CIBERSEGURIDAD

‘Bootcamp IX’

Informe Práctica Módulo Pentesting.

Maximiliano Dariel Altamirano.

Academia KeepCoding.

INDICE:

RESUMEN EJECUTIVO\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Pág. 3

METODOS\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Pág. 3

ALCANCE\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Pág. 4

RATING CRITICIDAD\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Pág. 4

RECOMENDACIONES | MITIGACIONES\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Pág. 4

HALLAZGOS TÉCNICOS\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Pág. 5

RESUMEN EJECUTIVO

Se nos encomendó la tarea de crear un informe con las vulnerabilidades sobre el host Metasploitable, iniciamos las acciones sobre el servidor el 29/01/2025 con un plazo de 4 días. Nuestro propósito fue identificar las vulnerabilidades más críticas tomando un enfoque ofensivo sobre el mismo.

Como consecuencia de las acciones hemos identificado las siguientes vulnerabilidades:

* 5 vulnerabilidades CRITICAS
* 2 vulnerabilidades MODERADAS

Con las evidencias recolectadas concluimos que el servidor es totalmente vulnerable a ataques maliciosos en su infraestructura, afectando a todos los usuarios de la red y la confidencialidad, disponibilidad e integridad de los datos resguardados.

Sugerimos las siguientes estrategias para revertir esta situación crítica:

* Actualizar herramientas, software, aplicativos y parches de seguridad de manera periódica.
* Capacitaciones al equipo de desarrollo con enfoque defensivo, trasladando la importancia de un sistema no vulnerable para agentes maliciosos.
* Implementar equipo de ciberseguridad, desarrollo en el área reporte de incidentes.
* Revisar las políticas de menor privilegios para cada rol junto con las políticas de usuarios y contraseñas.

MÉTODOS

Hemos aplicado las prácticas vigentes de vulnerabilidades, por lo que nos respaldamos en los estándares de penetración y frameworks de las entidades NIST, CVE y pruebas personalizadas garantizando la veracidad de la incidencia y su efectiva mitigación.

Implementamos las siguientes fases para el informe:

* Reconocimiento: recolectamos información de la máquina víctima, IP, puertos y SO.
* Enumeración y análisis de vulnerabilidades: enumeramos la cantidad de puertos abierto, servicios y versiones de cada una e identificamos las posibles vulnerabilidades conocidas.
* Explotación: evidenciamos las vulnerabilidades explotadas junto con las herramientas y métodos utilizados para ese fin. Mencionamos la criticidad del problema para su futura mitigación.
* Documentación: dejamos registro de todos los hallazgos realizados y sugerimos remediaciones para cada uno de las incidencias.

ALCANCE

Realizamos las pruebas de vulnerabilidad sobre “Metasploitable" desde el siguiente Host:

|  |  |
| --- | --- |
| Servidor host | Detalle |
| Metasploitable | 192.168.1.164 |

RATING CRITICIDAD

Definimos el nivel de severidad de cada vulnerabilidad y el rango de puntuación CVSS estandarizado:

|  |  |  |
| --- | --- | --- |
| Severidad | CVSS | Definición |
| Crítico | 9.0 - 10.0 | La explotación es sencilla y, por lo general, resulta en un compromiso a nivel del sistema. Se recomienda formar un plan de acción y parchear de inmediato. |
| Alto | 7.0 - 8.9 | La explotación difícil, pero podría causar privilegios elevados y una pérdida de datos o tiempo de inactividad. Se recomienda formar un plan de acción y parchear con prioridad. |
| Moderado | 4.0 - 6.9 | Las vulnerabilidades existen, pero no son explotables o requieren pasos adicionales, como la ingeniería social. Se recomienda formar un plan de acción y parchear después de que se hayan resuelto los problemas de alta prioridad. |
| Bajo | 0.1 - 3.9 | Las vulnerabilidades no son explotables, pero reducirían la superficie de ataque de una organización. Se recomienda elaborar un plan de acción y un parche durante la próxima ventana de mantenimiento. |

RECOMENDACIONES | MITIGACIONES

En la siguiente tabla evidenciamos los hallazgos por criticidad y la remediación recomendada para subsanar el problema.

|  |  |  |
| --- | --- | --- |
| Hallazgo | Severidad | Remediación |
| Ejecución de comandos de puerta trasera (CVE-2011-2523) | Crítico | Actualización de versión del servicio vsftpd. |
| Java RMI - Server Insecure Default Configuration Java Code Execution (CVE-2011-3556) | Crítico | Actualizaciones de seguridad, parches de seguridad. |
| Puerto: 1524/tcp; Servicio: bindshell | Critico | Eliminar el usuario root, desactivar el puerto expuesto. |
| PostgreSQL 8.2/8.3/8.4: UDF para ejecución de comandos | Critico | Actualización de versión del servicio PostgreSQL. Limitar permisos de usuarios. |
| Puerto 23: TELNET | Critico | Discontinuar el uso del servicio, reemplazar por protocolos de comunicaciones seguras |
| Samba 3.0.20 < 3.0.25rc3 (CVE-2007-2447) | Moderado | Actualización de versión del servicio Samba. |
| VNC Authentication Scanner (CVE-2001-0167) | Moderado | Actualización de versión del servicio VNC, parches de seguridad. |

HALLAZGOS TÉCNICOS

* **Ejecución de comandos de puerta trasera (CVE-2011-2523)**

**Puerto: 21/tcp; Servicio: ftp; Versión: vsftpd 2.3.4**

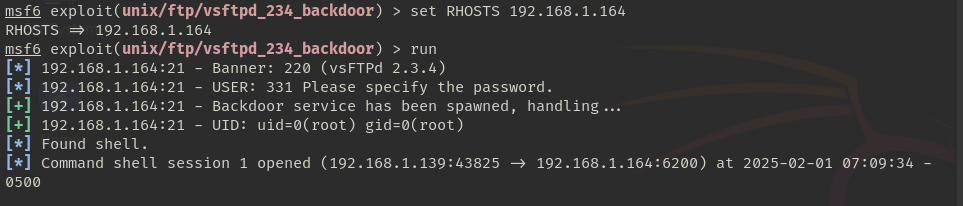
Descripción:

Este módulo explota una puerta trasera maliciosa que se agregó a la descarga de VSFTPD. Esta puerta trasera se introdujo en el archivo vsftpd-2.3.4.tar.gz entre el 30 de junio de 2011 y 1 de julio de 2011 según la información más reciente disponible. Esta puerta trasera fue eliminada el 3 de julio de 2011.

