127 018, Москва, Улица Образцова, 38
Телефон: (095) 933 1168
Факс: (095) 289 4367
http://www.CryptoPro.ru
E-mail: info@CryptoPro.ru



КРИПТО-ПРО

Приложение командной строки
(версия 3.0x)

Москва 2004

© ООО "Крипто-Про", 2000-2004. Все права защищены.

Авторские права на средство криптографической защиты информации КриптоПро CSP и эксплуатационную документацию зарегистрированы в Российском агентстве по патентам и товарным знакам (Роспатент). Свидетельство об официальной регистрации программ для ЭВМ № 2001610275 от 14 марта 2001 года.

Содержание

Содержание....2
Аннотация....3
Общие сведения....3
Системные требования....3
Использование программы....3
Запуск приложения....3
Критерий поиска сертификатов....3
Команды шифрования/расшифрования....4
Работа с пакетами файлов....5
Работы с подписями....6
Работа с сертификатами....7
Работа с запросами на сертификат....8
Команда для работы с серийным номером лицензии....10
Криптопровайдеры «КриптоПро»....11
Возвращаемые коды ошибок....11

Аннотация

Данный документ содержит общую информацию о программном продукте компании Крипто-Про “Приложение командной строки”: требования к системе, руководство по использованию приложения.

Общие сведения

Данный программный продукт представляет собой приложение командной строки для работы с сертификатами, шифрования/расшифрования данных, создания/проверки цифровых подписей и хеширования данных с использованием инфраструктуры открытых ключей.

Системные требования

Приложение работает в операционных системах:

- Microsoft Windows 98 (с установленным Microsoft Internet Explorer 5.0 и выше);
- Microsoft Windows NT 4.0 Service Pack 6a (с установленным Microsoft IE 5.0 и выше);
- Microsoft Windows Millennium Edition;
- Microsoft Windows 2000;
- Microsoft Windows XP;
- Solaris 9;
- FreeBSD 5;
- Red Hat Linux 7, 9.

Для использования команд **-creatcert** и **-pendcert** (см. раздел ”Работа с запросами на сертификат”) с адресом ЦС вида “\\сервер\имяЦС” необходимо запускать приложение в операционной системе Microsoft Windows 2000 Server.

Кроме этого, для работы приложения необходим криптопровайдер ”КриптоПро CSP”.

Использование программы

Запуск приложения

Программа реализована в виде исполняемого файла “cryptcp.exe”. Для ее запуска необходимо выполнить следующую команду:

[путь]cryptcp [<команда> [<опции и файлы>]]

путь – путь к месторасположению программы (например, “c:\utils\”);
cryptcp – имя исполняемого файла приложения;
команда – одна из допустимых команд (см. ниже);
опции – параметры команды (свои для каждой команды), начинающиеся с “-“;
файлы – имена одного или двух файлов, в зависимости от команды. Порядок файлов в командной строке относительно друг друга должен быть такой, как указано в описании команды.

Примечание: К понятию *файл* также относятся маски файлов.

Если не указать *команду*, то на экран выводится список всех доступных команд с их кратким описанием. Для получения более детального описания определенной команды, необходимо указать опцию **-help**.

При описании опций звездочкой (*) помечена опция по умолчанию (для нескольких взаимоисключающих опций).

Критерий поиска сертификатов

Критерий поиска сертификатов (далее – *КПС*) используется для задания сведений о субъектах, чьи сертификаты будут использоваться при выполнении команды (например, шифрование или подпись данных). Если в команда такова, что КПС должен удовлетворять только один сертификат, то такой КПС будет обозначаться *КПС1*. КПС задается в форме опций командной строки, которые имеют следующий синтаксис:

[-dn <RDN>]n раз [-{m|u}[<имя>]|-f <файл>]k раз
[-all|-1|-q[N]] [{-nochain|-errchain [-norev]}]

