ITX8060 Malware 2

Jaanus Kääp's Lab

Kaan Sadik KARADAG
156328IVCM

1- Introduction

In this document, the reversing process of “homework.exe” is described.

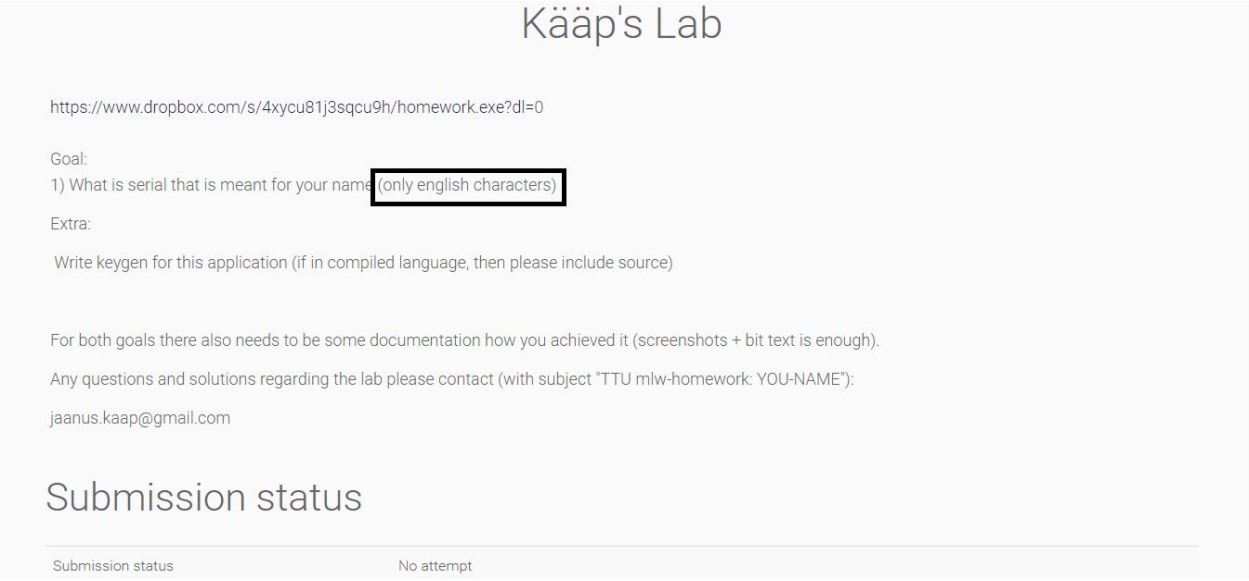
The tools that were used:

- Immunity Debugger
- Windows 7 VM (Obtained from Microsoft site)
- VirtualBox
- Python (2.7)

2- Reversing Process

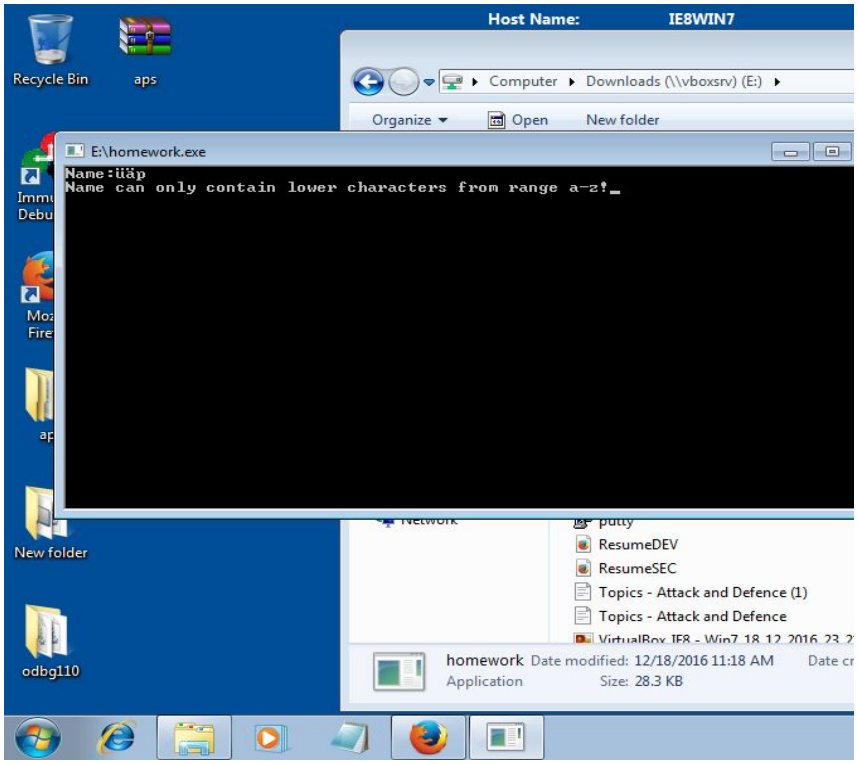
As being attended to the lab given by Jaanus Kääp, it was expected to have some sort of string manipulation.

