Project Proposal

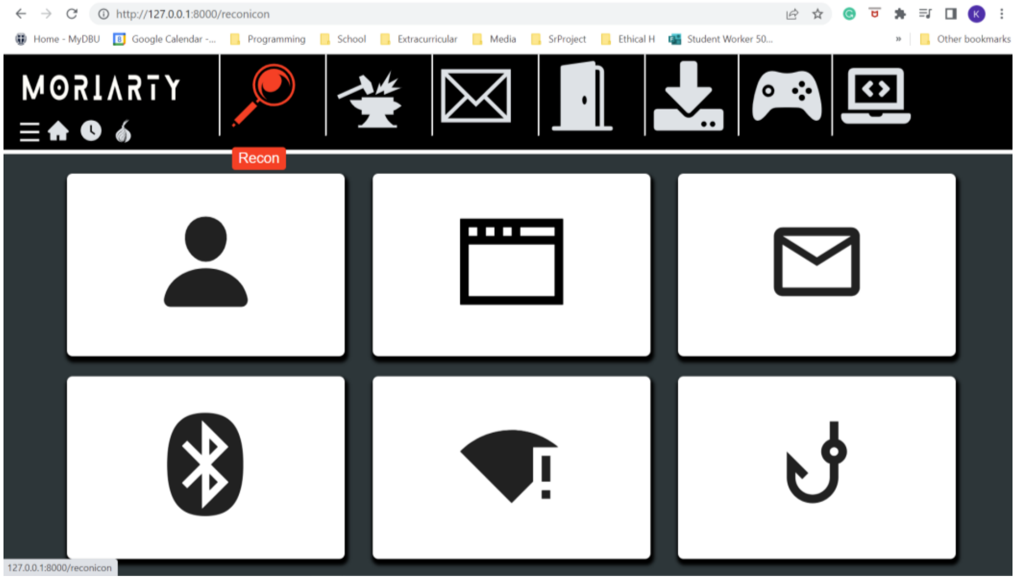
Project Title: Moriarty's Matrix (M&M)

1. Introduction

Opening Paragraph

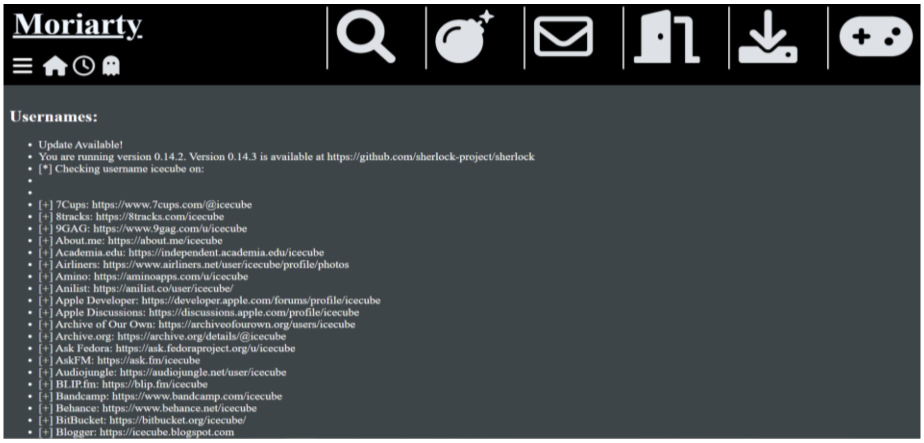
- **What is the project?** Moriarty Matrix is an offensive-focused educational hacking web-app (localhost) experience based on Lockheed Martin's Cyber Kill Chain, specifically:

  - Reconnaissance:



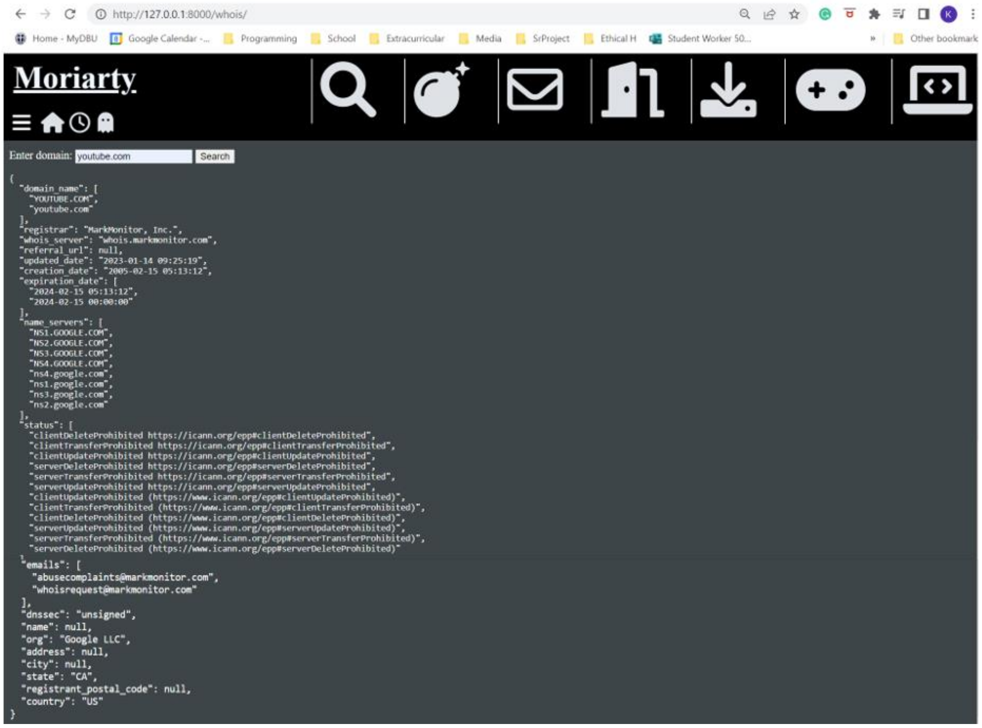
Some Recon Examples are below:

- **Username** (Enters username – outputs urls to accounts with that username, example "icecube")



```
Usernames:

  • Update Available!
  • You are running version 0.14.2. Version 0.14.3 is available at https://github.com/sherlock-project/sherlock
  • [*] Checking username icecube on:
  •
  •
  • [+] 7Cups: https://www.7cups.com/@icecube
  • [+] 8tracks: https://8tracks.com/icecube
  • [+] 9GAG: https://www.9gag.com/u/icecube
  • [+] About.me: https://about.me/icecube
  • [+] Academia.edu: https://independent.academia.edu/icecube
  • [+] Airliners: https://www.airliners.net/user/icecube/profile/photos
  • [+] Amino: https://aminoapps.com/u/icecube
  • [+] Anilist: https://anilist.co/user/icecube/
  • [+] Apple Developer: https://developer.apple.com/forums/profile/icecube
  • [+] Apple Discussions: https://discussions.apple.com/profile/icecube
  • [+] Archive of Our Own: https://archiveofourown.org/users/icecube
  • [+] Archive.org: https://archive.org/details/@icecube
  • [+] Ask Fedora: https://ask.fedoraproject.org/u/icecube
  • [+] AskFM: https://ask.fm/icecube
  • [+] Audiojungle: https://audiojungle.net/user/icecube
  • [+] BLIP.fm: https://blip.fm/icecube
  • [+] Bandcamp: https://www.bandcamp.com/icecube
  • [+] Behance: https://www.behance.net/icecube
  • [+] BitBucket: https://bitbucket.org/icecube/
  • [+] Blogger: https://icecube.blogspot.com
```

