**SPR600**

**Final Individual Projec**t

Attack Tutorial Report

Name: Ananthu Krishna Vadakkeppatt

Student ID: 154290217

Course ID: SPR600

Professor’s Name: Mohammed Al-Ani

Date: 07-04-2024

**Table of Contents**

[Network Setup 2](#_Toc1535058163)

[Network Table 3](#_Toc1056109519)

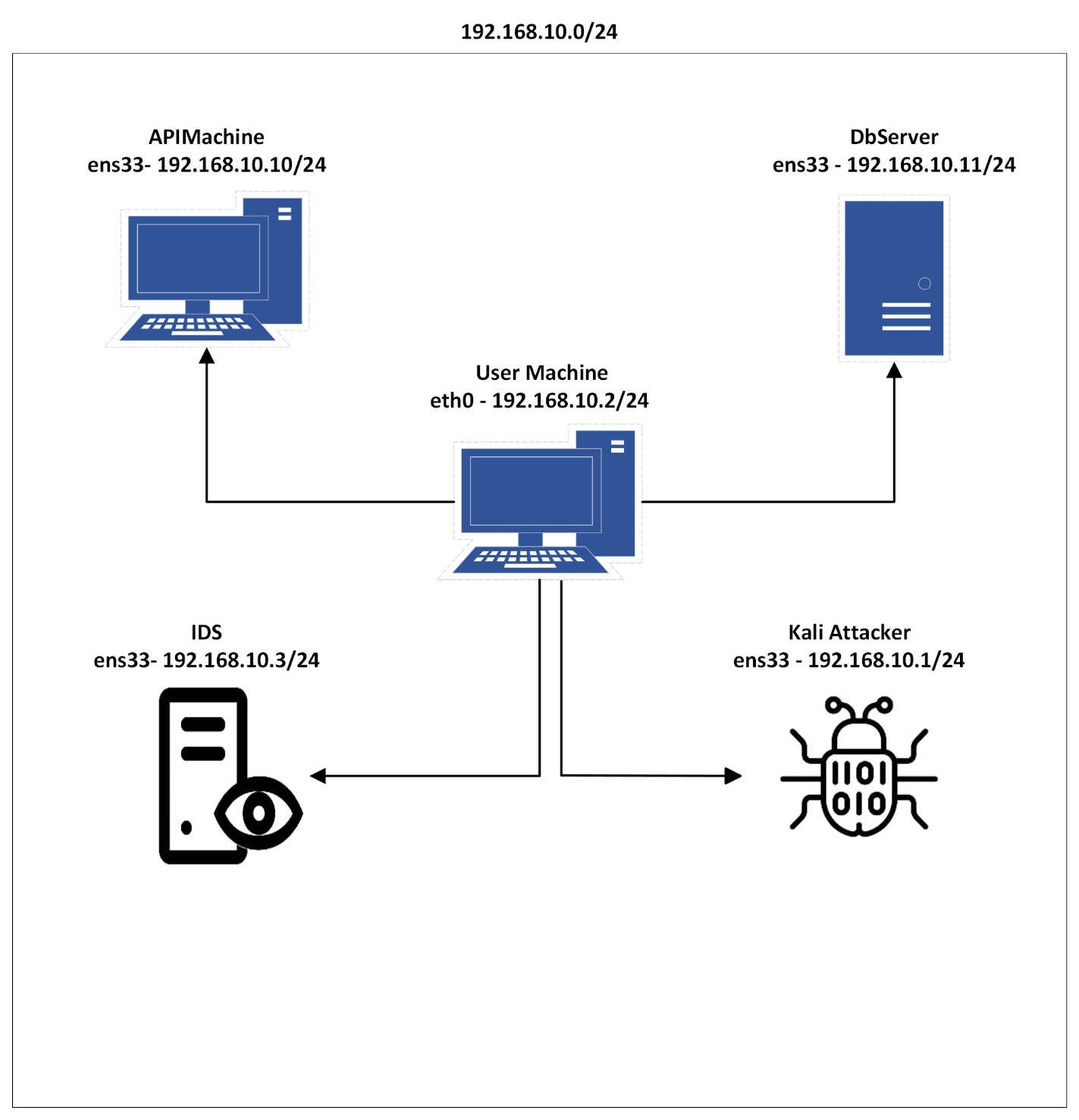
[Overview of The Infrastructure 4](#_Toc638536468)

[Attack Methodology 4](#_Toc910549942)

[Demonstration of The Attack 4](#_Toc247937378)

[References 4](#_Toc1427925691)

# **Network Setup**



# **Network Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.NO** | **IP** | **OS** | **ROLE** |
| 1. | 192.168.10.10 | Ubuntu 22.04 | API Server |
| 2. | 192.168.10.11 | Ubuntu 22.04 | Database Server |
| 3. | 192.168.10.2 | Windows 2016 Server | User Machine |
| 4. | 192.168.10.3 | Ubuntu 22.04 | Threat Hunting Machine |
| 5. | 192.168.10.1 | Kali | Attacker |

