Vinodh Balakrishna

Software Engineer | C/C++ Programmer | GNU/Linux Enthusiast

(913) 326 4029 vinodh.b.27@gmail.com www.linkedin.com/in/vinodh27 www.github.com/v1n0dh

Professional Summary

Software Engineer with around 2 years of experience in developing efficient and secure Linux firmware for embedded systems. Skilled in C/C++, Python, Rust, and shell scripting, with deep expertise in GNU/Linux environments. Specializes in system performance optimization, low-level programming, and implementing robust security solutions.

Technical Skills

Programming Languages: C/C++, Python, Shell Scripting, Java, Rust, Assembly (Basic)
Operating Systems: GNU Linux/Unix Systems, Embedded Linux Firmware
Networking: TCP/IP, UDP, SNMP, Networking Protocols
Tools and Technologies: Yocto Project, RDK, Data Structures, CMake, GDB, QEMU, OpenSSL, Wireshark, Jenkins, Git, Splunk, Vim

Experience

Comcast
Chennai, India

Embedded Linux Engineer | Platform Security Team (TPX)
Dec 2021 - July 2023

- Utilized the Yocto Project build system to design, develop, and maintain custom Linux firmware based on the RDK software stack for over 1 million CPE devices, including set-top boxes, routers, and TVs.
- Enhanced security in the custom WalledGarden Shell by implementing command logging and blocking unauthorized commands, significantly improving secure SSH access to CPE devices.
- Implemented a Tab Completion feature in the WalledGarden Shell using the readline library and a multi-threaded approach, enhancing user experience and increasing shell efficiency by 40%
- Integrated Software Bill of Materials (SBOM) functionality into the build system by meticulously tracking all software components, enhancing supply chain security and ensuring 100% transparency in the final product's software inventory.
- Enabled firmware security feature Address Space Layout Randomization (ASLR) for multiple RDKB devices by implementing position-independent code, which significantly reduced memory-related security vulnerabilities.
- Maintained and renewed XPKI certificates for TLS communications within various internal components, enhancing data protection and ensuring reliable, encrypted interactions between system components.
- Securely removed the video analytics module from RDKB devices, optimizing firmware performance, improving system efficiency, and reducing maintenance overhead.

Cognizant
Bangalore, India

Programmer Analyst Trainee (Intern)
Aug 2021 - Dec 2021

- Led a Proof of Concept to integrate ShellCheck for automated shell script error detection, improving the development and code review process.

Education

University of Central Missouri | Masters - Computer Science | *Grade:* 3.12 / 4.00 (CGPA)
Aug 2023 - Present

Coursework: Operating Systems, Algorithms, Computer Networking & Security, Compiler Design & Construction, Design of Cryptographic Algorithms, Web Application Security, Cloud Security

Bharath Institute of Higher Education and Research | B.Tech - Computer Science Engineering | *Grade:* 8.4 / 10.0
2017 - 2021

Projects

Vshell | *Source Code:* https://github.com/v1n0dh/vshell

- Developed a custom shell in C++, inspired by a coding challenge at codingchallenges.fyi, closely resembling bash.
- Incorporated key shell features, including command execution, I/O redirection, and pipes, facilitating efficient user interaction with the system.
- Integrated tab completion for files and command history, enhancing user experience and productivity.
- Successfully used by developers in testing environments, showcasing effectiveness in managing command-line tasks.

System Metrics Display for dwm | *Source Code:* https://github.com/v1n0dh/dwmstatus

- Created a custom status bar for the dwm window manager using C programming, displaying real-time system metrics such as network speeds, Wi-Fi network, CPU temperature, RAM usage, battery status, and clock.
- Gathered system statistics directly from low-level file structures like */sys/class* for battery and network information, and used X11 library to dynamically update the status bar.
- Integrated network speed calculation by monitoring changes in RX/TX bytes and implemented Wi-Fi network detection using shell commands.
- Optimized system resource usage by writing efficient C code, providing critical information without relying on external tools or scripts.

Bencode Parser | *Source Code:* https://github.com/v1n0dh/Bencode-Parser

- Developed a parser that converts Bencoded data (used in torrent files) into a human-readable JSON format.
- Built in C++ utilizing the Jsoncpp library for JSON object handling, creating an API making it handy for handling Bencoded data, enhancing compatibility.

Certifications

- Rust Essential Training by Barron Stone, Linkedin Learnings | Certificate Link
- Linux Device Drivers by Kevin Dankwardt, Linkedin Learnings | Certificate Link