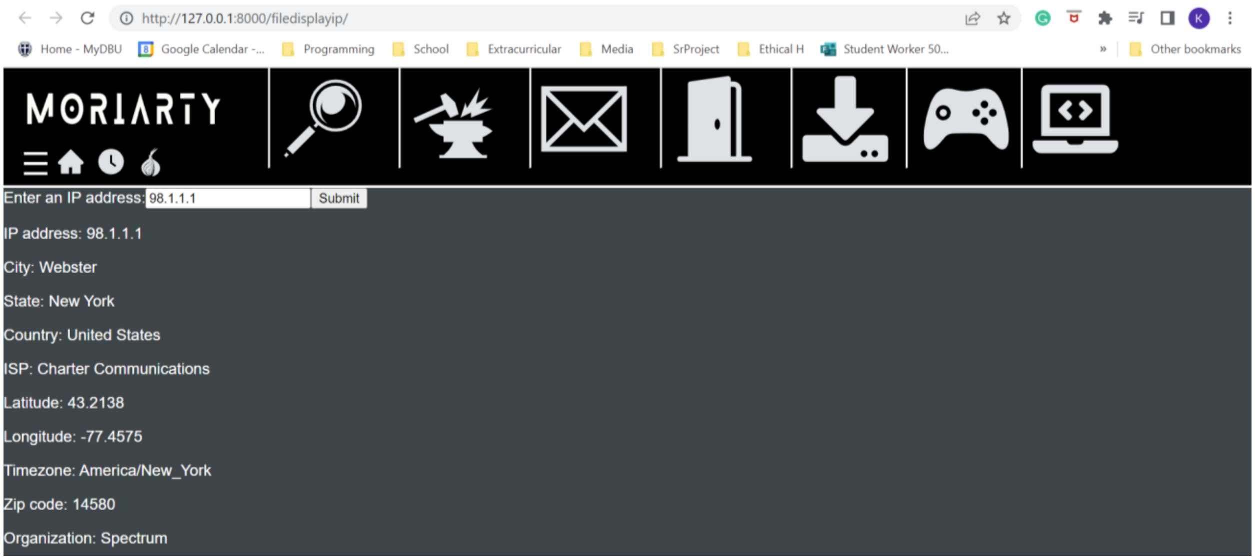
- **Domain**



Enter domain: youtube.com Search

```
{
  "domain_name": [
    "YOUTUBE.COM",
    "youtube.com"
  ],
  "registrar": "MarkMonitor, Inc.",
  "whois_server": "whois.markmonitor.com",
  "referral_url": null,
  "updated_date": "2023-01-14 09:25:19",
  "creation_date": "2005-02-15 05:13:12",
  "expiration_date": [
    "2024-02-15 05:13:12",
    "2024-02-15 00:00:00"
  ],
  "name_servers": [
    "NS1.GOOGLE.COM",
    "NS2.GOOGLE.COM",
    "NS3.GOOGLE.COM",
    "NS4.GOOGLE.COM",
    "ns4.google.com",
    "ns1.google.com",
    "ns3.google.com",
    "ns2.google.com"
  ],
  "status": [
    "clientDeleteProhibited https://icann.org/epp#clientDeleteProhibited",
    "clientTransferProhibited https://icann.org/epp#clientTransferProhibited",
    "clientUpdateProhibited https://icann.org/epp#clientUpdateProhibited",
    "serverDeleteProhibited https://icann.org/epp#serverDeleteProhibited",
    "serverTransferProhibited https://icann.org/epp#serverTransferProhibited",
    "serverUpdateProhibited https://icann.org/epp#serverUpdateProhibited",
    "clientUpdateProhibited (https://www.icann.org/epp#clientUpdateProhibited)",
    "clientTransferProhibited (https://www.icann.org/epp#clientTransferProhibited)",
    "clientDeleteProhibited (https://www.icann.org/epp#clientDeleteProhibited)",
    "serverUpdateProhibited (https://www.icann.org/epp#serverUpdateProhibited)",
    "serverTransferProhibited (https://www.icann.org/epp#serverTransferProhibited)",
    "serverDeleteProhibited (https://www.icann.org/epp#serverDeleteProhibited)"
  ],
  "emails": [
    "abusecomplaints@markmonitor.com",
    "whoisrequest@markmonitor.com"
  ],
  "dnssec": "unsigned",
  "name": null,
  "org": "Google LLC",
  "address": null,
  "city": null,
  "state": "CA",
  "registrant_postal_code": null,
  "country": "US"
}
```

