Fábrica de Noobs Reloaded

Esteganografia – Utilizando o OpenPuff

Depois de apresentar dois métodos simples de esteganografia, os quais poderiam ser facilmente burlados, mostrarei nos próximos vídeos três programas próprios para a função, cada um com suas vantagens e desvantagens.

Para facilitar a procura, deixarei, junto com este pdf, uma pasta com os programas utilizados. Hoje, utilizaremos o OpenPuff.

- **OpenPuff**

Esse programa permite ocultar qualquer tipo de arquivo dentro de outros, com uma variedade muito maior de formatos de saída, possibilidade de usar múltiplos arquivos, e principalmente: maior segurança.

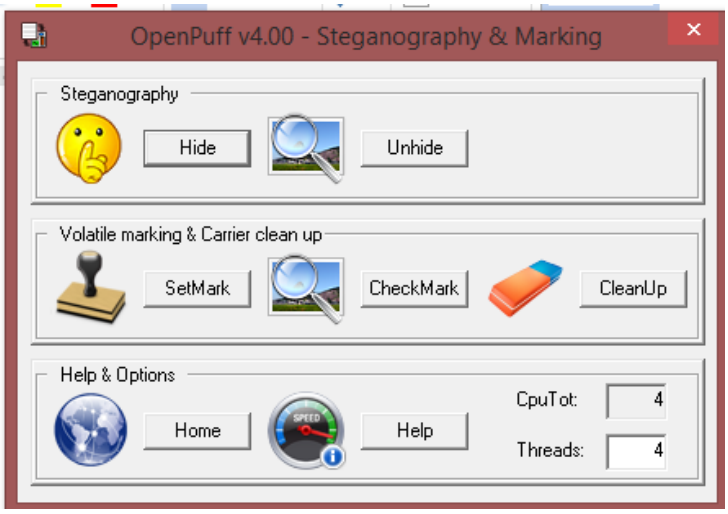
Em primeiro lugar, diferentemente dos outros modos, não há como o usuário sequer desconfiar que há um arquivo escondido dentro de outro, considerando que, mesmo com o arquivo final aberto no OpenPuff, é necessário uma série de chaves (até 3 senhas, de no máximo 32 caracteres para extraí-lo), fora a possibilidade de criar outro conjunto de senhas pra extrair um “decoy”, ou arquivo falso. Resumindo, é difícil até pra você recuperar seus próprios arquivos.

Instalação:

1. O programa é auto executável. Basta abrir a pasta e clicar em OpenPuff.

Vamos agora ao uso. Por ser um programa complexo, vou dividir em partes.

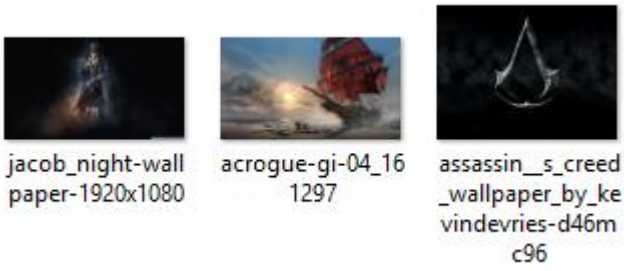
Essa é a interface inicial do programa:



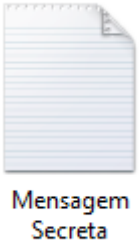
Na seção Steganography, podemos esconder e revelar dados ocultos.

Na seção Volatile marking & Carrier clean up, podemos criar marcas d'água para comprovar nossa autoria sobre algum conteúdo.

Começaremos com a seguinte proposta: usar as imagens de um álbum sobre Assassin's Creed logo abaixo para esconder o arquivo em seguida, feito no Excel, de forma que possa-se aplicar um conjunto de procedimentos aos arquivos finais, e recuperar o arquivo original.



arquivos "fachada"



