In a world where computers are used for almost all financial & personal written records, the need to defend data is more crucial than ever. In the Internet age, billions of people are accessing electronic information databases every second. Improperly protected data could open the door to many threats everything from identity theft to access to classified information that could likely harm national security. Security is not the only menace facing data managers. There is also the growing worry for defense against malicious viral attacks and ruinous data corruption that could easily put a commercial enterprise in a legal bind. Although most threats are not on the level of starting World War III, businesses and individuals can be destroyed by the lack of security concerning sensitive data. Appropriate data security methods can decrease the risk of losing important and private information and should be a top priority.

Steps to Better Data Security

Steganography is the art and science of writing hidden messages in such a way that no one, apart from the sender and intended recipient, suspects the existence of the message, a form of security through obscurity.

It is possible to hide a rar archive inside a png image file and then retrieve the files from this image.
cat picture.png archive.rar > hidden_archive_in_pic.png
This can also be done on Windows:
copy picture.png + archive.rar hidden_archive_in_pic.png
When you want to retrieve the hidden files, download the image, rename to .rar and extract

OutGuess

OutGuess is a universal steganographic tool that allows the insertion of hidden information into the redundant bits of data sources. The nature of the data source is irrelevant to the core of OutGuess. The program relies on data specific handlers that will extract redundant bits and write them back after modification. In this version the PNM and JPEG image formats are supported.http://www.outguess.org/

Steghide

Steghide is an open source steganography program for Windows and Linux that can hide data in image and audio files. The current version 0.5.1 offers compression, encryption, and an integrity check of the embedded data.

To use Steghide to embed the file secret.txt into example.bmp, you'd use the following command:

```
$ steghide embed -cf example.bmp -ef secret.txt
  Enter passphrase:
  Re-Enter passphrase:
  embedding "secret.txt" in "example.bmp.". done
```

The -cf argument specifies the cover file and -ef the embedded file.

The default encryption algorithm is AES/Rijndael with a keysize of 128 bits. However, if you want another encryption algorithm, you can run the steghide encinfo command to see all supported encryption algorithms.

If you have received a cover file that contains a file that has been embedded with Steghide, use the extract command to reveal the hidden file with an -sf argument:

```
$ steghide extract -sf example.bmp
  Enter passphrase:
  wrote extracted data to "secret.txt."
```

These are just basic examples of how to use Steghide. You can read the project's documentation to learn about several other useful commands.

Source http://www.linux.com/archive/feed/45440

Note steganography is not encryption it can be detected 90% of the time and decoded in under 5 mins for 100 files. Other 10% can be detected by brute force and decoded this takes more time.

PGP

Pretty Good Privacy

PGP version 6.5.8 is readily available at the MIT download site:
<http://web.mit.edu/network/pgp.html>

Getting Started with PGP

Sourcehttp://www.justlinux.com/nhf/Security/Getting_and_Installing_Command_Line_PGP_for_Linux.html

We are going to generate our public and private key pair. PGP can be evoked in any directory but lets stay where we are and type:
pgp -kg

You will be prompted to make some choices and generate some keystrokes that will make your keys unique. I chose a 2048-bit key and the RSA algorithm because that is what most of my fellow conspirators use.

When you are finished generating your keys type:
ls -a

Now a directory ".pgp" should be visible. In it reside your public and private keyrings, pubring.pkr and secring.skr. Let us proceed.
cd .pgp

Note: At any time now you can type "pgp -h" for the basic PGP commands available for your use. Also, in the /usr/doc/pgp-6.5.8 directory there are some excellent .txt and .pdf files you can read with vi, xpdf, or the Adobe Acrobat Reader.

In the /usr/doc/pgp-6.5.8 directory is a file named SampleKeys.asc. We are going to add a public key block to your public keyring–in a minute we are going to check to see if the .rpm file we used to install PGP was good.
cd /usr/doc/pgp-6.5.8

Type:
pgp -ka SampleKeys.asc

Let's check to see if we have a good version of the .rpm file. Type:
rpm –checksig PGPcmdln_6.5.8_Lnx_FW.rpm

If we have a good copy we will get a response which looks like this:
PGPcmdln_6.5.8_Lnx_FW.rpm: md5 OK

