**IoT Security and Privacy**

**Term Project – Secure Boot**

**(20 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. The instructor highly recommends students form groups given the complexity of this project. 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.

**Warning**:

In this assignment, students are asked to repeat [ESP32 Secure Boot Tutorial](https://github.com/PBearson/ESP32_Secure_Boot_Tutorial). Please follow the exact order of commands in the tutorial.

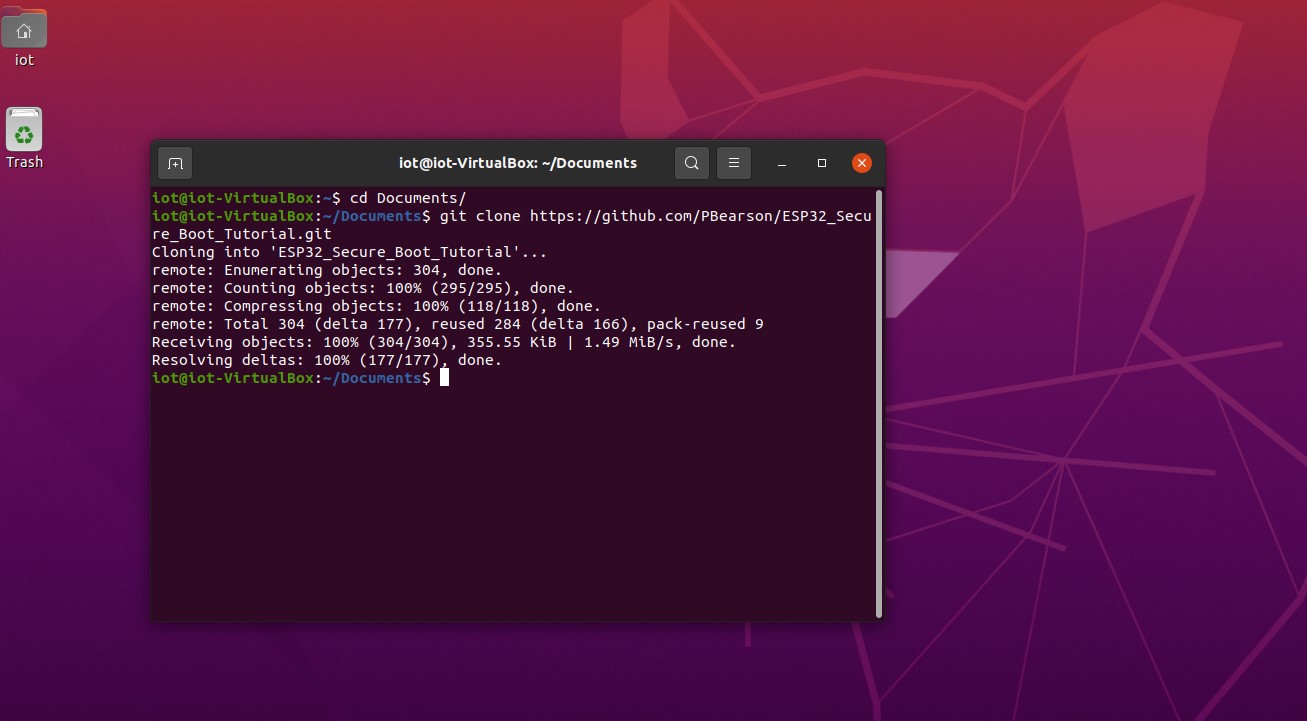
**All *espefuse.py* and *espsecure.py* related commands can only be used one time in a step**. If a student cannot continue because of errors, please just stop and state that the lab cannot be continued due to errors. Such assignment submission will lose at most 5 points depending on how far the lab is performed.

Read the entire tutorial at least two times before working on it.

