127018, Москва, Сущёвский Вал, 18
Телефон: (495) 995 4820
Факс: (495) 995 4820
https://CryptoPro.ru
E-mail: info@CryptoPro.ru

Средство Криптографической Защиты Информации

КриптоПро CSP Версия 5.0 R2 КС1 Исполнение 1-Base Приложения для создания TLS-туннеля (stunnel, stunnel_msspi)

ЖТЯИ.00101-02 93 03
Листов 22

2021 г

Содержание

Аннотация

В состав СКЗИ «КриптоПро CSP» версия 5.0 R2 КС1 исполнение 1-Base входит 2 реализации приложения для создания TLS-туннеля — `stunnel` и `stunnel_msspi`.

Приложения предназначены для создания TLS-защищенного соединения между клиентом и локальным (inetd-запускаемым) или удаленным сервером.

Приложение `stunnel_msspi` является развитием приложения `stunnel`.

Для установления TLS-соединения утилита `stunnel_msspi` использует вызовы интерфейса SSPI. Для работы с сертификатами ключей проверки ЭП и ключевыми контейнерами утилита использует вызовы интерфейса CryptoAPI (CAPIlite) СКЗИ «КриптоПро CSP».

1 Использование программы в ОС Windows

1.1 Общий синтаксис

```
stunnel [ [-install | -uninstall | -start | -stop]
[-quiet] [<filename>] ] | -help | -version | -sockets

stunnel_msspi [ [-install | -uninstall | -start | -stop | -reload | -reopen | -exit]
[-quiet] [<filename>] ] | -help | -version | -sockets | -options
```

<filename>	использовать указанный конфигурационный файл
-install	установить службу NT
-uninstall	удалить службу NT
-start	запустить службу NT
-stop	остановить службу NT
-reload	перезагрузить конфигурационный файл для работающей службы NT
-reopen	переоткрыть лог-файл для работающей службы NT
-exit	выйти из уже запущенного stunnel
-quiet	не показывать окна с сообщениями
-help	вывести справку (описание опций конфигурационного файла)
-version	вывести версию stunnel
-sockets	вывести опции сокетов
-options	вывести поддерживаемые опции TLS

1.2 Установка службы

Установка службы производится путём запуска следующей команды:

```
stunnel.exe -install

stunnel_msspi.exe -install
```

В дальнейшем служба для старта будет использовать файл `stunnel.exe (stunnel_msspi.exe)` из той папки, откуда была произведена установка.

1.3 Настройка службы

Перед установкой службы `stunnel (stunnel_msspi)` необходимо выбрать режим работы службы, установить сертификаты и сформировать файл конфигурации.

1. **Выбор варианта использования службы:**
   - в режиме клиента
   - в режиме сервера

В режиме клиента `stunnel (stunnel_msspi)` принимает трафик от клиентского приложения, зашифровывает его и отправляет на сервер. На сервере трафик расшифровывается и передаётся конечному приложению или другой службе на этом сервере.

2. **Установка сертификатов**

Для работы службы `stunnel (stunnel_msspi)` в режиме сервера обязательно нужен сертификат аутентификации сервера. Сервер может требовать, а может не требовать сертификат клиента при соединении клиента с сервером.

Как на клиенте, так и на сервере нужно установить необходимые сертификаты:

1) сертификат корневого Центра Сертификации (ЦС) должен быть установлен в хранилище «Доверенные корневые Центры Сертификации» локального компьютера;

2) если сертификат сервера или клиента выдан на подчинённом ЦС, сертификаты всех подчинённых ЦС в цепочке должны быть установлены в хранилище «Промежуточные Центры Сертификации» локального компьютера;

3) на сервере должен быть установлен сертификат сервера в хранилище «Личные» локального компьютера с привязкой к контейнеру закрытого ключа сервера;

4) если сервер требует сертификат клиента, то на клиентском компьютере должен быть установлен сертификат клиента в хранилище «Личные» локального компьютера с привязкой к контейнеру закрытого ключа клиента.

3. **Запись сертификатов в файл**

После установки сертификата сервера или клиента в хранилище необходимо дополнительно сохранить этот сертификат в файл на диске (без закрытого ключа, без цепочки сертификатов (файл *.cer) в формате BASE64 или DER).

