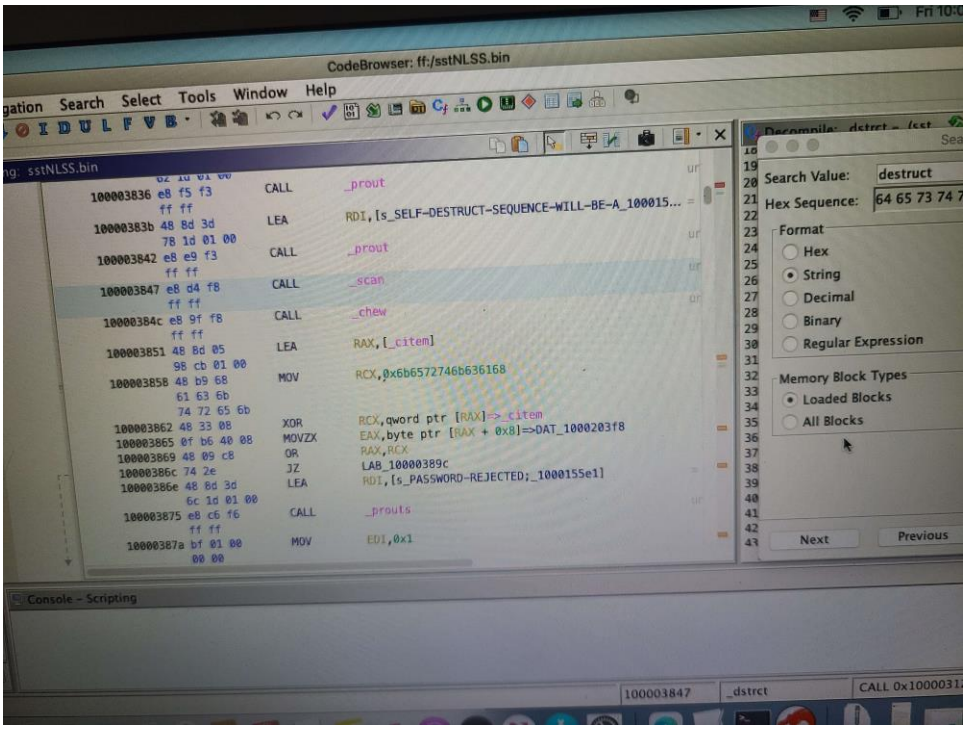
Nathan Luy

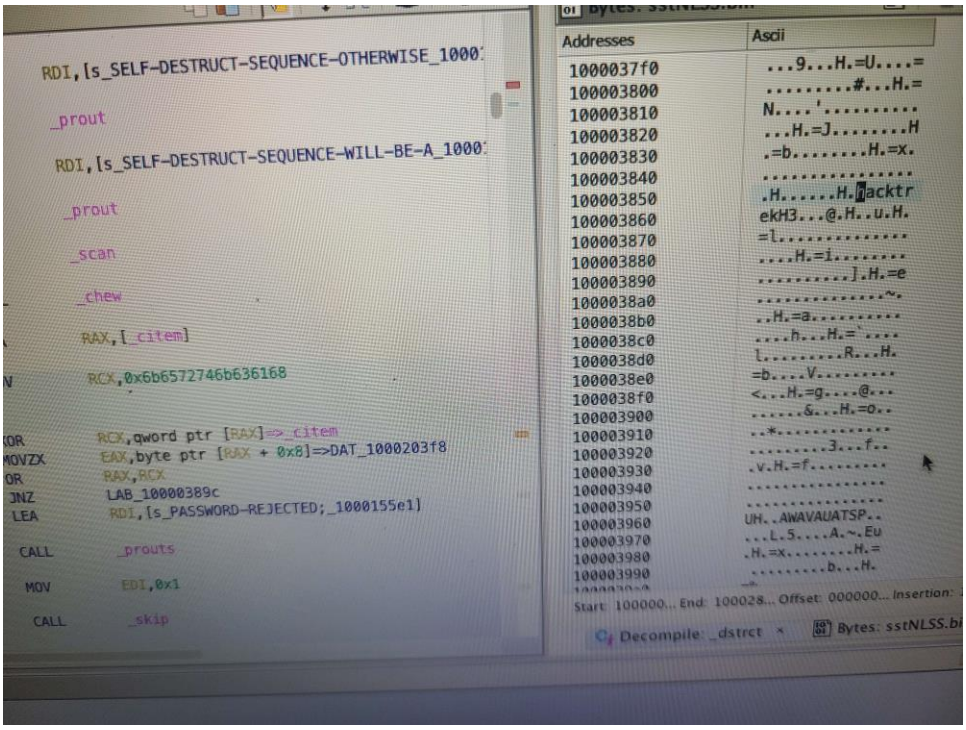
016881525

478 Malware Lab Write Up

Code Comments and Answered Questions

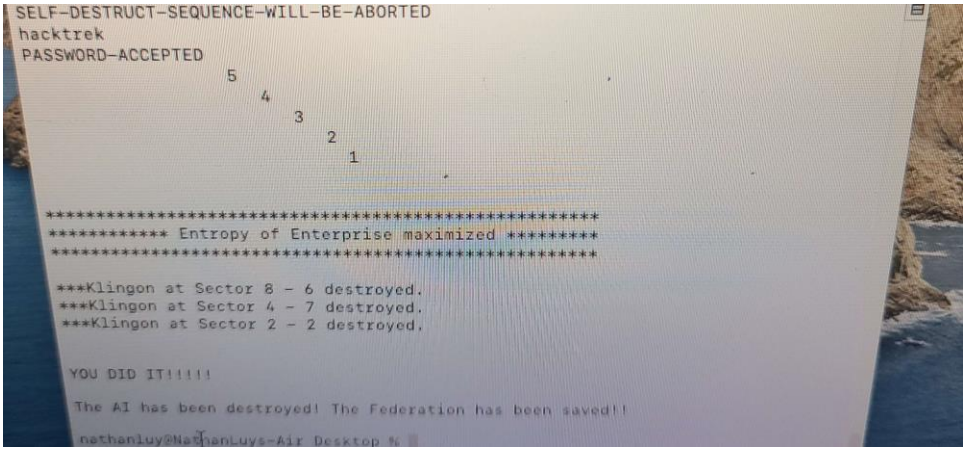


Using the search memory tool and the xref(or green links) I was able