

Information Security IA

Implementation of Honeypot using PenTbox



Sidharth Nair - 16010120032
Rahi Patil - 16010120038

Index

1

Introduction

2

Honeypot

3

PenTbox

4

**Stepwise
Demonstration of the
Tool**

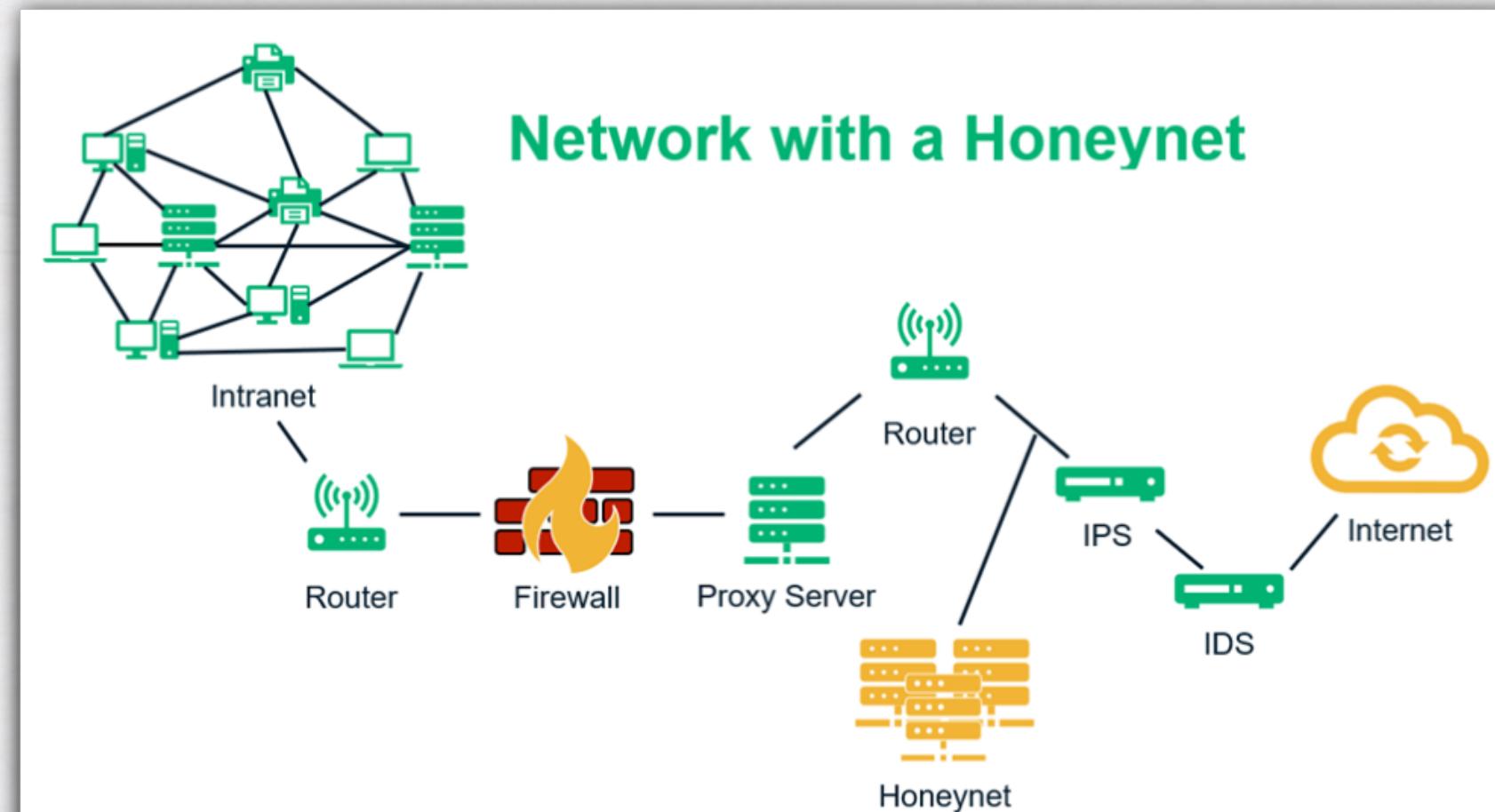
Introduction

- A Linux distribution with Debian roots called Kali Linux is made for penetration testing and digital forensics. Offensive Security oversees and provides maintenance for it.
- Kali Linux offers 600 penetration-testing applications, including Armitage, Nmap, Wireshark, John the Ripper, sqlmap, Aircrack-ing, Burp, and OWASP ZAP.
- It was created by Offensive Security employees Mati Aharoni and Devon Kearns through the rewriting of BackTrack, a Linux distribution they had previously used for information security testing and which was based on Knoppix. The Hindu deity Kali served as the name's inspiration.