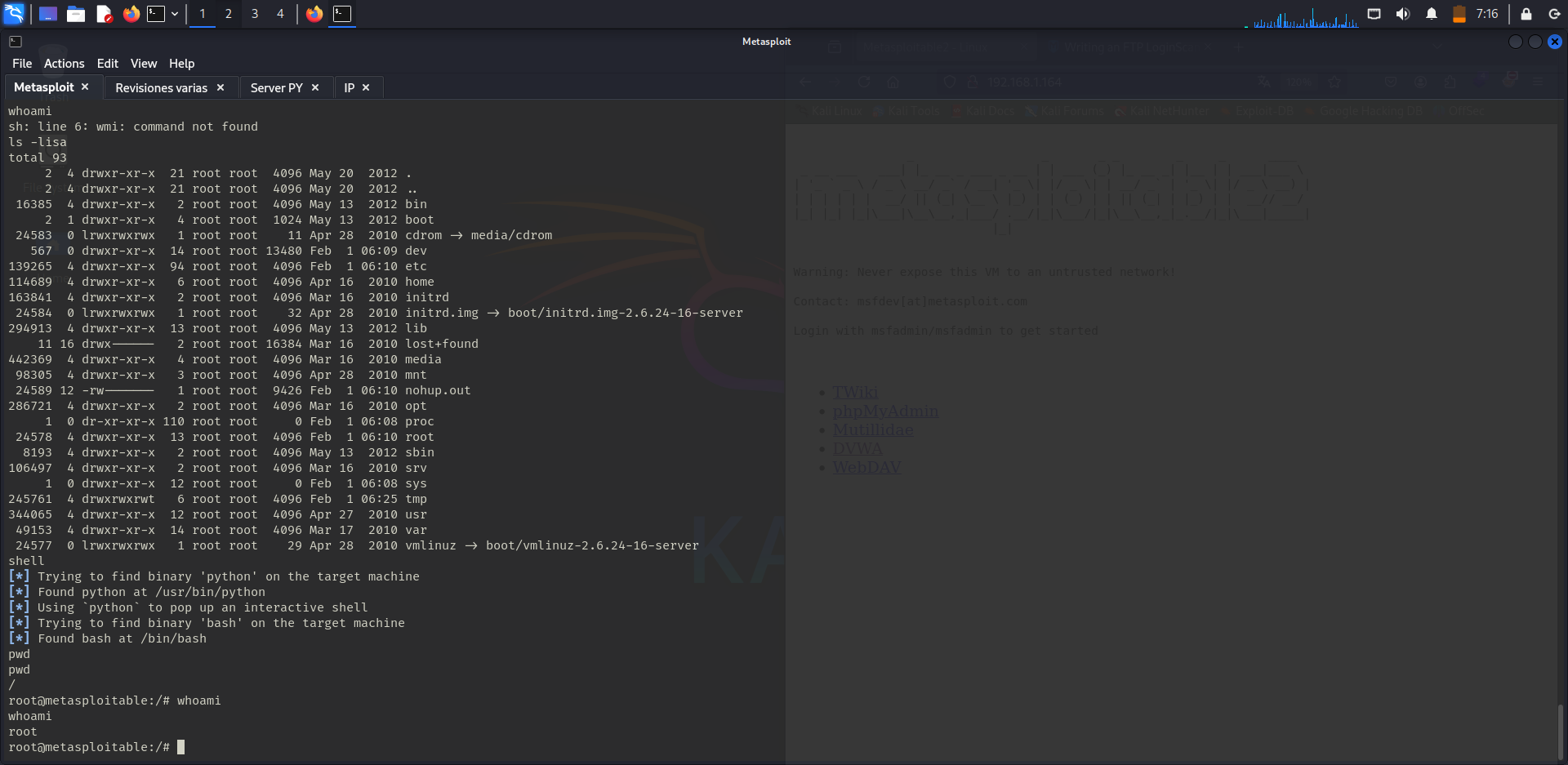
Iniciamos el módulo desde Metasploit, colocamos los parámetros solicitados de RHOSTS de nuestra máquina víctima logrando que esta máquina se conecte con nuestro LHOSTS.







Ya con el acceso, identificamos los privilegios de usuario y la información diponible de la máquina víctima.



Mitigación:

Actualizar a la versión más reciente de vsftpd con la corrección de la vulnerabilidad. Discontinuar el uso de versiones comprometidas con el problema.

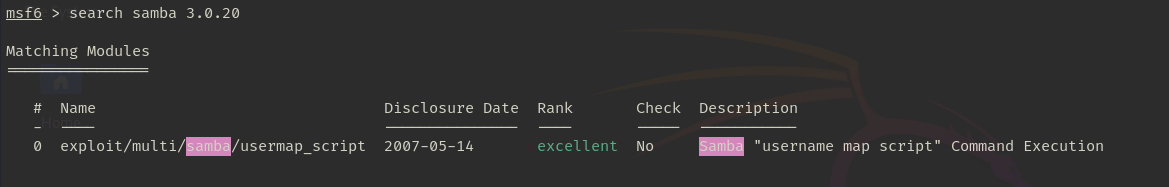
Herramientas utilizadas:

