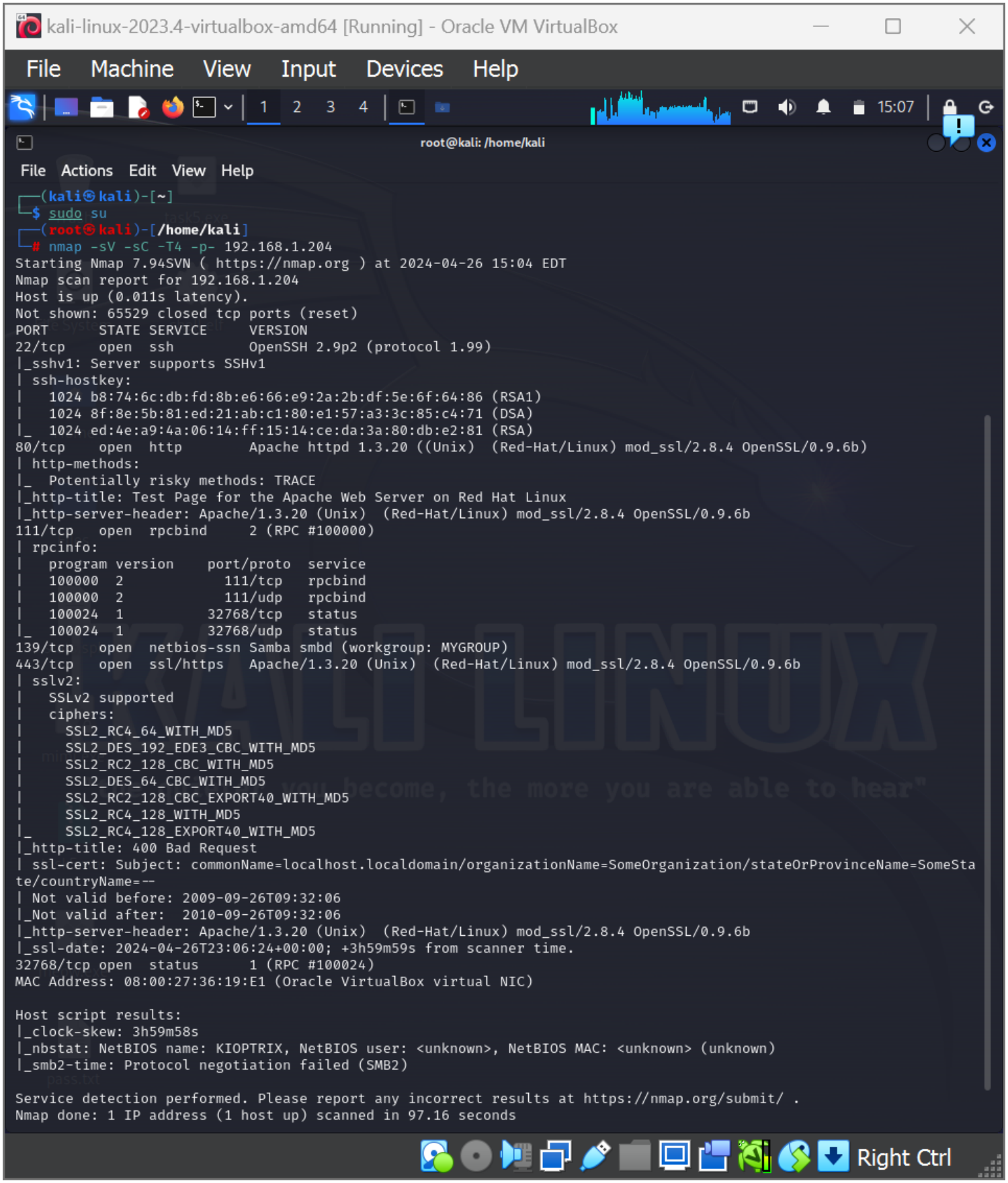
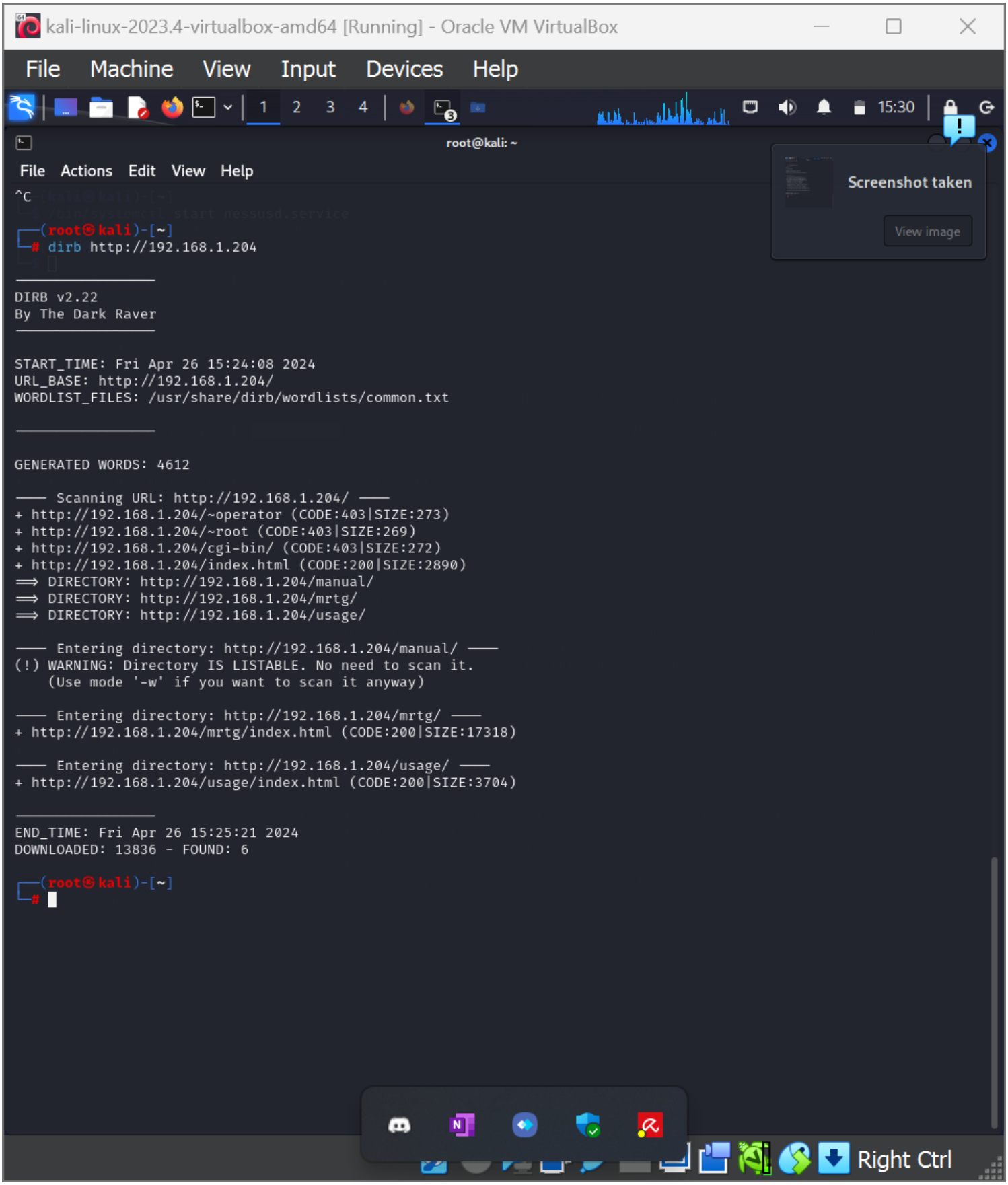
Scanning



I conducted an nmap scan on the target machine and identified several open ports, including 22, 80, 111, 139, and 443 among others.

Upon attempting to access port 80 via my browser, I encountered only the default Apache webpage, yielding no additional information. My next step was to use drib!

