

Seu Segundo Programa: Usando o Mal para o Bem

Segurança da Informação

Charles Tim Batista Garrocho

Instituto Federal de São Paulo – IFSP
Campus Campos do Jordão

garrocho.ifspcj.o.edu.br/SEGA6

charles.garrocho@ifsp.edu.br

Curso Superior de TADS



INSTITUTO FEDERAL

Usando o Mal para o Bem

Segundo Peiter Mudge Zatkó, não existem ferramentas ofensivas ou defensivas, existem simplesmente **ferramentas**.

Ao longo dessa **disciplina** você encontrará scripts e ferramentas de natureza ofensiva. Ex.: O Quebrador de Senha UNIX.

Uma pessoa mal intencionada pode usar esse programa para obter acesso não autorizado. No entanto, um programador poderia usar isso tanto para o **bem quanto para o mal**. Correto?

Um caso real aconteceu em 2007, quando o departamento de bombeiros de Brownsville (Texas), recebeu uma **denúncia anônima**, que John Zimmerman, de 50 anos, acessava pornografia infantil usando os recursos do departamento.



INSTITUTO FEDERAL

Usando o Mal para o Bem

O departamento de bombeiros **concedeu** a polícia acesso ao computador de trabalho e ao disco rígido externo de Zimmerman.

O departamento de polícia trouxe o programador da cidade, Albert Castillo, para **pesquisar os conteúdos** do computador de Zimmerman.

A investigação inicial de Castillo **encontrou** várias imagens pornográficas para adultos, mas sem pornografia infantil.

Entretanto, Castillo encontrou alguns **arquivos suspeitos**, incluindo um arquivo ZIP protegido por senha intitulado *Cindy 5*.

Castillo usou um **ataque de dicionário** para descriptografar os arquivos. Os arquivos descriptados resultantes mostraram pornografia infantil. Com essa informação, um juiz ordenou investigar o lar de Zimmerman, onde descobriram mais pornografia infantil.



INSTITUTO FEDERAL

Um Quebrador de Senha de Arquivo Zip – Contextualização

Para construir esse programa, será utilizado a **técnica de força bruta e um dicionário**, aplicado no primeiro programa de senha UNIX.

Entretanto, agora o quebrador de senha será utilizado para quebrar senhas de **arquivos compactados** (Zip).

Para isso, será utilizado a biblioteca de arquivos zip do Python, chamada de **ZipFile**. Essa biblioteca será fundamental para elaboração deste programa.

O principal método desta biblioteca que pode-se observar é o **extractall()**, que possui um parâmetro opcional para especificar uma senha.



INSTITUTO FEDERAL

Um Quebrador de Senha de Arquivo Zip – Implementação

Para construir esse programa, será utilizado a **técnica de força bruta e um dicionário**, aplicado no primeiro programa de senha UNIX.

Entretanto, agora o quebrador de senha será utilizado para quebrar senhas de **arquivos compactados** (Zip).

Para isso, será utilizado a biblioteca de arquivos zip do Python, chamada de **ZipFile**. Essa biblioteca será fundamental para elaboração deste programa.

O principal método desta biblioteca que pode-se observar é o **extractall()**, que possui um parâmetro opcional para especificar uma senha.



INSTITUTO FEDERAL

Um Quebrador de Senha de Arquivo Zip – Implementação

O script aceitará o arquivo criptografado e o dicionário. Para isso, será utilizado a biblioteca **optparse** para analisar as opções de entrada.

```
16 def inicio():
17     analisador = optparse.OptionParser("use %prog "+\
18         "-f <arquivozip> -d <dicionario>")
19     analisador.add_option('-f', dest='nomezip', type='string',\
20         help='especifique o arquivo zip')
21     analisador.add_option('-d', dest='nomedic', type='string',\
22         help='especifique o arquivo dicionario')
23     (opcoes, argumentos) = analisador.parse_args()
24     if (opcoes.nomezip == None) | (opcoes.nomedic == None):
25         print analisador.usage
26         exit(0)
27     else:
28         nomezip = opcoes.nomezip
29         nomedic = opcoes.nomedic
```

Caso o usuário não informe o arquivo criptografado e o arquivo dicionário o programa é finalizado com uma mensagem de erro.



INSTITUTO FEDERAL

Um Quebrador de Senha de Arquivo Zip – Implementação

Com os dois arquivos de entrada informados corretamente, é criada uma instância **zip** do arquivo criptografado utilizando a biblioteca **zipfile**.

```
31     ArquivoZip = zipfile.ZipFile(nomezip)
32     arquivoDici = open(nomedic)
33
34     for linha in arquivoDici.readlines():
35         senha = linha.strip('\n')
36         t = Thread(target=extrairArquivo, args=(ArquivoZip, senha))
37         t.start()
```

Após isso, é percorrido as palavras do **dicionário**. Isso é realizado através do **for** e da função **readlines** que cria uma lista de palavras.

Para cada palavra é iniciado uma *thread* que irá executar a função **extrairArquivo**.





INSTITUTO FEDERAL

Um Quebrador de Senha de Arquivo Zip – Implementação

A função **extrairArquivo** irá tentar extrair todas informações da instância zip através de uma senha.

Se a extração for bem sucedida, as informações serão extraídas e será impresso uma mensagem de senha encontrada.

```
8 def extrairArquivo(arquivo, senha):
9     try:
10         arquivo.extractall(pwd=senha)
11         print '[+] Senha encontrada: ' + senha + '\n'
12     except:
13         pass
```

Caso contrário, e a senha não seja a correta, ocorrerá uma excessão no código, e através do comando **pass**, *não será feito nada*, e o script   continuará sua execução.



Um Quebrador de Senha de Arquivo Zip – Executando

O script abaixo mostra o **resultado** da execução do quebrador de senha zip. Neste caso, a senha do arquivo zip foi quebrada com sucesso.

```
tim@charles:~$ python quebrador_senha_zip.py -f mal.zip -d dicionario.txt  
[+] Senha encontrada: laranja  
  
tim@charles:~$
```

No ubuntu, para compactar uma pasta com uma senha diferente, você terá que utilizar o seguinte comando em terminal:

zip -password laranja -er mal.zip mal

Este comando é necessário para **sinalizar** ao zip a necessidade de criptografia.



INSTITUTO FEDERAL

Modifique o script de forma que ao encontrar a senha do arquivo zip, ele envie a senha para o seu e-mail pessoal.

Além disso, tente anexar os arquivos encontrados no arquivo zipado ao corpo do e-mail.

