	00 101 110 11	<div>CYBER SEWER v3rsi0n_0</div>		AREA	
				Hardware	
				DATE	
				22-12-2017	
		WIFI DEAUTH 1.6 ESP8266 NodeMCU THOMPSON VANGLLER		ID	PAGE
HDWE - 01	1/8				

I. PURPOSE

My purpose with this project is show how vulnerable is the 802.11 WiFi standard and demonstrate the attacks we can do using the NodeMCU ESP8266 with the WiFi Deauth 1.6, this device is useful for performing pentest.


II. DISCLAIMER

These Tutorial are for educational purposes only. I am not responsible for how you use this device in any way shape or form. It's not a frequency jammer, its a attack, works by exploiting an old and known vulnerability in the 802.11 Wi-Fi protocol.

In some countries the use of this device maybe is unlawful, check the legal regulations in your country before using it.

For detailed information on law enforcement in UK (Wireless Telegraphy Act 2006), visit the website:

<http://www.legislation.gov.uk/ukpga/2006/36>

 <div>00 101 110 11</div>	CYBER SEWER v3rsi0n_0		AREA	
			Hardware	
			DATE	
			22-12-2017	
	WIFI DEAUTH 1.6 ESP8266 NodeMCU THOMPSON VANGLLER		ID	PAGE
			HDWE-01	2/8

III. ABOUT THE PROJECT

It's a device which plays out a **deauth** on WiFi Networks near our range. We need flash the code **esp8266_deauther** by **spacehuhn** from GitHub onto a **chip ESP8266**. That make this attack in a simple way.

You just need select the **AP's** you want assault, select a type of attack and play it! For whatever length of time that the attack is running, the chosen AP's can't interface with their network or slow it down. Also this code can run **beacon and probe request flooding attack**.

HOW IT's WORK

The 802.11 WiFi convention contains an alleged deauthentication outline. It is utilized to detach customers securely from a remote system. Since these bundles are decoded, you simply require the MAC address of the WiFi switch and of the customer device which you need to detach from the system. You don't should be in the system or know the password, it's sufficient to be in its range.

COMPONENTS

For this project you need to use:

- Board NodeMCU ESP8266;
- Code ESP8266 Deauther (spacehuhn)

Board NodeMCU ESP8266

The ESP8266 is a low-cost Wi-Fi chip with full TCP/IP stack and MCU (microcontroller unit) capability produced by Shanghai-based Chinese manufacturer, Espressif Systems.

The chip first came to the attention of western makers in August 2014 with the ESP-01 module, made by a third-party manufacturer, Ai-Thinker. This small module allows microcontrollers to connect to a Wi-Fi network and make simple TCP/IP connections using Hayes-style commands. The very low price and the fact that there were very few external components on the module which suggested that it could eventually be very inexpensive in volume, attracted many hackers to explore the module, chip, and the software on it.

en.wikipedia.org/wiki/ESP8266

General Features

Memory: 64 KiB instruction, 96 KiB data

Manufacturer: Espressif Systems

CPU: 32-bit @ 80 MHz


Flash Memory: 4Mb

Port: USB-TTL (CH340)

***For detailed technical specifications, access the link bellow, also can you find the datasheet:**

<http://espressif.com/en/products/hardware/esp8266ex/overview>

</HACK. THE. PLANET : #>

	00 101 110 11	CYBER SEWER v3rsi0n_0		AREA	
				Hardware	
				DATE	
		WIFI DEAUTH 1.6 ESP8266 NodeMCU THOMPSON VANGLLER		22-12-2017	
				ID	PAGE
		HDWE-01	3/8		

Code: ESP8266 Deauther - from *spacehuhn github.com*

“----

Introduction

This software allows you to perform a deauth attack with an ESP8266 against selected networks. With this software flashed onto a ESP8266 chip, you can select a target network and start different attacks.

The deauth attack will, if the connection is vulnerable, disconnect the devices from the network. Because the attack is running constantly, the devices will be disconnected again and again. Depending on the network, that can either block a connection or slow it down.