4. **Формирование файла конфигурации** (см. ниже)

1.3.1 Файл конфигурации

Каждая строка конфигурационного файла может быть:
- пустой строкой (игнорируется);
- комментарием, начинающимся с ';' (игнорируется);
- парой 'option_name = option_value';
- строкой '[service_name]', с которой начинается секция описания конкретной службы.

Параметр адреса любой опции может быть:
- номером порта;
- парой 'IP-адрес:номер порта', где IP-адрес — IP или доменное имя.

Примечание. stunnel_msspi содержит расширенный по сравнению с stunnel перечень опций, подробнее см. табл. 2

Таблица 1. Опции stunnel (Windows)

Параметр	**Описание**
	Global options
debug = [facility.]level	Уровень протоколирования: • emerg (0) • alert (1) • crit (2) • err (3) • warning (4) • notice (5) • info (6) • debug (7) Пример: debug = debug или debug = 7. По умолчанию: debug = notice. Все сообщения с заданным уровнем и уровнем ниже заданного будут запротоколированы. Параметры facility и level регистронезависимы.
output = file	Писать лог в file.
service = servicename	Имя службы NT для контрольной панели. По умолчанию: stunnel.
socket = a|l|r:option=value[:value]	Опции setsockopt() для сокета приема соединений (accept), локального (local) или удаленного (remote) сокетов. Формат значений для SO_LINGER: l_onof:l_linger. Формат значений для времени: tv_sec:tv_usec. Примеры: socket = l:SO_LINGER=1:60 — установить минутный таймаут для локального сокета socket = r:TCP_NODELAY=1 — выключить алгоритм Нейгла для удаленного сокета socket = a:SO_REUSEADDR=0 — запретить повторное использование портов TCP (по умолчанию разрешено) socket = a:SO_BINDTODEVICE=lo — принимать соединения только с loopback интерфейса

taskbar = yes | no	Разрешить использование иконки. По умолчанию: yes.
Service-level options	
Каждая секция в конфигурационном файле начинается с имени службы в квадратных скобках. Имя службы дает возможность различать протокольные сообщения.	
accept = [host:]port	Принимать соединения только с host:port. Если host не указан, то для всех адресов IPv4 данного компьютера.
cert = cert_file	Путь к сертификату в DER-кодировке. Соответствующий сертификат в хранилище должен иметь ссылку на закрытый ключ.
client = yes | no	Режим клиента (удаленный сервис использует TLS/SSL). По умолчанию: no (режим сервера).
connect = [host:]port	Соединять с удаленным сервером host:port. Если host не указан — с локальным хостом
delay = yes | no	Задержка для DNS запроса для опции connect. По умолчанию: no.
local = host	IP-адрес интерфейса, который должен быть использован для соединения с удаленным хостом.
verify = level	Уровень проверки сертификата. Отсутствие опции означает не требовать и не проверять сертификат удалённой стороны (допускается только в тестовых целях в случае клиента или для одностороннего TLS в случае сервера). Возможные значения: • 0 — Запрашивать сертификат, но не проверять его (допускается только в тестовых целях) • 1, 2 — Требовать наличия сертификата удалённой стороны и проверять его • 3 — Требовать наличия сертификата удалённой стороны и проверять его, а также требовать наличия сертификата в хранилище

Таблица 2. Опции stunnel_msspi (Windows)

Параметр	**Описание**
Global options	
iconActive = icon_file (только GUI)	Графическая иконка, отображаемая при наличии установленных подключений. На Windows указывается файл .ico размером 16x16 пикселей.
iconError = icon_file (только GUI)	Графическая иконка, отображаемая при ошибке конфигурации. На Windows указывается файл .ico размером 16x16 пикселей.
iconIdle = icon_file (только GUI)	Графическая иконка, отображаемая при отсутствии установленных подключений. На Windows указывается файл .ico размером 16x16 пикселей.