- IP



Enter an IP address: 98.1.1.1 Submit

IP address: 98.1.1.1

City: Webster

State: New York

Country: United States

ISP: Charter Communications

Latitude: 43.2138

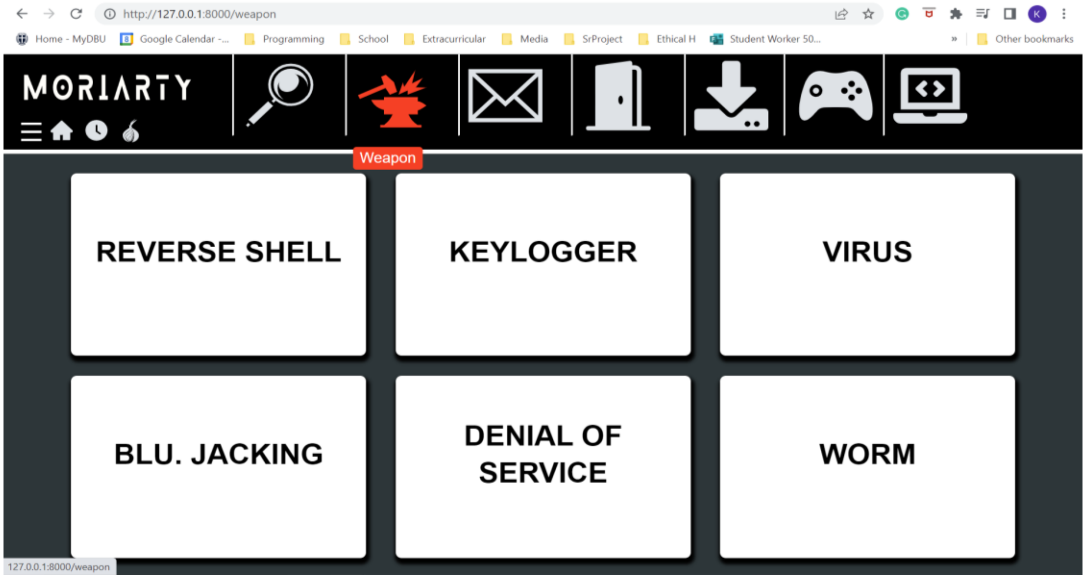
Longitude: -77.4575

Timezone: America/New_York

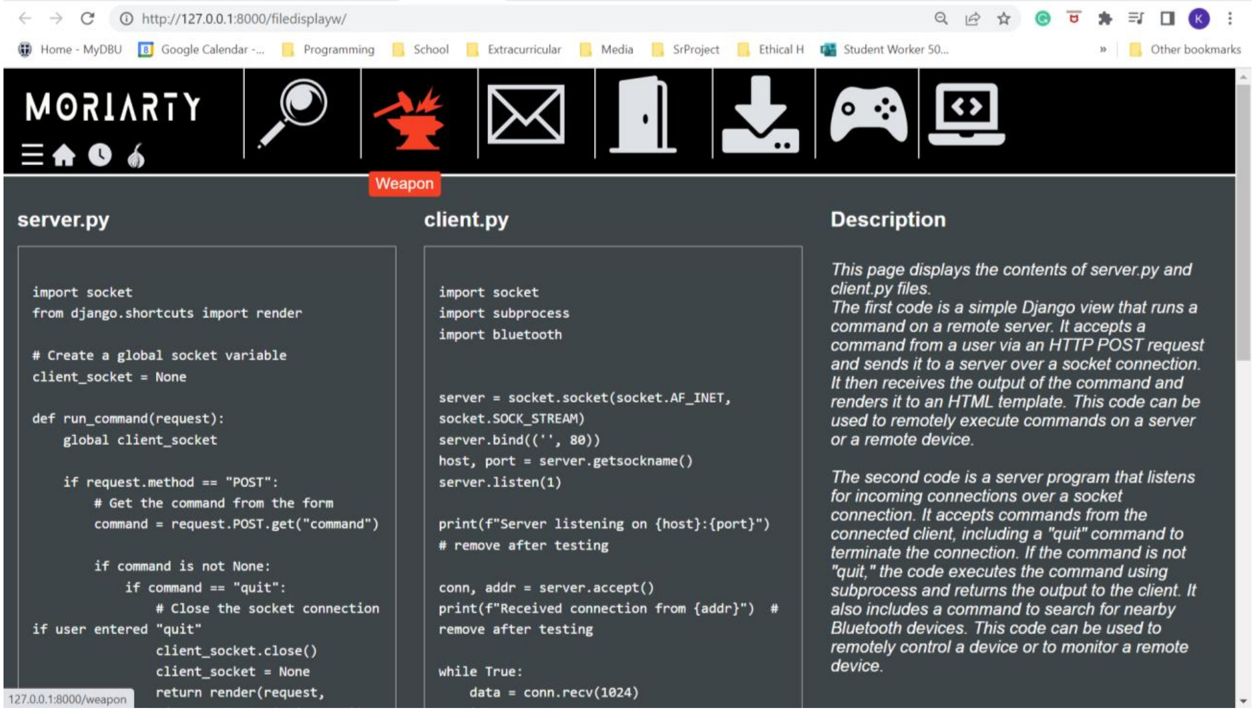
Zip code: 14580

Organization: Spectrum

- Weaponization:



- Reverse Shell



server.py

```
import socket
from django.shortcuts import render

# Create a global socket variable
client_socket = None

def run_command(request):
    global client_socket

    if request.method == "POST":
        # Get the command from the form
        command = request.POST.get("command")

        if command is not None:
            if command == "quit":
                # Close the socket connection
if user entered "quit"
                client_socket.close()
                client_socket = None
                return render(request,
```

client.py

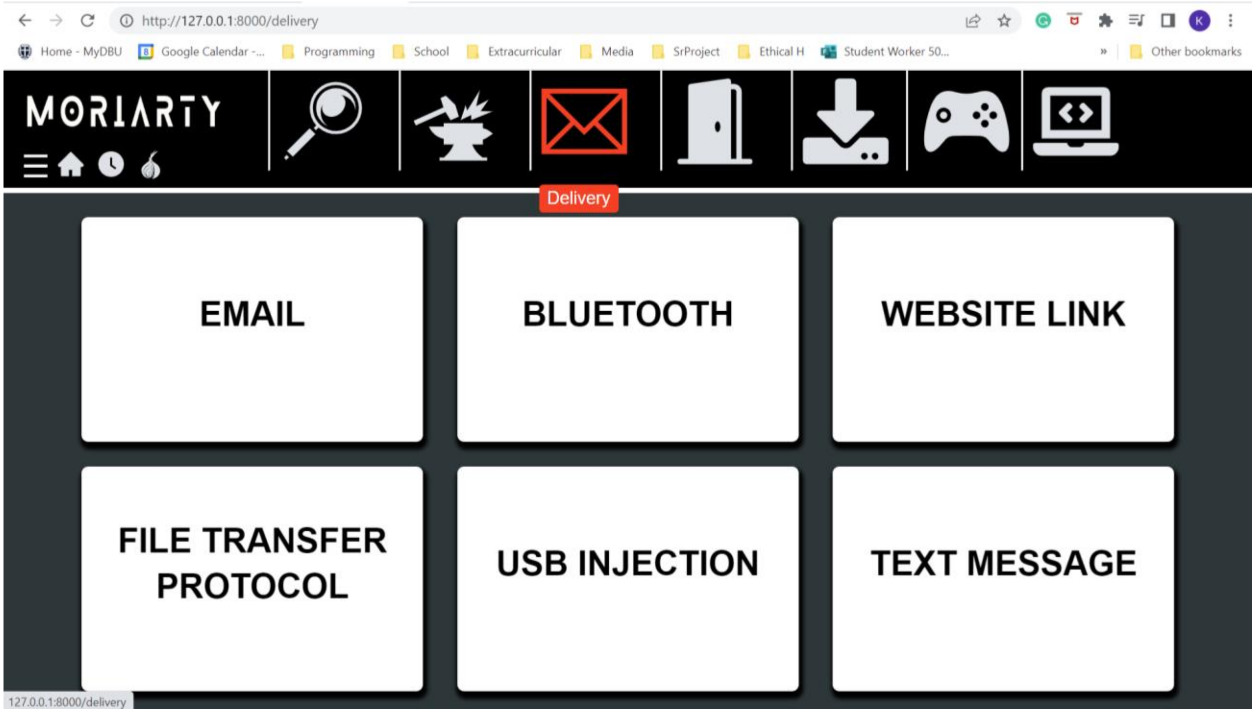
```
import socket
import subprocess
import bluetooth


server = socket.socket(socket.AF_INET,
socket.SOCK_STREAM)
server.bind(('', 80))
host, port = server.getsockname()
server.listen(1)

print(f"Server listening on {host}:{port}")
# remove after testing

conn, addr = server.accept()
print(f"Received connection from {addr}")  #
remove after testing

while True:
    data = conn.recv(1024)
```

Description

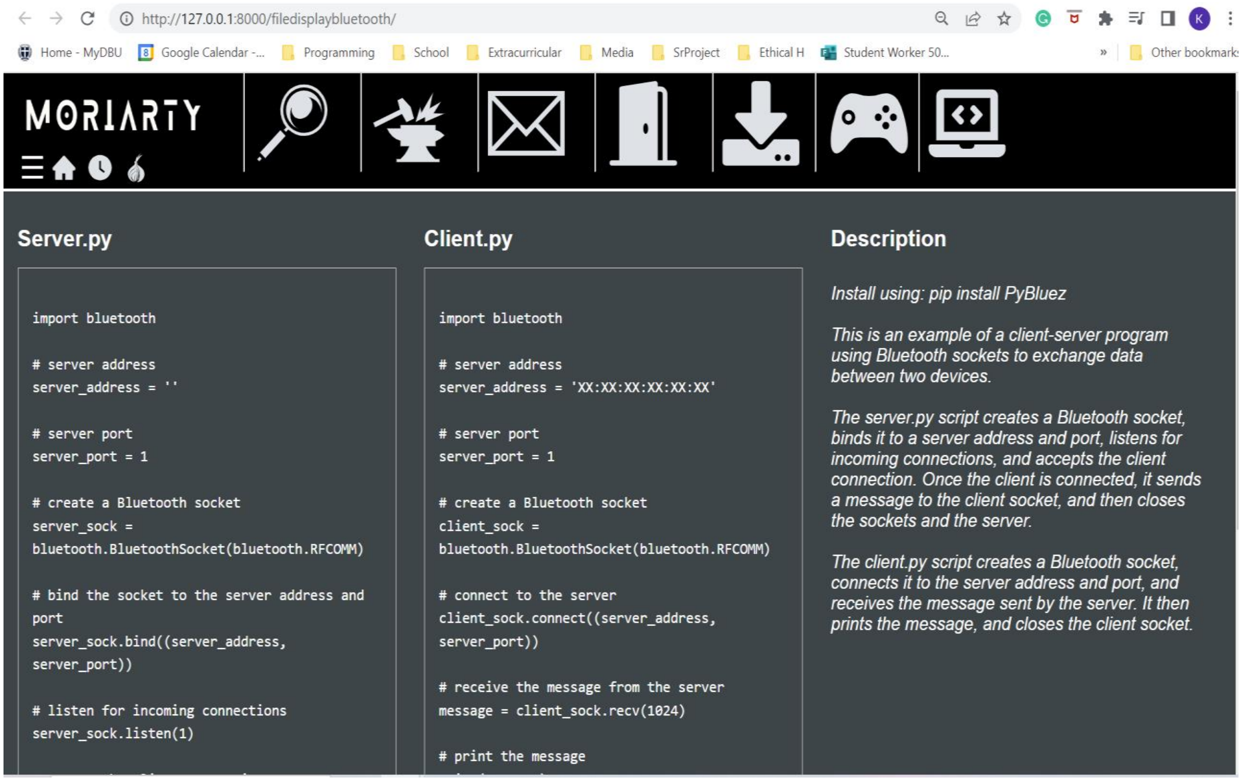
This page displays the contents of server.py and client.py files. The first code is a simple Django view that runs a command on a remote server. It accepts a command from a user via an HTTP POST request and sends it to a server over a socket connection. It then receives the output of the command and renders it to an HTML template. This code can be used to remotely execute commands on a server or a remote device.