The first hint was actually in task description:



It can be assumed that the input and the whole string manipulation would be on English characters. From this point, numerical values can also be eliminated.

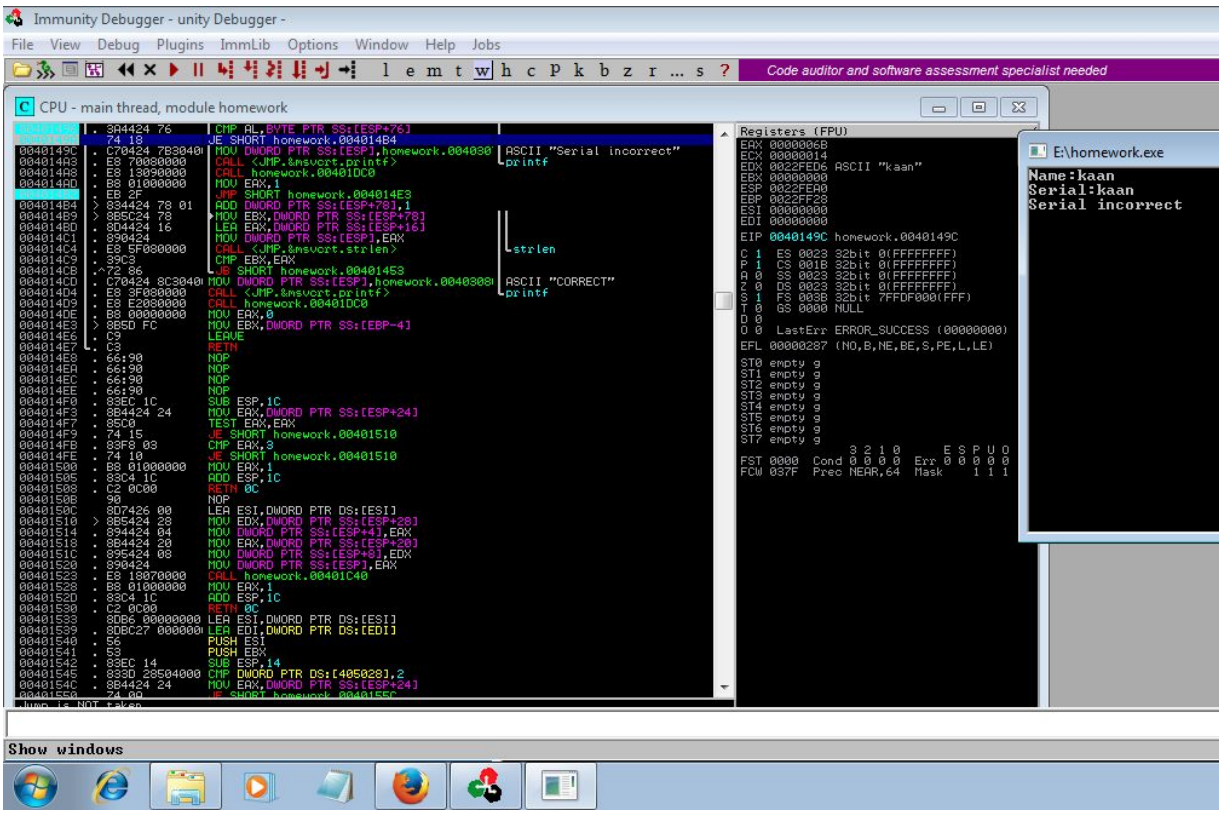
In order to have better understanding of input and output, some range values were tried.