log = append | overwrite	Обработка log файла. Позволяет выбрать, будет ли log файл (указанный с помощью параметра `output`) добавляться (append) или перезаписываться (overwrite) при открытии или повторном открытии. По умолчанию: append.
output = file	Писать лог в file.
taskbar = yes | no	Разрешить использование иконки. По умолчанию: yes.
Service-level options	
Каждая секция в конфигурационном файле начинается с имени службы в квадратных скобках. Имя службы дает возможность различать протокольные сообщения.	
accept = [host:]port	Принимать соединения только с host:port. Если host не указан, то для всех адресов IPv4 данного компьютера. Для прослушивания всех адресов IPv6: `accept = :::PORT`
CApath = directory	Хранилище доверенных корневых центров сертификации для поиска сертификатов при использовании опции verify = 3.
cert = cert	Цепочка сертификатов, которая используется stunnel для аутентификации на удаленном клиенте или сервере. Помимо пути до файла сертификата может быть указано имя сертификата, идентификатор ключа или отпечаток сертификата. Например, cert = `/path/to/my.example.com.cer` , или cert = my.example.com , или cert = bf3c4aa0255b7c65914a45866d86abbe1c18d512.
cert2 = cert2	Аналогично опции cert, для второго сертификата.
checkHost = host	Хост для выбора сертификата. Допустимо использовать несколько опций checkHost в одной секции, в этом случае отбираются сертификаты, соответствующие хотя бы одному из указанных имен хоста.
checkIP = IP	IP-адрес для выбора сертификата. Допустимо использовать несколько опций checkIP в одной секции, в этом случае отбираются сертификаты, соответствующие хотя бы одному из указанных IP-адресов.
ciphers = cipher_list	Перечень разрешенных алгоритмов TLS (TLSv1.2 и ниже). Опция не влияет на шифр-сюиты TLSv1.3. Несколько значений должны быть разделены двоеточиями.
pin = pin	Пароль от ключевого контейнера закрытого ключа.
pin2 = pin2	Аналогично опции pin, для второго контейнера.
checkSubject = subject	Значение поля Субъект для выбора сертификата, строка в формате X.500 (RFC 2253). Допустимо использовать несколько опций checkSubject в одной секции, в этом случае отбираются сертификаты, соответствующие хотя бы одному из указанных значений.

checkIssuer = issuer	Значение поля Издатель для выбора сертификата, строка в формате X.500 (RFC 2253). Допустимо использовать несколько опций checkIssuer в одной секции, в этом случае отбираются сертификаты, соответствующие хотя бы одному из указанных значений.
client = yes | no	Режим клиента (удаленный сервис использует TLS/SSL). По умолчанию: no (режим сервера).
connect = [host:]port	Соединять с удаленным сервером. Если host не указан — с локальным хостом. Допустимо использовать несколько опций connect в одной секции. Если доступны несколько адресов и/или указаны несколько опций connect, адрес выбирается в соответствии с опцией failover.
debug = [facility.]level	Уровень протоколирования: • emerg (0) • alert (1) • crit (2) • err (3) • warning (4) • notice (5) • info (6) • debug (7) Пример: debug = debug или debug = 7. По умолчанию: debug = notice, используется syslog. Все сообщения с заданным уровнем и уровнем ниже заданного будут запротоколированы. Параметры facility и level регистронезависимы.
delay = yes | no	Задержка для DNS запроса для опции connect. По умолчанию: no.
exec = path	Запустить указанную программу.
execArgs = $0 $1 $2	Аргументы для программы, указанной в exec, включая имя программы ($0). кавычки не поддерживаются. Аргументы разделяются произвольным количеством пробелов.
failover = rr | prio	Метод выбора хоста в случае нескольких целей, удовлетворяющих опции connect. rr — использовать алгоритм распределения нагрузки Round Robin; prio — использовать приоритетный порядок, указанный в конфигурационном файле. По умолчанию: prio.
ident = username	Идентификация пользователя IDENT (RFC 1413)
include = directory	Использовать все части файла конфигурации, расположенные в directory. Используются в алфавитном порядке. Рекомендованные форматы имен: для global options — 00-global.conf; для local service-level options — 01-service.conf, 02-service.conf...
local = host	IP-адрес — источник для удаленных подключений.
logId = id	Идентификатор типа соединения.