The second code is a server program that listens for incoming connections over a socket connection. It accepts commands from the connected client, including a "quit" command to terminate the connection. If the command is not "quit," the code executes the command using subprocess and returns the output to the client. It also includes a command to search for nearby Bluetooth devices. This code can be used to remotely control a device or to monitor a remote device.

- Delivery:



- **Bluetooth**



Server.py

```
import bluetooth

# server address
server_address = ''

# server port
server_port = 1

# create a Bluetooth socket
server_sock =
bluetooth.BluetoothSocket(bluetooth.RFCOMM)

# bind the socket to the server address and
port
server_sock.bind((server_address,
server_port))

# listen for incoming connections
server_sock.listen(1)
```

Client.py

```
import bluetooth

# server address
server_address = 'XX:XX:XX:XX:XX:XX'

# server port
server_port = 1

# create a Bluetooth socket
client_sock =
bluetooth.BluetoothSocket(bluetooth.RFCOMM)

# connect to the server
client_sock.connect((server_address,
server_port))

# receive the message from the server
message = client_sock.recv(1024)

# print the message
```

Description

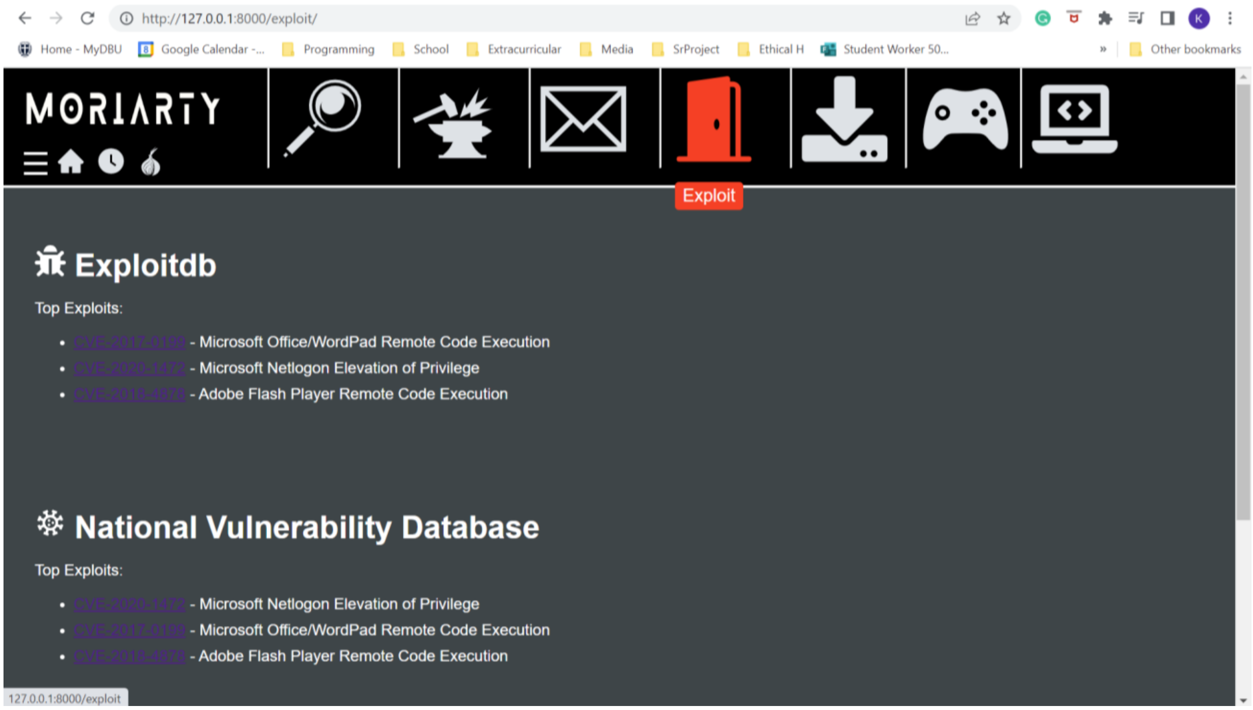
Install using: pip install PyBluez

This is an example of a client-server program using Bluetooth sockets to exchange data between two devices.

The server.py script creates a Bluetooth socket, binds it to a server address and port, listens for incoming connections, and accepts the client connection. Once the client is connected, it sends a message to the client socket, and then closes the sockets and the server.

The client.py script creates a Bluetooth socket, connects it to the server address and port, and receives the message sent by the server. It then prints the message, and closes the client socket.

- Exploitation:



Exploitdb

Top Exploits:

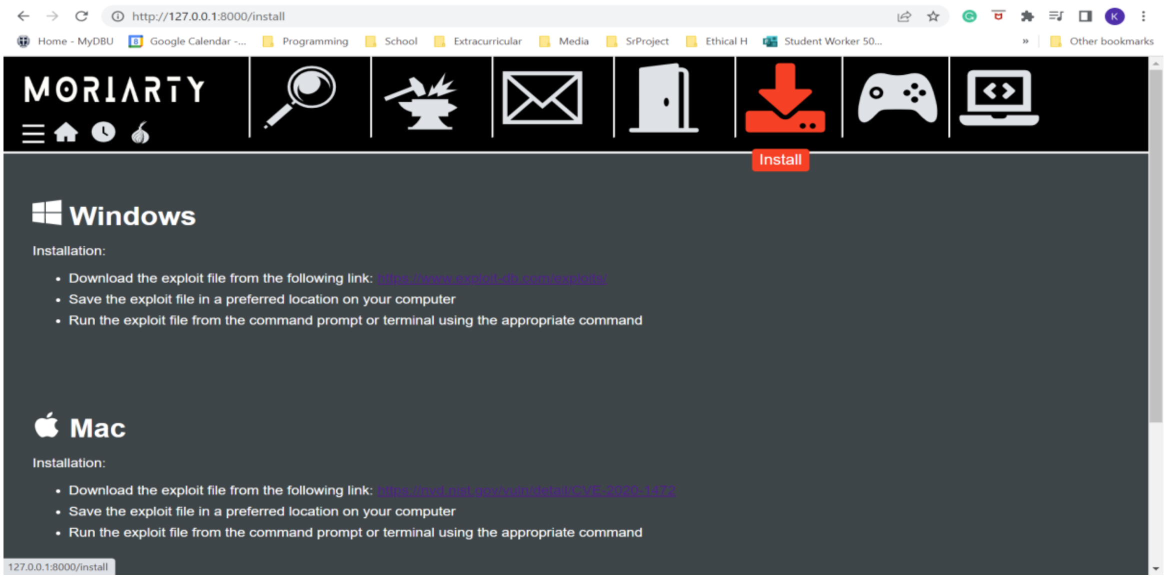
- CVE-2017-0199 - Microsoft Office/WordPad Remote Code Execution
- CVE-2020-1472 - Microsoft Netlogon Elevation of Privilege
- CVE-2018-4878 - Adobe Flash Player Remote Code Execution