# **Overview of The Infrastructure**

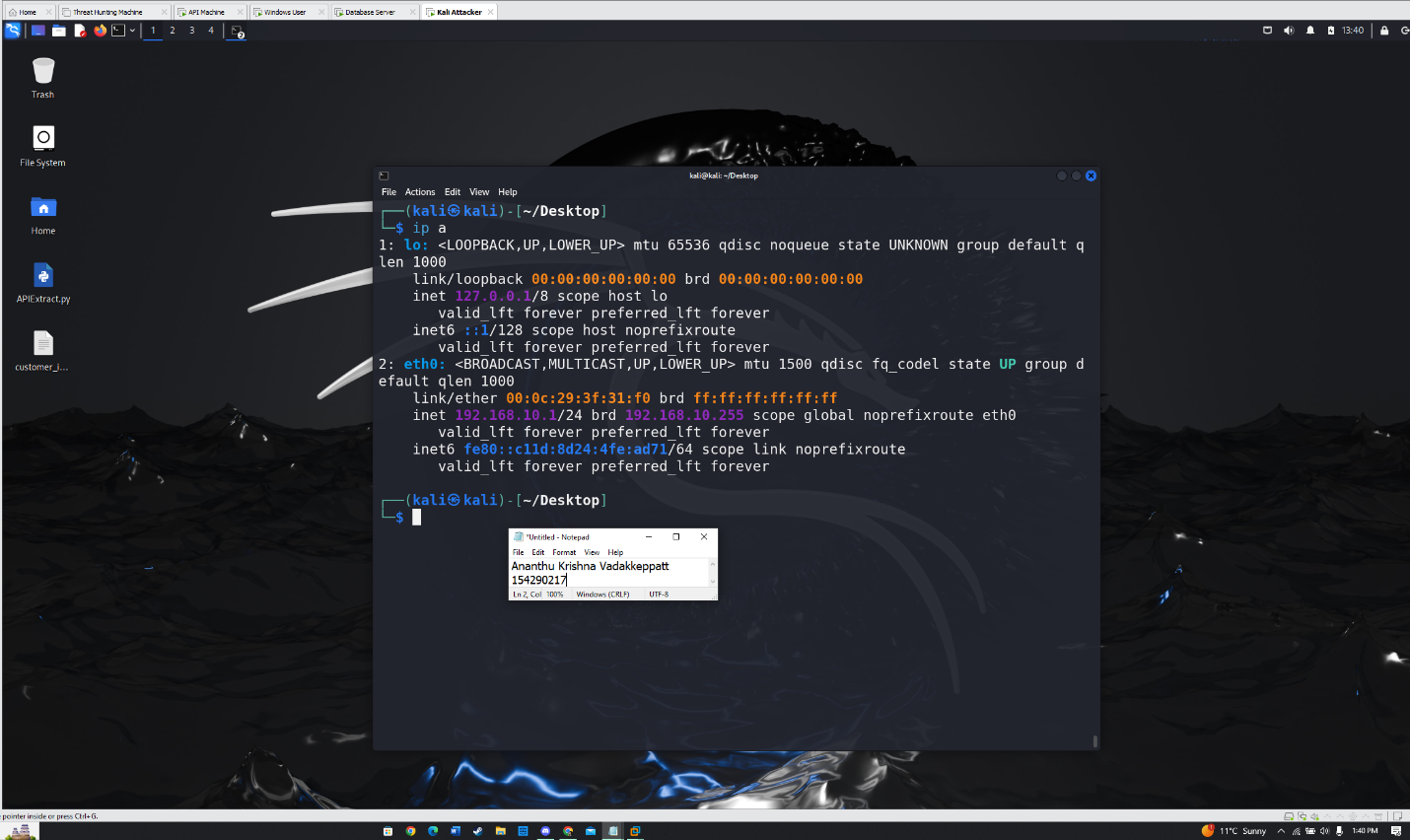
This attack is focused on mimicking the Optus 2022 Data Breach that involved the organization exposing an API that was unauthenticated. The attacker exfiltrated customer credentials from the same used to ask for ransom.

