



<p>Средство Криптографической Защиты Информации</p>	<p>КриптоПро CSP</p> <p>Версия 4.0 КС1</p> <p>Приложение для создания TLS-туннеля</p>
---	---

ЖТЯИ.00087-01 93 03

Листов 9

2016 г.

© ООО «КРИПТО-ПРО», 2000-2016. Все права защищены.

Авторские права на средства криптографической защиты информации типа «КриптоПро CSP» и эксплуатационную документацию зарегистрированы в Российском агентстве по патентам и товарным знакам (Роспатент).

Настоящий документ входит в комплект поставки программного обеспечения СКЗИ «КриптоПро CSP» версии 4.0; на него распространяются все условия лицензионного соглашения. Без специального письменного разрешения ООО «КРИПТО-ПРО» документ или его часть в электронном или печатном виде не могут быть скопированы и переданы третьим лицам с коммерческой целью.

Содержание

1. Системные требования	3
2. Использование программы на операционной системе Windows	4
2.1. Установка службы stunnel.....	4
2.2. Настройка службы stunnel.....	4
2.2.1.Выбор варианта использования.....	4
2.2.2.Установка сертификатов	4
2.2.3.Запись сертификатов в файл.....	4
2.2.4.Формирование файла конфигурации.....	4
2.2.4.1.Пример файла конфигурации для сервера.....	5
2.2.4.2.Пример файла конфигурации для клиента	5
2.3.Запуск службы	6
2.4.Удаление службы.....	6
3.Использование программы в среде UNIX.....	6
3.1. Реализации stunnel.....	6
3.2.Настройка stunnel.....	6
3.2.1.Выбор варианта использования	6
3.2.2.Установка сертификатов	6
3.2.3.Запись сертификатов в файл.....	7
3.2.4.Формирование файла конфигурации.....	7
3.2.4.1.Пример файла конфигурации для сервера.....	8
3.2.4.2.Пример файла конфигурации для клиента	8
3.2.5.Запуск службы	9

ЖТЯИ.00087-01 93 03. Приложение для создания TLS-туннеля.doc
Аннотация

Данный документ содержит общую информацию по использованию программного продукта «ЖТЯИ.00087-01 93 03. Приложение для создания TLS-туннеля», предназначенного для создания TLS защищенного соединения между клиентом и локальным (inetd-запускаемым) или удаленным сервером.

1. Системные требования

Windows

Включает программно-аппаратные среды:

Windows 7/8/8.1/Server 2003/2008 (x86, x64);

Windows Server 2008 R2/2012/2012 R2 (x64).

LSB Linux

Включает дистрибутивы Linux, удовлетворяющие стандарту Linux Standard Base ISO/IEC 23360 версии LSB 4.x:

CentOS 4/5/6/7 (x86, x64, POWER, ARM);

ТД ОС АИС ФССП России (GosLinux) (x86, x64);

Red OS (x86, x64);

Fedora 19/20 (x86, x64, ARM);

Mandriva Enterprise Server 5, Business Server 1 (x86, x64, ARM);

Oracle Linux 4/5/6/7 (x86, x64);

OpenSUSE 13.2, Leap 42 (x86, x64, ARM);

SUSE Linux Enterprise Server 11SP4/12, Desktop 12 (x86, x64, POWER, ARM);

Red Hat Enterprise Linux 4/5/6/7 (x86, x64, POWER, ARM);

Синтез-ОС.РС (x86, x64, POWER, ARM);

Ubuntu 10.04/12.04/12.10/13.04/14.04/14.10 (x86, x64, POWER, ARM);

Linux Mint 13/14/15/16/17 (x86, x64);

Debian 7/8 (x86, x64, POWER, ARM);

Astra Linux Special Edition (x86-64).

Unix

Включает программно-аппаратные среды:

ALT Linux 7 (x86, x64, ARM);

ROSA 2011, Enterprise Desktop X.1 (Marathon), Enterprise Linux Server (x86, x64);

РОСА ХРОМ/КОБАЛЬТ/НИКЕЛЬ (x86, x64);

FreeBSD 9, pfSense 2.x (x86, x64);

AIX 5/6/7 (POWER);

Mac OS X 10.7/10.8/10.9/10.10/10.11 (x64).

Solaris

Включает программно-аппаратные среды:

Solaris 10 (sparc, x86, x64);

Solaris 11 (sparc, x64).

2. Использование программы на операционной системе Windows

2.1. Установка службы stunnel

Установка делается путём запуска

stunnel.exe –install

В дальнейшем служба для старта будет использовать файл stunnel.exe из той папки, откуда была проведена установка.

Перед установкой нужно выбрать режим работы службы, установить сертификаты и сформировать файл конфигурации (см. далее по тексту данного документа).

2.2. Настройка службы stunnel

2.2.1. Выбор варианта использования

Службу stunnel можно использовать либо в режиме клиента, либо в режиме сервера. В режиме клиента stunnel 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.