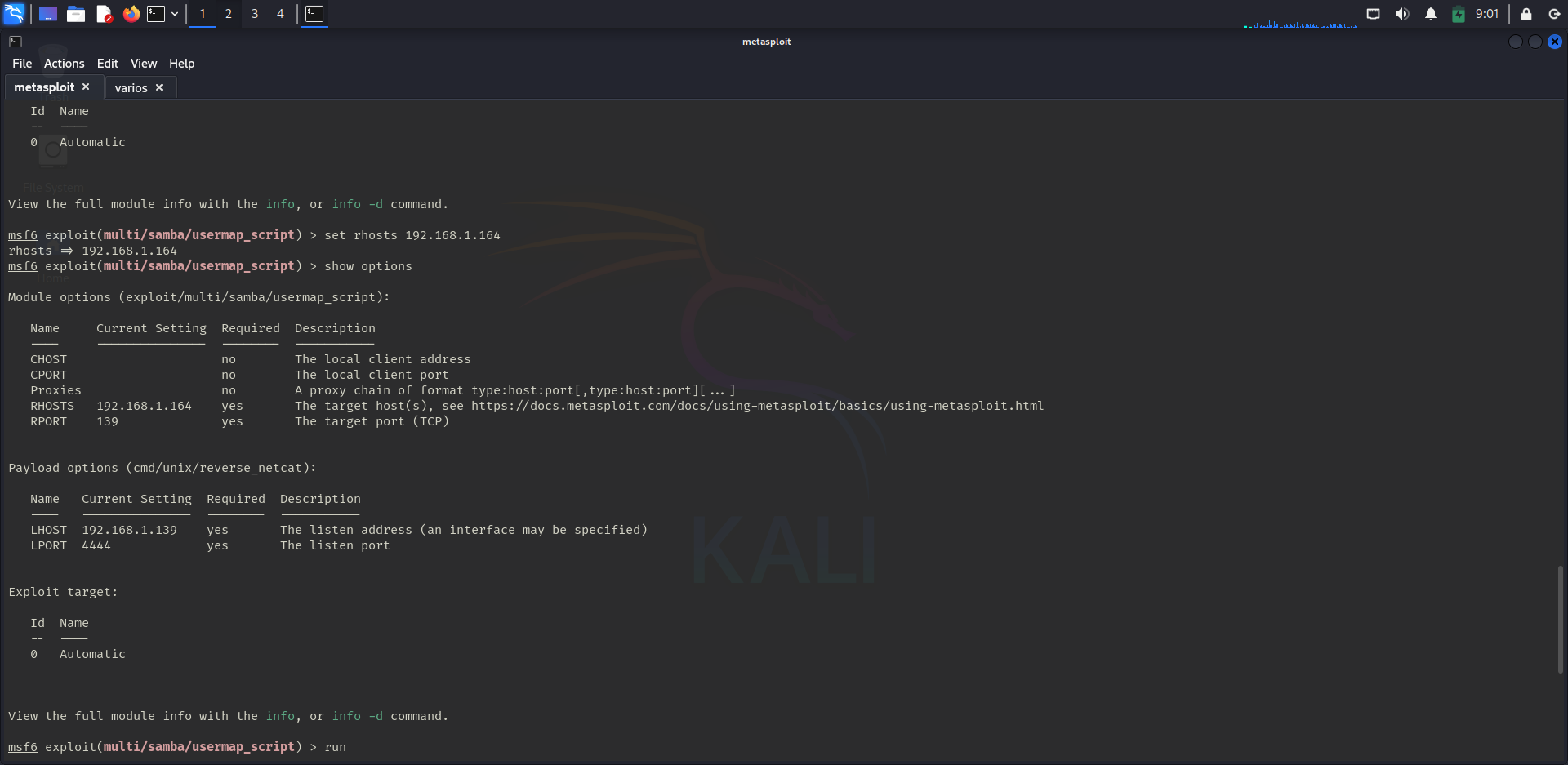
* NMAP
* Metasploit
* <https://www.exploit-db.com>
* **Samba 3.0.20 < 3.0.25rc3 - Ejecución del comando “Username” en el script de mapa (CVE-2007-2447)**

**Puerto: 445/tcp; Servicio: netbios-ssn, Versión: Samba smbd 3.0.20-Debian**

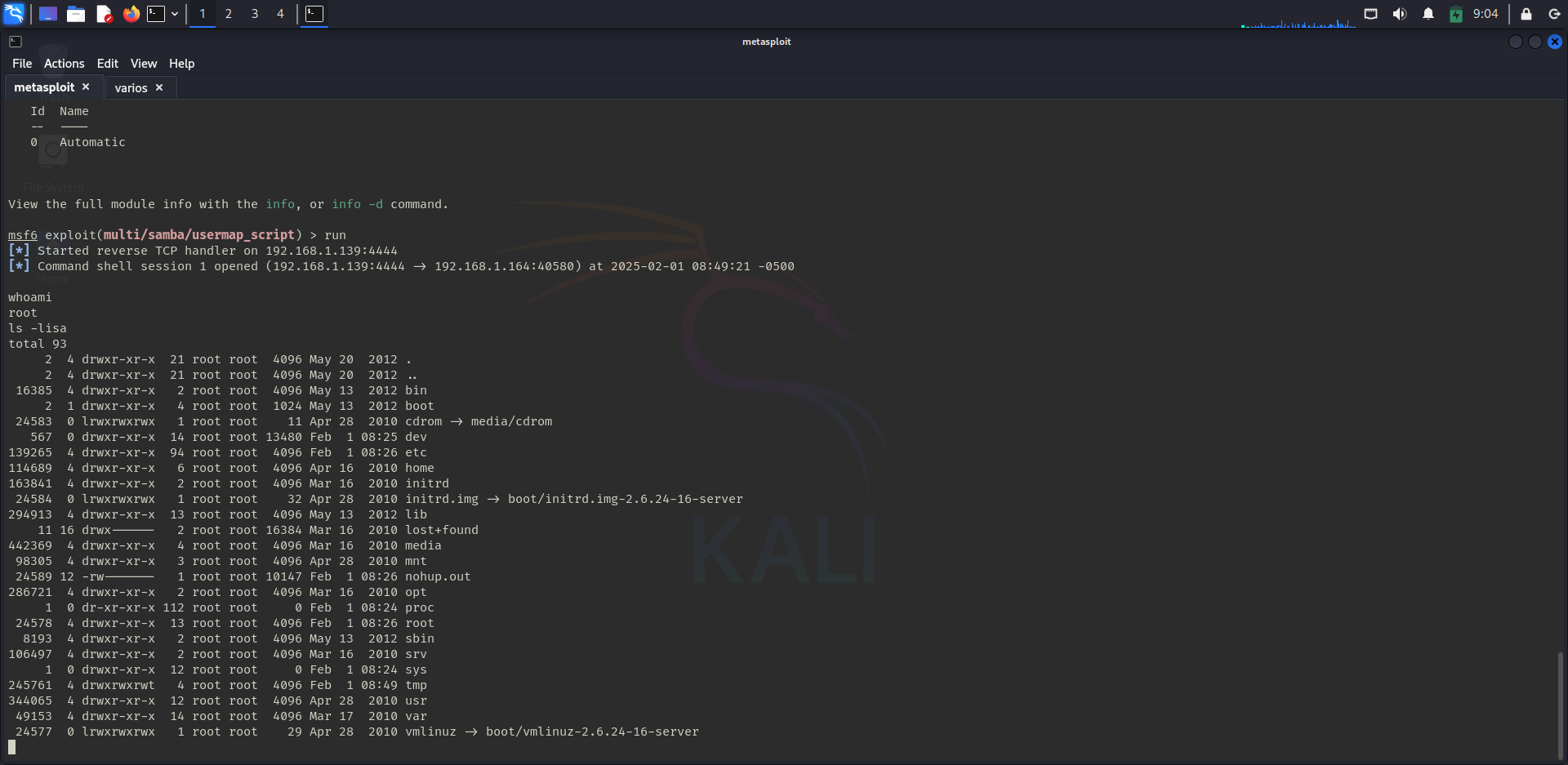
Descripción:

A partir de esta vulnerabilidad, el atacante puede ejecutar comando de forma arbitraria desde la Shell con acceso sin credenciales conocidas y con privilegio root. Desde Metasploit identificamos el módulo, colocamos el parámetro requerido de RHOSTS y lanzamos el exploit.





Evidenciamos el acceso con privilegios y la información de la máquina víctima.