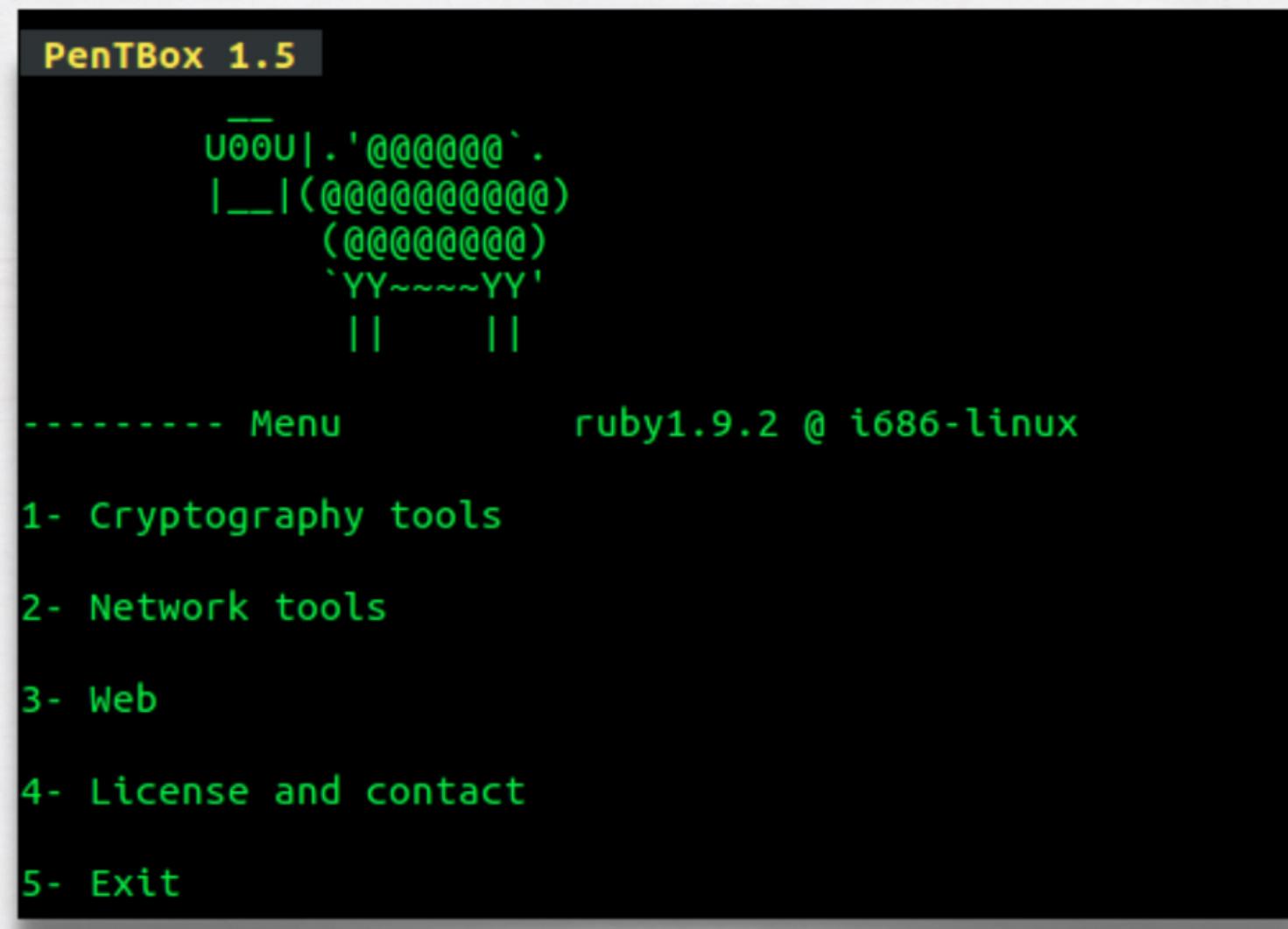
Honeypot

Honeypots are network-attached systems used by hackers to identify and research tactics and types of attacks. Large businesses and cybersecurity-related organisations are the main users, but crooks may use them to hoodwink researchers and disseminate false information.



