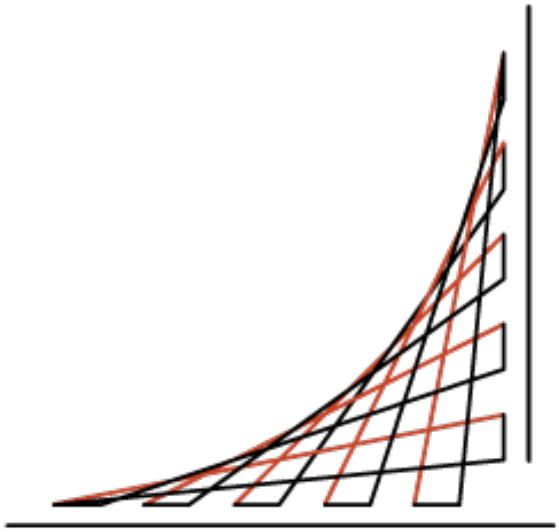

ESCUELA COLOMBIANA DE INGENIERÍA
JULIO GARAVITO

VIGILADA MINEDUCACIÓN

Gestión de Incidentes y Forensia Digital

Taller 1

Comandos para Live Response

Profesor:
Gerson David Quintero Rodríguez

Autores:

Carlos Andrés Amorocho Amorocho

Angie Daniela Ruiz Alfonso

Jorge Enrique Rodríguez Valderrama

Contents

1 Presentación

En los últimos años el concepto de Tecnología de la Información y Comunicación (TIC) ha tomado fuerza, este se refiere al conjunto de recursos, herramientas, equipos informáticos, aplicaciones, redes y medios que permiten la compilación, el procesamiento, almacenamiento, trasmisión de información como voz, datos, texto, video e imágenes.

Como resultado de este acelerado avance y del creciente uso de sistemas computacionales, los datos e información sensible están cada vez más expuestos, aumentando así el factor de riesgo para las personas y organizaciones ante incidentes informáticos, Incidentes que al materializarse pueden ocasionar daños irreparables.

Es allí donde haciendo uso de la Respuesta a Incidentes se pueden hallar una gran cantidad de ataques y evidencias digitales. En este documento se especificarán una serie de comandos para Live Response, con el propósito de detectar intrusiones en los sistemas y poder reaccionar de la manera más rápida y efectiva.

2 Objetivos Generales

- **Especificar los 10 comandos que consideramos más relevantes para Live Response en Linux.**
- **Especificar los 10 comandos que consideramos más relevantes para Live Response en Windows.**
- **Indicar las conclusiones obtenidas acerca de los comandos seleccionados para Live Response.**
- **Priorizar algunos ambientes del sistema para la gestión de incidentes (puertos, red, usuarios, permisos, entre otros).**
- **Facilitar al lector/usuario comandos que le ayuden en la investigación del caso, para obtener evidencias digitales o contención del algún incidente**

3 20 Comandos Relevantes Para Live Response

- **LINUX**
  - **find / -nouser -print:** Permite identificar y mostrar si un atacante creó algún usuario temporal para realizar algun tipo de ataque. Información que puede ser útil para iniciar una investigación de acuerdo con la trazabilidad de los hechos, permitiendo asi identificar posteriormente las acciones realizadas por este. [1]

  - **tail auth.log:** Permite evidenciar si un atacante ha tenido una conexión con el objetivo mediante los protocolos SSH o Telnet. La ubicación para ejecutar el comando debe ser: /var/log. Información que puede ser útil para identificar la puerta de acceso del atacante y su ambiente (interno si la IP coincide con la propia, externo si fue un robo de credenciales o explotación de Telnet). [1]

  - **history | less:** Permite ver el historial de comandos que el usuario ha escrito, por ende, es de suma importancia que este no pueda ser manipulado por terceros. Informacion que puede ser util para verificar y seguir la trazabilidad si algún usuario ejecutó comandos que puedan afectar la confidencialidad, integridad y disponibilidad del servidor. [1]

  - **uptime:** Permite ver el tiempo que ha estado funcionando el servidor, la hora actual del sistema, los usuarios que han iniciado sesión actualmente y los promedios de carga para el sistema. Información que puede ser útil para tener una trazabilidad de los hechos de una forma más precisa. [1]

  - **cat /proc/meminfo:** Permite ver el uso de la memoria del sistema. Más específicamente su uso total, memoria disponible, el volumen que está ocupado actualmente y también se incluye memoria RAM, buffer y swap. Información que puede ser util para identificar o descartar los objetivos del ataque (minería, entre otros). [1]

