

Правила использования программы Stunnel на ОС Linux

Программа stunnel предназначена для шифрования трафика между произвольным приложением на клиентском компьютере, которое работает с некоторым приложением или службой на удаленном компьютере (сервере). Шифрование делается между двумя экземплярами программы stunnel (на сервере и на клиенте) без необходимости вносить изменения в работу клиентского или серверного ПО. Кроме шифрования, можно настроить требование аутентификации клиента по сертификату клиента.

1. Установка пакета stunnel

Установка stunnel должна производиться после установки и настройки КриптоПро CSP.

Для работы с stunnel необходимо использовать CSP в исполнении KC2.

Дистрибутив поставляется в виде пакета cprosp-stunnel типа rpm.

Для его установки используются стандартные средства для установки rpm из состава дистрибутива. Для дистрибутивов Linux, основанных на rpm это утилита rpm:

```
rpm -i cprosp-stunnel-3.6.1-4.i486.rpm
```

Для дистрибутивов, основанных на deb, это утилита alien:

```
alien -kci cprosp-stunnel-3.6.1-4.i486.rpm
```

После установки пакета бинарный файл, предназначенный для запуска stunnel, будет помещён в /opt/cprosp/sbin/<архитектура> .

2. Настройка stunnel

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. Установка сертификатов

Установка сертификатов производится при помощи утилит certmgr и cryptsp 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;
- б) если сертификат сервера или клиента выдан на подчинённом ЦС - сертификаты всех подчиненных ЦС в цепочке должны быть установлены в хранилище CA;
- в) на сервере должен быть установлен сертификат сервера в хранилище My (текущего пользователя или 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 My (текущего пользователя или локального компьютера) с привязкой к контейнеру закрытого ключа клиента.

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.

2.4. Формирование файла конфигурации

В файл конфигурации заносятся следующие данные:

Параметр	Описание
<code>pid</code>	расположение файла <code>pid</code>
<code>accept</code>	ip-адрес (или символическое имя хоста) и порт, на которых <code>stunnel</code> принимает трафик
<code>connect</code>	ip-адрес (или символическое имя хоста) и порт, с которых <code>stunnel</code> передает трафик
<code>cert</code>	имя файла сертификата сервера или клиента
<code>client</code>	режим работы службы: клиент (значение «yes») или сервер (значение «no» или отсутствие параметра)
<code>mutual_auth</code>	для сервера: признак того, требуется ли сертификат клиента (значение «yes») или не требуется (значение «no»)
<code>socket</code>	дополнительные параметры TCP
<code>output</code>	имя файла журнала
<code>debug</code>	степень подробности журналирования: 0 – ничего не записывать, 1 – только ошибки, 3 – ошибки и предупреждения, 7 – все сообщения

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.

2.4.1. Пример файла конфигурации для сервера

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

2.4.2. Пример файла конфигурации для клиента

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

Если сертификат клиента не требуется, то в файл конфигурации сервера нужно добавить строку (в секцию [https])

```
mutual_auth = no
```

3. Запуск службы

Запуск службы производится командой

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

Для остановки необходимо завершить процесс stunnel.

4. Удаление пакета

Удаление пакета производится стандартными средствами операционной системы, предназначенными для удаления rpm. Для дистрибутивов, основанных на rpm:

```
rpm -e cprosp-stunnel
```

Для дистрибутивов, основанных на deb:

```
dpkg -P cprosp-stunnel
```