Arquivo original

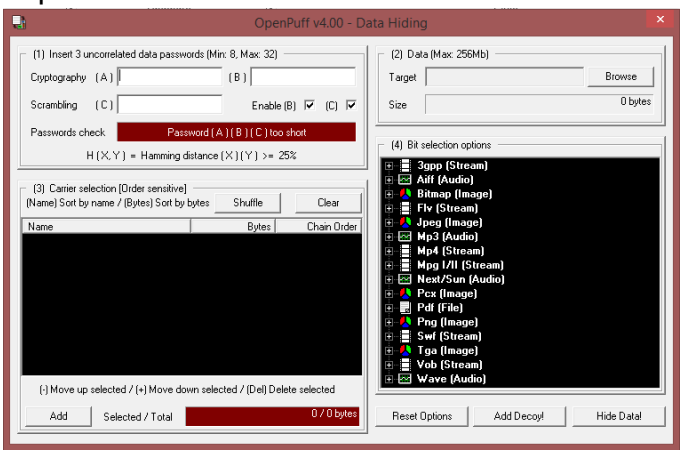
Um dos problemas que tínhamos nos outros métodos era o tamanho da imagem final: se uma foto em baixa resolução tem 2 mb, há algo suspeito. Com o OpenPuff, esse problema é solucionado não usando apenas uma imagem (ou outro tipo de arquivo) mas sim várias. Dessa forma, o arquivo original fica escondido em partes, uma dentro de cada arquivo camuflado.

Ocultar os Dados:

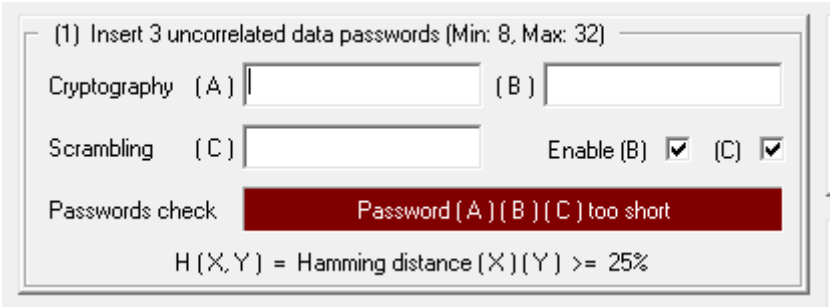
Vamos ao passo a passo:

- Para ocultar os arquivos:

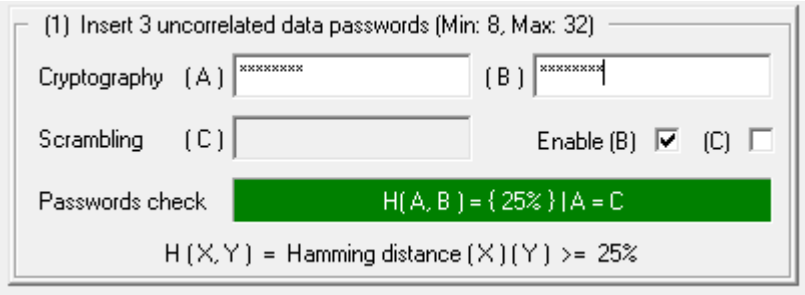
1. No menu inicial, clique em Hide.
2. A seguinte janela irá se abrir. Agora, devemos seguir alguns procedimentos:



- Definir as senhas para a extração.



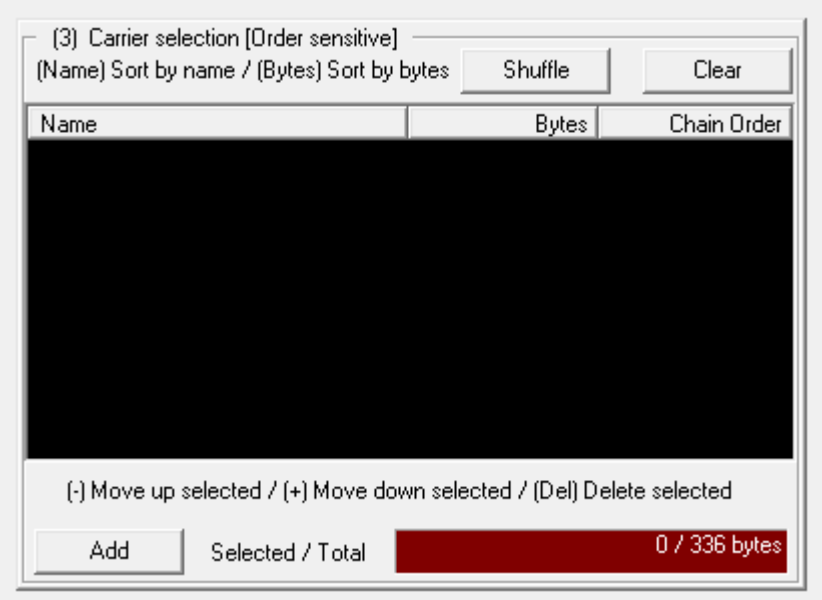
3. Observe que é possível usar até 3 senhas, de 8 a 32 caracteres, podendo usar apenas uma, se preferir.
4. Nos campos (A),(B) e (C) colocamos as senhas desejadas. Guarde essas para extrair depois.
5. Desmarcando as caixas Enable, podemos desabilitar as senhas adicionais.
6. Faça os ajustes necessários até o campo ficar verde, e siga para o próximo passo.
7. Nesse caso, irei colocar apenas as senhas A e B, como 12345678 e abcdefgh, respectivamente.