We need to extract a copy of your public key so that you can exchange keys with other PGP users. I am going to use the -a option to get an ASCII armored file. This is useful because you can view the output of the file with a text editor. You can demonstrate to others the uniqueness of public keys and win PGP converts, hopefully.
cd .pgp
pgp -kxa Chuck <userid> moose.asc <name of the public key file> pubring.pkr
vi moose.asc <Chuck Steele's public key>

moose.asc will look something like this when viewed with vi

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: PGP 6.5.8

mQENAzyiCa8AAAEIAMzeOhDS0y4pURNO0+iGuQUZuoZISLwCF4ZM/jZDOXwv403I
Ka9ITcOw2SLN+UF9yUXqwznYRk5xE1GZuc6wOe6RZH8pK2DWQ1mZR1bsLP1ZC1Z9
QNSrzZ4qMYatidQwUE5+iKmxpYJmeVX0gRV5MshHPGuXgsazvT4/TecuBw73F6oS
YLz33H/y7EzovasqWxgOBc7t2lEZTcsMJN+/2W6OEyH5jNdFA0VUtaHsnxH35KCL
HRh37GorkeMwrw80JOTHoFkbSk8do+6X54CnA42RReo8n4t1KZxNdQLq51dRXCN6
1qLdI8iiN70NOyts6ZGOVK3KogTtuSHUqX3aPKsABRG0IENodWNrIFN0ZWVsZSA8
c2FuZHNwdXJAc2VpaS5uZXQ+iQEVAwUQPKIJr7kh1Kl92jyrAQFcRwf/bFYzGaVz
e/QKTE97i1orPrfJMBpbbHvkdV1vm1gVQDEM7O7NYy5kYef57lI55yNqdjI4eowc
xoYRWNRUsnIMps2xy0RbjIhAJ5XUBT1tTrdTrQHQuV1QkV+Etbl7zXRbFnctORvl
e7N+BrL9gh4kE256NAFQpbBdCoZ30xudukGliBMAeb97RQmhbyey9JTooftm2Vuo
Izcsan/LlpWCToOiBHyyjYY3a6fdryvuDJePsxJNAoPsA02WPIH6Y7Tcnz/02Im5
R+URyl99cW6lj55WHn/p3PdCQpyJNKOfPjzRxY5j9WohuZU3H3YEV++8yI712xr0
TJNAQMR708/BXw==
=LdjR
-----END PGP PUBLIC KEY BLOCK-----
```

The rest is up to you. You will need to exchange public keys with other users and add their keys to your public keyring. This can be done via e-mail attachments, on a floppy, or by posting your key on a keyserver.

GPG

gpg is the main program for the GnuPG system.

SYNOPSIS

gpg [–homedir **name**] [–options **file**] [**options**] **command** [**args**]

Task: encrypt file

To encrypt single file, use command gpg as follows:
$ gpg -c filename

To encrypt myfinancial.info file, type the command:
$ gpg -c myfinancial.info
Output:

```
Enter passphrase:<YOUR-PASSWORD>
Repeat passphrase:<YOUR-PASSWORD>
```

This will create a *myfinancial.info.gpg* file.
Option:

- **-c :** Encrypt with symmetric cipher.

Caution if you ever forgot your password aka passphrase, you cannot recover the data as it use very strong encryption.

Task: decrypt file

To decrypt file use gpg command:
$ gpg myfinancial.info.gpg
Output:

```
gpg myfinancial.info.gpg
gpg: CAST5 encrypted data
Enter passphrase:<YOUR-PASSWORD>
```

Decrypt file and write output to file vivek.info.txt you can run command:
$ gpg myfinancial.info.gpg –o vivek.info.txt
Remember if file extension is .asc, it is a ASCII encrypted file and if file extension is .gpg, it is a binary

encrypted file.

Read the man page here http://linux.die.net/man/1/gpg

OpenSSL

To encrypt a file:

```
openssl des3 -salt -in infile.txt -out encryptedfile.txt
```

To decrypt the file:

```
openssl des3 -d -salt -in encryptedfile.txt -out normalfile.txt
```

Secure File Transfers

SFTP (secure file transfer protocol)

To connect to urfix.com here is what you should do.

```
sftp urfix.com
```

You will have to enter your username and password when prompted.

To upload your file file.pdf to the directory in urfix.com

```
cd /home/urfix/backup
put file.pdf
```

To download the file urfix.ps from the directory in urfix.com

```
get urfix.ps
```

If you want to download multiple files, say all PDF files,

```
get *.pdf
```

To quit, type exit. This is just the basic SFTP. To check all the options using SFTP, type man sftp from your shell prompt.

SCP (secure copy)

SCP is used for single file transfers unlike SFTP or FTP, where once connected, you can carry out any

number of transfers.

To upload the file urfix.pdf to the /home/user/backup in the remote computer urfix.com here is what you should do. Lets say the username and password for connecting to urfix.com are user and password respectively, read ahead.

scp urfix.pdf user@urfix.com:/home/user/backup/

You will be prompted for your password, which you should enter. It uploads the file and quits automatically.. all in one operation.

To download the file urfix.ps from the remote directory, here is what you must do.

scp user@urfix.com:/home/user/backup/urfix.ps

If you want to upload the entire perl directory (recursively) here is what you do.

scp -r /home/user/perl user@urfix.com:/home/user/backup/

Although these are only a few ways of protecting your sensitive data. make sure you research all your possible options and the caveats involved in your methods. Also take in consideration that users in your system might be able to open up and read RAM which can lead to many vulnerabilities such as sniffing out your keys and more.

Protect yourselves