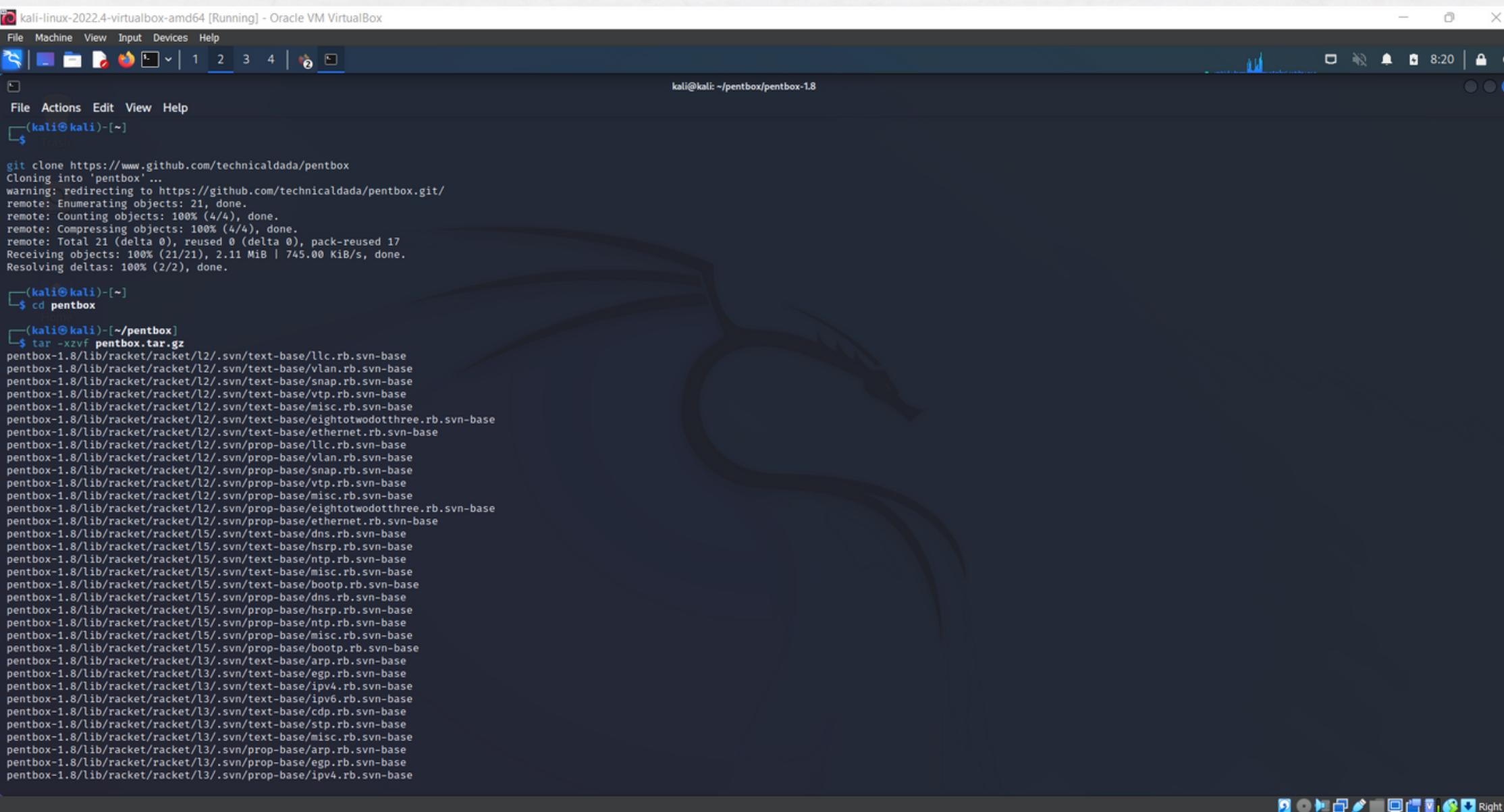
PenTbox

PenTBox is a security suite that can be used in penetration testing engagements to perform a variety of activities. Specifically these activities include from cracking hashes,DNS enumeration and stress testing to HTTP directory brute force.



Stepwise Demonstration

- To set up a honeypot in our Kali Linux system we need to download a tool from GitHub it called **Pentbox**. This tool is written in ruby language. To download this we use the following command:



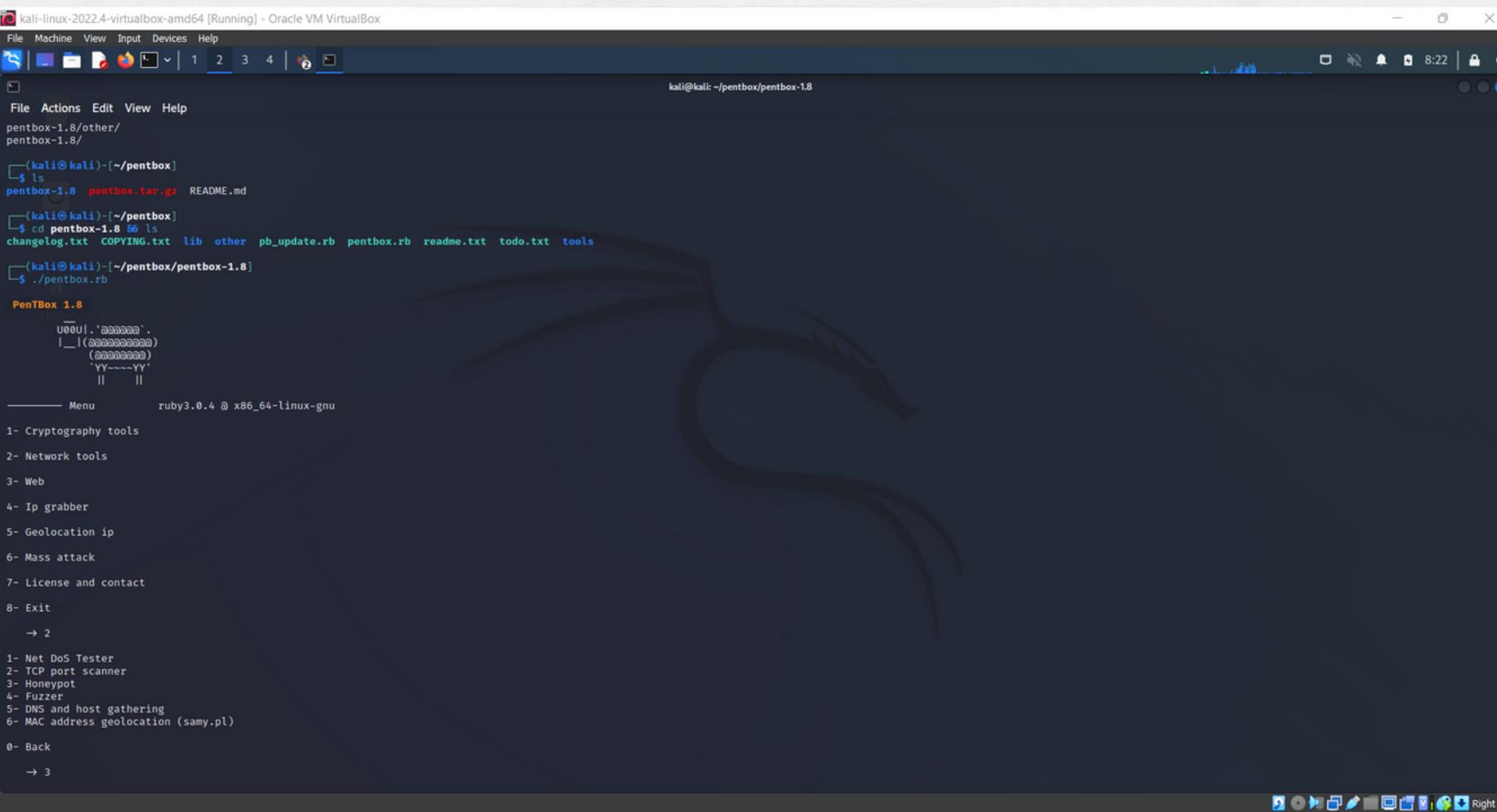
The screenshot shows a terminal window titled "kali-linux-2022.4-virtualbox-amd64 [Running] - Oracle VM VirtualBox". The terminal is running a command to clone the Pentbox repository from GitHub. The output of the command shows the progress of cloning, including object enumeration, compression, and receiving objects. After cloning, the user changes directory to the cloned repository and extracts its contents using tar. The terminal window has a dark background with light-colored text. The status bar at the bottom right shows the date and time as "8:20".

```
git clone https://www.github.com/technicaldada/pentbox
Cloning into 'pentbox'...
warning: redirecting to https://github.com/technicaldada/pentbox.git/
remote: Enumerating objects: 21, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 21 (delta 0), reused 0 (delta 0), pack-reused 17
Receiving objects: 100% (21/21), 2.11 MiB | 745.00 KiB/s, done.
Resolving deltas: 100% (2/2), done.

(kali㉿kali)-[~]
└─$ cd pentbox
(kali㉿kali)-[~/pentbox]
└─$ tar -xvf pentbox.tar.gz
pentbox-1.8/lib/racket/racket/12/.svn/text-base/l1c.rb.svn-base
pentbox-1.8/lib/racket/racket/12/.svn/text-base/vlan.rb.svn-base
pentbox-1.8/lib/racket/racket/12/.svn/text-base/snap.rb.svn-base
pentbox-1.8/lib/racket/racket/12/.svn/text-base/vtp.rb.svn-base
pentbox-1.8/lib/racket/racket/12/.svn/text-base/misc.rb.svn-base
pentbox-1.8/lib/racket/racket/12/.svn/text-base/eighttowodotthree.rb.svn-base
pentbox-1.8/lib/racket/racket/12/.svn/text-base/ethernet.rb.svn-base
pentbox-1.8/lib/racket/racket/12/.svn/prop-base/l1c.rb.svn-base