National Vulnerability Database

Top Exploits:

- CVE-2020-1472 - Microsoft Netlogon Elevation of Privilege
- CVE-2017-0199 - Microsoft Office/WordPad Remote Code Execution
- CVE-2018-4878 - Adobe Flash Player Remote Code Execution

- Installation:



Windows

Installation:

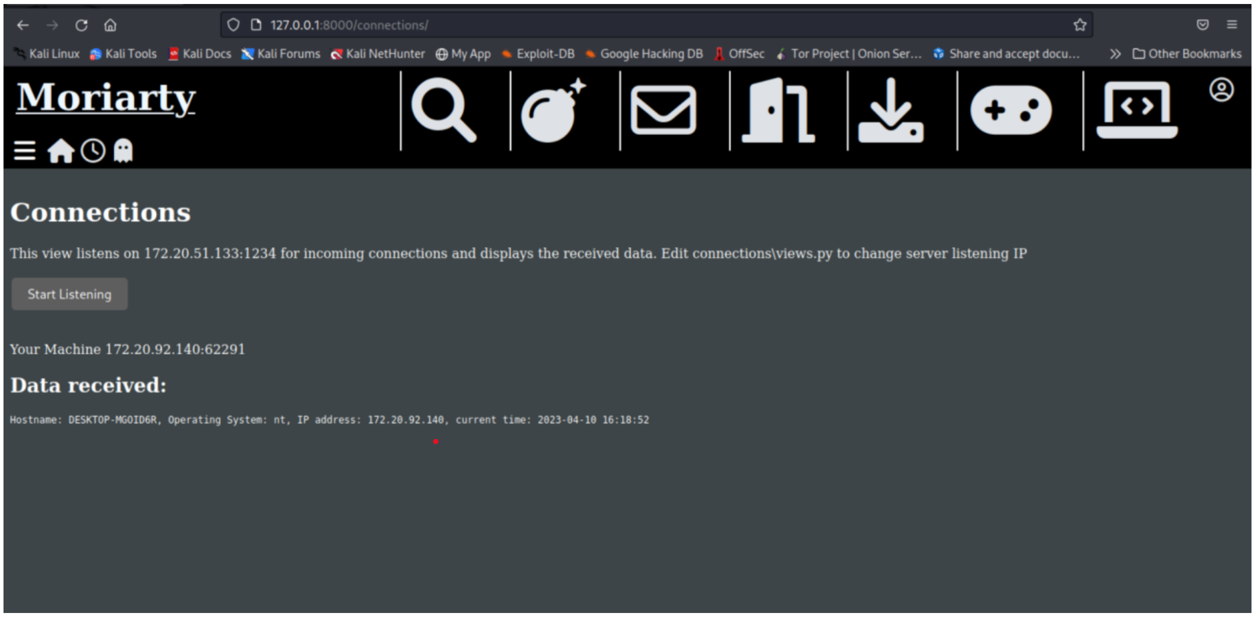
- Download the exploit file from the following link: https://www.exploit-db.com/exploits/
- Save the exploit file in a preferred location on your computer
- Run the exploit file from the command prompt or terminal using the appropriate command

Mac

Installation:

- Download the exploit file from the following link: https://nvd.nist.gov/vuln/detail/CVE-2020-1472
- Save the exploit file in a preferred location on your computer
- Run the exploit file from the command prompt or terminal using the appropriate command

- Command and Control:



Connections

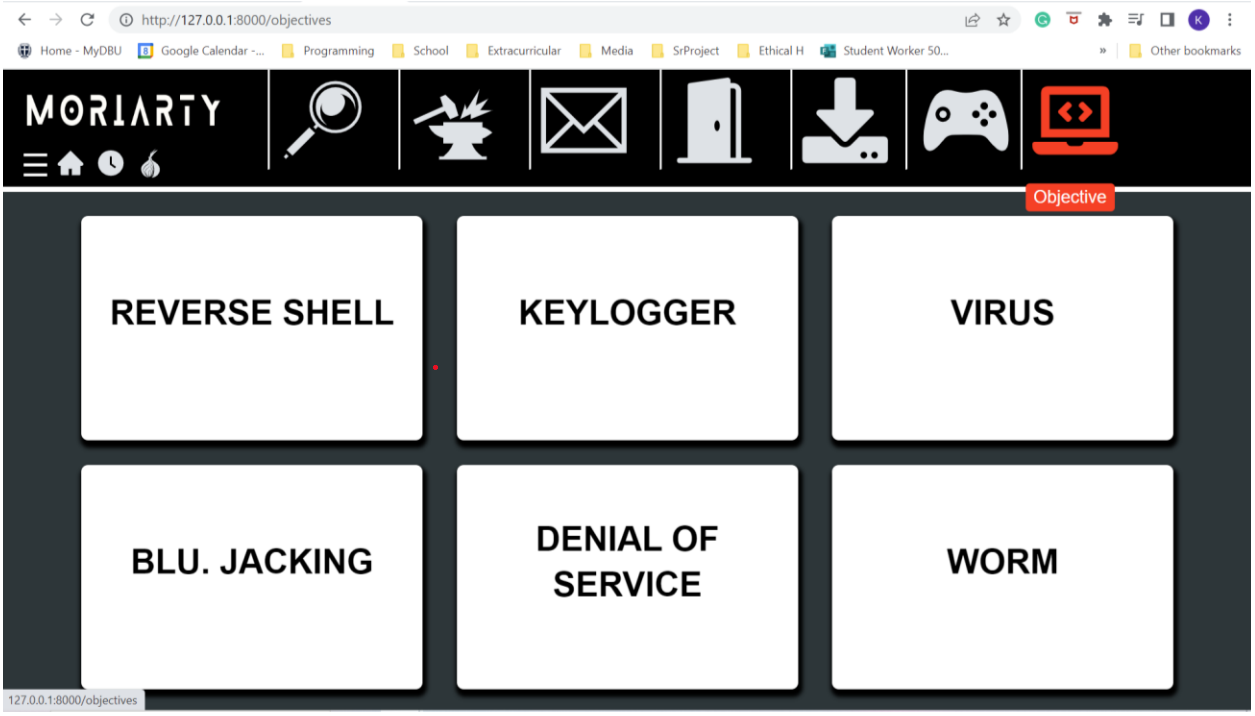
This view listens on 172.20.51.133:1234 for incoming connections and displays the received data. Edit connections\views.py to change server listening IP