How to protect yourself against it

With **802.11w-2009** the Wi-Fi protocol became encrypted management (and deauthentication) frames. This makes spoofing these packets way harder and the attack, in this form, ineffective. So make sure your router is up to date and has management frame protection enabled. Your client device (e.g your phone, notebook etc.) needs to support that too. Both ends of the connection need to use it!

The problem with that is, most routers use unencrypted management frames by default, don't provide any option to change that and don't provide any information about this issue.

I tested several networks and couldn't find one that wasn't vulnerable!

I made a **Deauth Detector** (<https://github.com/spacehuhn/DeauthDetector>) using the same ESP8266 to indicate high amounts of deauth frames. It can't protect you, but it can help you figure out if and when an attack is going on.


---”

spacehuhn

https://github.com/spacehuhn/esp8266_deauther

“Please don't refer to this project as "jammer", that totally undermines the real purpose of this project!”

Stefan Kremser- (@spacehuhn)

	00 101 110 11	<div>CYBER SEWER v3rsi0n_0</div>		AREA	
				Hardware	
				DATE	
				22-12-2017	
		<div>WIFI DEAUTH 1.6 ESP8266 NodeMCU THOMPSON VANGLLER</div>		ID	PAGE
HDWE - 01	4/8				

IV. Building Your Own Device

STEP 00

Buy the board nodeMCU ESP8266:

```
http://www.nodemcu.com/index_en.html#fr_54747661d775ef1a3600009e
```

STEP 01

With the board ESP8266 in hand we need download the code from github.com, we can use the terminal:

```
root@laptop:~# wget
https://github.com/spacehuhn/esp8266_deauther/releases/download/v.1.6/
esp8266_deauther_1mb.bin

HTTP request sent, awaiting response... 200 OK
Length: 569696 (556K) [application/octet-stream]
Saving to: 'esp8266_deauther_1mb.bin'
esp8266_deauther_1m 100%[=====>] 556.34K 203KB/s in
2.7s
2017-12-22 15:03:23 (203 KB/s) - 'esp8266_deauther_1mb.bin' saved
[569696/569696]
```

Check the download file:


```
root@laptop:~# ls -lh
drwxr-xr-x 3 root root 4.0K Dec 15 22:35 Desktop
drwxr-xr-x 3 root root 4.0K Aug 6 00:45 Documents
-rw-r--r-- 1 root root 557K Aug 8 12:09 esp8266_deauther_1mb.bin
```

STEP 02

For upload the code onto chip ESP8266 we need a flash software, for this tutorial I will use **esptool**, this software runing in Windows, MacOS, Linux. In this tutorial I will demonstrate using Ubuntu Linux. Install the esptool software , open the terminal and run the code bellow:

```
root@laptop:~# apt-get install esptool
```

...

	00 101 110 11	<div>CYBER SEWER v3rsi0n_0</div>		AREA	
				Hardware	
				DATE	
				22-12-2017	
		<div>WIFI DEAUTH 1.6 ESP8266 NodeMCU THOMPSON VANGLLER</div>		ID	PAGE
HDWE - 01	5/8				

STEP 03

Plug the **ESP8266 chip** through the **USB** into the computer. Check if the device has been recognized by the system.

```
root@laptop:~# lsusb
Bus 002 Device 002: ID 8087:0024 Intel Corp. Integrated Rate Matching Hub
...
Bus 004 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 003 Device 004: ID 1a86:7523 QinHeng Electronics HL-340 USB-Serial adapter
Bus 003 Device 002: ID 046d:c52b Logitech, Inc. Unifying Receiver
Bus 003 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
```

Note the device had been recognized (**Bus 003 Device 004: ID 1a86:7523 QinHeng Electronics HL-340 USB-Serial adapter**), now we can upload the code.