- Escolher o arquivo original.

8. Clique em Browse.
9. Selecione o arquivo, e observe quantos bytes ele ocupa, tendo um máximo de 256Mb.

- Inserir os arquivos falsos.

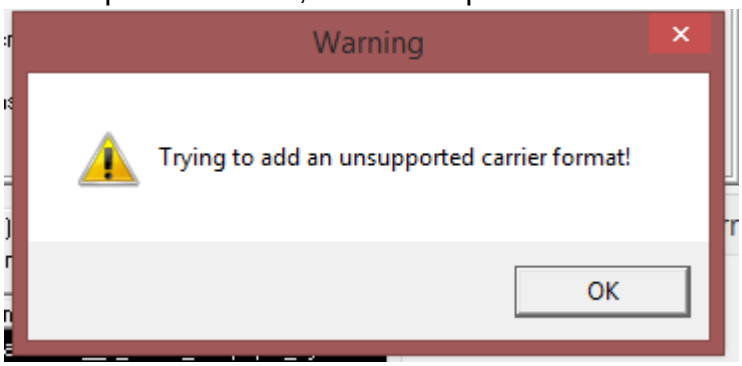


10. Ao completar o passo anterior, você irá notar que na caixa número 3 (acima), o campo Selected / Total se tornou vermelho. À esquerda da barra temos quantos bytes serão necessários para ocultar o arquivo, enquanto que, à direita, vemos quantos bytes estão disponíveis para tal.

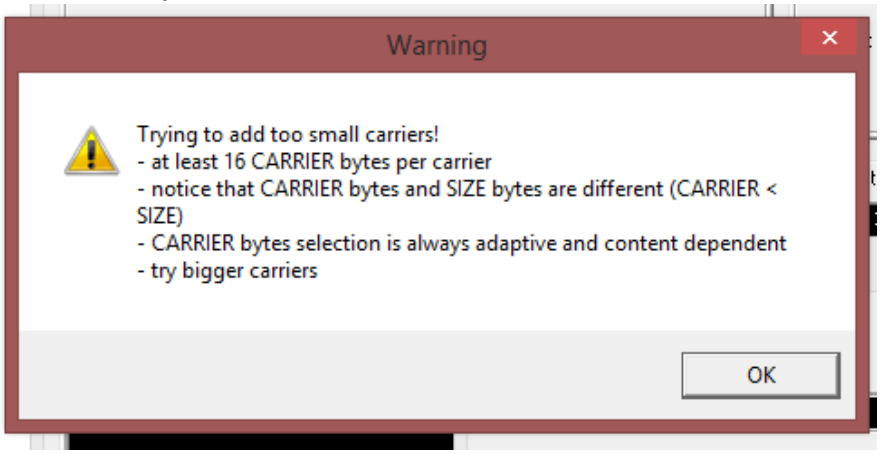
Inicialmente, esse valor sempre será 0. Observe que mesmo arquivos pequenos tendem a precisar de uma grande quantidade de bytes. Isso se deve ao nível de segurança usado no OpenPuff.

