CHALLENGE - FIND EASY PASS

DISASSEMBLING

- We are provided with a Windows Executable `EasyPass.exe`

```
$ file EasyPass.exe
EasyPass.exe: PE32 executable (GUI) Intel 80386, for MS Windows, 8 sections
```

- We can execute it using `wine`
- Once executed, it asks us for a password
- We might try a numbe of password but the response will always be `Wrong Passwords`
- In order to give us this error it must be confronting the provided vs the real password
- Let's look at the strings

```
$ strings EasyPass.exe
```

```
...
ZYYd
hxAE
Good Job. Congratulations
Wrong Password!
Uh5BE
...
```

- That's great there are both `Wrong Password!` (that we know) and `Good job. ...`
- If we disassemble this executable we should be looking for these two strings
- Using OllyDBG we can easily find the piece of code where they belongs

```
0045412B  |. 8B45 D8        MOV EAX,DWORD PTR SS:[EBP-28]
0045412E  |. 8B55 FC        MOV EDX,DWORD PTR SS:[EBP-4]
00454131  |. E8 F204FBFF    CALL EasyPass.00404628
00454136  |. 75 0C          JNZ SHORT EasyPass.00454144
00454138  |. B8 DC414500    MOV EAX,EasyPass.004541DC;  ASCII "Good Job. Congratulations"
0045413D  |. E8 EE38FDFF    CALL EasyPass.00427A30
00454142  |. EB 0A          JMP SHORT EasyPass.0045414E
00454144  |> B8 00424500    MOV EAX,EasyPass.00454200;  ASCII "Wrong Password!"
00454149  |. E8 E238FDFF    CALL EasyPass.00427A30
```

- Now, if we put a breakpoint at `0x00454131` and we inspect EAX and EDX we should find

1. EAX = The password we have used
2. EDX = The password to check for

- The content of `$EDX` is `fortran!` which is our password

  This code call a procedure `EasyPass.00404628` for checking if EAX == EDX and should returns 0 if the two are equal, another value if they are not. If the two are not equal it jumps to `0x00454144` and print `Wrong Password` otherwise go on and print `Good Job. Congratulations`

FLAGS

HTB{fortran!}