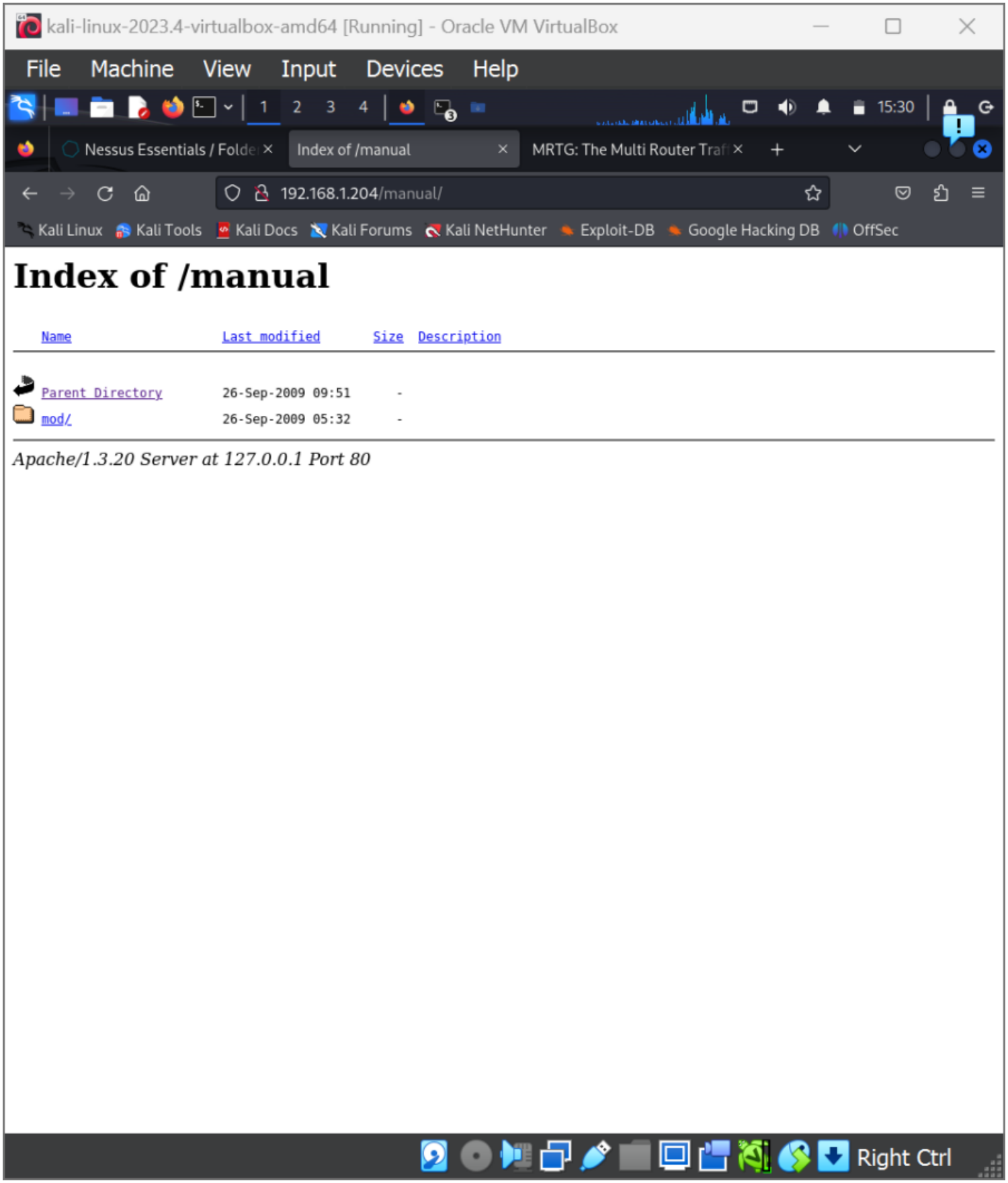
Enumeration



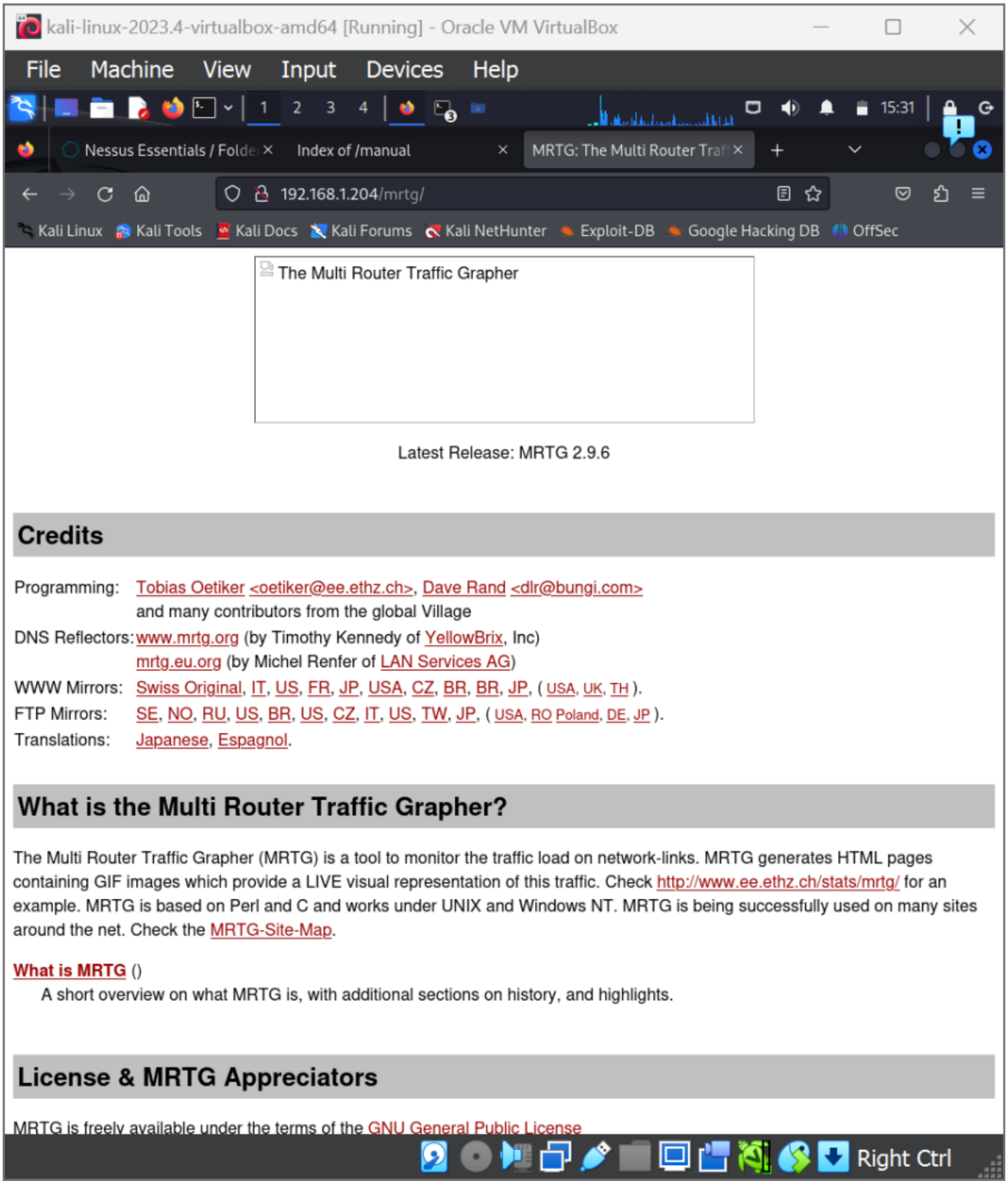
By discovering hidden directories and files. From the dirb scan results, it appears that there are hidden sites on 192.168.1.204:
/manual/

/mrtg/
/usage/
But they all don't hold useful information.

File 1 👇



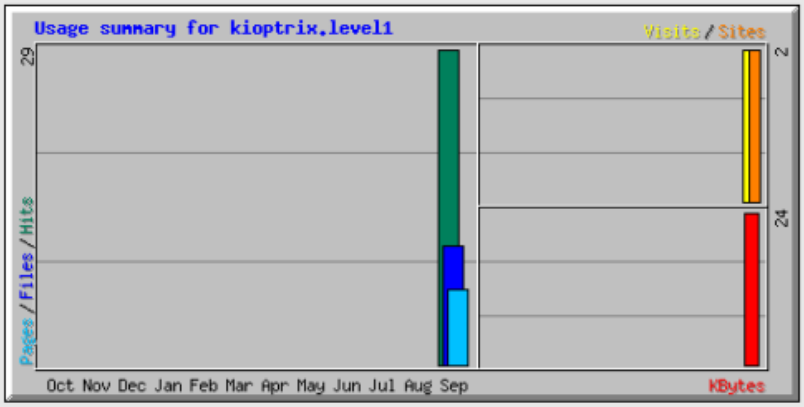
File 2 👇


kali-linux-2023.4-virtualbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
15:31
Nessus Essentials / Folde
Index of /manual
MRTG: The Multi Router Traf
192.168.1.204/mrtg/
Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec
The Multi Router Traffic Grapher
Latest Release: MRTG 2.9.6
Credits
Programming: Tobias Oetiker <oetiker@ee.ethz.ch>, Dave Rand <dlr@bungi.com>
and many contributors from the global Village
DNS Reflectors: www.mrtg.org (by Timothy Kennedy of YellowBrix, Inc)
mrtg.eu.org (by Michel Renfer of LAN Services AG)
WWW Mirrors: Swiss Original, IT, US, FR, JP, USA, CZ, BR, BR, JP, (USA, UK, TH).
FTP Mirrors: SE, NO, RU, US, BR, US, CZ, IT, US, TW, JP, (USA, RO Poland, DE, JP).
Translations: Japanese, Espagnol.
What is the Multi Router Traffic Grapher?
The Multi Router Traffic Grapher (MRTG) is a tool to monitor the traffic load on network-links. MRTG generates HTML pages containing GIF images which provide a LIVE visual representation of this traffic. Check http://www.ee.ethz.ch/stats/mrtg/ for an example. MRTG is based on Perl and C and works under UNIX and Windows NT. MRTG is being successfully used on many sites around the net. Check the MRTG-Site-Map.
What is MRTG ()
A short overview on what MRTG is, with additional sections on history, and highlights.
License & MRTG Appreciators
MRTG is freely available under the terms of the GNU General Public License
Right Ctrl

File 3 👇

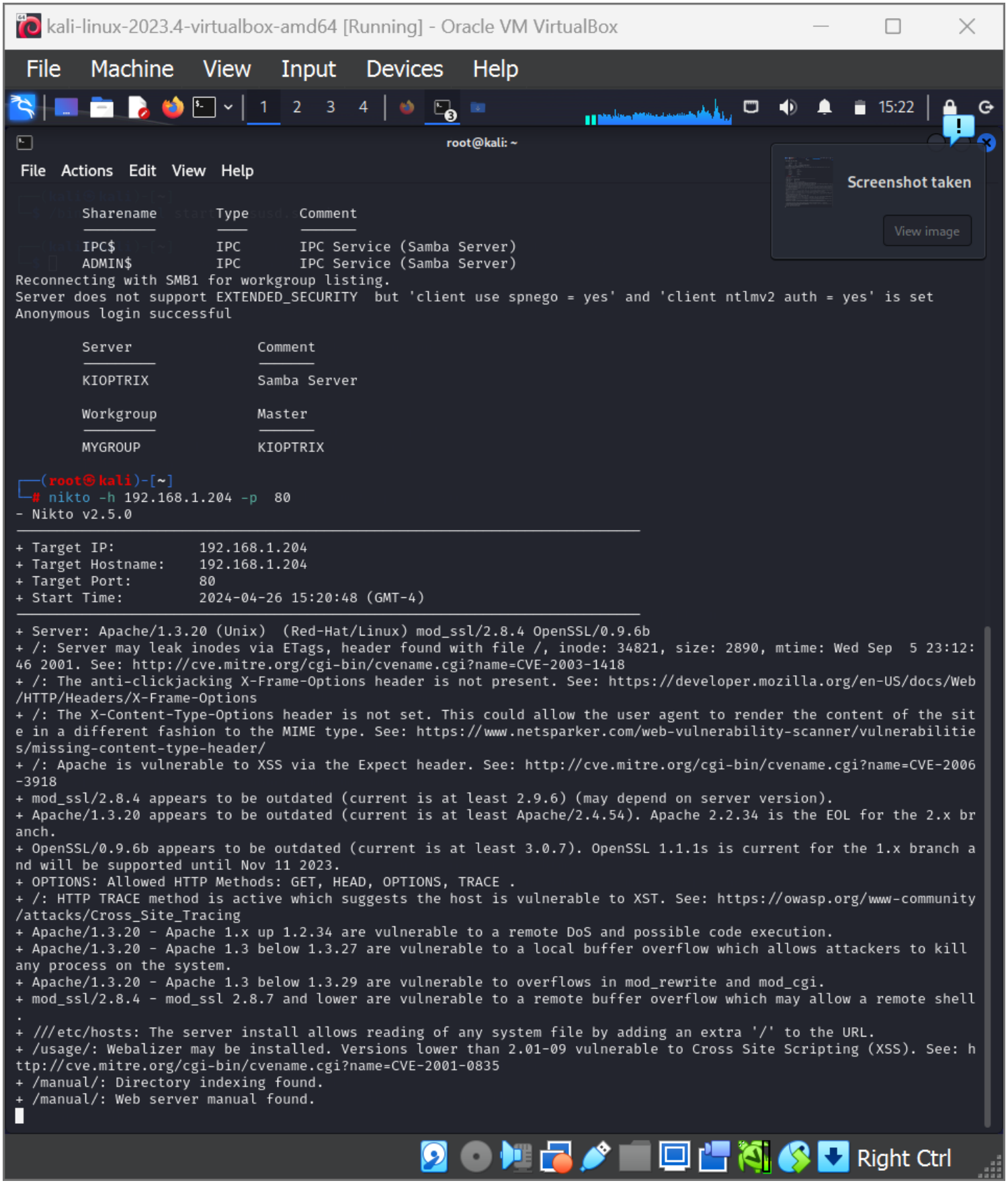
Usage Statistics for kioptrix.level1

Summary Period: Last 12 Months
Generated 27-Feb-2024 10:24 EST



Summary by Month										
Month	Daily Avg				Monthly Totals					
	Hits	Files	Pages	Visits	Sites	KBytes	Visits	Pages	Files	Hits
Sep 2009	29	11	7	2	2	24	2	7	11	29
Totals						**24**	**2**	**7**	**11**	**29**

