# Project 2 Part 2: Passwords

## Executable 1:

The answer: NaaZKLNihXNbcwcwxirNYpIr

I decided to use the first executable that was provided in SysLab to test my mystrings program. After a few iterations, I noticed some strings that were displayed: “Congratulations!” and the string prior was “Sorry! Not correct!” Because of their proximity, I assumed it the password could be located closely because I thought the congratulations and sorry statements could be the result of a conditional. Luckily there was a character sequence right above the “sorry” line was the password. I tried the string and it was the correct one. To confirm my answer, I disassembled main. I first looked at fgets and I noticed that my input was being stored in %edi and then a few lines later was something that looked like a comparison followed by data loading, a comparison and a jump not equals. I added break points at those stages to see if I could get any useful information. I saw that “repz cmpsb %es:(%edi),%ds:(%esi)” was a comparison of my input and some other data, so I examined the %esi register and it contained the answer I had received earlier- confirming my answer.

**Executable 2:**

The answer:

I decided to try mystrings again on the second executable, but that was not fruitful. Next, I noticed the various time functions so I began to examine them and I saw the day and month so I decided to try them, but none of those were the password. I could see that my input, stored in %ebx through the test call but %eax was set to my input after fgets but then overwritten with “”. I set a break at the function call to 0x80483b9 because it is the call directly before the “test” call (which I assumed to be the comparison because it jumps to print functions). I noticed that the function was called within main and it processed 20 steps so I tired passwords of a length of 20, but those did not work. I tried to disassemble this function but it said “no function contains program counter for selected frame.” So I then tried to see if $eax was manipulated within the 0x80483b9 function, it printed out EDT, EST, EPT and EWT. Once I saw EST I tried the current time, but that was also incorrect. I was unable to figure out this password.

**Executable 3:**

The answer:

I first tried to set a break point at main but it did not recognize main, so I then used my mystrings program to try and get more information about the program. I did not see a main, but I noticed \_\_libc\_start\_main which looked promising. Once I googled it I realized that the function preformed initialization methods, essentially an entrance into the program. I set a break point at this function and once I ran it I noticed that the input needed to be at least ten characters (including the new line) for the input to be processed. I recognized that the password needed to be longer than ten characters for the comparison, so I tried basic ten character strings (i.e. aaaaaaaaa, 111111111, etc.) to see if anything notable occurred. In doing so, I noticed that the program exited with error 024, so I went back to the mystrings output to see if I could try and find where the error was occurring an I thought it could possibly be the tolower or getchar. I then tried all capital letters to see if I still received the same error and I did. Next I set a break point at getchar and disassembled it. I was unable to set any break points at any calls within the getchar function (possibly because the file was dynamically linked?), but I did notice that the getchar function was called the string length number of times (similar to chomp) which makes me think that the program is going a by character comparison (possibly counting the number of occurrences of a certain character?). Next, I used the “info registers” to see which registers were being used; $eax, and $edx seemed the most promising but each examination. I was unable to determine the correct password. I would guess that the tolower function might be malfunctioning, but most likely: the getchar function is not working properly (possibly not removing the new line which would make any string comparison false).