Лабораторная работа №1
Дисциплина «Методы и средства защиты информации»
GPG

Евгений Хандыго, гр. 53501/3

28 февраля 2016 г.

Содержание

1 Постановка задачи

В рамках данной работы необходимо овладеть основными аспектами работы с утилитой gpg4win. Ход работы соответствует следующим пунктам:

- Изучить документацию gpg и gpg4win.
- Поставить собственную ЭЦП на файл.
- Проверить ЭЦП на стороннем файле.
- Обменяться зашифрованными сообщениями с другим пользователем gpg.
- Потренироваться в использовании утилиты gpg через интерфейс командной строки.

2 Поставить собственную ЭЦП на файл

Для того, чтобы начать работать с gpg, сначала необходимо сгенерировать собственную пару ключей, состоящую из публичной (сертификат) и приватной частей. Публичный ключ может распространяться свободно (в том числе и в сети Интернет), в то время как передача приватного ключа третьим лицам категорически запрещена. Для того, чтобы сгенерировать пару ключей с помощью gpg необходимо воспользоваться опцией `--gen-key`. Пример вывода утилиты gpg при использовании данной опции представлен ниже в листинге 2.1.

```
D:\apps\GPGPortable> gpg2.exe --gen-key
gpg (GnuPG) 2.0.29; Copyright (C) 2015 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Выберите тип ключа:
   (1) RSA и RSA (по умолчанию)
   (2) DSA и Elgamal
   (3) DSA (только для подписи)
   (4) RSA (только для подписи)
Ваш выбор? 1
длина ключей RSA может быть от 1024 до 4096 бит.
Какой размер ключа Вам необходим? (2048) 2048
Запрошенный размер ключа - 2048 бит
Выберите срок действия ключа.
         0 = без ограничения срока действия
      <n>  = срок действия ключа - n дней
      <n>w = срок действия ключа - n недель
      <n>m = срок действия ключа - n месяцев
      <n>y = срок действия ключа - n лет
Срок действия ключа? (0) 0
Срок действия ключа не ограничен
Все верно? (y/N) y
```

```
GnuPG необходимо составить ID пользователя в качестве идентификатора
 ↪  ключа.

Ваше настоящее имя: Test Name
Адрес электронной почты: test@example.com
Комментарий: demo certificate
Вы выбрали следующий ID пользователя:
    "Test Name (demo certificate) <test@example.com>"

Сменить (N)Имя, (C)Комментарий, (E)Адрес или (O)Принять/(Q)Выход? o
Для защиты закрытого ключа необходима фраза-пароль.

Необходимо получить много случайных чисел. Желательно, чтобы Вы
в процессе генерации выполняли какие-то другие действия (печать
на клавиатуре, движения мыши, обращения к дискам); это даст генератору
случайных чисел больше возможностей получить достаточное количество
 ↪  энтропии.
gpg: ключ E068E86E помечен как абсолютно доверенный.
открытый и закрытый ключи созданы и подписаны.

gpg: проверка таблицы доверия
gpg: требуется 3 с ограниченным доверием, 1 с полным, модель доверия
 ↪  PGP
gpg: глубина: 0 верных: 2 подписанных: 2 доверие: 0-,0q,0n,0m,0f,2u
gpg: глубина: 1 верных: 2 подписанных: 0 доверие: 2-,0q,0n,0m,0f,0u
gpg: срок следующей проверки таблицы доверия 2020-03-16
pub 2048R/E068E86E 2016-02-28
Отпечаток ключа = 9ECC B77F 5626 27A4 5F72  E53D 633E 298D E068 E86E
uid [абсолютное] Test Name (demo certificate) <test@example.com>
sub   2048R/12174CA0 2016-02-28
```

