. Будущие версии могут расширить список полей. 3 – идентичен 2, но поля разделены табуляцией .

--mute n	Записывать в журнал не больше n последовательных сообщений одной и той же категории. Полезно для ограничения повторяющегося журналирования схожих типов сообщений.
--compress [algorithm]	Включает алгоритм сжатия. Параметр algorithm может принимать значения «lzo», «lz4», также можно не указывать конкретных алгоритм. LZO и LZ4 – это разные алгоритмы сжатия, LZ4 в целом предлагает наилучшую производительность с наименьшей нагрузкой на процессор. Если алгоритм не указан, то сжатие будет отключено, однако кадрирование пакетов для сжатия по-прежнему будет включено, позволяя позже добавить другие настройки. Таким образом, наличие опции compress без указания алгоритма не эквивалентно отсутствию этой опции. Если на сервере параметр отсутствует, а на клиенте указан compress без параметра, соединение установится, но данные по нему передаваться не будут. Если сервер делает push опции compress, то соединение установится независимо от того, указана ли опция compress на клиенте и с каким значением. Таким образом, вписывать compress в клиентский конфиг имеет смысл только тогда, когда сервер не делает push. Вопросы безопасности: Сжатие и шифрование – опасная комбинация. Известно несколько методов взлома (см., например, CRIME и BREACH на TLS), позволяющих злоумышленнику восстановить зашифрованные данные, используя особенности сжатия данных. Если вы не уверены, что вышеизложенное не относится к вашему траффику, то рекомендуется НЕ включать сжатие.

--comp-lzo [mode]	УСТАРЕЛО Используйте новую опцию --compress взамен. Использовать быстрое сжатие LZO - можно добавлять до 1 байта на пакет для несжимаемых данных. mode может быть «yes», «no» или «adaptive» (умолчание). В настройках серверного режима возможно избирательно включать и выключать сжатие для индивидуальных клиентов. Сначала удостоверьтесь, что конфигурационный файл клиентской стороны обеспечивает избирательное сжатие, имея по меньшей мере одну директиву --comp-lzo, такую как --com-lzo no. Это отключит сжатие по умолчанию, но позволит будущей директиве, переданной с сервера, динамически изменять настройки включения/выключения/адаптации. Далее, в файле --client-config-dir укажите настройку сжатия для клиента, например: comp-lzo yes push «comp-lzo yes» Первая строка устанавливает настройку comp-lzo для серверной стороны соединения, вторая настраивает клиентскую сторону.
--comp-noadapt	При использовании вместе с --comp-lzo, эта опция отключает алгоритм адаптивного сжатия «OpenVPN-ГОСТ». Как правило, адаптивное сжатие обеспечивается --comp-lzo. Адаптивное сжатие пытается оптимизировать случай, где у вас включено сжатие, но вы посылаете в основном несжимаемые (или уже сжатые) пакеты по туннелю, такие как передача по FTP или rsync большого сжатого файла. С адаптивным сжатием «OpenVPN-ГОСТ» будет периодически запускать на пробу процесс сжатия, чтобы измерить его эффективность. Если данные, передаваемые по туннелю, уже сжаты, эффективность сжатия будет очень низкой, что заставит «OpenVPN-ГОСТ» отключить сжатие на период времени до следующего пробного сжатия.

--management socket-name unix [pw-file] (recommended) --management IP port [pw-file]	Включает сервер управления в сокете Unix с socket-name на тех платформах, которые его поддерживают, или на указанном порту TCP. pw-file, если указано, является файлом с паролем (пароль в первой строке) или «stdin» для ввода пароля со стандартного ввода. В то время как умолчательное поведение – создать сокет домена Unix, с которым можно соединиться любым процессом, можно использовать директивы --management-client-user и --management-client-group, чтобы ограничить адрес. Интерфейс управления предоставляет специальный режим, где связь с управлением TCP может работать через сам туннель. Чтобы включить этот режим, установите значение IP в «tunnel». Туннельный режим заставит интерфейс управления слушать соединение TCP на локальном адресе VPN интерфейса TUN/TAP. ОСТЕРЕГАЙТЕСЬ включения интерфейса управления через TCP. В этих случаях нужно ВСЕГДА использовать pw-file для защиты паролем интерфейса управления. Любой пользователь, кто может подключиться к этому TCP IP:port, сможет управлять и контролировать (и вмешиваться в) «OpenVPN-ГОСТ» процесс . Настоятельно рекомендуется установить IP в 127.0.0.1 (localhost), чтобы ограничить доступность сервера управления для локальных клиентов. Хотя порт управления предназначен для программного контроля других приложений над «OpenVPN-ГОСТ», возможно связаться с портом с помощью telnet, используя клиент telnet в «грубом» режиме. Установив соединение, наберите «help», чтобы получить список команд. Подробную документацию по интерфейсу управления см. в файле management-notes.txt в папке управления исходного дистрибутива «OpenVPN-ГОСТ».
--management-client	Интерфейс управления будет подключаться как клиент домена TCP/unix к IP:port, установленные с помощью --management, а не слушать в качестве TCP сервера или сокета домена unix. Если подключение в качестве клиента не удастся или будет разорвано, будет сгенерирован SIGTERM сигнал, который приведет к отключению «OpenVPN-ГОСТ».
--management-query-passwords	Канал управления запросами для пароля закрытого ключа и логина/пароля опции --auth-user-pass. Запрашивайте через канал управления только те элементы ввода, которые, как правило, запрашиваются с консоли.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--management-query-proxy	Канал управления запросами информации о прокси-сервере для конкретного --remote (только для клиентов).
--management-query-remote	Позволяет интерфейсу управления переписывать --remote директивы (только для клиентов).
--management-query-key	Разрешает использовать внешний файл закрытого ключа вместо --key опции (только для клиента).
--management-query-cert certificate-hint	Разрешает использование внешнего сертификата вместо --cert опции (только для клиентов). certificate-hint – это произвольная строка, которая передается интерфейсу управления клиента в качестве аргумента в NEED-CERTIFICATE уведомлении. Требует --management-external-key.
--management-forget-disconnect	Заставить «OpenVPN-ГОСТ» забывать пароли, когда сеанс управления заканчивается. Эта директива не влияет на логин и пароль опции --http-proxy. Они всегда кэшируются.
--management-hold	Запустить «OpenVPN-ГОСТ» в состоянии гибернации, пока клиент интерфейса управления не запустит ее явным образом командой hold release.
--management-signal	Послать «OpenVPN-ГОСТ» сигнал SIGUSR1, если сеанс управления обрывается. Это полезно, когда вы хотите разорвать сеанс «OpenVPN-ГОСТ» при отключении пользователя.
--management-log-cache n	Кэшировать последние n строк истории файла журнала для использования на канале управления.
--management-up-down	Сообщать о событиях подключения/отключения туннеля интерфейсу управления.
--management-client-auth	Дает клиенту интерфейса управления ответственность за аутентификацию клиентов после того, как их клиентский сертификат был проверен.
--management-client-pf	Клиенты интерфейса управления должны указать файл фильтра пакетов для каждого соединяющегося клиента.
--management-client-user u	Когда интерфейс управления слушает на сокете домена Unix, позволять только соединения для пользователя u.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--management-client-group g	Когда интерфейс управления слушает на сокете домена Unix, позволять только соединения для группы g.

--plugin module-pathname [init-string]	Загрузить подключаемый модуль из файла module-pathname, передавая init-string как аргумент в функцию инициализации модуля. В один процесс «OpenVPN-ГОСТ» можно загрузить несколько подключаемых модулей.
	Параметр module-pathname может быть либо именем файла, либо именем файла с относительным или полным путем. Формат имени файла и пути определяют, будет ли подключаемый модуль загружаться из каталога подключаемого модуля по умолчанию или из другой директории.
	`-{}-plugin path  Effective directory used`
	`==========`
	`myplug.so  DEFAULT_DIR/myplug.so`
	`subdir/myplug.so  DEFAULT_DIR/subdir/myplug.so`
	`./subdir/myplug.so  CWD/subdir/myplug.so`
	`/usr/lib/my/plug.so  /usr/lib/my/plug.so`
	DEFAULT_DIR заменяется каталогом подключаемого модуля по умолчанию, который задается по время сборки «OpenVPN-ГОСТ». CWD – это текущий каталог, в котором был запущен «OpenVPN-ГОСТ» или каталог, в который «OpenVPN-ГОСТ» перешел с помощью опции --cd перед опцией --plugin.
	Несколько модулей могут быть подключены каскадом, и можно использовать модули вместе со скриптами. «OpenVPN-ГОСТ» будет вызывать модули в том порядке, в котором они декларированы в конфигурационном файле. Если для одного и того же обратного вызова сконфигурированы и модуль, и скрипт, скрипт будет вызван последним. Если код возвращения модуля/скрипта контролирует аутентификационную функцию (такую как tls-verify, auth-user-pass-verify или client-connect), то каждый модуль и скрипт должны вернуть успех (0), чтобы соединение было аутентифицировано.

--keying-material-exporter label len	Сохранить Экспортированный Материал Ключей [RFC5705] из len байт (должен быть от 16 до 4095 байт), используя метку в среде (exported_keying_material) для использования плагинами в обратном вызове OPENVPN_PLUGIN_TLS_FINAL. Учтите, что метки экспортера потенциально могут конфликтовать с существующими PRF метками. Чтобы избежать этого, метки ДОЛЖНЫ начинаться с «EXPORTER».

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

8.3 Серверный режим

В «OpenVPN-ГОСТ» поддерживается многоклиентный серверный режим, который может быть включен с помощь опции --mode server. В серверном режиме «OpenVPN-ГОСТ» слушает на одном порту входящие клиентские соединения. Все клиентские соединения маршрутизируются через один интерфейс tun или tap. Этот режим предусматривает масштабирование и должен быть в состоянии поддерживать сотни и даже тысячи клиентов на достаточно быстрых машинах. В этом режиме должна использоваться аутентификация SSL/TLS.

--server network netmask ['nopool']	Вспомогательная директива, предназначенная для того, чтобы упростить конфигурацию серверного режима. Эта директива настраивает сервер «OpenVPN-ГОСТ», который присваивает клиентам адреса из указанных network/netmask. Сам сервер примет адрес «.1» данной сети в качестве использования как серверный конечный пункт локального интерфейса TUN/TAP. Например, --server 10.8.0.0 255.255.255.0 расшифровывается следующим образом:
	`mode server`
	`tls-server`
	`push "topology [topology]"`
	
	`if dev tun AND (topology == net30 OR`
	`topology == p2p):`
	`ifconfig 10.9.1.1 10.9.1.2`
	`if !nopool:`
	`ifconfig-pool 10.9.1.4 10.9.1.251`
	`route 10.9.1.0 255.255.255.0`
	`if client-to-client:`
	`push "route 10.9.1.0 255.255.255.0"`
	`else if topology == net30:`
	`push "route 10.9.1.1"`
	
	`if dev tap OR (dev tun AND topology == subnet):`
	`ifconfig 10.9.1.1 255.255.255.0`
	`if !nopool:`
	`ifconfig-pool 10.9.1.2 10.9.1.254 255.255.255.0`
	`push "route-gateway 10.9.1.1"`
	Не используйте опцию --server, если вы устанавливаете связь типа «мост». В этом случае пользуйтесь опцией --server-bridge.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--server-bridge gateway netmask pool-start-IP pool-end-IP
--server-bridge ['nogw']
	Вспомогательная директива, подобная --server, предназначенная для упрощения конфигурации «OpenVPN-ГОСТ» в серверном режиме в конфигурациях типа «мост».
	Если опция --server-bridge используется без каких-либо параметров, она включает режим DHCP-прокси, где подключающиеся клиенты будут получать IP-адрес для своих адаптеров TAP от DHCP-сервера, запущенного в локальной сети серверной стороны «OpenVPN-ГОСТ». Обратите внимание, что только клиенты, поддерживающие связь DHCP-клиента с адаптером TAP (такие как Windows) могут поддерживать этот режим. Опциональный флаг nogw указывает, что информацию о гейте не следует передавать клиенту.
	Чтобы сконфигурировать связь типа «мост», вам в первую очередь следует воспользоваться возможностями вашей операционной системы, чтобы связать «мостом» интерфейс TAP с интерфейсом NIC Ethernet. Например, в Linux это делается с помощью инструмента brctl, а в Windows XP это выполняется в Network Connections Panel выбором адаптеров TAP и Ethernet и щелчком правой клавишей на «Bridge Connections». Затем вы должны вручную установить IP и маску сети интерфейса «моста». Параметры gateway и netmask опции --server-bridge могут быть установлены или в IP и маску сети интерфейса «моста», или в IP и маску сети умолчательного гейта/-роутера подсоединенной «мостом» подсети.
	Наконец, укажите диапазон IP для подсоединенной «мостом» подсети, обозначенный параметрами pool-start-IP и pool-end-IP, чтобы «OpenVPN-ГОСТ» назначала их подключающимся клиентам.
	Например, server-bridge 10.8.0.4 255.255.255.0 10.8.0.128 10.8.0.254 расшифровывается следующим образом:
	`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"`
	
	В другом примере, --server-bridge (без параметров) расшифровывается как:
	`mode server`
	`tls-server`
	
	`push "route-gateway dhcp"`

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

	Или --server-bridge nogw расшифровывается как:
	`mode server`
	`tls-server`

--push option	Передать опцию конфигурационного файла обратно клиенту для удаленного выполнения. Обратите внимание, что option должна быть окружена двойными кавычками (""). Клиент должен указать --pull в своем конфигурационном файле. Набор опций, которые могут быть переданы, ограничен как их применимостью, так и соображениями безопасности. Некоторые опции, например, выполняющие скрипты, запрещены, поскольку они в сущности позволяют скомпрометированному серверу выполнять произвольный код на клиенте. Другие опции, такие как параметры TLS и MTU, не могут быть переданы, потому что клиенту нужно знать их до того, как инициируется соединение с сервером. Вот частичный список опций, которые в настоящее время могут быть переданы: --route, --route-gateway, --route-delay, --redirect-gateway, --ip-win32, --dhcp-option, --inactive, --ping, --ping-exit, --ping-restart, --setenv, --auth-token, --persist-key, --persist-tun, --echo, --comp-lzo, --socket-flags, --sndbuf, --rcvbuf.
--push-reset	Не наследовать глобальный список переданных опций для специального экземпляра клиента. Указывайте эту опцию в клиент-специфичном контексте, например, с конфигурационным файлом --client-config-dir. Эта опция будет игнорировать опции --push на глобальном уровне конфигурационного файла.
--push-remove opt	Выборочно удалить все --push опции, соответствующие “opt” из список опций для клиента. “opt” сопоставляется как подстрока со строкой опции, которая должна быть отправлена клиенту, поэтому --push-remove route удалит все --push route ... и выражения --push route-ipv6 ..., тогда как --push-remove ’route-ipv6 2001:’ удалит только IPv6 маршруты для 2001:... сетей.

