Muhammad Izham Bin Norhamadi  
B032020039

LAB TEST 2 MALWARE ANALYSIS AND DIGITAL INVESTIGATION

1) The type of file of the sample

- Executable and Linking Format (ELF)

2) The tool use to analyze the sample

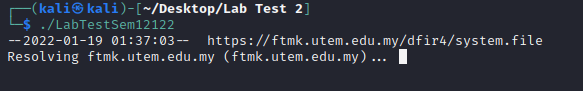
- Ghidra

- edb debugger

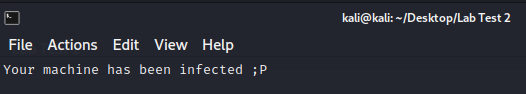
- IDA pro

3)The behavior of the sample do

1. Immediately after executing, the sample tries to resolve the connection to ftmk.utem.edu.my to get system.file



1. Then, the sample clears the terminal and the message “Your machine has been infected ;P” was shown



4) The traces of the activity the sample do

- The sample will attempt to resolve connection to ftmk.utem.edu.my, when the machine does not have connection it will still print “Your machine has been infected ;P” after some time.

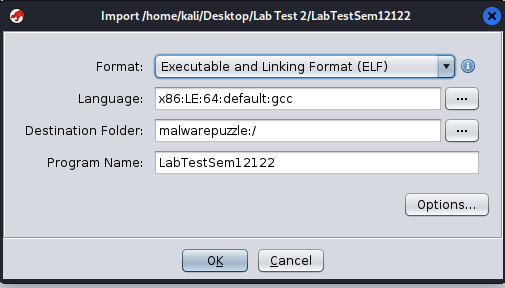
-The sample has filestream calls which can output characters into a file