11. Agora, devemos adicionar arquivos dos seguintes formatos para preencher os bytes necessários.

12. Para isso, clique em Add e selecione o arquivo desejado. Em tal procedimento, dois erros podem acontecer.

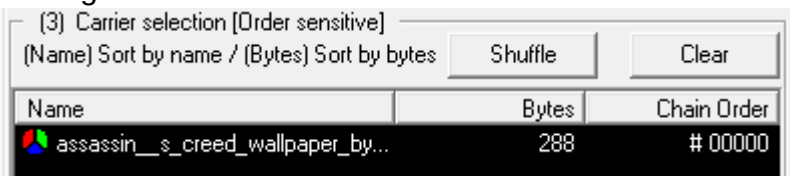


Esse indicia que o formato adicionado não é aceito pelo OpenPuff, provavelmente porque não condiz com os listados no passo 11.



E esse indica que o arquivo adicionado não tem bytes suficientes. É algo um tanto comum de acontecer, por isso, requer paciência.

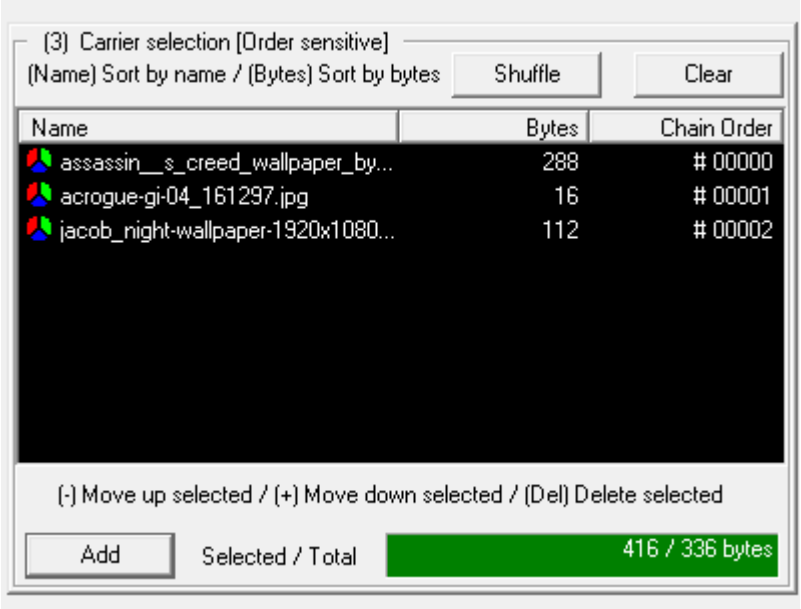
13. Toda vez que algum arquivo for adicionado, irá aparecer da seguinte forma:



Name representa o nome do arquivo, que será mantido após a conclusão.

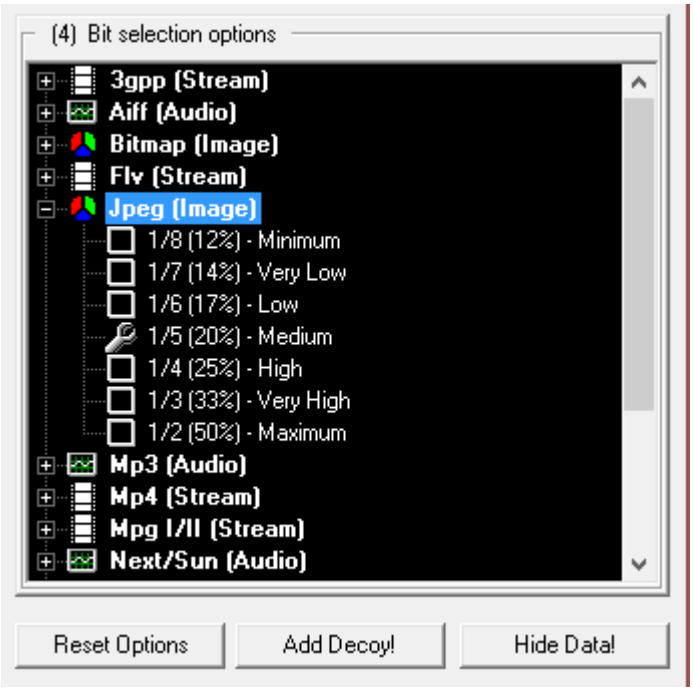
Bytes representa a quantidade de bytes disponíveis dentro do arquivo.
Chain Order representa a ordem com que eles são selecionados. Será fundamental para a posterior recuperação da memsagem. Não se preocupe em guardar tal ordem, uma vez que, ao final, será dado um relatório com todas as informações necessárias para extração.

