

Ahmed S. Darwish

+49 172 796 3572

darwish.07@gmail.com

darwish.chasingpointers.com

Summary

Operating Systems engineer, with good standing in the Linux open-source community

Experience

Harman Becker GmbH

Linux System Expert

Munich, Germany

September 2017 – Present

- Embedded Linux point of reference for company's BMW infotainment projects
- Received Harman's "Be Brilliant" award for fixing critical production-line bugs
- Keywords: Kernel, Yocto, Systemd

Valeo

Principal Software Engineer – Linux Kernel

Cairo, Egypt

July 2014 – August 2017

- Created a company-wide Embedded linux platform using Yocto and Bitbake layers
- Designed multiple security architectures for adopting Linux into the automotive domain
- Integrated secure-boot Linux solutions above Freescale's i.MX6 HABv4 chip
- Fixed upstream CAN networking drivers; **18 kernel patches** accepted
- Optimized U-Boot booting time, over Freescale i.MX6, from *2.5 seconds* to *50 milliseconds*
- Optimized total Linux system boot time, over i.MX6, from *30+ seconds* to *1 second*
- Provided project estimations; submitted team's yearly development-plans
- Mentored new embedded engineers; wrote official Linux development guidelines
- Valeo internal innovation challenge winner, 1st place: "Valeo in-Vehicle Cloud: A new automotive architecture" (*5 patents pending - EU office*)
- Keywords: Kernel, Security, Yocto, Systemd, ARM, R-Car H3, U-Boot, open-source

Vireton

Sr. Software Developer

Cairo, Egypt

May 2013 – July 2014

- Responsible for the company's backend service, written in Scala and R
- Directly involved in the service software lifecycle: Assessment, Research, Design, Implementation, Re-factoring, Benchmarking, Optimization, and Bug-fixing
- Developed the service mathematical and statistical forecasting engine
- Developed the service domain-specific language AST parser and evaluation package
- Developed the service storage layer using Orient Graph Database and Tinkerpop stack
- Developed the service integration testing framework. Debugged hard-to-trace bugs
- Provided small fixes upon all of the company's software stack, including Ruby on Rails, OpenERP Python, and system administration scripts
- Keywords: Software Engineering, Java, Scala, R, Apache Tinkerpop, NoSQL

Intel Labs

Wireless Application Engineer (Contractor)

Cairo, Egypt

December 2012 – May 2013

- For a team targeting 3GPP standard contribution, dissected Opnet's LTE simulation code to prototype new algorithms for small cell to macro cell handover
- Built and deployed lab's source-control servers. Implementation featured LDAP centralized authentication, fine-grained access control, project management tools, and automatic repo creation. All of provided services were tightly connected using custom-built Ruby, Apache mod_perl, and Bash scripts
- Keywords: Opnet, C, Ruby, LTE, mobile networks, libvirt, git, LDAP, Bash

Open-source Contributions

Security Research – Huawei routers 0-day

Jan - April 2017

- Exposed a zero-day vulnerability in the common Huawei Hg532n router
- Several ISPs, across the globe, ships this router *by default* to customers
- Authored a [Metasploit exploit module](#) for the vulnerability; got officially merged
- Huawei issued an [urgent press release](#) acknowledging & fixing the issue
- Keywords: Reverse engineering, IP networks, firewall, ARP, NAT, MIPS, Ruby

Contributing Member – Linux Audio Stack (PulseAudio)

2015 - present

- Authored [32 patches](#) officially accepted upstream
- Developed initial containers support: [memfds](#) as exclusive transport layer
- Developed runtime monitoring tools and generic bugfixes before each release
- Keywords: C, POSIX, containers, sand-boxing, latency

Contributing Member – Linux Kernel

2007 - 2015

- Authored [82 patches](#) officialy accepted upstream
- Helped in upstreaming the Smack Linux Security Module (LSM)
- Co-authored the [smackfs](#) virtual file system interface
- Added support for multiple Linux Security Modules ([security=](#) kernel boot parameter)
- Democratized Audit security-access subsystem for all security modules (out of SELinux)
- Integrated Smack security subsystem with the Audit subsystem
- Developed 'Saveoops' (pending state, [press review](#)): a shim which permanently saves the kernel log to disk upon fatal errors.
- Keywords: C, IPsec, mandatory access control, security auditing

Author – “Cute” Operating System

2010 - 2014

- Created a fully-preemptive 64-bit SMP kernel for the x86 PC architecture ([git tree](#))
- The developed components include a 512-Kbyte bootloader, virtual memory, dynamic memory allocation, concurrency primitives, an x86-64 optimized libc, APIC and IO-APIC interrupt-controllers support, BIOS MP-tables and e820 ACPI support, full SMP boot, a fair per-CPU scheduler, python scripts for plotting performance statistics, and several stress-tests for each kernel module
- Recently added persistence through a read/write Ext2 file system implementation

Technical Skills

Software Development

- Bare-metal and bootloaders development (U-boot, x86-64, ARM)
- Operating systems kernel and drivers development (Linux kernel, custom)
- User-space applications development (POSIX-compliant operating systems)
- Event-loop based applications design and implementation (POSIX, Javascript)
- *Multi-core* system-level and application-level programming (x86-64, ARMv7-A)
- Backend web services development in Java, Scala, and R
- Rapid prototyping and demos development in Python and Ruby

Software Design

- Software paradigms: low-level design, Object-oriented design, functional programming
- Embedded products overall network, security, and per-component architecture design
- Embedded Linux products build systems design using Yocto and Bitbake
- Source-code management using GNU Makefiles and CMake, Git and Mercurial
- Relational database design **algebra** and SQL querying

Software Analysis & Optimization

- Full-stack boot-time optimization for Embedded Linux systems (< 1 second)
- Benchmarking products performance; plotting results using Gnuplot and R
- Code spelunking: inspecting, tracing, and bug-fixing *huge* code bases
- Static and dynamic code analysis using Sparse, Valgrind, and EclEmma
- Hot code paths optimization using inline GCC assembly
- Linux Kernel debugging using kexec/kdump, crash analysis, and debugfs
- Ext2-based file systems data corruption analysis and debugging

System Administration

- Linux system administration using Systemd, Python, Ruby, and Bash scripts
- Practical experience in modern Linux components: systemd internals, logind, nspawn containers, memfds, d-bus communications, journal, PulseAudio, rtkit, polkitd
- TCP/IP, CAN, and Bluetooth networking and administration
- Building stateful IP firewalls using netfilter iptables
- Writing professional documentation using \LaTeX and Docbook

Education

Cairo University

2010

- Faculty of Computers and Information – B.Sc. Computer Science
- Graduation grade: Very Good. Graduation Project grade: Excellent