STEP 04

Upload the code (**esp8266_deauther_1mb.bin**) with the **esptool** command. We can sent to the chip via the serial write_flash command:


```
root@laptop:~# esptool -port /dev/ttyUSB0 write_flash 0x000000
esp8266_deauther_1mb.bin
esptool.py v2.3-dev
Connecting....
Detecting chip type... ESP8266
Chip is ESP8266EX
Uploading stub...
Running stub...
Stub running...
Configuring flash size...
Auto-detected Flash size: 4MB
Flash params set to 0x0240
Compressed 569696 bytes to 364969...
Wrote 569696 bytes (364969 compressed) at 0x00000000 in 32.2 seconds
(effective 141.6 kbit/s)....
Hash of data verified.

Leaving...
Hard resetting...
```

Your board are ready to use! **Well done!**

V. DRIVE INTO WIFI DEAUTH 1.6

First start connect you ESP8266 board onto a power supply (USB charge, bank power). Get your computer or smarthphone to control the ESP8266 board, scan for **pwned** network and connect it to access the browse page:

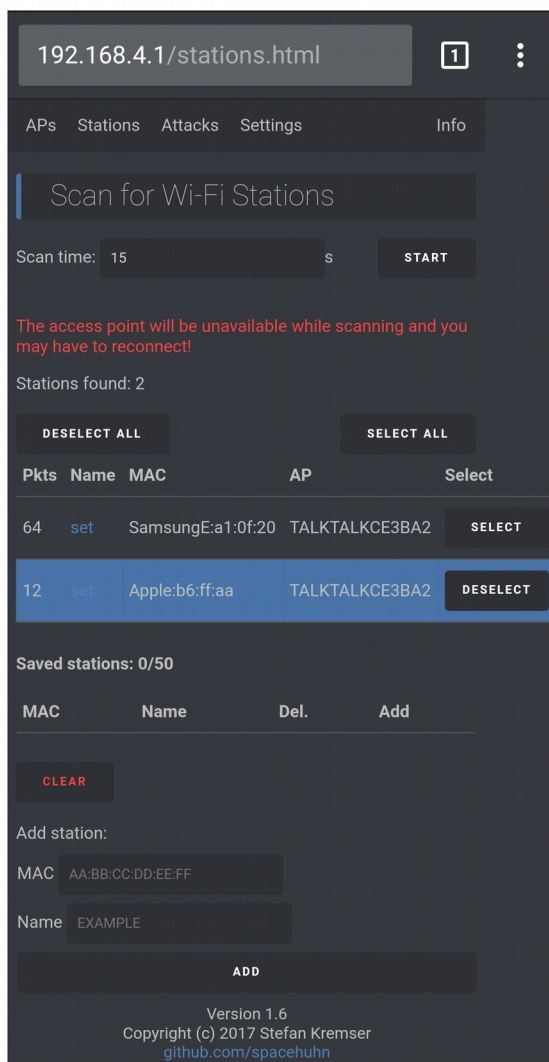
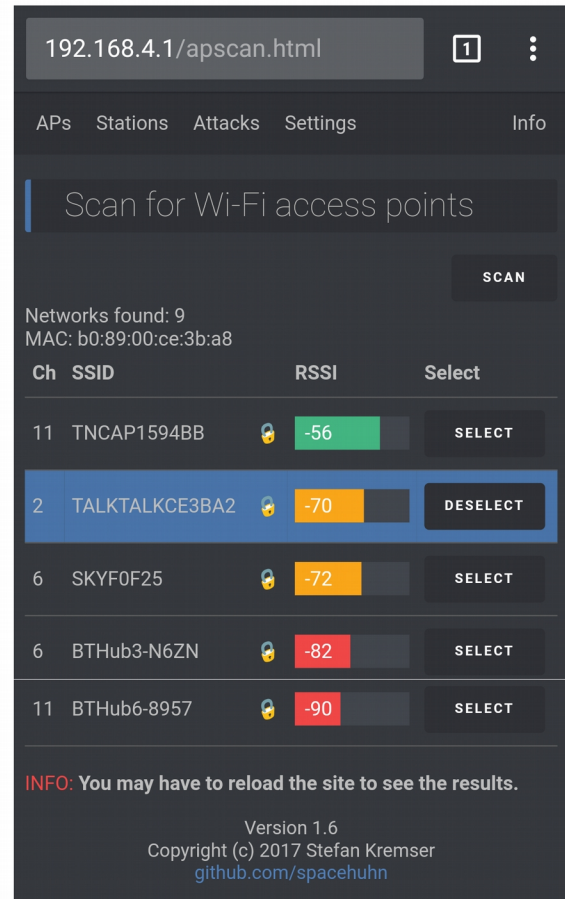
	00 101 110 11	CYBER SEWER v3rsi0n_0		AREA	
				Hardware	
				DATE	
		WIFI DEAUTH 1.6 ESP8266 NodeMCU THOMPSON VANGLLER		22-12-2017	
				ID	PAGE
		HDWE-01	6/8		

