127 018, Москва, Сущевский вал, д.18
Телефон: (495) 995 4820
Факс: (495) 995 4820
http://www.CryptoPro.ru
E-mail: info@CryptoPro.ru

Средство Криптографической Защиты Информации

КриптоПро CSP

Версия 4.0 R4 KC2

Приложение для создания TLS-туннеля

ЖТЯИ.00088-03 93 03

Листов 11

2018 г.

Содержание

Аннотация

Данный документ содержит общую информацию по использованию программного продукта «ЖТЯИ.00088-03 93 03. Приложение для создания TLS-туннеля», предназначенного для создания TLS защищенного соединения между клиентом и локальным (Iinetd-запускаемым) или удаленным сервером.

1. Системные требования

Windows

Включает программно-аппаратные среды:

Windows XP[1] (x86);
Windows 7/8/8.1/10/Server 2003/2008 (x86, x64);
Windows Server 2008 R2/2012/2012 R2/2016 (x64).

LSB Linux

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

CentOS 4/5/6 (x86, x64);
CentOS 7 (x86, x64, POWER, ARM, ARM64);
ОСь (OS-RT) (x64);
ТД ОС АИС ФССП России (GosLinux) (x86, x64);
Red OS (x86, x64);
Fedora 27/28/29 (x86, x64, ARM);
Oracle Linux 4/5/6 (x86, x64);
Oracle Linux 7 (x64);
OpenSUSE Leap 42, 15 (x86, x64, ARM, ARM64);
AlterOS (x64);
SUSE Linux Enterprise Server 11SP4 (x86, x64);
SUSE Linux Enterprise Server 12/15, Desktop 12/15 (x64, POWER, ARM64);
Red Hat Enterprise Linux 4/5/6 (x86, x64);
Red Hat Enterprise Linux 7 (x64, POWER, ARM64);
Синтез-ОС.РС (x86, x64);
КП «ЗОС «СинтезМ» в роли «Графический клиент» (x64);
КП «ОС «СинтезМ-К» в роли «Графический клиент» (x64);
Ubuntu 14.04/16.04 (x86, x64, POWER, ARM, ARM64);
Ubuntu 18.04/18.10 (x86, x64);
Linux Mint 17/18/19 (x86, x64);
Debian 7/8/9 (x86, x64, POWER, ARM, ARM64);
ОС Лотос (x86, x64);
Astra Linux Special Edition, Common Edition (x64, Эльбрус);
МСВСфера 6.3 Сервер (x64, ARM64).

Unix

Включает программно-аппаратные среды:

ОС Эльбрус версия 3 (Эльбрус);
ALT Linux 6/7 (x86, x64, ARM);
Альт Сервер 8, Альт 8 СП Сервер (x86, x64, ARM, ARM64);
Альт Рабочая станция 8, Альт Рабочая станция К 8, Альт 8 СП Рабочая станция (x86, x64, ARM, ARM64);

ROSA Fresh, Enterprise Desktop, Enterprise Linux Server (x86, x64);
РОСА ХРОМ/КОБАЛЬТ/НИКЕЛЬ (x86, x64);
FreeBSD 11, pfSense 2.x (x86, x64);
AIX 6/7 (POWER).

Solaris

Включает программно-аппаратные среды:

Solaris 10 (sparc, x86, x64);
Solaris 11 (sparc, x64).

Примечания:

1. Версия POSReady.

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 = 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
```

2.2.4.2.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
```

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 КС2.

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

б) если сертификат сервера или клиента выдан на подчинённом ЦС - сертификаты всех подчиненных ЦС в цепочке должны быть установлены в хранилище *CA*;
/opt/cprocsp/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/cprocsp/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/cprocsp/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/cprocsp/sbin/<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/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
```

3.2.4.2.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

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

3.3.Запуск службы

Запуск службы производится командой
/opt/cprocsp/sbin/<архитектура>/stunnel_thread "путь к файлу конфигурации"

Для остановки необходимо завершить процесс stunnel.