--push-peer-info	Передать дополнительную информацию о клиенте на сервер. Следующие данные всегда передаются серверу: IV_VER=<version> -- клинтская версия «OpenVPN-ГОСТ» IV_PLAT=[linux|solaris|mac|freebsd|win] – клиентская ОС IV_LZO_STUB=1 – если клиент был реализован с возможностью заглушки LZO IV_LZ4=1 – если клиент поддерживает сжатие LZ4 IV_PROTO=2 – если клиент поддерживает плавающий механизм peer-id IV_NCP=2 – согласуемые шифры, клиент поддерживает --cipher, передаваемый сервером, значение 2 и выше означает, что клиент поддерживает AES-GCM-128 и AES-GCM-256. IV_GUI_VER=<gui_id> <version> – версия пользовательского интерфейса, если он запущен. Когда --push-peer-info включен, дополнительно передается следующая информация: IV_HWADDR=<mac address> – MAC-адрес клиентского шлюза по умолчанию IV_SSL=<version string> – версия ssl, используемая клиентом, к примеру, "OpenSSL 1.1.1o CryptoPack4.0". IV_PLAT_VER=x.y - версия ОС, к примеру, 6.1 для Windows 7. UV_<name>=<value> – переменные среды клиента, имена которых начинаются с "UV_"
disable	Не допустить конкретного клиента (основывается на поле common name) к соединению. Не используйте эту опцию, чтобы не допускать клиента со скомпроментированным ключом или паролем. Используйте вместо этого CRL (список отзыва сертификатов, см. опцию --crl-verify). Эта опция должна быть ассоциирована с конкретным экземпляром клиента, что означает, что она должна быть указана или в клиентском конфигурационном файле с использованием --client-config-dir или динамически сгенерирована с использованием скрипта --client-connect.
--ifconfig-pool start-IP end-IP [netmask]	Указать набор подсетей, динамически назначаемых соединяющимся клиентам, подобно серверу DHCP. Для туннелей в tun-стиле каждому клиенту выделяется подсеть /30 (для совместимости с клиентами Windows). Для туннелей в tap-стиле выделяются индивидуальные адреса, и необязательный параметр netmask также будет передан клиентам.

--ifconfig-pool-persist file [seconds]	Сохранять/удалять данные ifconfig-pool в file, через интервалы seconds (600 по умолчанию), также как и при старте и завершении работы программы. Цель этой опции – предоставить долгосрочную ассоциацию между клиентами (обозначенными их полями common name) и виртуальным IP-адресом, присвоенным им из ifconfig-pool. Поддержка долгосрочной ассоциации полезна для клиентов, потому что она позволяет им эффективно использовать опцию --persist-tun. file - ASCII-файл с разделителями-запятыми, отформатированный как <Common-Name>,<IP-address>. Если seconds=0, file будет рассматриваться как файл только для чтения. Это полезно, если вы хотите рассматривать file как конфигурационный файл. Обратите внимание, что записи в этом файле рассматриваются «OpenVPN-ГОСТ» только как предложения, основанные на прошлых ассоциациях между common name и IP-адресом. Они не гарантируют, что данное common name будет всегда получать данный IP-адрес. Если вы хотите гарантированное присвоение, пользуйтесь опцией --ifconfig-push.
--ifconfig-pool-linear	Модифицирует директиву --ifconfig-pool так, чтобы присваивать клиентам индивидуальные адреса интерфейса TUN, а не подсети /30. ПРИМЕЧАНИЕ: Эта опция несовместима с клиентами Windows. Эта опция не рекомендуется к использованию и должна быть заменена на опцию --topology p2p, которая функционально эквивалентна ей.

--ifconfig-push local remote-netmask [alias]	Передать виртуальные IP-конечные пункты для клиентского туннеля, заменяя динамическое присвоение --ifconfig-pool. Параметры local и remote-netmask устанавливаются в соответствии с директивой --ifconfig, которую вы хотите выполнять на клиентской машине для конфигурирования удаленного конца туннеля. Обратите внимание, что параметры local и remote-netmask существуют с точки зрения клиента, а не сервера. Они могут быть DNS-именами, а не IP-адресами, в этом случае они будут разрешаться на сервере во время соединения клиента. Опциональный параметр alias может быть использован в случаях, где NAT приводит к тому, что локальная конечная точка не совпадает с точки зрения клиента и с точки зрения сервера. В этом случае local/remote-netmask будет ссылаться на представление сервера, в то время как alias/remote-netmask будет ссылаться на представление клиента. Эта опция должна быть ассоциирована с конкретным экземпляром клиента, что означает, что она должна быть указана или в конфигурационном файле экземпляра клиента с использованием --client-config-file, или динамически генерироваться с использованием скрипта --client-connect. Не забудьте также включить директиву --route в главный конфигурационный файл «OpenVPN-ГОСТ», который включает local, чтобы ядро знало, что его надо маршрутизировать к интерфейсу TUN/TAP сервера. Алгоритм выбора внутреннего клиентского IP-адреса работает следующим образом: 1 - Используйте --client-connect script, сгенерированный для статического IP 2 - Используйте файл --client-config dir для статического IP 3 - Используйте присвоение --ifconfig-pool для динамического IP
--ifconfig-ipv6-push ipv6addr/bits ipv6remote	Аналог --ifconfig-push для сетей IPv6

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--iroute network [netmask]	Сгенерировать внутренний маршрут для конкретного клиента. Параметр netmask, если опущен, по умолчанию равен 255.255.255.255. Эту директиву можно использовать, чтобы маршрутизировать фиксированную подсеть с сервера к конкретному клиенту, вне зависимости от того, откуда соединяется клиент. Помните, что вы также должны добавить маршрут к таблице маршрутизации системы (как при пользовании директивой route). Причина, по которой нужны два маршрута, заключается в том, что директива --route маршрутизирует пакет из ядра в «OpenVPN-ГОСТ». Когда пакет туда попадает, директива --iroute маршрутизирует его к указанному клиенту. Эта опция должна быть указана или в конфигурационном файле клиентского экземпляра с использованием --client-config-dir, или в динамически сгенерированном с использованием скрипта --client-connect. Директива --iroute также имеет важное взаимодействие с --push «route...». --iroute в сущности определяет подсеть, принадлежащую конкретному клиенту (назовем этого клиента A). Если вы хотите, чтобы другие клиенты могли достичь подсети A, вы должны использовать --push «route...» вместе с опцией --client-to-client, чтобы добиться этого. Чтобы все клиенты могли видеть подсеть A, «OpenVPN-ГОСТ» должна передать этот маршрут всем клиентам КРОМЕ A, поскольку подсеть уже принадлежит A. «OpenVPN-ГОСТ» достигает этого, не передавая маршрут клиенту, если он совпадает с одним из iroute клиента.
--client-to-client	Поскольку сервер «OpenVPN-ГОСТ» работает со множеством клиентов через один и тот же интерфейс tun или tap, он по сути является маршрутизатором. Флаг --client-to-client говорит «OpenVPN-ГОСТ» маршрутизировать трафик от клиента к клиенту внутри себя, а не передавать весь исходящий от клиентов трафик на интерфейс TUN/TAP. Когда используется эта опция, каждый клиент «увидит» остальных клиентов, которые в настоящее время подключены. В противном случае каждый клиент будет видеть только сервер. Не используйте эту опцию, если вы хотите фильтровать туннельный трафик через брандмауэр, используя собственные правила клиентов.
duplicate-cn	Позволить нескольким клиентам с одним и тем же common name одновременно подключаться. В отсутствие этой опции «OpenVPN-ГОСТ» отсоединит экземпляр клиента при подключении клиента с тем же common name.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--client-connect cmd	Запустить cmd при подключении клиента. cmd содержит путь к скрипту (или исполняемой программе), опционально используется с аргументами. Путь и аргументы могут быть заключены в одинарные или двойные кавычки и/или отделены обратной косой чертой, а также должны быть разделены одним или более пробелом. Команде передаются common name и IP-адрес только что аутентифицировавшегося клиента как переменные среды (см. раздел переменных среды ниже). Команде так же передается путь к только что созданному временному файлу в качестве последнего аргумента (после всех аргументов cmd), чтобы команда использовала его для передачи директив динамически сгенерированного конфигурационного файла в «OpenVPN-ГОСТ». Если скрипт хочет генерировать динамический конфигурационный файл для использования на сервере при подключении клиента, он должен записать его в файл с именем, указанным в качестве последнего аргумента. См. ниже в описании опции --client-config-dir описание опций, которые могут быть правомерно использованы в динамически сгенерированном конфигурационном файле. Обратите внимание, что код возврата script имеет значение. Если script возвращает ненулевой ошибочный статус, клиент будет отключен.
--client-disconnect cmd	Опция похожа на --client-connect, но вызывается при закрытии экземпляра клиента. Не будет вызвана, если для этого экземпляра не были ранее вызваны скрипт --client-connect и подключаемые модули (если определены) с успешным кодом возврата (0). Исключение из этого правила - если скрипт --client-disconnect или подключаемые модули подключены каскадом, и по крайней мере одна из функций опции --client-connect выполнена успешно, затем ВСЕ функции client-disconnect для скриптов и плагинов будут вызваны при удалении объекта экземпляра клиента, даже в тех случаях, когда некоторые из взаимосвязанных функций client-connect вернули ошибочный статус. --client-disconnect команде передается тот же путь, что и команде --client-connect в качестве последнего аргумента (после всех аргументов, указанных в cmd).

--client-config-dir dir	Указать каталог dir для индивидуальных клиентских конфигурационных файлов. После того, как подключающийся клиент аутентифицирован, «OpenVPN-ГОСТ» ищет в этом каталоге файл, название которого совпадает с common name клиента по правилам X.509. Если такой файл существует, он будет открыт и прочитан в поисках клиент-специфичных конфигурационных опций. Если соответствующий файл не найден, «OpenVPN-ГОСТ» попытается открыть и прочитать умолчательный файл под названием DEFAULT, который может быть предоставлен, но не обязателен. Учтите, что конфигурационные файлы должны быть читаемыми для «OpenVPN-ГОСТ» после отказа от прав привилегированного пользователя. Этот файл может указывать фиксированный IP-адрес для данного клиента с помощью --ifconfig-push, как и фиксированные подсети, принадлежащие клиенту, с помощью --iroute. Одно из полезных свойств этой опции – то, что она позволяет удобно создавать, редактировать или удалять клиентские конфигурационные файлы при действующем сервере, без необходимости перезапуска сервера. В клиент-специфичном контексте правомочны следующие опции: --push, --push-reset, --push-remove, --iroute, --ifconfig-push и --config.
--ccd-exclusive	Требовать, в качестве условия аутентификации, чтобы подключающийся клиент имел файл --client-config-dir.
--tmp-dir dir	Указать каталог dir для временных файлов. Этот каталог будет использоваться для передачи временных данных в основной процесс «OpenVPN-ГОСТ». Обратите внимание, что каталог должен быть доступен для записи «OpenVPN-ГОСТ» процессу после отказа от прав привилегированного пользователя. Этот каталог используется в следующих случаях: • скриптами --client-connect для динамической генерации клиент-специфичных конфигурационных файлов. • перехват плагина OPENVPN_PLUGIN_AUTH_USER_PASS_VERIFY для возврата успеха/неудачи через auth_control_file при использовании метода отложенной аутентификации • перехват плагина OPENVPN_PLUGIN_ENABLE_PF для передачи правил фильтрации через pf_file
--hash-size r v	Установить размер таблицы хэшей реальных адресов в r, а виртуальных адресов в v. По умолчанию размеры обеих таблиц ограничены 256 адресами.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--bcast-buffers n	Назначить n буферов для широковещательных датаграмм (по умолчанию 256).
--tcp-queue-limit n	Максимальное количество исходящих пакетов, ставящихся в очередь перед TCP (по умолчанию 64). Когда «OpenVPN-ГОСТ» передает по туннелю данные с устройства TUN/TAP удаленному клиенту по TCP-соединению, возможно, что устройство TUN/TAP может производить данные с большей скоростью, нежели может поддержать TCP-соединение. Когда количество исходящих пакетов, поставленных в очередь перед отправкой на TCP-сокет, достигает этого предела для данного клиентского соединения, «OpenVPN-ГОСТ» начнет игнорировать исходящие пакеты, направленные этому клиенту.

--tcp-nodelay	Этот макрос устанавливает флаг сокета TCP_NODELAY на сервере и передает его соединяющимся клиентам. Флаг TCP_NODELY отключает алгоритм Нагля на TCP-сокетах, заставляя передавать пакеты немедленно с низким временем задержки, а не ждать короткий промежуток времени, чтобы объединить несколько пакетов в один большой пакет. В VPN-приложениях, работающих по TCP, TCP_NODELAY, как правило, является хорошим оптимизатором времени ожидания.
	Макрос расшифровывается следующим образом:
	`if mode server:`
	`socket-flags TCP\_NODELAY`
	`push "socket-flags TCP\_NODELAY"`

--max-clients n	Ограничить сервер n одновременно подключающимися клиентами.
--max-routes-per-client n	Позволить максимум n внутренних маршрутов на одного клиента (по умолчанию 256). Это предназначено для того, чтобы помочь сдержать DoS-атаки, когда аутентифицированный клиент захлестывает сервер пакетами, которые, как кажется, приходят с множества уникальных MAC-адресов, заставляя сервер истощать виртуальную память по мере того, как расширяется его внутренняя таблица маршрутизации. Эту директиву можно использовать в файле --client-config-dir или автоматически генерировать скриптом --client-connect, чтобы заменить глобальное значение для конкретного клиента. Обратите внимание, что эта директива влияет на внутреннюю таблицу маршрутизации «OpenVPN-ГОСТ», а не на таблицу маршрутизации ядра.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--stale-routes-check n [t]	Удалить маршруты, которые не были активны в течение n секунд (т.е. времени устаревания). Эта проверка выполняется каждые t секунд (т.е. интервал проверки). Если t не задано, то по умолчанию используется n. Эта опция помогает уменьшить размер таблицы динамической маршрутизации. См.также --max-routes-per-client
--connect-frec n sec	Позволить максимум N новых соединений в sec секунд от клиентов. Это предназначено для того, чтобы сдержать DoS-атаки, которые захлестывают сервер запросами на соединение, используя сертификаты, которые в конце концов не удается аутентифицировать. Но это несовершенное решение, потому что в реальном сценарии DoS могут также быть отвергнуты правомочные соединения. Для наилучшей защиты от DoS-атак в серверном режиме используйте --proto udp и либо --tls-auth, либо --tls-crypt.

--learn-address cmd	Запустить скрипт или команду оболочки cmd, чтобы проверить правильность клиентских виртуальных адресов или маршрутов. cmd содержит путь к скрипту (или исполняемой программе), опционально используется с аргументами. Путь и аргументы могут быть заключены в одинарные или двойные кавычки и/или отделены обратной косой чертой, а также должны быть разделены одним или более пробелом. cmd может выполняться с тремя параметрами: 1. operation - «add», «update» или «delete», основанный на том, добавляется ли адрес к внутренней таблице маршрутизации «OpenVPN-ГОСТ», модифицируется или удаляется. 2. address - адрес узнается или не узнается. Это может быть адрес IPv4, такой как «198.162.10.14», подсеть IPv4, такая как «198.162.10.0/24», или MAC-адрес Ethernet (когда используется --dev tap) такой как «00:FF:01:02:03:04». 3. common name - Поле common name сертификата, ассоциированного с клиентом, связанным с этим адресом. Присутствует только для операций «add» и «update», но не для «delete». В методах «add» и «update», если скрипт возвращает код ошибки (не ноль), «OpenVPN-ГОСТ» отвергает адрес и не модифицирует свою внутреннюю таблицу маршрутизации. Как правило, скрипт cmd использует информацию, предоставленную выше, чтобы установить подходящие данные брандмауэра на интерфейсе TUN/TAP. Поскольку «OpenVPN-ГОСТ» предоставляет ассоциацию между IP-адресом или MAC-адресом и аутентифицированным полем common name клиента, она позволяет определенному пользователем скрипту конфигурировать политики доступа брандмауэра с учетом высокоуровневого common name клиента, а не низкоуровневых клиентских виртуальных адресов.