redirect = [host:]port	Перенаправлять клиентские подключения TLS при ошибках проверки сертификатов. Эта опция работает только в серверном режиме.
reset = yes | no	Отправить TCP RST в случае ошибки. По умолчанию: yes.
retry yes | no	Повторно исполнить опции connect + exec после отключения. По умолчанию: no.
sni = service_name (режим клиента)	Значение расширения SNI (RFC 3546). Пустой SERVICE_NAME отключает отправку расширения SNI.
socket = a|l|r:option=value[:value]	Опции setsockopt() для сокета приема соединений (accept), локального (local) или удаленного (remote) сокетов. Формат значений для SO_LINGER: l_onof:l_linger. Формат значений для времени: tv_sec:tv_usec. Примеры: socket = l:SO_LINGER=1:60 — установить минутный таймаут для локального сокета socket = r:TCP_NODELAY=1 — выключить алгоритм Нейгла для удаленного сокета socket = a:SO_REUSEADDR=0 — запретить повторное использование портов TCP (по умолчанию разрешено) socket = a:SO_BINDTODEVICE=lo — принимать соединения только с loopback интерфейса
sslVersion = version	Версия TLS: • all • SSLv3 • TLSv1 • TLSv1.1 • TLSv1.2 • TLSv1.3
sslVersionMax = version	Максимально поддерживаемая версия TLS: • all (по умолчанию) • SSLv3 • TLSv1 • TLSv1.1 • TLSv1.2 • TLSv1.3
sslVersionMin = version	Минимально поддерживаемая версия TLS: • all • SSLv3 • TLSv1 (по умолчанию) • TLSv1.1 • TLSv1.2 • TLSv1.3
stack = bytes (кроме fork)	Размер стека потока. По умолчанию: 65536 байт.

TIMEOUTbusy = seconds	Время ожидания данных в секундах.
TIMEOUTclose = seconds	Время ожидания close_notify в секундах.
TIMEOUTconnect = seconds	Время ожидания подключения к удаленному хосту в секундах.
TIMEOUTidle = seconds	Время поддержки неиспользуемого соединения в секундах.
verify = level	Уровень проверки сертификата. Отсутствие опции означает не требовать и не проверять сертификат удалённой стороны (допускается только в тестовых целях в случае клиента или для одностороннего TLS в случае сервера). Возможные значения: • 0 — Запрашивать сертификат, но не проверять его (допускается только в тестовых целях) • 1, 2 — Требовать наличия сертификата удалённой стороны и проверять его • 3 — Требовать наличия сертификата удалённой стороны и проверять его, а также требовать наличия сертификата в хранилище (параметр CApath будет использован в качестве имени хранилища сертификатов, в котором будет осуществлена проверка сертификата)

Примеры файлов конфигурации

Далее приведены примеры файлов конфигурации для клиента и сервера для следующей задачи. Клиент с компьютера comp1 должен установить соединение с веб-сервером (srv1.test.ru), причём трафик должен быть зашифрован и клиент должен быть аутентифицирован по сертификату.

Пример файла конфигурации для сервера:

```
output=c:\stun-srv\stun.log
socket = l:TCP_NODELAY=1
socket = r:TCP_NODELAY=1
debug = 7

[https]
accept=srv1.test.ru:1502
connect = srv1.test.ru:80
cert=C:\stun-srv\srvcer.cer
verify=2
```

Пример файла конфигурации для клиента:

```
output=c:\stun-cli\stun.log
socket = l:TCP_NODELAY=1
```

```
socket = r:TCP_NODELAY=1
debug = 7

[https]
client = yes
accept=comp1:1500
connect = srv1.test.ru:1502
cert=C:\stun-cli\clicer.cer
verify=2
```

1.4 Запуск службы

Запуск, останов и изменение параметров службы запуска осуществляются через стандартную оснастку управления службами (`services.msc`).

1.5 Удаление службы

Удаление службы `stunnel (stunnel_msspi)` производится путём запуска следующей команды:

```
stunnel.exe -remove

stunnel_msspi.exe -remove
```

2 Использование программы в среде UNIX

В состав СКЗИ под управлением ОС Unix входят утилиты `stunnel` и `stunnel_msspi`. Служба `stunnel` представлена в 2 реализациях — с использованием библиотеки pthread и с использованием fork, бинарные файлы называются `stunnel_thread` и `stunnel_fork` соответственно.