5) The Two Flag you can get from the sample file

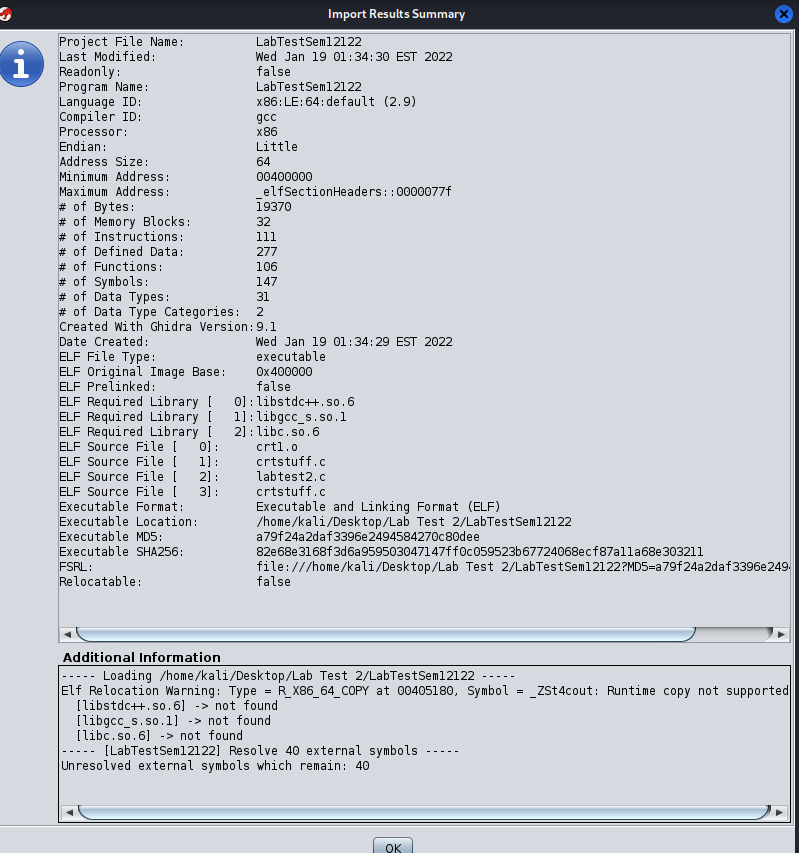
BITS3453<labtest2\_flag\_one>

Static Analysis

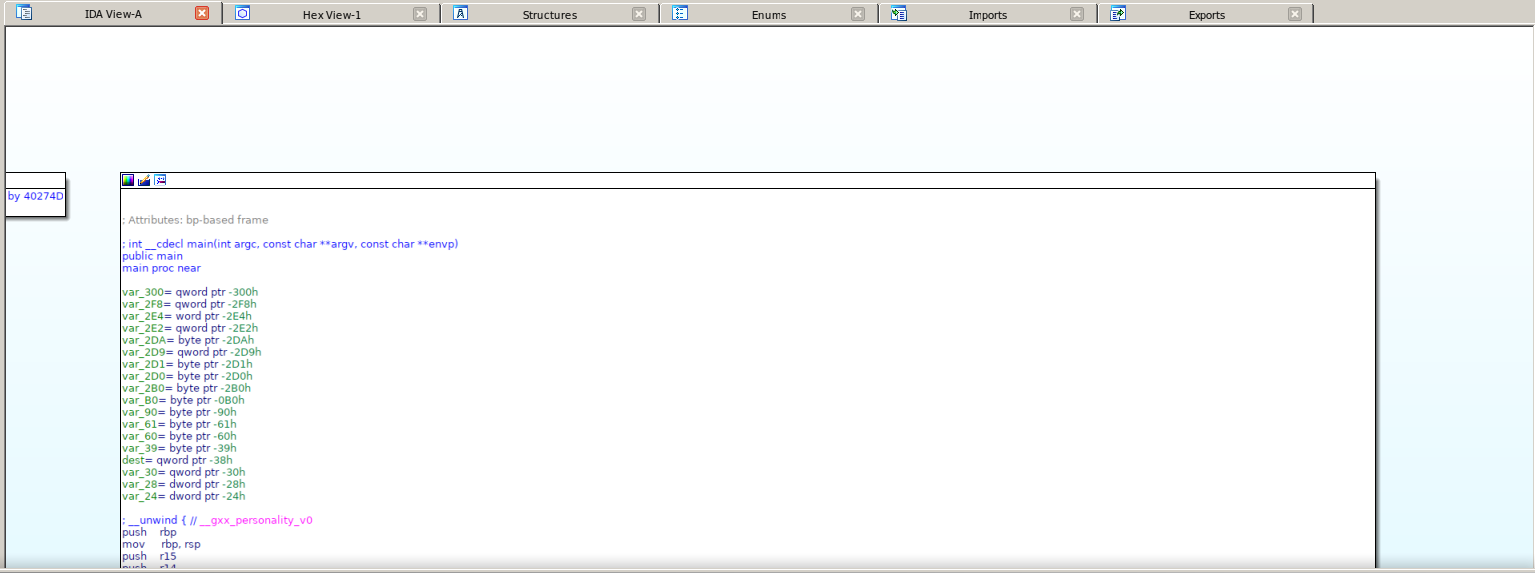
1. Import file to Ghidra and IDA pro



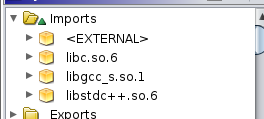
2. Check import result summary for details of the ELF



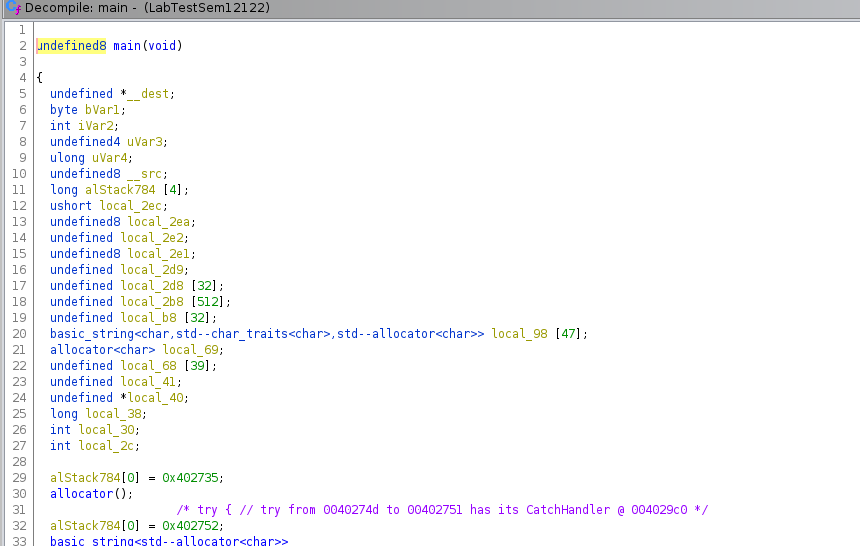
3. IDA pro has IDA View to view the structure of the ELF