-dn – указание строк для поиска в RDN (иначе поиск не зависит от RDN);
RDN – список строк (через запятую), используемых для поиска сертификатов. Будут найдены сертификаты, в RDN субъекта которых присутствуют все эти строки.
-m – поиск осуществляется в хранилищах компьютера (LOCAL_MACHINE);
-u* – поиск осуществляется в хранилищах пользователя (CURRENT_USER);
имя – название хранилища (по умолчанию “My” для создания подписи или расшифровки и “My+Addressbook” для остальных случаев);
-f – в качестве хранилища используется сообщение или файл сертификата;
файл – имя файла;
-all* – использовать все найденные сертификаты (* для КПС);
-1* – будет найден только один сертификат, иначе – ошибка (* для КПС1);
-q[N] – если найдено менее N сертификатов, то вывести запрос для выбора нужного (по умолчанию N=10);
-nochain – не проверять цепочки найденных сертификатов;
-norev – не проверять сертификаты в цепочке на предмет отозванности;
-errchain – завершать выполнение с ошибкой, если хотя бы один сертификат не прошел проверку.

Примеры использования КПС можно найти в описаниях команд, использующих его.

Примечание: Если внутри опции **имя** или **RDN** присутствуют пробелы, то ее необходимо заключить в кавычки. То же относится к именам файлов и папок.

Пример:
Иван Иванов,a@b.c – неверно;
“Иван Иванов,a@b.c” – верно;
CN=Иванов,E=a@b.c – верно.

Команды шифрования/расшифрования

-encr <КПС> [-der] <входной файл> <сообщение>

Зашифровать данные и создать сообщение.
КПС – КПС получателей;
-der – использовать формат DER вместо BASE64;
входной файл – файл, содержащий входные данные;
сообщение – файл, который будет содержать созданное сообщение.

Примечание: Для того чтобы зашифровать данные “на себя”, необходимо указать КПС своего сертификата.

Примеры:

```
cryptcp -encr -dn "Иванов Петр,ivanov@bank.ru" -uMy -der test.txt
 test1.msg
```

Зашифровать содержимое файла "test.txt" в "test1.msg" (бинарный формат), используя ВСЕ сертификаты хранилища "Личные" ("My") текущего пользователя (а не локального компьютера), содержащие в поле "Субъект" ("Subject") подстроки "Иванов Петр" *и* "ivanov@bank.ru".

```
cryptcp -encr -f "a:\Petr’s cert.p7b" test.txt test1.msg
```

Зашифровать содержимое файла "test.txt" в "test1.msg" (формат BASE64), используя сертификат из файла "a:\Petr’s cert.p7b".

-decr <КПС1> [-start] <сообщение> <выходной файл>

Расшифровать данные из сообщения.
КПС1 – КПС получателя;
-start – открыть (запустить) полученный файл;

сообщение – файл, содержащий сообщение;
выходной файл – файл, в который будут записаны данные из сообщения.

Пример:

```
cryptcp -decr -dn "Иванов Петр,ivanov@bank.ru" -start test.msg
 test2.txt
```

Расшифровать сообщение из файла "test.msg в файл "test2.txt", используя закрытый ключ, связанный с сертификатом хранилища "Личные" ("My") текущего пользователя, содержащим в поле "Субъект" ("Subject") подстроки "Иванов Петр" и "ivanov@bank.ru", а затем открыть полученный файл.

Работа с пакетами файлов

[-dir <папка>] -hash [-provtype <N>] [-provname <CSP>] <маска файлов>

Получить “хеши” файлов и записать их в файлы “имя_исходного_файла.hsh”.

-dir – указать папку для файлов с хешами, иначе – текущая;
-provtype – указать тип криптопровайдера (**N**) (по умолчанию 71);
-provname – указать имя криптопровайдера (**CSP**);
маска файлов – стандартная маска хешируемых файлов.