2.1 Общий синтаксис

```
stunnel [<filename>] | -fd N | -help | -version | -sockets

stunnel_msspi [<filename>] | -fd N | -help | -version | -sockets | -options
```

<filename>	использовать указанный конфигурационный файл
-fd N	прочитать конфигурационный файл из указанного дескриптора
-help	вывести справку (описание опций конфигурационного файла)
-version	вывести версию stunnel
-sockets	вывести опции сокетов
-options	вывести поддерживаемые опции TLS

2.2 Настройка службы

1. **Выбор варианта использования службы:**
   - в режиме клиента
   - в режиме сервера

В режиме клиента `stunnel` (`stunnel_msspi`) принимает трафик от клиентского приложения, зашифровывает его и отправляет на сервер. На сервере трафик расшифровывается и передаётся конечному приложению или другой службе на этом сервере.

2. **Установка сертификатов**

Установка сертификатов производится при помощи утилит `certmgr` и `cryptcp` из состава СКЗИ.

Для работы службы в режиме сервера обязательно нужен сертификат аутентификации сервера. Сервер может требовать, а может не требовать сертификат клиента при соединении клиента с сервером.

Как на клиенте, так и на сервере нужно установить необходимые сертификаты:

1) сертификат корневого Центра Сертификации (ЦС) должен быть установлен в хранилище ROOT:

```
/opt/cprocsp/bin/<архитектура>/certmgr -inst -file root.cer -store ROOT
```

2) если сертификат сервера или клиента выдан на подчинённом ЦС, сертификаты всех подчинённых ЦС в цепочке должны быть установлены в хранилище CA:

```
/opt/cprocsp/bin/<архитектура>/certmgr -inst -file ca.cer -store CA
```

3) на сервере должен быть установлен сертификат сервера в хранилище My (текущего пользователя или локального компьютера) с привязкой к контейнеру закрытого ключа сервера:

```
/opt/cprocsp/bin/<архитектура>/certmgr -inst -file server.cer -cont '\\.\HDIMAGE\server'
```

4) если сервер требует сертификат клиента, то на клиентском компьютере должен быть установлен сертификат клиента в хранилище My (текущего пользователя или локального компьютера) с привязкой к контейнеру закрытого ключа клиента:

```
/opt/cprocsp/bin/<архитектура>/certmgr -inst -file client.cer -cont '\\.\HDIMAGE\client'
```

3. **Запись сертификатов в файл**

После установки сертификата сервера или клиента в хранилище необходимо дополнительно сохранить этот сертификат в файл на диске в формате DER.

Если сертификат в виде файла отсутствует, его можно сохранить из хранилища или из контейнера при помощи утилиты `certmgr` из состава КриптоПро CSP:

```
/opt/cprocsp/sbin/<архитектура>/certmgr -expr -dest server.cer -cont '\\.\HDIMAGE\server'
```

4. **Формирование файла конфигурации** (см. ниже)

2.2.1 Файл конфигурации

Каждая строка конфигурационного файла может быть:
- пустой строкой (игнорируется);
- комментарием, начинающимся с ';' (игнорируется);
- парой 'option_name = option_value';
- строкой '[service_name]', с которой начинается секция описания конкретной службы.

Параметр адреса любой опции может быть:
- номером порта;
- парой 'IP-адрес:номер порта', где IP-адрес — IP или доменное имя;
- путь к Unix сокету.

Примечание. stunnel_msspi содержит расширенный по сравнению с stunnel перечень опций, подробнее см. табл. 5

Таблица 4. Опции stunnel (Linux)

Параметр	Описание
Global options	
chroot = directory	Каталог вызова функции chroot(), которая вызывается после разбора конфигурационного файла. **Примечание:** • желательно использовать одновременно с опцией setuid; • при создании замкнутого окружения в нем должны быть установлены все внешние объекты CryptoAPI/SSPI. CApath, pid, exec должны находиться в нем. Например, если Вы используете TCP Wrappers, Вы должны скопировать файлы /etc/hosts.allow и /etc/hosts.deny в каталог chroot.