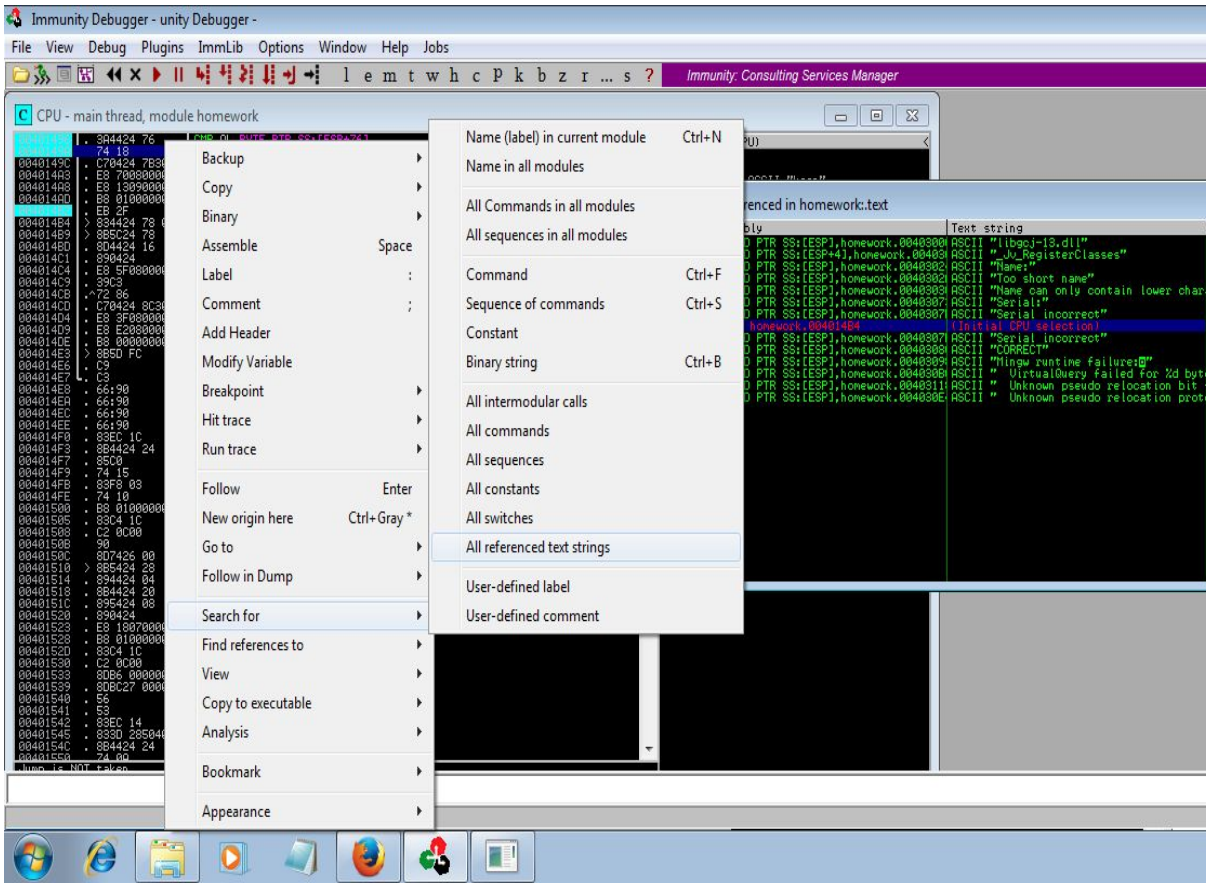
From here it was clear that:

- No Numerical Values
- No special Characters
- Just Lower case English Alphabet [a-z]

The first look to the code in debugger:



In here nothing can be derived. So, the first thing to look for is strings. Using “All referenced text strings” under “Search for”, some more logical stuff can be identified. This actually led the way to break the code apart.