to pinpoint where the password for self destruct was located by typing in words related to it.Like password,destruct,ect. 100003858.

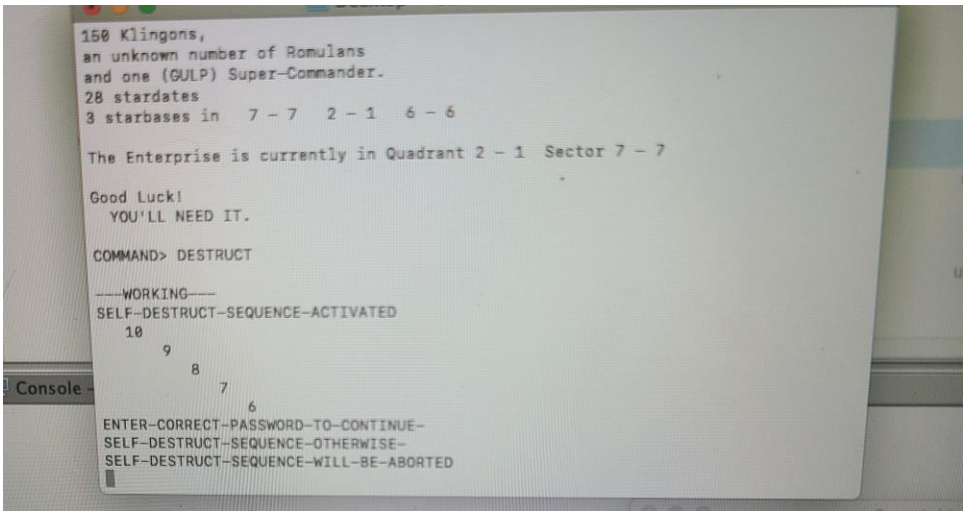


Eventually was able to figure out that the password was hacktrek by putting the bytes view into ascii.