Mitigación:

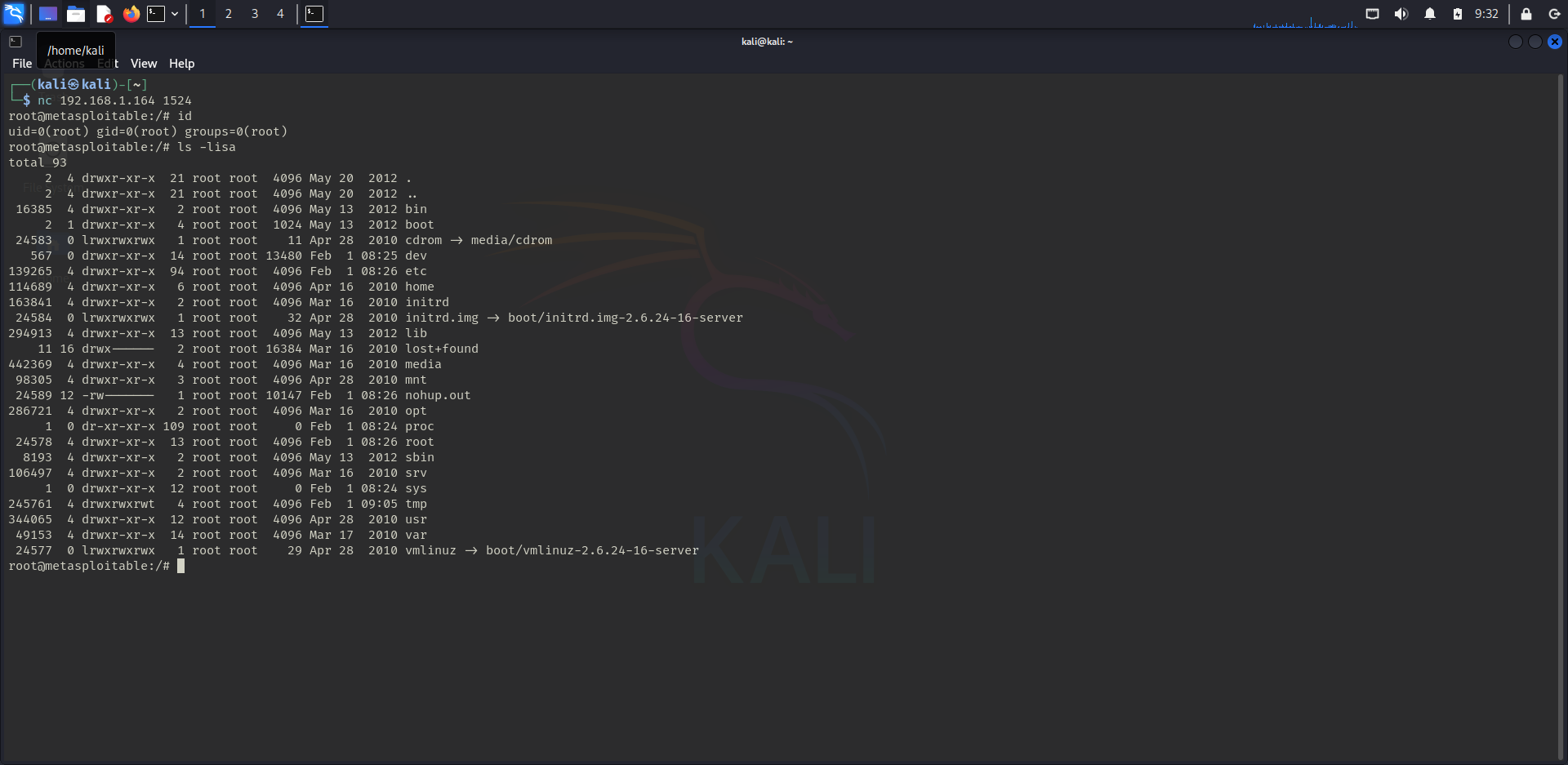
Actualizar a la versión más reciente de Samba con la corrección de la vulnerabilidad.

Herramientas utilizadas:

* NMAP
* Metasploit
* <https://www.exploit-db.com>
* **Puerto: 1524/tcp; Servicio: bindshell; Versión: Metasploitable root Shell**

Descripción:

Identificamos el puerto 1524 abierto con servicio de Shell vinculado y privilegios root. Dejamos en evidencia la vulnerabilidad utilizando la herramienta Netcat configurando los parámetros IP y Puerto correspondientes (*nc 192.168.1.164* *1524*)



Mitigación:

Eliminar el usuario root y configurar correctamente los puertos (desactivar puerto 1524)

Herramientas utilizadas:

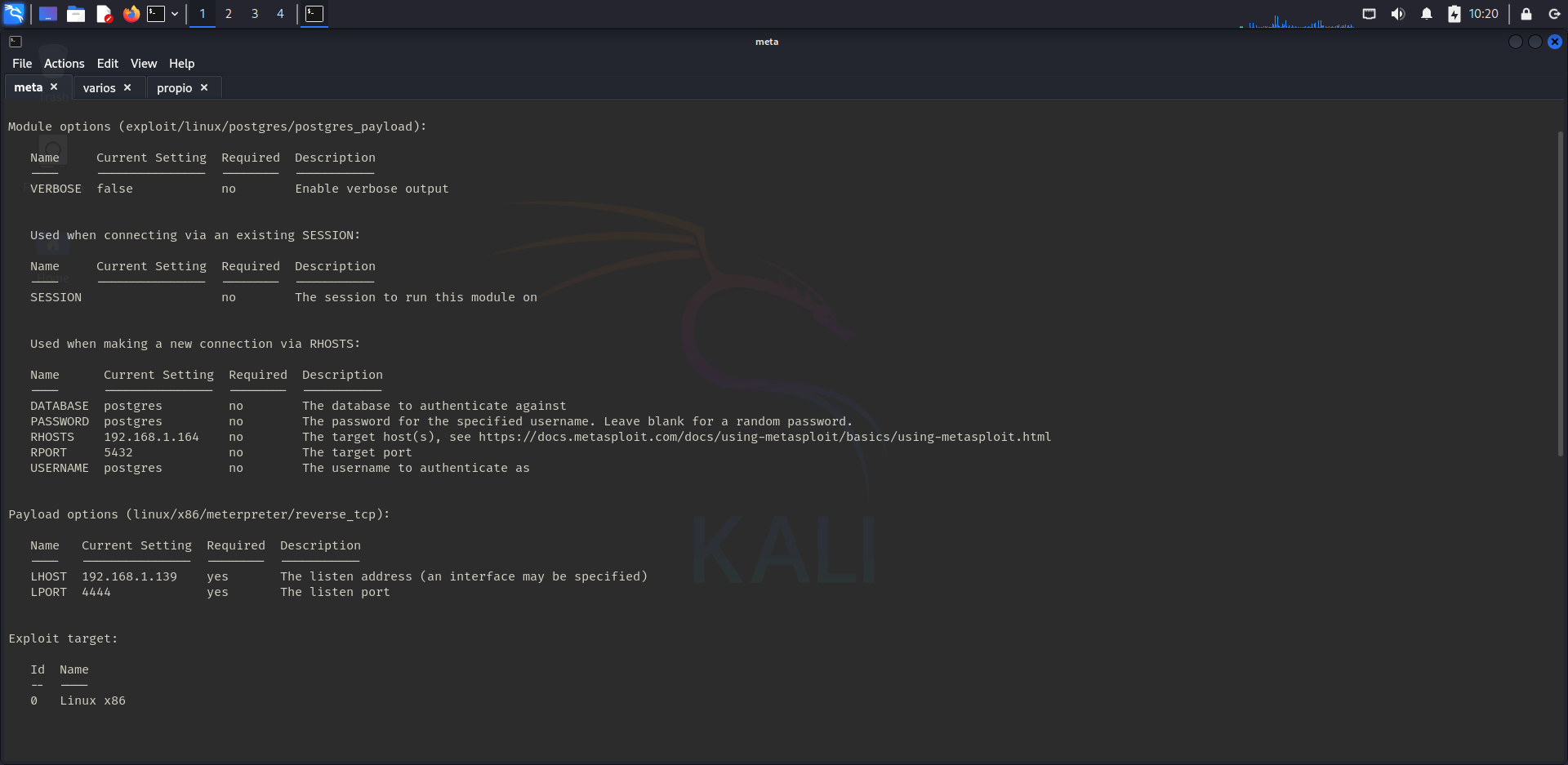
* NMAP
* NetCat
* **PostgreSQL 8.2/8.3/8.4: UDF para ejecución de comandos (CVE: sin datos)**

