BLUETOOTH Advertising

Goals of this challenge

- Flashing the firmware
- Solving the challenge

Challenge Description: The bluetooth device name is advertised, find it and send it to the DVID board over UART !

Flashing the firmware

- Connect the **DVID board** to the **computer** using the **USB AVR Programmer**, and start the script
- When the AVR programming has completed, the **board should restart** and you should the following
- Flash.sh

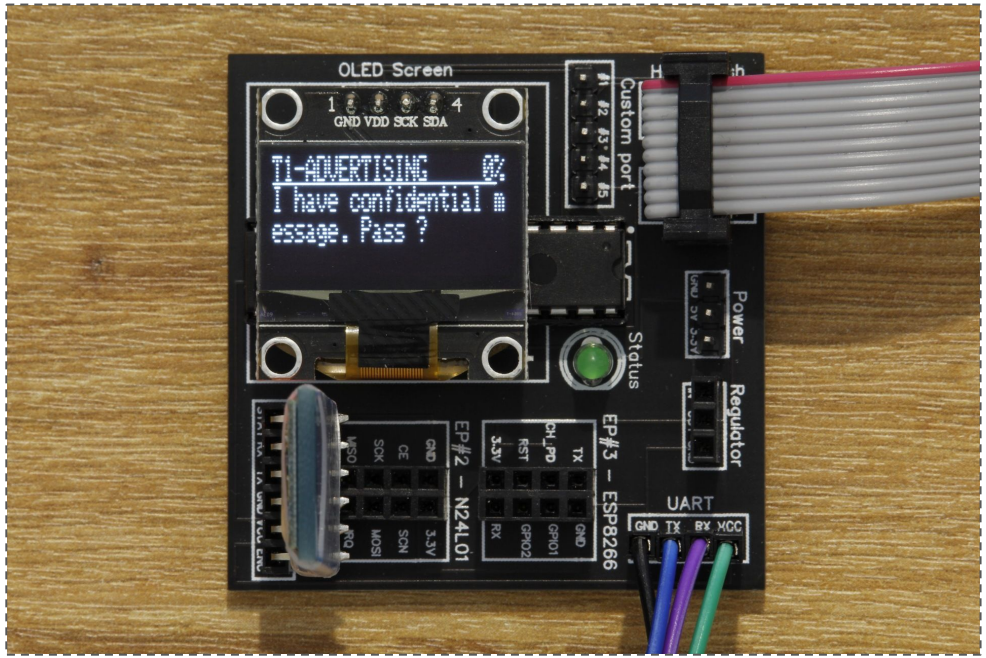
```
#!/bin/bash

if [[ ! -d "./DVID/" ]]; then
    git clone https://github.com/vulcainreo/DVID
fi

pushd ./DVID/trainings/bluetooth/advertising/
avrdude -F -v -p atmega328p -P /dev/ttyUSB0 -c
usbasp -u -U
flash:w:advertising.ino.with_bootloader.arduino_st
andard.hex
popd
```

Note:

Avrdude for windows
https://github.com/mariusgreuel/avrdude/releases/download/v7.0-windows/avrdude-v7.0-windows-windows-arm64.zip
Flash.sh
https://hastebin.skyra.pw/raw/likojafomo

Flashing the firmware

- Now, we will connect the DVID board to our computer using the USB AVR programmer, and start the script. When the AVR programming has completed, the board should restart and you should see the following



Note:

Avrdude for windows
https://github.com/mariusgreuel/avrdude/releases/download/v7.0-windows/avrdude-v7.0-windows-windows-arm64.zip
Flash.sh
https://hastebin.skyra.pw/raw/likojafomo

Solving the challenge

- The DVID board is **advertising** its **bluetooth address and name.**
- To read it, we can either use a phone or bluetooth discovery service, or we can use **bluetoothctl** command, In the bluetooth advertising data, we find:

```
The bluetooth address : 00:13:AA:00:25:91
The device name : dvid-807
```

- Now that we have the device name **dvid-807,** we can send it over UART using this script :
- And we win !

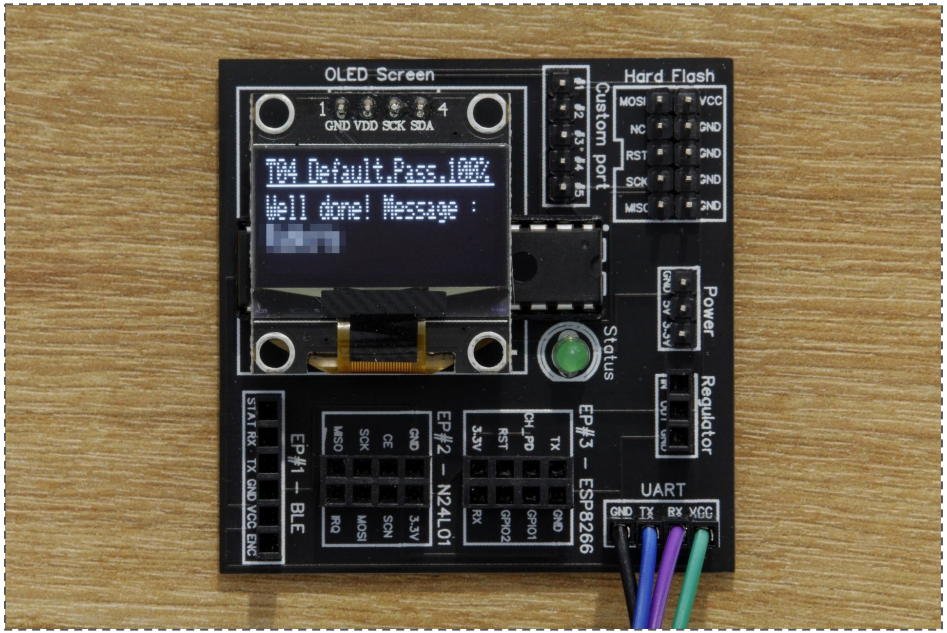
```
#!/usr/bin/env python3

import serial

ser = serial.Serial("/dev/ttyUSB0",
9600)
ser.write(b'dvid-807\r\n')
```

Note:

More about Bluetooth LE Advertisements
https://www.novelbits.io/bluetooth-low-energy-advertisements-part-1/

Solving the challenge

- After a few tries with this wordlist, we find the default password ! It is **password**!



These kind of vulnerabilities might seem stupid, but they are **extremely common.** Unfortunately, many IoT devices often come with weak default username and password **(such as "login: admin, password: admin")** and therefore are vulnerable to dictionary attacks. We can take as an example the **Mirai botnet**, which **scans the Internet looking for IoT devices with default username and password using dictionary attacks over telnet.** Once a new vulnerable device has been discovered, the Mirai botnet binary is injected in the device and it joins the Mirai botnet, **making it stronger and capable of finding new devices faster.**

These botnets are then used to launch bigger attacks such as **distributed denial of service (DDoS)** on more robust structures. For example, the Mirai botnet was widely known for a major distributed denial of service attack on DynDNS in october 2016.