Listing 2.1: Вывод утилиты gpg при вызове с опцией `--gen-key`

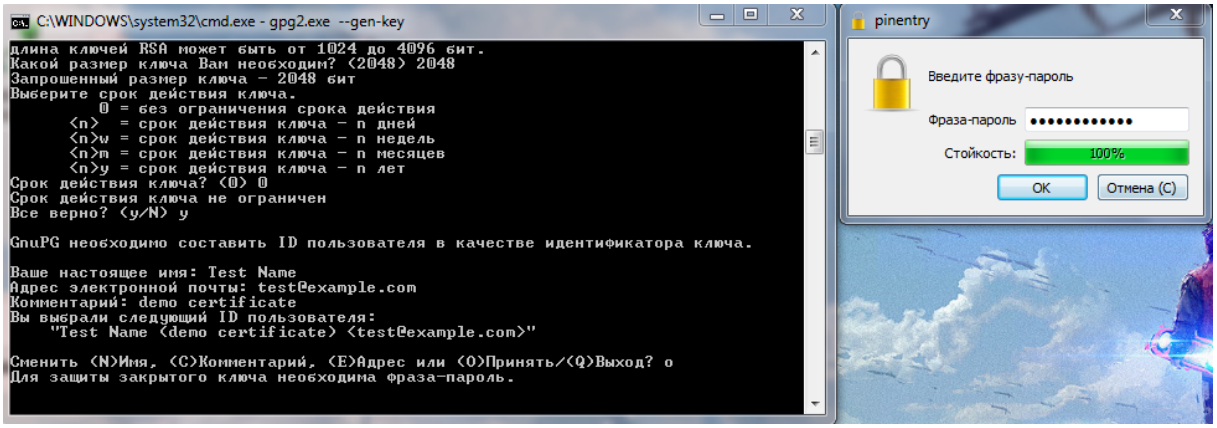

Рис. 1: Форма ввода пароля при генерации ключей в gpg

При этом в строке 34 вывода происходит предоставление пользователю формы для ввода пароля генерируемого приватного ключа. В случаях, когда приватный ключ скомпрометирован, данный пароль является последней ступенью защиты перед злоумышленниками. Форма ввода пароля показана на рисунке 1. После ее ввода требуется также подтверждение выбранного пароля.
Для дальнейшего использования созданных ключей необходимо сначала экспортировать свой сертификат для того, чтобы его можно было передать лицам, заинтересованным в отправке зашифрованных и/или подписанных сообщений пользователю. Для этого необходимо воспользоваться опцией `--export`. Дополнительно также можно использовать опции `--armor`, которая преобразовывает вывод из бинарного формата в формат ASCII, и `--output` для указания файла записи результата. Пример вызова утилиты gpg с опцией `--export` представлен в листинге 2.2.

```
D:\apps\GPGPortable>gpg2.exe --armor --export (demo certificate)
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG v2

mQENBFbSu4IBCADE6lxU9p2d/Oo76PKJ/dJiNI+RFkT7l8HZPsRYpBJPQ1si4dR/
E0L2mUctws1Dio2KkUsqdJ0bsI8R4d+d2p/LJBeN5GxLf3GmFxzfhJG/5Ko6lYiX
kxzYa+sEh1JWMf3f/aBfeWPIrae/x91ZjgNqVcZNUNFRClcYropQxxEGMb5NKaJK
aWgmhaH7UkKwBzzbylouV3LMXqUu+bhALp/kZIuf9lSPkaskBNQLM7NTg4+CnRSL
25yN3Wc+2EwpW1bOmV0fTKn/f+5F+I1LOYUrLTtsdlFG1dYSf9uTnM0KDjdCCQhZ
uTZnXYdKTITeOkFE9oA9C/CO76DJFJwDOgMfABEBAAG0L05hbWUgVGVzdCAodGVz
dCBjZXJ0aWZpY2F0ZSkgPHRlc3RAZXhhbXBsZS5jb20+iQE5BBMBCAAjBQJW0ruC
AhsDBwsJCAcDAgEGFQgCCQoLBBYCAwECHgECF4AACgkQRlBXaxqDVLaa9AgAoAJj
6kFBl+h+6+lGzZZN/PH4e+bfdJ2FoM2IMrtZEJKIbSx45RVw2GhjOSV1vNh/PEdH
...
```

Listing 2.2: Вывод утилиты gpg при вызове с опциями `--export` и `--armor`

Одним из основных сценариев использования утилиты gpg — это установка электронной цифровой подписи. Для установки ЭЦП на какой-либо файл необходимо воспользоваться опцией `--sign` или `--detach-sign` как показано в листинге 2.3.

```
D:\apps\GPGPortable>gpg2.exe --local-user E068E86E --detach-sign
 ↪  --armor --sign greeting.txt

Необходима фраза-пароль для доступа к закрытому ключу пользователя:
    "Test Name (demo certificate) <test@example.com>"
2048-битный ключ RSA, ID E068E86E, создан 2016-02-28
```