Generated by Webalizer Version 2.01



Nikto Scan Findings:

Mod_SSL Vulnerability: Nikto identified a potential vulnerability in Mod_SSL, indicating a risk of remote buffer overflow.

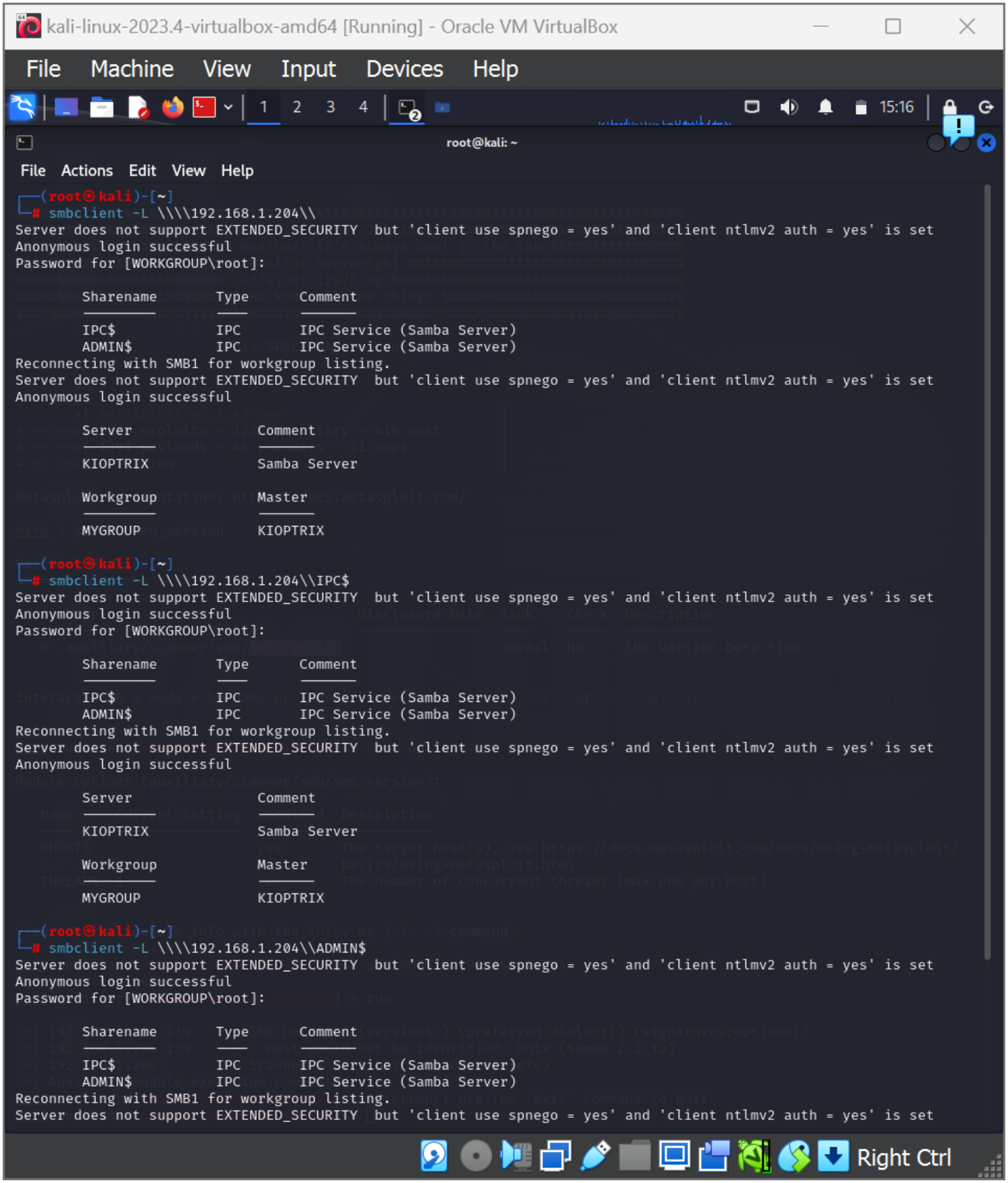
Apache Vulnerabilities (Overflows): The scan revealed vulnerabilities in

Apache, suggesting susceptibility to buffer overflows.

Denial of Service (DoS): Nikto flagged potential weaknesses that could lead to denial of service attacks, rendering the server unresponsive.

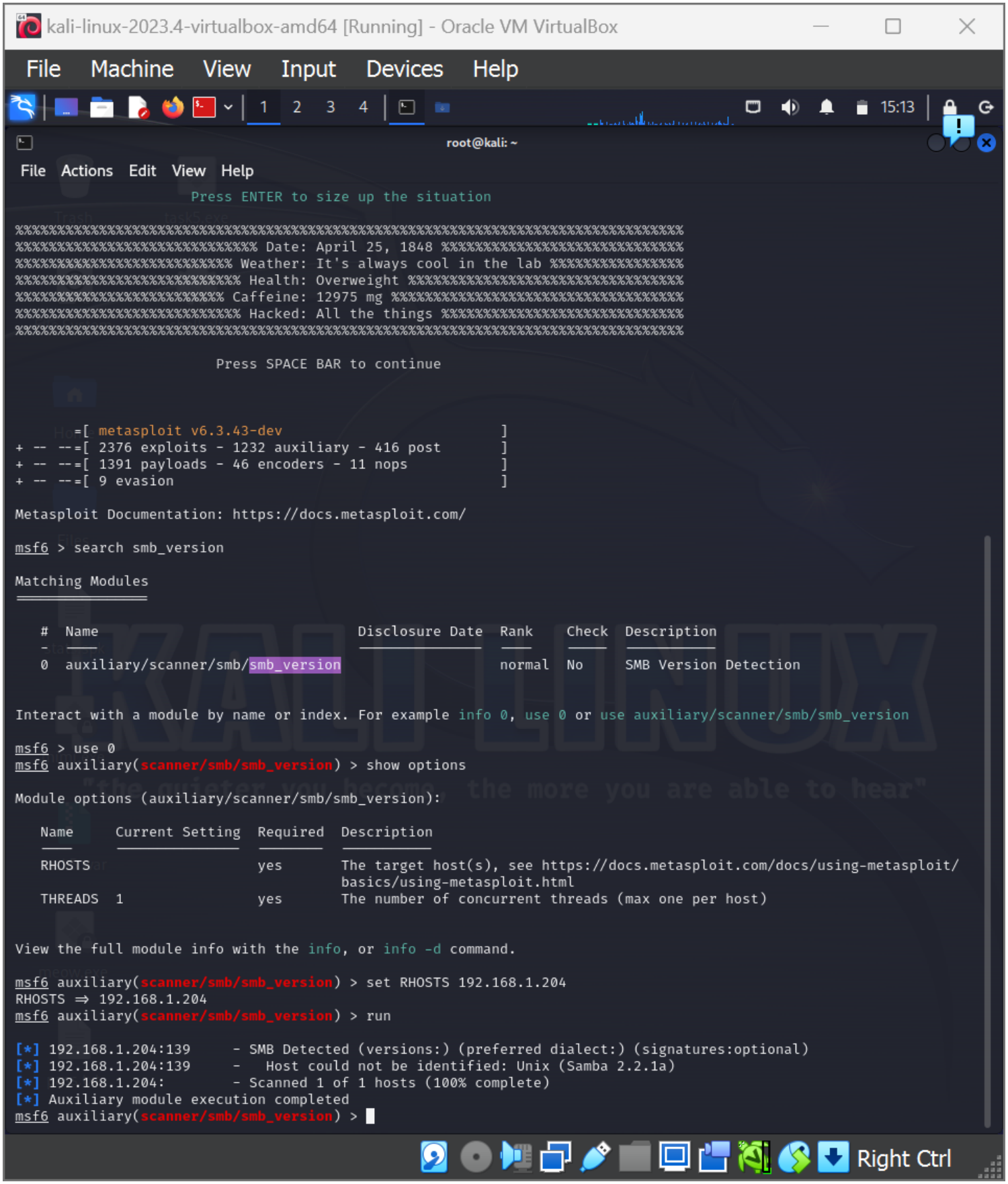
Code Execution: Identified the possibility of code execution vulnerabilities on the server, posing a significant security risk.