pentbox-1.8/lib/racket/racket/12/.svn/prop-base/vlan.rb.svn-base
pentbox-1.8/lib/racket/racket/12/.svn/prop-base/snap.rb.svn-base
pentbox-1.8/lib/racket/racket/12/.svn/prop-base/vtp.rb.svn-base
pentbox-1.8/lib/racket/racket/12/.svn/prop-base/misc.rb.svn-base
pentbox-1.8/lib/racket/racket/12/.svn/prop-base/eighttowodotthree.rb.svn-base
pentbox-1.8/lib/racket/racket/12/.svn/prop-base/ethernet.rb.svn-base
pentbox-1.8/lib/racket/racket/15/.svn/text-base/dns.rb.svn-base
pentbox-1.8/lib/racket/racket/15/.svn/text-base/hsrp.rb.svn-base
pentbox-1.8/lib/racket/racket/15/.svn/text-base/ntp.rb.svn-base
pentbox-1.8/lib/racket/racket/15/.svn/text-base/misc.rb.svn-base
pentbox-1.8/lib/racket/racket/15/.svn/text-base/bootp.rb.svn-base
pentbox-1.8/lib/racket/racket/15/.svn/prop-base/dns.rb.svn-base
pentbox-1.8/lib/racket/racket/15/.svn/prop-base/hsrp.rb.svn-base
pentbox-1.8/lib/racket/racket/15/.svn/prop-base/ntp.rb.svn-base
pentbox-1.8/lib/racket/racket/15/.svn/prop-base/misc.rb.svn-base
pentbox-1.8/lib/racket/racket/15/.svn/text-base/arp.rb.svn-base
pentbox-1.8/lib/racket/racket/13/.svn/text-base/egp.rb.svn-base
pentbox-1.8/lib/racket/racket/13/.svn/text-base/ipv4.rb.svn-base
pentbox-1.8/lib/racket/racket/13/.svn/text-base/ipv6.rb.svn-base
pentbox-1.8/lib/racket/racket/13/.svn/text-base/cdp.rb.svn-base
pentbox-1.8/lib/racket/racket/13/.svn/text-base/stp.rb.svn-base
pentbox-1.8/lib/racket/racket/13/.svn/text-base/misc.rb.svn-base
pentbox-1.8/lib/racket/racket/13/.svn/prop-base/arp.rb.svn-base
pentbox-1.8/lib/racket/racket/13/.svn/prop-base/egp.rb.svn-base
pentbox-1.8/lib/racket/racket/13/.svn/prop-base/ipv4.rb.svn-base
```

