

ASSIGNMENT-3 README

November 11, 2019

1 Introduction

The aim was to convert the WebAssembly instruction trace to X86 instruction trace with the type value and address.

1.1 Procedure

- We used a flask server to allow the WebAssembly to send the hook values generated to a server because the browser was not able to read the files from the system due to security issues.
- The browser generated the instrumented information on the console of the browser.
- This data was sent to the server in batches of 1000 each in JSON format.(A batch consisted of 1000 instructions)
- The server received requests till the requested number of instructions are instrumented or entire file is instrumented and each batch was also written to a name.tar.gz file on the fly.
- The server also performed the task of mapping the WebAssembly instructions into the equivalent X86 instructions by using a mapping defined in a csv file.
- Also we calculated the program counter earlier in order to reduce the overhead so that it doesn't need to be recalculated for every instruction.
- We assumed that there are a fixed number of register in the X86 architecture and used them in local/global indexes and when these indexes exceeded 16 then we pushed them into the memory (assuming that the memory started from 0x0) and generated instructions accordingly.

1.2 How to Run

- Extract the file and execute the run.sh file with first parameter as the number of lines and the second argument is the program path.
- Sometimes the browser may require refreshing, in that case refresh the browser and then run the script again
- Also while running the script, file not found error may come. Also ignore those errors since we require removing the old files if they exist.