4. The imports has libgcc which means the ELF was written in C++ language



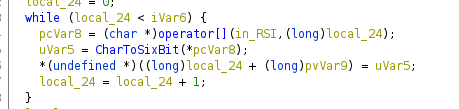
5. Go to the main function to analyse the process of the ELF with the decompiled to C language window on Ghidra



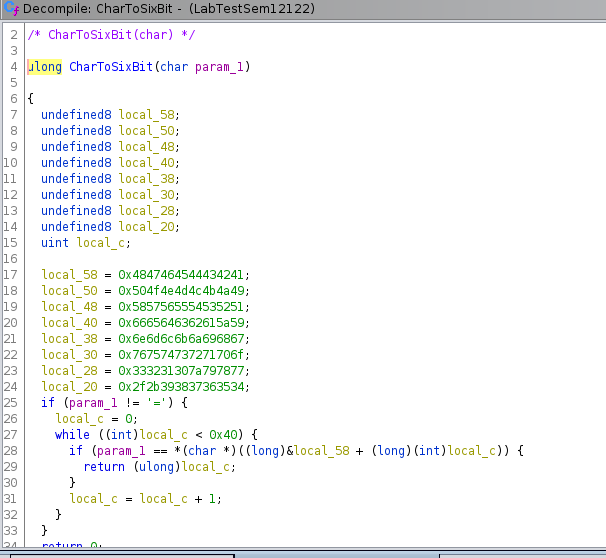
6. Then, the main function calls the Decode function with integer parameter



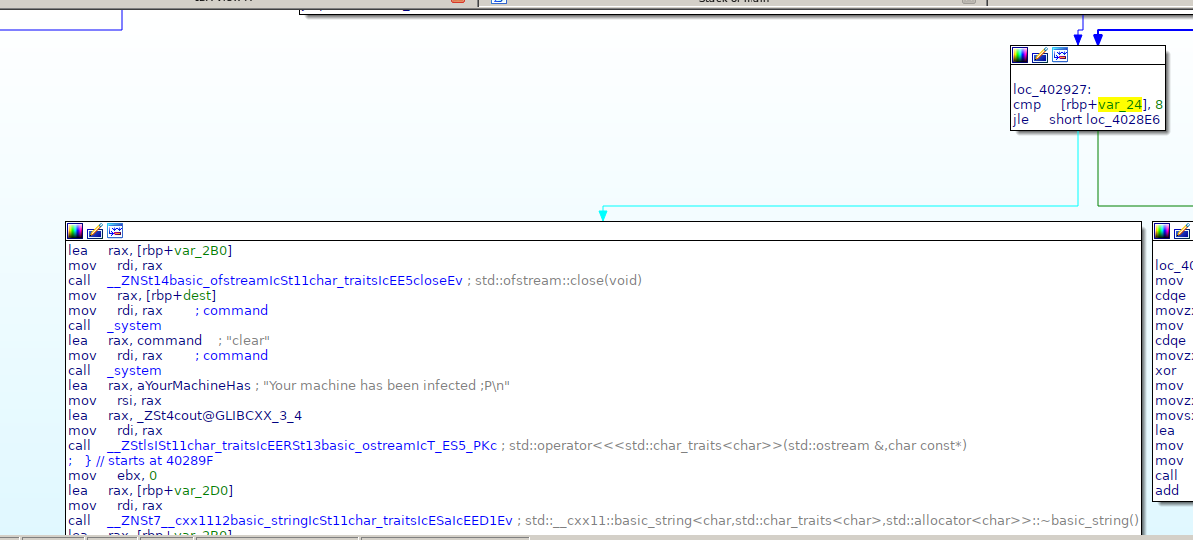
7. Then the function calls CharToSixBit function in loops