Примечание: Если опция **-provname** не указана, то будет использован провайдер по умолчанию указанного типа (**-provtype**). Если указанная папка не существует, то она будет создана. В версии программы 2.52 для вычисления хеша в случае использования "КриптоПро CSP" используется ГОСТ Р34.11-94, в предыдущих версиях - ГОСТ 28147-89.

Пример:

```
cryptcp -hash -dir hashes -provtype 75 *.exe
```

Создать для всех файлов с расширением "exe" текущей папки хеши и записать их в папку "hashes". При хешировании использовать криптопровайдер по умолчанию для типа 75.

[-dir <папка>] -vhash [-provtype <N>] [-provname <CSP>] <маска файлов>

Проверить “хеши” файлов, созданные с помощью предыдущей команды.

-dir – указать папку с файлами, содержащими хеши, иначе – текущая;
-provtype – указать тип криптопровайдера (**N**) (по умолчанию 71);
-provname – указать имя криптопровайдера (**<CSP>**);
маска файлов – стандартная маска проверяемых файлов.

Примечание: Если опция **-provname** не указана, то будет использован провайдер по умолчанию указанного типа (**-provtype**). В версии программы 2.52 для вычисления хеша в случае использования "КриптоПро CSP" используется ГОСТ Р34.11-94, в предыдущих версиях - ГОСТ 28147-89.

Пример:

```
cryptcp -vhash -dir c:\hashes -provtype 75 *.exe
```

Проверить для всех файлов с расширением "exe" текущей папки хеши, используя хеши, хранящиеся в папке "c:\hashes". При хешировании использовать криптопровайдер по умолчанию для типа 75.

[-dir <папка>] -signf <КПС1> <маска файлов> [-cert] [-crl] [-der]

Создать подписи файлов и записать их в файлы “имя_исходного_файла.sgn”.

-dir – указать папку для файлов с подписями, иначе – текущая;
КПС1 – КПС автора подписи;
-cert – добавлять в подписи сертификат отправителя;
-crl – добавлять в подписи список отозванных сертификатов;

-der – использовать формат DER вместо BASE64;
маска файлов – стандартная маска подписываемых файлов.

Примечание: Если указанная папка не существует, то она будет создана.

Пример:

```
cryptcp -signf –dir \signs –uMyCerts –dn "Иванов
 Петр,ivanov@bank.ru" d:\*.doc
```

Подписать все файлы с расширением "doc" из корневой папки диска "d:", используя закрытый ключ, связанный с сертификатом хранилища "MyCerts" текущего пользователя, содержащим в поле "Субъект" ("Subject") подстроки "Иванов Петр" и "ivanov@bank.ru", полученные подписи сохранить в папке “signs” в корне текущего диска.

[-dir <папка>] -vsignf <КПС> <маска файлов>

Проверить подписи файлов, созданные с помощью предыдущей команды.
-dir – указать папку с файлами, содержащими подписи, иначе – текущая;
КПС – КПС автора подписи;
маска файлов – стандартная маска проверяемых файлов.

Пример:

```
cryptcp -vsignf –dir \signs –uMyCerts d:\*.doc
```

Проверить все файлы с расширением "doc" из корневой папки диска "d:", используя созданные ранее подписи из папки “signs” в корне текущего диска. Поиск сертификата для проверки подписей искать в хранилище "MyCerts" текущего пользователя.

Работа с подписями

-sign <КПС1> [-nocert] [-crl] [-der] <входной файл> <сообщение>

Подписать данные и создать сообщение.
КПС1 – КПС автора подписи;
-nocert – не добавлять в сообщение сертификат отправителя;
-crl – добавление списка отозванных сертификатов;
-der – использовать формат DER вместо BASE64;
входной файл – файл, содержащий входные данные;
сообщение – файл, который будет содержать созданное сообщение.