Exploitation



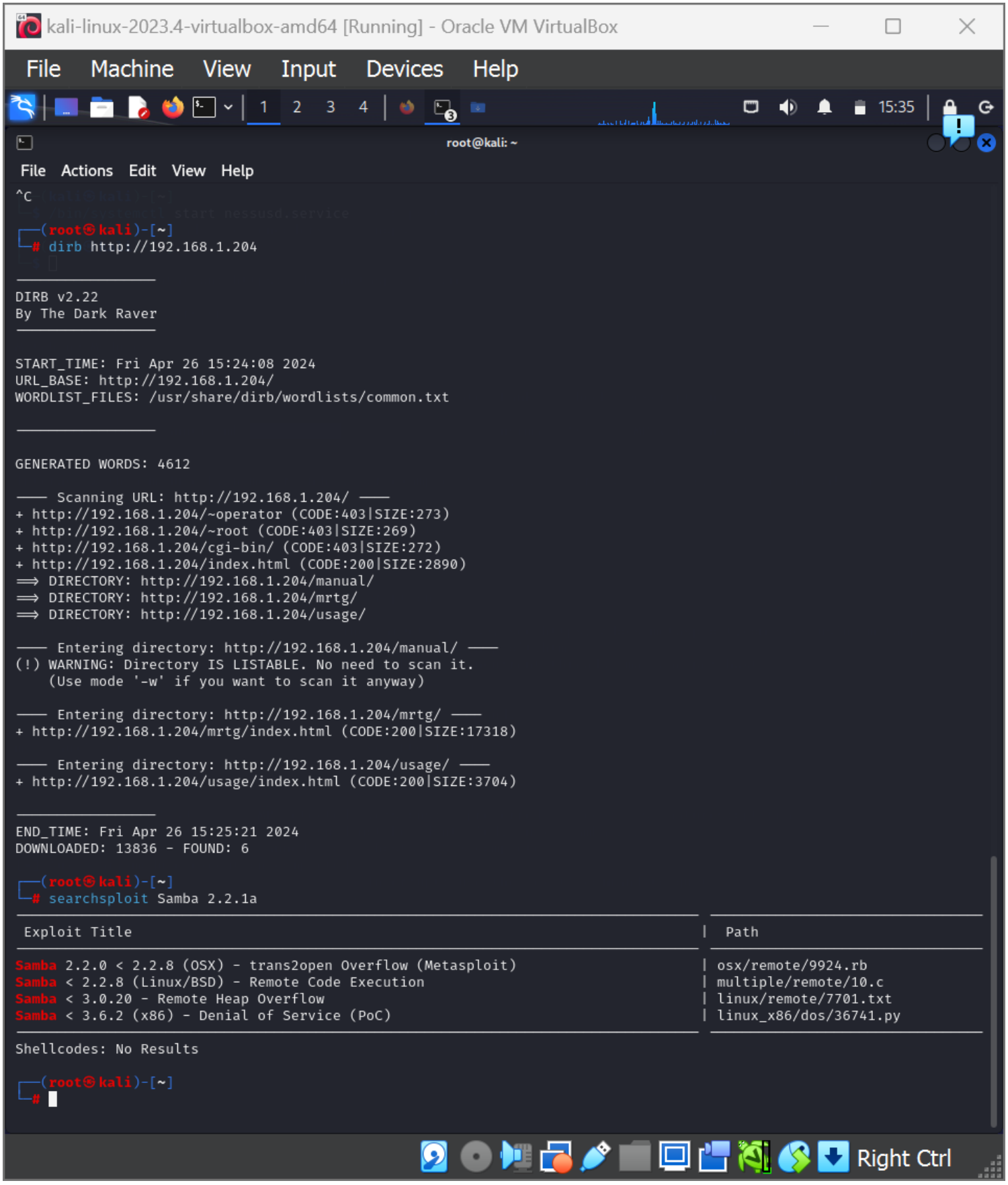
ⓘ smbclient is used to list the available shares on an SMB (Server Message Block)

I successfully listed the available shares on the SMB server at the specified IP address.



Using Metasploit to Determine SMB Version:

I accessed the Metasploit console and utilized the SMB version detection exploit module to ascertain the version of SMB (Server Message Block) protocol implemented on the target machine.
The smb version is 2.2.1a



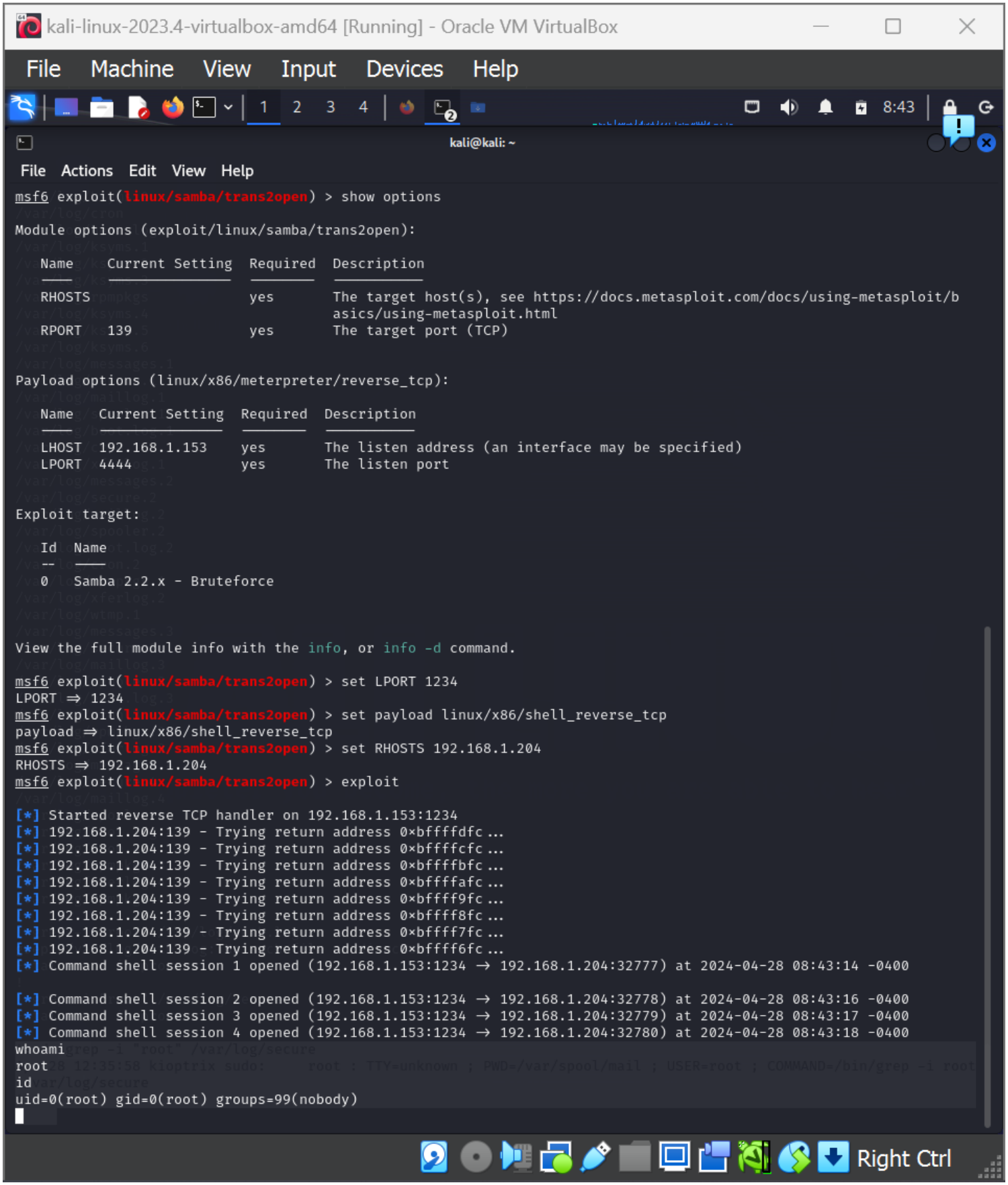
I employed the searchsploit command and integrated the version of SMB obtained through SMB version exploitation.

• The output you see consists of titles of various exploits available in the Exploit-DB.

• Each title corresponds to a specific security issue or vulnerability that has been documented and categorized.

This search led to the identification of a relevant exploit, specifically the "metasploit trans2pen overflow," indicating a potential vulnerability within the SMB protocol.

Privilege Escalation



Exploiting SMB Vulnerability with Metasploit:

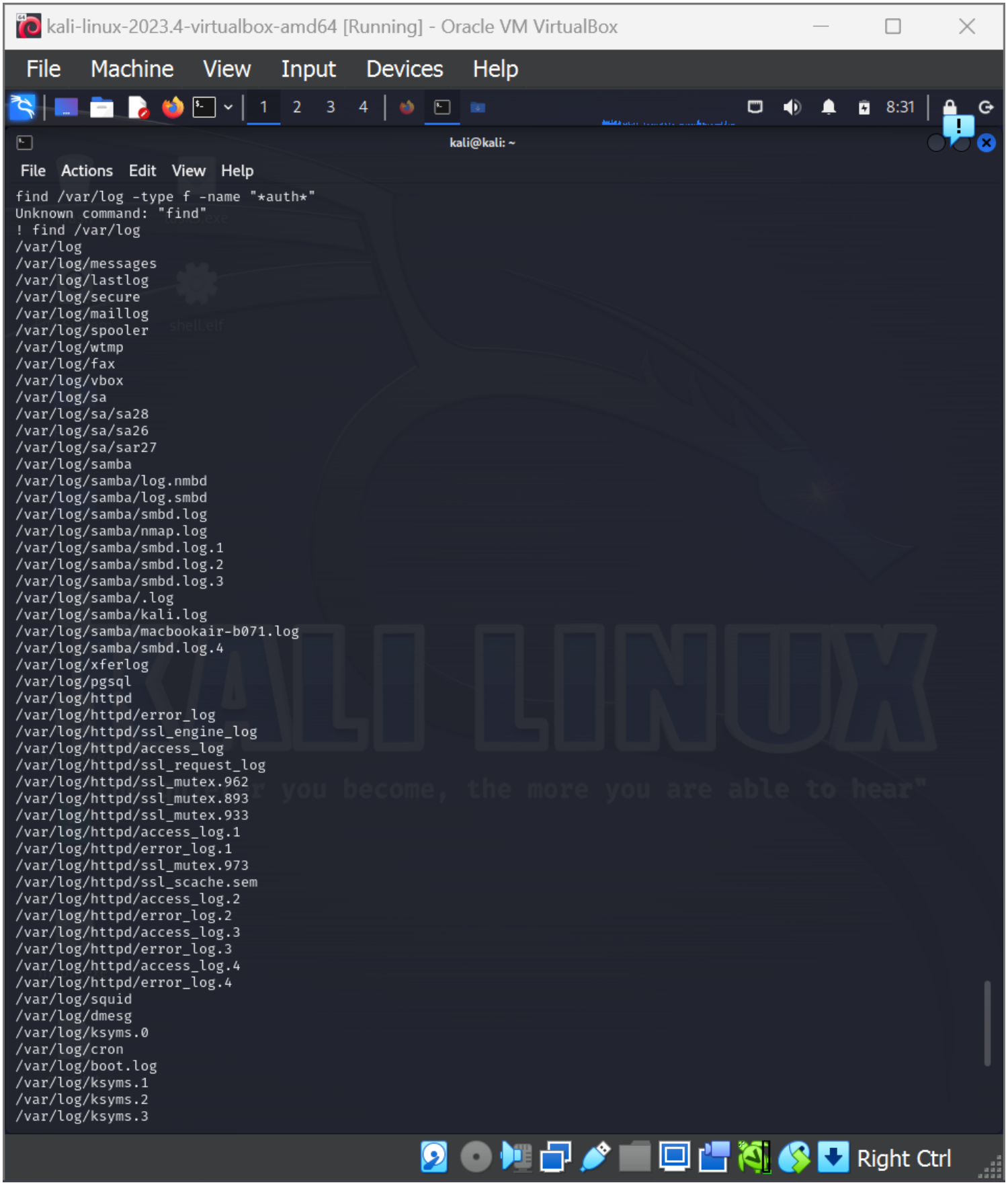
After discovering the exploit using Searchsploit, I navigated to the Metasploit console and selected an exploit module targeting SMB and the "trans2open" vulnerability. Following this, I configured the payload to initiate a shell reverse TCP connection.

