

Tengu Marauder Vanguard Blackhat USA 2025

System Background

The **Tengu Marauder Vanguard** is a mobile robotics platform powered by a raspberry pi 4 and an **ESP32** or similar controller, integrated with Python (Flask) and can be controlled via serial or the new web interface. It is designed for field testing, network operations, and educational use in cybersecurity and robotics.

- **Controller:** Raspberry Pi 4 (or equivalent)
- **Motor Driver:** SunFounder Robot HAT or equivalent (PWM + GPIO pins)
- **Language:** Python 3 (Flask app)
- **Control Interface:** Web-based buttons or keyboard inputs
- **Libraries Used:** `robot_hat`, `flask`, `serial`, `cv2`

Motor Control and Logic

Each motor is controlled via:

- **PWM Pin:** Controls speed
- **Direction Pin:** Controls forward/reverse polarity

```
motor_right = Motor(PWM('P12'), Pin('D4'))
motor_left = Motor(PWM('P13'), Pin('D5'))
```

Movement Functions:

```
def move(forward=True):    motor_right.speed(50 if forward else -50)
motor_left.speed(-50 if forward else 50)
def turn(right=True):    motor_right.speed(-50 if right else 50)
motor_left.speed(-50 if right else 50)
def stop():    motor_right.speed(0)    motor_left.speed(0)
```

Demo Script

For the sake of this demo a pre-set script has been created to run on the robots. There may be some mismatched voltages that may result in the wheels turning in opposite directions. This can be easily rectified from updated the PWN lines in the python code.

```
sudo python3 Tengu-Marauder-Vanguard/Tests/motorcontrolv2.py
```

Web Interface Control


The web interface routes like `/motor/forward`, `/motor/left`, etc., call the corresponding Python functions. You can control the robot using buttons on the Flask-based control panel.

To start your Flask-based **web interface** (like the one powering Tengu Marauder Vanguard) using a **Python virtual environment (venv)**, follow these steps:

Step-by-Step: Start Flask Web Interface with `venv`

1. Navigate to your project folder `cd ~/Desktop/Tengu-Marauder-Vanguard`

2. Activate the virtual environment source `venv/bin/activate``

 If you haven't created a `venv` yet:

```
python3 -m venv venv  
source venv/bin/activate
```

3. Install Requirements (if not already done)

bash

Copy code

```
pip install -r requirements.txt
```

4. Run the Flask App

```
python3 Control/operatorcontrol.py
```

Or if you want to explicitly run it with Flask:

```
export FLASK_APP=Control/operatorcontrol.py
export FLASK_ENV=development
flask run --host=0.0.0.0 --port=5000
```

This makes it accessible on your local network at `http://(your-pi-ip):5000`

5. To Stop the Server

Press `CTRL + C` in the terminal.

ESP32 Marauder CLI Training Workbook

Duration: ~30 minutes

Required: ESP32 Marauder device, USB cable, serial terminal (screen, PuTTY, Termux, etc.)

1. Introduction to the CLI (5 min)

What is it?

The ESP32 Marauder CLI lets you interact with your device over serial using commands for scanning, attacks, and more.

Why use it?

- Lightweight
- No GUI required
- Fast for real-time testing

Connection:

Use a serial terminal (e.g., screen, PuTTY):

```
screen /dev/ttyUSB0 115200
```

Then type:

help

 [CLI Docs](#)

2. Core CLI Commands Overview (5 min)

Category	Commands
Admin	reboot
Wi-Fi Scanning	scanap , scansta , stopscan
Sniffing	sniffbeacon , sniffdeauth , sniffpmkid
Wi-Fi Attack	attack -t deauth
Aux Commands	channel , clearap , listap , select

3. Live Demo CLI Walkthrough (7 min)

Simulated Session

```
scanap 1          # Scan nearby APs on channel 1

listap            # List discovered APs

select ap 0       # Select AP at index 0

attack -t deauth  # Begin deauth attack

stopscan          # Stop scanning or attack
```

```
reboot # Restart device
```

Optional:

Switch channels:

```
channel 6
```

4. Hands-On Exercise (10 min)


Task: Perform AP Scan and Targeted Deauth

Step 1: Connect via USB

```
screen /dev/ttyUSB0 115200
```

Step 2: Run Commands

```
scanap 6  
listap  
select ap 0  
attack -t deauth
```

 Observe logs and reactions in terminal.

Step 3: Stop Attack

```
stopscan
```

Learning Objectives

- ✓ Identify nearby access points
- ✓ Select a specific target
- ✓ Launch a deauth flood
- ✓ Stop attack safely and reset

5. Advanced Topics / Q&A (5 min)

- `select ssid -f 'Guest'` — filter targets by name
- `clearap -a` — clear AP list
- `sniffpmkid` — capture PMKID hashes
- Firmware v1.7.0 adds: TCP port scan, join Wi-Fi, more

Resources:

- [CLI Docs](#)
- [Flipper Zero App](#)
- [Smol Reference](#)

Summary

- Connected to the device and ran `help`
- Performed AP scan and targeted attack
- Learned how to stop and reboot cleanly

Pro Tips

“Plug in your Flipper Zero or Marauder, then run:

```
screen /dev/ttyUSB0 115200 to access the CLI.”
```

“You can script multiple commands or use automation with Smol or Flipper Lab tools.”

Wrap-Up & Next Steps

- Practice command chaining
- Explore scripting with `.json` workflows
- Review the full [ESP32Marauder CLI](#)

● **Always test only on networks you own or have explicit permission to assess.**