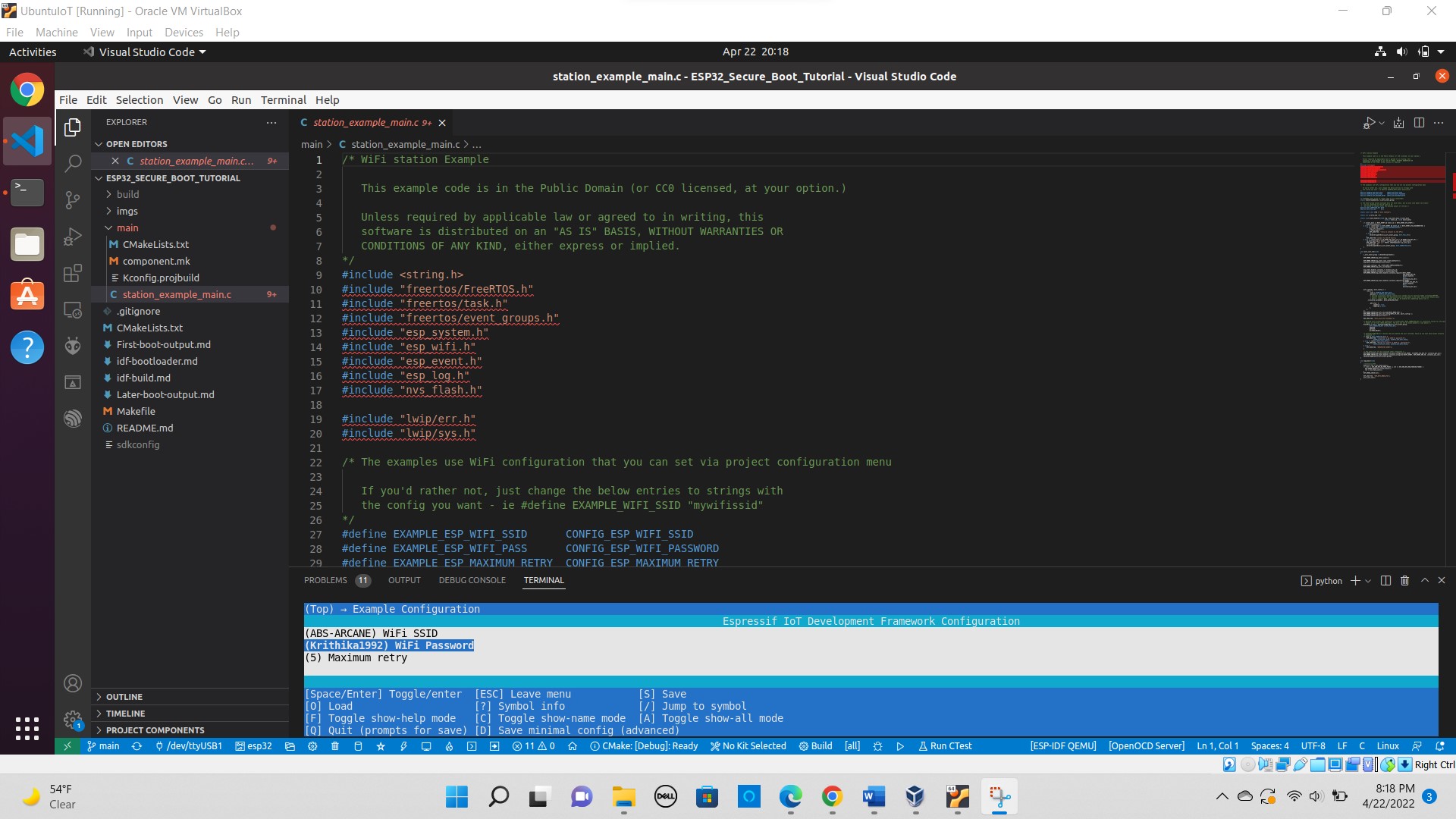
**Questions**

1. *Step 0* *Prerequisite: Install the ESP-IDF extension*. The new Ubuntu VM [UbuntuIoT.ova](https://www.cs.uml.edu/~xinwenfu/VMs/UbuntuIoT.ova) can be downloaded at<https://www.cs.uml.edu/~xinwenfu/VMs/>. The Ubuntu VM is the same as the one used for the flash encryption project.

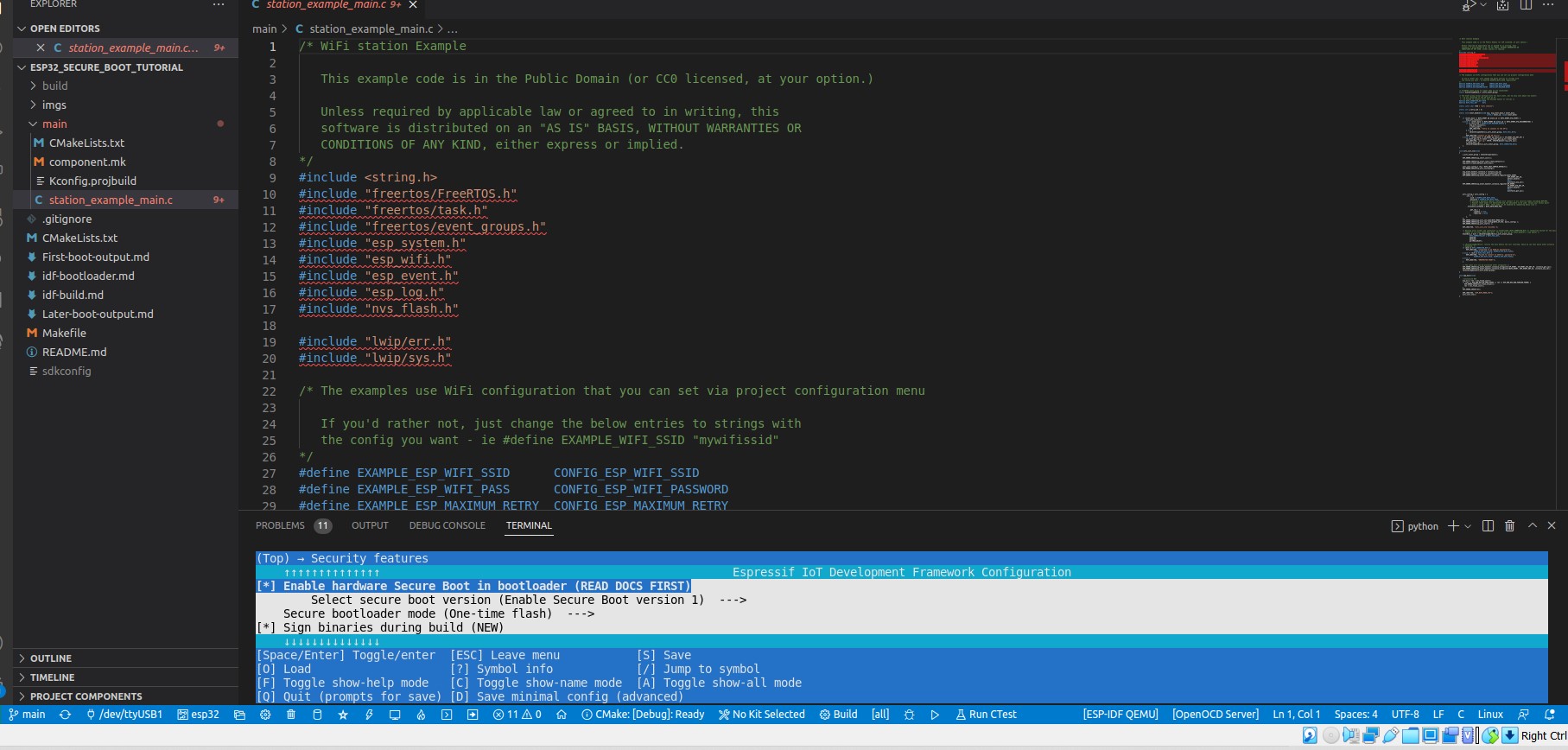
1. *Step 1. Download this repository*. Include a screenshot of the folder of the downloaded project. (1 point)



