EE 14 Lab 4: Password "cracking"

Lab report due a week after your lab session (17-21 February 2025)

1 Introduction

In this lab, you'll dive straight into some assembly code to decipher a password.

After successfully completing this lab, you should be able to:

- Run assembly code on your Nucleo, and use the debugger to single-step through instructions.
- Read assembly code with more than a couple dozen instructions.
- Decipher instructions and memory locations on a live system.

Documentation focus: Step-by-step instructions. There will be many times when you need to write instructions for others — other engineers on your team, internal customers and users, technical support, or even actual non-technical customers (yikes!).

Doing this clearly and concisely is actually really hard (a fact I am reminded of every time I write a lab handout!) which is why it's worth practicing.

2 Prelab

There is no prelab for this week, yay!

3 In lab

L1: Download your customized assembly code for this lab, following the link on the course website.

L2: Create an empty PlatformIO project with the same settings as usual:

- **Board**: ST Nucleo L432KC
- **Framework**: CMSIS

L3: Copy the source code you downloaded (`main.s`) into the `src` directory of your project. You should be able to build and run the project without any additional work.

Open the serial monitor (using 9600 baud) to interact with the program. You should see a message saying "Hello, please enter your super secret lab 4 login". If you don't, try hitting "enter" and it should reset.

When you type in the correct password, you should see the message "Success! Your key is..." followed by another password-like string. This is your "key".

L4: The rest of this lab is simple: figure out your key. When you've recovered your key (whether by finding your password or some other way), you're done!

.

Some ideas to get started

We hope part of the fun of this lab is getting to choose your own approach, but here are a few things that might help get you started:

1. The passphrase and key are both based on the EFF dice-generation wordlist.
2. The password and key are obfuscated, but the code "deobfuscates" them before comparing with the password you typed in. So the password exists in clear text in memory, if you know when and where to look.

   As a side note, you should never, ever do this on a system that needs to be secure. The right thing to do is to store the "obfuscated" password using a cryptographically-secure hash (e.g., SHA-256) and then hash what the user typed in the same way to compare. This way, you can still authenticate users, even though it's impossible for you (and hopefully any attackers) to decrypt the passwords.
3. Try running the code in the debugger and observe how it works. There is a lot of code, but all the action is in `main` and `deobfuscate`.
4. All of the strings are stored together in the assembly code. You should be able to take a good guess which ones are the obfuscated password and key.
5. Several memory addresses are encoded exactly the same way we've covered in class over the last week or so (with a PC-relative load that pulls in a physical address). Can you figure out which registers store which pieces of information?
6. You're more than welcome to modify the assembly code. Obviously if you can cause the code to skip the password check, then you'll get your key easily!

4 What to turn in

Your lab report should contain the following:

- A cover page with your name, date, lab section, and lab TA names, and a 2-sentence description of this lab.
- Your key (and password, if you recovered it along the way).
- Clear step-by-step instructions for how someone else could "crack" a password if they were given a similar setup. Focus on getting the right level of detail for someone who understands the Nucleo board / VSCode but has not done this process before (i.e., a classmate who hasn't done this lab yet!)

  It's really easy to forget steps, so you might find it helpful to write your instructions, and then try to follow them yourself, making notes about any steps that might be ambiguous or overlooked entirely.
- Answers to the following questions:
  - What do you understand now, that you didn't before the lab?
  - What are you confused or curious about? (You're still not allowed to say "nothing"!)
  - How long did it take you to complete this lab?