debug = [facility.]level	Уровень протоколирования: • emerg (0) • alert (1) • crit (2) • err (3) • warning (4) • notice (5) • info (6) • debug (7) Пример: debug = debug или debug = 7. По умолчанию: debug = notice, используется syslog. Все сообщения с заданным уровнем и уровнем ниже заданного будут запротоколированы. Параметры facility и level регистронезависимы.
foreground = yes | no	Использование foreground режима. Оставаться в foreground режиме (не использовать fork) и писать протокол в stderr (если не указан output). По умолчанию: no (т.е в режиме демона).
output = file	Писать лог в file, а не в syslog. Допускается указание стандартного потока вывода /dev/stdout.
pid = file	Файл для сохранения pid. Если аргумент не задан, то pid не сохраняется. Если задана опция chroot, то путь указывается относительно ее.
service = servicename	Имя службы. В inetd-режиме это имя для TCP Wrapper библиотеки. По умолчанию: stunnel.
setgid = groupname	Выполняется setgid() для указанной группы.
setuid = username	Выполняется setuid() для указанного пользователя.
socket = a|l|r:option=value[:value]	Опции setsockopt() для сокета приема соединений (accept), локального (local) или удаленного (remote) сокетов. Формат значений для SO_LINGER: l_onof:l_linger. Формат значений для времени: tv_sec:tv_usec. Примеры: socket = l:SO_LINGER=1:60 — установить минутный таймаут для локального сокета socket = r:TCP_NODELAY=1 — выключить алгоритм Нейгла для удаленного сокета socket = a:SO_REUSEADDR=0 — запретить повторное использование портов TCP (по умолчанию разрешено) socket = a:SO_BINDTODEVICE=lo — принимать соединения только с loopback интерфейса
Service-level options	
Каждая секция в конфигурационном файле начинается с имени службы в квадратных скобках. *Имя службы используется библиотекой libwrap (TCP Wrappers), а также дает возможность различать протокольные сообщения.*	

accept = [host:]port	Принимать соединения на host:port
cert = cert_file	Путь к сертификату в DER-кодировке. Соответствующий сертификат в хранилище должен иметь ссылку на закрытый ключ.
client = yes | no	Режим клиента (удаленный сервис использует TLS/SSL). По умолчанию: no (режим сервера).
connect = [host:]port	Соединять с удаленным сервером host:port. Если host не указа — с локальным хостом.
delay = yes | no	Задержка для DNS запроса для опции connect. По умолчанию: no.
local = host	IP-адрес интерфейса, который должен быть использован для соединения с удаленным хостом.
verify = level	Уровень проверки сертификата. Отсутствие опции означает не требовать и не проверять сертификат удалённой стороны (допускается только в тестовых целях в случае клиента или для одностороннего TLS в случае сервера). Возможные значения: • 0 — Запрашивать сертификат, но не проверять его (допускается только в тестовых целях) • 1, 2 — Требовать наличия сертификата удалённой стороны и проверять его • 3 — Требовать наличия сертификата удалённой стороны и проверять его, а также требовать наличия сертификата в хранилище

Таблица 5. Опции stunnel_msspi (Linux)

Параметр	**Описание**
Global options	
chroot = directory	Каталог вызова функции chroot(), которая вызывается после разбора конфигурационного файла. **Примечание:** • желательно использовать одновременно с опцией setuid; • при создании замкнутого окружения в нем должны быть установлены все внешние объекты CryptoAPI/SSPI. CApath, pid, exec должны находиться в нем. Например, если Вы используете TCP Wrappers, Вы должны скопировать файлы /etc/hosts.allow и /etc/hosts.deny в каталог chroot.
foreground = yes | quiet | no	Использование foreground режима. Оставаться в foreground режиме (не использовать fork) и писать протокол в stderr (если не указан output). По умолчанию: no (т.е в режиме демона).