I Successfully gained the session.

Privilege Escalation Clarification:

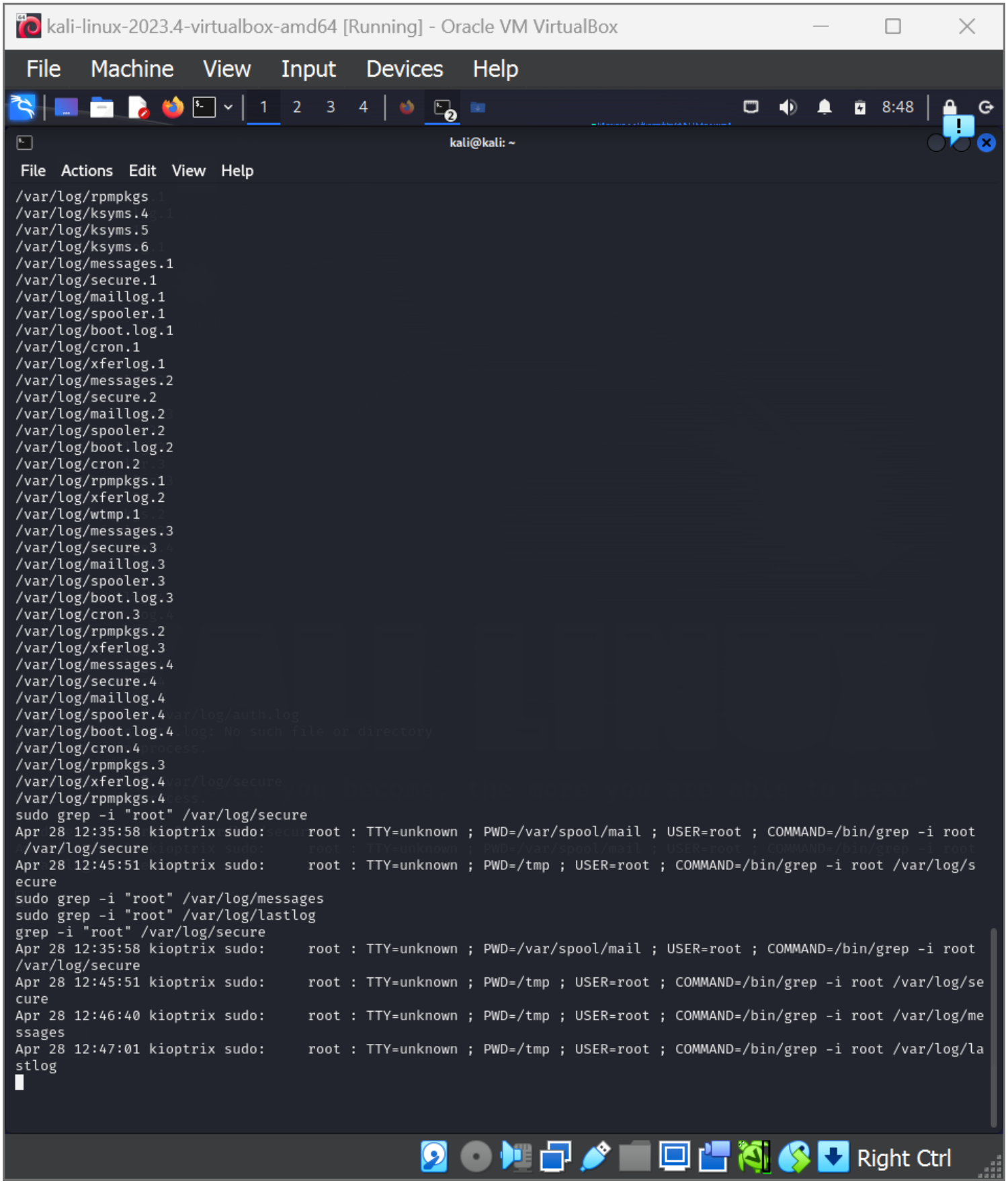
It's important to note that upon exploiting the vulnerability with my chosen payload, I was already authenticated as a root user. Therefore, there was no need for additional privilege escalation as the highest level of access had already been attained.

Logs on Target



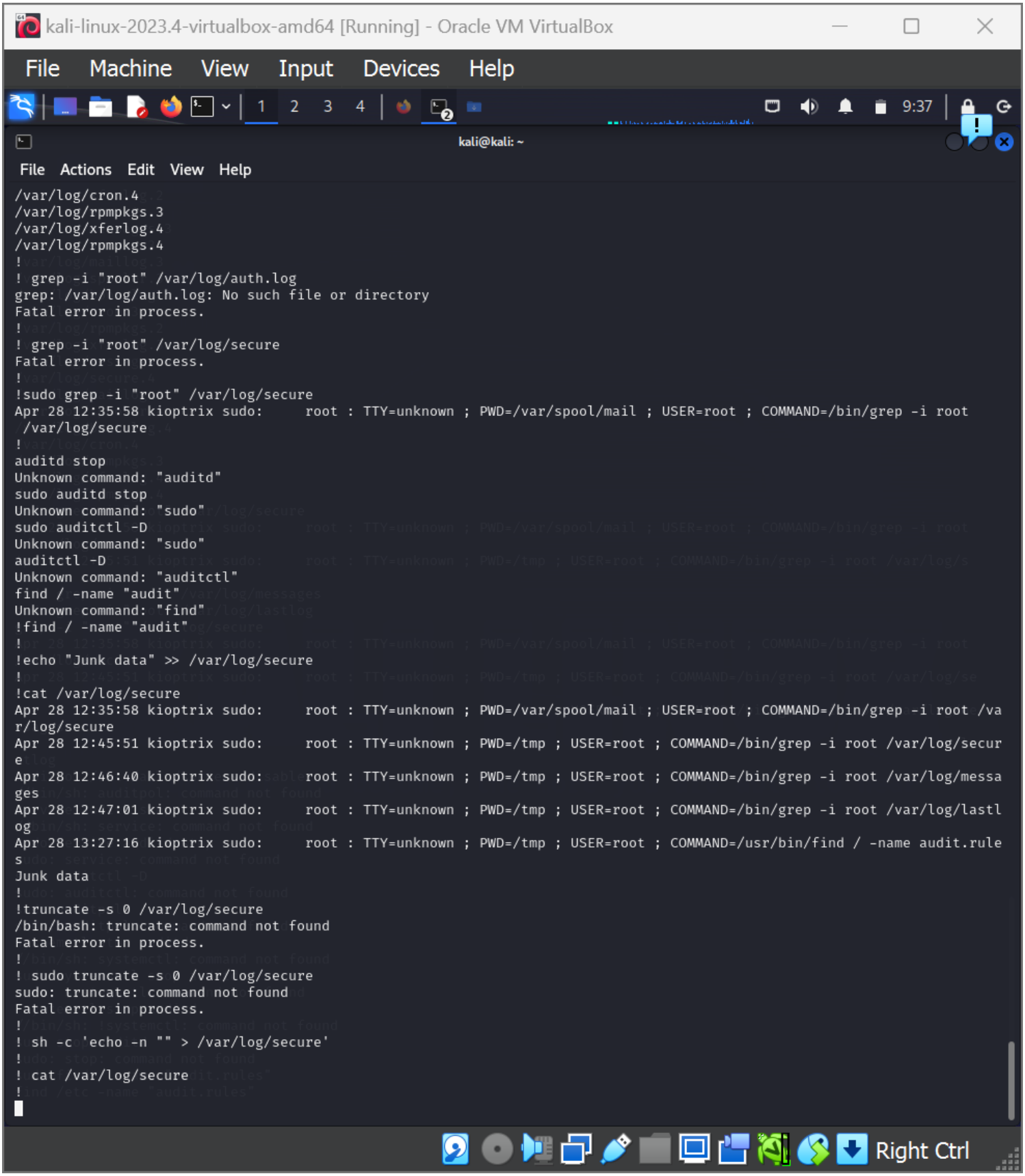
Covering Tracks:

As part of securing my actions, I initiated a search for the logs directory to ensure any potential traces of my activity could be identified and managed effectively.



Reviewing System Logs:

Upon examining the system logs, I found that the secure log contained records, whereas both the messages log and last log were found to be empty.