8. CharToSixBit function encodes word lengths on multiple of 6

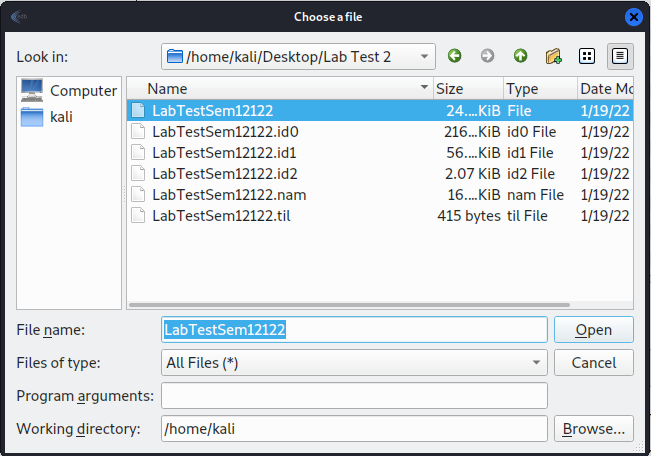


9. In main function, the executable loops rbp+var\_24 (with value 0) with 8 and adding its value by one, calling outstream function then prints out “Your machine has been infected :P”, and the program ends.

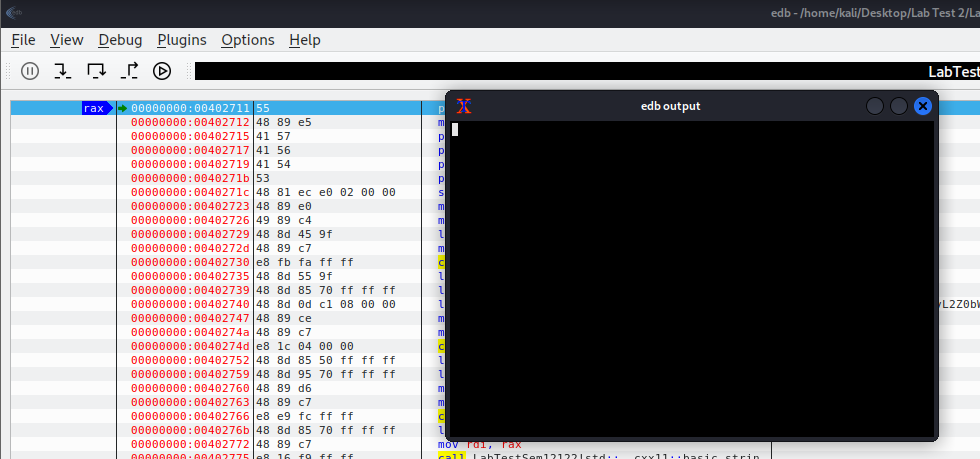


Dynamic Analysis

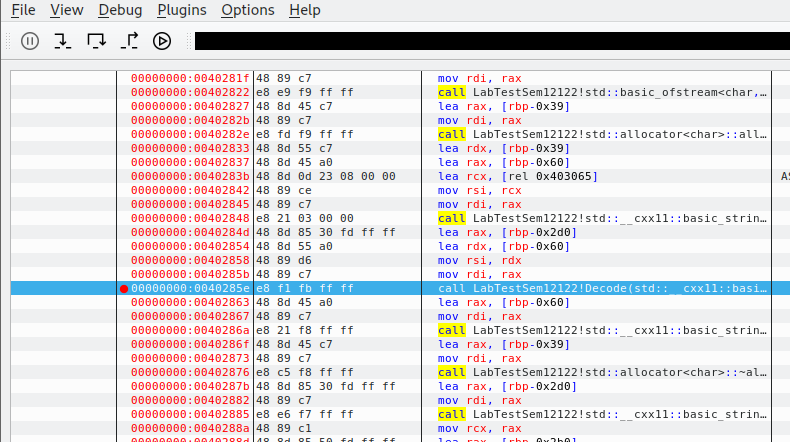
1. Open the ELF file in edb debugger



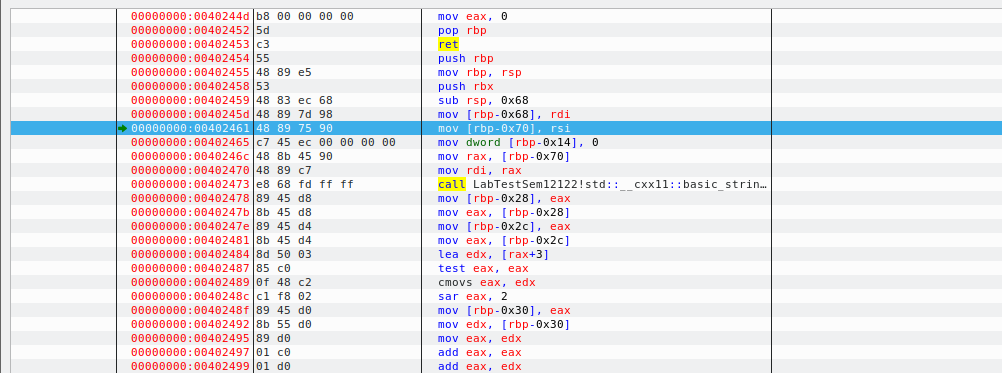
