Marc R. Bevand

San Diego, CA, USA
Email: m.bevand@gmail.com
Web: www.zorinaq.com

Summary

I am passionate about: computer and network security, reverse engineering, security vulnerability research and exploitation, GPGPU, assembly optimization, decentralized cryptocurrencies (Bitcoin,) entrepreneurship, angel investing, etc.

Experience

Angel Investor
San Diego, CA, USA; Jul 2016–present

I fund seed stage companies, sometimes series A. Sector: primarily the tech industry. Investments to date: 3.

Information Security Engineer — Google
Mountain View, CA, USA; Jan 2014–Jun 2015

Helped ensure that Google's software and infrastructure is designed and implemented to the highest security standards. Performed security audits, risk analysis, application-level vulnerability testing, and security code reviews on external and internal systems and products. Member of the second-level responders behind security@google.com, investigated and triaged hundreds of reports from external security researchers. Conducted approximately 50 on-site and phone interviews for various engineering positions. Sample of my accomplishments:

- *Thread*: I performed a review of the Thread specification, a low-power wireless mesh networking protocol, designed by a consortium of companies of which Google/Nest is a member. My report identified 15+ security issues including practical attacks related to key/IV reuse, MACs, password hashes, etc. My discoveries were critical to make early in the design phase, given that Thread may become as ubiquitous as WiFi.
- *Confidential project A*: I was the main security reviewer (along with a coworker) on this project which involved 100+ engineers. I discovered 26 security vulnerabilities and advised the teams how to fix them. My findings spanned areas such as cryptographic errors, kernel, device driver, and firmware bugs, communication protocol flaws, etc. My review helped make the product significantly more secure before its release, and as a result my coworker and I received personal congratulations from this project's Vice President.
- *Confidential project B*: I reviewed a third-party application that would have been deployed in-house, and uncovered many security issues. I subsequently made a presentation of its technical details, and it became popular internally and was presented to hundreds of security engineers at a company summit.

Cofounder & CEO — Thin Air Ventures
Lake Forest, CA, USA; Apr 2013–Dec 2013

Cofounded with 2 partners. We were the second company in the world (after Canaan) to ship Bitcoin ASIC miners to the public, in October 2013. We purchased 10k A3256 chips from Canaan, and manufactured improved Avalon 1

clones. We designed custom chassis and heatsinks, sourced electronic components, contracted 2 PCB manufacturers as well as 2 electronic equipment assemblers to spread risk. Thanks to careful preparation it took us only 10 days to go from taking delivery of the 10k chips to having the entire batch of machines manufactured, tested, and ready for sale. By the time our total revenues grew to 400k USD and we started making profits, as the majority investor, I unfortunately had to shut the company down because US immigration denied me the necessary visa.

Senior Software Developer — Adconion Direct
Aliso Viejo, CA, USA; Jul 2012–Dec 2013

Performed software reverse engineering to disassemble and debug unknown code using IDA Pro, windbg, etc. Assessed scalability and performance issues in a high traffic, high transaction infrastructure (hosts with 100k+ IP aliases, DB tables with 500M+ rows, etc.) Within the first 2 months of hire: reverse engineered a 64-bit DLL and its data file format, a 5MB 32-bit Linux binary and its encrypted network protocol, improved the execution time of edge cases of the standard Linux ip(1) tool by 70x, and automated manual processes that took hours to run in seconds. Within the first 3.5 months my technical expertise was the direct source of 1+ million USD of additional revenues.

Principal Architect, Security — Rapid7
El Segundo, CA, USA; Jul 2009–Jul 2012

Developed key features of NeXpose, a vulnerability assessment tool sold primarily to large corporations and government agencies. Sole engineer responsible for maintaining Rapid7's PCI ASV status (Payment Card Industry - Approved Scanning Vendor.) Advised various internal groups on technical directions. Automated Nexpose hardware Linux appliance installation. Optimized product build time from 30-40min down to 10-15min. Enhanced Nexpose's custom SMB/CIFS stack to support the latest authentication protocols, developped «pass-the-hash» in Nexpose, tied it to Metasploit.