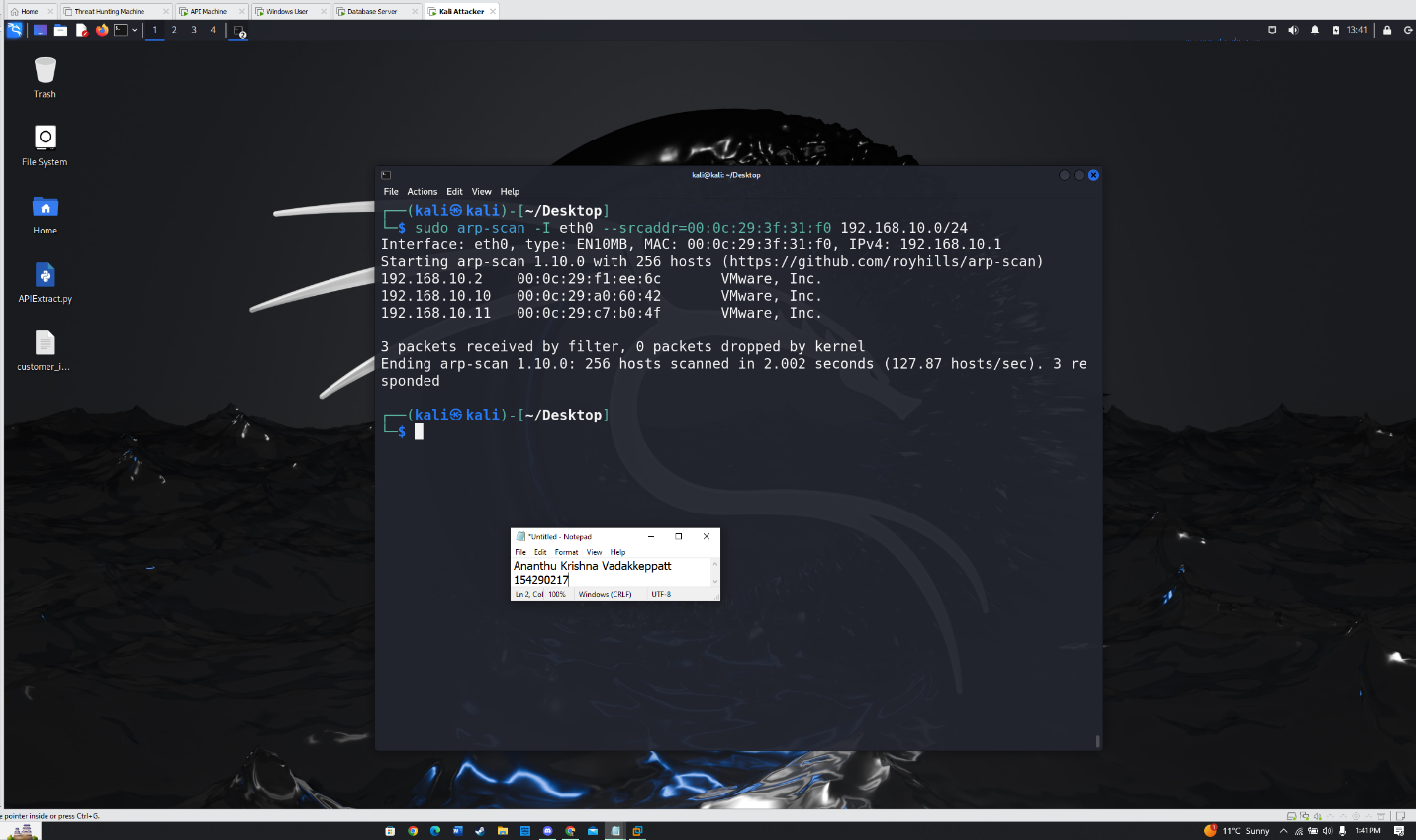
In the replica of this attack, I have set up an infrastructure with five machines across the 192.168.10.0/24 network. The three target machines include a database server, an API hosting machine and an end user that contains a domain. The database server is equipped using the MySQL tool to create a database and store customer sensitive data. On the other hand, the API machine is used to deploy the unauthenticated API using the flask framework. Along with these machines there is a user end windows that is configured to run a domain of the organization and manage these critical services.