Listing 2.3: Вывод утилиты gpg при вызове с опцией `--detach-sign`

После выполнения этой команды будет создан отдельный файл цифровой подписи. Его содержание представлено в листинге 2.4

```
-----BEGIN PGP SIGNATURE-----
Version: GnuPG v2

iQEcBAABCAAGBQJW0yBNAAoJEGM+KY3gaOhuWDMH/jA0A3PAJp0ZA/g76rAxROIx
Qjenkzb8RsXP7LQZkrqDZYu0zzti94JbWV7mtwjmMSBzqAKO7VJb24MkdWdzZciT
cmGCkHOcaHyVFeycrzF7FuFhbG3I3JNundeC9eJJOrZgqmSxE8G9FKGbaqAFuiG6
knjNDAr+fWJDgqA5rTmP9bz3sj2xrbTX0u2iHUhUJKuWUgsSKmIQ/qye4AClhlNE
+b1vOlp6BfZoVhGat6MsYcFkldRMh7I7XTCuDiTLiLbXftgYGahDxlWjHHoHvaLY
vIpHaxpPVqVvGkw8VtvKrEFg0/JAOVR8pAkxVFRPPZemOsiSMXd4ZA+d3dYX33A=
=uumv
-----END PGP SIGNATURE-----
```

Listing 2.4: Пример цифровой подписи

3 Проверить ЭЦП на стороннем файле

Для демонстрации проверки подлинности ЭЦП с помощью утилиты gpg были скачаны следующий файлы:

- Инсталятор Gpg4win-Vanilla 2.3.0.
- Его цифровая подпись.
- Сертификат, с помощью которого была сделана подпись.

Теперь необходимо импортировать полученный сертификат. Для этого предназначена опция `--import`. Результат вывода утилиты gpg при использовании данной опции представлен в листинге 3.1.

```
D:\apps\GPGPortable>gpg2.exe --import another-certificate.asc
gpg: ключ EC70B1B8: импортирован открытый ключ
    "Intevation File Distribution Key <distribution-key@intevation.de>"
gpg: Всего обработано: 1
gpg:    импортировано: 1
gpg: требуется 3 с ограниченным доверием, 1 с полным, модель доверия
 ↪ PGP
gpg: глубина: 0 верных: 2 подписанных: 1 доверие: 0-,0q,0n,0m,0f,2u
gpg: глубина: 1 верных: 1 подписанных: 0 доверие: 1-,0q,0n,0m,0f,0u
```

Listing 3.1: Вывод утилиты gpg при вызове с опцией `--import`

Для дальнейшего использования импортированного сертификата необходимо подтвердить его полномочия (подписать). Согласно общепринятому мнению, данный этап подтверждения является Ахиллесовой пятой gpg, поскольку пользователь системы должен быть на сто

процентов уверен в личности создателя сертификата. Для решения данной проблемы рекомендуется проводить личные встречи с автором сертификата, однако на практике, как правило, ограничиваются сверкой отпечатков сертификата. Также стоит отметить, что здесь имеет место быть концепция web of trust, согласно которой, получаемый сертификат уже может содержать определенное количество подписей от других пользователей. Это позволяет судить о степени доверия полученному сертификату по количеству других пользователей, которые подписали его. Для того, чтобы подтвердить полномочия некоторого сертификата необходимо запустить утилиты gpg с опцией `--edit-key` и затем ввести команду `sign`. Пример выполнения данной операции представлен в листинге 3.2.

```
D:\apps\GPGPortable>gpg2.exe --local-user E068E86E --edit-key EC70B1B8
gpg (GnuPG) 2.0.29; Copyright (C) 2015 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.


pub  1024D/EC70B1B8 создан: 2010-03-19  годен до: 2020-03-16
 ↪  применимость: SC
    доверие: неизвестной   достоверность: неизвестной
[неизвестно] (1). Intevation File Distribution Key
 ↪  <distribution-key@intevation.de>

gpg> sign

pub  1024D/EC70B1B8 создан: 2010-03-19  годен до: 2020-03-16
 ↪  применимость: SC
    доверие: неизвестной   достоверность: неизвестной
 Отпечаток главного ключа: 61AC 3F5E E4BE 593C 13D6  8B1E 7CBD 620B
 ↪  EC70 B1B8

     Intevation File Distribution Key <distribution-key@intevation.de>

Срок действия данного ключа истекает 2020-03-16.
Вы уверены, что хотите подписать этот ключ
своим ключом "Test Name (demo certificate) <test@example.com>"
 ↪  (E068E86E)?

Действительно подписать? (y/N) y

Необходима фраза-пароль для доступа к закрытому ключу пользователя:
    "Test Name (demo certificate) <test@example.com>"
2048-битный ключ RSA, ID E068E86E, создан 2016-02-28
```

