Práctica de laboratorio: Encriptar y desencriptar datos utilizando una herramienta de Hacker
Objetivos

Parte 1: Crear y encriptar archivos

Parte 2: Recuperar Contraseñas de Archivos Zip Cifrados

Aspectos Básicos / Escenario

Supongamos que trabajamos en una gran corporación que tiene una política corporativa relacionada con los medios extraíbles. Específicamente, estipula que solo se pueden copiar documentos comprimidos cifrados en unidades Flash USB portátiles.

En esta situación hipotética, el Director Financiero (Chief Financial Officer, CFO) está en un viaje de negocios y se ha puesto en contacto con ustedes frenético pidiéndoles ayuda para resolver una emergencia. Mientras estaba de viaje por negocios, trató de descomprimir documentos importantes desde un archivo zip cifrado en una unidad USB. Sin embargo, la contraseña provista para abrir el archivo zip no es válida. El CFO se puso en contacto con nosotros para ver si podíamos hacer algo.

Nota: La situación es simple y solo sirve a modo de ejemplo.

Es posible que haya algunas herramientas disponibles para recuperar contraseñas olvidadas. Esto es especialmente cierto en situaciones como esta, en las que el analista especializado en ciberseguridad podría obtener la información pertinente del CFO. La información pertinente podría ser la longitud de la contraseña y una idea de cuál podría ser. Conocer la información pertinente es radicalmente útil cuando se está tratando de recuperar una contraseña.

Entre algunos ejemplos de utilidades y programas para recuperar contraseñas podemos mencionar los siguientes: hashcat, John the Ripper y Lophtcrack. En nuestro caso, utilizaremos **fcrackzip**, una simple utilidad de Linux para recuperar las contraseñas de archivos zip cifrados.

Tengan presente que los ciberdelincuentes pueden utilizar esas mismas herramientas para averiguar contraseñas desconocidas. Aunque no podrían acceder a cierta información pertinente, con el tiempo es posible que averigüen las contraseñas para abrir archivos zip cifrados. El tiempo necesario depende de la solidez y de la longitud de la contraseña. Las contraseñas más largas y más complejas (que realizan una combinación de diferentes tipos de caracteres) son más seguras.

En esta práctica de laboratorio:

- Crearemos y cifraremos archivos de texto de ejemplo.
- Descifraremos el archivo zip cifrado.

Nota: Esta práctica de laboratorio debe utilizarse solo con fines instructivos. Los métodos aquí presentados NO se deben emplear para asegurar datos realmente sensibles.

Recursos necesarios

- Máquina virtual "CyberOps Workstation"

Instrucciones

Parte 1: Crear y Cifrar archivos

En esta parte crearemos algunos archivos de texto que se utilizarán para generar los archivos zip cifrados del próximo paso.

Paso 1: Crear archivos de texto

a. Iniciar la Máquina Virtual CyberOps Workstation.

b. Abrir una ventana del terminal. Verificar que están en el directorio de inicio de analyst. Si no es así, introducir **cd ~** en el prompt del terminal.

c. Crear una carpeta nueva de nombre Zip-Files con el comando **mkdir Zip-Files**.

d. Ingresar a ese directorio con el comando **cd Zip-Files**.

e. Introduzca el siguiente texto para crear tres archivos de texto.

```
[analyst@secOps Zip-Files]$ echo This is a sample text file > sample-1.txt
[analyst@secOps Zip-Files]$ echo This is a sample text file > sample-2.txt
[analyst@secOps Zip-Files]$ echo This is a sample text file > sample-3.txt
```

f. Utilicen el comando **ls** para verificar que se hayan creado los archivos.

```
[analyst@secOps Zip-Files]$ ls -l
total 12
-rw-r--r-- 1 analyst analyst 27 May 13 10:58 sample-1.txt
-rw-r--r-- 1 analyst analyst 27 May 13 10:58 sample-2.txt
-rw-r--r-- 1 analyst analyst 27 May 13 10:58 sample-3.txt
```

Paso 2: Comprimir y cifrar los archivos de texto

A continuación, crearemos varios archivos comprimidos cifrados con contraseñas de diversas longitudes. Para hacerlo, cifraremos los tres archivos de texto con la utilidad **zip**.

