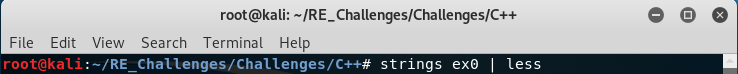
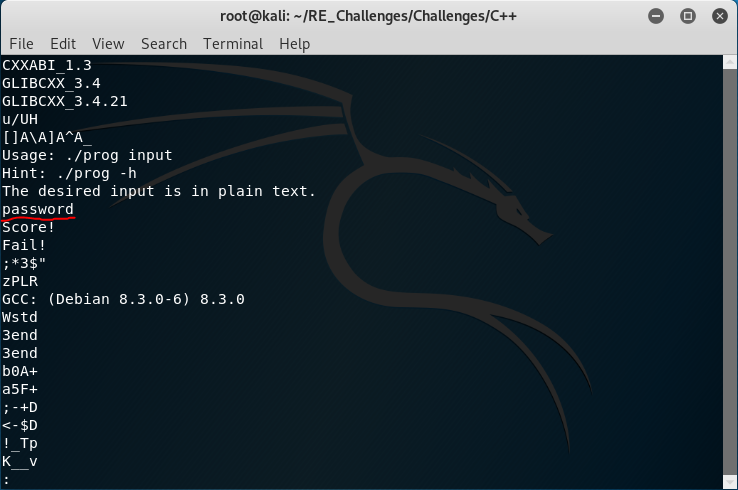
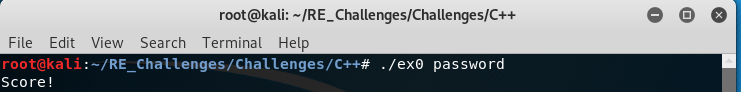
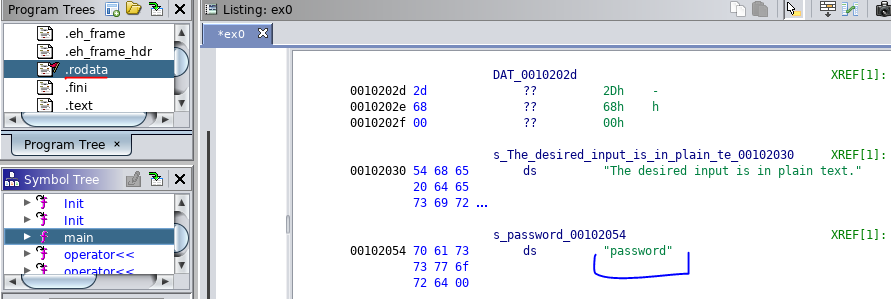
***Ex0***

According to the help message in the challenges we are looking for a string in plain text. There are many approaches to solving this, sometimes even with the same tool. We will explore a few of these in this help document. To begin we start with the most straight forward, the “strings” command.

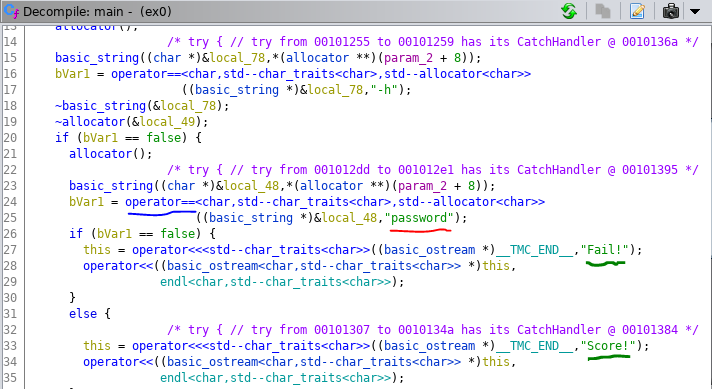
An effective way to use this command is by “piping”, sending, the output of the strings command to the less program. Less will display its input in a paged representation allowing you to freely navigate through the output.

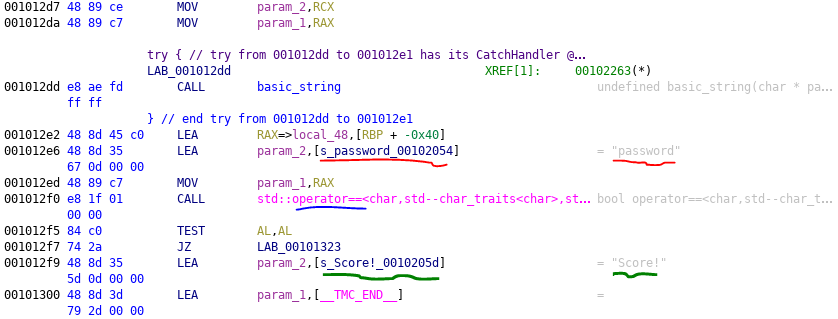
Scrolling down with the arrow keys we see the word “password” right before our desired “Score!” phrase.

That was rather straight forward, now let’s crack this binary open with ghidra. To run ghidra execute the ghidraRun script found in the ghidra directory in your home directory.



In the .rodata section of the elf, underlined in red, we find data that is only ever read from and never written to. This is commonly used to reference strings that were hard coded into the program. If we look into this elf section, we will find all of the strings used inside main such as our “Score!” message and even the correct input “password”, underlined in blue.

Now let’s navigate to the function main and look in the decompilation window. This probably looks liked mangled garbage, but it serves its purpose. Scrolling through we will find the “Score!” and “Fail!” message, underlined in green, in conditional blocks. Right before this we see an equality, underlined in blue, being checked between a stack variable and the string “password” with the result of the equality check being stored in the variable that toggles which conditional block is executed afterwards.

Finally let’s look in the disassembly view of the main function. We find ourselves in a similar situation to the decompiled view, first identify our “Score!” message, underlined in green. We find a string “password” that is being loaded into the address of a parameter to a call to the equality operator, underlined in blue right before our “Score!” message.