OpenPuff

OpenPuff Steganography and Watermarking, sometimes abbreviated **OpenPuff** or **Puff**, is a freeware steganography tool for Microsoft Windows created by Cosimo Oliboni and still maintained as independent software. The program is notable for being the first steganography tool (version 1.01 released on December 2004) that:

- Lets users hide data in more than a single carrier file. When hidden data are split among a set of carrier files you get a carrier chain, with no enforced hidden data theoretical size limit (256MB, 512MB, ... depending only on the implementation)
- implements 3 layers of hidden data obfuscation (cryptography, whitening and encoding)
- extends deniable cryptography into deniable steganography

Cryptography: It is the practice and study of techniques for secure communication in the presence of third parties. More generally, it is about constructing and analyzing protocols that overcome the influence of adversaries and that are related to various aspects in information security such as data confidentiality, data integrity, authentication, and non-repudiation. Modern cryptography intersects the disciplines of mathematics, computer science, and electrical engineering. Applications of cryptography include ATM cards, computer passwords, and electronic commerce.

Decorrelation is a general term for any process that is used to reduce autocorrelation within a signal, or cross-correlation within a set of signals, while preserving other aspects of the signal. A frequently used method of decorrelation is the use of a matched linear filter to reduce the autocorrelation of a signal as far as possible. Since the minimum possible autocorrelation for a given signal energy is achieved by equalizing the power spectrum of the signal to be similar to that of a white noise signal, this is often referred to as **signal whitening**

Use

OpenPuff is used primarily for anonymous asynchronous data sharing:

- the sender hides a hidden stream inside some public available carrier files (*password* + *carrier files* + *carrier order* are the secret key)
- the receiver unhide the hidden stream knowing the secret key

The advantage of steganography, over cryptography alone, is that messages do not attract attention to themselves. Plainly visible encrypted messages — no matter how unbreakable — will arouse suspicion, and may in themselves be incriminating in countries where encryption is

illegal. Therefore, whereas cryptography protects the contents of a message, steganography can be said to protect both messages and communicating parties.

Watermarking is the action of signing a file with an ID or copyright mark. OpenPuff does it in an invisible steganographic way, applied to any supported carrier. The invisible mark, being not password protected, is accessible by everyone (using the program)

Supported Formats

Images: BMP, JPG, PCX, PNG, TGA (Truevision Graphics Adapter)

Audios: AIFF, MP3, NEXT/SUN, WAV

Videos: #GP, FLV, MP$, MPG, SWF, VOB

Flash-Adobe: PDF

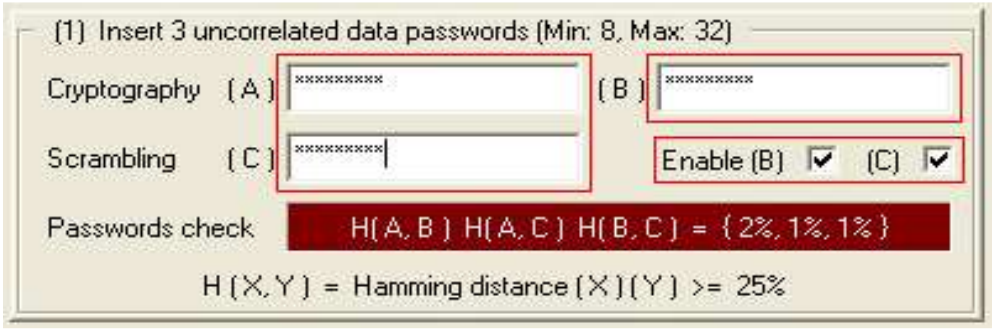
Carriers will keep their format

- **In**: 32 bits per plane TGA, **Out**: 32 bits per plane TGA
- **In**: Stereo WAV, **Out**: Stereo WAV
- **In**: RGB + Alpha BMP, **Out**: RGB + Alpha BMP

etc…

Data Hiding Steps

1.