Пример:

```
cryptcp -sign -mMy -dn Седов -q5 -nocert -crl -der test.txt
 test2.msg
```

Подписать содержимое файла "test.txt" и создать подписанное сообщение "test2.msg" (в бинарном виде), не включающее в себя используемый сертификат, но включающее список отозванных сертификатов центра сертификации, выдавшего используемый сертификат. Поиск используемого сертификата происходит следующим образом:

1. Находятся все сертификаты хранилища "Личные" текущего пользователя и локального компьютера.

2. Если их нашлось более пяти, то - ошибка, иначе пользователю будет предложено выбрать один из найденных сертификатов.

-addsign <КПС1> [-nocert] [-crl] <сообщение>

Добавить цифровую подпись в сообщение.
КПС1 – КПС автора подписи;
-nocert – не добавлять в сообщение сертификат отправителя;
-crl – добавление списка отозванных сертификатов;
сообщение – файл, содержащий сообщение.

Пример:

```
cryptcp -addsign –m –dn "Иванов Петр,ivanov@bank.ru" test.msg
```

Добавить в подписанное сообщение "test.msg" подпись, используя закрытый ключ, связанный с сертификатом хранилища "Личные" ("My") локального компьютера, содержащим в поле "Субъект" ("Subject") подстроки "Иванов Петр" и "ivanov@bank.ru". В добавленную подпись будет включен сертификат открытого ключа автора подписи.

-delsign <КПС1> <сообщение>

Удалить цифровую подпись из сообщения.

КПС1 – КПС автора подписи;
сообщение – файл, содержащий сообщение.

-verify [<КПС> [-verall]] [-start] <сообщение> [<выходной файл>]

Проверка цифровых подписей.

КПС – КПС авторов подписей;
-verall – проверять все подписи (иначе – только подписи авторов из КПС);
-start – открыть (запустить) полученный файл;
сообщение – файл, содержащий сообщение;
выходной файл – файл, в который будут записаны данные из сообщения.

Примечание: Если в сообщении содержится сертификат кого-то из авторов подписей, то используется именно этот сертификат.

Примеры:

```
cryptcp -verify -dn ivanov@bank.ru -verall test2.msg test2.txt
```

Проверить подпись сообщения "test2.msg", используя один из найденных сертификатов в хранилищах "Личные" ("My") и "Другие пользователи" ("AddressBook") текущего пользователя, содержащих в поле "Субъект" ("Subject") подстроку “ivanov@bank.ru” и записать содержимое подписанного сообщения в файл "test2.txt".

```
cryptcp -verify test2.msg
```

Проверить все подписи сообщения "test2.msg", используя сертификаты, содержащиеся в сообщении. Если для какой-либо подписи в сообщении сертификат не удалось найти, то подпись проверена не будет.

Работа с сертификатами

-copycert <КПС> [-{dm|du}[<имя>]|-df <файл> [-der]]

Копировать сертификаты в заданное хранилище.

КПС – КПС, которые надо скопировать;
-dm – копирование в хранилище компьютера (LOCAL_MACHINE);
-du* – копирование в хранилище пользователя (CURRENT_USER);
имя – название конечного хранилища (по умолчанию “My”);
-df – в качестве хранилища используется файл сертификата;
файл – имя файла;
-der – использовать формат DER вместо BASE64 (только с ключом **-df**).

Примечание: Если указан ключ **-df**, то, в случае, если найден только один сертификат, создается файл типа “.cer”, иначе – “.p7b”.

Пример:

```
cryptcp -copycert –u –df a:\MyCerts.p7b
```

Копирует все сертификаты хранилища "Личные" ("My") текущего пользователя в файл "a:\MyCerts.p7b" (в кодировке BASE64).

-CSPcert [-provtype <N>] [-provname <CSP>] [-cont <контейнер>]
[-ku|-km] [-{dm|du}[<имя>]|-df <файл> [-der]]

Скопировать сертификат из ключевого контейнера в заданное хранилище.