Senior Software Engineer — Rapid7
El Segundo, CA, USA; Jul 2007–Jul 2009

Lead the team developing security checks for Nexpose. Researched distributed SCMs, switched our 2GB repository with 100k files and 20k changesets from CVS to Mercurial, trained developers. Built a virtual machine lab from scratch for 2000+ VMs, 200+ templates, spread across 4 servers which I spec'd (total 300GB RAM, 10TB disk,) all running QEMU or KVM, with a custom web interface to manage and access VMs, which I wrote and open-sourced (Qemudo.)

Software Engineer — Rapid7
Torrance, CA, USA; Sep 2005–Jul 2007

Research and exploitation of security vulnerabilities. Reverse engineering of Microsoft security patches and proprietary network protocols. Developped security checks for Nexpose. Took the initiative of porting Nexpose to 64-bit Linux (30k lines of C++ functions called from Java to mostly parse and craft network packets,) which turned out to be a great foresight as a year later Rapid7 urgently needed to migrate appliance customers away from 32-bit.

System & IT Security Engineer — SmartJog USA Inc.
Los Angeles, CA, USA; Dec 2004–Sep 2005

Designer and developer of RBC (reliable bit cast: multicast encrypted file transfer protocol for satellite and internet links, with strong error-recovery capabilities.) This was the core software of their platform for secure digital delivery of media content for the movie industry. PKI deployment (multiple CAs, 500+ certs.) Automation of 2-factor authentication USB token deployment to our customers. System and network administration.

System & IT Security Engineer — SmartJog S.A.
Paris, France; Sep 2003–Dec 2004

Linux development to secure the SmartJog platform.

Software Development Intern — SNAISO
Issy-les-Moulineaux, France; Mar 2003–Sep 2003

Wrote software modules to secure one of their IDS products (authentication, encryption, access permissions, etc.)

Teacher Assistant — ÉPITA
Le Kremlin Bicêtre, France; 2002

Instructed students in C language development on Unix systems (NetBSD, Solaris, Digital Unix.) Graded their projects.

Software Development Intern — MiniSAT
Paris, France; 2001

Designed a back-office tool to manage one of their interactive TV software application.

Website Testing Intern — WStore
Issy-les-Moulineaux, France; 1999

Audit of their web site, a computer hardware store.

Education

Master's Degree, Information Technology; ÉPITA (*École Pour l'Informatique et les Techniques Avancées*,) SRS Department (*Systèmes, Réseaux et Sécurité*: System, Network and Security); Paris, France (1998–2003.)

Skills

Cryptocurrencies: author of hdminer (fastest Bitcoin GPU miner back in 2010–2011,) author of SILENTARMY (fastest Equihash GPU miner in 2016, won a 10k USD bounty for first place in the Zcash Open Source Miner Challenge.)

Cryptography: discovered cryptographic vulnerabilities in Merkle Tree Proof (won a 9k USD bounty), Thread protocol.

Assembly optimization: author of the world's fastest RC4 symmetric cipher implementation for AMD64 (+50% speedup compared to the then-current OpenSSL version,) author of OpenSSL's MD5 message-digest optimized for AMD64 (+65% speedup.)

Kernel programming: addition of a syscall to FreeBSD allowing (a)synchronous process execution, integration of an "active" link type into the OpenBSD FFS filesystem, development of a new filesystem for NetBSD.

Network programming: OS fingerprinting tool, ARP spoofing and TCP hijacking, miscellaneous clients and servers (SMTP, FTP, NNTP, IRC, etc.)

System programming: implementation of a shell, creation of an assembler and its associated virtual machine.

Unix system administration: IP networking, firewalling, NAT, DNS, DHCP, TFTP, NFS, HTTP, FTP, SSH, Samba, LFS (Linux From Scratch) systems, excellent ability to troubleshoot and debug various problems.

Programming languages: C, Java, Perl, Python, AMD64/x86 assembly, OpenCL, ATI CAL IL assembly, Bourne shell scripting, Smattering: C++, Ruby, Sparc assembly, SQL, Pascal, OCaml.

Development on various platforms: Linux, Android, OpenBSD, FreeBSD, NetBSD, Sun Solaris, Digital Unix (OSF/1,) Windows, DOS.

(Updated: November 11, 2018)