Your Machine 172.20.92.140:62291

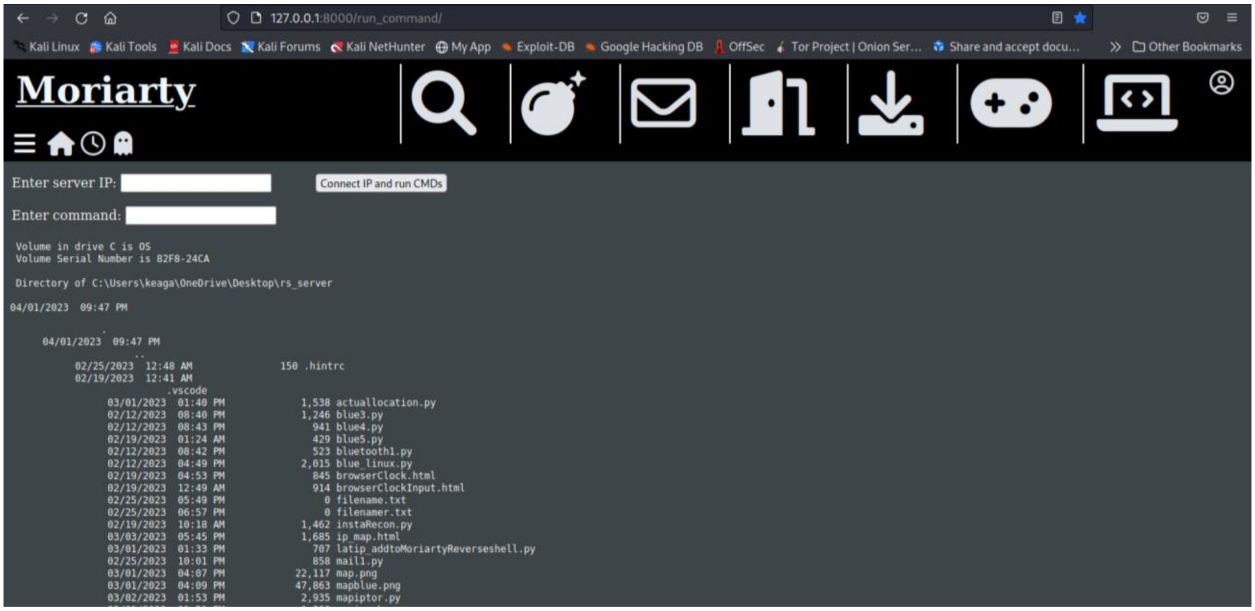
Data received:

```
Hostname: DESKTOP-MGOID6R, Operating System: nt, IP address: 172.20.92.140, current time: 2023-04-10 16:18:52
```

- Actions on Objectives:



- Reverse Shell



Enter server IP:

Enter command:

```
 Volume in drive C is OS
 Volume Serial Number is 82F8-24CA

 Directory of C:\Users\keaga\OneDrive\Desktop\rs_server

04/01/2023  09:47 PM

                .
     04/01/2023  09:47 PM
                     ..
          02/25/2023  12:48 AM              150 .hintrc
          02/19/2023  12:41 AM
                         .vscode
               03/01/2023  01:40 PM            1,538 actuallocation.py
               02/12/2023  08:40 PM            1,246 blue3.py
               02/12/2023  08:43 PM              941 blue4.py
               02/19/2023  01:24 AM              429 blue5.py
               02/12/2023  08:42 PM              523 bluetooth1.py
               02/12/2023  04:49 PM            2,015 blue_linux.py
               02/19/2023  04:53 PM              845 browserClock.html
               02/19/2023  12:49 AM              914 browserClockInput.html
               02/25/2023  05:49 PM                0 filename.txt
               02/25/2023  06:57 PM                0 filenamer.txt
               02/19/2023  10:18 AM            1,462 instaRecon.py
               03/03/2023  05:45 PM            1,685 ip_map.html
               03/01/2023  01:33 PM              707 latip_addtoMoriartyReverseshell.py
               02/25/2023  10:01 PM              858 mail1.py
               03/01/2023  04:07 PM           22,117 map.png
               03/01/2023  04:09 PM           47,863 mapblue.png
               03/02/2023  01:53 PM            2,935 mapiptor.py
```

Moreso, it has additional features which may be applicable to numerous other situations. Some features include:

- A map showing the user's current and past IP locations

- HamBar showing and explaining the different pages

-Home Menu (default starting page) shows practical ways how to use this app and example output

In addition to these localhost features, Moriarty Matrix will have a publicly hosted website explaining the project and allowing users to download from it (in addition to its GitHub repo, unless if it is a paid product then it will only be available for download from the website)

- **What is the motivation for this project?** To create awareness of the hacking-lifecycle so it better prepares students, employees, cyber-enthusiasts, and security testers. Moriarty Matrix is inclusive with a heavy use of icons in this project, reducing confusion for language barriers and helps with the community understanding. The instructive icons goal is to make it so that in Labs every one has a similar guidance for what to use.

- **What is novel about it?** It provides a sleek web-based user interface and user experience instead of the usual command line interface. With the command line interface it can be difficult to see the big picture and how everything works together, it can also be difficult to understand and time consuming. Moriarty's Matrix, allows a smooth transition in to this field, provides numerous tools, explains uses, and allows the user to be the demonstrator of the possible attack. Moreso, because of the relatable UI of Moriarty's Matrix, it allows people with little cyber experience to understand and perform attacks (for educational and training purposes only). Icons are used heavily in this project to reduce confusion for language barriers and helps with the community understanding.

- **Anything else to orient the reader?** This Moriarty Matrix (website) is for educational, training, use only. When this product is used, the blame does not fall on the creator or anyone who provided input to this project.

Challenges

- **Briefly describe the main challenges and how to address them.** There were numerous challenges to begin with, some of the main ones are:

  - Understanding how to create a full stack website:

  <u>Backend:</u> Django python, I had no prior experience with Django and handling the forms, views, settings, and how to design various URLs. I came into numerous errors when creating a URL for a React page but discovered the different ways to register it in the Backend. Frontend: React and vanilla html & JavaScript for certain features

  - Understanding how malware is created, obfuscated, and the actual code and tools used with this process.

  -Screen size adjustment using CSS, I mainly utilized @media and flexbox for this but some of the components also aided with re-sizing. On the other hand, the components can be more difficult with resizing (browser-px-adjustments) making me have to implement component specific styles, and when able to use CSS I also have to use !important for overriding styles. Furthermore, I had to make some unique React workarounds to help with the styling and transitions.

- Virtual Machines caused some issues throughout this project, with the IP address, MAC address, and some others like the Bluetooth adapter. However I was able to fix the IP address and MAC address issues, and would need to buy a separate device for Bluetooth as it is not compatible with my host machine.

- Throughout this project I been coding on different machines mainly Linux based distributions such as Kali Linux, on the DBU Wi-Fi I was unable to complete any steps using Kali Linux, so I had to use a VPN, otherwise even simple tasks such as ‘pip install’ would not download.

- With the Kali Linux distributions, problems arose not just because of the network I was on but because of the versions. Certain versions have special features such as pre-install python modules while others had protocols to not allow the installation of certain modules.

- Using GitHub, has been a great resource for this project, due to issue with Virtual Machines and corruptions of their OS’s, that way I can push and pull my project code from it no matter the location or if something is lost. GitHub and Vscode, were having some difficulties as I may be logged into one account on a machine that would have been deleted or removed, so I have been using various creative solutions through this cloud service and hope to use it consistently as a host for one of these web applications.

- When studying the hacker-mindset and learning about certain features or code, I discovered that while technology advances as fast as a goldfish forgets (or ages), many cyber security practices seem to escalate quickly but still have their foundations on the old. Meaning certain practices, code, and libraries, will be outdated however if you can figure out the techniques there will be a way to achieve the same result with added benefits.

- **Is the technology known or new to the team?** I have used some React before but creating URLs, using this much JavaScript, and all of the Django python was new to me.

2. **Customer/User Need**

- **Who is the primary customer/user?**
  - Universities & Colleges
  - Students
  - Professors
  - Companies (Demoing and allowing hands on explanations to business partners and fellow workers)
  - Cyber-enthusiasts

- **Who are the secondary stakeholders?** The secondary stakeholders may include IT security departments or security researchers who may use the tool for analysis or testing purposes. Moreso, it includes anyone who could potentially be impacted by a cybersecurity attack, such as customers, partners, and employees of the primary customers/users.

- **What do the stakeholders want? Why?** The stakeholders want an application that will help them better understand the hacking lifecycle and protect themselves and their organizations from potential threats. They also want a user-friendly interface that is easy to use and provides hands-on training opportunities. Furthermore, the stakeholders want an educational tool that is easy to use, visually appealing, and provides a clear understanding of how to defend against hacking attempts.