-provtype – указать тип криптопровайдера (**N**) (по умолчанию 71);

-provname – указать имя криптопровайдера (**CSP**);
-cont – задать имя ключевого **контейнера** (по умолчанию выбор из списка);
-ku* – использовать контейнер пользователя (CURRENT_USER);
-km – использовать контейнер компьютера (LOCAL_MACHINE);
-dm – копирование в хранилище компьютера (LOCAL_MACHINE);
-du* – копирование в хранилище пользователя (CURRENT_USER);
имя – название конечного хранилища (по умолчанию “My”);
-df – в качестве хранилища используется сообщение или файл сертификата;
файл – имя файла;
-der – использовать формат DER вместо BASE64.

Примечание: Если опция **-provname** не указана, то будет использован провайдер по умолчанию указанного типа (**-provtype**). Для операционных систем семейства UNIX в качестве параметра опции **-cont** необходимо указывать имя контейнера вместе со считывателем в формате “\\.\имя_считывателя\имя_контейнера” (например “\\.\HDIMAGE\cont_name”).

Пример:

```
cryptcp -CSPcert -km -cont WebServer -df a:\WebServer.cer -der
```

Копирует сертификат из ключевого контейнера "WebServer" криптопровайдера по умолчанию для типа 71 локального компьютера в файл "a:\WebServer.cer" (в кодировке DER).

-delcert <КПС> [-yes]

Удаление сертификатов из хранилища.
КПС – КПС удаляемых сертификатов;
-yes – автоматически отвечать на все вопросы “Да”.

Пример:

```
cryptcp -delcert -m -dn OldServer
```

Удаляет все сертификаты хранилища "Личные" ("My") локального компьютера, содержащие в поле "Subject" подстроку "OldServer".

Работа с запросами на сертификат

-creatrqst -dn <RDN> [-provtype <N>] [-provname <CSP>] [-SMIME] [-nokeygen|-exprt] [-ex|-sg|-both] [-ku|-km] [-cont <имя>] [-silent] [-askpin] [-certusage <OIDs>] [-der] <имя файла>

Создание запроса сертификата и сохранение его в файле PKCS #10.
RDN – список имен полей RDN (например: CN, O, E, L) и их значений вида: **<ИмяПоля1>=<ЗначечиеПоля1>[,<ИмяПоля2>=<ЗначечиеПоля2>...]**
-provtype – указать тип криптопровайдера (**N**) (по умолчанию 71);
-provname – указать имя криптопровайдера (**CSP**);
-nokeygen – использовать существующие ключи из указанного контейнера;
-SMIME – включить возможности SMIME (по умолчанию – нет; только Windows);
-exprt – пометить ключи как экспортируемые;
-ex – создать/использовать ключи для обмена зашифрованными данными;
-sg – создать/использовать ключи только для работы с подписями;
-both* – создать/использовать оба типа ключей;
-ku* – использовать контейнер пользователя (CURRENT_USER);
-km – использовать контейнер компьютера (LOCAL_MACHINE);
-cont – задать имя ключевого **контейнера** (по умолчанию выбор из списка);
-silent – генерация ключа без пользовательского интерфейса криптопровайдера;
-askpin – запрашивать ПИН при создании ключевого контейнера (только UNIX);
-certusage – задать назначения сертификата (**OIDs**). Если назначений несколько, то их необходимо указать через запятую (например, ”1.3.6.1.5.5.7.3.4, 1.3.6.1.5.5.7.3.2”);

-der – использовать формат DER вместо BASE64;
имя файла – имя файла, в котором следует сохранить запрос.

Примечание: Если опция **-provname** не указана, то будет использован провайдер по умолчанию указанного типа (**-provtype**). Далее, если не указаны опции **-nokeygen** и **-cont**, то имя контейнера сгенерирует криптопровайдер. Для операционных систем семейства UNIX в качестве параметра опции **-cont** необходимо указывать имя контейнера вместе со считывателем в формате "\\.\имя_считывателя\имя_контейнера" (например "\\.\HDIMAGE\cont_name").

