YUVRAJ SAXENA

+918679164624 ysaxenax@gmail.com linkedin.com/in/0xXA github.com/0xXA

PROFESSIONAL SUMMARY

My research interests are in Systems Security, Software Security, and Automated Program Analysis. My goal is to advance reverse engineering and vulnerability research of complex systems. I'm actively contributing to open source projects and collaborating with academic professionals. In my free time, I step outside my comfort zone by fuzzing complex software system for vulnerabilities to test myself against tough challenges. So far I've been awarded **$10000**+ by Google through Research Grants, bug bounty, and OSS rewards.

VULNERABILITY DISCLOSURE

CVE-2025-13425 – Denial of Service in google/osv-scalibr

- Discovered a remotely triggerable null pointer dereference / out-of-bounds issue in osv-scalibr's parsing logic when processing untrusted vulnerability feeds. The issue could crash automated analysis pipelines.

HONOURS AND AWARDS

- Google vulnerability research grant

SKILLS

Environments:	Linux, Android, iOS
Programming Languages:	C, C++, Assembly, Python, Java
Frameworks:	Angr, libFuzzer, AFL++, Syzkaller
Debugging & RE Tools:	GDB, IDA Pro, Ghidra, Frida, QEMU
Security Tools:	Binwalk, Burp Suite
Technical Areas:	Fuzzing, Symbolic Execution, Reverse Engineering, Code Review
	OS Internals, Mobile Security, IoT Security, Malware Analysis

RESEARCH AND TEACHING EXPERIENCE

Security Researcher — **June 2025 – Present**
Freelance — **remote, USA**

- Selected for **Meta's** invite-only **WhatsApp** Private Processing Security Research Program to collaborate with security engineers on secure messaging and to improve the security of private processing components

Arizona State University (ASU) — **November 2024 – January 2025**
Research Study Assistant — **remote, USA**

- Invited by **Dr. Adam Doupé** and **Zion Basque** to participate in a research study on LLM-aided reverse engineering
- Collaborated with researchers from **ASU**, **Eurecom**, and the **University of Padua**, applying reverse engineering expertise using IDA Pro with an LLM plugin; provided structured feedback, bug reports, and actionable improvement suggestions

Teaching Assistant — **August 2024 – September 2024**
Delhi Institute of Engineering and Technology — **Meerut, India**

- Took in-depth lessons of the Computer Security module (30 hours)
- Supervised teammates on our final year research project (practical)

Security Researcher — **January 2018 – Present**
Freelance — **remote**

- Ported **Linux Kernel** for closed-source Android devices (**Symphony**, **Intex**, **Micromax**)
- Reverse engineered display drivers from firmware for **Mediatek** and **Qualcomm** chips
- Bypassed **anti-cheat engine** in **PUBG** and security checks in banking apps
- Reverse engineered firmware of **IoT** devices (**Routers**, **Blood Pressure Meters**, **Modems**)

OPEN SOURCE CONTRIBUTIONS

Google's AOSP, Google's OSV-SCALIBR, Binwalk, Frida, Radare2, Termux, Droidimg, AFLplusplus, libssh, file or libmagic

RESEARCH

Supply Chain Security

- Discovered a supply chain compromise vulnerability in Google's OSV Scalibr OSS project. This tools is widely used for Software Composition Analysis (SCA) and automated vulnerability scanning by internal stakeholders and open-source projects, including within CI/CD pipelines. The vulnerability is currently pending CVE assignment. Evidence of the discovery and the implemented fix is available in PRs #2008, #2030, and #2105

Detection Engineering

- I have worked closely with Google's Open Source Vulnerability (OSV) team on their detection engineering project called "OSV-Scalibr", where I contributed by developing software extensions for Embedded Filesystem Extraction. These plugins aids the tool in extracting Softwre Bill of Materials (SBOMs) from virtual machine disk image and software distribution file formats like VMDK, VDI, OVA, and Android APK. These changes helped the project cover significat gaps in inventory extraction for famous file formats

OS & Vulnerability Research

- I have spent good amount of time researching the internals of Android and Linux ecosystems. I can reverse engineer and fuzz operating system components using a range of vulnerability research techniques. Most recently, I reverse engineered Android framework's **abx.jar** to implement **parser** that converts the Android Package Manager's **Packages.xml** from **Android Binary XML (ABX)** into standard XML for easier analysis. I also collaborated with **Jakub Jelen** at **Red Hat** on a grammar-aware differential fuzzing project targeting **libssh**'s' configuration parser, to help align its behavior with **OpenSSH**

Mobile & IoT Security

- I've been digging into the security of mobile and IoT devices since **2018**. This includes reversing Android apps, bypassing PUBG's anti-cheat, and analyzing firmware from routers, modems, and smartwatches. Along the way, I've built tools to make reverse engineering faster and easier

Cryptography & Network Security

- I enjoy working with cryptographic algorithms and the math behind them, like number theory, logic, and algebraic structures. I've implemented **cryptographic** algorithms by modifying existing ones or building them from scratch when needed

EDUCATION

Delhi Institute of Engineering and Technology **2021 – 2025**

Bachelor of Technology (B.Tech) in Computer Science and Engineering

- Thesis Title: Fuzz Testing of Open Source Software Systems (Grade: A+)
- Relevant Coursework: **Operating Systems**, **Cryptography & Network Security**, **Computer Security**, **Computer Networks**

PROJECTS

Google PoC

- A step-by-step guide to reproduce **CVE-2025-13425** that I found in Google's OSS project

Hacking Sharp Vision

- Documented the process of reverse engineering and exploiting a **fiber modem** firmware
- Created a **malicious** firmware by bypassing modem's integrity checks

Snapchat Emulator Bypass

- A tool to bypass **emulator detection** in the Snapchat Android app

abx2xml-go

- Developed a Go library to convert **Packages.xml** from **Android Binary XML** to a regular XML.

Binwalk

- Introduced **sdate** data type for parsing ASCII timedate strings
- Added **ZTE** firmware header parsing support
- Fixed the entropy generation bug to generate the **entropy graph**
- Improved parser **performance** and fixed minor bugs

Cryptographic Automations

- Developed utilities to automate the cipher **mathematical** calculations phase

Mkernel

- Built utilities for automating the **Linux Kernel** building process