**Puerto: 5432/tcp; Servicio: postgresql; Versión:PostgreSQL DB 8.3.0 - 8.3.7**

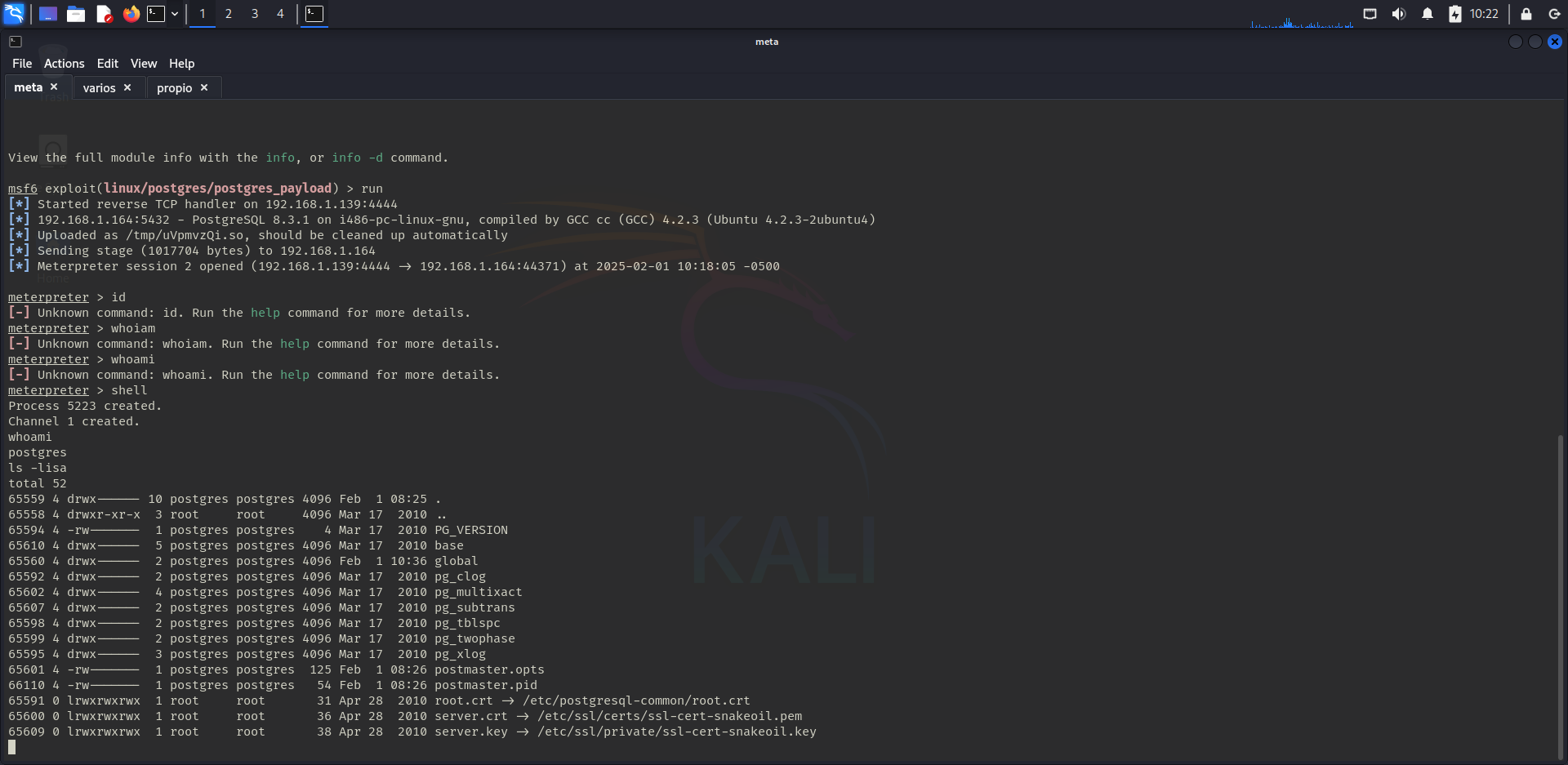
Descripción:

En algunas instalaciones Linux predeterminadas de PostgreSQL, la cuenta de servicio postgres puede escribir en el directorio /tmp y también puede obtener bibliotecas compartidas UDF desde allí, lo que permite la ejecución de código arbitrario. Este módulo compila un archivo de objeto compartido de Linux, lo carga en el host de destino mediante el método UPDATE pg\_largeobject de inyección binaria y crea una UDF (función definida por el usuario) a partir de ese objeto compartido. Debido a que la carga útil se ejecuta como el constructor del objeto compartido, no necesita cumplir con versiones específicas de la API de Postgres.

A partir de la información obtenida, utilizaremos el módulo exploit/linux/postgres/postgres\_payload en metasploit. Realizamos las configuraciones de LHOSTS y RHOSTS correspondientes y lanzamos el exploits.



Evidenciamos el acceso y la información correspondiente a la máquina víctima.