- **What is their desired overall experience?** The desired overall experience for the user is to gain knowledge and practical experience with hacking techniques in a safe and controlled environment.

User Requirements

- **Write SMART user stories based on the stakeholder needs and wants. Use the "As a ... I want ... so that ..." template.**
  - As a university student, I want to use Moriarty Matrix to learn and practice ethical hacking techniques so that I can better understand how to defend against them in the future. I want to use hands-on training to better prepare for potential cybersecurity threats, so that I can protect myself and my future organization.

  - As a professor, I want to use Moriarty Matrix in my classroom to provide hands-on training opportunities for my students. I want to use this tool to better prepare my students for potential cybersecurity threats, and to help them gain practical experience with hacking techniques in a safe and controlled environment.

  - As a company, I want to demo Moriarty Matrix to my business partners and fellow workers to showcase the potential cybersecurity threats that we face, and to better prepare our employees for potential cyber attacks. I want to use this tool to improve the overall security of our organization, and to provide a clear understanding of how to defend against hacking attempts.

  - As a cyber-enthusiast, I want to use Moriarty Matrix to gain knowledge and practical experience with hacking techniques in a safe and controlled environment. I want to use this tool to improve my skills and better prepare for potential cybersecurity threats, so that I can contribute to improving overall cybersecurity.

- **Write acceptance tests for the user stories, using the "Given ... when .... then ..." template.**

- Given an account as a student, for Moriarty Matrix (through localhost), when I access the application, then I should be able to navigate to the different stages of the Cyber Kill Chain and better understand the process of how to perform attack.

- Given an account as a professor, for Moriarty Matrix (through localhost), when I access the application, then I should be able to create and manage assignments for my students, view their progress, and provide feedback to help them improve their skills based on Moriarty Matrix.

- Given an account as a company, for Moriarty Matrix (through localhost), when I access the application, then I should be able to demo the tool to my business partners and fellow workers, and provide them with hands-on training opportunities to improve their overall cybersecurity skills.

- Given an account as a cyber-enthusiast, for Moriarty Matrix (through localhost), when I access the application, then I should be able to gain knowledge and practical experience with hacking techniques in a safe and controlled environment and improve my overall cybersecurity skills.

With each user understanding if anything goes wrong or if they abuse this technology the creator(s) and anyone involved is not to blame.

3. Project Goals

- **What customer problem have you chosen to address?** The customer problem addressed is the lack of a user-friendly educational tool that provides a clear understanding of the hacking lifecycle and how to defend against it. Furthermore, aims to address is the lack of understanding of the hacking lifecycle, particularly among those who are new to the field. Also creating original connections (some using premade libraries or resources, but using it is an original manner) for each of the seven parts of the Cyber Kill Chain.

- **In implementation-free terms, what user benefit will the system provide?** The system will provide a user-friendly interface that allows users to better understand the hacking lifecycle and perform hands-on training to prepare for potential cybersecurity threats.

- **How will the benefit support the customer's desired overall experience?** By providing a user-friendly interface and hands-on training opportunities, Moriarty Matrix will help users gain a better understanding of the hacking lifecycle, which will ultimately help them protect themselves and their organizations from potential threats.

Measures of Success

- **Who outside the team have you tested the idea on?** With their permission, I have tested it on students, professors, coworkers, and housemates such as having them go through the steps, use recon phases, etc…

- **How will you know whether the customer got their desired benefits?**

  To determine whether a new idea is viable, it is important to test it to receive feedback from potential customers. When testing the idea, the most useful feedback comes from people who represent the target market, as they can provide valuable insight into the product's value proposition.
  Overall, it should provide an insight to the user, simplify and demystify the offensive security (hacker's mindset), and make understanding the overall vulnerability identification and exploitation process better.

- **What are your customer-centric measures of success?**

The customer-centric measures of success for Moriarty Matrix include:

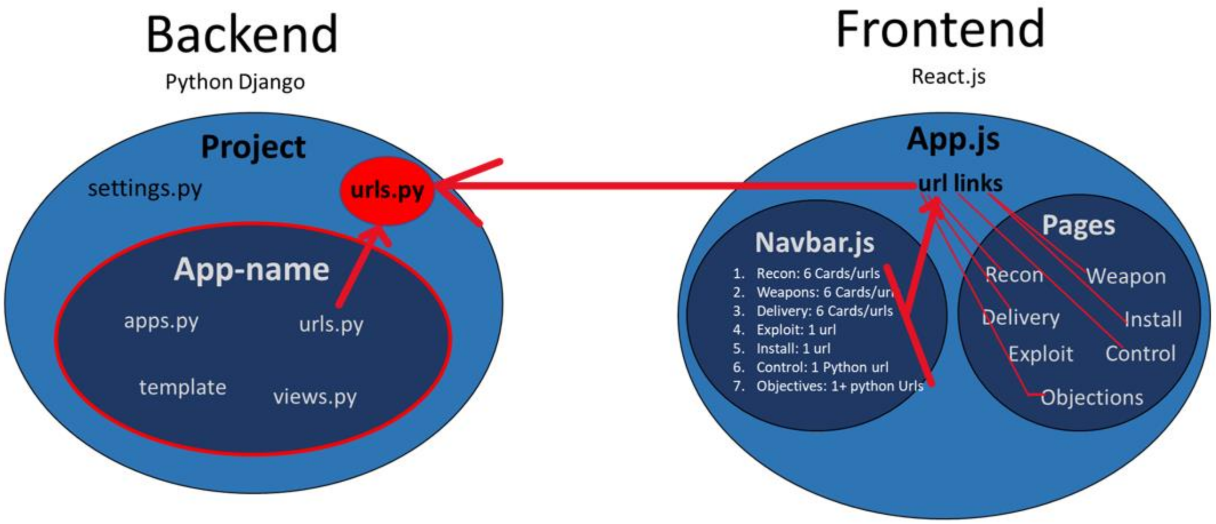
- Number of downloads and active users on the web app
- Interest to cyber it creates in someone
- User engagement and retention rates

- Positive feedback and reviews from universities, students, professors, companies, and cyber-enthusiasts
- Number of successful ethical hacking techniques performed using the tool.
- Number of IT security departments or security researchers who use the tool for analysis or testing purposes.
- Number of organizations who adopt Moriarty Matrix for employee training and education.

4. System Description

For this proposal, a rough draft of this section is enough.

- Draw a block diagram to show the proposed system will interact with external services, databases ... Clearly mark the boundaries of the system.



- Red Represents Every URL (both Backend & Frontend)
- All urls are stored in the Project urls.py at **mm_python\urls.py**
- Every Python App has its own url/page (at least one url per app)

- **Use the above diagram to introduce the system.** As shown on the diagram, red represents each URL/page (currently this project has 35 and counting). In the backend for Django, each Django design has only 1 Project and then can have countless apps inside. The Django Project acts as the "main function" everything has to pass through it and everything is dependent on the Project programs. The React application is inside of the Django Project and must also have a connection. Unlike Django Framework, React is more flexible with the uses of its pages.

- **What are the main elements of the proposed system?**
  The Main elements are Django and React. Within this web-application the most vital are React (App.js, Navbar.js, Pages Folder)

5. **Solution Approach**

   A brief rough draft of this section is enough for this proposal.

   - **Briefly describe how the system will work.** The Moriarty Matrix is an innovative system that can help in data analysis and decision-making. The system collects data from various sources and processes the information to generate insights It is essential to test the system's functionality to identify any issues or gaps. The team can evaluate the adequacy of their test strategy by analyzing the test results and feedback from customers.