--auth-user-pass-verify cmd method	Требует, чтобы клиент предоставил логин и пароль (возможно, в добавление к клиентскому сертификату) для аутентификации. «OpenVPN-ГОСТ» выполнит cmd как команду оболочки, чтобы проверить правильность логина и пароля, предоставленных клиенту. cmd содержит путь к скрипту (или исполняемой программе), опционально используется с аргументами. Путь и аргументы могут быть заключены в одинарные или двойные кавычки и/или отделены обратной косой чертой, а также должны быть разделены одним или более пробелом. Если значение параметра method установлено в «via-env», «OpenVPN-ГОСТ» вызовет script с переменными среды username и pasword, которым в качестве значений присвоены строки логина и пароля, предоставленные клиентом. Имейте в виду, что этот способ небезопасен на некоторых платформах, которые делают среду процесса публично видимой другим непривилегированным процессам. Если значение параметра method установлено в «via-file», «OpenVPN-ГОСТ» запишет логин и пароль в первые две строки временного файла. Наименование файла будет передано в качестве аргумента в script, и файл будет автоматически удален «OpenVPN-ГОСТ» после того, как скрипт пришлет код завершения. Положение временного файла контролируется опцией --tmp-dir, и по умолчанию будет текущим каталогом, если не указано другое. В целях безопасности подумайте над тем, чтобы установить --tmp-dir в несохраняемую среду, такую как /dev/shm (если доступна), чтобы предотвратить запись файла с логином и паролем на жесткий диск. Скрипт должен просмотреть логин и пароль, возвращая успешный код (0) если запрос клиента на аутентификацию следует принять, или код ошибки (1), чтобы отказать клиенту. Эта директива предназначена для того, чтобы дать возможность интерфейсу в стиле подключаемых модулей расширить аутентификационные возможности «OpenVPN-ГОСТ». Чтобы защититься от клиента, передающего злонамеренно созданные строки логина или пароля, логин всегда должен состоять только из следующих символов: букв алфавита и цифр, подчеркивания, дефиса, точки или at-коммерческого (@). Строка пароля может состоять из любых символов, которые можно напечатать, кроме CR и LF. Любые некорректные символы в строках логина или пароля будут превращены в знак подчеркивания.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

	Все определенные пользователем скрипты должны избегать создания уязвимости защиты при работе с этими строками. Никогда не используйте эти строки так, чтобы они могли быть проинтерпретированы интерпретатором оболочки или чтобы специальные символы в них становились частью escape-последовательности.
--auth-gen-token [lifetime]	После успешной аутентификации логин/пароля «OpenVPN-ГОСТ» сервер с этой опцией генерирует временный аутентификационный токен и отправляет его клиенту. При следующих повторных подключениях «OpenVPN-ГОСТ» клиент будет отправлять этот токен вместо пароля пользователя. Сервер на своей стороне выполнит внутреннюю проверку аутентификации токена и НЕ будет выполнять никаких дополнительных проверок аутентификации, направленных на внешние сконфигуренные механизмы аутентификации имени пользователя/пароля. Аргумент lifetime определяет, как долго действует сгенерированный токен. lifetime задается в секундах. Если lifetime не задан или равен 0, то токен никогда не перестает действовать. Эта функция полезна для сред, где настроено использование одноразовых паролей (OTP) как часть проверки подлинности пользователя/пароля, и этот механизм аутентификации не поддерживает никаких аутентификационных токенов.
--opt-verify	Клиенты, которые соединяются с опциями, несовместимыми с опциями сервера, будут отключены. Опции, которые будут сравниваться на совместимость, включают dev-type, link-mtu, tun-mtu, proto, ifconfig, comp-lzo, fragment, keydir, cipher, auth, keysize, secret, no-replay, no-iv, tls-auth, key-method, tls-server, и tls-client. Эта опция требует, чтобы опция --disable-occ НЕ использовалась.

auth-user-pass-optional	Позволить соединение с клиентами, которые не указывают логин и пароль. Как правило, когда указаны опции --auth-user-pass-verify или --management-client-auth (или аутентификационный подключаемый модуль), серверный демон «OpenVPN-ГОСТ» будет требовать у подключающихся клиентов указать логин и пароль. Эта опция делает передачу логина и пароля клиентами опциональной, передавая ответственность определенному пользователем аутентификационному модулю или скрипту, который будет принимать или отвергать клиента на основе других факторов (таких как набор полей сертификата X.509). Когда используется эта опция, и подключающийся клиент не передает логин и пароль, определенный пользователем аутентификационный модуль или скрипт будет рассматривать логин и пароль как пустые строки. Аутентификационный модуль или скрипт ДОЛЖЕН иметь логику, чтобы заметить это состояние и ответить соответственно.
--client-cert-not-required	УСТАРЕЛО Не запрашивать клиентский сертификат, клиент будет аутентифицироваться только с помощью логина и пароля. Имейте в виду, что использование этой директивы менее безопасно, чем требование сертификатов со всех клиентов. Учтите, эта опция была заменена на более гибкую --verify-client-cert. Опция --verify-client-cert none – функциональный эквивалент --client-cert-not-required.

--verify-client-cert none|optional|require	Указать требуется ли от клиента предоставлять валидный сертификат. Возможные варианты: none: сертификат клиента не требуется. Клиент должен аутентифицироваться только с помощью имени пользователя/пароля. Имейте в виду, что использование этой директивы менее безопасно, чем требование сертификатов ото всех клиентов. Если вы используете эту директиву, вся ответственность ложиться на ваш --auth-user-pass-verify script, поэтому помните, что ошибки в вашем скрипте потенциально могут поставить под угрозу безопасность вашей VPN. --verify-client-cert none – функциональный эквивалент --client-cert-not-required optional: клиент может предъявить сертификат, но это необязательно. Когда используется эта директива, вы должны также использовать --auth-user-pass-verify script, чтобы убедиться, что клиент прошел аутентификацию с помощью сертификата или имени пользователя и пароля, а возможно и с помощью и того, и другого. Аналогично, вся ответственность за аутентификацию ложиться на ваш --auth-user-pass-verify script, поэтому помните, что ошибки в вашем скрипте потенциально могут поставить под угрозу безопасность вашей VPN. require: это опция по умолчанию. Клиент должен предъявить сертификат, иначе будет отказано в доступе к VPN. Если вы не используете эту директиву (или используете --verify-client-cert require), но используете --auth-user-pass-verify script, тогда «OpenVPN-ГОСТ» выполнит двойную аутентификацию. Верификация клиентского сертификата И --auth-user-pass-verify script должны завершиться успехом, чтобы клиент был аутентифицирован и допущен к VPN.
--username-as-common-name	Для аутентификации с помощью --auth-user-pass-verify, использовать аутентификационный логин, а не common name из клиентского сертификата в качестве common name.
--no-name-remapping	УСТАРЕЛО --no-name-remapping – альтернативное имя для --compat-names no-remapping опции. Она обеспечивает совместимость с конфигурациями сервера с помощью параметра no-remapping. Обратите внимание: эта опция устарела. Поэтому убедитесь, что вы поддержите новый X.509 формат имени, описанный в --compat-names опции как можно скорей.

--port-share host port [dir]	При работе в TCP-серверном режиме, использовать порт «OpenVPN-ГОСТ» совместно с другим приложением, таким как HTTPS-сервер. Если «OpenVPN-ГОСТ» примет соединение на своем порту, которое использует протокол, отличный от протокола «OpenVPN-ГОСТ», она проксирует соединение к серверу на host:port. В настоящее время работает только с HTTP/HTTPS, хотя теоретически ее в будущем возможно расширить на другие протоколы, такие как ssh. dir указывает опциональный каталог, где будет сгенерирован временный файл для каждого прокси соединения с именем N и содержащий C, где N – исходный IP:port, а C – исходный IP:порт соединения с прокси-сервером. Этот каталог может быть использован прокси-сервером в качестве словаря для определения источника соединения. Каждый сгенеренный файл будет автоматически удален после разрыва прокси-соединения. Не реализована в Windows.

8.4 Клиентский режим

Используйте клиентский режим, подключаясь к серверу «OpenVPN-ГОСТ», который имеет в конфигурации опции --server, --server-bridge или --mode server.

--client	Вспомогательная директива, предназначенная для того, чтобы упростить конфигурацию клиентского режима «OpenVPN-ГОСТ». Эта директива эквивалентна следующему:
	`pull`
	`tls-client`

--pull	Эта опция должна применяться на клиенте, соединяющемся с многоклиентным сервером. Она указывает, что «OpenVPN-ГОСТ» должна принять опции, переданные сервером, если они являются частью правомочного набора передаваемых опций (обратите внимание, что опция --pull подразумевается опцией --client). В частности, --pull позволяет серверу передавать клиенту маршруты, так что не следует использовать --pull и --client в ситуациях, когда вы не доверяете серверу контроль над таблицей маршрутизации клиента.

--pull-filter accept|ignore|reject text	Опции фильтрации, получаемые от сервера, если опция начинается с text. Запускается на клиенте. Флаг accept разрешает опцию, ignore удаляет ее и reject устанавливает ошибку и запускает перезапуск SIGUSER1. Фильтры могут указываться несколько раз, и каждый применяется в том порядке, в каком перечислены. Фильтрация каждой опции оканчивается, как только совпадение найдено. Несовпадшие опции принимаются по умолчанию. Сравнение префикса используется для сопоставления текста и полученной опции, чтобы --pull-filter ignore "route"удалил все переданные опции, начинающиеся с "route которые могут включать в себя, к примеру, route-gateway. Заключите текст в кавычки, чтобы учесть пробелы. --pull-filter accept "route 192.168.1.-pull-filter ignore "route " - удалят все маршруты, которые не начинаются с 192.168.1. Эта опция может быть использована только на клиентах. Обратите внимание, что reject может привести к повторяющемуся циклу сбоя и переподключения, если только не указано несколько устройств и подключение к следующему устройству не окажется успешным. Чтобы молча игнорировать опцию, отправленную сервером, используйте ignore.
-auth-user-pass [up]	Аутентифицироваться на сервере с помощью логина и пароля. up – файл, содержащий логин и пароль в 2 строках. Если строка пароля отсутсвует, то «OpenVPN-ГОСТ» предложит ввести ее. Если параметр up опущен, логин и пароль будут запрашиваться с консоли. Серверная конфигурация должна указать скрипт --auth-user-pass-verify, чтобы проверить логин и пароль, предоставленные клиентом.

--auth-retry type	Контролировать, как «OpenVPN-ГОСТ» отвечает на ошибки проверки логина и пароля, такие как ответ с клиентской стороны на сообщение сервера AUTH_FAILED или неудача проверки пароля закрытого ключа. Как правило, используется, чтобы предотвратить аутентификационные ошибки от фатальности на клиентской стороне, и позволить перезапросить логин и пароль в случае ошибки. Сообщение AUTH_FAILED генерируется сервером, если клиент не проходит аутентификацию --auth-user-pass, или если серверный скрипт --client-connect возвращает статус ошибки, когда клиент пытается подключиться. type может быть одним из: none - Клиент выходит с фатальной ошибкой (умолчание) nointeract - Клиент попытается переподключиться без перезапроса логина и пароля --auth-user-pass. Используйте эту опцию для необслуживаемых клиентов. interact - У клиента будет перезапрошен логин и пароль и/или пароль закрытого ключа перед попыткой переподключения. Обратите внимание, что хотя эта опция не может быть передана, ей можно управлять с интерфейса управления.
--static-challenge t e	Включить статический протокол запроса/ответа, используя текст запроса t с флагом эха, заданным e (0|1). Флаг эха указывает, следует ли или нет повторять ответ пользователя на запрос. См. management-notes.txt из дистрибутива «OpenVPN-ГОСТ» для описания протокола запроса/ответа «OpenVPN-ГОСТ» .
--server-poll-timeout n, --connect-timeout n	Циклически опрашивая возможные удаленные сервера, чтобы подключиться, тратить не больше n секунд на ожидание ответа, прежде чем перейти к следующему серверу. Значение по умолчанию 120с. Этот таймаут включает таймаут прокси-сервера и TCP-соединения.

--explicit-exit-notify [n]	В UDP-клиентском режиме или в режиме точка-в-точку посылать серверу/партнеру сообщение о выходе, если туннель перезапускается или процесс «OpenVPN-ГОСТ» завершается. В клиентском режиме, при выходе/перезапуске, эта опция сообщит серверу, что следует немедленно закрыть его объект экземпляра клиента, а не ждать таймаута. Параметр n (по умолчанию 1) управляет максимальным количеством попыток, которые предпримет клиент, посылая сообщение о выходе. В UDP-серверном режиме посылать команду канала управления RESTART подключенным клиентам. Параметр n (по умолчанию = 1) контролирует поведение клиентов. Когда n = 1, клиент попытается повторно подключиться к тому же серверу, когда n = 2, клиент перейдет к следующему серверу. «OpenVPN-ГОСТ» не будет отправлять никаких уведомлений о выходе, если эта опция не включена.
--allow-recursive-routing	Когда эта опция установлена, «OpenVPN-ГОСТ» не будет сбрасывать входящие tun пакеты с тем же пунктом назначения, что и хост.

8.5 Опции зашифрования канала данных

Эти опции имеют значение и для статического режима, и для режима TLS-договоренного ключа (должны быть совместимы между партнерами)

--secret file [direction]	Обеспечить режим зашифрования статического ключа (не TLS). Использовать ранее полученный обоими партнерами секретный ключ file, который был сгенерирован с помощью опции --genkey. Опциональный параметр direction обеспечивает использование 4 различных ключей (HMAC-отправка, шифр-зашифрование, HMAC-получение, шифр-расшифрование), так что каждое направление течения данных имеет различный набор ключей HMAC и зашифования. Это имеет ряд желательных защитных свойств, включая исключение определенных видов DoS-атак и атак перепроигрывания сообщений. Когда параметр direction опущен, 2 ключа используются в двух направлениях, один для HMAC и один для зашифрования/расшифрования. Параметр direction всегда должен иметь дополняющие друг друга значения на обеих сторонах соединения, т.е. одна сторона должна использовать «1», а другая использовать «0», или обе стороны должны его опустить полностью. Параметр direction требует, чтобы файл file содержал 2048-битный ключ. Режим зашифровки на статическом ключе имеет определенные преимущества, наиболее важное из которых – легкость конфигурирования. Нет сертификатов, удостоверяющих центров или сложных договорных хэндшейков и протоколов. Единственное требование – чтобы у вас был ранее существующий защищенный канал с вашим партнером (такой как ssh), чтобы изначально скопировать ключ. Это требование, вместе с фактом, что ваш ключ никогда не меняется, если вы не сгенерируете вручную новый, делает этот режим несколько менее безопасным, чем режим TLS (см. ниже). Если атакующему удается украсть ваш ключ, все, что было когда-либо зашифровано им, скомпрометировано. Сравните это с идеальными возможностями секретности в TLS-режиме (с использованием обмена ключами Диффи-Хеллмана), где даже если атакующему удалось украсть ваш закрытый ключ, он не получит никакой информации, которая помогла бы ему расшифровать предыдущие сеансы.

