Syed Muhammad Abubakar Attiq

CONTACT

+92-324-9724291

s.abubakar2019@outlook.com

@Syed-Abubakar

EDUCATION

FAST NUCES, Islamabad

MS(Computer Science) 2020 - 2023

University of Gujrat, Gujrat

BS(Electrical Engineering) 2014-2019

SKILLS

- Programming and Languages
  - C/C++
  - Python
  - OS(Linux)
  - Bash
- Miscellaneous

OpenKE, Data Visualization, Network Analysis, Malware Analysis, Digital Forensics, Technical communication, technical documentation, Ghidra, Flare VM, Beowulf Clustering, OMP threaded code, Data Manipulation.

SUMMARY

A result-oriented and passion driven individual, keeping a level head and thinking out of the box to aid in finishing assigned tasks. Proficient in programming languages like C, C++ and python with the potential to learn other languages as well.

WORK EXPERIENCE

- Full stack web development Intern
  - Company: Red Buffer
  - Duration: 6 weeks.
  - Experience:
    - NodeJs + ExpressJs
    - MySQL database server
    - ReactJs Frontend

THESIS

- An Explainable approach to memory forensic based detection of Malware using Machine Learning.
  - A model Agnostic approach utilizing a surrogate classification model.
  - Generates Explanations in the form of if-else rules.

Projects

- Recreating pipe functionality to accept input on CLI (C/C++)
  - Recreated pipe functionality as on Linux.
  - Input accepted through CLI as an argument.
  - Process forks and execs as required by the argument.
  - Dependent on the argument output of one system call to be redirected as input to another system call.
- List based graph implementation to return number of cuts and crossings (C++)
  - Implemented list structure/adjacency matrix to satisfy graph.
  - Nodes and edges fed from an input file requiring file handling.
  - Returning number of cuts and crossings based on the list/matrix.
  - Creating a placeholder node to delete any node input for deletion.
  - Returning number of cuts and crossings for the new graph.
- Priority Queue based scheduler (C++)
  - Required file handling procedures as input file has to be fed via .txt file.
  - Parsing the input file and tokenizing as required.
  - Scheduling based on time; amount left etc.
  - Compact and optimized code.
- Achieving consensus between threads using MPI (C++)
  - Using C++along with MPI library to create multithreaded code.
  - Achieving consensus between threads to achieve terminating condition.
- Visualization based on datasets from Kaggle (Python)
  - Cleaning the data frame of redundant/Null values.
  - Analyzing the data frame based on remaining values.
  - Creating appropriate visualizations representing the answers to the research questions.
- Profile generation based on Open-Source Intelligence techniques
  - Led the team in creating a profile on an individual.
  - Used Numerous open-source tools: Spiderfoot, Dorking etc.
  - Managed to pull out personal information on the individual.
- Malware Analysis
  - Used different tools like Ghidra, PE studio.
  - Determined whether the binary was made with malicious intent.
  - Used FlareVM to analyze it statically and dynamically.