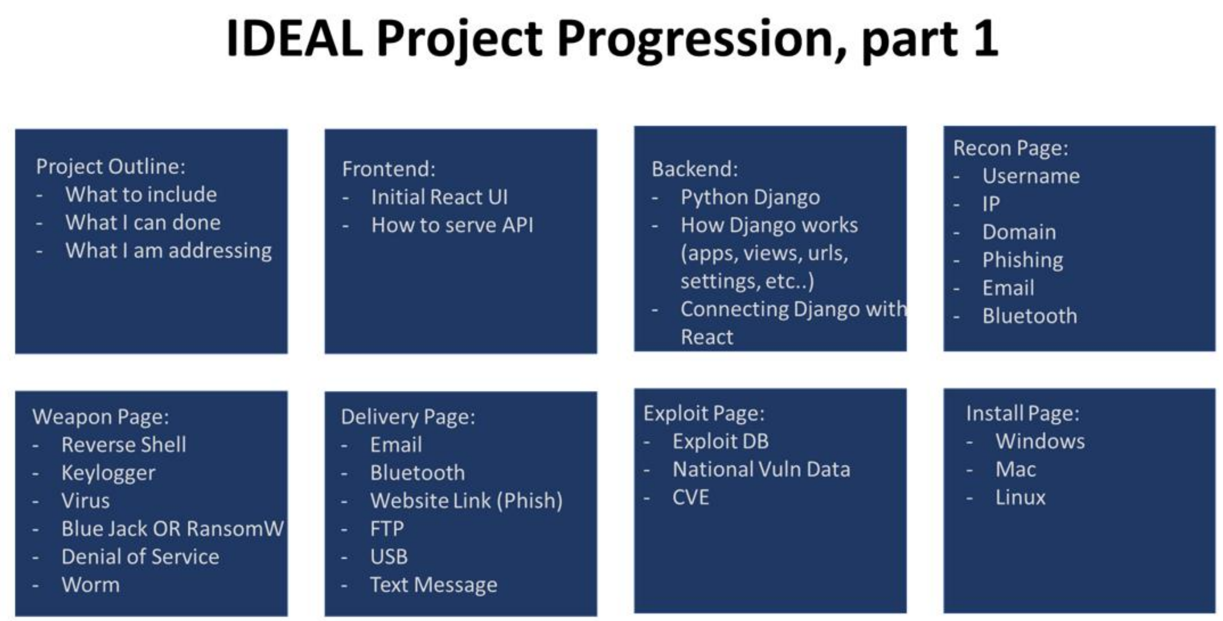
   - **What platforms, tools, libraries, and the like will you use?** Python (Django, Flask, sockets, OS, datetime, system, Bluetooth), HTML, CSS, VsCode, JavaScript (React, node, NPM, axios, CORS)

   - **How will you test it?** I have had both technical and non-technical students and professors test my project.

   - **How will you evaluate the adequacy of your test strategy?** Ensuring the functionality and the capability of the user to understand the use of the project and see its importance. Moreso, I have had both technical and non-technical students and professors test my project.

6. **Project Management**

   - Build a Time Management chart for your project



IDEAL Project Progression, part 1

Project Outline:
- What to include
- What I can done
- What I am addressing

Frontend:
- Initial React UI
- How to serve API

Backend:
- Python Django
- How Django works (apps, views, urls, settings, etc..)
- Connecting Django with React

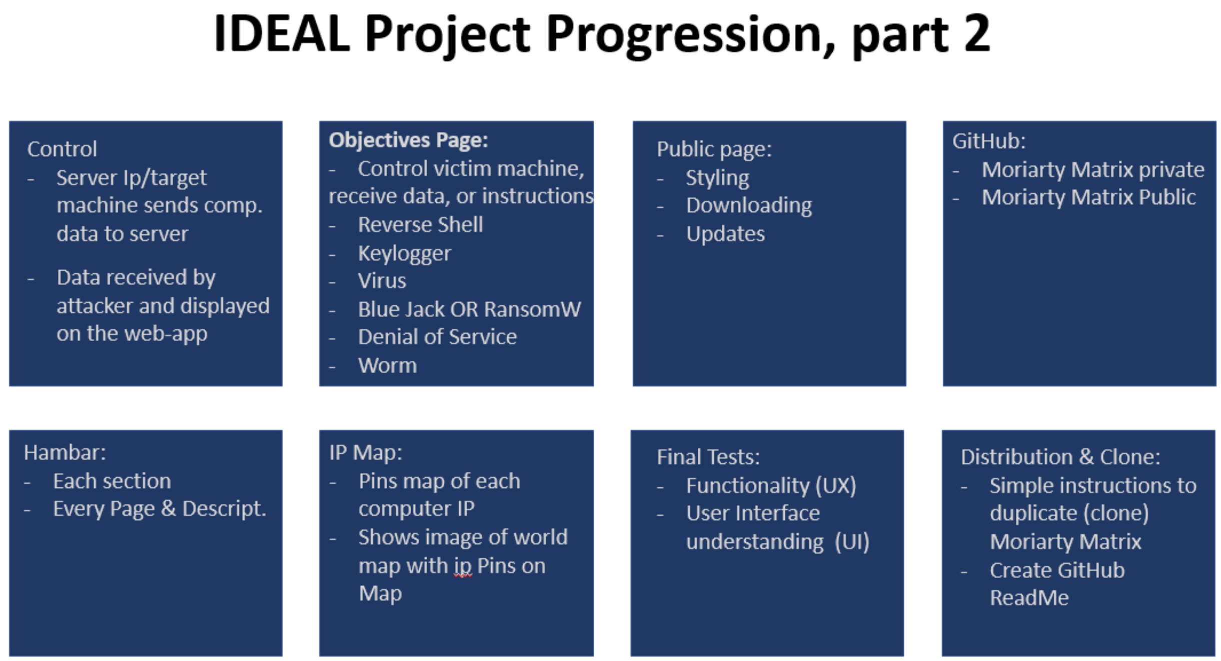
Recon Page:
- Username
- IP
- Domain
- Phishing
- Email
- Bluetooth

Weapon Page:
- Reverse Shell
- Keylogger
- Virus
- Blue Jack OR RansomW
- Denial of Service
- Worm

Delivery Page:
- Email
- Bluetooth
- Website Link (Phish)
- FTP
- USB
- Text Message

Exploit Page:
- Exploit DB
- National Vuln Data
- CVE

Install Page:
- Windows
- Mac
- Linux



IDEAL Project Progression, part 2

Control
- Server Ip/target machine sends comp. data to server
- Data received by attacker and displayed on the web-app

Objectives Page:
- Control victim machine, receive data, or instructions
- Reverse Shell
- Keylogger
- Virus
- Blue Jack OR RansomW
- Denial of Service
- Worm

Public page:
- Styling
- Downloading
- Updates

GitHub:
- Moriarty Matrix private
- Moriarty Matrix Public

Hambar:
- Each section
- Every Page & Descript.

IP Map:
- Pins map of each computer IP
- Shows image of world map with ip Pins on Map

Final Tests:
- Functionality (UX)
- User Interface understanding (UI)

Distribution & Clone:
- Simple instructions to duplicate (clone) Moriarty Matrix
- Create GitHub ReadMe

7. Constraints and Risks

- **Are there any social, ethical, policy, or legal constraints?** The Moriarty Matrix project faces social, ethical, and legal constraints that the team must adhere to. The system must comply with data privacy and security policies to protect user data.

- **Will you have access to the data, services, and resources you need?** As of now, I have access to all the resources I need. However, extra laptops or machines would be helpful for testing my project on.
- **Is there anything else you might need?** If I am going to host my Moriarty Matrix (current local-host website) on the www, then I would need financial assistance and the domain name registered, because it is a dynamic webpage.