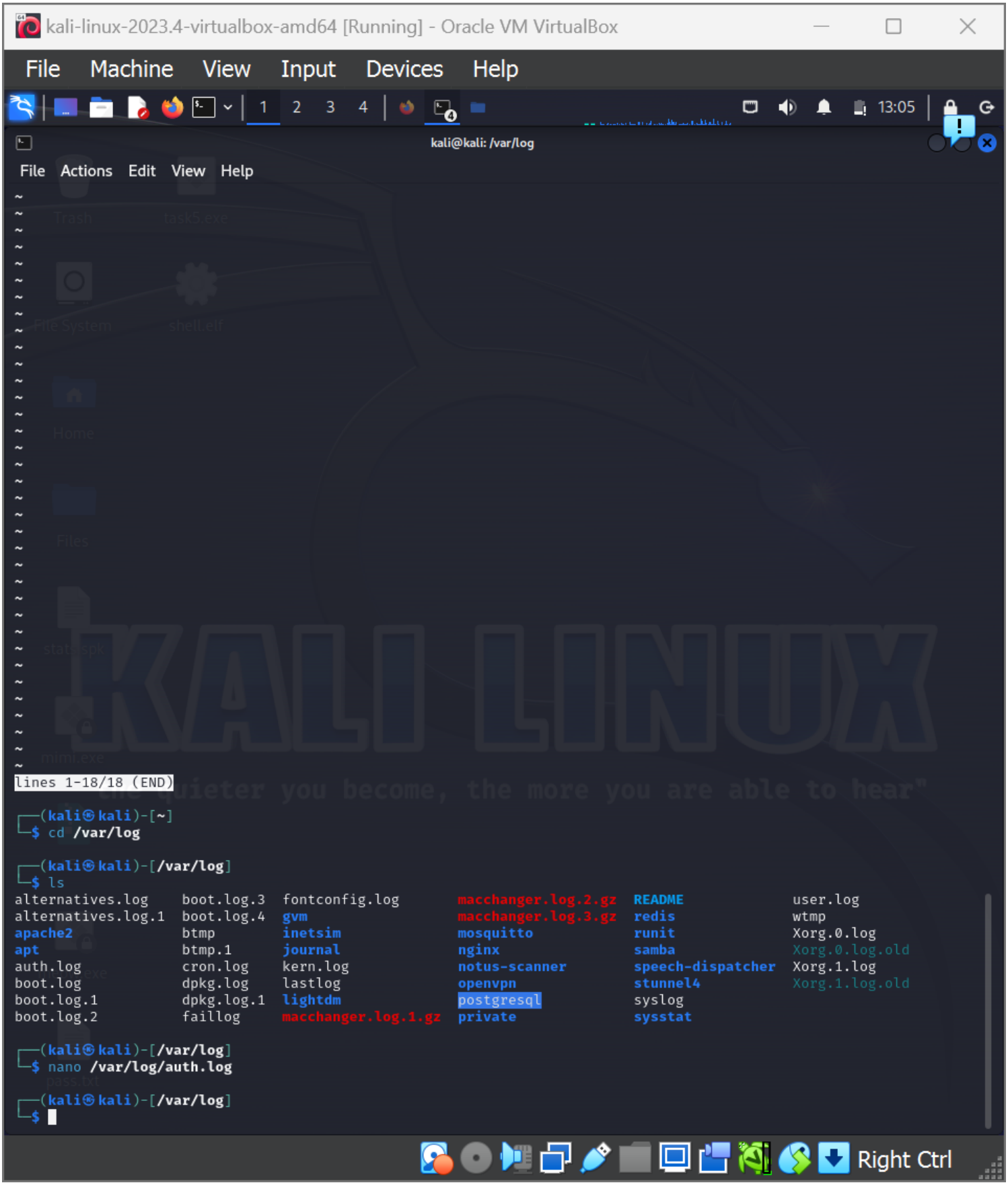
Taking Action Regarding Logs:

Using the command echo "Junk data" >> /var/log/secure, I added "Junk data"

to the end of the secure log file.
I’ve introduced noise into the log. While it won’t immediately affect system functionality, it can indeed make it harder to spot important events or security issues. System administrators rely on clean logs to troubleshoot and detect anomalies effectively.

To delete any potential traces of my activity, I executed the command echo "" > /var/log/secure, effectively emptying the contents of the secure log file.
The > symbol is used for output redirection in the shell.
In this case, it redirects the output of the echo command (which is an empty string) to the specified file (/var/log/secure).
The empty string ("") effectively overwrites the contents of the file with nothing, resulting in an empty file.

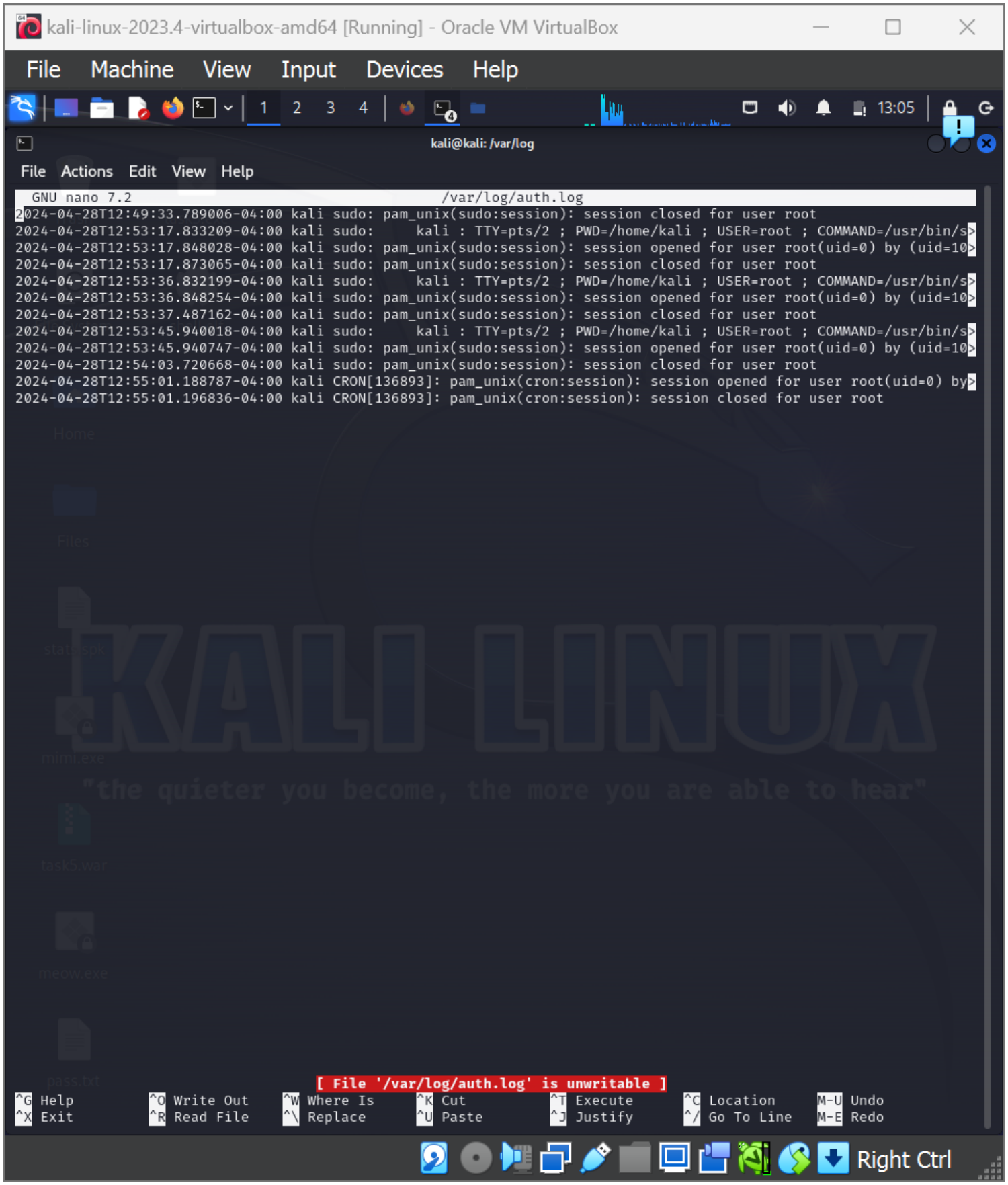
Logs on My Machine



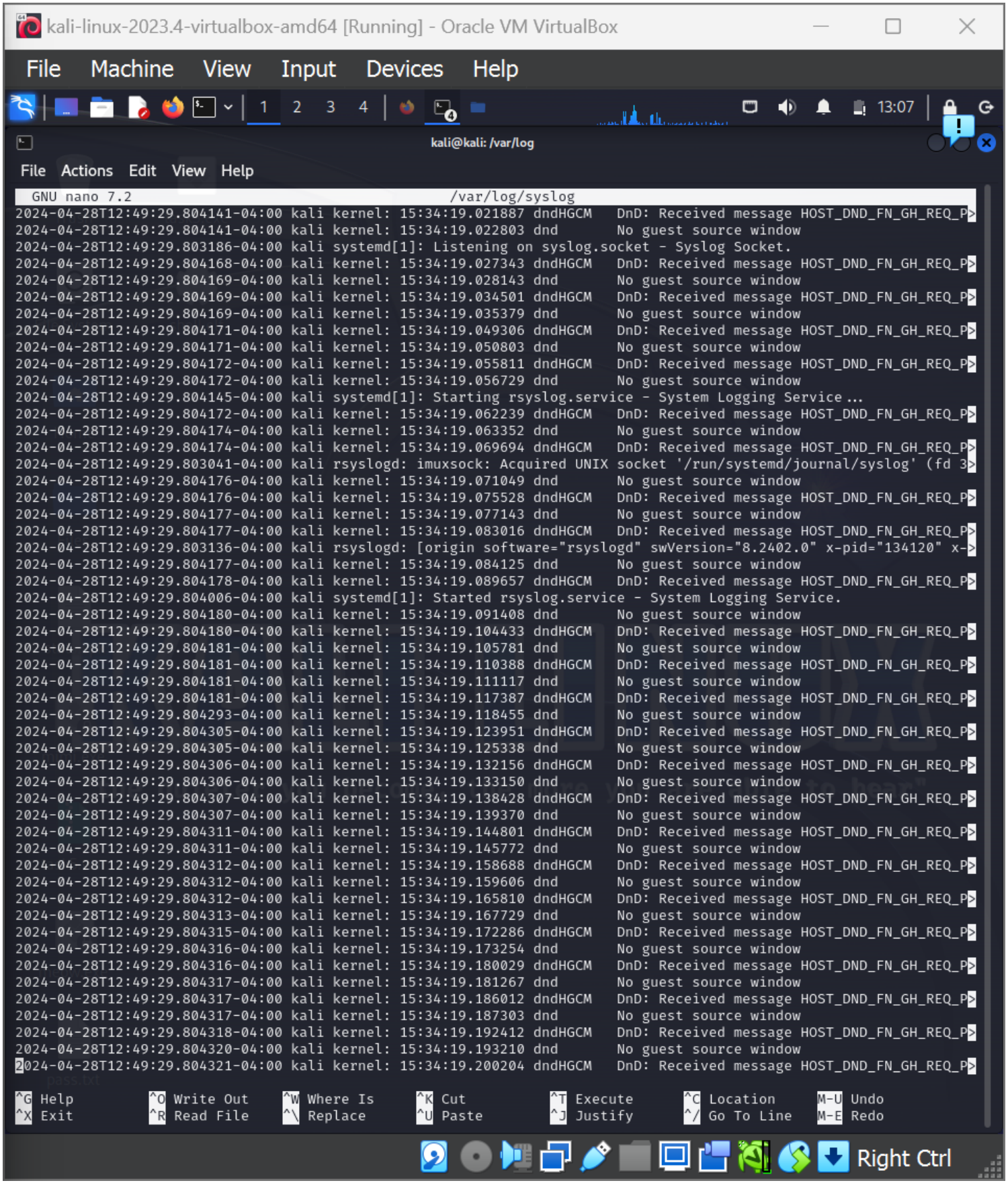
Following the deletion of logs on the target machine, I proceeded to review the logs directory on the attacker machine. Noteworthy logs identified included the

authentication log, system log, and kernel log, which were promptly targeted for deletion.