Stepwise Demonstration

- Then we need to go into the pentbox folder by using cd command. Here we have a compressed file named pentbox.tar.xz and to extract it. Then we run this ruby tool by using simple command as following: ./pentbox.rb



```
kali@kali: ~/pentbox/pentbox-1.8
File Machine View Input Devices Help
pentbox-1.8/other/
pentbox-1.8/
(kali㉿kali)-[~/pentbox]
└─$ ls
pentbox-1.8.tar.gz README.md
(kali㉿kali)-[~/pentbox]
└─$ cd pentbox-1.8 && ls
CHANGELOG.txt COPYING.txt lib other pb_update.rb pentbox.rb readme.txt todo.txt tools
(kali㉿kali)-[~/pentbox/pentbox-1.8]
└─$ ./pentbox.rb

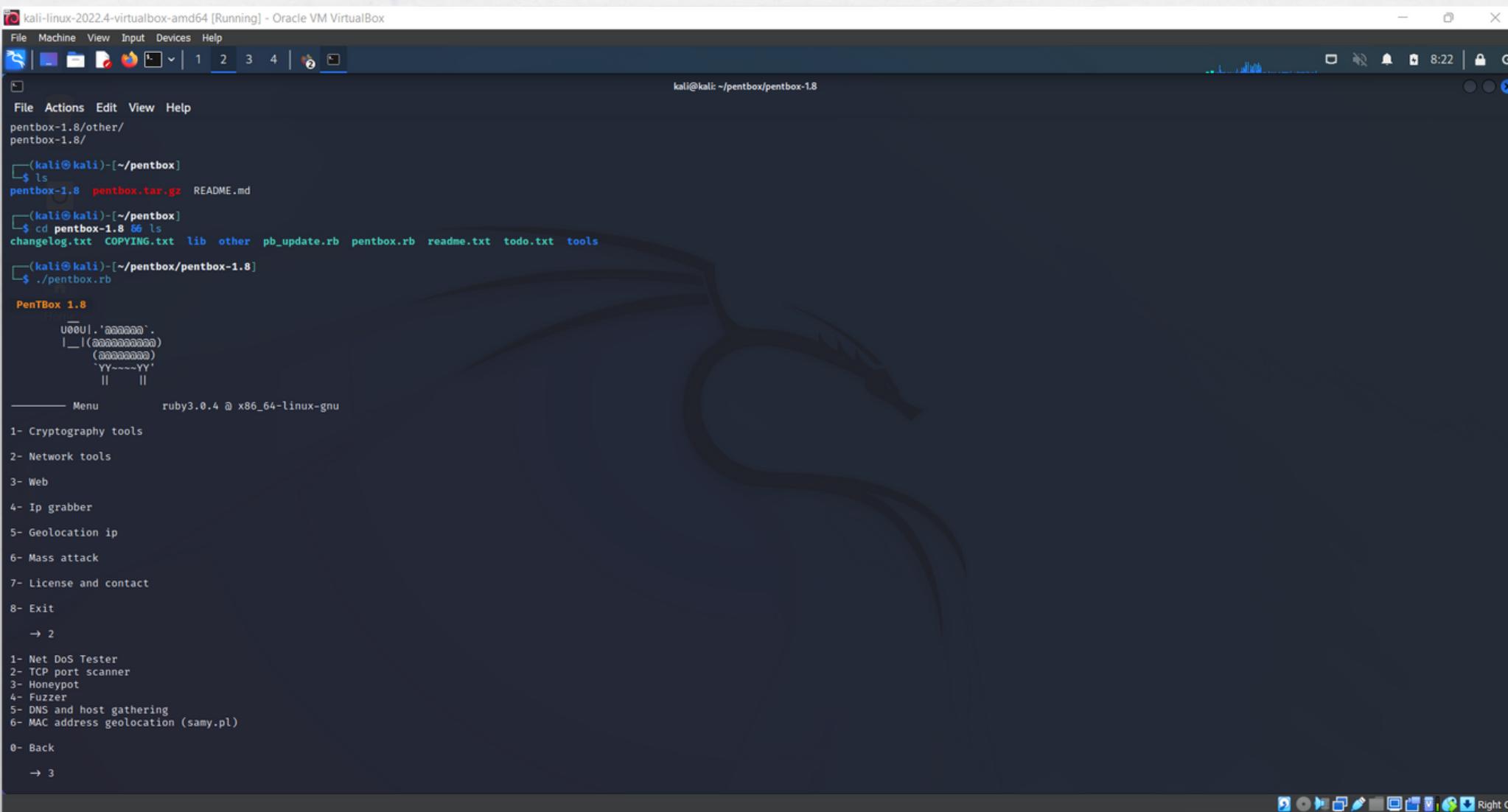
PenTBox 1.8
  _U_00U|,'_aaaaaaa'.
  |_((aaaaaaaaaa))
  |  (aaaaaaaaaa)
  YY----YY'
  ||  ||
----- Menu ----- ruby3.0.4 @ x86_64-linux-gnu
1- Cryptography tools
2- Network tools
3- Web
4- IP grabber
5- Geolocation ip
6- Mass attack
7- License and contact
8- Exit
→ 2

1- Net DoS Tester
2- TCP port scanner
3- Honeypot
4- Fuzzer
5- DNS and host gathering
6- MAC address geolocation (samy.pl)

0- Back
→ 3
```

Stepwise Demonstration

- Then this tool will open. Here we need to go to the Network tools option. Then we can see the Honeypot option.

