**EXPERIMENT NO. 11**

|  |
| --- |
| **Student Name and Roll Number:HARSHIT PARASHAR 18CSU081** |
| **Semester /Section: CS-V -CYSA1** |
| **Link to Code:** |
| **Date: 10/11/2020** |
| **Faculty Signature:** |
| **Marks:** |

|  |
| --- |
| **Objective:** Analyze an exe using Immunity Debugger |
| **Outcome:** To practice disassembling and modifying binaries. |
| **Problem Statement:** Examine the 00000.exefile using Immunity Debugger |
| **Background Study:** Immunity Debugger is a powerful new way to write exploits, analyze malware, and reverse engineer binary files. It builds on a solid user interface with function graphing, the industry's first heap analysis tool built specifically for heap creation, and a large and well supported Python API for easy extensibility. |
| **Algorithm/Flowchart:** |
| **Code(Solution):** |
| **Sample Outputs:**  First we will download the 00000.exe file and try to launch it and after it ask for Launch code we will enter 1 and it will show a message which is basically a error message .      Now we will start Immunity debugger and will load our first file named as “00000.exe” We will find for the word JMP and JNZ which are highlighted in green and we will right click on both of them one by one and will assemble them.      After clicking on assemble we will find a address there we will remove it and will use “NOP” there and same on JNZ    Once its done we will see NOP listed so many times on screen.        Then we will right click on same screen and will use “Copy to Executable” and then will click “All modifications”.  After that we will have another screen popped up there we will right click again and will save the file by name “00000NEWexe’  Once file is saved run that file and we will get the answer |
| **Question Bank:**   1. Differentiate between Immunity Debugger, x64Dbg and OllyDbg.   Ans - Immunity Debugger is a powerful new way to write exploits, analyse malware, and reverse engineer binary files. It builds on a solid user interface with function graphing, the industry's first heap analysis tool built specifically for heap creation, and a large and well supported Python API for easy extensibility.  X64Dbg is an open-source binary debugger for Windows, aimed at malware analysis and reverse engineering of executables you do not have the source code for. There are many features available and a comprehensive plugin system to add your own.  OllyDbg is an x86 debugger that emphasizes binary code analysis, which is useful when source code is not available. It traces registers, recognizes procedures, API calls, switches, tables, constants and strings, as well as locates routines from object files and libraries. |