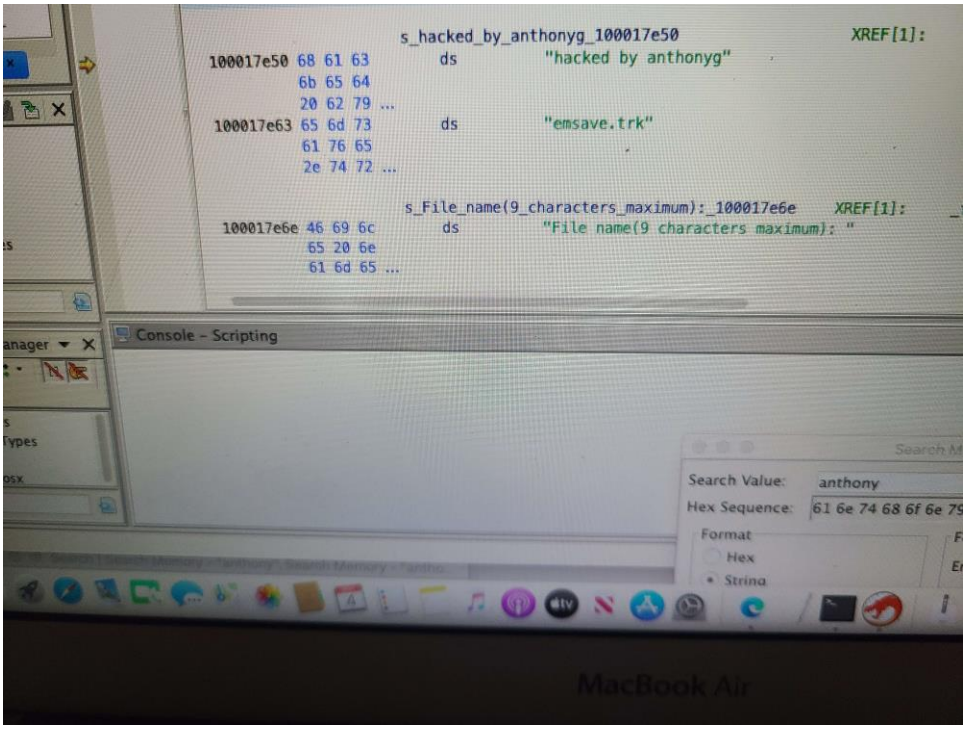
Also on 10000386c changed the assemble instruction of jz(jump if zero) to jnz(jump if non zero) in order to bypass the password check and go to the destruction sequence.



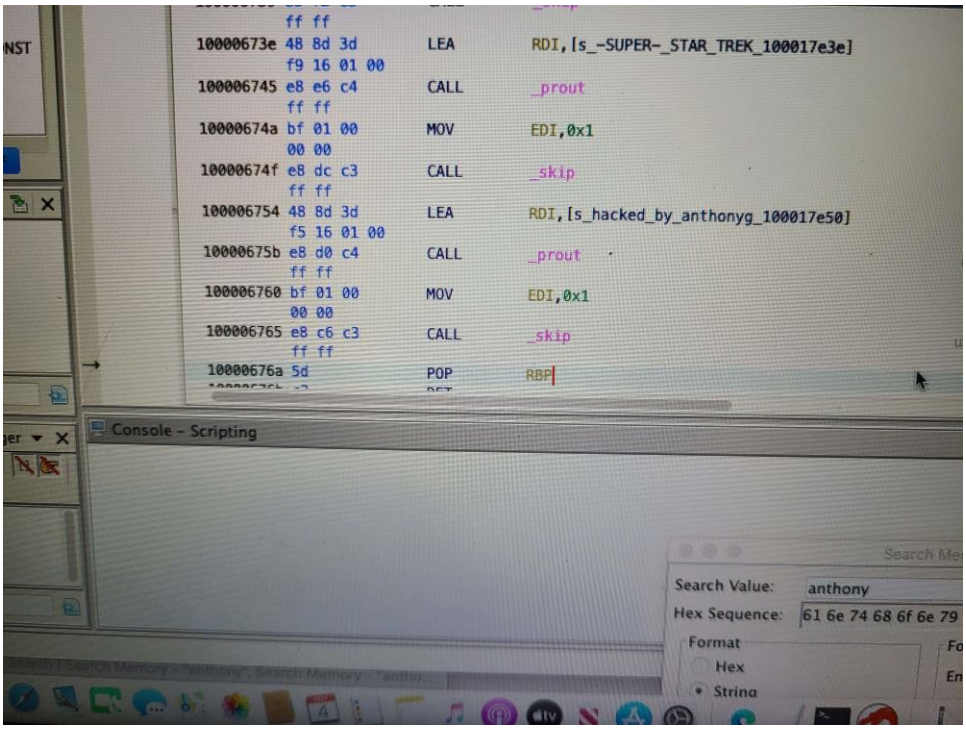
Put it into the game to check.