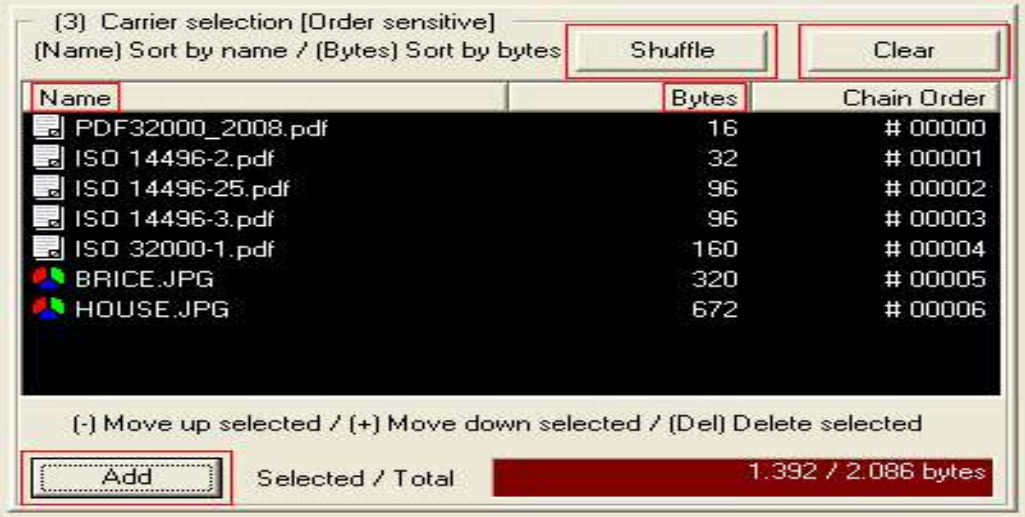


Insert three separate passwords. Each password has to be different (at bit level) and at least 8 c characters long. Password type and number can be easily customized disabling the second (B) and/or the third (C) password. Disabled passwords will be set as the first (A) password.

2.

Choose the secret data you want to hide (typically a zip/rar/… archive).

3.



Shuffle: Random shuffle all carriers
Clear: Discard all carriers
Add: Add new carriers to the list
Name / Bits: Sort carriers by name or bits
+ / - : Move selected carriers up/down
Del: Delete selected carriers

Until selected bytes < total bytes try

- Adding new carriers
- Increasing bit selection level

4.

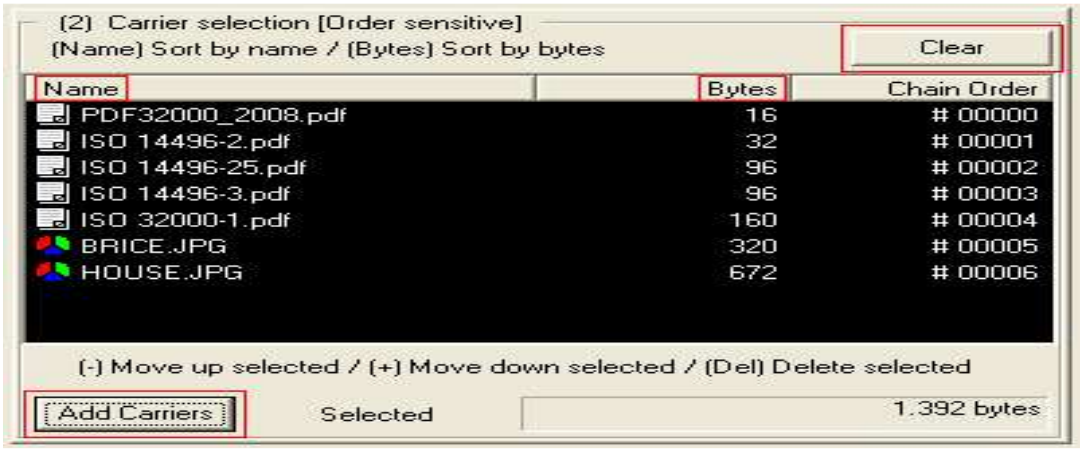
Reset Options: Reset all bit selection level to normal
Add Decoy: Add a decoy (Deniable Steganography)
Hide: Start hiding

Data Unhiding Steps

1.

Inserts your passwords, enabling only those used at the hiding time.

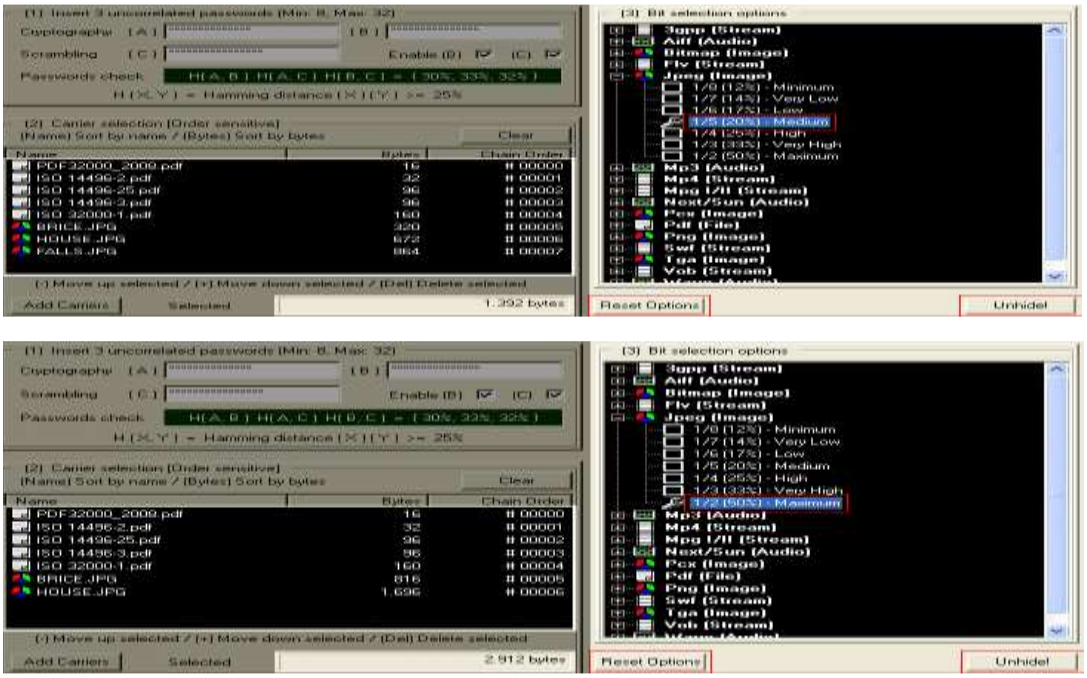
2.