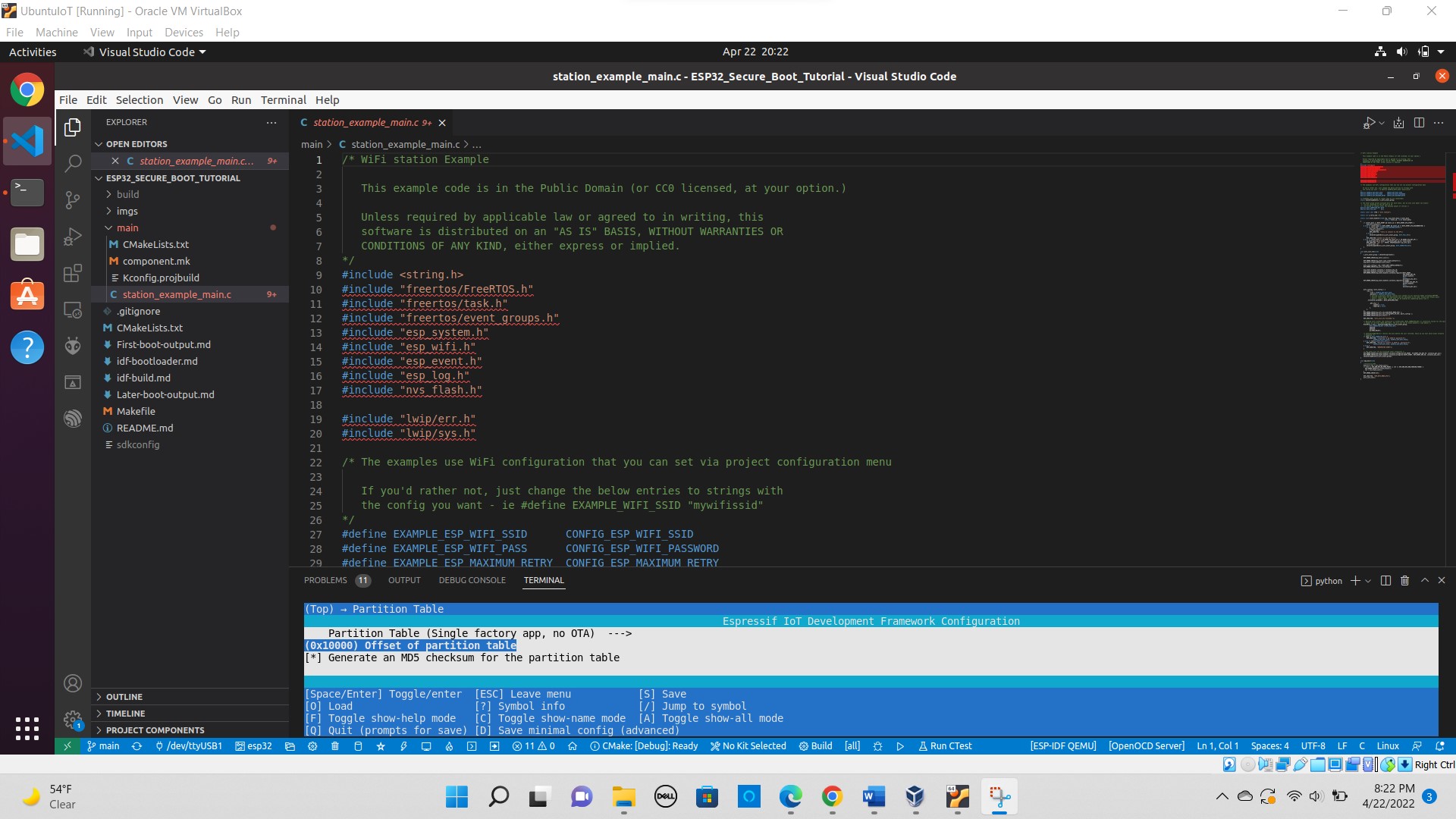
1. *Step 2. Configure the app*.
2. Include a screenshot of configured WiFi. (1 point)



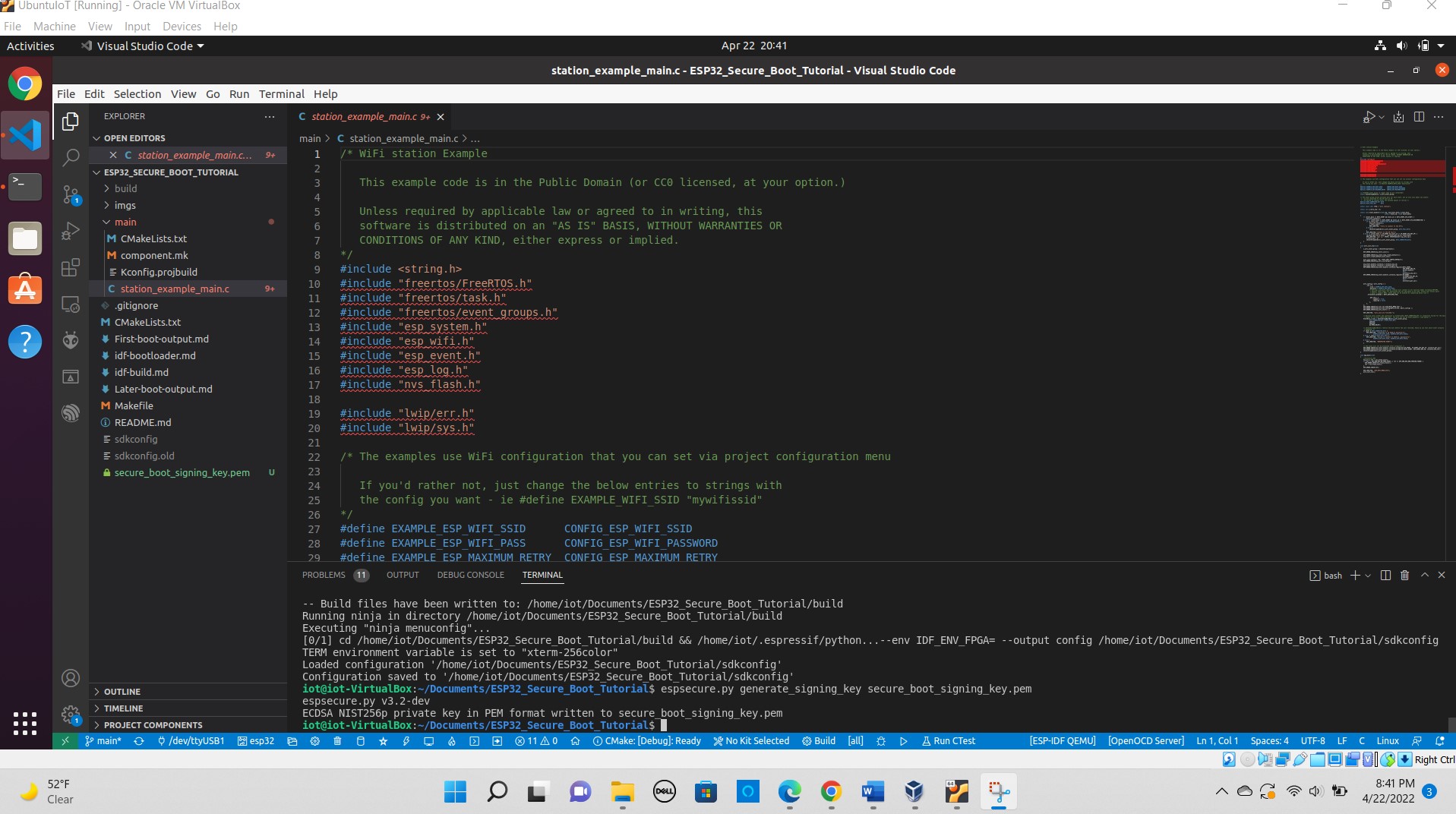
1. Include a screenshot of configured Secure Boot. (1 point)