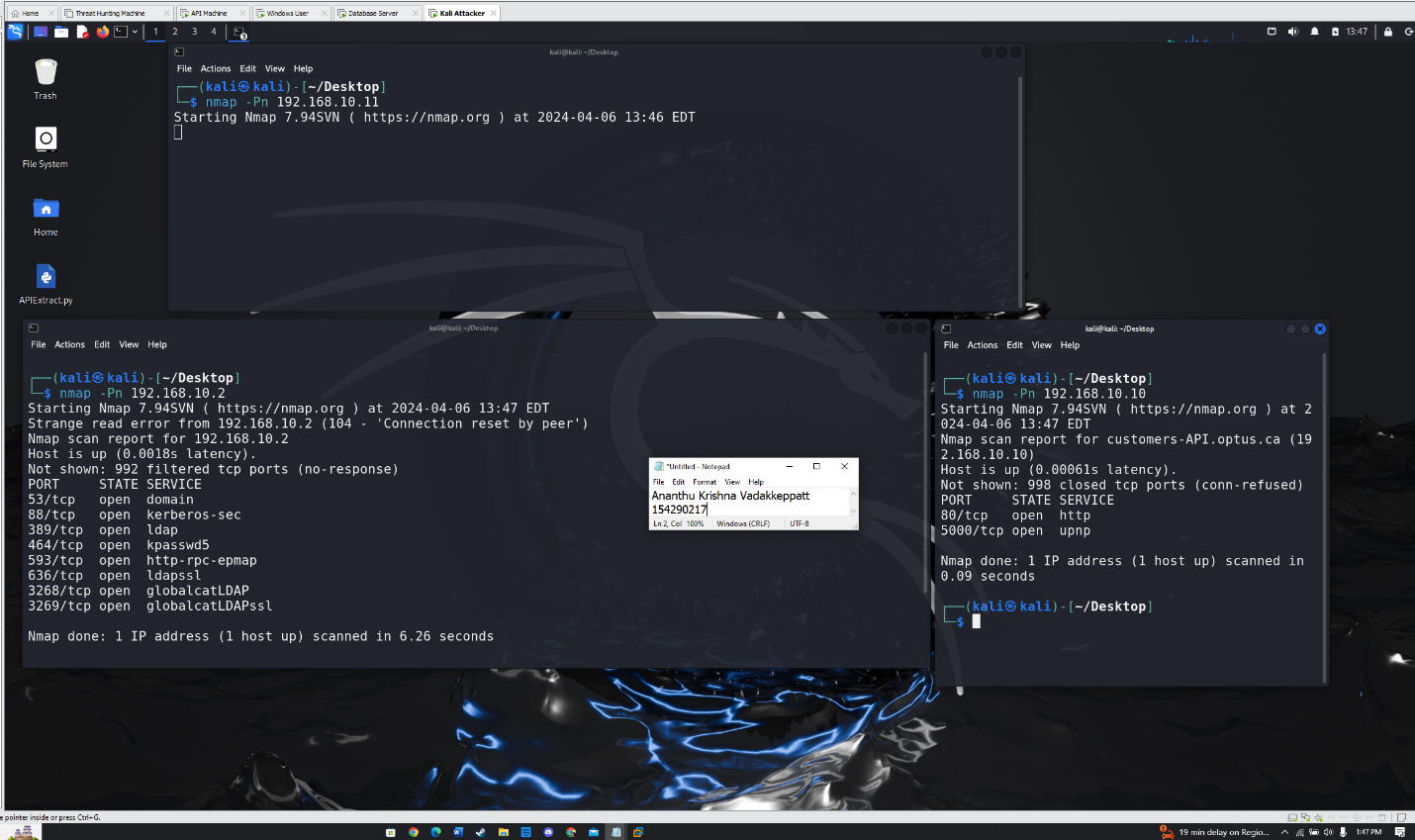
# **Attack Methodology**

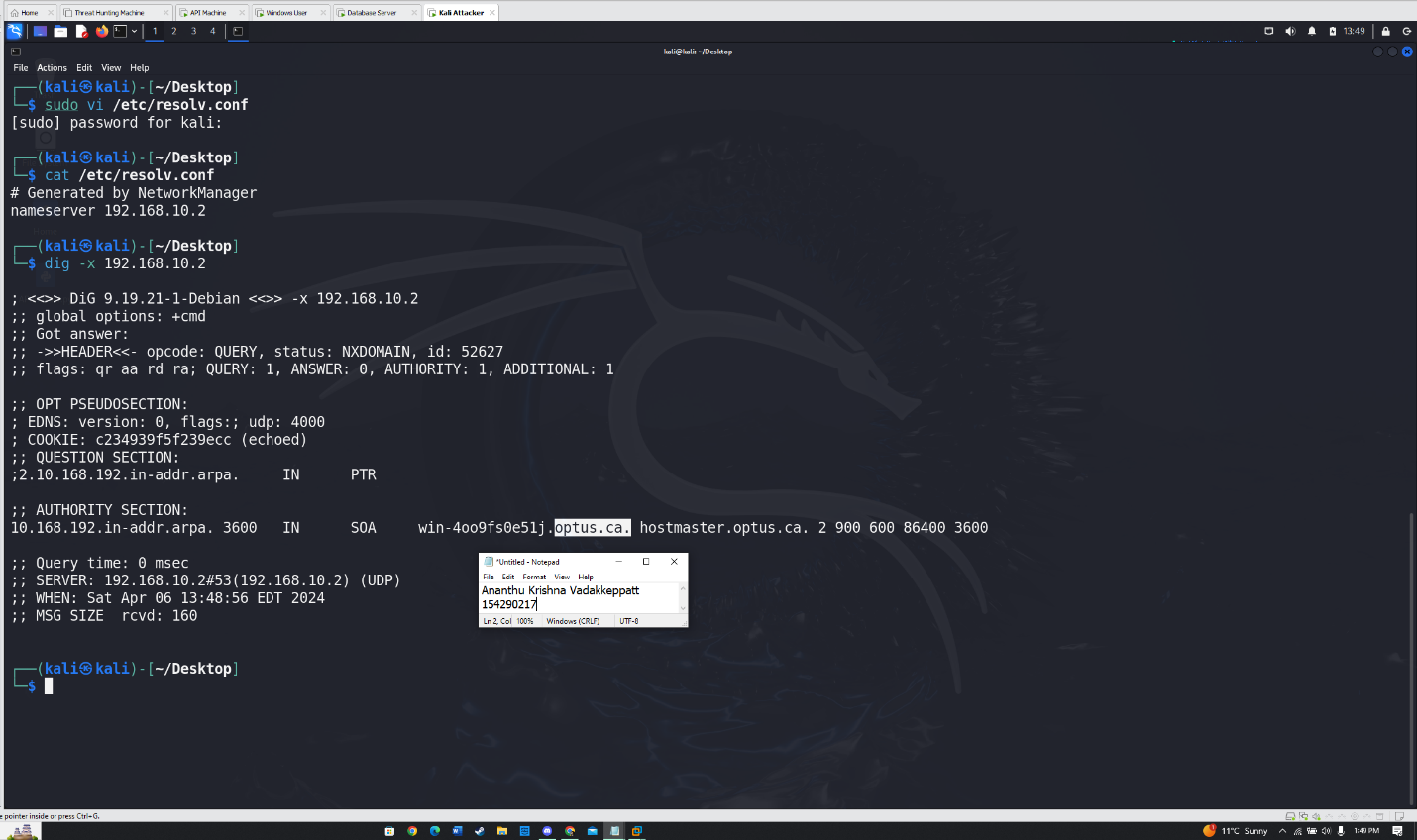
This replica of the attack focuses on bringing in the core concepts identified during the attack. The vulnerable API was set up using a flask framework script and the database was set using MySQL server. As the API was public facing, we assume that the attacker was already able to interact with infrastructure. In this setup, we have used the Arp scan module to identify the hosts within the environment. Furthermore, after identifying the hosts I have run Nmap commands to identify the ports as well as the services running on the same. Out of the three hosts one of them does not respond to Nmap’s calls mostly due to firewalls. While the other two hosts host two web applications and a domain. Hence then my attention shifts towards the open webservices that are found. I have made use of the Dirb tool that is used to scan online directories. The first web application that was seen to run on port 80 was found to be a default Apache webpage. However, the second web app that is found running on port 5000 is seen to be an open API that exposes sensitive credentials of the customers. To exfiltrate the information contained within the vulnerable API I was able to extract this data from the endpoint using a python script

