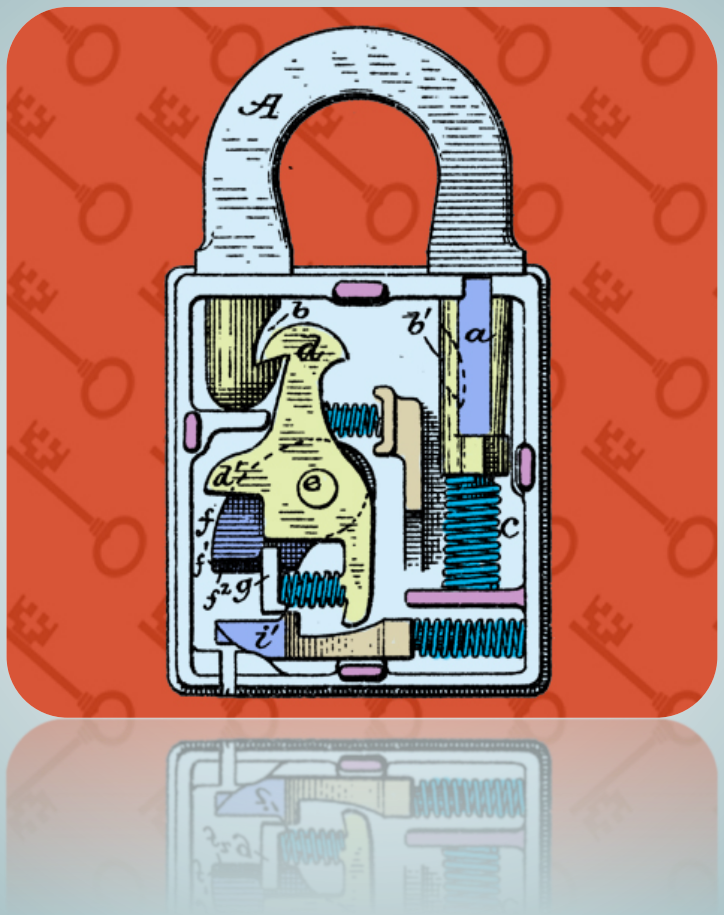
Garantía y Seguridad en Sistemas y Redes

Práctica 4. Rootkits en Espacio de Usuario



Esteban Stafford

Departamento de Ingeniería
Informática y Electrónica

Práctica 4

Rootkits en espacio de usuario

1. Introducción

Un rootkit es un programa o conjunto de programas que se instalan en una máquina comprometida, con el objeto de ocultar actividades maliciosas. Principalmente, los rootkits abren un canal a la máquina, para permitir que el atacante pueda acceder de a la misma sin ser detectado. En esta práctica se verá `Azazel` clasificado como rootkit de area de usuario, porque no modifica el fincionamiento del kernel. Para entrar en ejecución este malware se instala como librería compartida. De manera que cuando los programas que se ejecutan en el sistema se arrancan, incluyen el código del rootkit en su espacio de memoria.

2. Objetivos

Los objetivos fundamentales que persigue esta práctica son los siguientes:

- Ver de que manera se puede instalar un rootkit.
- Aprender las técnicas de ocultación de rootkit de espacio de usuario.

3. Material Necesario

El rootkit `Azazel` se puede encontrar de manera sencilla en GitHub, en la dirección https://github.com/chokepoint/azazel. En código fuente existe un fichero `config.py` en el que se pueden realizar varios ajustes sobre el funcionamiento del rootkit. La compilación se hace fácilmente con el comando `make`, pero para compilarlo en la máquina virtual necesita varios paquetes. Primero `build-essential`, para instalar el entorno de desarrollo de `gcc`. Luego se requieren las cabeceras de varias librerías: `elibssl-dev`, `libpam0g-dev` y `libpcap-dev`.

4. Desarrollo

4.1. Compilación e instalación

Antes de comenzar, conviene resaltar que normalmente un atacante no instalaría un rootkit como haremos en la práctica. Él lo compilaría previamente una máquina virtual con una instalación similar a la de la máquina víctima. Y despues, a través de algún engaño o vulnerabilidad lo instalaría en la víctima.

Antes de arrancar la máquina virtual toma una instantánea (snapshot) del estado actual de la misma. De manera que cuando termine esta práctica se pueda volver al

estado inicial fácilmente. Dentro de la máquina virtual compila el rootkit con el comando `make`, pero no lo instales. Revisa el fichero `Makefile` y estudia el script de instalación. ¿Que hace el fichero `/etc/ld.so.preload`? ¿Existe ya en la máquina virtual?

Ahora instala el rootkit ejecutando el comando `sudo make install`. Cumprueba que se crean los ficheros mencionados en el script de instalacion. ¿Existen? ¿Se pueden ver con un `ls`?

Revisa el código fuente de `azazel.c`. ¿Encuentras alguna pista de que está haciendo el rootkit para ocultar su instalación?

4.2. Funciones del rootkit

Una vez instalado vamos a ver qué funcionalidad ofrece `azazel`. En cada paso busca en el código la función correspondiente. Verás que al final para cada funcionalidad se modifica el comportamiento estándar de una o varias llamadas al sistema. Identifica cuales son.

El rootkit `Azazel` tiene varias funciones para ocultar la presencia del atacante. Ya se ha visto que es capaz de ocultar `/etc/ld.so.preload`. ¿Cómo lo hace? ¿Oculta algún otro fichero más?

Un administrador que observe que su máquina se está comportando de manera extraña, puede ser que use el comando `strace` para depurar las llamadas al sistema de algún proceso. ¿Que verá este administrador?

Si el atacante posee una cuenta local no privilegiada, el rootkit le permite acceso privilegiado a traves del comando `su - rootme`. ¿Cómo se ha creado este usuario? ¿Ha dejado algún rastro en los logs del sistema? ¿Aparece la sesión con el comando `who`?

Para ocular la session abierta con `su` se puede ejecutar `CLEANUP_LOGS="<dispositivo>" ls`.

Si por el contrario el atacante no tiene una cuenta local, necesita acceder por la red directamente. Esto es posible a traves del puerto 22 (`ssh`). Desde la máquina local, ejecuta el comando `ncat <dirección>22 -p 61040`, a continuación introduce la clave `changeme`. ¿Qué pasa si no fijas el puerto local?

Mientras la conexión está activa, haz un listado de conexiones con `netstat` ¿Aparece la conexión remota?

Usa `ps` para ver los procesos que están ejecutándose. Puedes usar `lsof` para recabar más información. ¿Puedes encontrar la shell de la conexión remota? ¿Presentan los procesos de ssh alguna anomalia? Usando `who` ¿se puede detectar la conexión?

4.3. Detección y desinstalación

Si ahora tuvieras que analizar una máquina que se está comportando de manera extraña. ¿Que pruebas harías para detectar la presencia de `azazel`?

Una vez detectado, ¿se te ocurre alguna manera de desinstalar el rootkit? ¿Es posible hacerlo con la máquina en marcha, o es necesario arrancarla con un LiveCD?