1. Drew Sadler
2. Read and Write
3. File size of 1
4. 00000000: 6161 6161 6161 6161 6161 6161 6161 6161 aaaaaaaaaaaaaaaa
   00000010: 6161 6161 6161 6161 6161 6161 6161 6161 aaaaaaaaaaaaaaaa
   00000020: 6161 6161 6161 6161 6161 6161 6161 6161 aaaaaaaaaaaaaaaa
   00000030: 6161 0a   aa.
5. I can tell it's an ascii text file due to ~ being marked as 7E and that is the beginning of the hex value in the ascii table for that character, as well as the 20's for the spaces in the beginning, also the many values of 0 to represent the empty spaces.
6. The null character of value 0
7. 7f45 4c46
8. 00000310: 0100 0000 0000 0000 2f6c 6962 3634 2f6c ..../lib64/l
   00000320: 642d 6c69 6e75 782d 7838 362d 3634 2e73 d-linux-x86-64.s
   00000330: 6f2e 3200 0000 0000 0400 0000 1000 0000 o.2.
   This is possibly calling to the hopper linux library to process the elf file of the executable
9. Type: EXEC (Executable file)
   Machine: Advanced Micro Devices X86-64

   Type shows the file that is processed and what the file is ending in, in this case a (.exe)
   Machine showing what type of machine I'm using either a 32-bit or a 64-bit with in this case running on a x64 system
10. Section to Segment mapping:
    Segment Sections...
     00
     01   .interp
     02   .interp .note.gnu.property .note.gnu.build-id .note.ABI-tag .gnu.hash .dynsym .dynstr .gnu.version .gnu.version_r .rela.dyn .rela.plt
     03   .init .plt .text .fini
    The .text segment starts of the offset of 03
11.   94: 0000000000401126   37 FUNC   GLOBAL DEFAULT   14 main
    The 94th byte has main in it.