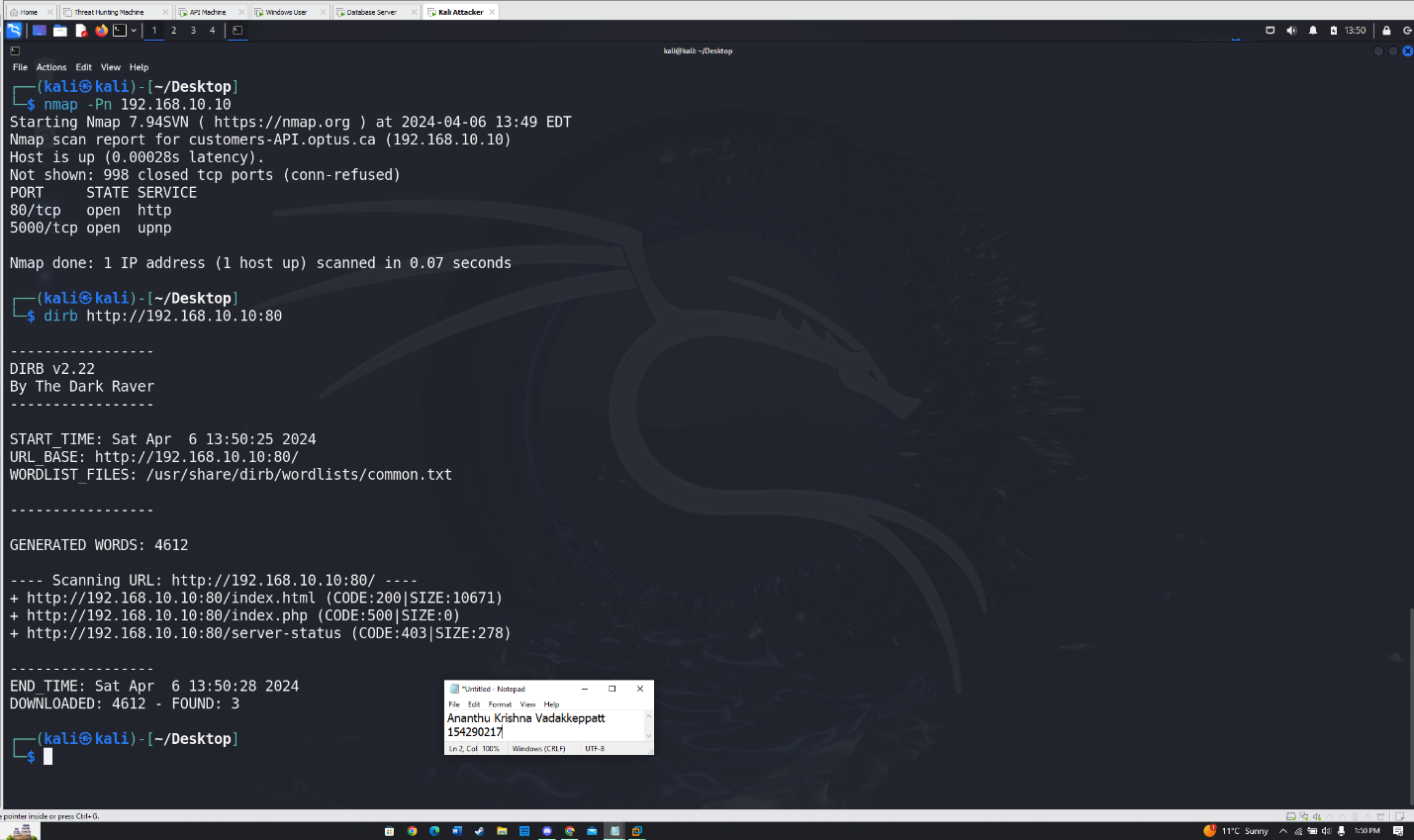
# **Demonstration of The Attack**

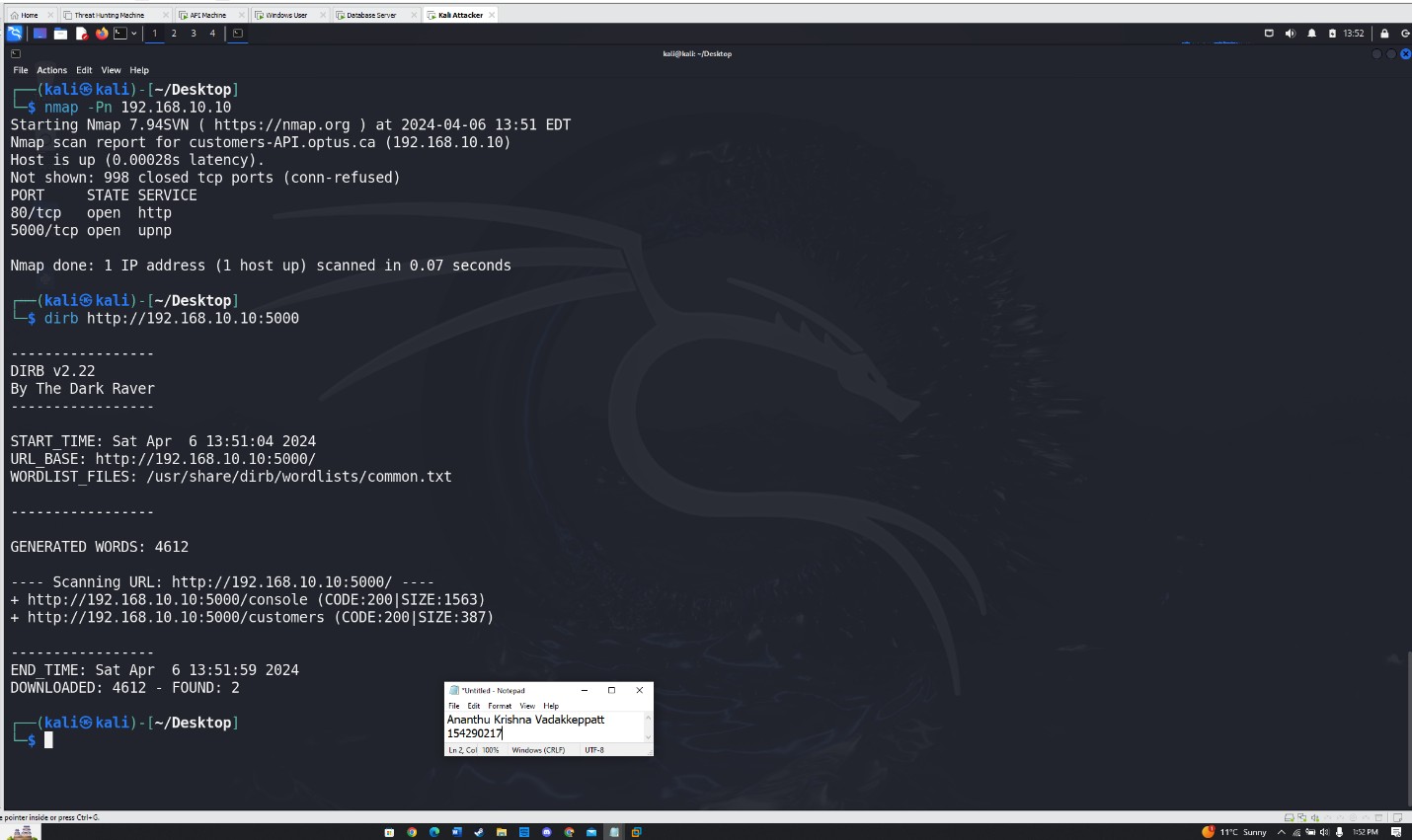