Listing 3.2: Вывод утилиты gpg при вызове с опцией `--edit-key`

Для проверки электронной подписи теперь достаточно вызывать утилиту gpg с опцией

`--verify` как показано в листинге 3.3.

```
D:\apps\GPGPortable>gpg2.exe --verify gpg4win-vanilla-2.3.0.exe.sig
↪ gpg4win-vanilla-2.3.0.exe
gpg: Подпись создана 11/24/15 22:06:35 Russia TZ 2 Standard Time ключом
↪ DSA с ID EC70B1B8
gpg: Действительная подпись от "Intevation File Distribution Key
↪ <distribution-key@intevation.de>" [полное]
```

Listing 3.3: Вывод утилиты gpg при вызове с опцией `--verify`

4 Обменяться зашифрованными сообщениями с другим пользователем gpg

Для демонстрации передачи сообщений зашифрованных с помощью gpg был получен, импортирован и подписан сертификат другого пользователя gpg. Для подтверждения этого рассмотрим вывод утилиты gpg, запущенной с опцией `--list-keys`, представленный в листинге 4.1. Испортированный ключ соответствует пользователю `Sergey Klimov (Lab3) <ksa1993@yandex.ru>`.

```
D:\apps\GPGPortable>gpg2.exe --list-keys
D:/apps/GPGPortable/home/pubring.gpg
------------------------------------
pub 2048R/1A8354B6 2016-02-28
uid [абсолютное] Name Test (test certificate) <test@example.com>
sub 2048R/7E07D464 2016-02-28

pub 2048R/C75B147B 2016-02-25
uid [  полное  ] Sergey Klimov (Lab3) <ksa1993@yandex.ru>
sub 2048R/409CA923 2016-02-25

pub 2048R/E068E86E 2016-02-28
uid [абсолютное] Test Name (demo certificate) <test@example.com>
sub 2048R/12174CA0 2016-02-28

pub 1024D/EC70B1B8 2010-03-19 [    годен до: 2020-03-16]
uid [  полное  ] Intevation File Distribution Key
↪ <distribution-key@intevation.de>
```

Listing 4.1: Вывод утилиты gpg при вызове с опцией `--list-keys`

Для шифрования и подписи некоторого файла необходимо вызвать утилиту gpg с опциями `--sign` и `--encrypt`, а также с помощью опций `--local-user` и `--recipient` указать

идентификатор сертификатов отправителя и получателя соответственно. Пример вывода утилиты при шифровании файла указан в листинге 4.3. При этом, после вызова приведенной команды будет создан greeting.txt.gpg, содержащий зашифрованные данные и электронную цифвровую подпись.

```
D:\apps\GPGPortable>gpg2.exe --local-user E068E86E --recipient C75B147B
 ↪  --sign --encrypt greeting.txt

Необходима фраза-пароль для доступа к закрытому ключу пользователя:
    "Test Name (demo certificate) <test@example.com>"
2048-битный ключ RSA, ID E068E86E, создан 2016-02-28
```

Listing 4.2: Вывод утилиты gpg при вызове с опциями `--encrypt` и `--sign`

Пересылка файла и его рассшифровка получателем прошли успешно.
Описанная выше операция по пересылке шифрованного сообщения была также в точности повторена со сменой ролей получателя и отправителя. От другого пользователя gpg был получен зашифрованный файл pic.jpg.gpg, который был расшифрован с помощью опции `--decrypt` утилиты gpg, как показано в листинге **??**. Расшифрованный файл представлен на рисунке 2.

```
D:\apps\GPGPortable>gpg2.exe --local-user E068E86E --output pic.jpg
 ↪  --decrypt pic.jpg.gpg

Необходима фраза-пароль для доступа к закрытому ключу пользователя:
    "Test Name (demo certificate) <test@example.com>"
2048-битный ключ RSA, ID E068E86E, создан 2016-02-28

gpg: зашифровано 2048-битным ключом RSA с ID E068E86E, созданным
 ↪  2016-02-28
      "Test Name (demo certificate) <test@example.com>"
gpg: Подпись создана 02/28/16 15:00:23 Russia TZ 2 Standard Time ключом
 ↪  RSA с ID E068E86E
gpg: Действительная подпись от "Sergey Klimov (Lab3)
 ↪  <ksa1993@yandex.ru>" [полное]
```

Listing 4.3: Вывод утилиты gpg при вызове с опциями `--decrypt`

Рис. 2: Результат расшифровки сообщения

5 Заключение

В данной работе были рассмотрены некоторые практические аспекты работы с утилитой gpg. Из проведенных экспериментов можно сделать вывод, что gpg является мощным и в то же время простым инструментом шифрования и подписи файлов. К сожалению, он не лишен недостатков и имеет свои «слабые» точки, которые, тем не менее, в большинстве своем связаны с человеческим фактором, нежели с внутренними дефектами данного продукта.