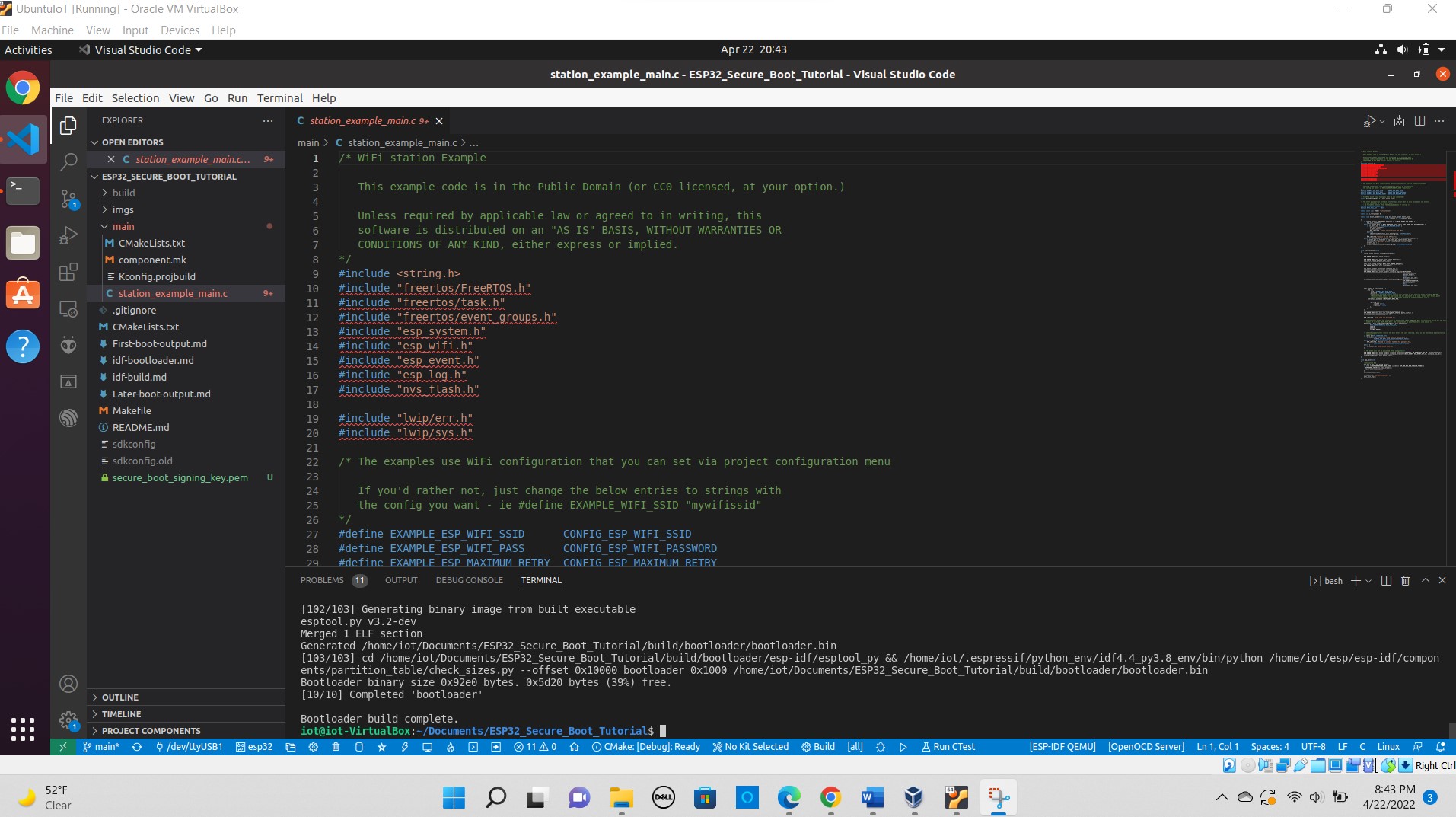
1. Include a screenshot of configured Partition Table. (1 point)



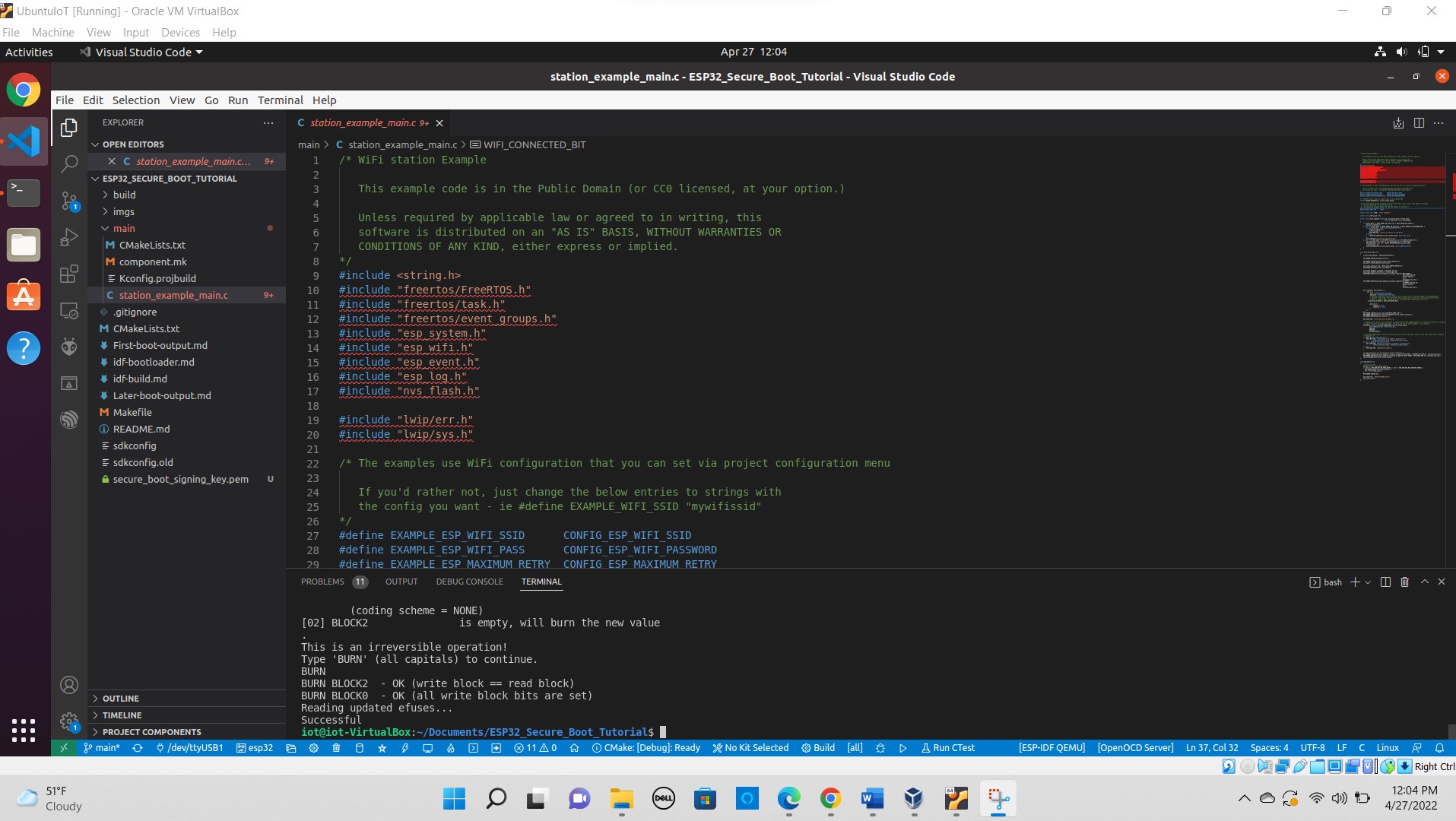
1. *Step 3.* *Generate secure boot signing key*. Include the output from the command generating secure boot signing key. (2 points)