14. Vá adicionando arquivos até a caixa inferior ficar verde. Note que não há problema nenhum em extrapolar a quantidade necessária. Caso após ter ultrapassado, coloque mais arquivos, eles não serão usados.



15. Clicar em Shuffle fará com que a ordem dos arquivos seja aleatorizada. Clicar em Clear irá apagar todos os arquivos.
16. Apertando + e – podemos alterar a ordem dos mesmos.

- Alterar nível de uso de Bits.



17. Agora, nos resta definir o nível de segurança. Na caixa preta, temos os tipos de arquivos aceitos (no passo anterior) e seus respectivos níveis de seleção. Na prática, não há muita diferença entre eles. Vale observar que, quanto mais alto for a porcentagem, maior espaço livre estará disponível nos arquivos, e portanto, menor segurança.
18. Clicando em Reset Options podemos retornar aos níveis originais. Mais adiante nesse tutorial mostrarei como utilizar Add Decoy!.

- Exportando arquivo Esteganografado.

19. Clique em Hide Data!.
20. Selecione uma pasta para a conclusão.
21. Clique em Ok.
22. Aguarde.
23. Se tudo ocorrer bem, o programa irá retornar um relatório semelhante a esse:

**** Begin of Report ****

Bit Selection:
Jpeg (Image) <- 1/5 (20%) - Medium

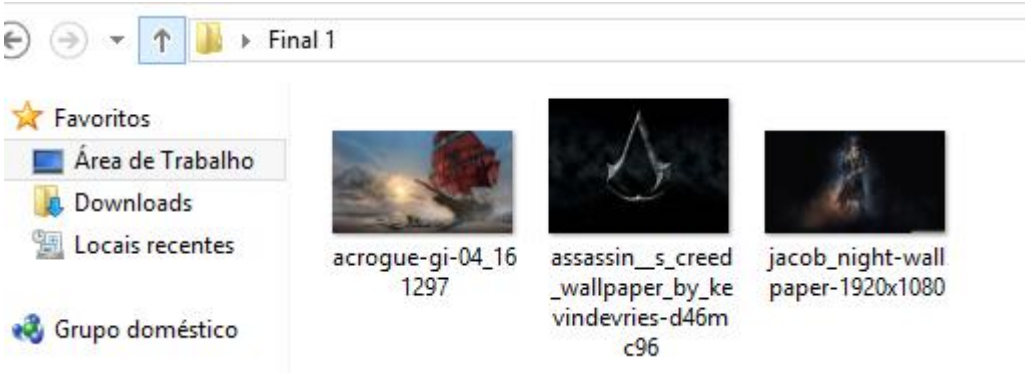
Carriers:

#00000 <- assassin__s_creed_wallpaper_by_kevindevries-d46mc96.jpg
#00001 <- acrogue-gi-04_161297.jpg
#00002 <- jacob_night-wallpaper-1920x1080.jpg

**** End of Report ****

A parte em verde representa o nível de bits, e deve ser configurado conforme especificado na hora da extração. Se você não alterou nada ao ocultar, não será necessário alterar também ao extrair.
A parte em amarelo representa a ordem dos arquivos inseridos. Deverá ser seguida ao fazer a extração.

24. Acesse a pasta escolhida, e verifique o resultado final:



25. Observe que o tamanho das images continua o mesmo. Isso é um fator crucial no quesito segurança, que o programa cumpre bem.

Recuperar os Dados:

Agora, vamos ver como recuperar os arquivos escondidos.

1. Na interface incial, clique em Unhide.
2. Na caixa (1), insira as senhas A, B e C especificadas.
3. Na caixa (2), insira os arquivos finais na sequência especificada no relatório.
4. Na caixa (3), marque o nível de bits, também especificado no relatório.
5. Clique em Unhide!.
6. Selecione a pasta.
7. Clique em Ok.
8. Aguarde.
9. Se tudo ocorrer bem, irá surgir um relatório indicando o arquivo recuperado, e seu tamanho.
10. Vá na pasta especificada e confira.