SSID: pwned
password: deauther
webpage: 192.168.4.1

SCAN WIFI NETWORKS

Click on tab **APs** and click on the **SCAN** bottom, after it, then it will show all **WiFi networks found**.

We can **select** one/multiples **SSID**, for it just click on **select bottom**, we can improve this SCAN looking for devices on it, or go streigth to **Attacks**.




SCAN FOR CLIENT DEVICES

Click on tab **Stations** and click on the **START** button, after it you will be disconnected from the device, you need to reconnect and refresh the page, then it will show all **WiFi devices found** on the network we have selected before.

We can **select** one/multiples **devices**, for it just click on **select bottom**. Also we can give **name** and **save devices**.

Now we can go to **Attack!**

CK.THE.PLANET : #>

	00 101 110 11	CYBER SEWER v3rsi0n_0		AREA	
				Hardware	
				DATE	
		WIFI DEAUTH 1.6 ESP8266 NodeMCU THOMPSON VANGLLER		22-12-2017	
				ID	PAGE
		HDWE-01	7/8		

Note: While scanning the ESP8266 will shut down its access point, so you may have to go to your settings and reconnect to the Wi-Fi network manually!

PusK>lin ATTACK

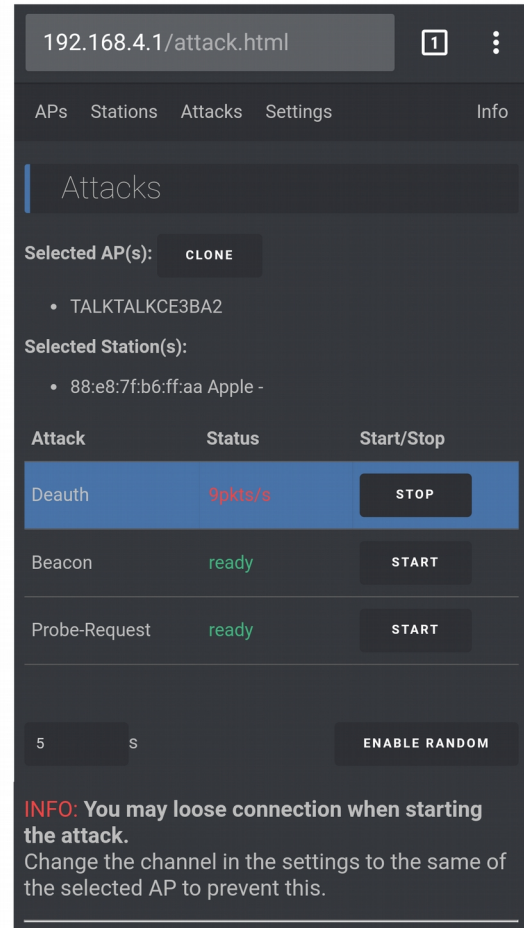
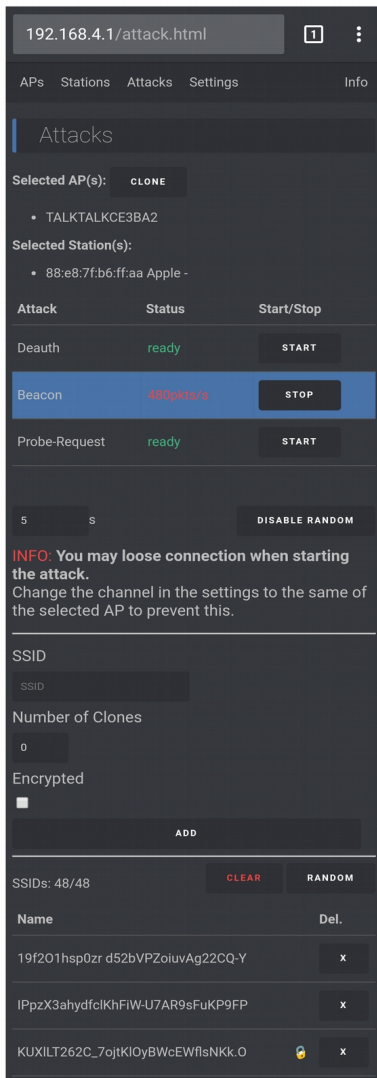
DEAUTH