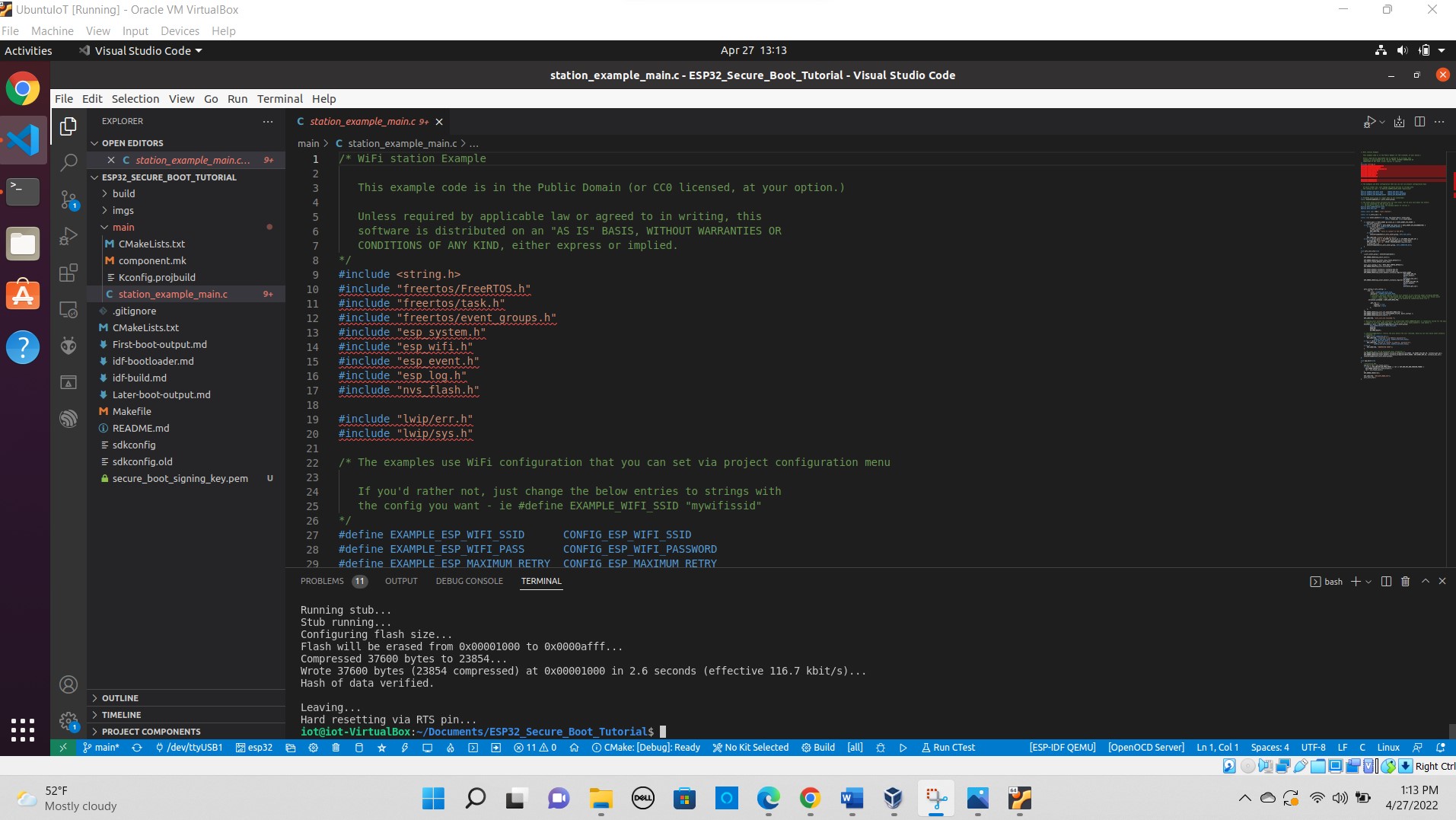
1. *Step 4. Build bootloader*. Include the output from building bootloader. (2 points)



