Revbw.exe Write-up

- The architecture of the revbw.exe file is i386. I obtained this information from running objdump on the file and providing the -f flag. I learned that the revbw.exe is an executable for Windows by running a simple file command on the executable.
- According to ghidra, the compare for the shell command is called at memory address: 01339cc5. I found this by first utilizing the search function within ghidra. I selected program text and selected the checkbox of defined data values. Then this jumped to the shell command text in the exe and after hovering over the reference in the binary it opened a window of where the strcmp occurs in the assembly code.
- The Windows API that is utilized is the Processthreads API function CreateProcessA and the function used for the shell is cmd.exe. I found this by using the search result for shell. I clicked the link to jump to the shell code. I then scrolled down until I found external calls and the call was to Kernel32.dll and CreateProcessA. I went into the decompiled code and stepped back 3 bytes from the stack pointer to Local_1258 and found the value set at that memory address and local_1254 to get the application and command parameters.
- The get_ip compare for the function is executed at the memory address: 0133a0a9. I found this by once again using the search functionality of ghidra to search the data values in the program text. I then hovered over the reference and it showed where get_ip's comparison is done.
- The command line used to get the ip address is powershell.exe. The command to get the ip address is a url invocation which is encoded in base 64. The command is: $r= Invoke-WebRequest -URI "https://ifconfig.me/ip"; $content = $r.Content;echo $. I got this invoked url request by using the base64 -d command. I got the powershell.exe from walking through the decompiled code and finding the memory address that showed the two parameters for the powershell call.
- The base address I found to be the address: 01337000. The entry point address is address: 013384a0. The main function is entry. The base address and entry point address were obtained by running objdump on the exe and using the -x and -f flags. Then the main function was function FUN_01338ab8 was obtained by looking at the ghidra symbol table for entry and then stepping through the decompiled entry code.
- The memory address that this is called at from the main function is address: 01338bef. The start address of this function in memory is address: FUN_01338549. This function takes one parameter and the parameter is an int of size 4.
- MessageBoxA is called at address: 01338a72. The address of the function itself is address: FUN_013389cf. The text contained inside the caption input parameter is "Hello Universe". The text contained in the message parameter was "This is a really long message This is a really long message This is a really long message This is a really long message This is a really long message This is a really long message." I went into the function trees and found the messageboxA function. I then changed local_f2, local_f7, and ppc_var3 to char**. I then got the addresses of the Base64 encoded strings and the base64 inside the function itself and used Base64 -d to decode the caption and the second part of each string. This produced the message and caption.