Mitigación:

Actualizar a la versión más reciente de PostgreSQL, configurar de forma estricta los permisos a los directorios (Principio de mínimo privilegio) y de herramientas de monitoreo (firewalls)

Herramientas utilizadas:

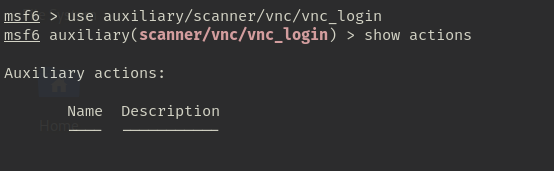
* NMAP
* Metasploit
* <https://www.rapid7.com/db/modules/exploit/linux/postgres/postgres_payload/>
* **VNC Authentication Scanner (CVE-2001-0167)**

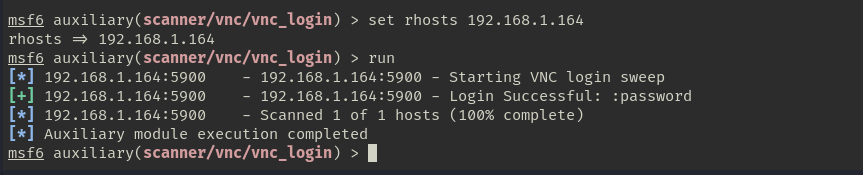
**Puerto: 5900/tcp; Servicio: vnc; Versión: VNC (protocol 3.3)**

Descripción:

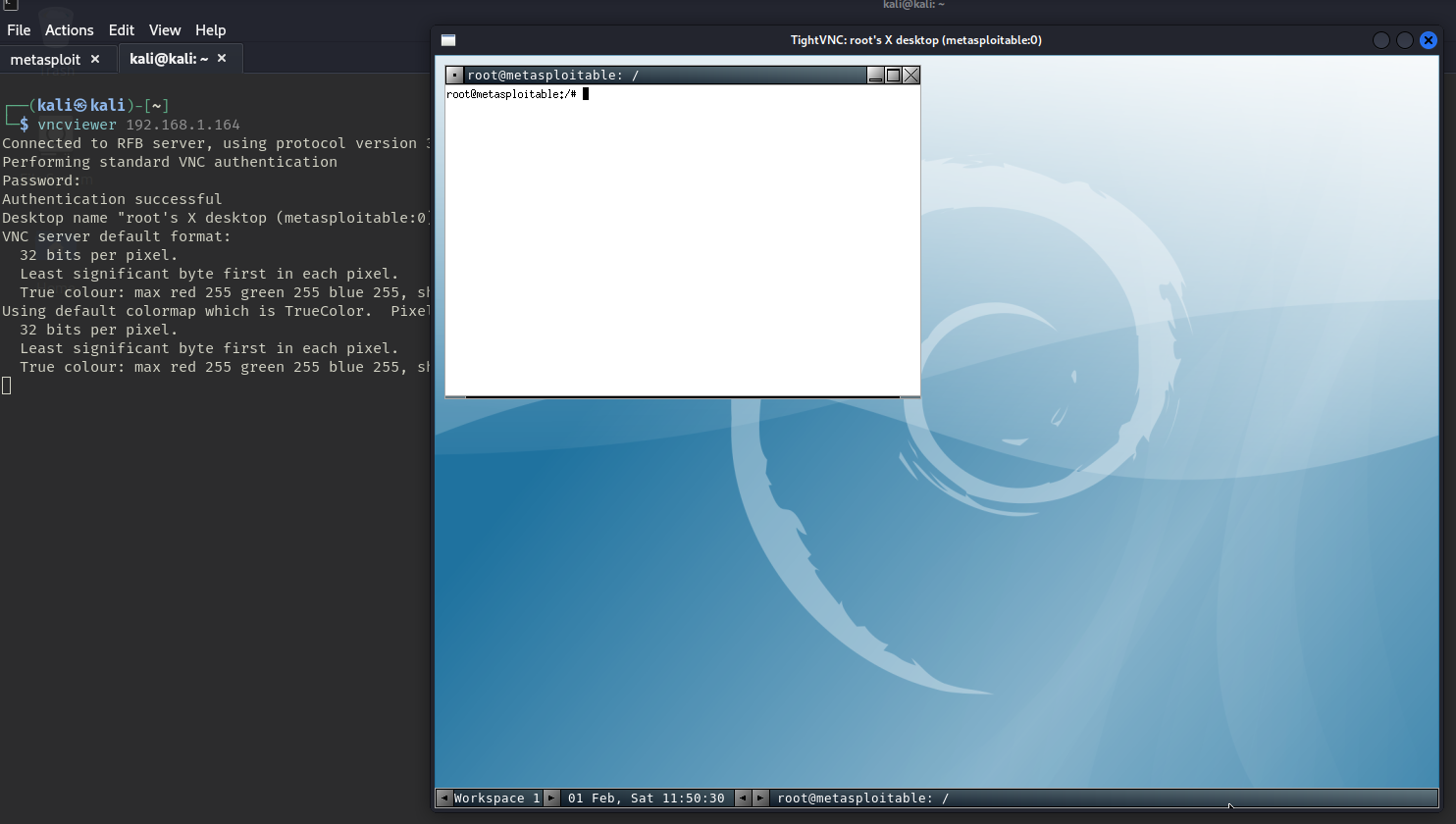
Este módulo probará un servidor VNC en una variedad de máquinas e informará los inicios de sesión exitosos. Actualmente, es compatible con las versiones 3.3, 3.7, 3.8 y 4.001 del protocolo RFB mediante el método de autenticación de desafío-respuesta de VNC.

Accedemos al módulo de Metasploit, colocamos las configuraciones correspondientes al payload y obtenemos las credenciales necesarias para el acceso VNC luego de lanzar el exploit.





Logramos el acceso con el comando *vncviewer* desde la terminal de la máquina atacante y evidenciamos el ingreso con privilegios root a la máquina victima.



Mitigación:

Actualizar a la versión más reciente de VNC, parches de seguridad, configuraciones de herramientas de monitoreos (firewalls), desactivar el protocolo si no es realmente necesario su uso.