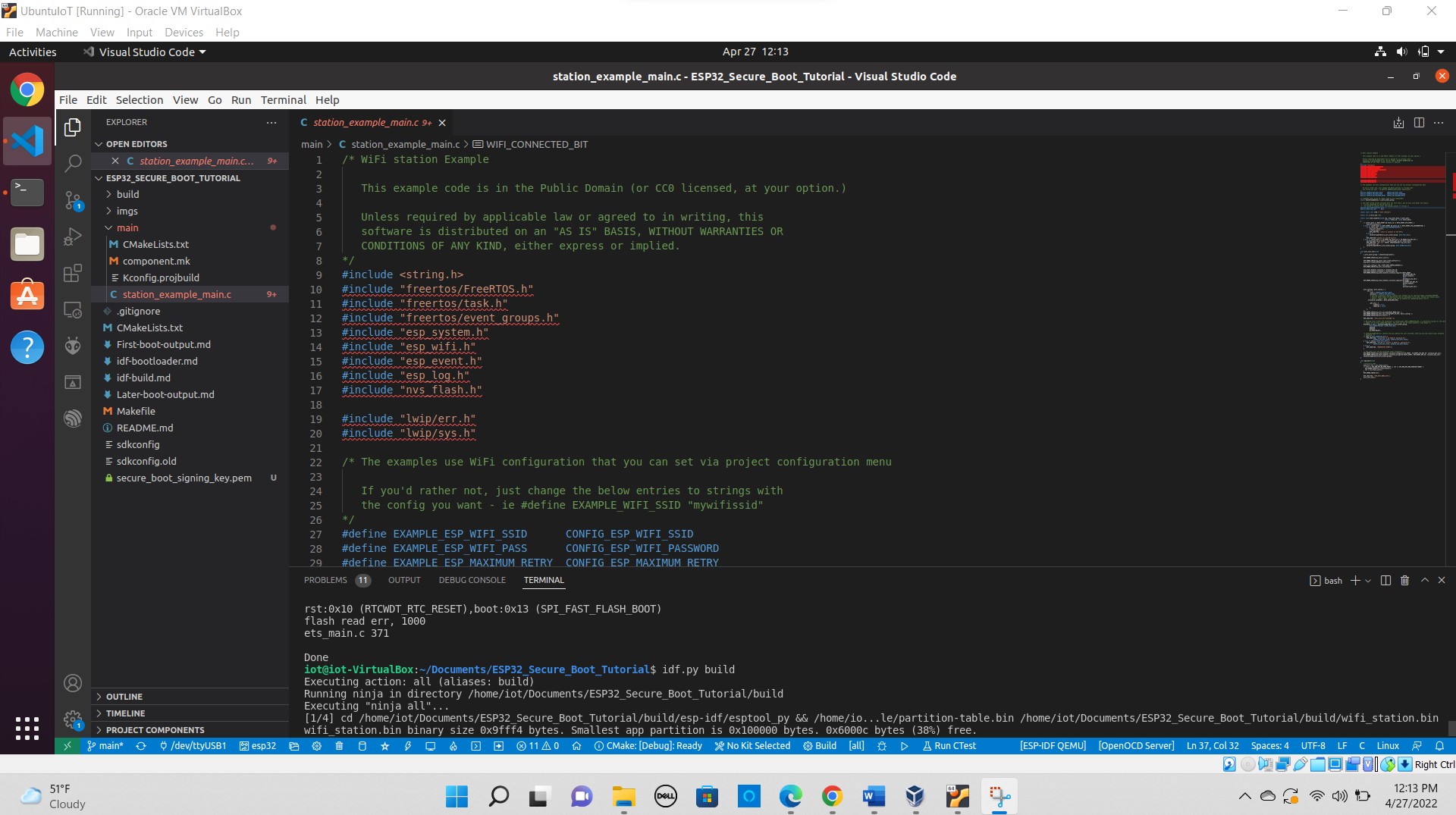
1. *Step 5. Burn secure bootloader key into eFuse*. Include a screenshot of burning secure bootloader key into eFuse, (2 points)

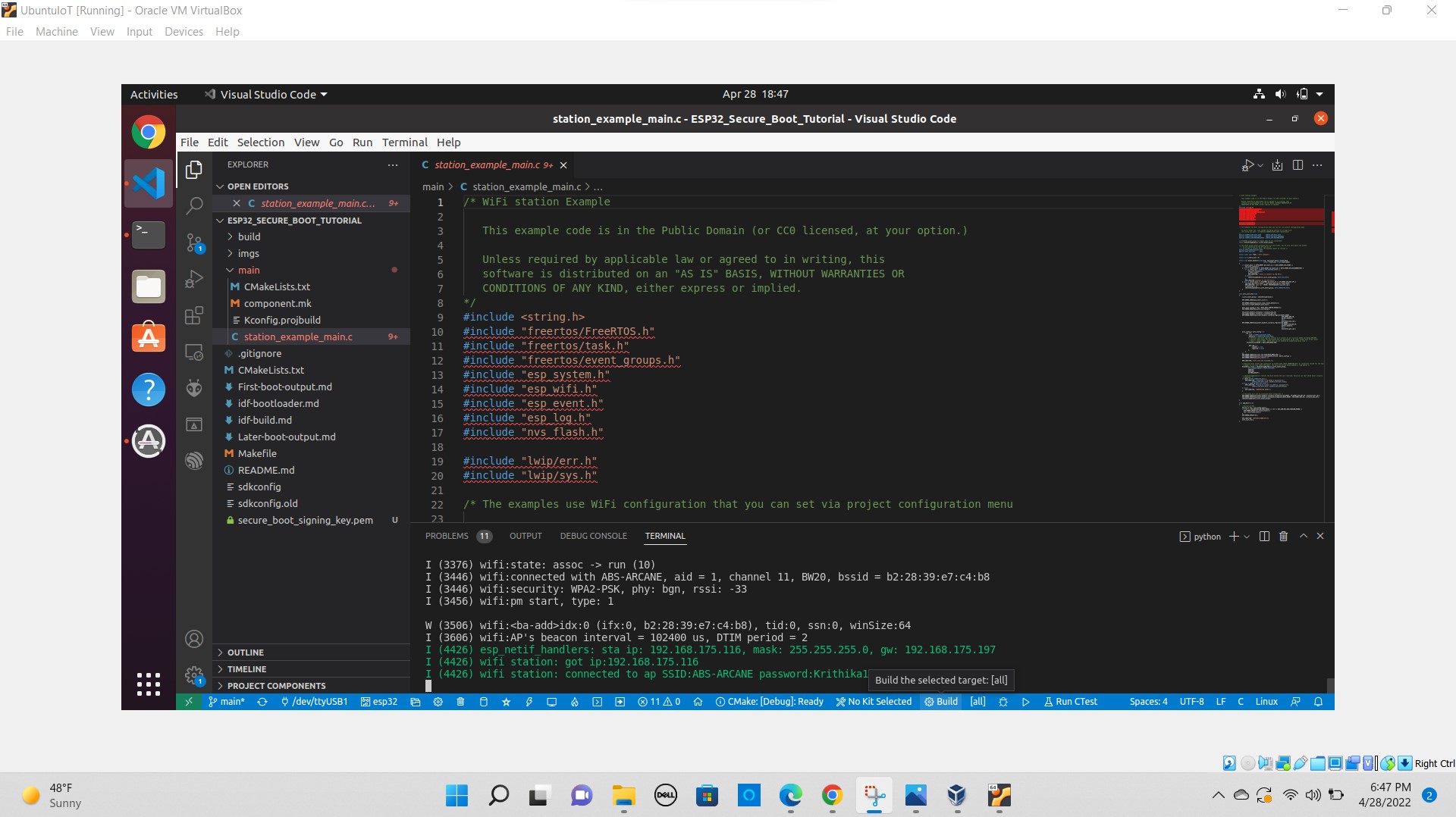


1. *Step 6. Upload the bootloader*. Include a screenshot of uploading the bootloader. (2 points)