	Еще один преимущественный аспект режима зашифрования со статическим ключом тот, что это протокол, в котором отсутствуют хэндшейки, нет никаких отличительных подписей или признаков (таких как заголовок последовательности хэндшейка протокола) которые отмечали бы зашифрованные пакеты как сгенерированные «OpenVPN-ГОСТ». Любой, подслушивающий на проводе, не увидит ничего, кроме случайно выглядящих данных.
--key-direction	Альтернативный способ указать опциональный параметр направления для --tls-auth и --secret опций. Полезно при использовании встроенных файлов (см. раздел о встроенных файлах).
--auth alg	Аутентифицировать пакеты, используя алгоритм дайджеста alg, следует использовать алгориитмы имитовставки magma-mac или kuznyechik-mac, либо алгоритмы HMAC md_gost12_256 или md_gost12_512, для обеспечения совместимости с «OpenVPN-ГОСТ» из состава СКЗИ «МагПро КриптоПакет 3.0» допускается использование алгоритма gost-mac. Чтобы отключить аутентификацию, установите значение параметра alg в none. Если используется режим шифрования MGM (см. опцию --cipher), то при защите передаваемых данных параметр --auth игнорируется, используется аутентификация режима MGM.
--cipher alg	Шифровать пакеты с помощью алгоритма шифрования alg. Следует использовать один из следующих алгоритмов: magma-ctr-acpkm, kuznyechik-ctr-acpkm, magma-mgm или kuznyechik-mgm. Для обеспечения совместимости с «OpenVPN-ГОСТ» из состава СКЗИ «МагПро КриптоПакет 3.0» допускается использование алгоритма gost89.

--ncp-ciphers alglist	Опция задаёт список допустмых алгоритмов шифрования пакетов, разделённых двоеточием, следует указывать один или несколько из следующих пакетов: magma-ctr-acpkm, kuznyechik-ctr-acpkm, magma-mgm или kuznyechik-mgm. Если на обоих концах соединения указана эта опция, для шифрования пакетов будет использован первый из общих алгоритмов этих списков, опция --cipher в этом случае игнорируется. Если опция задана только у одной из сторон соединения, а на второй стороне указана опция --ncp-disable либо используется «OpenVPN-ГОСТ» из состава СКЗИ «МагПро КриптоПакет 3.0», в котором механизм согласования алгоритма шифрования не поддерживается, то будет использован алгоритм шифрования другой стороны, если он либо указан в опции --cipher, либо присутствует в списке --ncp-ciphers
--ncp-disable	Отключает протокол согласования алгоритма шифрования, для шифрования будет использован алгоритм, указанный в опции --cipher.
--keysize n	УСТАРЕЛО Размер ключа шифрования в битах (опционально). Если не указана, по умолчанию равна умолчательному значению, специфичному для данного шифра. Опция --show-chipers (см. ниже) показывает все доступные шифры OpenSSL, их умолчательные размеры ключей, и можно ли изменить размер ключа. Меняйте умолчательный размер ключа с осторожностью. Многие шифры не были широко криптоанализированы с нестандартными длинами ключей, и более длинный ключ может не предоставлять реальной гарантии большей безопасности и даже уменьшить степень защиты.
--prng alg [nsl]	(Дополнительная) для PRNG (генератора псевдослучайных чисел) использовать алгоритм дайджеста alg (по умолчанию sha1) и установить nsl (по умолчанию 16) в размер в байтах длины секрета nonce (между 16 и 64). Установите значение параметра alg в none, чтобы отключить PRGN и использовать вместо нее функцию OpenSSL RAND_bytes для всех требований псевдослучайных чисел в «OpenVPN-ГОСТ».
--engine [engine-name]	Обеспечить функциональность криптографического модуля engine OpenSSL, основанную на аппаратных возможностях. Если указан параметр engine-name, использовать определенный криптографический модуль engine. Используйте опцию --show-engines, чтобы получить список криптографических модулей engine, которые поддерживаются OpenSSL.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--no-replay	УСТАРЕЛО (Дополнительная) Отключить защиту «OpenVPN-ГОСТ» против атак перепроигрывания. Не используйте эту опцию, если вы не готовы заплатить уменьшением степени защиты за увеличение эффективности. «OpenVPN-ГОСТ» предоставляет защиту от перепроигрывания датаграмм по умолчанию. Защита от перепроигрывания выполняется помечанием каждой исходящей датаграммы идентификатором, который гарантированно уникален для используемого ключа. Партнер, который получает датаграмму, проверяет уникальность идентификатора. Если этот идентификатор уже был получен в предыдущей датаграмме, «OpenVPN-ГОСТ» проигнорирует пакет. Защита от перепроигрывания важна для того, чтобы отразить такие атаки, как атака переполнения SYN, где атакующий слушает на проводе, перехватывает пакет SYN (идентифицируя его по контексту, в котором он встречается по отношению к другим пакетам), потом передает получающему партнеру множество копий этого пакета. Защита от перепроигрывания в «OpenVPN-ГОСТ» реализована несколько различными способами, в зависимости от режима управления ключами, который вы выбрали. В режиме статического ключа или при использовании шифров CFB или OFB, «OpenVPN-ГОСТ» использует 64-битный уникальный идентификатор, который сочетает метку времени с увеличивающимися последовательными номерами. При использовании режима TLS для обмена ключами и шифра CBC, «OpenVPN-ГОСТ» использует только 32-битный последовательный номер без метки времени, поскольку «OpenVPN-ГОСТ» может гарантировать уникальность этой величины для каждого ключа. Как в IPSec, если последовательный номер приближается к нулю, «OpenVPN-ГОСТ» запускает новый обмен ключами. Чтобы производить проверку на перепроигрывание, «OpenVPN-ГОСТ» использует алгоритм «скользящего окна», используемый IPSec.

--replay-window n [t]	Использовать скользящее окно для защиты от перепроигрывания размера n и временное окно в t секунд. По умолчанию n равно 64 (умолчание IPSec), а t равно 16 секундам. Эта опция актуальна только в режиме UDP, т.е. когда или указана опция --proto udp, или вообще не указана опция --proto. Когда «OpenVPN-ГОСТ» туннелирует IP-пакеты через UDP, существует возможность, что пакеты могут быть потеряны или доставлены в неправильном порядке. Поскольку «OpenVPN-ГОСТ», как и IPSec, эмулирует слой физической сети, она примет последовательность пакетов, пришедших в неправильном порядке, и доставит такие пакеты в том же порядке, в каком они были получены, в стек протоколов TCP/IP, если они удовлетворяют некоторым ограничениям. 1. Пакет не может быть перепроигрыванием (если только не указана опция --no-replay, которая отключает защиту от перепроигрывания) 2. Если пакет прибывает в неправильном порядке, он может быть принят только если разница между его последовательным номером и самым большим уже принятым последовательным номером меньше n. 3. Если пакет прибывает в неправильном порядке, он будет принят только если он прибывает не позже чем t секунд после любого пакета, содержащего более высокий последовательный номер. Если вы используете сетевую связь с большой магистралью (что означает широкую полосу пропускания и большой период ожидания), вы можете захотеть использовать большие величины n. Особенно часто этого требует спутниковая связь. Если вы запускаете «OpenVPN-ГОСТ» с --verb 4, вы увидите сообщение «Replay-window backtrack occurred [x]» каждый раз, когда отступление максимального последовательного номера увеличивается. Это может быть использовано для калибровки n. Существуют некоторые противоречия по поводу подходящего способа обращения с изменением порядка пакетов в слое безопасности. А именно, до какой степени должен слой безопасности защищать инкапсулированный протокол от атак, которые притворяются разновидностями обычной потери и перестановки пакетов, которые случаются в IP-сетях? Подход IPSec и «OpenVPN-ГОСТ» состоит в том, чтобы разрешить перестановку пакетов внутри определенного фиксированного окна последовательных номеров.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

	«OpenVPN-ГОСТ» дополняет модель IPSec, ограничивая размер окна во времени, как и в пространстве последовательности. «OpenVPN-ГОСТ» также добавляет TCP-транспорт как опцию (не предлагаемую IPSec), в этом случае «OpenVPN-ГОСТ» может занять очень строгую позицию по отношению к удалению и перестановке сообщений: не позволять этого. Поскольку TCP гарантирует надежность, любая потеря или перестановка пакетов может быть сочтена атакой. В этом смысле можно возразить, что туннельный TCP-транспорт предпочтителен при туннелировании не-IP или UDP-протоколов приложений, которые могут быть уязвимы по отношению к атаке удаления или перестановки пакетов, которая не удается в пределах нормальных операционных параметров IP-сетей. Поэтому утверждается, что никогда не следует туннелировать не-IP-протокол или UPD-протокол приложения через UPD, если протокол может быть уязвим по отношению к атаке удаления или перестановки пакетов, которая не удается в пределах нормальных операционных параметров того, что следует ожидать от физического IP-слоя. Проблема легко решается простым использованием TCP в качестве транспортного слоя VPN.
--mute-replay-warnings	Не выводить предупреждения о перепроигрывании, которые часто бывают ложной тревогой в WiFi-сетях. Эта опция сохраняет безопасность кода защиты от перепроигрывания без вывода множества сообщений о дублирующихся пакетах.

--replay-persist file	Сохранять состояние защиты от перепроигрывания между сеансами, используя файл file для сохранения и перезагрузки состояния. Эта опция усиливает защиту против атак перепроигрывания, особенно когда вы используете «OpenVPN-ГОСТ» в динамическом контексте (например, с опцией -inetd), когда сеансы «OpenVPN-ГОСТ» часто начинаются и останавливаются. Эта опция сохранит на диске копию текущего состояния защиты от перепроигрывания (т.е. самую последнюю метку времени пакета и последовательный номер, полученные от удаленного партнера), так что если сеанс «OpenVPN-ГОСТ» остановлен и начат снова, он отвергнет любое переигрывание пакетов, которые были уже получены в предыдущей сессии. Эта опция имеет смысл только в том случае, когда защита от перепроигрывания включена (по умолчанию) и вы используете или --secret (режим общих секретных ключей) или TLS-режим с --tls-auth.
--no-iv	УСТАРЕЛА (Дополнительная) Отключить использование IV (вектора инициализации шифра). Не используйте эту опцию, если вы не готовы заплатить меньшей степенью защиты за большую эффективность. «OpenVPN-ГОСТ» использует IV по умолчанию и требует его для режимов CFB и OFB (которые совершенно не защищены без него). Использование IV важно для безопасности, когда несколько сообщений зашифровываются/расшифровываются с использованием одного и того же ключа. IV реализован по-разному в зависимости от используемого режима шифрования. В режиме CBC, «OpenVPN-ГОСТ» использует псевдослучайный IV для каждого пакета. В режиме CFB/OFB «OpenVPN-ГОСТ» использует в качестве IV уникальный последовательный номер и метку времени. На самом деле, в режиме CFB/OFB «OpenVPN-ГОСТ» использует оптимизацию датаграмм, экономящую место, которая использует уникальный идентификатор для защиты от перепроигрывания датаграмм в качестве IV.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--test-crypto	Выполнить самотестирование криптографических опций «OpenVPN-ГОСТ», зашифровывая и расшифровывая тестовые пакеты с использованием опций зашифрования каналов данных, указанных выше. Эта опция не требует функционирующего партнера, поэтому может быть указана без --dev и --remote. Типичным использованием --test-crypto будет что-то подобное следующему: openvpn-gost --test-crypto --secret key или openvpn-gost --test-crypto --secret key --verb 9 Эта опция очень полезна для тестирования «OpenVPN-ГОСТ» после того, как она была портирована на новую платформу, или чтобы изолировать проблемы в компиляторе, криптобиблиотеке OpenSSL или криптографическом коде «OpenVPN-ГОСТ». Поскольку это режим самотестирования, проблемы с зашифрованием и аутентификацей могут быть решены независимо от проблем с сетью и туннелем.

8.6 Опции протокола TLS

Протокол TLS используется для организации и защиты контрольного канала, по которому стороны обмениваются, в частности, ключами защиты данных.

Режим TLS - самый мощный криптографический режим «OpenVPN-ГОСТ» как с точки зрения безопасности, так и гибкости. Режим TLS работает, устанавливая каналы управления и данных, которые мультиплексируются через один порт TCP/UDP. «OpenVPN-ГОСТ» инициирует сеанс TLS по каналу управления и использует его для обмена ключами шифрования и HMAC для защиты канала данных. TLS режим использует высокий уровень надежности поверх TCP/UDP соединения для всех коммуникаций канала управления, тогда как канал данных, по которому идут зашифрованные туннельные данные, перенаправляется без какого-либо посредничества. В результате вы получаете достоинства обоих подходов: быстрый канал данных, который пересылается через TCP/UDP с накладными расходами только на шифрование, расшифрование и функции HMAC, и канал управления, который обеспечивает все функции безопасности TLS, включая аутентификацию, основанную на сертификатах, и прямую секретность Диффи-Хеллмана.

Чтобы установить TLS-канал, каждый партнер, который запускает «OpenVPN-ГОСТ», должен иметь собственную локальную пару сертификат/ключ (--cert и --key), подписанную на корневом сертификате, указанном в --ca.

Когда два партнера подключаются друг к другу, каждый представляет другому свой локальный сертификат. Затем каждый партнер проверяет, что его партнер предоставил сертификат, подписанный на корневом сертификате, указанном в --ca. Обратите внимание, что сертификат партнёра A должен быть подписан на корневом сертификате, указанном в параметре --ca партнёра B, и наоборот.

Если эта проверка успешна для обоих партнеров, то будет успешно установлено TLS-соединение, оба партнера обменяются временными сеансовыми ключами, и туннель начнет

передавать данные.

Дистрибутив «OpenVPN-ГОСТ» содержит набор скриптов easy-gost для управления сертификатами и ключами.

--tls-server	Включить TLS и принять на себя роль сервера в TLS-соединении.
--tls-client	Включить TLS и принять на себя роль клиента в TLS-соединении.
--tls-version-min version ['or-highest']	Запрещает использовать версии протокола TLS ниже указанной.
--tls-version-max version	Запрещает использовать версии протокола TLS выше указанной.
--ca file	Файл сертификата удостоверяющего центра (CA) в формате .pem, также именуемый корневым сертификатом. Этот файл может включать несколько сертификатов в формате .pem, конкатенированных вместе. Вы можете сконструировать свой собственный сертификат и закрытый ключ удостоверяющего центра, используя такую команду, как: `openssl req -nodes -new` Затем отредактируйте ваш файл openssl.cnf и отредактируйте переменную certificate так, чтобы она указывала на ваш новый корневой сертификат ca.crt. Только для тестовых целей, дистрибутив «OpenVPN-ГОСТ» включает образец сертификата удостоверяющего центра (ca.crt). Конечно, вам никогда не следует использовать тестовые сертификаты и тестовые ключи из дистрибутива «OpenVPN-ГОСТ» в рабочей обстановке, поскольку вследствие того факта, что они включены в дистрибутив «OpenVPN-ГОСТ», они не предоставляют никакой защиты.
--capath dir	Каталог, содержащий доверенные сертификаты (CA и Список отозванных сертификатов). Когда используется опция --capath, также должны быть предоставлены корректные CRL для удостоверяющих центров. Ожидается, что удостоверяющий центр в каталоге capath будет называться <hash>.r<n>. Для получения дополнительной информации см. опцию --CApath команды openssl verify и опцию -hash команд openssl x509 и openssl crl.