Fig.1 IP Address used by the Attacker.

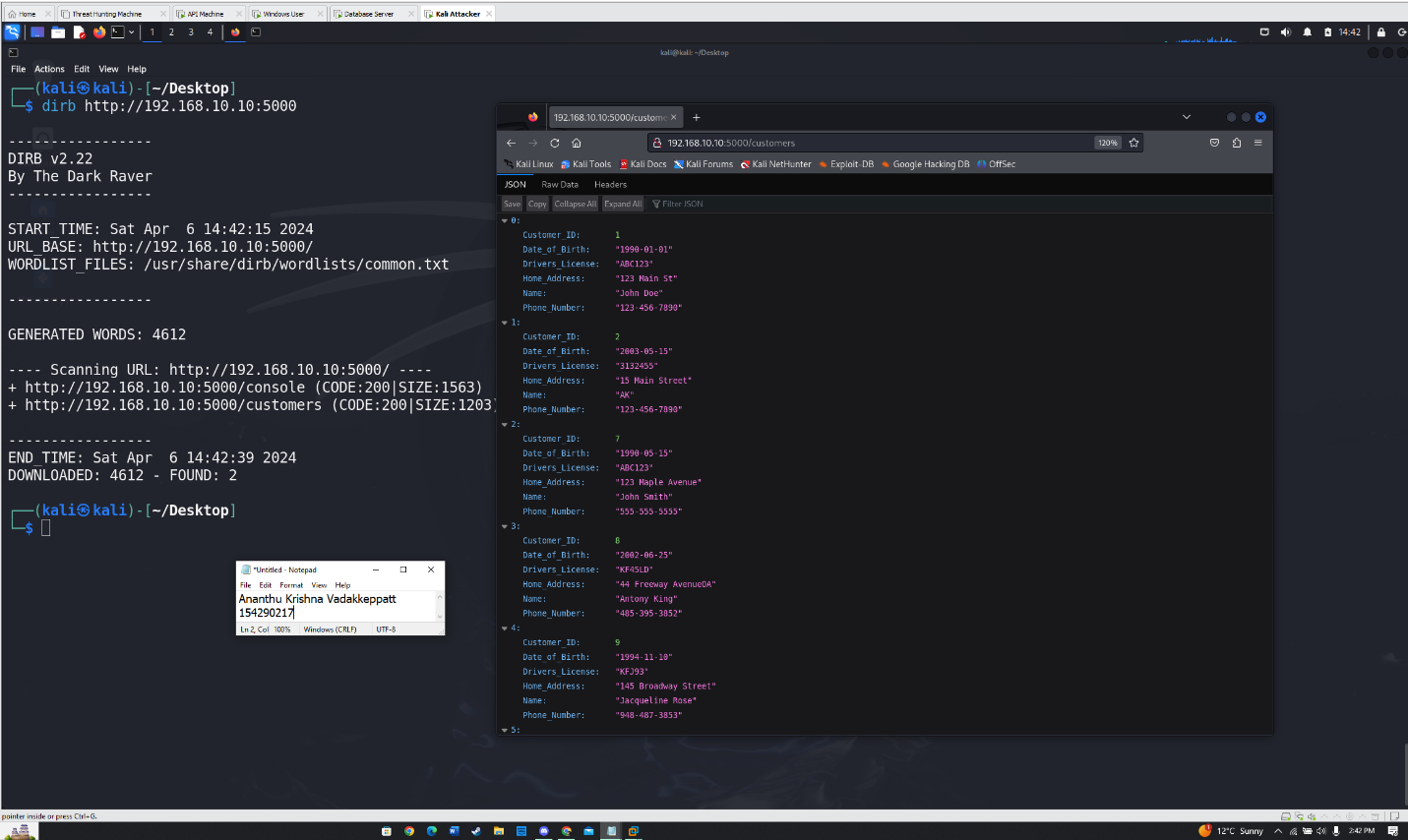
Fig.2 Arp Scan done withing the network.