2.2.2. Установка сертификатов

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:

- а) сертификат корневого Центра Сертификации (ЦС) – в хранилище «Доверенные корневые Центры Сертификации» локального компьютера;
- б) если сертификат сервера или клиента выдан на подчинённом ЦС - сертификаты всех подчиненных ЦС в цепочке должны быть установлены в хранилище «Промежуточные Центры Сертификации» локального компьютера;
- в) на сервере должен быть установлен сертификат сервера в хранилище «Личные» локального компьютера с привязкой к контейнеру закрытого ключа сервера;
- г) 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

2.2.3. Запись сертификатов в файл

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

2.2.4. Формирование файла конфигурации

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.

В файл конфигурации заносятся следующие опции:

Параметр	Описание
debug	Уровень протоколирования.
output	Писать лог в file
service	Имя сервиса.
socket	Опции setsockopt() для сокета приема соединений, а так же для локального и удаленного сокетов.
Service-mode options.	
accept	Принимать соединения на host:port.
Cert	Сертификат в der кодировке. Соответствующий сертификат в хранилище должен иметь ссылку на закрытый ключ.
client	Режим клиента (удаленный сервис использует TLS/SSL).
connect	Соединять с удаленным сервером host:port
delay	Задержка для DNS запроса для 'connect' опции.
verify	Уровень проверки сертификата удаленного компьютера 0 — Игнорировать сертификат 1 — Проверять сертификат если есть 2 — Всегда проверять сертификат 3 — Проверять наличие сертификата в хранилище TrustedUsers

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.

2.2.4.1. Пример файла конфигурации для сервера

```
output=c:\stun-srv\stun.log
socket = !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
```

2.2.4.2. Пример файла конфигурации для клиента

```
output=c:\stun-cl\stun.log
socket = !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
```

2.3.Запуск службы

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

2.4.Удаление службы

Удаление службы делается путём запуска
stunnel.exe -remove

3.Использование программы в среде UNIX

3.1. Реализации stunnel

Существует две реализации службы stunnel: с использованием библиотеки pthread и с использованием fork, бинарные файлы называются stunnel_thread и stunnel_fork соответственно. Stunnel с использованием fork возможно использовать только с КриптоПро CSP исполнение KC2.

3.2.Настройка stunnel

3.2.1.Выбор варианта использования

Службу stunnel можно использовать либо в режиме клиента, либо в режиме сервера. В режиме клиента stunnel 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.

3.2.2.Установка сертификатов

Установка сертификатов производится при помощи утилит certmgr и cscptcp из состава КриптоПро CSP.

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:

а) сертификат корневого Центра Сертификации (ЦС) – в хранилище *ROOT*;
/opt/cproccsp/bin/<архитектура>/certmgr -inst -file root.cer -store ROOT

ЖТЯИ.00087-01 93 03. Приложение для создания TLS-туннеля.doc

б) если сертификат сервера или клиента выдан на подчинённом ЦС - сертификаты всех подчиненных ЦС в цепочке должны быть установлены в хранилище CA;

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

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

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

г) 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/cproccsp/bin/<архитектура>/certmgr -inst -file client.cer -cont '\\.\HDIMAGE\client'
```

3.2.3.Запись сертификатов в файл

После установки сертификата сервера или клиента в хранилище нужно 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/cproccsp/bin/<архитектура>/certmgr -expr -dest server.cer -cont '\\.\HDIMAGE\server'
```

3.2.4.Формирование файла конфигурации

Параметр	Описание
chroot	Каталог вызова функции chroot(), которая вызывается после разбора конфигурационного файла stunnel.
debug	Уровень протоколирования.
foreground	foreground режим.
output	Писать лог в file, а не в syslog.
Pid	Файл для сохранения pid.
service	Имя сервиса.
Setgid	Выполняется setgid() в эту группу.
Setuid	Выполняется setuid() под этого пользователя
socket	Опции setsockopt() для сокета приема соединений, а так же для локального и удаленного сокетов.
Service-mode options.	
accept	Принимать соединения на host:port.
cert	Сертификат в der кодировке. Соответствующий сертификат в хранилище должен иметь ссылку на закрытый ключ.
client	Режим клиента (удаленный сервис использует TLS/SSL).

connect	Соединять с удаленным сервером host:port
delay	Задержка для DNS запроса для 'connect' опции.
local	Интерфейс, который должен быть использован для соединения с удаленным хостом.
Verify	Уровень проверки сертификата удаленного компьютера 0 — Игнорировать сертификат 1 — Проверять сертификат если есть 2 — Всегда проверять сертификат 3 — Проверять наличие сертификата в хранилище TrustedUsers

Для более подробного описания опций и примеров их использования используйте команду *man stunnel*

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.

3.2.4.1.Пример файла конфигурации для сервера

```
pid=/var/opt/cproccsp/tmp/stunnel_serv.pid
output=/var/opt/cproccsp/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
```

3.2.4.2.Пример файла конфигурации для клиента

```
pid=/var/opt/cproccsp/tmp/stunnel_cli.pid
output=/var/opt/cproccsp/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
```

Описание всех доступных в конфигурационном файле опций можно найти, вызвав в консоли *man stunnel*

3.3.Запуск службы

Запуск службы производится командой

/opt/cprosp/sbin/<архитектура>/stunnel_thread "путь к файлу конфигурации"

Для остановки необходимо завершить процесс stunnel.