--cert file	Файл сертификата своего открытого ключа в формате .pem. Каждая установка «OpenVPN-ГОСТ» должна иметь свои собственные файлы сертификата и закрытого ключа. Сертификат должен быть подписан ключом удостоверяющего центра, чей открытый ключ находится в файле, указанном в опции --ca другой стороны. Вы легко можете получить необзодимые сертификаты в удостоверяющем центре либо создать свою собственную ключевую инфраструктуру, используя набор скриптов easy-gost, входящий в комплект поставки СКЗИ «МагПро КриптоПакет».
--extra-certs file	Укажите файл, содержащий один или несколько PEM сертификатов (объединенных вместе), которые завершают локальную цепочку сертификатов. Эта опция полезна для «разделенных» удостоверяющих центров, где удостоверяющий центр для серверных сертификатов отличается от удостоверяющего центра для клиентских сертификатов. Размещение сертификатов в этом файле позволяет им быть использованными для завершения локальной цепочки сертификатов, не доверяя им проверку сертификата, предоставленного партнером, как это было бы в случае, если бы сертификаты были помещены в ca file.
--key file	Файл своего закрытого ключа в формате .pem. Используйте закрытый ключ, соответствующий сертификату, указанному в оции --cert.
--pkcs12 file	Файл формата PKCS#12, содержащий локальный закрытый ключ, локальный сертификат и корневой сертификат УЦ. Эту опцию можно использовать вместо --ca, --cert и --key.
--verify-hash hash [alg]	Указать хэш-вектор SHA1 или SHA256 для сертификата уровня 1. Сертификат уровня 1 – это сертификат CA, который подписал сертификат другой стороны. Когда принимается соединение от партнера, отпечаток сертификата уровня 1 должен совпасть с hash, иначе сертификат не пройдет верификацию. Hash указывается в формате XX:XX:. . . Например: AD:B0:95:D8:09:C8:36:45:12:A9:89:C8:90:09:CB:13:72:A6:AD:16 Флаг alg может иметь значения либо SHA1, либо SHA256. Если не указан, по умолчанию равен SHA1 .

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--cryptoapicert select-string	Загрузить сертификат и закрытый ключ из системного хранилища сертификатов Windows (только для Windows/OpenSSL). Используйте эту опцию вместо --cert и --key. Это позволяет использовать любую смарт-карту, поддерживаемую Windows, а также любой сертификат, находящийся в хранилище сертификатов, где у вас есть доступ к закрытому ключу. Этот вариант был протестирован с несколькими различными смарт-картами (GemSAFE, Cryptoflex и eID почтового отделения Швеции) на стороне клиента, а также с импортированным сертификатом программного обеспечения PKCS12 на стороне сервера. Чтобы выбрать сертификат на основе поиска подстроки в теме сертификата: cryptoapicert "SUBJ:Peter Runestig" Чтобы выбрать сертификат на основе отпечатка сертификата: cryptoapicert "THUMB:f6 49 24 41 01 b4 ..." Шестнадцатеричная строка отпечатка может быть легко скопирована и вставлена из хранилища сертификатов Windows .
--tls-cipher list	Список разрешенных криптонаборов для TLS 1.2, разделенных двоеточиями. GOST2012-KUZNYECHIK-KUZNYECHIKOMAC, для обеспечения совместимости с «OpenVPN-ГОСТ» из состава СКЗИ «МагПро КриптоПакет 3.0» допустимо использовать криптонабор GOST2012-GOST8912-GOST8912.
--tls-ciphersuites list	Список разрешенных криптонаборов для TLS 1.3, разделенных двоеточиями. Следует использовать один или несколько криптонаборов из следующего списка: TLS_GOSTR341112_256_WITH_MAGMA_MGM_S, TLS_GOSTR341112_256_WITH_MAGMA_MGM_L, TLS_GOSTR341112_256_WITH_KUZNYECHIK_MGM_S и TLS_GOSTR341112_256_WITH_KUZNYECHIK_MGM_L
--tls-groups list	Список разделенных двоеточиями разрешенных групп элиптических кривых, используемых в протоколе TLS 1.3. Следует использовать одну или несколько групп из следующего списка: GC256A:GC256B:GC256C:GC256D:GC512A:GC512B:GC512C

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--tls-timeout n	Установить таймаут повторной передачи пакетов на контрольном канале TLS, если никакого отзыва от удаленного партнера не приходит в течение n секунд (по умолчанию n=2). Когда «OpenVPN-ГОСТ» отправляет контрольный пакет своему партнеру, она ожидает получение отзыва в течение n секунд, она еще раз передаст пакет в соответствии с TCP-подобным алгоритмом экспоненциального возврата. Этот параметр применяется только к пакетам контрольного канала. На пакеты канала данных (которые несут зашифрованные туннельные данные) никогда не приходит отзыв, и они никогда не передаются повторно, потому что сетевые протоколы более высокого уровня, работающие на верху туннеля, такие как TCP, ожидают, что эта роль оставлена им.
--reneg-bytes n	Пересогласовать ключ канала данных после того, как n байтов отправлены или получены (по умолчанию отключено). «OpenVPN-ГОСТ» позволяет выразить срок жизни ключа как количество зашифрованных/расшифрованных байтов, количество пакетов или секунд. Пересогласование ключа будет произведено насильно, если любой из партнеров соответствует любому из этих трех критериев. При использовании шифров с размером блока шифра менее 128 бит для опции --reneg-bytes по умолчанию устанавливается значение 64 МБ, если только оно не отключено явным образом путем установки значения 0, но это НАСТОЯТЕЛЬНО НЕ РЕКОМЕНДУЕТСЯ, поскольку оно предназначено для дополнительной защиты против вектора атаки SWEET32. Для получения дополнительной информации см. опцию --cipher .
--reneg-pkts n	Пересогласовать ключ канала данных после того, как n пакетов отправлены и получены (по умолчанию отключено).

--reneg-sec n	Пересогласовать ключ канала данных после n секунд (по умолчанию 3600). При использовании двухфакторной аутентификации обратите внимание, что эта умолчательная величина может привести к тому, что конечного пользователя будут принуждать переаутентифицироваться раз в час. Также помните, что эта опция может быть использована и на клиенте, и на сервере, и пересогласование запустит тот из партнеров, у кого указана меньшая величина. Частая ошибка – установить --reneg-sec в большую величину на клиенте или на сервере, в то время как другая сторона по-прежнему использует умолчательную величину в 3600 секунд, что значит, что пересогласование по-прежнему будет происходить каждые 3600 секунд. Решение состоит в том, чтобы увеличить величину --reneg-sec и на клиенте, и на сервере, или установить ее в 0 на одной из сторон соединения (чтобы отключить), и в выбранную величину на другой стороне.
--hand-window n	Окно хэндшейка – основанный на TLS обмен ключами должен завершиться в течение n секунд после инициации хэндшейка любым из партнеров (по умолчанию 60 секунд). Если хэндшейк не завершается успешно, мы попытаемся переустановить наше соединение с нашим партнером и попытаемся снова. Даже в случае провала хэндшейка мы будем использовать наш завершающий время действия ключ до --tran-window секунд, чтобы поддержать непрерывность передачи туннельных данных.
tran-window n	Окно передачи – наш старый ключ может жить столько секунд после того, как начинается новое пересогласование ключа. Эта возможность позволяет корректно перейти от старого ключа к новому и удаляет последовательность пересогласования ключа с критичного пути форвардинга туннельных данных.
--single-session	После исходного соединения с удаленным партнером, запретить все новые соединения. Использование этой опции означает, что удаленный партнер не может подключиться, отключиться и снова подключиться. Если демон перегружается от сигнала или опции --ping-restart, он позволит одно новое соединение. Опцию --single-session можно использовать с опциями --ping-exit или --inactive, чтобы создать один динамический сеанс, который отключится по завершении.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--tls-auth file [direction]	Добавить дополнительный слой аутентификации HMAC на верх контрольного канала TLS для защиты от DoS-атак. В двух словах, --tls-auth подключает своего рода «брандмауэр HMAC» на TCP/UDP-порту «OpenVPN-ГОСТ», где пакеты контрольного канала TLS, несущие некорректную имитовставку, могут быть немедленно проигнорированы без ответа. Для выработки имитовставки используется алгоритм, указанный в параметре --auth. **Внимание!** Алгоритмы magma-mac, kuznyechik-mac и gost-mac не могут быть использованы для выработки имитовставки пакетов контрольного канала, поэтому если вы хотите использовать опцию --tls-auth, в параметре --auth необходимо указать md_gost12_256 или md_gost12_512. Параметр file (необходим) – ключевой файл, который может быть в одном из двух форматов. 1. Файл статического ключа «OpenVPN-ГОСТ», созданный опцией --genkey (необходим, если используется параметр direction). 2. Файл парольной фразы в свободной форме. В этом случае ключ HMAC получается при взятии безопасного хэша этого файла, подобно командам md5sum и sha1sum. «OpenVPN-ГОСТ» сначала пробует формат (1), и если файл не удается прочесть как файл статического ключа, будет использоваться формат (2). См. в опции --secret информацию об опциональном параметре direction. Опция --tls-auth рекомендуется, когда вы запускаете «OpenVPN-ГОСТ» в режиме, в котором она слушает пакеты с любого IP-адреса, например когда опция --remote не указана или указана с --float. Обоснование этой опции таково. TLS требует многопакетного обмена, прежде чем она сможет аутентифицировать партнера. В течение этого времени перед аутентификацией «OpenVPN-ГОСТ» назначает ресурсы (память и CPU) этому потенциальному партнеру. Потенциальный партнер также открывает много частей «OpenVPN-ГОСТ» и библиотеки OpenSSL пакетам, которые он посылает. Сегодня наиболее удачные сетевые атаки пытаются либо использовать ошибки в программах (такие как атаки переполнения буфера) или вынудить программу поглотить столько ресурсов, что ее становится невозможно использовать. Конечно, первая линия защиты - всегда предоставлять чистый, хорошо просмотренный код. «OpenVPN-ГОСТ» была написана с предотвращением атак переполнения буфера в качестве первого приоритета. Но как показала история, многие из наиболее широко используемых сетевых приложений время от времени падали жертвами атак переполнения буфера.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

	Поэтому в качестве второй линии защиты «OpenVPN-ГОСТ» предлагает этот специальный слой аутентификации на верху контрольного канала TLS, так чтобы каждый пакет в контрольном канале аутентифицировался подписью HMAC и уникальным ID для защиты от атак перепроигрывания. Эта подпись также должна помочь защититься от DoS-атак (атак отказа в обслуживании). Важным ключевым правилом в уменьшении уязвимости к DoS-атакам является минимизировать количество ресурсов, которые может потребить потенциальный, но еще не аутентифицированный клиент. Опция --tls-auth делает это, подписывая каждый пакет контрольного канала TLS подписью HMAC, включая пакеты, которые отправляются до того, как уровень TLS имел шанс аутентифицировать партнера. Результат таков, что пакеты без корректной подписи могут быть проигнорированы сразу после получения, прежде чем у них будет шанс поглотить дополнительные системные ресурсы, такие как инициализация TLS-хэндшейка. Опция --tls-auth может быть усилена добавлением опции --replay-persist, которая сохранит статус защиты «OpenVPN-ГОСТ» от атак перепроигрывания в файл, чтобы он не терялся при перезапусках. Следует подчеркнуть, что эта возможность опциональна и что файл пассфразы/ключа, используемый с опцией --tls-auth, не дает партнеру ничего, кроме возможности инициализировать хэндшейк TLS. Он не используется для зашифрования или аутентификации каких-либо туннельных данных. Вместо этого используйте --tls-crypt, если вы хотите использовать файл ключа не только для аутентификации, но и для шифрования канала управления TLS.

--tls-crypt keyfile	Зашифровать и аутентифицировать все пакеты канала управления с помощью ключа из файла ключей. (Подробнее см. --tls-auth) Шифрование (и аутентификация) пакетов канала управления: • обеспечивает большую конфиденциальность, скрывая сертификат, используемый для TLS-соединение, • затрудняет идентификацию трафика «OpenVPN-ГОСТ» как такового, • обеспечивает дешевую постквантовую защиту от злоумышленников, которые никогда не узнают предварительно разделенный ключ (т.е. никакой прямой секретности). В отличие от --tls-auth, --tls-crypt НЕ требует от пользователя установки --key-direction. Вопросы безопасности Все партнеры используют один и тот же предварительный групповой ключ --tls-crypt для аутентификации и шифрования сообщений канала управления. Чтобы убедиться, что коллизии синхропосылок остаются маловероятными, этот ключ не следует использовать для шифрования более 2^{48} сообщений канала управления между клиентом и сервером или 2^{48} между сервером и клиентом. Типичное начальное согласование составляет около 10 пакетов в каждом направлении. Предполагая, что и первоначальное согласование, и повторное согласование составляют не более 2^{16} (65536) пакетов (для консервативности), а повторное согласование происходит каждую минуту для каждого пользователя (24/7), это ограничивает время жизни ключа tls-crypt до 8171 года, разделенное на количество пользователей. Таким образом, установка с 1000 пользователями должна менять ключ не реже одного раза в восемь лет. (И установка с 8000 пользователей - каждый год.) Если возникнут коллизии синхропосылок, это может привести к тому, что безопасность --tls-crypt деградирует до того же уровня безопасности, что и при использовании --tls-auth. То есть канал управления все еще выигрывает от дополнительной защиты против активных атак посредника и DoS-атак, но больше не может обеспечивать дополнительную конфиденциальность и постквантовую безопасность сверх того, что предлагает сам TLS.

--askpass [file]	Получить пароль к сертификату с консоли или из файла [file], прежде чем мы демонизируемся. Вы можете захотеть дополнительно защитить ваш закрытый ключ паролем. Конечно, это означает, что каждый раз, когда «OpenVPN-ГОСТ» читает ключ, вы должны ввести пароль. Однако если вы запускате «OpenVPN-ГОСТ» в режиме демонизации (см. опцию --daemon), чтение закрытого ключа происходит после того, как процесс стал демоном, и ввод пароля оказывается невозможен. Опция --askpass решает эту проблему, при её наличии «OpenVPN-ГОСТ» запросит у вас пароль прежде, чем демонизируется, а затем использует введённый пароль при чтении ключа. Если указан файл file, читать пароль из первой строки этого файла. Помните, что хранение вашего пароля в файле до определенной степени понижает дополнительную защиту, которую предоставляет использование зашифрованного ключа.
--askpin	Получить pin-код доступа к токену, прежде чем мы демонизируемся. Если вы используете закрытый ключ на токене, каждый раз, когда «OpenVPN-ГОСТ» обращается к токену, вы должны вводить pin-код. Однако если вы запускате «OpenVPN-ГОСТ» в режиме демонизации (см. опцию --daemon), чтение закрытого ключа происходит после того, как процесс стал демоном, и ввод pin-кода оказывается невозможен. Опция --askpin решает эту проблему, при её наличии «OpenVPN-ГОСТ» запросит у вас pin-код прежде, чем демонизируется, а затем использует введённый pin-код для доступа к токену.
--auth-nocache	Не кэшировать логины и пароли опции --askpass или --auth-user-pass в виртуальной памяти. Если эта директива указана, она заставит «OpenVPN-ГОСТ» немедленно забыть вводы логина и пароля, как только они использованы. В результате, когда «OpenVPN-ГОСТ» нужны логин и пароль, она запросит их ввод со стандартного ввода, что может произойти несколько раз за время сеанса «OpenVPN-ГОСТ». Эта директива не влияет на логин и пароль опции --http-proxy. Он всегда кэшируется.