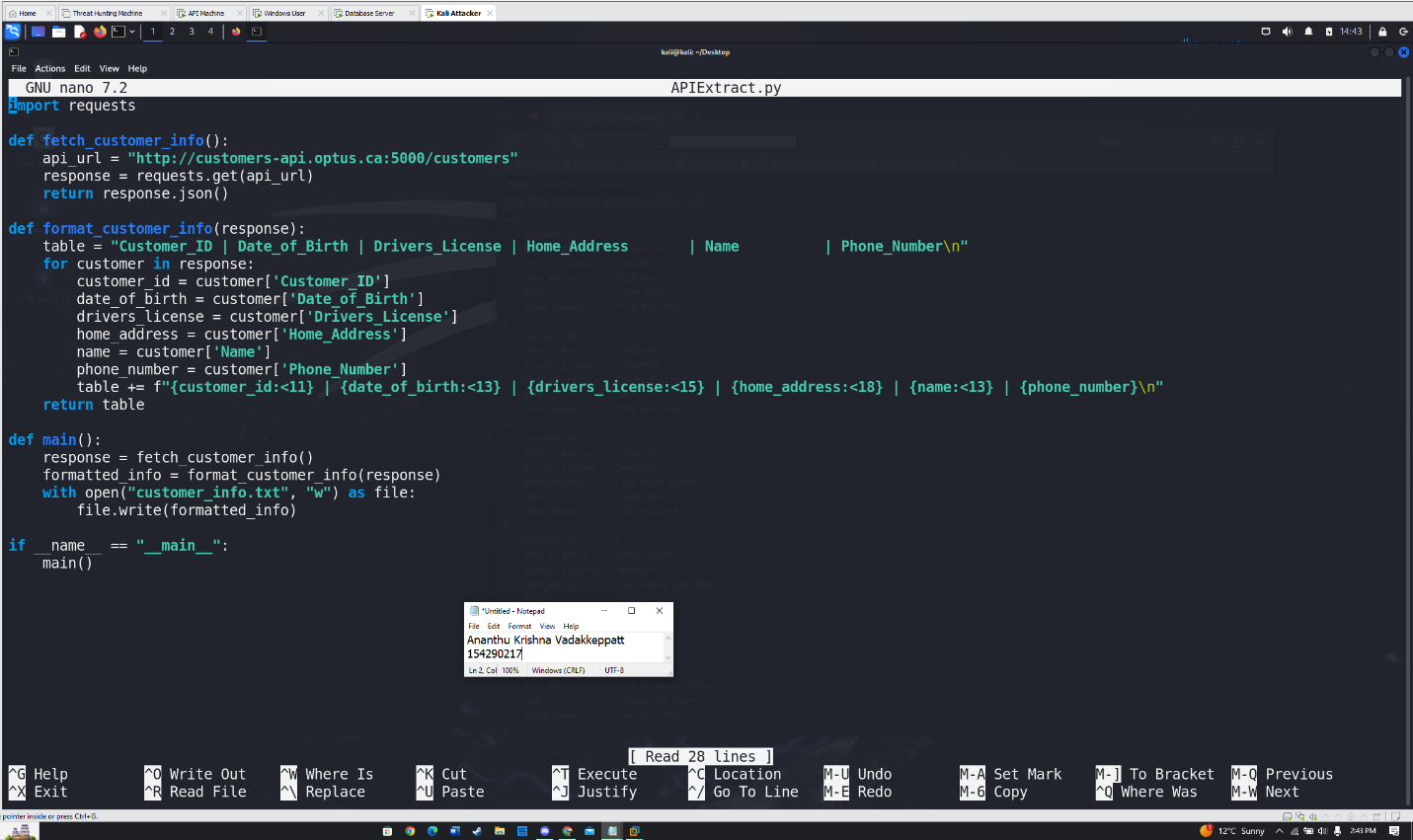
Fig.3 Nmap Scans done on all the discovered Hosts.

