7.6.13 Команда `pkcs12`

7.6.13.1 Описание команды

Команда pkcs12 позволяет создавать и интерпретировать файлы формата PKCS#12 (иногда называемые файлами формата PFX). Файлы формата PKCS#12 используются для переноса между компьютерами закрытых ключей и соответствующих им сертификатов.

7.6.13.2 Формат ввода команды

openssl pkcs12 [-export] [-chain] [-inkey filename] [-certfile filename] [-name name] [-caname name] [-in filename] [-out filename] [-noout] [-nomacver] [-nocerts] [-clcerts] [-cacerts] [-nokeys] [-info] [-gost89 | -nodes] [-noiter] [-maciter | -nomaciter | -nomac] [-twopass] [-descert] [-certpbe cipher] [-keypbe cipher] [-macalg digest] [-keyex] [-keysig] [-password arg] [-passin arg] [-passout arg] [-CAfile file] [-CApath dir] [-CSP name]

7.6.13.3 Опции команды

У данной команды имеется множество опций, значение некоторых зависит от того, создается или интерпретируется файл формата PKCS#12. По умолчанию считается, что файл интерпретируется. Можно создать файл формата PKCS#12, используя опцию -export (см. ниже).

7.6.13.3.1 Опции интерпретирования файлов

Опция	**Описание**
-in filename	Определяет имя интерпретируемого файла формата PKCS#12. По умолчанию файл считывается со стандартного ввода
-out filename	Файл для записи сертификатов и закрытых ключей, по умолчанию - стандартный вывод. Всё записывается в формате PEM

Опция	Описание
-password arg, -passin arg	Указывает, где содержится пароль для входного файла (т.е. файла формата PKCS#12). Для получения дополнительной информации по формату аргумента arg см. раздел 7.4
-passout arg	Источник пароля для зашифровывания любых полученных закрытых ключей. Для получения дополнительной информации по формату аргумента arg см. раздел 7.4.
-noout	Отменяет вывод ключей и сертификатов (однако будет проверена корректность входного файла и возможность его распаковки)
-clcerts	Выводит только клиентские сертификаты (не выводит сертификаты УЦ)
-cacerts	Выводит только сертификаты УЦ (не выводит клиентские сертификаты)
-nocerts	Отменяет вывод любых сертификатов
-nokeys	Отменяет вывод закрытых ключей
-info	Выводит дополнительную информацию о структуре файла формата PKCS#12, использованных алгоритмах и количестве итераций
-gost89	Использовать алгоритм ГОСТ 28147-89 для зашифрования закрытых ключей перед выводом (при импорте ключей для российских криптографических алгоритмов следует использовать именно этот алгоритм шифрования)
-des	Использовать алгоритм DES для шифрования закрытых ключей перед выводом
-des3	Использовать алгоритм тройной DES для шифрования закрытых ключей перед выводом. Используется по умолчанию
-idea	Использовать алгоритм IDEA для шифрования закрытых ключей перед выводом
-aes128, -aes192, -aes256	Использовать алгоритм AES для шифрования закрытых ключей перед выводом
-camellia128, -camellia192, -camellia256	Использовать алгоритмы Camellia для шифрования закрытых ключей перед выводом
-nodes	Выводить закрытые ключи в незашифрованном виде
-nomacver	Не проверять целостность контейнера

Опция	Описание
-twopass	Запрашивать у пользователя отдельные пароли для контроля целостности и для расшифрования

7.6.13.3.2 Опции создания файлов

Опция	Описание
-export	Указывает, что файл формата PKCS#12 будет создан, а не интерпретирован
-out filename	Указывает имя создаваемого файла формата PKCS#12. По умолчанию используется стандартный вывод
-in filename	Указывает имя входного файла, из которого следует считать сертификаты и закрытые ключи, по умолчанию используется стандартный вход. Они все должны быть в формате PEM. Порядок значения не имеет, но должен присутствовать один закрытый ключ и соответствующий ему сертификат. Если присутствуют дополнительные сертификаты, они также будут включены в файл формата PKCS#12
-inkey filename	Файл, из которого считывается закрытый ключ. Если эта опция не указана, закрытый ключ должен присутствовать во входном файле
-name friendlyname	Указывает «дружественное имя» для сертификата и закрытого ключа. Обычно это имя указывается в выпадающих списках приложений, импортирующих файл
-certfile filename	имя файла, из которого следует считать дополнительные сертификаты
-caname friendlyname	Указывает «дружественное имя» для других сертификатов. Эта опция может быть использована несколько раз, чтобы определить имена всех сертификатов в порядке их появления. Netscape игнорирует дружественные имена других сертификатов, а MSIE их показывает
-password arg, -passout arg	Указывает, где содержится пароль для выходного файла (т.е. файла формата PKCS#12). Для получения дополнительной информации по формату аргумента arg см. раздел 7.4
-passin password	Источник пароля для расшифровывания и ввода закрытых ключей. Для получения дополнительной информации по формату аргумента arg см. раздел 7.4