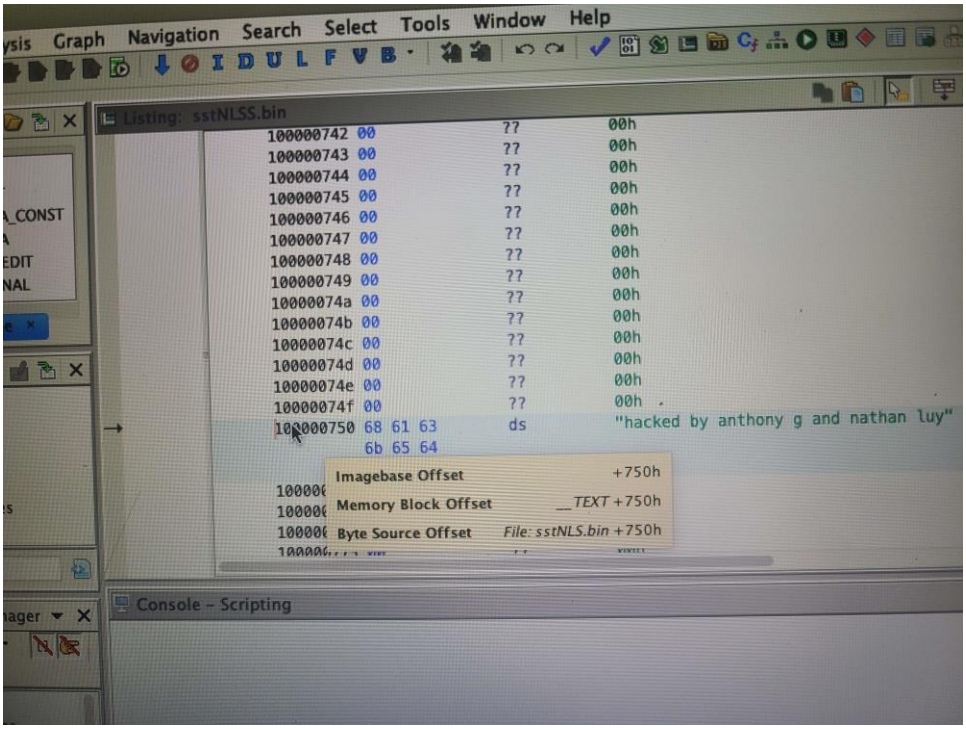


This is when I patched the instruction to bypass the password check.

