Impossible Login

By MetaCTF

Mitigations

- NX - Yes
- PIE - YES
- Canary - Yes
- relro - YES

Win Condition

Login as root. The `motd()` function opens up a flag file and prints it out to the user. We need the root user's flag file.

Road Blocks

There's a call to `strncmp()` in `main()` that checks if the username input is "root". If so, the program returns early

Getting the Passwords

The strings in the binary contain a list of username and passwords alternating in a giant array. Here we can see *root's* password is *cant_guess_this*.

Vulnerability

The usage of `scanf()` to read in both the username and password are vulnerable to buffer overflow attacks. In this case, we want to exploit the one used to read in th password field.

By examining the assembly, we can tell that the password field is store past the username on the stack. This means we can overflow the username by putting more data than expected into the password prompt. We can allow calculate how much we need to pad the password in order to start writing to the username variabe. This also means the username prompt is useless to us here, so it can be anything.

Luckily here `strncmp` is used with the size of the username/password strings. This means any username of password that **starts with** the correct values are valid. This allows us to pad the password.

i.e.

```
Username: a
Password: cant_guess_thisAAAAAAAAAAAAAAAAAroot
Welcome Back root
```

Flag: `MetaCTF{P@ssw0rDS_r_0pti0n4l}`