--auth-token token	Эту опцию нельзя использовать напрямую в конфигурационных файлах, она запускается из скрипта --client-connect или --plugin, который подключается к вызовам OPENVPN_PLUGIN_CLIENT_CONNECT or OPENVPN_PLUGIN_CLIENT_CONNECT_V2. Эта опция дает возможность заменить пароль клиента на аутентификационный токен в течение всего времени существования «OpenVPN-ГОСТ» клиента. Каждый раз, когда происходит переподключение и вызывается --auth-user-pass-verify script или --plugin, использующий OPENVPN_PLUGIN_AUTH_USER_PASS_VERIFY, токен будет передаваться в качестве пароля вместо пароля, предоставленного пользователем. Аутентификационный токен может быть сброшен только при полном переподключении, когда сервер может отправить новые опции клиенту. Введенный пользователем пароль никогда не будет сохраняться после установки аутентификационного токена. Если сервер «OpenVPN-ГОСТ» отклоняет аутентификационный токен, клиент получит AUTH_FAIL и будет отключен. Целью этого является подключение двухфакторных методов аутентификации, таких как HOTP или TOTP, которые будут использоваться без необходимости получать новый OTP код каждый раз, когда согласовывается повторное подключение. Другой случай использования - это кэширование аутентификационных данных на клиенте без необходимости кэширования пароля пользователя в памяти в течение всей сессии подключения. Чтобы использовать эту функцию, --client-connect script или --plugin должны поместить "auth-token UNIQUE_TOKEN_VALUE"в файл/буфер для данных динамической конфигурации. Это заставит сервер в дальнейшем передавать это значение клиенту, который заменит локальный пароль на UNIQUE_TOKEN_VALUE. Более новые клиенты будут возвращаться к исходному методу обработки пароля после неудачной аутентификации. Старые клиенты продолжат использовать значение токена и реагировать в соответствии с --auth-retry.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--tls-verify cmd	Выполнить команду оболочки cmd, чтобы проверить имя X509 name текущего TLS-соединения, которое уже прошло все остальные сертификационные тесты (кроме отзыва через директиву --crl-verify; тест на отзыв производится после теста --tls-verify). cmd содержит путь к скрипту (или исполняемой программе), опционально используется с аргументами. Путь и аргументы могут быть заключены в одинарные или двойные кавычки и/или отделены обратной косой чертой, а также должны быть разделены одним или более пробелом. cmd должна вернуть 0, чтобы позволить TLS-хэндшейку продолжаться, или 1 в случае неудачи. При выполнении cmd два аргумента добавляются после всех аргументов, указанных в cmd следующим образом: cmd certificate_depth subject Эти аргументы являются, соответственно, текущей глубиной сертификата и именем X509 узла. Эта возможность полезна, если партнер, которому вы хотите доверять, имеет сертификат, который был подписан удостоверяющим центром, который также подписал много других сертификатов, если вы не хотите доверять им всем, а предпочитаете избирательно относиться к тому, какой сертификат партнера вы примете. Эта функциональность позволяет вам написать скрипт, который будет проверять имя X509 name сертификата и решит, следует его принимать или нет. В качестве простого скрипта на perl, который проверяет поле common name сертификата, см. файл verify-cn в дистрибутиве «OpenVPN-ГОСТ». См. в разделе «Переменные среды» ниже информацию о дополнительных параметрах, передаваемых как переменные среды. Обратите внимание, что команда cmd может быть командой оболочки с несколькими аргументами, в этом случае все аргументы, созданные «OpenVPN-ГОСТ», будут добавлены в конец строки cmd, чтобы создать строку, которая будет передана в скрипт.
--tls-export-cert directory	Сохранять сертификаты, которые клиенты используют при подключении, в указанном каталоге. Выполняется перед вызовом --tls-verify. Сертификаты будут использовать временное имя и будут удалены, когда скрипт tls-verify завершит работу. Имя файла, использующееся для сертификата, доступно через переменную среды peer_cert.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--x509-username-field [ext:]fieldname	Поле в subject сертификата X.509, которое будет использоваться в качестве имени пользователя (по умолчанию = CN). Как правило эта опция указывается с именем поля одним из следующих вариантов: --x509-username-field emailAddress --x509-username-field ext:subjectAltName Первый пример использует значение "emailAddress"в качестве имени пользователя в поле Subject сертификата. Во втором примере префикс ext: для обозначения того, что будет осуществлен поиск поля rfc822Name (email) в fieldname "subjectAltName"расширения X.509, которое будет использоваться в качестве имени пользователя. В случае, когда найдено несколько адресов электронной почты в ext:fieldname, выбирается последний. Когда используется эта опция, --verify-x509-name опция будет сопоставляться с указанным fieldname вместо Common Name. Поддерживаются только subjectAltName и issuerAltName расширения X.509. Обратите внимание: у этой опции есть функция, которая преобразует все fieldname в нижнем регистре в символы верхнего регистра, к примеру, ou -> OU. fieldname в смешанном регистре или с префиксом ext: будет оставлено как есть. Эта функция повышения регистра устарела и будет удалена в дальнейшем.
--verify-x509-name name type	Принимать соединения только в том случае, если имя узла X.509 равно name. Удаленный хост также должен пройти все остальные проверки верификации. Какое X.509 имя сравнивается с name, зависит от значения type, которое может быть “subject”, чтобы соответствовать полю subject DN (по умолчанию), либо “name”, чтобы соответствовать subject RDN, либо "name-prefix соответствующий префиксу subject RDN. Какой RDN берется для проверки имени, зависит от опции --x509-username-field, но по умолчанию используется общее имя (CN), т.е. сертификат с subject DN "C=RU, ST=NA, L=Chita, CN=Server-1"будет соответствовать и --verify-x509-name ’C=RU, ST=NA, L=Chita, CN=Server-1’ и --verify-x509-name Server-1 name

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

или же вы можете использовать --verify-x509-name Server-name-prefix если хотите, чтобы клиент принимал соединение только от "Server-1 "Server-2"и т.п. --verify-x509-name – полезная замена опции --tls-verify для проверки удаленного хоста, потому что --verify-x509-name работает в --chroot окружении без каких-либо зависимостей. Использование name prefix полезная альтернатива для управления CRL (Certificate Revocation List) на клиенте, поскольку она позволяет клиенту отклонять все сертификаты кроме тех, которые связаны с назначенными серверами. Примечание. Применяйте name prefix только в том случае, если вы используете «OpenVPN-ГОСТ» сертификат удостоверяющего центра (CA), который находится под вашим контролем. Никогда не используйте эту опцию с name prefix, когда сертификаты клиента подписаны сторонней организацией, к примеру, коммерческим удостоверяющим веб центром (Web CA).	

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--x509-track attribute	Сохранить значение атрибута X509 узла в среде для использования плагинами и интерфейсом управления. Добавьте ‘+’, чтобы сохранить значения из полной цепочки сертификатов. Значения будут закодированы в виде X509_<depth>_<attribute>=<value>. Несколько --x509-track опций могут быть заданы для отслеживания нескольких атрибутов.
--tls-exit	Отключиться, если согласование TLS не удается.
--ns-cert-type client|server	Устарело. Используйте более современный эквивалент: --remote-cert-tls. Потребовать, чтобы сертификат партнера был подписан явно заданным определением nsCertType «client» или «server». Это полезная защитная опция для клиентов, обеспечивающая тот факт, что хост, с которым они соединяются, определенно является сервером. См. скрипт easy-rsa/build-key-server в качестве примера, как создать сертификат с полем nsCertType, установленным в значение «server». Если поле nsCertType серверного сертификата установлено в значение «server», клиенты могут проверить это с помощь опции --ns-cert-type server. Это важная предосторожность для защиты против атаки «человек посередине», где авторизованный клиент пытается соединиться с другим клиентом, притворяясь сервером. Атака легко предотвращается, если клиенты проверяют серверный сертификат с использованием одной из опций --ns-cert-type, --tls-remote или --tls-verify.
--remote-cert-ku v...	Потребовать, чтобы сертификат партнера был подписан явно заданным key usage. Это полезная защитная опция для клиентов, обеспечивающая тот факт, что хост, с которым они соединяются, определенно является сервером. Значение keyUsage, если оно присутствует в сертификате, валидируется с помощью TLS библиотеки в течение TLS-соединения. Указание этой опции без атрибутов требует наличия этого расширения (чтобы библиотека TLS проверифицировала значение). Если список v... также указан, в поле keyUsage должны быть заданы как минимум тот же набор битов, что и в одном из значений списка v . . . key usage следует закодировать в шестнадцатеричное число, можно указать более одного key usage.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--remote-cert-eku oid	Потребовать, чтобы сертификат партнера был подписан явно заданным extended key usage. Это полезная защитная опция для клиентов, обеспечивающая тот факт, что хост, с которым они соединяются, определенно является сервером. extended key usage следует закодировать в нотации OID или символическом представлении OpenSSL.
--remote-cert-tls client|server	Потребовать, чтобы сертификат партнера был подписан явно заданными key usage и explicit key usage, основанными на правилах TLS, приведенных в RFC3280. Это полезная защитная опция для клиентов, обеспечивающая тот факт, что хост, с которым они соединяются, определенно является сервером. Или, наоборот, для сервера, чтобы убедиться, что только хосты с клиентским сертификатом могут подключаться. Опция --remote-cert-tls client эквивалентна --remote-cert-ku 80 08 88 --remote-cert-eku «TLS Web Client Authentication». Key usage - digitalSignature и/или keyAgreement. Опция --remote-cert-tls server эквивалентна --remote-cert-ku a0 88 --remote-cert-eku «TLS Web Server Authentication». Key usage - digitalSignature и (keyEnchpherment или keyAgreement). Это важная предосторожность для защиты против атаки «человек посередине», где авторизованный клиент пытается соединиться с другим клиентом, притворяясь сервером. Атака легко предотвращается, если клиенты проверяют серверный сертификат с использованием одной из опций --remote-cert-tls, --tls-remote или --tls-verify.

-crl-verify crl	Проверить, не указан ли сертификат партнера в файле crl в формате PEM. CRL (список отзыва сертификатов) используется, когда конкретный ключ скомпрометирован, но вся PKI сохранена. Предположим, что у вас была PKI, состоящая из УЦ, корневого сертификата и нескольких клиентских сертификатов. Предположим, что ноутбук, содержащий ключ и сертификат клиента, был украден. Добавив украденный сертификат в файл CRL, вы можете отвергнуть любое соединение, которое пытается им воспользоваться, сохраняя общую целостность PKI. Единственный случай, когда необходимо пересоздавать всю PKI заново - если был скомпрометирован сам корневой сертификат. Если опциональный флаг dir задан, включится другой режим, где crl – каталог, содержащий файлы, проименованные отозванными серийными номерами (файлы могут быть пустыми, содержимое никогда не читается). Если клиент запрашивает соединение, где серийный номер сертификата клиента (десятичная строка) совпадает с именем какого-либо файла в этом каталоге, то соединение будет отклонено. Учтите, что файл или каталог crl зачитывается каждый раз, когда подключается партнер, если вы отключаете права привилегированного пользователя с помощью --user, убедитесь, что этот пользователь обладает достаточными правами, чтоб прочесть этот файл.

8.7 Информация по библиотеке SSL

--show-ciphers	(Независимая) Показать все алгоритмы шифров, которые можно использовать с опцией --cipher
--show-digests	(Независимая) Показать все алгоритмы дайджестов, которые можно использовать с опцией --auth

--show-tls	(Независимая) Показать криптонаборы TLS. «OpenVPN-ГОСТ» использует TLS для защиты канала управления, по которому происходит обмен ключами, используемыми для защиты трафика VPN. Шифрнаборы TLS будут отсортированы от самого высокого предпочтения (самый безопасный) к самому низкому. Для TLS 1.2 будут показаны все доступние криптонаборы. Для TLS 1.3 будут показаны только те криптонаборы, которые опеделены текущим значением опции --tls-ciphersuites. Обратите внимание, что может ли набор шифров из списка действительно работать, зависит от конкретной настройки обоих узлов.
--show-engines	(Независимая) Показать доступные в настоящее время модули engine, поддерживаемые библиотекой OpenSSL.

8.8 Создание случайного ключа

Эти опции используются только для режима зашифрования со статическим ключом, не для TLS.

--genkey	(Независимая) Создать случайный ключ для использования в качестве общего секрета с опцией --secret. Этот файл должен быть передан партнеру по ранее существовавшему защищенному каналу, такому как scp.
--secret file	Записать ключ в файл file.

8.9 Режим конфигурации сохраняемого туннеля TUN/TAP

Доступен на Linux 2.4.7+. Эти опции включают в себя независимый режим «OpenVPN-ГОСТ», который может быть использован для создания и удаления сохраняемых туннелей.

--mktun	(Независимая) Создать сохраняемый туннель на платформах, которые поддерживают их, таких, как Linux. Как правило, туннели TUN/TAP существуют только в течение того периода времени, когда приложение держит их открытыми. Эта опция пользуется способностью драйвера TUN/TAP создавать сохраняемые туннели, которые сохраняются в течение нескольких запусков «OpenVPN-ГОСТ» и закрываются только когда их удаляют или когда машина перезагружается. Одно из преимуществ сохраняемых туннелей - то, что они исключают необходимость для отдельных скриптов --up и --down запускать соответствующие команды ifconfig и route. Эти команды могут быть помещены в тот же скрипт оболочки, который начинает или заканчивает сеанс «OpenVPN-ГОСТ». Еще одно преимущество в том, что открытые соединения через туннель, основанный на TUN/TAP, не будут перезагружены, если партнер по «OpenVPN-ГОСТ» перезапускается. Это может быть полезным, чтобы предоставить непрерывную связь через туннель в случае DHCP-перезагрузки открытого IP-адреса партнера (см. опцию --ipchange выше). Один недостаток сохраняемых туннелей состоит в том, что труднее автоматически конфигурировать их величину MTU (см. опции --link-mtu и --tun-mtu выше). На некоторых платформах, таких как Windows, туннели TAP-Win32 сохраняются по умолчанию.
--rmtun	(Независимая) Удалить сохраняемый туннель
--dev tunX | tapX	Устройство TUN/TAP
--user user	Опциональный пользователь, который должен быть владельцем этого туннеля
--group group	Опциональная группа, которая должна быть владельцем этого туннеля.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

8.10 Опции, специфичные для Windows

--win-sys path	Установить путь к системному каталогу Windows для использования при поиске системных выполняемых файлов, таких как route.exe и netsh.exe. По умолчанию, если эта директива не указана, «OpenVPN-ГОСТ» использует переменную среды SystemRoot. Эта опция изменила поведение в последней версии «OpenVPN-ГОСТ». Раньше приходилось определять --win-sys env, чтобы использовать переменную среды SystemRoot, иначе по умолчанию использовался C:\WINDOWS. Больше не нужно использовать ключевое слово env, оно будет просто проигнорировано. При обнаружении это слова предупреждающее сообщение будет сохранено в конфигурационном файле.
--ip-win32 method	Используя опцию --ifconfig в Windows, установить адрес и сетевую маску адаптера TAP-Win32 с использованием method. Не используйте эту опцию, если вы также используете --ifconfig. manual - Не устанавливать IP-адрес или сетевую маску автоматически. Вместо этого вывести сообщение на консоль, требующее от пользователя сконфигурировать адаптер вручную и указывающее IP/сетевую маску, которые «OpenVPN-ГОСТ» ожидает в качестве установок адаптера. dynamic [offset] [lease-time] - Автоматически установить IP-адрес и сетевую маску, отвечая на DHCP-запросы, созданные ядром. Этот режим - вероятно, «самое чистое» решение для установки свойств TCP/IP, поскольку он использует хорошо известный протокол DHCP. Но существуют два предварительных требования для использования этого режима.