a. Utilicen el siguiente comando para crear un archivo zip cifrado de nombre **file-1.zip** que contenga los tres archivos de texto:

```
[analyst@secOps Zip-Files]$ zip –e file-1.zip sample*
```

b. Cuando se le solicite una contraseña, introduzca una de un carácter de su elección. En el ejemplo se introdujo la letra **B**. Introduzcan la misma letra cuando se les solicite verificarla.

```
[analyst@secOps Zip-Files]$ zip -e file-1.zip sample-*
Enter password:
Verify password:
  añadiendo: sample-1.txt (almacenado 0%)
  añadiendo: sample-2.txt (almacenado 0%)
  añadiendo: sample-3.txt (almacenado 0%)
```

c. Repitan el procedimiento para crear los siguientes 4 archivos

- **file-2.zip** con una contraseña de 2 caracteres de su elección. En nuestro ejemplo, utilizamos **R2**.
- **file-3.zip** con una contraseña de 3 caracteres de su elección. En nuestro ejemplo, utilizamos **0B1**.
- **file-4.zip** con una contraseña de 4 caracteres de su elección. En nuestro ejemplo, utilizamos **Y0Da**.
- **file-5.zip** con una contraseña de 5 caracteres de su elección. En nuestro ejemplo, utilizamos **C-3P0**.

d. Utilice el comando **ls -l f*** para verificar que se hayan creado todos los archivos comprimidos.

```
[analyst @secOps Zip Files] $ ls -l f*
-rw-r--r-- 1 analyst analyst 643 May 13 11:01 file-1.zip
-rw-r--r-- 1 analyst analyst 643 May 13 11:02 file-2.zip
-rw-r--r-- 1 analyst analyst 643 May 13 11:03 file-3.zip
-rw-r--r-- 1 analyst analyst 643 May 13 11:03 file-4.zip
-rw-r--r-- 1 analyst analyst 643 May 13 11:03 file-5.zip
```

e. Traten de abrir un zip con una contraseña incorrecta, tal como se muestra.

```
[analyst@secOps Zip-Files]$ unzip file-1.zip
Archive: file-1.zip
[file-1.zip] sample-1.txt password:
password incorrect--reenter:
password incorrect--reenter:
   skipping: sample-1.txt incorrect password
[file-1.zip] sample-2.txt password:
password incorrect--reenter:
password incorrect--reenter:
   skipping: sample-2.txt incorrect password
[file-1.zip] sample-3.txt password:
password incorrect--reenter:
password incorrect--reenter:
   skipping: sample-3.txt incorrect password
```

Parte 2: Recuperar Contraseñas de Archivos Zip Cifrados

En esta parte utilizaremos la utilidad **fcrackzip** para recuperar contraseñas olvidadas de archivos comprimidos cifrados. Fcrackzip busca archivos cifrados en cada archivo zip dado para adivinar la contraseña utilizando métodos de fuerza bruta.

El motivo por el cual creamos archivos zip con contraseñas de diversas longitudes es ver si la longitud de la contraseña tiene alguna influencia sobre el tiempo necesario para descubrirla.

Paso 1: Introducción a fcrackzip

En la ventana del terminal, introducimos el comando **fcrackzip –h** para ver las opciones del comando asociadas.

En nuestros ejemplos utilizaremos las opciones de comando **–v**, **-u** y **-l**. La opción -l se incluirá a lo último porque especifica la posible longitud de la contraseña. Tienen plena libertad de experimentar con otras opciones.

Paso 2: Recuperar Contraseñas utilizando fcrackzip

a. Ahora traten de recuperar la contraseña del archivo **file-1.zip**. Recordemos que se utilizó una contraseña de un carácter para cifrar el archivo. Por lo tanto, utilizaremos el siguiente comando **fcrackzip**:

```
[analyst@secOps Zip-Files]$ fcrackzip -vul 1-4 file-1.zip
found file 'sample-1.txt', (size cp/uc 39/ 27, flags 9, chk 5754)
found file 'sample-2.txt', (size cp/uc 39/ 27, flags 9, chk 5756)
found file 'sample-3.txt', (size cp/uc 39/ 27, flags 9, chk 5757)


PASSWORD FOUND!!!!: pw == B
```

Nota: La longitud de la contraseña se podría haber definido en menos de 1 a 4 caracteres.