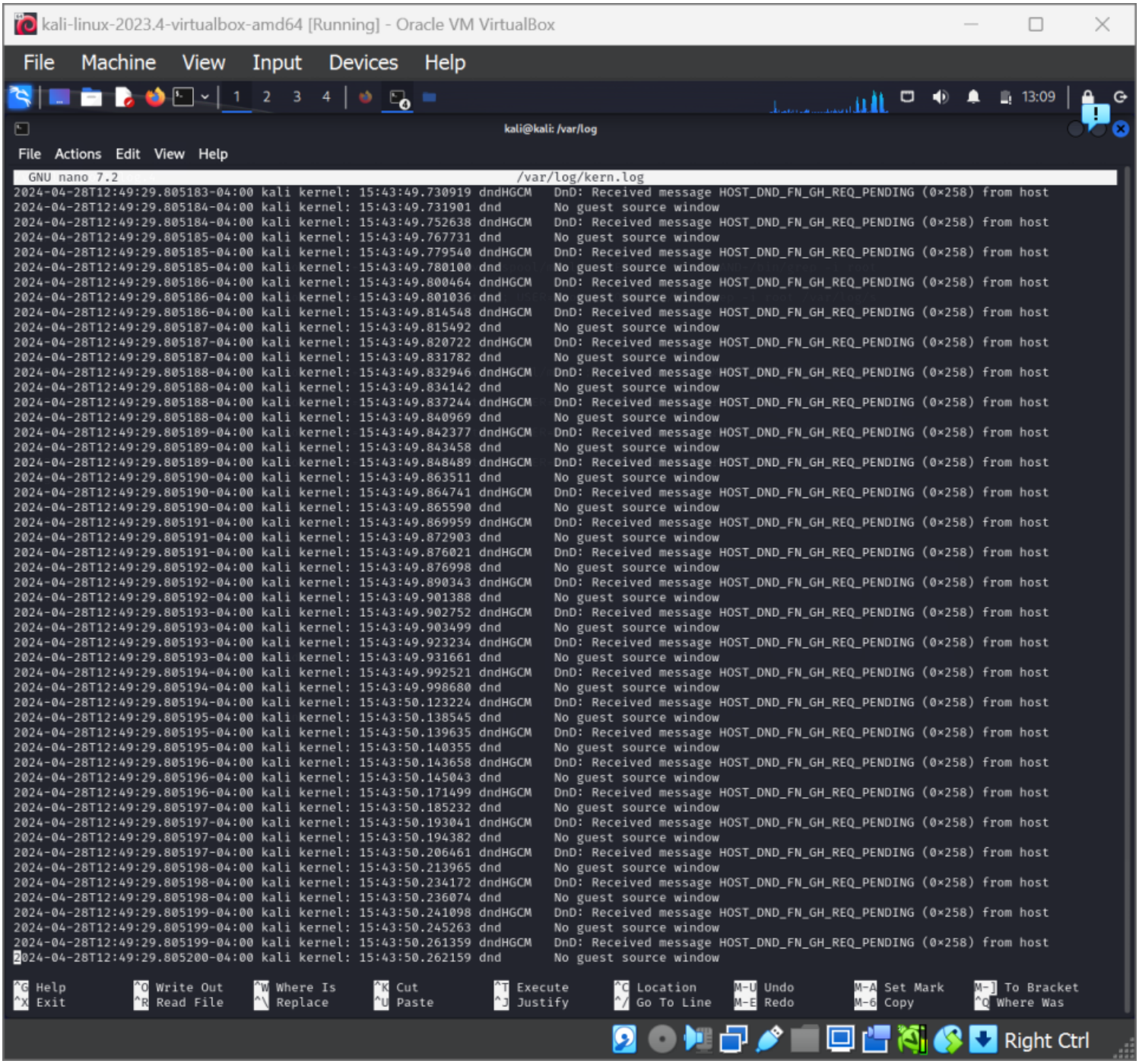
auth.log 👇

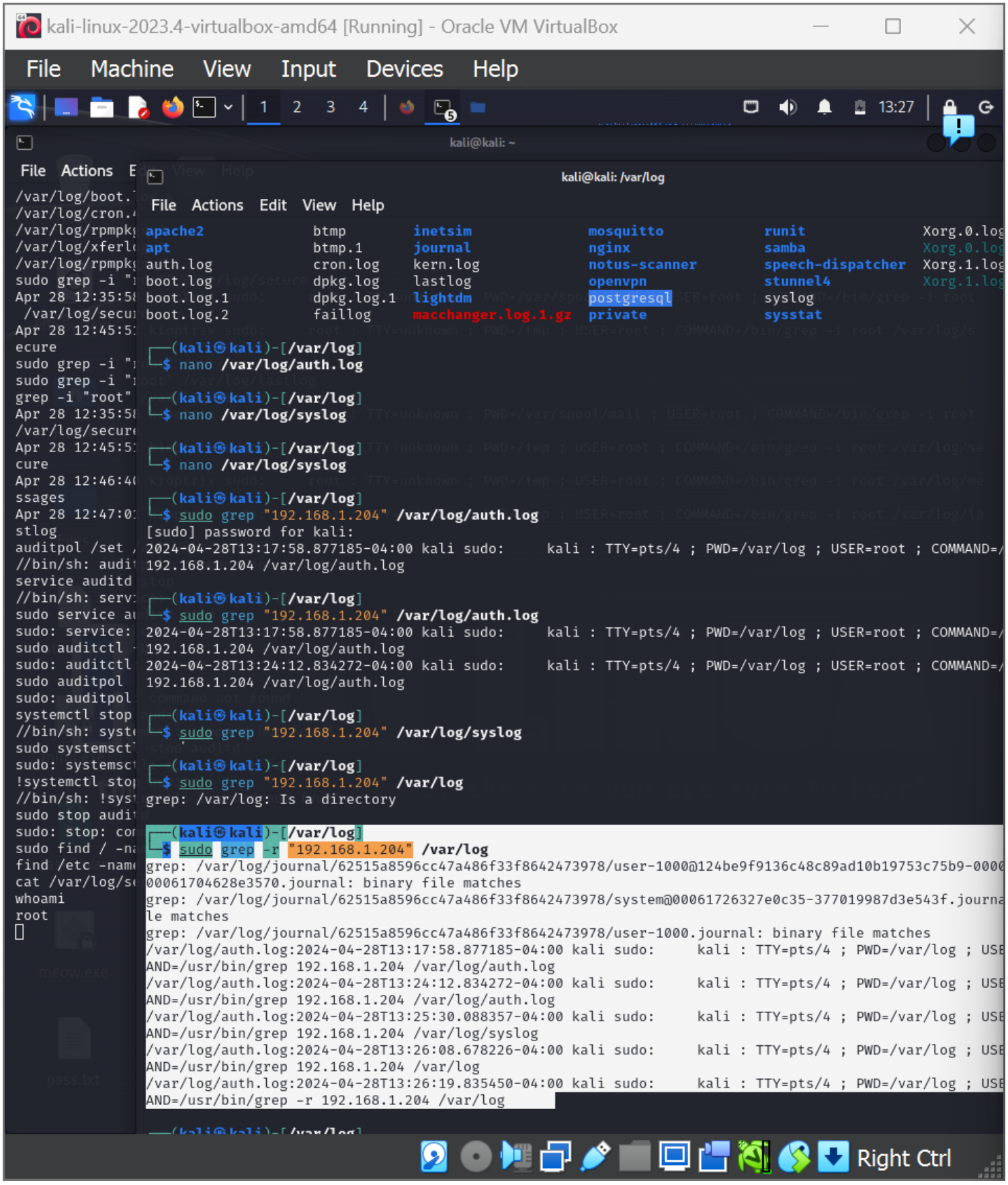


syslog 👇

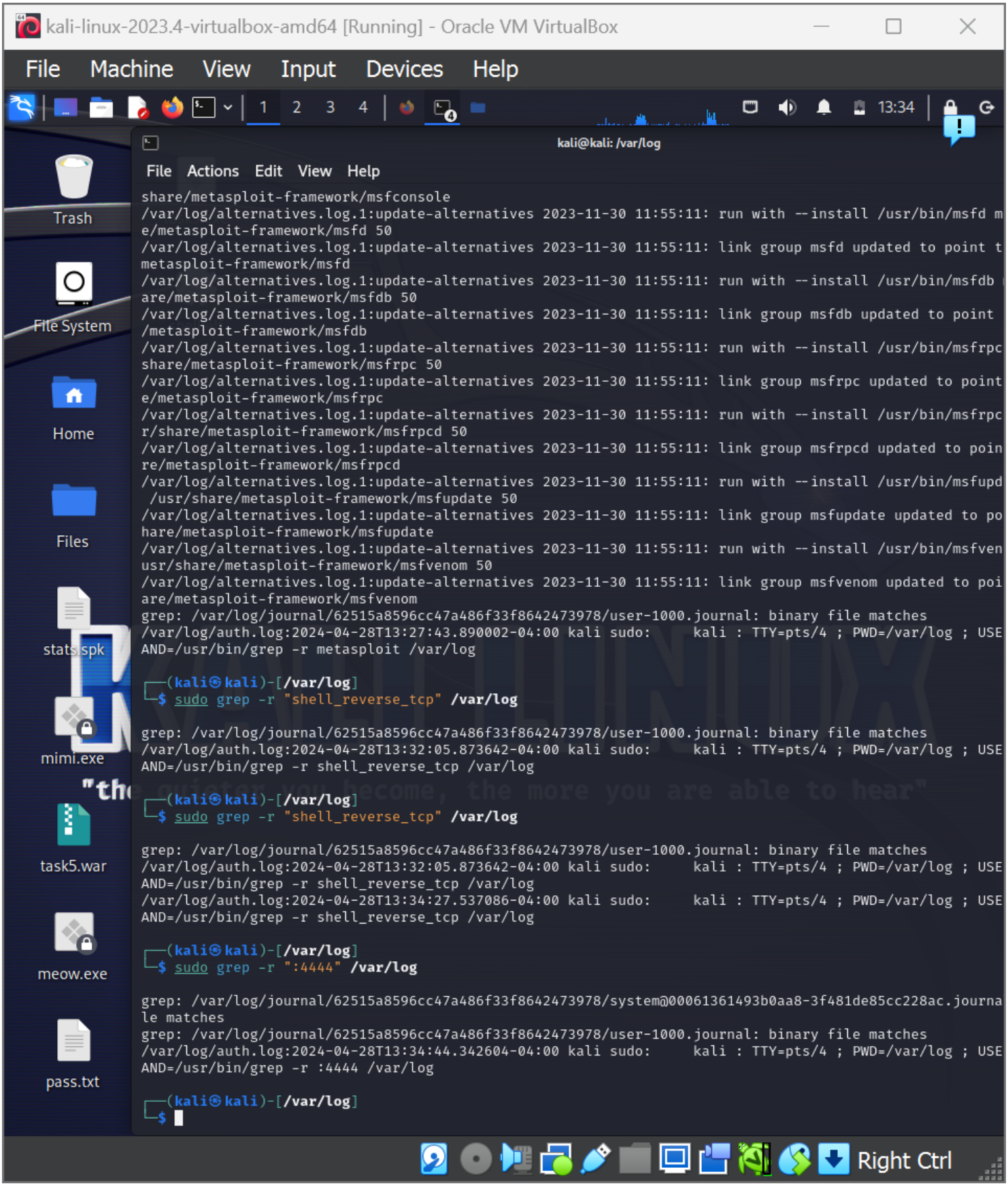


kern.log 👇

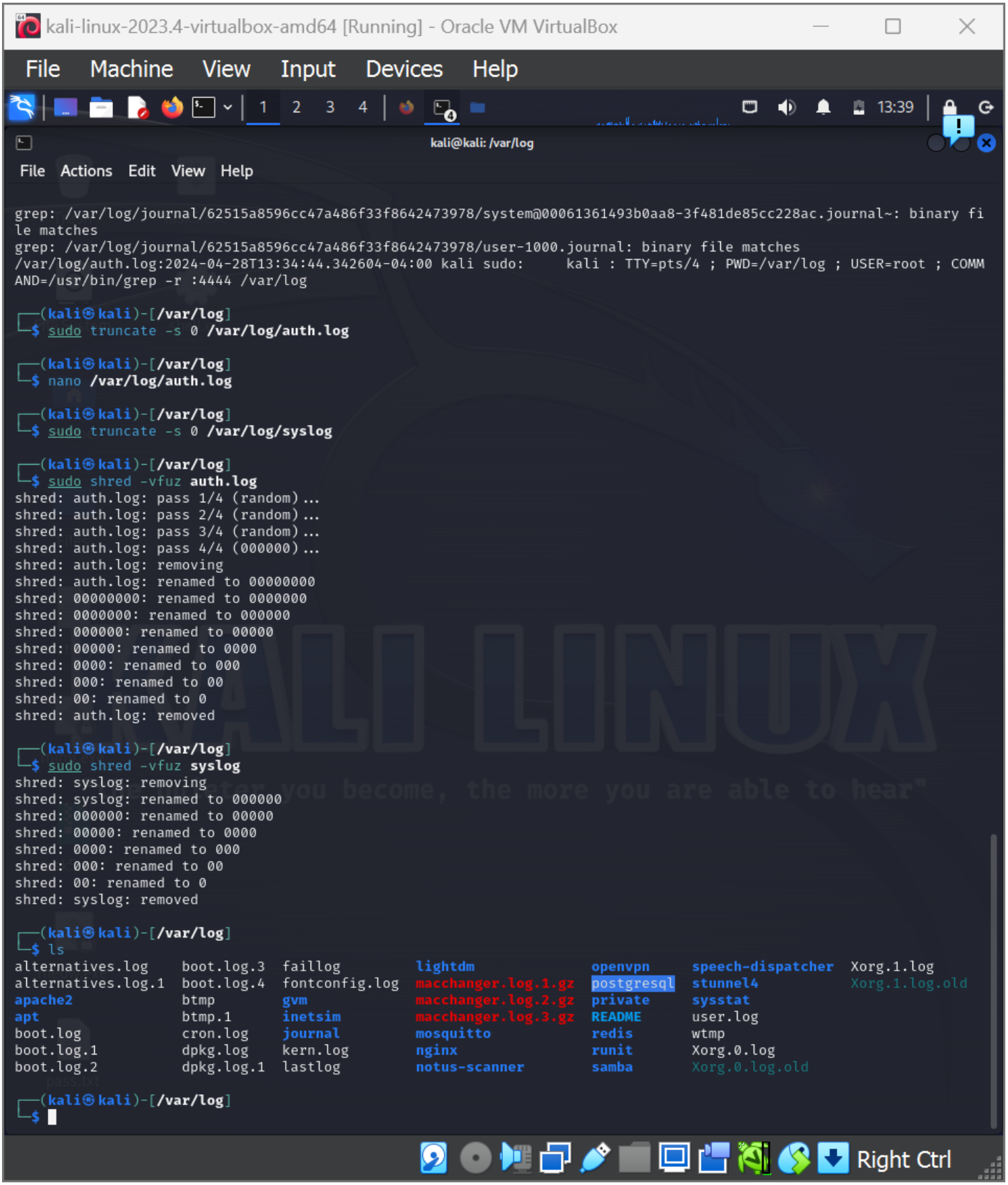

kali-linux-2023.4-virtualbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
13:09
kali@kali: /var/log
File Actions Edit View Help
GNU nano 7.2 /var/log/kern.log
2024-04-28T12:49:29.805183-04:00 kali kernel: 15:43:49.730919 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805184-04:00 kali kernel: 15:43:49.731901 dnd No guest source window
2024-04-28T12:49:29.805184-04:00 kali kernel: 15:43:49.752638 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805185-04:00 kali kernel: 15:43:49.767731 dnd No guest source window
2024-04-28T12:49:29.805185-04:00 kali kernel: 15:43:49.779540 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805185-04:00 kali kernel: 15:43:49.780100 dnd No guest source window
2024-04-28T12:49:29.805186-04:00 kali kernel: 15:43:49.800464 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805186-04:00 kali kernel: 15:43:49.801036 dnd No guest source window
2024-04-28T12:49:29.805186-04:00 kali kernel: 15:43:49.814548 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805187-04:00 kali kernel: 15:43:49.815492 dnd No guest source window
2024-04-28T12:49:29.805187-04:00 kali kernel: 15:43:49.820722 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805187-04:00 kali kernel: 15:43:49.831782 dnd No guest source window
2024-04-28T12:49:29.805188-04:00 kali kernel: 15:43:49.832946 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805188-04:00 kali kernel: 15:43:49.834142 dnd No guest source window
2024-04-28T12:49:29.805188-04:00 kali kernel: 15:43:49.837244 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805188-04:00 kali kernel: 15:43:49.840969 dnd No guest source window
2024-04-28T12:49:29.805189-04:00 kali kernel: 15:43:49.842377 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805189-04:00 kali kernel: 15:43:49.843458 dnd No guest source window
2024-04-28T12:49:29.805189-04:00 kali kernel: 15:43:49.848489 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805190-04:00 kali kernel: 15:43:49.863511 dnd No guest source window
2024-04-28T12:49:29.805190-04:00 kali kernel: 15:43:49.864741 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805190-04:00 kali kernel: 15:43:49.865590 dnd No guest source window
2024-04-28T12:49:29.805191-04:00 kali kernel: 15:43:49.869959 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805191-04:00 kali kernel: 15:43:49.872903 dnd No guest source window
2024-04-28T12:49:29.805191-04:00 kali kernel: 15:43:49.876021 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805192-04:00 kali kernel: 15:43:49.876998 dnd No guest source window
2024-04-28T12:49:29.805192-04:00 kali kernel: 15:43:49.890343 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805192-04:00 kali kernel: 15:43:49.901388 dnd No guest source window
2024-04-28T12:49:29.805193-04:00 kali kernel: 15:43:49.902752 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805193-04:00 kali kernel: 15:43:49.903499 dnd No guest source window