Click on **tab Attacks** and choose the type of attack you want to use, for this example I will use the purpose of this document, **Deauth Attack**. For do it, just click on the **START** button, that can either **block a connection or slow it down** on the **device/network** selected on **APs/Stations**.

Note:

Disassociating clients can be done for a number of reasons:

- Recovering a hidden ESSID. This is an ESSID which is not being broadcast. Another term for this is “cloaked”;
- Capturing WPA/WPA2 handshakes by forcing clients to reauthenticate;
- Generate ARP requests (Windows clients sometimes flush their ARP cache when



disconnected).

BEACON

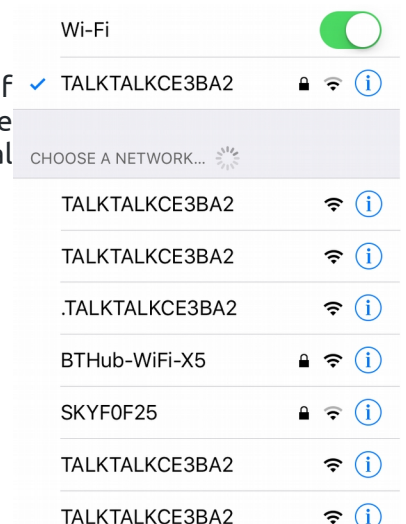
For do Beacon Flood Attack first click on **CLONE** button and next click on **START** button, that will **create** a lot of **fake APs** for the device we have selected. Also we can customize this attack for show **RANDOM** encrypted AP names and customize some settings.


Note:

Beacon flood is where thousands of illegitimate beacons are generate to make it difficult for individual machine to find the legitimate AP.

Look the client side while this attack are running beacon flood attack:

</HACK . THE . PLANET : #>



	00 101 110 11	CYBER SEWER v3rsi0n_0		AREA	
				Hardware	
				DATE	
		WIFI DEAUTH 1.6 ESP8266 NodeMCU THOMPSON VANGLLER		22-12-2017	
ID	PAGE				
HDWE-01	8/8				

SETTINGS

For customize the ESP8266 network or improve each tab from the software click on tab **Settings**.

WIFI

On WiFi setting we can **modify** the **SSID**, change the **password**, configure **channel** for perform attacks, for SCAN all network are close, we can select the channel 1. Also we can modify MAC address or use Random MAC.

AP SCAN

On AP SCAN setting we can **modify** settings for SCAN **hide AP Networks** and setting select **multiple SSIDs**.

STATION SCAN

On STATION SCAN setting we can **change the length** of time that the ESP8266 **will look for the devices**.

ATTACK

On ATTACK setting we can **modify** settings for **improve** the use of attack.

ENJOY!

192.168.4.1/settings.html 1

APs Stations Attacks Settings Info

Settings

Wi-Fi

SSID

pwned

Password (min.8 chars)

deauther

Hide SSID (be careful with this setting!)

☐

Channel

2

MAC

de:4f:22:11:05:06

Random MAC

☐

AP Scan

Scan Hidden APs

☒

Select multiple SSIDs

☐

Station Scan

Default Scan Time

15 s

Attack

Timeout (0 = no timeout)

300 s

Use LED

☒

LED Pin

2