Shuffle: Random shuffle all carriers
Clear: Discard all carriers
Add: Add new carriers to the list
Name / Bits: Sort carriers by name or bits
+ / - : Move selected carriers up/down
Del: Delete selected carriers

Add carriers that have been processed during the hide task.

3.



Reset Options: Reset all bit selection level to normal

Unhide: Start Unhiding

After

- Typing twice the same password
- Adding all the carriers in the right order
- Setting all bit selection levels to the original value

Start the unhiding task.

Mark Setting Steps

1.

Insert mark: Your mark

Type once your mark.

2.

Clear: Discard all carriers
Add: Add new carriers to the list
Name: Sort carriers by name
Del: Delete selected carriers
Set Mark: Start mark setting

Add all the carriers that need to be marked.
Start the setting task.

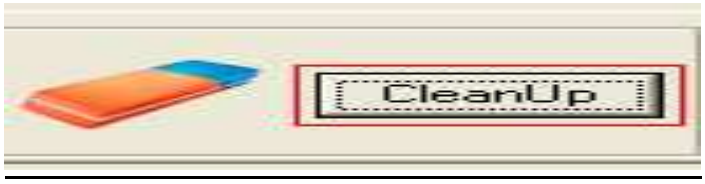
Mark Checking Steps

1. 

   Clear: Discard all carriers
   Add: Add new carriers to the list
   Name: Sort carriers by name
   Del: Delete selected carriers
   Set Mark: Start mark setting

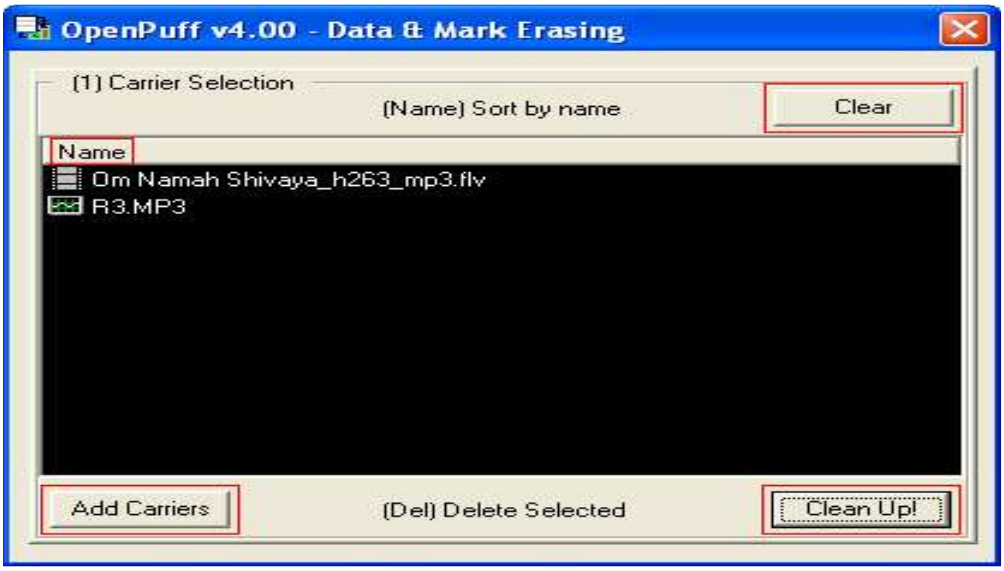
Add all the carriers that need to be checked. Start the checking task.

Data & Mark Erasing Steps

1. 

   Select Clean Up.

2.



Clear: Discard all carriers
Add: Add new carriers to the list
Name: Sort carriers by name
Del: Delete selected carriers
Set Mark: Start mark setting

Add all the carriers that need to be cleaned and start the cleaning task.