Пример:

```
cryptcp -creatrqst c:\request.der -provtype 75 -cont Ivanov
 -dn "E=ivanov@bank.ru,CN=Иванов Петр" -both -ku
 -provname "Crypto-Pro GOST R 34.10-2001 Cryptographic Service
 Provider"
```

Создать запрос на субъект "E=ivanov@bank.ru,CN=Иванов Петр", используя открытый ключ, сгенерированный в контейнере "Ivanov" текущего пользователя криптопровайдером "Crypto-Pro GOST R 34.10-2001 Cryptographic Service Provider" (тип - 75) и сохранить его в файл c:\request.der в кодировке Base64. Назначения ключа - подпись и шифрование.

-instcert [-provtype <N>] [-provname <CSP>] [-cont <имя>] [-ku|-km] [-{dm|du}[<имя>]] [-noCSP] <имя файла>

Установка сертификата из файла PKCS #7, содержащего изданный сертификат.

-provtype – указать тип криптопровайдера (**N**) (по умолчанию 71);
-provname – указать имя криптопровайдера (**CSP**);
-cont – задать имя ключевого **контейнера** (по умолчанию выбор из списка);
-ku* – использовать контейнер пользователя (CURRENT_USER);
-km – использовать контейнер компьютера (LOCAL_MACHINE);
-dm – установка в хранилище компьютера (LOCAL_MACHINE);
-du* – установка в хранилище пользователя (CURRENT_USER);
имя – название конечного хранилища для установки (по умолчанию "My");
-noCSP – не сохранять сертификат в контейнере криптопровайдера;
имя файла – имя файла, содержащего сертификат.

Примечание: Если указана опция **noCSP**, то опции **provname**, **provtype**, **cont**, **km, ku** игнорируются. Если опция **-provname** не указана, то будет использован провайдер по умолчанию указанного типа (**-provtype**). Для операционных систем семейства UNIX в качестве параметра опции **-cont** необходимо указывать имя контейнера вместе со считывателем в формате "\\.\имя_считывателя\имя_контейнера" (например "\\.\HDIMAGE\cont_name").

-creatcert –dn <RDN> [-provtype <N>] [-provname <CSP>] [-SMIME] [-nokeygen|-exprt] [-{ex|sg|both}] [-cont <имя>] [-ku|-km] [-certusage <OIDs>] [-CA <адрес ЦС>] [-{dm|du}[<имя>]] [-noCSP] [-silent] [-askpin] [-FileID <Имя файла>]

Создать запрос на сертификат, отправить его в центр сертификации, получить выписанный сертификат и установить его.

RDN – список имен полей RDN (например: CN, O, E, L) и их значений вида:
<ИмяПоля1>=<ЗначениеПоля1>[,<ИмяПоля2>=<ЗначениеПоля2>...]
-provtype – указать тип криптопровайдера (**N**) (по умолчанию 71);
-provname – указать имя криптопровайдера (**CSP**);
-SMIME – включить возможности SMIME (по умолчанию – нет; только Windows);
-nokeygen – использовать существующие ключи из указанного контейнера;
-exprt – пометить ключи как экспортируемые;
-ex – создать/использовать ключи для обмена зашифрованными данными;
-sg – создать/использовать ключи только для работы с подписями;