2. Press play once and analyse any notable functions of the program



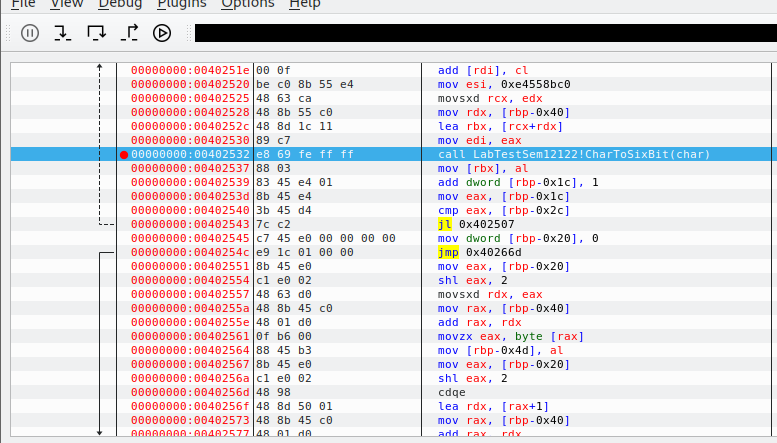
3. Toggle breakpoints on notable function such as Decoder



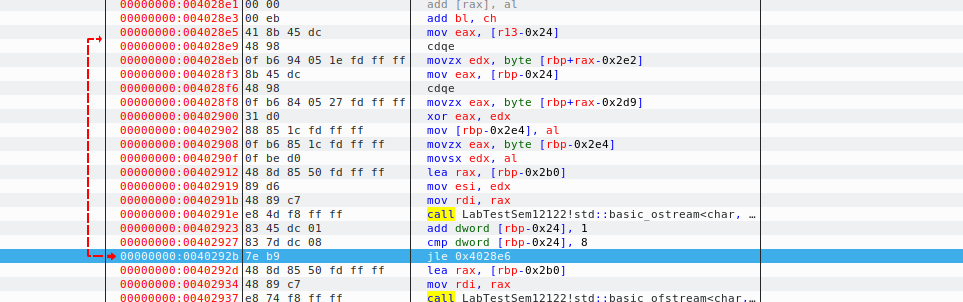
4. Step into Decoder function



5. Toggle breakpoint on CharToSixBit function and step into function



6. Back in main function, the program loops output stream until rbp-0x24 value is more than 8



7. Then the program prints out “Your machine has been infected ;P”