2024-04-28T12:49:29.805193-04:00 kali kernel: 15:43:49.923234 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805193-04:00 kali kernel: 15:43:49.931661 dnd No guest source window
2024-04-28T12:49:29.805194-04:00 kali kernel: 15:43:49.992521 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805194-04:00 kali kernel: 15:43:49.998680 dnd No guest source window
2024-04-28T12:49:29.805194-04:00 kali kernel: 15:43:50.123224 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805195-04:00 kali kernel: 15:43:50.138545 dnd No guest source window
2024-04-28T12:49:29.805195-04:00 kali kernel: 15:43:50.139635 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805195-04:00 kali kernel: 15:43:50.140355 dnd No guest source window
2024-04-28T12:49:29.805196-04:00 kali kernel: 15:43:50.143658 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805196-04:00 kali kernel: 15:43:50.145043 dnd No guest source window
2024-04-28T12:49:29.805196-04:00 kali kernel: 15:43:50.171499 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805197-04:00 kali kernel: 15:43:50.185232 dnd No guest source window
2024-04-28T12:49:29.805197-04:00 kali kernel: 15:43:50.193041 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805197-04:00 kali kernel: 15:43:50.194382 dnd No guest source window
2024-04-28T12:49:29.805197-04:00 kali kernel: 15:43:50.206461 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805198-04:00 kali kernel: 15:43:50.213965 dnd No guest source window
2024-04-28T12:49:29.805198-04:00 kali kernel: 15:43:50.234172 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805198-04:00 kali kernel: 15:43:50.236074 dnd No guest source window
2024-04-28T12:49:29.805199-04:00 kali kernel: 15:43:50.241098 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805199-04:00 kali kernel: 15:43:50.245263 dnd No guest source window
2024-04-28T12:49:29.805199-04:00 kali kernel: 15:43:50.261359 dndHGCM DnD: Received message HOST_DND_FN_GH_REQ_PENDING (0×258) from host
2024-04-28T12:49:29.805200-04:00 kali kernel: 15:43:50.262159 dnd No guest source window
^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location M-U Undo M-A Set Mark M-] To Bracket
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^/ Go To Line M-E Redo M-6 Copy ^Q Where Was
Right Ctrl



In the process of detecting logs on the attacker machine, I initiated a search for entries associated with the IP address of the target machine. This initial step aimed to identify any logs related to my attack.



In addition to searching for entries linked to the IP address of the target machine, I extended the search to include logs associated with the Metasploit reverse shell I deployed and the port utilized during the attack. This comprehensive approach aimed to uncover any relevant traces of the attack within the logs.



Upon locating logs pertaining to the target IP address, Metasploit reverse shell usage, and the port utilized, I proceeded to mitigate potential exposure by both truncating and shredding the files. This action effectively rendered the

logs empty, minimizing the visibility of the attack within the log files.a