-both* – создать/использовать оба типа ключей;
-ku* – использовать контейнер пользователя (CURRENT_USER);
-km – использовать контейнер компьютера (LOCAL_MACHINE);
-cont – задать имя ключевого **контейнера** (по умолчанию выбор из списка);
-certusage – задать назначения сертификата (**OIDs**). Если назначений несколько, то их нужно указать через запятую (например, ”1.3.6.1.5.5.7.3.4,1.3.6.1.5.5.7.3.2”);
-CA – указать адрес центра сертификации, иначе это адрес “CP CSP Test CA”;
адрес ЦС – вида “http://xxx.yyy/zzz” или “\\сервер\имяЦС” (см. ”Системные требования”);
-dm – установка в хранилище компьютера (LOCAL_MACHINE);
-du* – установка в хранилище пользователя (CURRENT_USER);
имя – название конечного хранилища для установки (по умолчанию “My”);
-noCSP – не сохранять сертификат в контейнере криптопровайдера;
-silent – генерация ключа без пользовательского интерфейса криптопровайдера;
-askpin – запрашивать ПИН при создании ключевого контейнера (только UNIX);
-FileID – имя файла, используемого для записи идентификатора запроса в случае “отложенной выдачи” сертификата (см. **-pendcert**). Если не файл не указан, то идентификатор будет выведен на экран.

Примечание: Если опция **-provname** не указана, то будет использован провайдер по умолчанию указанного типа (**-provtype**). Далее, если не указаны опции **-nokeygen** и **-cont**, то имя контейнера сгенерирует криптопровайдер. Для операционных систем семейства UNIX в качестве параметра опции **-cont** необходимо указывать имя контейнера вместе со считывателем в формате “\\.\имя_считывателя\имя_контейнера” (например “\\.\HDIMAGE\cont_name”).

`-pendcert [-provtype <N>] [-provname <CSP>] [-cont <имя>] [-ku|-km] [-CA <адрес ЦС>] [-{dm|du}[<имя>]] [-noCSP] [-FileID <Имя файла>]`

Проверить, не выпущен ли сертификат, запрос на который был отправлен ранее, получить выписанный сертификат и установить его.

-provtype – указать тип криптопровайдера (**N**) (по умолчанию 71);
-provname – указать имя криптопровайдера (**CSP**);
-cont – задать имя ключевого **контейнера** (по умолчанию выбор из списка);
-ku* – использовать контейнер пользователя (CURRENT_USER);
-km – использовать контейнер компьютера (LOCAL_MACHINE);
-CA – указать адрес центра сертификации, иначе это адрес “CP CSP Test CA”;
адрес ЦС – вида “http://xxx.yyy/zzz” или “\\сервер\имяЦС” (см. ”Системные требования”);
-dm – установка в хранилище компьютера (LOCAL_MACHINE);
-du* – установка в хранилище пользователя (CURRENT_USER);
имя – название конечного хранилища для установки (по умолчанию “My”);
-noCSP – не сохранять сертификат в контейнере криптопровайдера;
-FileID – имя файла, содержащего идентификатор запроса. Если не файл не указан, то идентификатор нужно будет ввести вручную.

Примечание: Если указана опция **noCSP**, то опции **provname**, **provtype**, **cont**, **km, ku** игнорируются. Если опция **-provname** не указана, то будет использован провайдер по умолчанию указанного типа (**-provtype**). Для операционных систем семейства UNIX в качестве параметра опции **-cont** необходимо указывать имя контейнера вместе со считывателем в формате “\\.\имя_считывателя\имя_контейнера” (например “\\.\HDIMAGE\cont_name”).

Команда для работы с серийным номером лицензии (только для Windows)

`-sn [<серийный номер>]`

Сохранить/показать серийный номер лицензии.

серийный номер – серийный номер, который необходимо сохранить (можно указывать как с разделителями, так и без них).

Примечание: Для того чтобы посмотреть сохраненный серийный номер, достаточно указать команду **–sn** без параметра. В операционных системах семейства UNIX используется серийный номер лицензии криптипровайдера.

Пример:

```
cryptcp -sn P020G-Q0010-A5000-01UXA-XUFFD
```

Сохраняет указанный серийный номер лицензии на компьютере.