Опция	Описание
-chain	Если эта опция указана, делается попытка включить всю цепочку сертификатов для пользовательского сертификата. Для поиска сертификатов используется стандартное хранилище сертификатов УЦ. Если найти все необходимые сертификаты не удается, это считается фатальной ошибкой
-keypbe alg, -certpbe alg	Эти опции позволяют выбрать алгоритм, используемый для зашифрования закрытого ключа и сертификатов. При экспорте ключей российских криптографических алгоритмов Следует указать gost89 в качестве параметра alg
-keyex|-keysig	Указывает, следует ли использовать закрытый ключ для обмена ключами или только для подписывания. Эта опция интерпретируется только MS Internet Explorer и подобными приложениями MS. Опция -keysig указывает, что ключ можно использовать только для подписи. Ключи, используемые только для подписи, можно использовать в подписывании сообщений формата S/textbackslash MIME, контрольной подписи Active X и в аутентификации SSL-клиента, хотя из-за ошибки в коде только Internet Explorer начиная с версии 5.0 поддерживает использование ключей только для подписи для аутентификации SSL-клиента
-macalg digest	Указывает алгоритм дайджеста, используемого для контроля целостности контейнера. При экспорте ключей российских криптографических алгоритмов следует указать md_gost12_512, md_gost12_256 или md_gost94 (последний только в режиме совместимости со старыми версиями, см. раздел 7.6.13.6
-nomaciter, -noiter	Устанавливают число итераций при вычислении кода аутентификации и ключа шифрования в значение 1. Эти опции следует использовать только в том случае, если вы хотите создать файлы, совместимые с Internet Explorer 4.0
-maciter	Устанавливает число итераций при вычислении кода аутентификации в умолчательное значение (2048). Опция включена для совместимости с предыдущими версиями OpenSSL.
-nomac	Создать контейнер без контроля целостности
-CAfile file	Файл, содержащий сертификаты доверенных удостоверяющих центров
-CApath dir	Каталог, содержащий сертификаты доверенных удостоверяющих центров. Этот каталог должен быть стандартным каталогом сертификатов, то есть с каждым сертификатом должна быть связана хэш-сумма поля subject name

Опция	Описание
-CSP name	Записывает значение параметра name в формате Microsoft CSP
-twopass	Запрашивает у пользователя отдельные пароли для контроля целостности и для зашифрования: большая часть приложений всегда предполагает, что они совпадают, так что эта опция лишает их возможности прочесть создаваемый файл формата PKCS#12

7.6.13.4 Примечания

Хотя у этой команды существует большое количество опций, большинство из них используется очень редко. Для интерпретации файла формата PKCS#12 достаточно использовать только опции -in и -out, для создания - также опции -export и -name.

Если не указана ни одна из опций -clcerts, -cacerts или -nocerts, все сертификаты будут выведены в порядке, в котором они находятся во входных файлах формата PKCS#12. Нет гарантии, что первый присутствующий сертификат соответствует закрытому ключу. Некоторые приложения, требующие закрытый ключ и сертификат, предполагают, что первый сертификат в файле соответствует закрытому ключу, поэтому другой порядок сертификатов может стать проблемой. Эту проблему решает опция -clcerts, выводящая только сертификат, соответствующий закрытому ключу. Если требуются сертификаты УЦ, их можно вывести в отдельный файл, используя опции -nokeys и -cacerts, чтобы вывести только сертификаты УЦ.

Алгоритмы -keypbe и -certpbe позволяют указать конкретные алгоритмы для зашифрования закрытых ключей и сертификатов. В «МагПро КриптоПакет» в этом качестве используется алгоритм gost89.