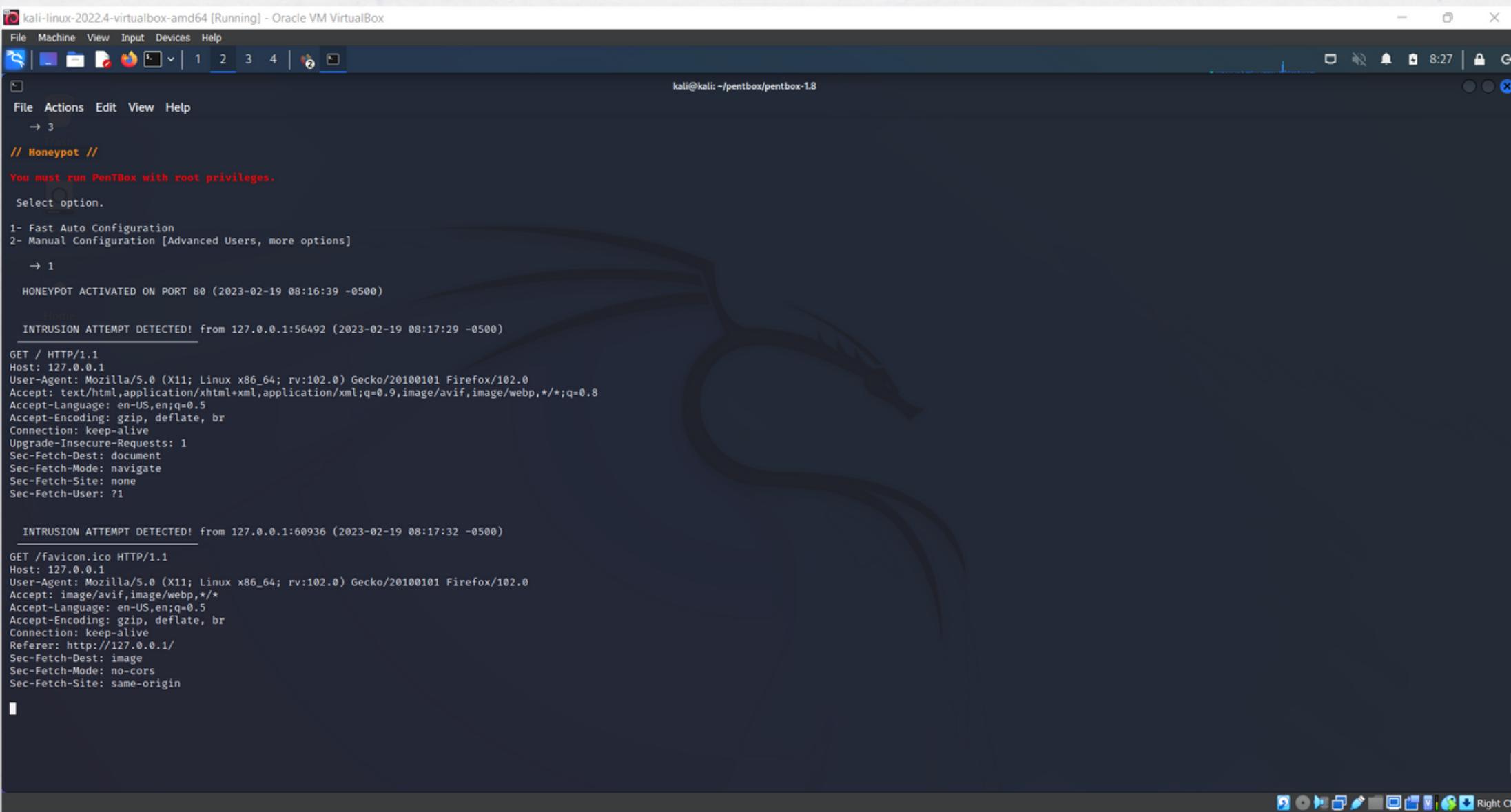


```
kali@kali: ~/pentbox/pentbox-1.8
File Machine View Input Devices Help
File Actions Edit View Help
pentbox-1.8/other/
pentbox-1.8/
(kali㉿kali)-[~/pentbox]
$ ls
pentbox-1.8 pentbox.tar.gz README.md
(kali㉿kali)-[~/pentbox]
$ cd pentbox-1.8 & ls
changelog.txt COPYING.txt lib other pb_update.rb pentbox.rb readme.txt todo.txt tools
(kali㉿kali)-[~/pentbox/pentbox-1.8]
$ ./pentbox.rb

PenTBox 1.8
  _U_ .'.aaaaaa'.
  |_(aaaaaaaaaaa)
  (aaaaaaaaaaa)
  'Y~~~YY'
  ||  ||
----- Menu      ruby3.0.4 @ x86_64-linux-gnu
1- Cryptography tools
2- Network tools
3- Web
4- Ip grabber
5- Geolocation ip
6- Mass attack
7- License and contact
8- Exit
→ 2
1- Net DoS Tester
2- TCP port scanner
3- Honeypot
4- Fuzzer
5- DNS and host gathering
6- MAC address geolocation (samy.pl)
0- Back
→ 3
```

Stepwise Demonstration

- Here we can choose 1 for auto configuration this will be fast or we can choose 2 for manual configuration. Manual configuration contains more options but it is for advanced users.



The screenshot shows a terminal window titled "kali-linux-2022.4-virtualbox-amd64 [Running] - Oracle VM VirtualBox". The window displays a configuration interface for a honeypot. The text in the terminal is as follows:

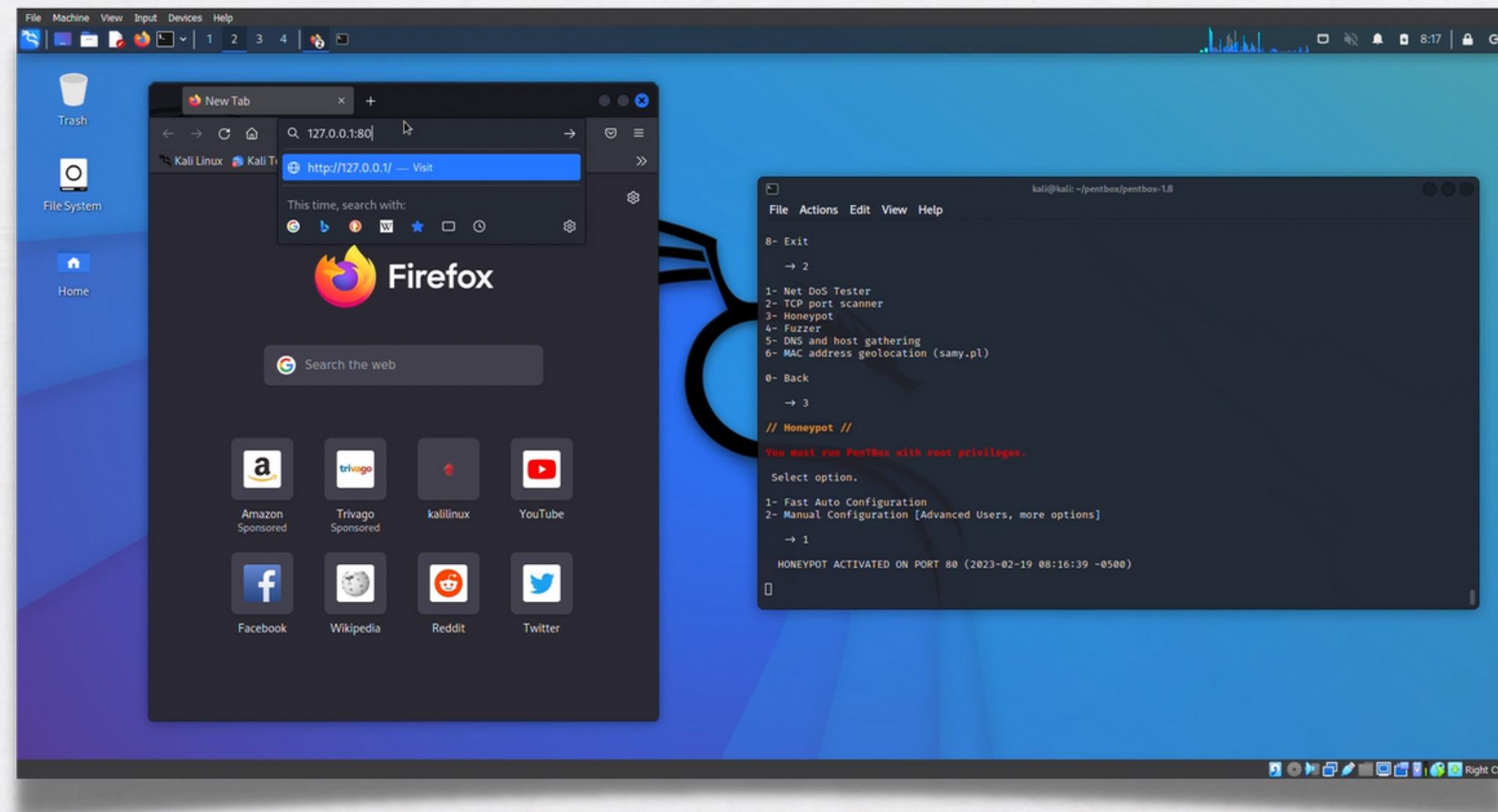
```
// Honeypot //
You must run PenTBx with root privileges.
Select option.
1- Fast Auto Configuration
2- Manual Configuration [Advanced Users, more options]
→ 1
HONEYBOT ACTIVATED ON PORT 80 (2023-02-19 08:16:39 -0500)

INTRUSION ATTEMPT DETECTED! from 127.0.0.1:56492 (2023-02-19 08:17:29 -0500)
GET / HTTP/1.1
Host: 127.0.0.1
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: none
Sec-Fetch-User: ?1

INTRUSION ATTEMPT DETECTED! from 127.0.0.1:60936 (2023-02-19 08:17:32 -0500)
GET /favicon.ico HTTP/1.1
Host: 127.0.0.1
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
Accept: image/avif,image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Referer: http://127.0.0.1/
Sec-Fetch-Dest: image
Sec-Fetch-Mode: no-cors
Sec-Fetch-Site: same-origin
```

Stepwise Demonstration

- Now we can see that we have successfully run honeypot in our localhost on port 80. To check how it works we can go to browser and check our localhost that is 127.0.0.1:80 and then check in the terminal where we started honeypot



Stepwise Demonstration

The screenshot shows a terminal window titled "kali-linux-2022.4-virtualbox-amd64 [Running] - Oracle VM VirtualBox". The terminal displays the following text:

```
// Honeypot //
You must run PentBox with root privileges.
Select option.
1- Fast Auto Configuration
2- Manual Configuration [Advanced Users, more options]
→ 1
HONEYBOT ACTIVATED ON PORT 80 (2023-02-19 08:16:39 -0500)

Home
INTRUSION ATTEMPT DETECTED! from 127.0.0.1:56492 (2023-02-19 08:17:29 -0500)
GET / HTTP/1.1
Host: 127.0.0.1
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: none
Sec-Fetch-User: ?1

INTRUSION ATTEMPT DETECTED! from 127.0.0.1:60936 (2023-02-19 08:17:32 -0500)
GET /favicon.ico HTTP/1.1
Host: 127.0.0.1
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
Accept: image/avif,image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Referer: http://127.0.0.1/
Sec-Fetch-Dest: image
Sec-Fetch-Mode: no-cors
Sec-Fetch-Site: same-origin
```

*Thank
you*