Криптопровайдеры «КриптоПро»

CryptoPro CSP 1.1	
Имя	Crypto-Pro Cryptographic Service Provider
Тип	2

CryptoPro CSP 2.0		
Имя	Crypto-Pro GOST R 34.10-94 Cryptographic Service Provider	Crypto-Pro GOST R 34.10-2001 Cryptographic Service Provider
Тип	71	75

CryptoPro CSP 3.0 (kc1)		
Имя	Crypto-Pro GOST R 34.10-94 KC1 CSP	Crypto-Pro GOST R 34.10-2001 KC1 CSP
Тип	71	75

CryptoPro CSP 3.0 (kc2)		
Имя	Crypto-Pro GOST R 34.10-94 KC2 CSP	Crypto-Pro GOST R 34.10-2001 KC2 CSP
Тип	71	75

Возвращаемые коды ошибок

Код ошибки (DEC)	Код ошибки (HEX)	Описание ошибки
536871012	20000064	Мало памяти
536871013	20000065	Не удалось открыть файл
536871014	20000066	Операция отменена пользователем
536871015	20000067	Некорректное преобразование BASE64
536871016	20000068	Если указан параметр '-help', то других быть не должно
536871112	200000C8	Указан лишний файл
536871113	200000C9	Указан неизвестный ключ
536871114	200000CA	Указана лишняя команда
536871115	200000CB	Для ключа не указан параметр
536871116	200000CC	Не указана команда
536871117	200000CD	Не указан необходимый ключ:
536871118	200000CE	Указан неверный ключ:
536871119	200000CF	Параметром ключа '-q' должно быть натуральное число
536871120	200000D0	Не указан входной файл
536871121	200000D1	Не указан выходной файл
536871122	200000D2	Команда не использует параметр с именем файла
536871123	200000D3	Не указан файл сообщения

536871212	2000012C	Не удалось открыть хранилище сертификатов:
536871213	2000012D	Сертификаты не найдены
536871214	2000012E	Найдено более одного сертификата (ключ '-1')
536871215	2000012F	Команда подразумевает использование только одного сертификата
536871216	20000130	Неверно указан номер
536871217	20000131	Нет используемых сертификатов
536871218	20000132	Данный сертификат не может применяться для этой операции
536871219	20000133	Цепочка сертификатов не проверена
536871220	20000134	Криптопровайдер, поддерживающий необходимый алгоритм не найден
536871312	20000190	Не указана маска файлов
536871313	20000191	Указаны несколько масок файлов
536871314	20000192	Файлы не найдены
536871315	20000193	Задана неверная маска
536871316	20000194	Неверный хеш
536871412	200001F4	Ключ '-start' указан, а выходной файл нет
536871413	200001F5	Содержимое файла - не подписанное сообщение
536871414	200001F6	Неизвестный алгоритм подписи
536871415	200001F7	Сертификат автора подписи не найден
536871416	200001F8	Подпись не найдена
536871417	200001F9	Подпись не верна
536871512	20000258	Содержимое файла - не зашифрованное сообщение
536871513	20000259	Неизвестный алгоритм шифрования
536871514	2000025A	Не найден сертификат с соответствующим секретным ключом
536871612	200002BC	Не удалось инициализировать COM
536871613	200002BD	Контейнеры не найдены
536871614	200002BE	Не удалось получить ответ от сервера
536871615	200002BF	Сертификат не найден в ответе сервера
536871616	200002C0	Файл не содержит идентификатор запроса:
536871617	200002C1	Некорректный адрес ЦС
536871618	200002C2	Получен неверный Cookie
536871712	20000320	Серийный номер содержит недопустимое количество символов
536871713	20000321	Неверный код продукта
536871714	20000322	Не удалось проверить серийный номер
536871715	20000323	Не удалось сохранить серийный номер
536871716	20000324	Не удалось загрузить серийный номер
536871717	20000325	Лицензия просрочена

Примечание: Кроме кодов, приведенных в таблице, приложение может возвращать код любой системной ошибки Windows.