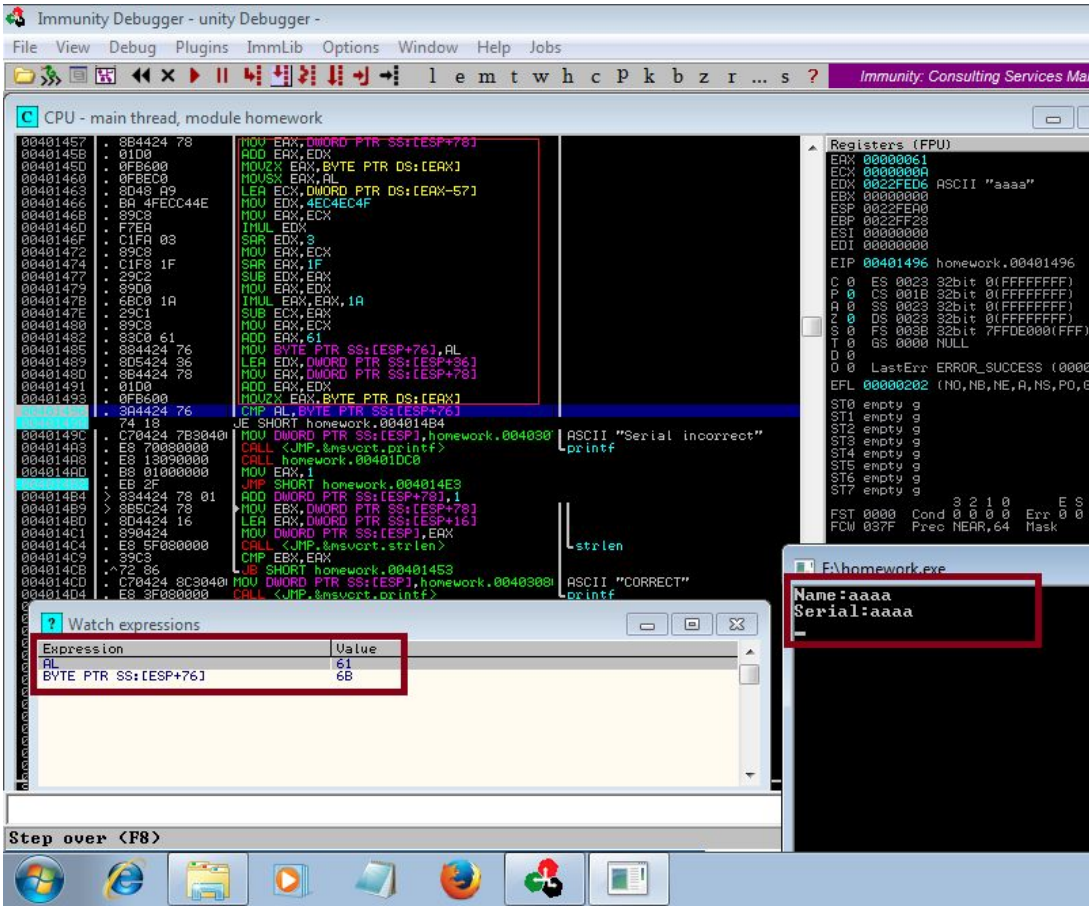


The fun part begins when the actual algorithm can be found before the compare (if) statement. After finding the part illustrated in the picture below, it was identified that:

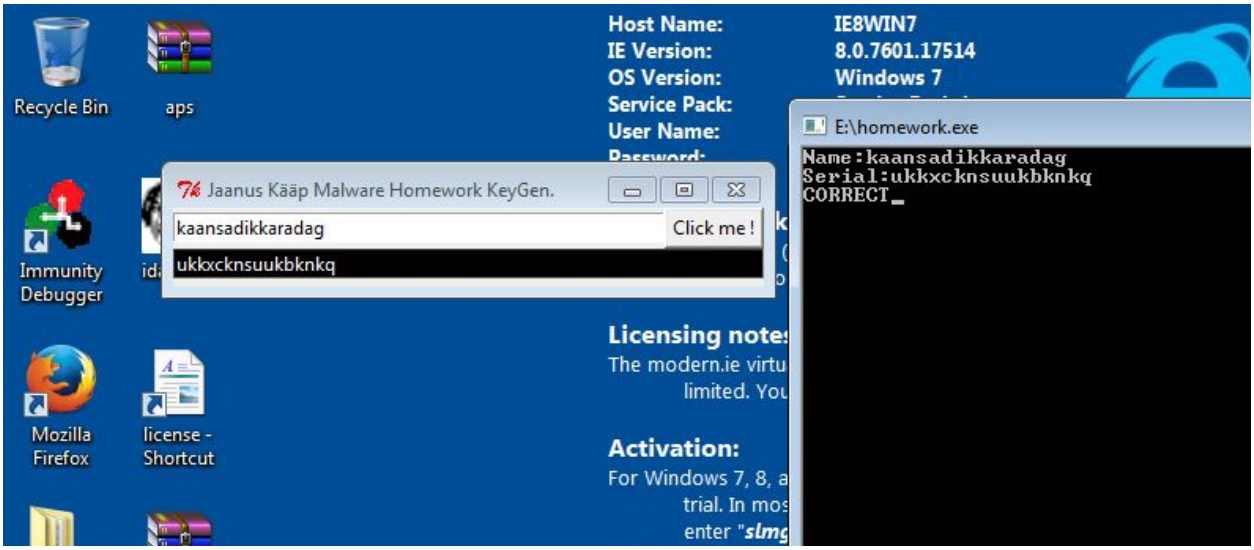
- The program loops exactly the same time as input length. So if the input is “kaan”, it will loop 4 times. This indicates there is some kind of manipulation char by char.
- When I googled “SAR”, “MOVZX” and “MOVSX” with “IMUL” instruction, so many pages related to “bit shifting” algorithms came up. From here, it can be understood that there could be a shifting algorithm like having bbbb when input is aaaa with 1 index shift.
- Using the lecturer’s advice, input “aaaa” was used to identify the algorithm. As they can be seen in expression watchtable, AL (lower 8 bits of EAX) was being compared to that area and for every “61” => a, the value thats being compared is “6B”=> k. After testing with 2-3 letters, it can be identified that

  List englishletters;
  For char in inputstring:
  Serial += englishletters[char.index + 10]

  The serial consists of the chars, shifted 10 times forwards in English alphabet in lower characters.



3- Conclusion



I've created a Keygen using Python. Notes about the Keygen was written in README file.
As a bonus, I've tried it to look and SOUND like a real Keygen:).

I personally thank you for the great lab we've held. I've always wanted to improve myself in reverse engineering and take a MSc Thesis subject from it. I've learned great things and will use these skills in the soonest CTF event.