log = append | overwrite	Обработка log файла. Позволяет выбрать, будет ли log файл (указанный с помощью параметра `output`) добавляться (append) или перезаписываться (overwrite) при открытии или повторном открытии. По умолчанию: append.
output = file	Писать лог в file, а не в syslog. Допускается указание стандартного потока вывода /dev/stdout.
pid = file	Файл для сохранения pid. Если аргумент не задан, то pid не сохраняется. Если задана опция chroot, то путь указывается относительно ее.
syslog = yes | no	Включить протоколирование в syslog. По умолчанию: yes.
Service-level options	
Каждая секция в конфигурационном файле начинается с имени службы в квадратных скобках. *Имя службы используется библиотекой libwrap (TCP Wrappers), а также дает возможность различать протокольные сообщения.*	
accept = [host:]port	Принимать соединения только с host:port. Если host не указан, то для всех адресов IPv4 данного компьютера. Для прослушивания всех адресов IPv6: `accept = :::PORT`
CApath = directory	Хранилище доверенных корневых центров сертификации для поиска сертификатов при использовании опций verify = 3. Если задана опция chroot, то путь указывается относительно её.
cert = cert	Цепочка сертификатов, которая используется stunnel для аутентификации на удаленном клиенте или сервере. Помимо пути до файла сертификата может быть указано имя сертификата, идентификатор ключа или отпечаток сертификата. Например, cert = `/path/to/my.example.com.cer` , или cert = my.example.com , или cert = bf3c4aa0255b7c65914a45866d86abbe1c18d512.
cert2 = cert2	Аналогично опции cert, для второго сертификата.
checkHost = host	Хост для выбора сертификата. Допустимо использовать несколько опций checkHost в одной секции, в этом случае отбираются сертификаты, соответствующие хотя бы одному из указанных имен хоста.
checkIP = IP	IP-адрес для выбора сертификата. Допустимо использовать несколько опций checkIP в одной секции, в этом случае отбираются сертификаты, соответствующие хотя бы одному из указанных IP-адресов.
ciphers = cipher_list	Перечень разрешенных алгоритмов TLS (TLSv1.2 и ниже). Опция не влияет на шифр-сюиты TLSv1.3. Несколько значений должны быть разделены двоеточиями.
pin = pin	Пароль от ключевого контейнера закрытого ключа.
pin2 = pin2	Аналогично опции pin, для второго контейнера.

checkSubject = subject	Значение поля Субъект для выбора сертификата, строка в формате X.500 (RFC 2253). Допустимо использовать несколько опций checkSubject в одной секции, в этом случае отбираются сертификаты, соответствующие хотя бы одному из указанных значений.
checkIssuer = issuer	Значение поля Издатель для выбора сертификата, строка в формате X.500 (RFC 2253). Допустимо использовать несколько опций checkIssuer в одной секции, в этом случае отбираются сертификаты, соответствующие хотя бы одному из указанных значений.
client = yes | no	Режим клиента (удаленный сервис использует TLS/SSL). По умолчанию: no (режим сервера).
connect = [host:]port	Соединять с удаленным сервером. Если host не указан — с локальным хостом. Допустимо использовать несколько опций connect в одной секции. Если доступны несколько адресов и/или указаны несколько опций connect, адрес выбирается в соответствии с опцией failover.
debug = [facility.]level	Уровень протоколирования: • emerg (0) • alert (1) • crit (2) • err (3) • warning (4) • notice (5) • info (6) • debug (7) Пример: debug = debug или debug = 7. По умолчанию: debug = notice, используется syslog. Все сообщения с заданным уровнем и уровнем ниже заданного будут запротоколированы. Параметры facility и level регистронезависимы.
delay = yes | no	Задержка для DNS запроса для опции connect. По умолчанию: no.
exec = path	Запустить указанную программу. Если задана опция chroot, то путь указывается относительно её.
execArgs = $0 $1 $2	Аргументы для программы, указанной в exec, включая имя программы ($0). кавычки не поддерживаются. Аргументы разделяются произвольным количеством пробелов.
failover = rr | prio	Метод выбора хоста в случае нескольких целей, удовлетворяющих опции connect. rr — использовать алгоритм распределения нагрузки Round Robin; prio — использовать приоритетный порядок, указанный в конфигурационном файле. По умолчанию: prio.
ident = username	Идентификация пользователя IDENT (RFC 1413)

include = directory	Использовать все части файла конфигурации, расположенные в directory. Используются в алфавитном порядке. Рекомендованные форматы имен: для global options — 00-global.conf; для local service-level options — 01-service.conf, 02-service.conf...
local = host	IP-адрес — источник для удаленных подключений.
logId = id	Идентификатор типа соединения.
pty = yes | no	Использовать псевдотерминал для опции exec.
redirect = [host:]port	Перенаправлять клиентские подключения TLS при ошибках проверки сертификатов. Эта опция работает только в серверном режиме.
reset = yes | no	Отправить TCP RST в случае ошибки. По умолчанию: yes.
retry yes | no	Повторно исполнить опции connect + exec после отключения. По умолчанию: no.
service = servicename	Имя службы. По умолчанию: stunnel.
setgid = groupname	Если global option - выполнить setgid() для указанной группы и очистить все остальные группы. Если service-level option — выполнить setgid() для сокета, указанного в accept.
setuid = username	Если global option - выполнить setuid() для указанного пользователя. Если service-level option — установить владельца для сокета, указанного в accept.
sessiond = [host:]port	Адрес sessiond для кэширования TLS.
sni = service_name (режим клиента)	Значение расширения SNI (RFC 3546). Пустой SERVICE_NAME отключает отправку расширения SNI.
socket = a|l|r:option=value[:value]	Опции setsockopt() для сокета приема соединений (accept), локального (local) или удаленного (remote) сокетов. Формат значений для SO_LINGER: l_onof:l_linger. Формат значений для времени: tv_sec:tv_usec. Примеры: socket = l:SO_LINGER=1:60 — установить минутный таймаут для локального сокета socket = r:TCP_NODELAY=1 — выключить алгоритм Нейгла для удаленного сокета socket = a:SO_REUSEADDR=0 — запретить повторное использование портов TCP (по умолчанию разрешено) socket = a:SO_BINDTODEVICE=lo — принимать соединения только с loopback интерфейса

