

Muhammad Hassan

+1 (540) 824-8188 | hassanaleem9@gmail.com | linkedin.com/in/hassanaleem1 | github.com/hassanaleem

EDUCATION

Virginia Tech

PhD in Computer Science - CGPA 4.0/4.0

Jan 2025 - Dec 2029 (expected)

Blacksburg, Virginia

Selected Graduate Coursework: Program Analysis & Software Testing (Advanced Topics in SE), Compiler

Optimizations, Translator Design

Lahore University of Management Sciences

Bachelor of Science in Computer Science – Major GPA **3.70/4.00**

Sep 2019 - Jun 2023

Lahore, Pakistan

PUBLICATIONS

Muhammad Hassan, Tahir, T., Farrukh, M., Naveed, A., Naeem, A., Shaon, F., Zaffar, F., Gehani, A., & Rahaman, S. (2023). "Evaluating Container Debloaters." 8th IEEE Secure Development Conference (SecDev), 2023.

<https://www.csl.sri.com/users/gehani/papers/SecDev-2023.Debloat.pdf>

EXPERIENCE

Virginia Tech - Language And Compiler Design Lab

Graduate Research Assistant

Jan 2025 - Present

Blacksburg, Virginia

- Developing **constant-time transformations** to reduce timing side-channel leakage.
- Improving **symbolic execution** through control flow merging for faster bug detection.

Aerodyne Group

Software Engineer

Jul 2023 - Dec 2024

Lahore, Pakistan

- Developed a 2D/3D annotation interface for digital-twin models.
- Optimized the 3D rendering pipeline, increasing frame rate from 15 FPS to 240 FPS..

Lahore University of Management Science

Research Assistant

Jan 2023 - Oct 2024

Lahore, Pakistan

- Developed and evaluated BloatProfiler, a benchmarking framework for Docker image debloating.
- Evaluated the effectiveness of CVE mitigation, assessed the correctness of the debloated images, and measured the achieved size reduction.

Jobture

Software Engineering Intern

Jun 2021 - Sep 2021

Remote

- Created the infrastructure to collect Job listings via APIs + Web scraping and automated with cron and AWS.

PROJECTS

Control Flow Merging: Implemented control flow merging to mitigate path explosion in symbolic execution (**LLVM, KLEE**). Reduced bug finding time from 20 hours to 4 hours for mid-to-large size programs and from 1 hour to a few seconds for smaller benchmarks.

Constant Time Code: Manipulating the control flow of a program to ensure that the code always runs in constant time regardless of the input. As a result, it is able to thwart time attacks for a large number of programs.

LLM-Based Debloating: Developed a pipeline that complements debloating techniques using Azure AI GPT 3.5, RAG architecture, and LLVM coverage analysis, resulting in improved generality and security of debloated applications.

Taint Analyzer: Developed an inter-procedural taint analyzer that operates at IR level to detect vulnerabilities.

ProdeBench: Developed and evaluated ProdeBench, a benchmarking framework for automated software debloating, comprehensive measurements on generated binaries, assessed their correctness, measured the size reduction, analyzed gadget expressivity, and evaluated debloat time for standalone tools and various compositions.

Additional projects: Audio Sentiment Analysis (MFCC, Chroma, Mel-spectrogram, Fourier); Twitter Real/Fake Mining; De-verse (blockchain-based censorship-resistant social media app); Netflix Recommender (NLP + unsupervised); LUMSApp (Android portal: grades, deadlines, CGPA calculator, forums).

SKILLS

Programming Languages: Python, C/C++, Assembly, JavaScript, Solidity, Haskell.

Security: Network Forensics, Cryptography, Linux, Shell Scripting.

Frameworks & Libraries: LLVM/Clang, KLEE, SMT Solvers, CUDA, TensorFlow, PyTorch, OpenCV, Scikit-learn
Angular, Node, Django, React.

Other Technologies: Git, Docker, Large Language Models, Program Analysis, Vagrant, Three.js, Konva.js, SQL, PostgreSQL, Figma, Matlab, HTML, CSS.