7.6.13.5 Особенности работы с ключами российских криптографических алгоритмов

При экспорте ключей российских криптографических алгоритмов для шифрования и контроля целостности данных контейнера PKCS#12 следует использовать только алгоритмы ГОСТ. Поэтому в обязательном порядке должны быть указаны опции -keypbe и -macalg, про этом в опции -keypbe должно быть указано значение gost89, а в опции -macalg - значение md_gost12_256 или md_gost12_512. Если дополнительно указывается опция -certpbe, предписывающая шифровать не только закрытые ключи, но и сертификаты, в этой опции также следует указывать значение gost89.

При импорте ключей российских криптографических алгоритмов следует указывать либо опцию -gost89 (шифровать файл закрытого ключа алгоритмом ГОСТ 28147-89), либо опцию -nodes (не шифровать файл закрытого ключа). Использование других алгоритмов шифрования файла закрытого ключа не допускается.

7.6.13.6 Совместимость с предыдущими версиями «МагПро КриптоПакет»

СКЗИ «МагПро КриптоПакет» 3.0 реализует создание и проверку кода аутентификации контейнеров PKCS#12 с ключами российских криптографических алгоритмов в соответствии с Методическими рекомендациями Технического комитета по стандартизации «Криптографическая защита информации» (ТК 26). Предыдущие версии СКЗИ «МагПро КриптоПакет» были

разработаны до принятия указанных рекомендаций, поэтому вырабатывают код аутентификации по другому алгоритму. Поэтому контейнер, созданный версией 3.0, не будет читаться предыдущими версиями «МагПро КриптоПакет» и наоборот.

Для того, чтобы «МагПро КриптоПакет 3.0» смог обработать контейнер PKCS#12, сформированный предыдущими версиями «МагПро КриптоПакет», нужно установить переменную окружения `LEGACY_GOST_PKCS12` (значение переменной может быть любым).

Для того, чтобы средствами «МагПро КриптоПакет 3.0» создать контейнер PKCS#12, который могли бы прочитать предыдущие версии «МагПро КриптоПакет», нужно выставить три переменные окружения:

```
GOST_PBE_HMAC=md_gost94
CRYPT_PARAMS=id-Gost28147-89-CryptoPro-A-ParamSet
LEGACY_GOST_PKCS12 в поизвольное значение
```

и при выполнении команды `pkcs12 -export` параметру `-macalg` следует указать значение `md_gost94`, а параметру `-keypbe` – значение `gost89`

7.6.13.7 Совместимость с СКЗИ других производителей

При импорте контейнеров PKCS#12, сформированных криптосредствами других производителей, никаких дополнительных действий как правило не требуется. Если контейнер был создан старой версией СКЗИ, может потребоваться выставить переменную окружения `LEGACY_GOST_PKCS12` (см. 7.6.13.6).

Контейнеры PKCS#12, созданные «МагПро КриптоПакет 3.0» с настройками по умолчанию, нормально читаются криптосредствами компании «ИнфоТеКС», а вот продукты компании «Крипто-Про» прочитать их не смогут. Для формирования контейнера, совместимого с «КриптоПро CSP», нужно установить переменную окружения `CRYPTOPRO_PK_WRAP` (значение переменной может быть любым).

7.6.13.8 Примеры

Интерпретировать файл формата PKCS#12 и вывести его в файл:

```
openssl pkcs12 -in file.p12 -out file.pem -gost89
```

Вывести только клиентские сертификаты в файл:

```
openssl pkcs12 -in file.p12 -clcerts -out file.pem
```

Не зашифровывать закрытый ключ:

```
openssl pkcs12 -in file.p12 -out file.pem -nodes
```

Вывести информацию о файле формата PKCS#12:

```
openssl pkcs12 -in file.p12 -info -noout
```

Создать файл формата PKCS#12:

```
openssl pkcs12 -export -in file.pem -out file.p12 -name <<My
Certificate>> -keypbe gost89 -macalg md_gost12_256
```

Включить дополнительные сертификаты:

```
openssl pkcs12 -export -in file.pem -out file.p12 -name <<My
Certificate>> -keypbe gost89 -macalg md_gost12_256 -certfile
othercerts.pem
```

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