¿Cuánto tiempo es necesario para descubrir la contraseña?

b. Ahora traten de recuperar la contraseña del archivo **file-2.zip**. Recuerden que se utilizó una contraseña de dos caracteres para cifrar el archivo. Por lo tanto, utilizaremos el siguiente comando **fcrackzip**:

```
[analyst@secOps Zip-Files]$ fcrackzip –vul 1-4 file-2.zip
found file 'sample-1.txt', (size cp/uc 39/ 27, flags 9, chk 5754)
found file 'sample-2.txt', (size cp/uc 39/ 27, flags 9, chk 5756)
found file 'sample-3.txt', (size cp/uc 39/ 27, flags 9, chk 5757)


PASSWORD FOUND!!!!: pw == R2
```

¿Cuánto tiempo es necesario para descubrir la contraseña?

c. Repetir el procedimiento y recuperar la contraseña del archivo **file-3.zip**. Recordemos que se utilizó una contraseña de tres caracteres para cifrar el archivo. Cronometrar la operación para averiguar cuánto tiempo se necesita para descubrir una contraseña de 3 letras. Utilicen el siguiente comando **fcrackzip**:

```
[analyst@secOps Zip-Files]$ fcrackzip –vul 1-4 file-3.zip
found file 'sample-1.txt', (size cp/uc 39/ 27, flags 9, chk 5754)
found file 'sample-2.txt', (size cp/uc 39/ 27, flags 9, chk 5756)
found file 'sample-3.txt', (size cp/uc 39/ 27, flags 9, chk 5757)


PASSWORD FOUND!!!!: pw == 0B1
```

¿Cuánto tiempo es necesario para descubrir la contraseña?

d. ¿Cuánto tiempo es necesario para averiguar una contraseña de cuatro caracteres? Repitan el procedimiento y recuperen la contraseña del archivo **file-4.zip**. Cronometrar la operación para saber cuánto tiempo es necesario para descubrir la contraseña con el siguiente comando **fcrackzip**:

```
[analyst@secOps Zip-Files]$ fcrackzip –vul 1-4 file-4.zip
found file 'sample-1.txt', (size cp/uc 39/ 27, flags 9, chk 5754)
found file 'sample-2.txt', (size cp/uc 39/ 27, flags 9, chk 5756)
found file 'sample-3.txt', (size cp/uc 39/ 27, flags 9, chk 5757)
checking pw X9M~

PASSWORD FOUND!!!!: pw == Y0Da
```

¿Cuánto tiempo es necesario para descubrir la contraseña?

e. ¿Cuánto tiempo es necesario para averiguar una contraseña de cinco caracteres? Repetir el procedimiento y recuperar la contraseña del archivo **file-5.zip**. La contraseña tiene una longitud de cinco caracteres; por ese motivo, tenemos que definir la opción **-l** del comando en **1-5**. Nuevamente, cronometrar la operación para saber cuánto tiempo es necesario para descubrir la contraseña con el siguiente comando **fcrackzip**:

```
[analyst@secOps Zip-Files]$ fcrackzip –vul 1-5 file-5.zip
found file 'sample-1.txt', (size cp/uc 39/ 27, flags 9, chk 5754)
found file 'sample-2.txt', (size cp/uc 39/ 27, flags 9, chk 5756)
found file 'sample-3.txt', (size cp/uc 39/ 27, flags 9, chk 5757)
checking pw C-H*~

PASSWORD FOUND!!!!: pw == C-3P0
```

¿Cuánto tiempo es necesario para descubrir la contraseña?

f. Recuperar una contraseña de 6 caracteres utilizando fcrackzip

Aparentemente, se necesita más tiempo para descubrir contraseñas más largas y, por lo tanto, son más seguras. Sin embargo, una contraseña de 6 caracteres no desalentará a un ciberdelincuente.

¿Cuánto tiempo creen que demoraría fcrackzip para descubrir una contraseña de 6 caracteres?

Para responder esa pregunta, procedemos a crear un archivo de nombre **file-6.zip** con una contraseña de 6 caracteres de su elección. En nuestro ejemplo utilizamos **Jarjar**.

```
[analyst@secOps Zip-Files]$ zip –e file-6.zip sample*
```

g. Repitan el procedimiento para recuperar la contraseña del archivo **file-6.zip** con el siguiente comando **fcrackzip**:

```
[analyst@secOps Zip-Files]$ fcrackzip –vul 1-6 file-6.zip
```

¿Cuánto tiempo demora fcrackzip para descubrir la contraseña?

La simple verdad es que las contraseñas más largas son más seguras porque se necesita más tiempo para descubrirlas.

¿Qué longitud recomendarías para que una contraseña sea segura?