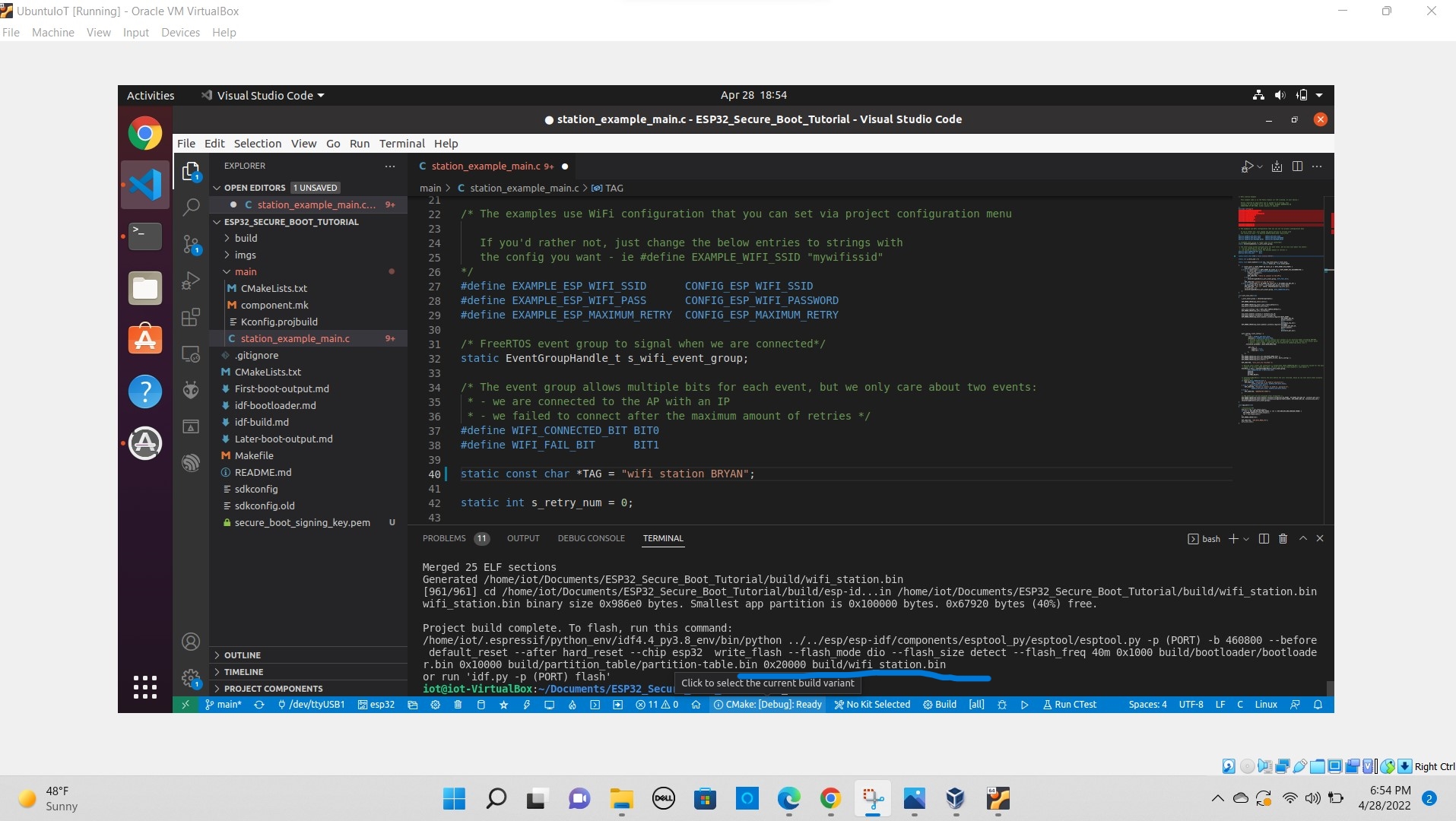


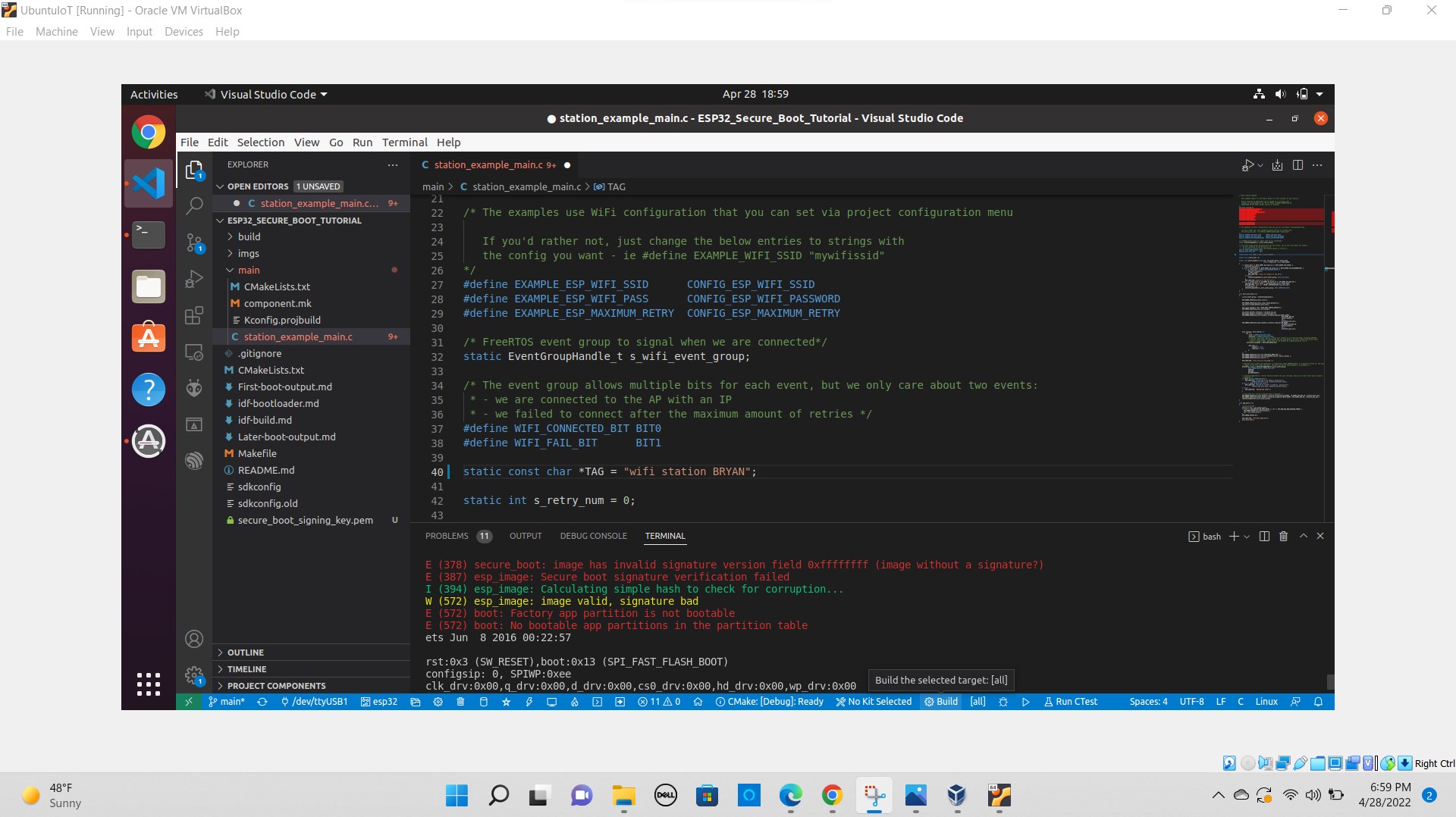
1. *Step 7. Flash app and others*. Include a screenshot showing secure boot works. (4 points)

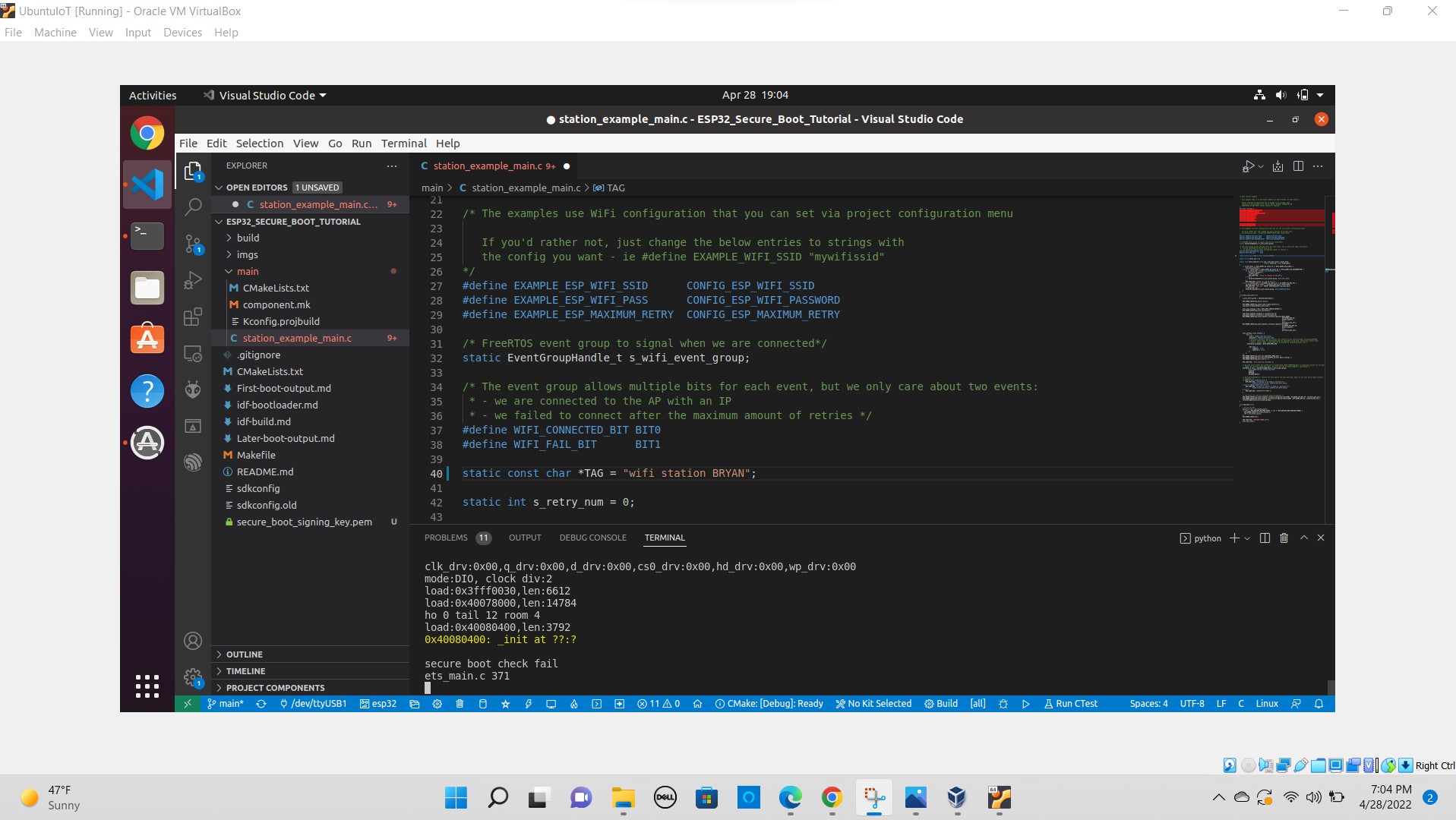




1. *Step 8. Change application code*. Include a screenshot showing changing the application will fail secure boot. (4 points)







1. (1 BONUS POINT) After the tasks above are done, each student fills out this very short anonymous survey (1 minute) about this lab<https://ucf.qualtrics.com/jfe/form/SV_4U6YfaoAhug7834>

DONE