  - **top:** Permite conocer los procesos que se están ejecutando en tiempo real en el sistema; nos proporciona una información más detallada, como lo son ID del proceso, usuario que lo ejecuta, entre otros. Esto con el fin de detectar algún proceso sospechoso o que no haya sido iniciado por orden de un interno de la organización. [1]

  - **service –status-all:** Permite identificar los servicios que tenemos en el sistema. En la pantalla veremos unos símbolos (+, -) que nos dirá si el servicio está activo o no, lo cual nos facilitara la detección de algún servicio que no haya sido iniciado por alguien de TI y este activo, o viceversa. [1]

  - **cat /etc/crontab:** Este archivo permite constatar las tareas programadas que podemos usar en el sistema; con esta ayuda podemos identificar si alguna

de ellas no fue creada por alguien de la organización. [1]

- **cat /etc/hosts:** Este documento permite ver los nombres de dominio, las direcciones IP que estén configuradas por el sistema. Información que es de gran importancia debido a que los atacantes pueden suplantar nombres de dominio y direcciones IP para engañar a las víctimas. [1]

- **netstat -nap:** Permite visualizar todos los puertos que se estan usando en el sistema, al igual que verificar la dirección a la cual se está conectando (local o externa, si es interna), lo cual es de bastante utilidad para verificar que no existan conexiones sospechosas. [1]

- **WINDOWS**
  - **wmic process list full:** Se ejecuta como administrador en powershell y nos permite ver un listado completo de los procesos que se están ejecutando en el sistema, Esto con el fin de detectar algún proceso sospechoso o que no haya sido iniciado por orden de un interno de la organización. [1]

  - **net start:** Permite identificar los servicios que se encuentran actualmente activos en el sistema, lo cual nos facilitara la detección de algún servicio que no haya sido iniciado por alguien de TI y este activo, o viceversa. [1]

  - **tasklist /svc:** Permite analizar los procesos del sistema, al ejecutarlo podemos evidenciar cuales son los servicios asociados a cada proceso que se encuentra en ejecución. Con el fin de identificar si fueron comprometidos algunos servicios o se crearon nuevos asociados a algún proceso. [1]

  - **shtask:** Permite conocer las tareas programadas que fueron creadas en el sistema, debido a que con esta información podemos detectar si hay alguna anomalía en estas. [1]

  - **wmic startup get caption command:** Mediante Powershell, permite ver las aplicaciones que se están ejecutando cuando iniciamos el sistema. Información que es útil para verificar que los programas que inician junto al sistema sean programas auténticos y no se encuentre algún programa maligno ejecutándose en segundo plano. [1]

  - **reg query HKLM/SOFTWARE/Microsoft/Windows/CurrentVersion/Run:** Permite ver el registro de la máquina local, lo cual es útil en caso de que algún programa sospechoso y poco sofisticado se encuentre en la máquina. [1]

  - **netstat -ano:** Permite visualizar los puertos que están escuchando alguna comunicación y las direcciones IP con las cuales se está comunicando, conexiones entrantes y salientes, tablas de enrutamiento y un detalle de las es-

tadísticas. Con ayuda de esta información, podemos identificar si existe alguna conexión con algún dispositivo que no se encuentre en nuestra red u organización. [1]

- **Get-SMBShare:** Mediante Powershell, permite ver las rutas/carpetas compartidas en el sistema y sobre que entorno se encuentran, ya sea que estén en un entorno controlado, por defecto o compartido para una IP remota. [1]

- **forfiles /D -10 /S /M *.exe /C "cmd /c echo @path":** Permite ver que archivos pueden ser maliciosos o los que terminan en una extensión en particular, de igual forma podemos conocer sus detalles como la ruta o su última fecha de modificación. Con ayuda de este comando podremos identificar archivos maliciosos, que pueden estar escondidos bajo la imagen de un software original; documentos .exe con imagen Word, por ejemplo. [1]

- **netsh firewall show config:** Permite ver la configuración y los ajustes del Firewall, por ende, es importante ejecutarlo con regularidad debido a que con él podemos evidenciar el tráfico entrante y saliente. [1]

4 Conclusiones

En caso de que se compruebe una intrusión, la información aportada por los comandos seria de bastante ayuda para la investigación y dado el caso contarían como evidencia digital, debido a que podemos obtener datos sobre la trazabilidad de los hechos, conexiones remotas, usuarios con privilegios, ejecución de comandos, entre otros; que nos pueden indicar como contener el Incidente de la manera más rápida y efectiva.

5 References

[1] G. Technologies, *INCIDENT RESPONSE CHEATSHEET WINDOWS AND LINUX*.