Secure Boot Lab experience

General description:

In this lab, you are required to implement a secure boot process on the board. You need to use a **Timer** which has already been covered (in other courses) and also the **DWT** register. When the board powers on and starts running the booting process it should check at certain points in time the values of your **Timer** and the **DWT** register and compare the current values of that with the stored ones. If they are the same the booting process can move on, if not there should be an error and stop the system from booting any further.

The Data Watchpoint and Trace(DWT) unit is a debug unit that provides watchpoints and system profiling for the processor. The DWT unit contains several counters that can count:

- Clock cycles
- Folded instructions
- LSU operations
- Sleep cycles
- Interrupt overhead
- Cycles per Instruction(CPI)

This unit has multiple registers of interest. However, for this particular implementation, two register will be used: DWT_CTRL and DWT_CCYNT.

On the one hand, the DWT Control Register(DWT_CTRL) is used to enable the DWT unit and it has an address of 0xE0001000. You should be able to set the proper value of this register in orden to enable the DWT. Further information in: https://developer.arm.com/documentation/100734/0100/Debug-for-ARMv8-M/Other-debug-functionality/DWT#:~:text=for%20your%20feedback-,DWT,are%20triggered%20by%20processor%20events.

On the other hand, the DWT current PC sampler Cycle count Register(DWT_CCYNT) is used to count the number of core cycles, it has an address of 0xE0001004 and can only be read. This count can measure elapsed execution time. Further information: https://developer.arm.com/documentation/ddi0337/e/System-Debug/DWT/Summary-and-description-of-the-DWT-registers?lang=en#BABFDDBA

Another important thing is that with the purpose of working with the debug mode and with the DWT, the Debug Exception and Monitor Control Register(DEMCR) should be used as

Co-funded by the European Union Erasmus+ La Région Auvergne-Rhône-Alpes

well. This register has an address of 0xE000EDFC. To activate debug exception and monitor control, one has to set the bit 24 of the DEMCR to 1.

More information about DWT register:
https://developer.arm.com/documentation/ddi0337/e/ch11s05s01

Task 0: Opening the project

To ease the implementation of the lab experience an empty project was created for you on github that you can clone and use.

The project can be cloned from the following link:

https://github.com/balintbujtor/OSES_project.git

Use the “empty_project” branch that only includes the function skeletons that you will need to use.

Task 1: Timer

Create a function that will use a **Timer** to measure the time passed at specific points in time during the booting process. You should look into the booting sequence and determine the best points and most vulnerable ones where you want to measure this time. In the end, you store these values.

Discuss with your classmates or one of the professors the possible vulnerabilities of this approach.

Task 2: SysTick timer (optional if you have time)

Create a function that uses **SysTick Timer** to measure the time and to sophisticate the booting process. You can also see and note the differences between the values that you get in Task 1 and Task 2.

Co-funded by the European Union
Erasmus+

Task 3: DWT

Note: For getting values of the DWT register you need the board!

Create a function that reads values of DWT registers at different points in time during the booting process, and stores those values.

Compare and discuss the differences and similarities of the DWT approach and the other approaches

Look at the general description for more information about this register.

Hints:

- Check the bsp.c file and search for the pointers definitions of the three registers(DEMCR, DWT_CCYNT, DWT_CTRL) and work with these pointers.
- Read the links and analyze which bits have to be set in order to properly configure the system.
- You should create 4 functions:
    - Function to reset the DWT counter to 0
    - Function to enable the cycle counter
    - Function to disable the cycle counter
    - Function to read the value of the cycle counter
- The registers have a size of 32 bits
- To activate the debug exception, one has to set the bit 24 of the DEMCR to 1.
- To enable the counter, one has to set the bit 0 of the DWT_CCYNT to 1.
- To disable de counter, one has to set the bit 0 of the DWT_CCYNT to 1.

Task 4: Create a function for all these values and place them at the right places in the booting process

Create a function that takes functions from tasks 1, 2, and 3, runs them, then stores their values in an adequate way. Implement this function in good positions during the booting process finding the most vulnerable places that attackers might use.

If you make any changes in your code or in the functions that you have written the values of the various checks are going to change. Why does that happen?

Co-funded by the European Union
Erasmus+
La Région Auvergne-Rhône-Alpes

Task 5: Function that checks read values to stored one and gives permission to secure boot

Create a function that checks values read from the function created in task 4, then compares it to the “correct” ones.

Does it matter if the correct values are measured from the SW debugger or from the HW? Why?

Is one “correct” value enough to compare to? Why?

Did you implement a secure boot process on the LandTiger board? Why?

Discuss these questions with one of your classmates or with one of the professors

Attacks (self-evaluation):

As an additional exercise, you can try to implement some basic attacks like changing the size of the stack, and see if your secure boot implementation can detect these intrusions properly.

Extra exercise (optional, if you have time):

Sophisticate your secure boot functionality by adding other different checks, like reading the SP value.

Proposed solution:

You can find a proposed solution on the following link:

Use the “development” branch you can find our proper solution.

Co-funded by the European Union
Erasmus+
La Région Auvergne-Rhône-Alpes