**IoT Security and Privacy**

**Assignment 3 – ESP32 UART and Flash Hack**

**(10 points)**

### Instructions

**Please read the instructions carefully. Those students who fail to follow the instructions may get a zero score for this assignment.**

1. This is a group assignment while it can be done individually. Each group can have at most two students. Each student MUST submit the group report even if it is the same for the group members. Those who do not submit the report will get ZERO for this assignment.
2. Answer each question following the original question. Do NOT delete the original question.
3. Answers to all questions must be put into **ONE** document.
4. Students must put answers following each question in this assignment. The instructor will not grade a report with only answers in it and the student gets zero for such an assignment. An assignment report must include original questions.
5. Students MUST submit the finished assignment in either Microsoft Word or pdf format to Blackboard. The doc must be submitted as ONE standalone file and cannot be tarred or zipped into a container.
6. All required files or docs must be submitted in one submission. Note: Blackboard allows unlimited number of submissions of an assignment and the instructor counts the last one.
7. Refer to [Print screen](http://en.wikipedia.org/wiki/Print_screen) on how to take a screenshot.
8. Underlined blue text points to a web link. Ctrl + Click to follow link.

**Questions**

1. There are three serial ports on the ESP32. UART0 is often used for programming and communication with outside of ESP32 while the other two UART ports are unused. For example, the Micro-USB connector of our IoT kit is connected to UART0 through a USB-UART bridge chip. If there is no protection of UART0, a hacker can access the flash through UART.
   1. Please explain how a hacker may identify UART pins on a PCB if there is no obvious label for the UART. (1 point)
   2. The access to UART can be protected by a password if the firmware is written properly. That is, a user must enter the correct password to upload firmware and communicate with the IoT kit. Please discuss how a hacker may recover the password. (1 point)
2. Please read [ESP32 UART and Flash Hack](https://github.com/xinwenfu/ESP32-UART-and-Flash-Hack) at GitHub. Read [Build and run the WiFi station firmware](https://github.com/xinwenfu/ESP32-UART-and-Flash-Hack#build-and-run-the-wifi-station-firmware) within the tutorial. Use a screenshot to show the ESP32 is connected to a WiFi router (e.g., a hotspot on a smartphone) with the firmware from the WiFi station example project. (2 points)
3. Please read [Retrieve partition table](https://github.com/xinwenfu/ESP32-UART-and-Flash-Hack#retrieve-partition-table) within the tutorial.
   1. Use a screenshot to show the partition table of the ESP32. (1 point)
   2. Explain the partition table. (1 point)
4. Please read [Search firmware for sensitive info](https://github.com/xinwenfu/ESP32-UART-and-Flash-Hack/blob/main/README.md#search-firmware-for-sensitive-info) within the tutorial. Use a screenshot to show the WiFi password can be found in the firmware within a hex editor such as wxHexEditor. (2 points)
5. Please read [Change the firmware](https://github.com/xinwenfu/ESP32-UART-and-Flash-Hack/blob/main/README.md#change-the-firmware) and perform Option 0.
   1. Use a screenshot to show esptool.py can be used to write the changed firmware back to the ESP32. (1 point)
   2. With the changed firmware, can the ESP32 still connect to the WiFi router? Why? (1 point)