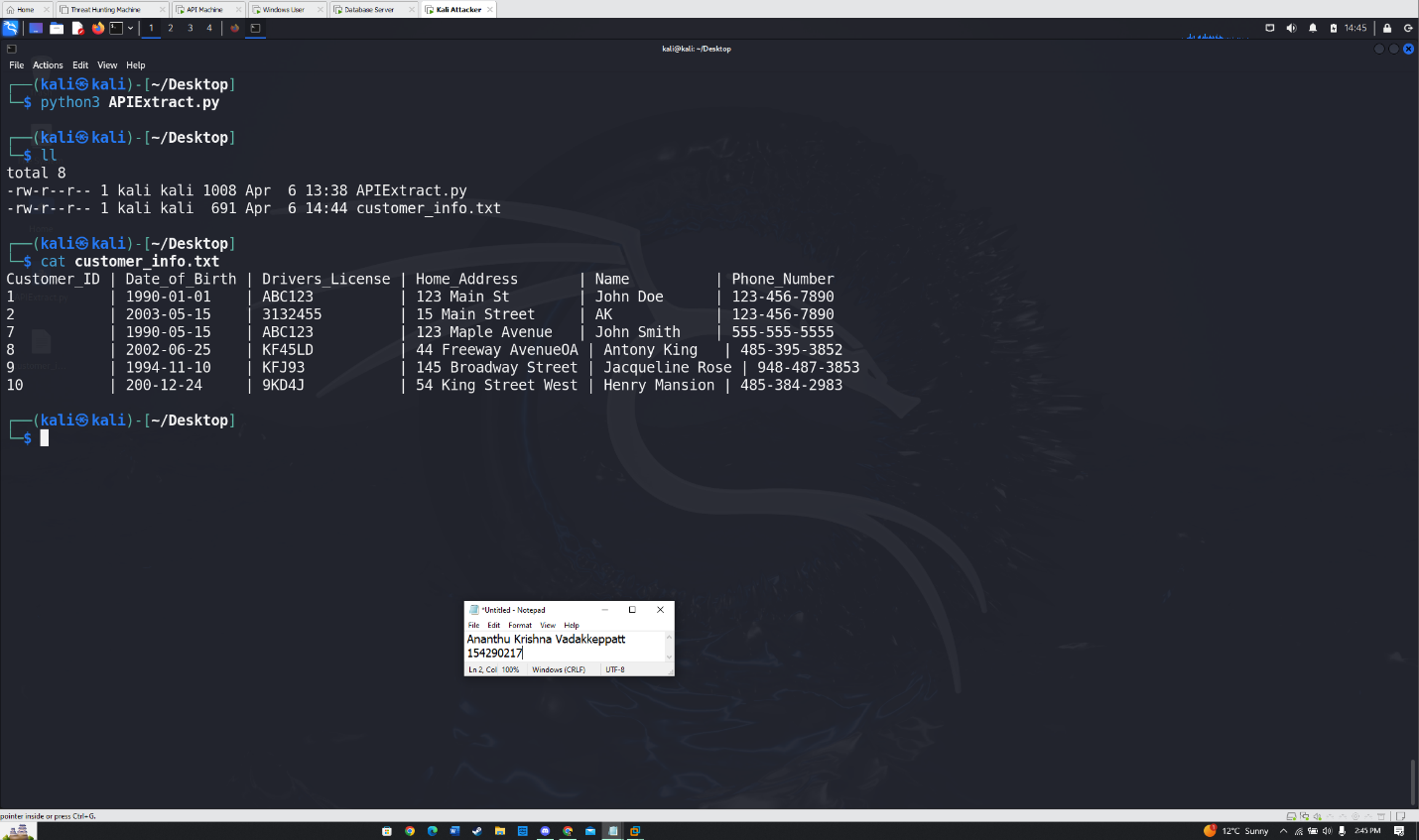
Fig.4 Dig command used to discover the domain.

Fig.5 Traversing the webpage for hidden pages on Port 80.

Fig.6 Traversing the website for hidden directories on port 5000.

Fig.7 Able to access the open API from the attacker’s machine.

Fig.8 Script used to exfiltrate the data from the API.

Fig.9 Executing the python script and writing the data to the file on the attacker’s machine

# **References**

1. *How did the Optus data breach happen? | UpGuard*. (n.d.-c). <https://www.upguard.com/blog/how-did-the-optus-data-breach-happen>
2. Security in mind. (2022, June 27). *Lets talk WireShark: Dirb scan - Traffic Analysis* [Video]. YouTube. <https://www.youtube.com/watch?v=zXG3HS0Nh88>
3. Keary, T., & Keary, T. (2023b, December 29). *Wireshark Cheat Sheet – Commands, captures, Filters & shortcuts*. Comparitech. <https://www.comparitech.com/net-admin/wireshark-cheat-sheet/>
4. Dyouri, A. (2022, December 20). *How to make a web application using Flask in Python 3*. DigitalOcean. <https://www.digitalocean.com/community/tutorials/how-to-make-a-web-application-using-flask-in-python-3>
5. *arp-scan | Kali Linux Tools*. (n.d.). Kali Linux. <https://www.kali.org/tools/arp-scan/>
6. *dirb | Kali Linux Tools*. (n.d.). Kali Linux. <https://www.kali.org/tools/dirb/>