Herramientas utilizadas:

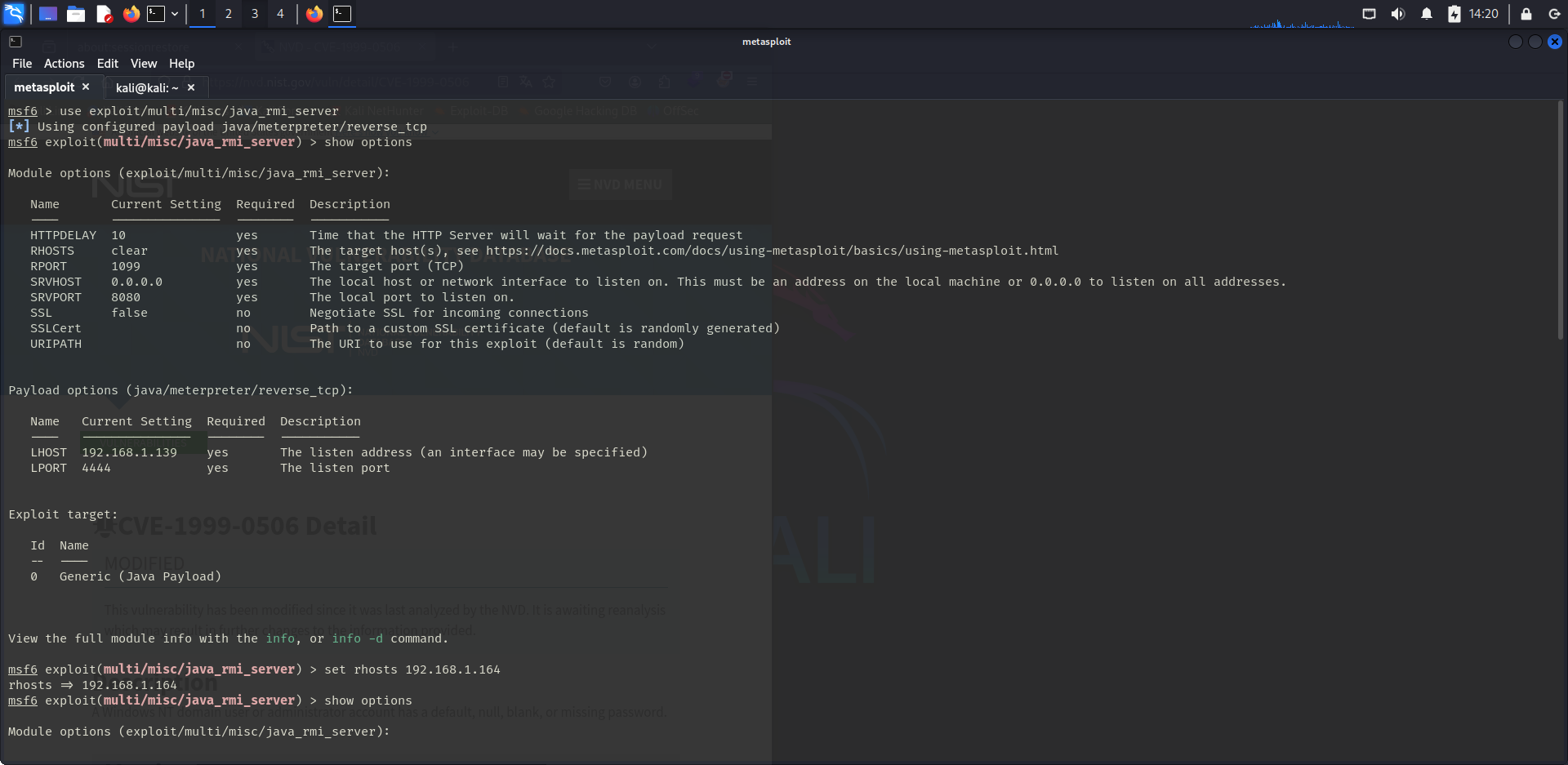
* NMAP
* Metasploit
* <https://www.rapid7.com/db/modules/auxiliary/scanner/vnc/vnc_login/>
* **Java RMI - Server Insecure Default Configuration Java Code Execution (CVE-2011-3556)**

**Puerto: 1099/tcp; Servicio: java-rmi; Versión: GNU Classpath grmiregistry**

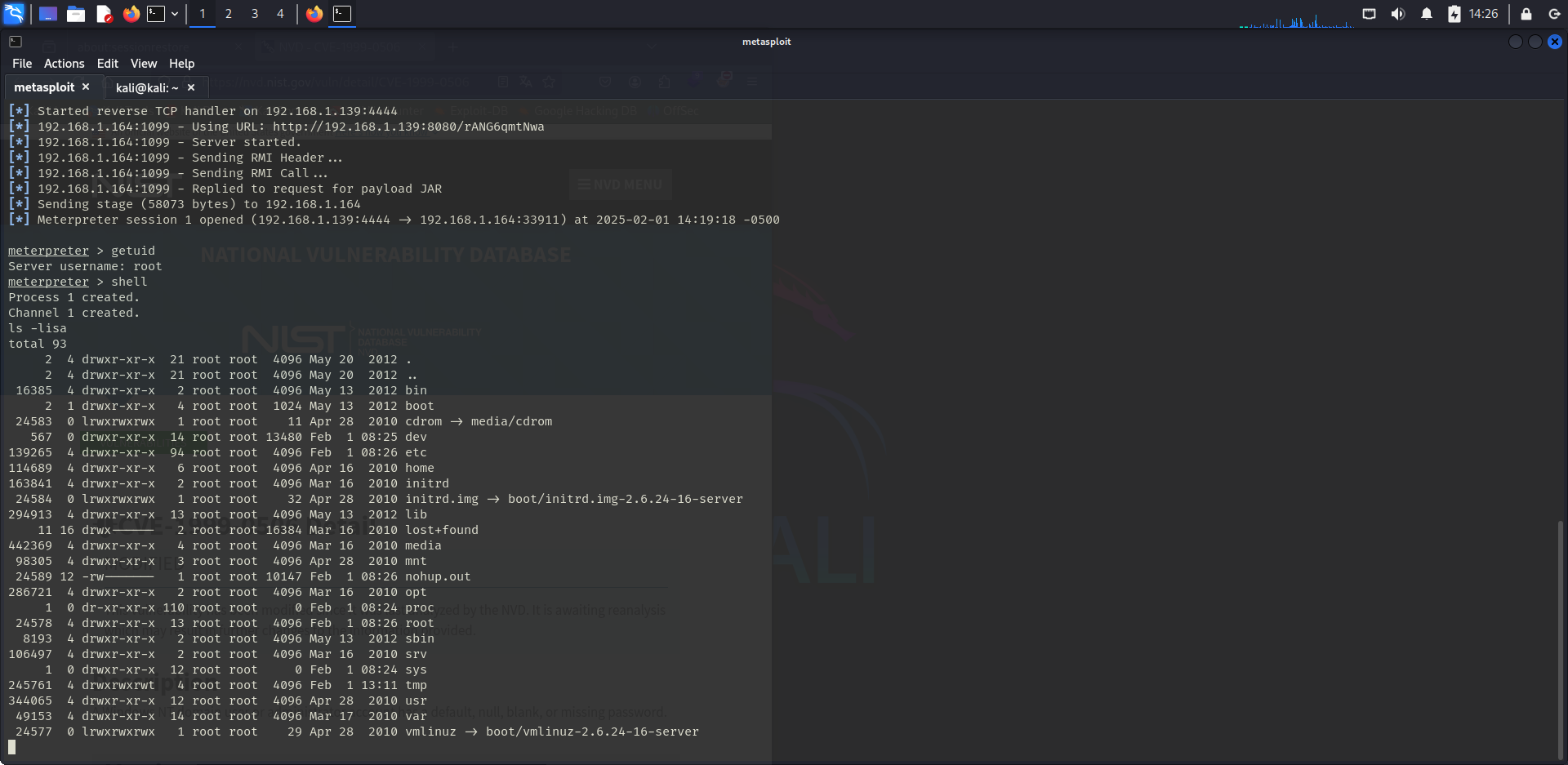
Descripción:

Este módulo aprovecha la configuración predeterminada de los servicios RMI Registry y RMI Activation, que permiten cargar clases desde cualquier URL remota (HTTP). Como invoca un método en el recolector de basura distribuido RMI que está disponible a través de cada punto final RMI, se puede utilizar tanto contra rmiregistry como contra rmid, y también contra la mayoría de los demás puntos finales RMI (personalizados). Tenga en cuenta que no funciona contra los puertos Java Management Extension (JMX), ya que estos no admiten la carga de clases remotas, a menos que otro punto final RMI esté activo en el mismo proceso Java. Las llamadas a métodos RMI no admiten ni requieren ningún tipo de autenticación.

Configuramos los parámetros RHOSTS del módulo Metasploit y lanzamos el exploit.



Evidenciamos la vulnerabilidad identificando el usuario root en el meterpreter y enlistando el contenido de la máquina víctima.



Mitigación:

Actualizar Java a la versión más reciente sin la vulnerabilidad mencionada, aplicar parche de seguridad, implementación de herramientas de monitoreos (firewalls).

Herramientas utilizadas:

* NMAP
* Metasploit
* <https://www.rapid7.com/db/modules/exploit/multi/misc/java_rmi_server/>
* <https://www.exploit-db.com/exploits/17535>
* **Puerto 23: TELNET**

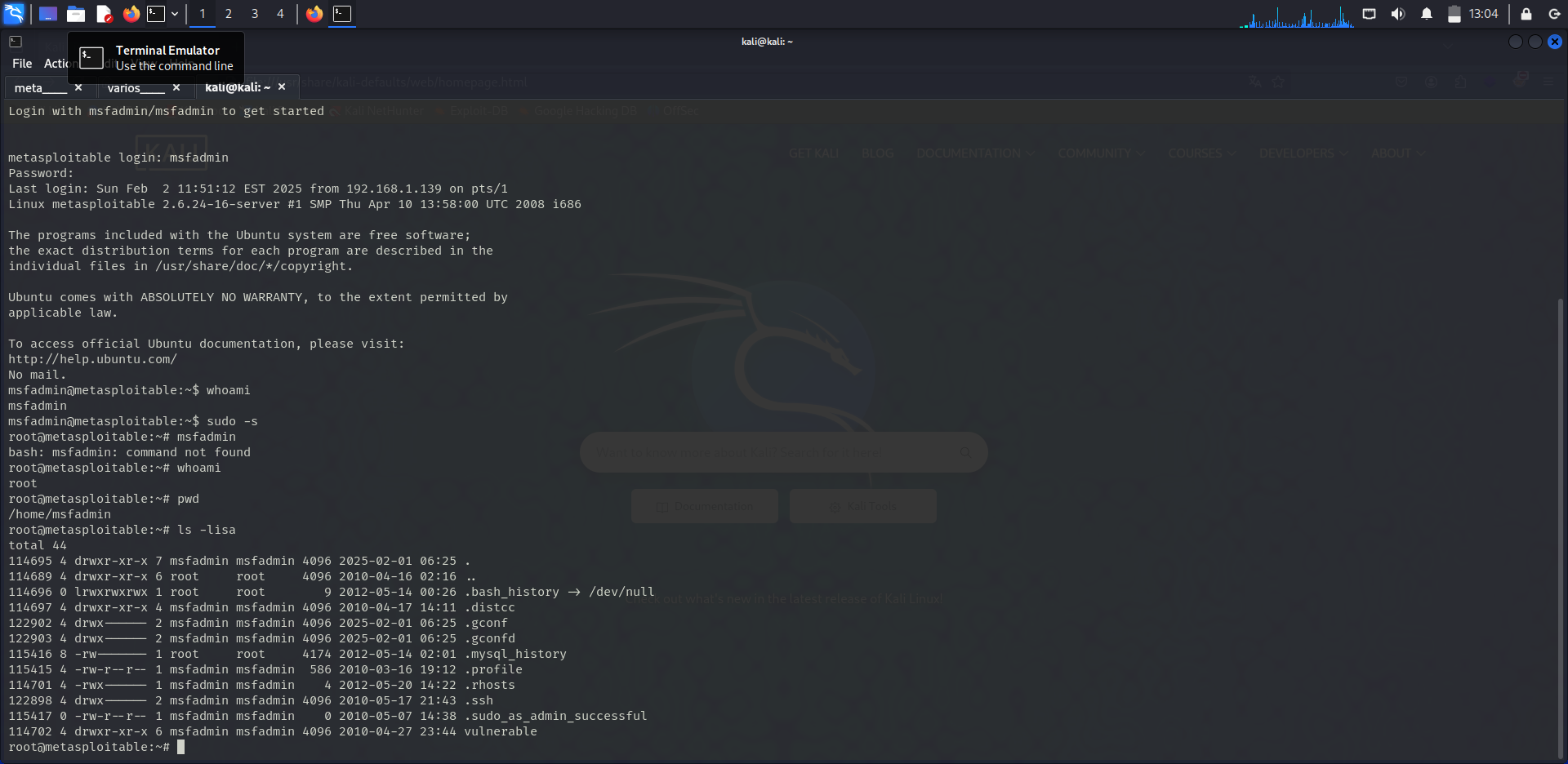
**Puerto: 23/tcp; Servicio: telnet; Versión: Linux telnetd**

Descripción:

Identificamos el puerto 23 abierto con el protocolo telnet. Este permite la comunicación de datos en texto plano sin cifrar. Utilizaremos el comando *telnet + IP* en el terminal de nuestra maquina atacante para acceder a la información en claro.

Ya dentro de la Shell identificamos el usuario actual e intentamos forzar la escalada de privilegios de forma exitosa con el comando *sudo -s* reciclando la contraseña conocida “msfadmin”.





Mitigación:

No utilizar el protocolo si no es realmente necesario, desactivar el puerto. Reemplazar el protocolo por uno con datos cifrados y conexiones seguras como SSH. implementación de herramientas de monitoreos (firewalls).

Herramientas utilizadas:

* NMAP
* Kali