	1. Свойства TCP/IP для адаптера TAP-Win32 должны быть установлены в «Obtain an IP address automatically» 2. «OpenVPN-ГОСТ» необходимо потребовать IP-адрес в подсети для использования в качестве адреса виртуального сервера DHCP. По умолчанию в режиме -dev tap «OpenVPN-ГОСТ» возьмет, как правило, неиспользуемый первый адрес в подсети. Например, если ваша подсеть 192.168.4.0, а сетевая маска 255.255.255.0, то «OpenVPN-ГОСТ» возьмет IP-адрес 192.168.4.0 в качестве адреса виртуального сервера DHCP. В режиме --dev tun «OpenVPN-ГОСТ» заставит сервер DHCP вести себя так, будто он установлен в удаленном конечном пункте. Опциональный параметр offset - целое число между -256 и 256, по умолчанию равное 0. Если offset больше 0, сервер DHCP будет вести себя как IP-адрес при сетевом адресе + offset. Если offset меньше 0, DHCP будет вести себя как IP-адрес при широковещательном адресе + offset. Можно использовать команду Windows ipconfig /all, чтобы показать, чем Windows считает адрес сервера DHCP. «OpenVPN-ГОСТ» «потребует» этот адрес, поэтому обязательно используйте свободный адрес. Тем не менее, различные экземпляры «OpenVPN-ГОСТ», включая различные концы одного и того же соединения, могут использовать один и тот же адрес виртуального сервера DHCP. Параметр lease-time контролирует время аренды присваивания DHCP, данного адаптеру TAP-Win32, и выражается в секундах. Как правило, предпочтительно очень большое время аренды, потому что оно предотвращает потерю маршрутов, включающих адаптер TAP-Win32, когда система засыпает. Умолчательное время аренды - один год. netsh - автоматически установить IP-адрес и маску сети, используя командно-строчную команду Windows «netsh». Этот способ, похоже, корректно работает на Windows XP, но не на Windows 2000. ipapi - автоматически установить IP-адрес и маску сети, используя Windows IP Helper API. Этот подход не имеет идеальной семантики, хотя тестирование показало, что на практике он работает хорошо. Если вы используете эту опцию, лучше всего оставить свойства TCP/IP для адаптера TAP-Win32 в умолчательном состоянии, т.е. «Obtain an IP address automatically».

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

	adaptive - (Умолчание) изначально попытаться использовать способ dynamic, затем перейти к netsh, если согласование DHCP с адаптером TAP-Win32 не завершается успешно за 20 секунд. Такие случаи известны, когда пакеты некоего постороннего брандмауэра, установленного на клиентской машине, блокируют согласование DHCP, используемое адаптером TAP-Win32. Обратите внимание, что если происходит переход к netsh, TCP/IP-свойства адаптера TAP-Win32 будут переустановлены с DHCP на статические, и это заставит будущие перезапуски «OpenVPN-ГОСТ», использующей режим adaptive, сразу же использовать netsh, не используя сначала dynamic. Чтобы «переключить» режим adaptive с использования netsh, запустите «OpenVPN-ГОСТ» по крайней мере один раз с использованием режима dynamic, чтобы восстановить TCP/IP-свойства адаптера TAP-Win32 в конфигурацию DHCP.
--route-method m	Какой способ m использовать для добавления маршрутов в Windows? adaptive (умолчание) - сначала попытаться использовать IP helper API. Если это не удается, вернуться к команде оболочки route.exe. ipapi - использовать IP helper API. exe - вызвать команду оболочки route.exe.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--dhcp-option type [parm]	Установить расширенные TCP/IP-свойства адаптера TAP-Win32, опция должна использоваться с --ip-win32 dynamic или --ip-win32 adaptive. Эту опцию можно использовать, чтобы установить дополнительные TCP/IP-свойства адаптера TAP-Win32, и особенно полезна для конфигурирования клиента «OpenVPN-ГОСТ» для достижения сервера Samba через VPN. DOMAIN name - установить DNS-суффикс, специфичный для соответствующего соединения. DNS addr - установить адрес первичного сервера доменных имен. Повторить эту опцию, чтобы установить адреса вторичных DNS-серверов. Обратите внимание, что DNS IPv6 сервера сейчас настраиваются с помощью netsh (существующий DHCP код может использовать только IPv4 DHCP, а этот протокол разрешает только IPv4 адреса где-либо). Опция будет помещена в среду, так что --up скрипт может действовать в соответствии с этим, если необходимо. WINS addr - Установить адрес первичного WINS-сервера (NetBIOS over TCP/IP Name Server). Повторить эту опцию, чтобы установить адреса вторичных WINS-серверов. NBDD addr - Установить адрес первичного NBDD-сервера (NetBIOS over TCP/IP Datagram Distribution Server). Повторить эту опцию, чтобы установить адреса вторичных NBDD-серверов. NTP addr - установить адрес первичного NTP-сервера (Network Time Protocol). Повторить эту опцию, чтобы установить адреса вторичных NTP-серверов. NBT type - установить тип узла NetBIOS over TCP/IP. Возможные опции: 1 - b-узел (широковещание), 2 - p-узел (именные запросы точка-в-точку к WINS-серверу), 4 - m-узел (широковещание, затем сервер имен запросов) и 8 - h-узел (сервер имен запросов, затем широковещание). NBS scope-id - установить область действия NetBIOS over TCP/IP. ID области действия NetBIOS предоставляет расширенный сервис имен для модуля NetBIOS over TCP/IP (известного как NBT). Главная цель ID области действия NetBIOS - изолировать трафик NetBIOS в одной сети так, чтобы он проходил только через узлы с тем же ID области действия NetBIOS. ID области действия NetBIOS - символьная строка, которая добавляется к имени NetBIOS. ID области действия NetBIOS на двух хостах должен совпадать, или два хоста не смогут общаться. ID области действия NetBIOS также позволяет компьютерам использовать одно и то же компьютерное имя, потому что у них разные ID области действия. ID области действия становится частью имени NetBIOS, делая это имя уникальным. DISABLE-NBT - отключить NetBIOS-over-TCP/IP.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

	Обратите внимание, что если опция --dhcp-option передана через --push клиенту, работающему не в Windows, опция будет сохранена в окружении клиента перед тем, как будет вызван скрипт up, под названием «foreign_option_n».
--tap-sleep n	Заставить «OpenVPN-ГОСТ» заснуть на n секунд сразу же после того, как состояние адаптера TAP-Win32 установлено в «connected». Эта опция предназначена для использования при проблемах с опциями --ifconfig и --ip-win32, и используется, чтобы дать адаптеру TAP-Win32 время подняться, прежде чем к нему будут применены операции Windows IP Helper API.
--show-net-up	Вывести взгляд «OpenVPN-ГОСТ» на системную таблицу маршрутизации и список сетевых адаптеров в syslog или файл журнала после того, как был поднят адаптер TUN/TAP и были добавлены какие-либо маршруты.
--block-outside-dns	Блокировать DNS сервера из других сетевых адаптеров, чтобы предотвратить DNS утечки. Эта опция предотвращает доступ любых приложений к TCP или UDP порту 53 кроме одного внутри туннеля. Для этого используется Windows Filtering Platform (WFP). Эта опция считается неизвестной на не-Windows платформах, вызывая ошибку. Возможно, вы захотите использовать --setenv opt или --ignore-unknown-option, чтобы проигнорировать указанную ошибку. Учтите, что отправка неизвестных опций с сервера не вызывает фатальных ошибок.
--dhcp-renew	Попросить Windows обновить аренду адаптера TAP при запуске. Эта опция, как правила, не является необходимой, потому что Windows автоматически запускает пересогласование DHCP на адаптере TAP, когда он поднимается, однако если вы установили свойство Media Status адаптера TAP-Win32 в «Always Connected», вам может понадобиться этот флаг.
--dhcp-release	Попросить Windows освободить аренду адаптера TAP при выключении. Эта опция имеет те же оговорки, что и --dhcp-renew.
--register-dns	Запустить net stop dnscache, net start dnscache, ipconfig /flushdns и ipconfig /registerdns при инициации соединения. Известно, что это заставляет Windows узнавать переданные DNS-серверы.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--pause-exit	Вывести сообщение «press any key to continue» на консоль, прежде чем программа «OpenVPN-ГОСТ» завершает работу. Эта опция автоматически используется проводником Windows, когда «OpenVPN-ГОСТ» запущена с конфигурационным файлом, выбранным с помощью контекстного меню проводника.
--service exit-event [0|1]	Следует использовать, когда «OpenVPN-ГОСТ» автоматически запускается другой программой в таком контексте, что никакое взаимодействия с пользователем через монитор или клавиатуру невозможно. В общем, конечным пользователям не следует иметь необходимость явным образом использовать эту опцию, потому что она автоматически добавляется сервисной оболочкой «OpenVPN-ГОСТ», когда данная конфигурация «OpenVPN-ГОСТ» запускается как сервис. exit-event - имя объекта глобального события Windows, и «OpenVPN-ГОСТ» будет непрерывно проверять состояние этого объекта события и завершит работу, когда объект станет сигнализированным. Второй параметр указывает исходное состояние параметра exit-event и, как правило, по умолчанию равен 0. Может выполняться одновременно несколько процессов «OpenVPN-ГОСТ» с одним и тем же параметром exit-event. В любом случае контролирующий процесс может сигнализировать exit-event, заставляя все такие процессы завершиться. При выполнении процесса «OpenVPN-ГОСТ» с директивой --service, у «OpenVPN-ГОСТ», вероятно, не будет консольного окна для вывода сообщений о статусе и ошибках, поэтому полезно использовать опции --log или --log-append, чтобы записывать эти сообщения в файл.
--show-adapters	(Независимая) Показать доступные адаптеры TAP-Win32, которые могут быть выбраны с помощью опции --dev-node. В системах, отличных от Windows, команда ifconfig предоставляет сходную функциональность.

--allow-nonadmin [TAP-adapter]	(Независимая) Установить TAP-адаптер так, чтобы позволить доступ от не-администраторов. Если параметр TAP-adapter опущен, все TAP-адаптеры системы будут сконфигурированы так, чтобы позволить неадминистративный доступ. Настройка неадминистративного доступа будет продолжаться только тот период времени, пока объект устройства TAP-Win32 и драйвер остаются загруженными, и ее необходимо будет снова устанавливать после перезагрузки, или если драйвер был выгружен и снова загружен. Эту директиву может использовать только администратор.
--show-valid-subnets	(Независимая) Показать действительные подсети для эмуляции --dev tun. Поскольку драйвер TAP-Win32 экспортирует интерфейс ethernet в Windows, и поскольку устройства TUN по своей природе работают точка-в-точку, для драйвера TAP-Win32 необходимо накладывать некоторые ограничения на выбор адресов в конечном пункте TUN. А именно, конечные пункты точка-в-точку, использованные в эмуляции устройства TUN, должны быть два адреса из середины подсети /30 (сетевая маска 255.255.255.252).
--show-net	(Независимая) Показать взгляд «OpenVPN-ГОСТ» на системную таблицу маршрутизации и список сетевых адаптеров.

8.11 Автономные опции отладки

--show-gateway [v6target]	(Независимая) Показать текущий IPv4 и IPv6 шлюз по умолчанию и интерфейс к шлюзу (если упомянутый протокол включен). Если IPv6 адрес передается в качестве аргумента, IPv6 маршрут отправляется для этого хоста.

8.12 Опции, относящиеся к IPv6

Следующие опции существуют для поддержки IPv6 туннелирования в точка-точка и клиент-сервер режимах. Все опции смоделированы по образцу их аналогов в IPv4, так что более детальные описания, приведены там, применимы и тут (за исключением --topology, который не влияет на IPv6).

--ifconfig-ipv6 ipv6addr/bits ipv6remote	Сконфигурирует IPv6-адрес ipv6addr/bits для устройств TUN. ipv6remote используется как адрес, на который должны маршрутизироваться пакеты для --route-ipv6, если не задан gateway.

--route-ipv6 ipv6addr/bits [gateway] [metric]	Настроить IPv6 маршрутизацию в системе для отправки указанной сети IPv6 в «OpenVPN-ГОСТ» TUN. Параметр gateway используется только для IPv6 маршрутов, проходящих через TAP устройства, и если он отсутствует, то будет использовано ipv6remote поле из --ifconfig-ipv6 .
--server-ipv6 ipv6addr/bits	Удобная функция для включения нескольких IPv6 опций одновременно, а именно --ifconfig-ipv6, --ifconfig-ipv6-pool и --push tun-ipv6. Принимается только если установлен “--mode server”, либо “--server”. Отправка --tun-ipv6 директивы выполняется для старых клиентов, которым требуется явное указание “--tun-ipv6” в их конфигурации .
--ifconfig-ipv6-pool ipv6addr/bits	Определяет диапазон адресов IPv6 для динамического назначение клиентам. Диапазон начитается с ipv6addr и выдаваемый адрес увеличивается на 1 для каждого следующего клиента.
--ifconfig-ipv6-push ipv6addr/bits ipv6remote	Используется для ccd/per-client настройки статического IPv6 интерфейса. См. детали в --client-config-dir и --ifconfig-push .
--iroute-ipv6 ipv6addr/bits	Используется для ccd/per-client настройки статического IPv6 интерфейса. См. детали по настройке и использованию в --iroute, и по взаимодействию --iroute и --route.

8.13 Скриптование и переменные среды

«OpenVPN-ГОСТ» экспортирует ряд переменных среды, использующихся в скриптах, определенных пользователем.

8.13.1 Порядок выполнения скриптов

--up	Выполняется после того, как связывается сокет TCP/UDP и открывается TUN/TAP.
--tls-verify	Выполняется, когда у нас есть пока еще недоверенный удаленный партнер.
--ipchange	Выполняется после аутентификации соединения или изменения удаленного IP-адреса.
--client-connect	Выполняется в режиме --mode server сразу после клиентской аутентификации.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

--route-up	Выполняется после аутентификации соединения, или сразу же, или спустя несколько секунд, время ожидания определяется опцией --route-delay.
--route-pre-down	Выполняется прямо перед удалением маршрутов.
--client-disconnect	Выполняется в режиме --mode server при закрытии экземпляра клиента.
--down	Выполняется после закрытия TCP/UDP и TUN/TAP.
--learn-address	Выполняется в режиме --mode-server, когда или адрес/маршрут IPv4 или адрес MAC добавляется во внутреннюю таблицу маршрутизации «OpenVPN-ГОСТ».
--auth-user-pass-verify	Выполняется в режиме --mode-server при подключении новых клиентов, когда клиент еще не является доверенным.

8.13.2 Типы и преобразование строк

В определенных случаях «OpenVPN-ГОСТ» выполняет преобразование символов в строках. А именно, любые символы, не входящие в набор разрешенных символов для каждого типа строк, будут преобразованы в знак подчеркивания.

Вопрос. Почему преобразование строк необходимо?

Ответ. Это важная защитная функция, предназначенная для того, чтобы предотвратить злонамененное кодирование строк из недоверенных источников, которые передаются как параметры в скрипты, сохраняются в среде, используются в качестве common name, переводятся в имя файла и т.д.

Вопрос. Можно ли отключить преобразование строк?

Ответ. Да, используя опцию --no-name-remapping, однако эту опцию следует считать дополнительной.

Вот краткое описание имеющихся в «OpenVPN-ГОСТ» типов строк и разрешенных классов символов для каждой строки.

Имена, составленные по правилам X.509: буквы и цифры, знак подчеркивания (_), дефис (-), точка (.), at-коммерческое (@), двоеточие (:), косая черта (/) и знак равенства (=). Буквы и цифры определяются как символы, которые заставляют функцию библиотеки C isalnum() вернуть значение «истинно».

Поля Common Name: Буквы и цифры, знак подчеркивания (_), дефис (-), точка (.) и at-коммерческое (@).

--auth-user-pass username: То же, что у полей Common Name, с одним исключением: параметр username передается в плагин OPENVPN_PLUGIN_AUTH_USER_PASS_VERIFY как есть, без преобразования.

--auth-user-pass password: Любой «печатный» символ, кроме CR и LF. Печатные символы определяются как символы, которые заставляют функцию библиотеки C isprint() вернуть значение «истинно».

