

War-Walk How-To Guide

Version: 1.0

Author: PixelatedDwarf

Purpose: Step-by-step instructions for conducting a passive SIGINT sweep using Kali Linux and the ALFA AWUS036AXM.

Prerequisites

Before you begin, ensure you have the following:

- A laptop running **Kali Linux** (Live USB or installed).
 - **ALFA AWUS036AXM** Wi-Fi adapter (or similar with Monitor Mode support).
 - **GPS Receiver** (USB dongle or smartphone with `gpsd` support).
 - A backpack with a mesh window or antenna port (optional but recommended).
 - Power bank for extended walks.
-

Part 1: The Beginner Track

Goal: Get a basic map of Wi-Fi networks and Bluetooth beacons.

Step 1: Prepare Your Hardware

1. **Shutdown** your laptop if it is running.
2. **Plug in** the ALFA adapter into a USB 3.0 port.
3. **Plug in** your GPS receiver.
4. **Power on** the laptop.

Step 2: Verify Drivers

1. Open a **Terminal**.
2. Type `lsusb` and press Enter.
3. Look for "Realtek" or "Atheros" in the list. If you see it, the driver is loaded.
4. Type `iwconfig`. You should see your external adapter listed (usually `wlan1` or `wlx...`). If it says "no wireless extensions," the driver is missing. Reboot or install `firmware-realtek`.

Step 3: Configure GPS

1. Type `sudo systemctl start gpsd`.
2. Type `cgps -s`.
3. Wait for the screen to show latitude and longitude. If it says "No satellites," move outside or near a window.

Step 4: Launch Kismet

1. Type `sudo kismet` and press Enter.

2. In the Kismet interface (text or web UI):
 - Select your **ALFA adapter** as the source.
 - Enable **GPS** as the source.
 - Set the output format to **SQLite** (default).
3. Click **Start**.

Step 5: The Walk

1. Put the laptop in your backpack.
2. Walk your neighborhood at a normal pace.
3. Do not stop unless necessary.
4. Walk for 30 to 60 minutes to cover your Area of Operations.
5. Return home.

Step 6: Export Data

1. Stop Kismet (Ctrl+C).
 2. Navigate to the Kismet log folder (usually `~/ .local/share/kismet/`).
 3. Use the tool `kismetdb_to_csv` to convert the database to a CSV file.
 - Command: `kismetdb_to_csv -i your_log.kdb -o walk_data.csv`
 4. You now have a raw data file ready for the next step.
-

Part 2: The Advanced Track (Corrected)

Goal: Deep spectrum analysis, SDR integration, and **Electronic Terrain Baseline** creation.

Step 1: Advanced Adapter Tuning

1. Ensure your ALFA adapter is in **Monitor Mode**.
 - Command: `sudo ip link set wlan1 down`
 - Command: `sudo iw dev wlan1 set monitor none`
 - Command: `sudo ip link set wlan1 up`
2. Increase transmission power (if supported) for better sensitivity:
 - Command: `sudo iw dev wlan1 set txpower fixed 3000 (30dBm).`

Step 2: SDR Integration (Optional)

1. Connect your **RTL-SDR** or **HackRF**.
2. Install `rtl_power` or `gnuradio`.
3. Run a background sweep on the 433MHz and 915MHz bands to detect non-Wi-Fi sensors (garage doors, alarms).
 - Command: `rtl_power -f 433M:435M:10k -i 10s -e 1h sdr_log.csv`
4. Merge this data with your Wi-Fi data later.

Step 3: Baseline Capture & Filtering

Crucial Concept: Do **not** delete data. Your goal is to build a persistent **Electronic Terrain Baseline**.

1. Run the **OUI Lookup Script** in **Full Capture Mode**.

- Command: `python3 ouillookup.py -i walk_data.csv -o full_baseline.csv`
 - *This script tags every network with its vendor (e.g., "Netgear Router", "Ring Camera", "Unknown AP") but keeps ALL entries.*
2. **Why this matters:** By keeping every router and access point, you create a historical record. Six months from now, if a police van with a mobile Wi-Fi node parks on your street, or a new surveillance camera is installed, you can compare the new data against this baseline.
- **New Entry:** A network that wasn't there before is flagged as a potential **Anomaly** (Mobile Unit, New Sensor).
 - **Missing Entry:** A network that disappeared might indicate a device was moved or taken offline.
 - **Signal Shift:** A sudden spike in signal strength might indicate a device was upgraded or moved closer.
3. **Optional Filtered View:** If you only want to visualize the *threats* for a quick map, run the script with the `--filter` switch.
- Command: `python3 ouillookup.py -i full_baseline.csv -o threat_view.csv --filter`
 - *This creates a subset for the map generator, highlighting only surveillance vendors while preserving the full baseline for future comparison.*

Step 4: Map Generation

1. Run the **Map Generator** using your **Filtered View** for clarity, but keep the **Full Baseline** saved for future diffs.
 - Command: `python3 map_generator.py -s threat_view.csv -o threat_map.html`
 2. Open `threat_map.html` in your browser.
 3. Use the legend to identify
 - **Red** (High Confidence)
 - **Orange** (Silent)
 - **Yellow** (Private) zones.
-

Troubleshooting

Issue: Adapter not detected.

- Solution: Replug the USB. Try a different port. Run `dmesg | grep usb` to see kernel errors.

Issue: GPS not locking.

- Solution: Move to an open area. Ensure `gpsd` is running. Check cable connections.

Issue: Kismet crashes.

- Solution: Ensure you are running as `sudo`. Check disk space.

Issue: No signals detected.

- Solution: You may be in a dead zone, or the adapter is not in Monitor Mode. Run `iwconfig` again to verify.
-

Safety & OPSEC

- **Blend In:** Do not look like you are hacking. Walk casually.
- **Legal:** Passive scanning is generally legal. Do not connect to networks. Do not inject packets.
- **Privacy:** Do not store personal data of neighbors in your map. Anonymize addresses if sharing.
- **Battery:** Bring a power bank. Kismet and GPS drain batteries quickly.