sslVersion = version	Версия TLS: • all • SSLv3 • TLSv1 • TLSv1.1 • TLSv1.2 • TLSv1.3
sslVersionMax = version	Максимально поддерживаемая версия TLS: • all (по умолчанию) • SSLv3 • TLSv1 • TLSv1.1 • TLSv1.2 • TLSv1.3
sslVersionMin = version	Минимально поддерживаемая версия TLS: • all • SSLv3 • TLSv1 (по умолчанию) • TLSv1.1 • TLSv1.2 • TLSv1.3
stack = bytes (кроме fork)	Размер стека потока. По умолчанию: 65536 байт.
TIMEOUTbusy = seconds	Время ожидания данных в секундах.
TIMEOUTclose = seconds	Время ожидания close_notify в секундах.
TIMEOUTconnect = seconds	Время ожидания подключения к удаленному хосту в секундах.
TIMEOUTidle = seconds	Время поддержки неиспользуемого соединения в секундах.
transparent = none | source | destination | both	Прозрачная поддержка прокси.

verify = level	Уровень проверки сертификата. Отсутствие опции означает не требовать и не проверять сертификат удалённой стороны (допускается только в тестовых целях в случае клиента или для одностороннего TLS в случае сервера). Возможные значения: • 0 — Запрашивать сертификат, но не проверять его (допускается только в тестовых целях) • 1, 2 — Требовать наличия сертификата удалённой стороны и проверять его • 3 — Требовать наличия сертификата удалённой стороны и проверять его, а также требовать наличия сертификата в хранилище (параметр CApath будет использован в качестве имени хранилища сертификатов, в котором будет осуществлена проверка сертификата)

Примечание. Описание всех доступных в конфигурационном файле опций можно найти, вызвав в консоли `man stunnel (stunnel_msspi)`.

Далее приведены примеры файлов конфигурации клиента и сервера для следующей задачи. Клиент с компьютера comp1 должен установить соединение с веб-сервером (srv1.test.ru), причём трафик должен быть зашифрован и клиент должен быть аутентифицирован по сертификату.

Пример файла конфигурации для сервера

```
pid=/var/opt/cprocsp/tmp/stunnel_serv.pid
output=/var/opt/cprocsp/tmp/stunnel_serv.log
socket = l:TCP_NODELAY=1
socket = r:TCP_NODELAY=1
debug = 7

[https]
accept=srv1.test.ru:1502
connect = srv1.test.ru:80
cert=/etc/stunnel/server.cer
verify=2
```

Пример файла конфигурации для клиента

```
pid=/var/opt/cprocsp/tmp/stunnel_cli.pid
output=/var/opt/cprocsp/tmp/stunnel_cli.log
socket = l:TCP_NODELAY=1
socket = r:TCP_NODELAY=1
debug = 7

[https]
client = yes
accept=comp1:1500
connect = srv1.test.ru:1502
```

```
cert=/etc/stunnel/client.cer
verify=2
```

2.3 Запуск службы

Запуск службы производится следующей командой:

```
/opt/cprocsp/sbin/<архитектура>/stunnel "путь к файлу конфигурации"
```

```
/opt/cprocsp/sbin/<архитектура>/stunnel_msspi "путь к файлу конфигурации"
```

2.4 Остановка службы

Для остановки необходимо завершить процесс `stunnel (stunnel_msspi)`.