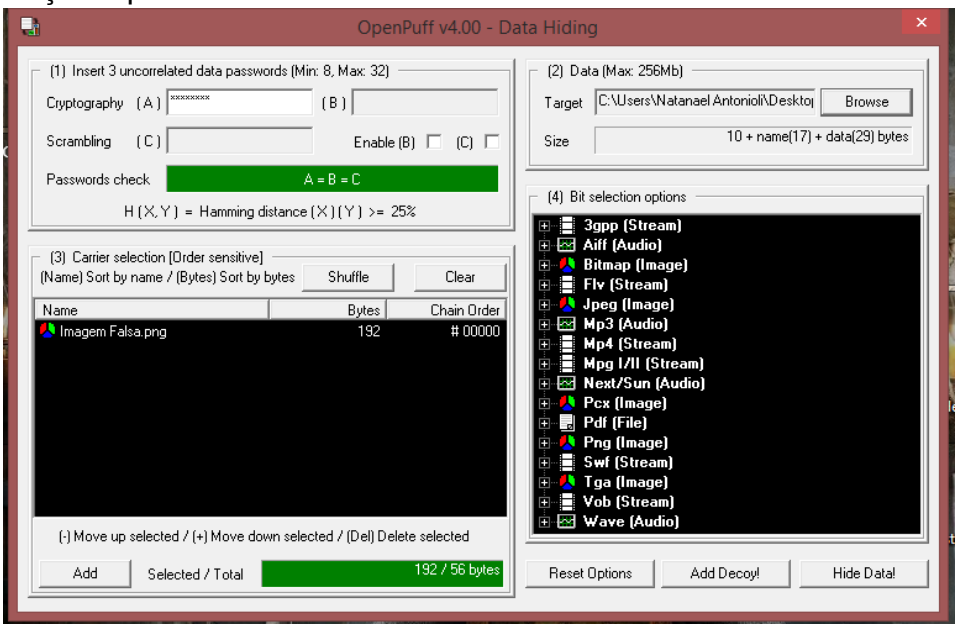
Bônus: inserindo uma Decoy.

Uma decoy é um arquivo falso que podemos inserir juntamente com o arquivo original para ser extraído caso uma determinada senha seja inserida. O único requisito para isso é que o tamanho da decoy e do arquivo verdadeiro sejam os mesmos.

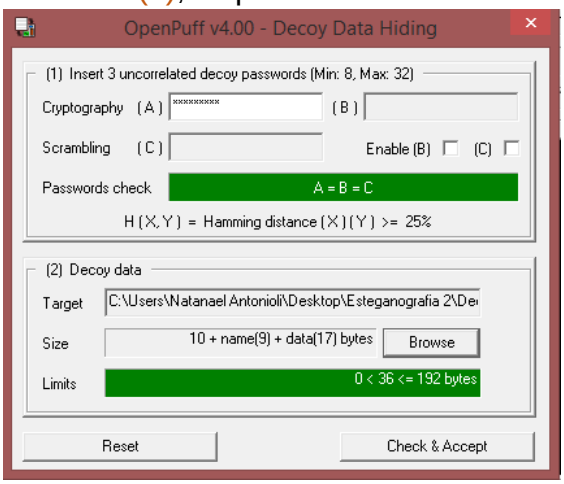
Para facilitar, usaremos dois arquivos em texto como originais e decoy, ambos de 1KB, e uma única imagem como fachada, lembrando que o procedimento é o mesmo para qualquer outro tipo ou tamanho de arquivo.

O arquivo verdadeiro pode ser extraído com a senha 12345678, e o falso, com 123456789.

1. Faça os passos anteriormente descritos até o 18.



2. Clique em Add Decoy!.
3. Na caixa (1), insira a(s) senha(s) da decoy. Elas precisam ser obrigatoriamente diferentes das do arquivo original.
4. Na caixa (2), clique em Browse e insira a decoy.



5. Clique em Check & Accept.
6. Continue a sequência a partir do passo 19. Ao extrair com a senha especificada, teremos a decoy.