**TheMiniHat**

This guide details how I built a fully portable, headless, offline hacking device using a Raspberry Pi 5. The goal was to create a lightweight, standalone pen-testing tool that requires no internet, no monitor, and no external network to operate. It broadcasts its own Wi-Fi network and can be accessed via SSH from an iPhone, Android device, or laptop.

**Components Used**

* **Raspberry Pi 5**
* **Raspberry Pi 5 Hat+ with SSD** (for fast boot and storage)
* **SSK M.2 NVME SATA SSD Enclosure Adapter** (to flash OS onto SSD)
* **Official Pi 5 Case**
* **ALFA Network AWUS036ACS** (for wireless pen-testing)
* **ASUS USB-BT500**(for wireless pen-testing)
* **Baseus MagSafe 10,000mAh Power Bank** (compact and portable)
* **Initial setup done with** standard HDMI monitor, keyboard, mouse, computer, and internet connection

**Use Case: Why Do This?**

* Fully portable wireless penetration testing lab (imagine the possibilities)
	+ Rogue Access Point and Evil Twin Attacks
	+ Packet Injection and Deauthentication Attacks
	+ Wireless Reconnaissance (WIFI and Bluetooth)
* No router or internet required
* Secure and personal AP for field control or testing
* Controlled from an inconspicuous phone via Wi-Fi (SSH, VNC, or Web Tools)

**Step 1: OS Setup**

* Flash **Kali Linux for Raspberry Pi 5** to your SSD (Using Pi Imager on computer or laptop)
* Transfer SSD into Hat+
* Boot the Pi with a keyboard, mouse, monitor, and internet connection (Just for initial setup)
* Update and Upgrade the Kali with any tools you want

**Step 2: Install Required Packages**

sudo apt update

sudo apt install hostapd dnsmasq openssh-server -y

sudo systemctl stop hostapd

sudo systemctl stop dnsmasq

**Step 3: Assign Static IP to wlan0-Integrated NIC**

Edit the dnsmasq.conf file:

sudo nano /etc/dnsmasq.conf

Paste:

interface wlan0

 static ip\_address=192.168.4.1/24

 nohook wpa\_supplicant

Enable the service:

sudo systemctl enable dhcpcd

**Step 4: Configure dnsmasq**

Backup and Create:

sudo mv /etc/dnsmasq.conf /etc/dnsmasq.conf.orig

sudo nano /etc/dnsmasq.conf

Paste:

interface=wlan0

bind-interfaces

no-resolv

dhcp-range=192.168.4.10,192.168.4.100,255.255.255.0,24h

dhcp-option=3,192.168.4.1

dhcp-option=6,192.168.4.1

**Step 5: Configure hostapd**

sudo nano /etc/hostapd/hostapd.conf

Paste:

interface=wlan0

driver=nl80211

ssid=KaliPi

hw\_mode=g

channel=6

ieee80211n=1

wmm\_enabled=1

country\_code=US

ieee80211d=1

auth\_algs=1

wpa=2

wpa\_passphrase=RedHatPass

wpa\_key\_mgmt=WPA-PSK

rsn\_pairwise=CCMP

ignore\_broadcast\_ssid=0

Point to the config:

sudo nano /etc/default/hostapd

DAEMON\_CONF="/etc/hostapd/hostapd.conf"

**Step 6: Make wlan0 Unmanaged by NetworkManager**

Create or edit the configuration file:

sudo nano /etc/NetworkManager/conf.d/unmanaged-wlan0.conf

Paste:

[keyfile]

unmanaged-devices=interface-name:wlan0

Restart NetworkManager:

sudo systemctl restart NetworkManager

**Step 7: Add Boot-Time Static IP Fix Script**

Create the script:

sudo nano /usr/local/bin/fix-wlan0.sh

Paste:

#!/bin/bash

ip addr flush dev wlan0

ip addr add 192.168.4.1/24 dev wlan0

ip link set wlan0 up

systemctl restart dnsmasq

systemctl restart hostapd

Make it executable:

sudo chmod +x /usr/local/bin/fix-wlan0.sh

Create the systemd service:

sudo nano /etc/systemd/system/fix-wlan0.service

Paste:

[Unit]

Description=Fallback IP assignment for wlan0

After=network-pre.target

Before=hostapd.service

[Service]

ExecStart=/usr/local/bin/fix-wlan0.sh

Type=oneshot

RemainAfterExit=yes

[Install]

WantedBy=multi-user.target

Enable it:

sudo systemctl daemon-reload

sudo systemctl enable fix-wlan0.service

**Step 8: Start and Enable All Services**

sudo systemctl enable hostapd

sudo systemctl enable dnsmasq

sudo systemctl enable ssh

sudo reboot

**Step 9: Connect from iPhone or Laptop**

* On iPhone, Termius app works great
* Wi-Fi SSID: **KaliPi**
* Password: **RedHatPass**
* SSH from any device:

ssh kali@192.168.4.1

**Troubleshooting**

* After booting give the AP 1-2 minutes to fully come online.
* If clients get stuck connecting: run sudo systemctl restart hostapd dnsmasq
* If no DHCP lease: check sudo cat /var/lib/misc/dnsmasq.leases
* If wlan0 is DOWN after boot: run /usr/local/bin/fix-wlan0.sh

**Power and Portability**

* **Baseus 10,000mAh MagSafe** runs this setup for around 2 hours
* Any 5V/3A USB-C battery works (Anker, Zendure, Shargeek, etc.)

**Legal and Ethical Considerations**

This guide is intended **strictly for educational and lawful security testing purposes**. The setup described here is powerful — you’ve essentially built a pocket-sized hacker lab — but with great power comes great... legal responsibility.

**Only use this device on networks and systems you own or have explicit written permission to test.** That means no “but I was just testing the hotel Wi-Fi” excuses.

Use this set up to learn, to teach, to defend, or to impress your hacker friends — **not to become a headline** on a cybersecurity blog.

Anyways… **Enjoy!**