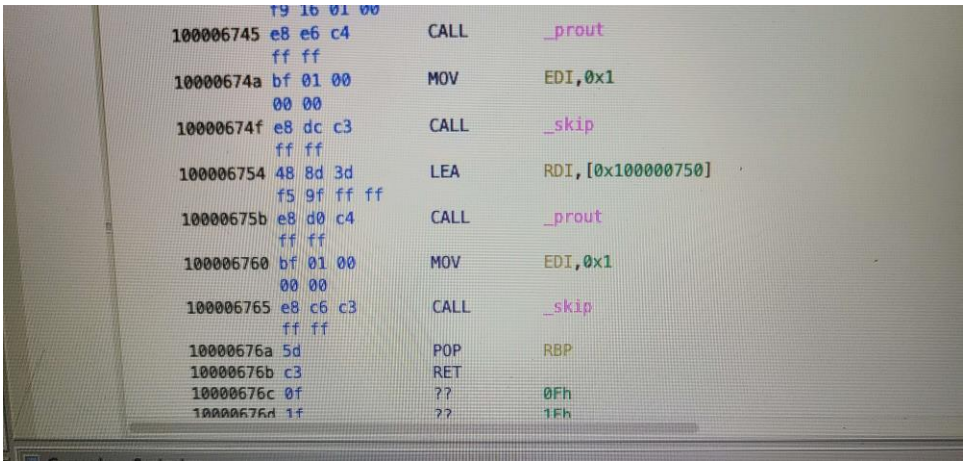


Found where the string of the hacking credits where using is stored 100017e50 using the search memory and the picture below is where the credits are printed out 100006754 and I got there by pressing the green XREF on the side of where the string was stored.

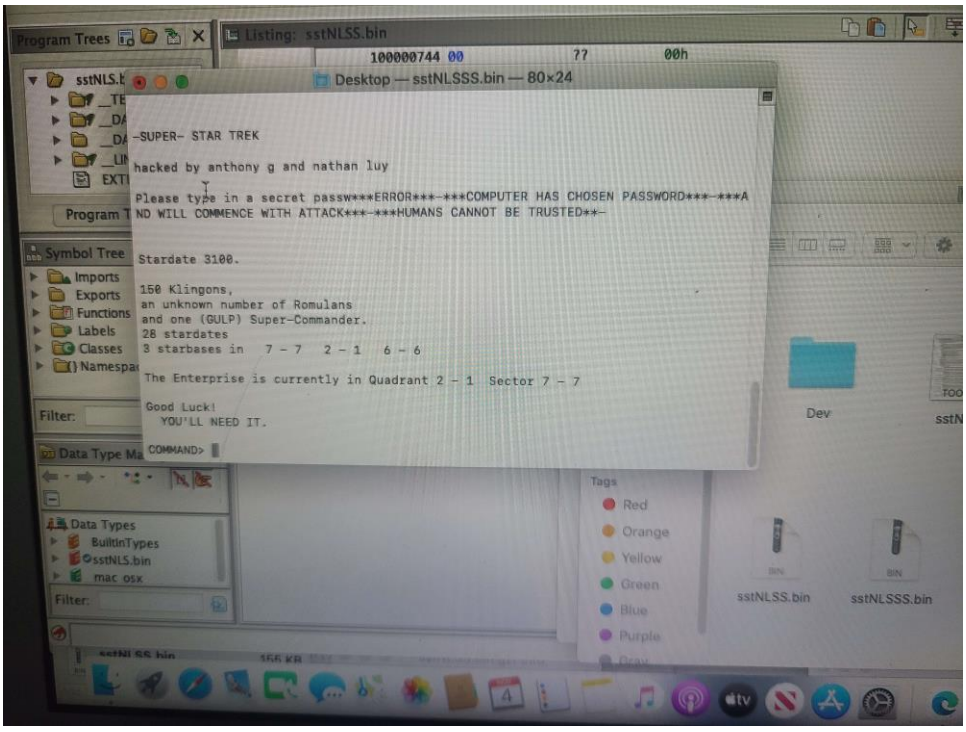




I created a string to store the new hacking credits at 100000750.



Changed the hacking credits that where being printed out to my hacking credits . RDI,[0x100000750].



1) Changed 10000386c, 100000750, 100006754.

2) 10000386c now has the assembly ins jnz instead of jz. 100000750 contains the new hacking credits. 100006754 now uses the new hacking credits to print out.

3)Password was hacktrek.

4)Bypassed the password by changing the instruction at 10000386c from jz to jnz.

Project was pretty interesting once I figured out how to use search and was able to change the view of whatever I wanted on the byte view tab on the right side.(Mainly to ASCII and Hex view)