Práctico 5 - El ultimo: Fuzzing + Forensia

Deadline parte ENTREGABLE: 1 de Diciembre 23:59

Parte entregable

1. Fuzzeando un parser de BMP (http://paulbourke.net/dataformats/bmp/parse.c)

   ¡DESCARGAR AQUI!

   Compilar con make o algo similar a:
   gcc -O0 -o parse -no-pie -fno-pic -fno-stack-protector -m32 parse.c paulslib.c -lm

   Analizar usando alguna herramienta de fuzzing como zzuf o afl.
   Encontrar un crash que permita controlar EIP y documentar el proceso.

2. Resolver el programa binario "**r1**" del TP4 utilizando ejecución simbólica con python usando angr o manticore (u otra herramienta de ejecución simbólica)

3. En una investigación forense es muy importante recolectar la mayor cantidad de evidencias posible, pues en principio, todo puede darnos información importante y/o valiosa. Entre la información más volátil e importante tenemos la información que fluye a través de la red y la memoria RAM de un sistema. En ciertas versiones de windows se puede realizar un dump de la memoria usando el comando dd y enviarla a una estación de trabajo forense que se encuentre escuchando desde un netcat/cryptcat:
   dd if=\\.\PhysicalMemory | cryptcat 10.0.0.1 9000 -p "password"

   La tool/distro Helix ya viene con algunas herramientas de análisis y obtención de evidencia forense y windows provee opciones nativas interesantes.

   a. Investigar las opciones para hacer un dump de memoria en linux. Describir brevemente al menos 2 opciones. ej: LiME
   b. Realizar un dumpeo de memoria de alguna máquina virtual a elección del alumno. (enviar screenshots y hashes, documentando el proceso)

4. La recuperación de archivos es también clave a la hora de buscar evidencia eliminada, para este fin existen muchas herramientas libres de recuperación como por ejemplo TestDisk & PhotoRec[1] o Recuva[2]

[1] http://www.cgsecurity.org/wiki/TestDisk_Download
[2] https://www.ccleaner.com/recuva

a. Usar un pendrive/SD/HD vacío (físico o virtual), copiarle una imagen, vídeo o archivo de audio. Además copiarle el archivo: fsecret_doc.docx

b. Eliminarlos y luego recuperarlos usando Photorec[3] asegurando la recuperación.

c. Analizar las herramientas John the ripper, HashCat, Y Djohn[4] y decidir cual usar para atacar el archivo docx.

d. Obtener la clave del docx utilizando la herramienta elegida en 3.
(pueden usar los diccionarios 1 y 2)

5) Descargar el siguiente dump de memoria:
ubuntu-10.04.3-i386-LiveCD-kbeast.mem.bz2

- Investigar y probar opciones de análisis de la memoria usando el framework volatility para tratar de descubrir si contiene el rootkit Kbeast?
- ¿Qué otra manera de investigar la presencia de un rootkit o malware encuentran?

[3] Pueden usar la tool de recuperación de preferencia.
[4] http://download.openwall.net/pub/projects/john/contrib/parallel/djohn/djohn-0.9.8.1.tgz