--client-config-dir filename как производные от полей common name или username:

Буквы и цифры, знак подчеркивания (_), дефис (-) и точка (.), за исключением «.» и «..» в качестве отдельных строк, а также символ at-коммерческое (@), добавленный для совместимости с классом символов common name.

Имена переменных среды: буквы и цифры или знак подчеркивания (_).

Значения переменных среды: любой печатный символ.

Во всех случаях, символы в строке, которые не являются членами разрешенного класса символов для этого типа строк, преобразовываются в знак подчеркивания.

8.13.3 Переменные среды

Однажды установленная переменная среды сохраняется, пока ее не переустановят или не будет перезапущена система.

В серверном режиме переменные среды, установленные «OpenVPN-ГОСТ», определяются в соответствии с объектами клиентов, с которыми они ассоциированы, поэтому не должно быть случаев, когда скрипты получают доступ к ранее установленным переменным, которые относятся к другим экземплярам клиентов.

bytes_received	Общее количество байтов, полученных от клиента во время сеанса VPN. Устанавливается перед выполнением скрипта --client-disconnect.
bytes_sent	Общее количество байтов, переданных клиенту во время сеанса VPN. Устанавливается перед выполнением скрипта --client-disconnect.
common_name	Поле X509 common name аутентифицированного клиента. Устанавливается перед выполнением скриптов --client-connect, --client-disconnect и --auth-user-pass-verify.
config	Имя первого файла --config. Устанавливается при инициации программы и переустанавливается при SIGNUP.
daemon	Устанавливается в «1», если указана директива --daemon, и в «0» во всех остальных случаях. Устанавливается при инициации программы и переустанавливается при SIGNUP.
daemon_log_redirect	Устанавливается в «1», есди определены директивы log или log-append, и в «0» во всех остальных случаях. Устанавливается при инициации программы и переустанавливается при SIGNUP.
dev	Действительное имя устройства TUN/TAP, включая номер, если он существует. Устанавливается перед выполнением скриптов --up или --down.

dev_idx	В Windows индекс устройства TUN/TAP адаптера (для использования в netsh.exe вызовах, которые иногда неправильно работают с именами интерфейсов). Устанавливается перед выполнением скриптов --up или --down.
foreign_option_{n}	Опция, переданная директивой --push клиенту, который не обладает встроенной поддержкой этой опции, например, опция --dhcp-option в системе, отличной от Windows, будет записана в эту последовательность переменных среды перед выполнением скрипта --up.
ifconfig_broadcast	Широковещательный адрес для сегмента виртуальной сети ethernet, определяемый из опции --ifconfig при использовании опции --dev tap. Устанавливается перед тем, как «OpenVPN-ГОСТ» вызывает команду ifconfig или netsh (версия ifconfig для Windows), что обычно случается до выполнения скрипта --up.
ifconfig_ipv6_local	IP-адрес локального конечного пункта VPN, указанный в опции --ifconfig-ipv6 (первый параметр). Устанавливается перед тем, как «OpenVPN-ГОСТ» вызывает команду ifconfig или netsh (версия ifconfig для Windows), что обычно случается до выполнения скрипта --up.
ifconfig_ipv6_netbits	Длина префикса IPv6 сети на VPN интерфейсе. Получается из параметра /nnn IPv6-адреса, указанного в опции --ifconfig-ipv6 (первый параметр). Устанавливается перед тем, как «OpenVPN-ГОСТ» вызывает команду ifconfig или netsh (версия ifconfig для Windows), что обычно случается до выполнения скрипта --up .
ifconfig_ipv6_remote	IP-адрес удаленного конечного пункта VPN, указанный в опции --ifconfig-ipv6 (второй параметр) при использовании опции --dev tun. Устанавливается перед тем, как «OpenVPN-ГОСТ» вызывает команду ifconfig или netsh (версия ifconfig для Windows), что обычно случается до выполнения скрипта --up.
ifconfig_local	IP-адрес локального конечного пункта VPN, указанный в опции --ifconfig (первый параметр). Устанавливается перед тем, как «OpenVPN-ГОСТ» вызывает команду ifconfig или netsh (версия ifconfig для Windows), что обычно случается до выполнения скрипта --up.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

ifconfig_remote	IP-адрес удаленного конечного пункта VPN, указанный в опции --ifconfig (второй параметр) при использовании опции --dev tun. Устанавливается перед тем, как «OpenVPN-ГОСТ» вызывает команду ifconfig или netsh (версия ifconfig для Windows), что обычно случается до выполнения скрипта --up.
ifconfig_netmask	Маска подсети сегмента виртуальной сети ethernet, указанная как второй параметр опции --ifconfig при использовании опции --dev tap. Устанавливается перед тем, как «OpenVPN-ГОСТ» вызывает команду ifconfig или netsh (версия ifconfig для Windows), что обычно случается до выполнения скрипта --up.
ifconfig_pool_local_ip	Локальный виртуальный IP-адрес для туннеля TUN/TAP, взятый из директивы --ifconfig-push, если она указана, или же из диапазона ifconfig (который управляется директивой конфигурационного файла --ifconfig pool). Устанавливается только для туннелей с опцией --dev tun. Эта опция устанавливается на сервере перед выполнением скриптов --client-connect и --client-disconnect.
ifconfig_pool_netmask	Сетевая маска виртуального IP для туннеля TUN/TAP, взятая из директивы --ifconfig-push, если она указана, или же из файла ifconfig pool (который управляется директивой конфигурационного файла --ifconfig pool). Устанавливается только для туннелей с опцией --dev tap. Эта опция устанавливается на сервере преде выполнением скриптов --client-connect и --client-disconnect.
ifconfig_pool_remote_ip	Удаленный виртуальный IP-адрес для туннеля TUN/TAP, взятый из директивы --ifconfig-push, если она указана, или же из файла ifconfig pool (который управляется директивой конфигурационного файла --ifconfig pool). Эта опция устанавливается на сервере перед выполнением скриптов --client-connect и --client-disconnect.
link_mtu	Максимальный размер пакета (без учета IP-заголовка) туннельных данных в режиме транспорта в UDP-туннеле. Устанавливается перед выполнением скриптов --up и --down.
local	Параметр опции --local. Устанавливается при запуске программы и переустанавливается при SIGNUP.

local_port	Номер локального порта, указанный опцией --port или --lport. Устанавливается при запуске программы и переустанавливается при SIGNUP.
password	Пароль, предоставленный подключающимся клиентом. Устанавливается перед выполнением скрипта --auth-user-pass-verify, только когда указан модификатор via-env, и удаляется из среды после того, как скрипт возвращает значение.
proto	Параметр опции --proto. Устанавливается при запуске программы и переустанавливается при SIGNUP.
remote_{n}	Параметр опции --remote. Устанавливается при запуске программы и переустанавливается при SIGNUP.
remote_port_{n}	Номер удаленного порта, указанный опцией --port или --rport. Устанавливается при запуске программы и переустанавливается при SIGNUP.
route_net_gateway	Ранее существовавший умолчательый IP-гейт в таблице маршрутизации системы. Устанавливается перед выполнением скрипта --up.
route_vpn_gateway	Умолчательный гейт, используемый опциями --route, указанный либо в опции --route-gateway либо в качестве второго параметра опции --ifconfig, когда указана опция --dev tun. Устанавливается перед выполнением скрипта --up.
route_{parm}_{n}	Набор переменных, которые определяют каждый добавляемый маршрут, и устанавливаются перед выполнением скрипта --up. parm может иметь значения «network», «netmask», «gateway» или «metric». n - номер маршрута «OpenVPN-ГОСТ», начиная с 1. Если сеть или гейт являются разрешимыми именами DNS, будут записываться трансляции их IP-адресов вместо их имен, как указано в командной строке или конфигурационном файле.

route_ipv6_{parm}_{n}	Набор переменных, которые определяют каждый добавляемый IPv6 маршрут, и устанавливаются перед выполнением скрипта --up. parm может иметь значения «network» или «gateway» («netmask» содержится как “/nnn” в route_ipv6_network_{n}, в отличие от IPv4, где он передается в виде отдельной переменной среды). n - номер маршрута «OpenVPN-ГОСТ», начиная с 1. Если сеть или гейт являются разрешимыми именами DNS, будут записываться трансляции их IP-адресов вместо их имен, как указано в командной строке или конфигурационном файле.
peer_cert	Имя временного файла, содержащее сертификат клиента при подключении. Полезно в сочетании с --tls-verify .
script_context	Устанавливается в «init» или «restart» перед выполнением скрипта up/down. Дополнительную информацию см. в описании опции --up.
script_type	Один из следующих скриптов: up, down, ipchange, route-up, tls-verify, auth-user-pass-verify, client-connect, client-disconnect, или learn-address. Устанавливается перед выполнением любого скрипта.
signal	Причина выхода или перезагрузки. Может быть одним из sigusr1, sighup, sigterm, sigint, inactive (управляемым опцией --inactive), ping-exit (управляемым опцией --ping-exit), ping-restart (управляемым опцией --ping-restart), connection-reset (запускается при перезагрузке TCP-соединения), error, or unknown (неизвестный сигнал). Эта переменная устаналивается непосредственно перед выполнением скрипта down.
time_ascii	Метка времени соединения клиента, отформатированная как строка, которую может прочитать человек. Устанавливается перед выполнением скрипта --client-connect.
time_duration	Продолжительнсть (в секундах) сеанса клиента, который сейчас отсоединяется. Устанавливается перед выполнением скрипта --client-disconnect.
time_unix	Метка времени соединения клиента, отформатированная как юниксовая целая величина даты/времени. Устанавливается перед выполнением скрипта --client-connect.

tls_digest_{n} / tls_digest_sha256_{n}	Содержит отпечаток сертификата SHA1/SHA256, где n - уровень проверки. Устанавливается только для TLS-соединений. Устанавливается перед выполнением скрипта --tls-verify.
tls_id_{n}	Серия полей сертификатов от удаленного партнера, где n - уровень проверки. Устанавливается только для TLS-соединений. Устанавливается перед выполнением скрипта --tls-verify.
tls_serial_{n}	Серийный номер сертификата от удаленного партнера, где n - уровень проверки. Устанавливается только для TLS-соединений. Устанавливается перед выполнением скрипта --tls-verify. Представлен в виде десятичной строчки, к примеру, «933971680», который подходит для выполнения последовательных OCSP запросов (для OpenSSL не добавляйте к строке «0x»). Если что-то пойдет не так при чтении значения из сертификата, он будет пустой строкой, поэтому ваш код должен проверять на такой случай. См. contrib/OCSP_check/OCSP_check.sh скрипт для примера.
tls_serial_hex_{n}	По аналогии с tls_serial_{n}, но в шестнадцатеричном формате (к примеру, «12:34:56:78:9A»).
tun_mtu	MTU устройства TUN/TAP. Устанавливается перед выполнением скрипта --up или --down.
trusted_ip (or trusted_ip6)	Действительный IP-адрес подключающегося клиента или партнера, который был аутентифицирован. Устанавливается перед выполнением скриптов --ipchange, --client-connect и --client-disconnect.
trusted_port	Действительный номер порта подключающегося клиента или партнера, который был аутентифицирован. Устанавливается перед выполнением скриптов --ipchange, --client-connect и --client-disconnect.
untrusted_ip (or untrusted_ip6)	Действительный IP-адрес подключающегося клиента или партнера, который еще не был аутентифицирован. Иногда используется, чтобы выполнить функцию nmap для подключающегося хоста в скрипте --tls-verify, чтобы обеспечить правильную работу брандмауэра. Устанавливается перед выполнением скриптов --tls-verify и --auth-user-pass-verify.

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения

untrusted_port	Действительный номер порта подключающегося клиента или партнера, который был аутентифицирован. Устанавливается перед выполнением скриптов --tls-verify и --auth-user-pass-verify.
username	Логин, предоставленный подключающимся клиентом. Устанавливается перед выполнением скрипта --auth-user-pass-verify, только когда указан модификатор via-env.

X509_{n}_{subject_field}	Поле X509 subject из сертификата удаленного партнера, где n - уровень проверки. Устанавливается только для TLS-соединений. Устанавливается перед выполнением скрипта --tls-verify. Эта переменная похожа на tls_id_n, за исключением того, что поля-компоненты X509 subject удалены, и никакого преобразования строк в значениях этих полей не происходит (за исключением преобразования управляющих символов в знак подчеркивания) Например, на сервере «OpenVPN-ГОСТ» при использовании образца клиентского сертификата из каталога sample-keys (client.crt) будут установлены следующие переменные (обратите внимание, что уровень проверки равен 0 для клиентского сертификата и 1 для сертификата УЦ):
	`X509_0_emailAddress=me@myhost.mydomain`
	`X509_0_CN=Test-Client`
	`X509_0_O=OpenVPN-TEST`
	`X509_0_ST=NA`
	`X509_0_C=KG`
	`X509_1_emailAddress=me@myhost.mydomain`
	`X509_1_O=OpenVPN-TEST`
	`X509_1_L=BISHKEK`
	`X509_1_ST=NA`
	`X509_1_C=KG`

8.14 Поддержка встроенных файлов

«OpenVPN-ГОСТ» позволяет для опций --ca, --cert, --extra-certs, --key, --pkcs12, --secret, --crl-verify, --http-proxy-user-pass, --tls-auth и --tls-crypt включать содержимое соответствующих файлов непосредственно в файл конфигурации. Каждый встроенный файл начинается со строки <option> и заканчивается строкой </option>. Пример использования встроенного файла:

```
<cert>
-----BEGIN CERTIFICATE-----
[...]
-----END CERTIFICATE-----
</cert>
```

При использовании функции встроенных файлов с --pkcs12 встроенный файл должен быть закодирован в base64. Кодирование файла .p12 в base64 можно осуществить, к примеру, с помощью OpenSSL, запустив openssl base64 -in input.p12.

8.15 Сигналы

SIGHUP	Заставляет «OpenVPN-ГОСТ» закрыть все сетевые соединения и соединения TUN/TAP, перезапуститься, перечитать конфигурационный файл (если есть) и заново открыть сетевые соединения и соединения TUN/TAP.
SIGUSR1	Похож на SIGHUP, за исключением того, что конфигурационный файл не перечитывается, и, возможно, не закрывается устройство TUN/TAP, перечитываются ключевые файлы, сохраняется локальный IP-адрес/порт, или сохраняется последний аутентифицированный удаленный IP-адрес/порт, основанные на опциях --persist-tun, --persist-key, --persist-local-ip, and --persist-remote-ip соответственно (см. выше). Этот сигнал также может быть сгенерирован внутри «OpenVPN-ГОСТ» состоянием таймаута под управлением опции --ping-restart. Этот сигнал в сочетании с опцией --persist-remote-ip может быть отправлен, когда подлежащие параметры сетевого интерфейса хоста меняются, например, когда хост - клиент DHCP и ему присвоен новый IP-адрес. Дополнительную информацию см. в описании опции --ipchange выше.
SIGUSR2	Заставляет «OpenVPN-ГОСТ» вывести свою текущую статистику (в файл syslog, если используется опция --daemon, в противном случае на стандартный вывод).
SIGINT, SIGTERM	Заставляет «OpenVPN-ГОСТ» корректно завершить работу.

Лист регистрации изменений									
Номера листов (страниц)					Всего листов (страниц) в докум.	№ документа	Входящий сопроводительного докум. и дата	Подпись	Дата
Изм.	измененных	замененных	новых	аннулированных					

Порядковый № изменения	Подпись лица, ответственного за изменение	Дата внесения изменения