

# Vadim Vygonets

## Software Engineer

### Contact info

*Phone:* +49 176 6040 3230  
*Address:* Holteistraße 13, 10245 Berlin  
*E-mail:* vadik@vygo.net

### Experience

#### *Areas of Expertise:*

Programming: embedded (bare metal, FreeRTOS), bootloaders, drivers, concurrency, systems, kernel (Linux), networking. Parsing: context-free (*yacc*), regular. Security. Scripting, integration, builds (Makefiles, *ldscripts*). Strong Unix knowledge, familiarity with hardware.

#### *Programming Languages:*

C (*lex*, *yacc*), Go, Assembly (ARM, MIPS, 8086), ANS Forth, Perl, awk, Bourne shell, SQL.

#### *Skills:*

Writing clean code. Solving hard bugs. Working with large codebases. Having both high-level and detailed perspectives on systems. Quickly learning new technologies and concepts.

### Employment

#### *Since 2019:*

##### ***Freelancer: Firmware Developer***

Developing embedded software for clients.

#### *2022 – 2025:*

##### ***Confidential project***

#### *2021 – 2022:*

##### ***weeve GmbH: Embedded Security Engineer***

Developing a suite for automatic rewriting of ARM Cortex-M ELF binaries to prevent stack overflow and return address smashing, using TrustZone-M and shadow stacks. Implemented protection for exception handlers.

#### *2020 – 2021:*

##### ***MBition GmbH: Expert Software Engineer***

Developing software.

#### *2019:*

##### ***Concept Reply GmbH: Firmware Developer***

Developed ESP-IDF-based firmware for an ESP32 device for a new Grohe product, as a part of the firmware development team; partially led the development. Wrote an AWS IAM authentication client, a UART message dispatcher, management of other system components over UART, MQTT communication with cloud over TCP/IP, error and log propagation. Rewrote UART message read function.

#### *2019:*

##### ***An Internet Consultancy: Firmware Developer***

Developed Amazon FreeRTOS-based firmware for an ESP32 device for the company's client as a part of the firmware development team. Worked in close cooperation with back-end, mobile and security teams. Wrote a BLE provisioning subsystem, drivers for an SPI accelerometer, upgraded FreeRTOS, improved make-based build system. Developed secure boot and FLASH encryption during factory initialisation, including secure encryption key distribution. Designed and implemented an encrypted device configuration protocol. Participated in developing Over-The-Air upgrades, I<sup>2</sup>C pressure sensor drivers, security analysis. Fixed bugs, reduced memory use.

2017 – 2019:

**MatchX:** *Lead Software Developer*

Developing firmware for LoRa LPWAN IoT devices; leading a small team. Wrote firmware for LoRa nodes with BLE upgrade capabilities for Dialog ARM Cortex-M microcontroller from scratch in C, using Dialog SDK, FreeRTOS and IBM LMIC library. Maintained OpenWRT-based firmware for LoRa gateways. Built factory QA/flashing stations based on Raspberry Pi and MatchX LoRa nodes, with software in C and Go, for testing LoRa node and gateway hardware, flashing firmware and distributing cryptographic keys synchronised to the backend. Lead the development of LoRa back-end software. Worked in close cooperation with the hardware developer.

2016 – 2017:

**Airfy:** *Senior Software Developer*

Developing OpenWRT-based firmware for consumer WiFi routers. Ported OpenWRT and LEDE to the new platform. Maintained firmware components written in C, awk and Bourne shell. Solved bugs and developed new features in back-end components. Wrote new back-end daemons in Go.

2014 – 2016:

**Mobisol:** *Embedded Software Engineer*

Developing firmware for solar rural electrification systems as part of the firmware team, in close cooperation with hardware, back-end and mobile developers. As the principal maintainer of AVR platform, developed new FLASH and GSM modem drivers, improved concurrency of communication subsystem, developed new features, optimized firmware size and memory usage, defined and implemented peripheral communication protocols, improved build automation. Developed an upgrading bootloader and a robust backwards and forwards compatible key-value store optimized for FLASH for a new ARM-based platform. Lead the design of cryptographically secure payment tokens for offline systems.

2012 – 2014:

**Recurity Labs:** *Security Consultant*

Security analysis of clients' software and hardware systems; programming.

2010 – 2011:

**Soundcloud:** *Architecture Developer*

Automatic server installation and upgrades using Chef and Ruby.

2008 – 2010:

**Jungo:** *Software Engineer, Residential Gateway Professional Services group*

Part of the group developing portable residential gateway software. Worked across the product codebase, including proprietary userland and kernel code, third party drivers, Linux kernel, bootloader (Das U-Boot), uClibc, Asterisk, Samba. Fixed kernel crashes and bugs in third-party wireless and DOCSIS network interface drivers. Ported U-Boot and Linux kernel to a new ARM-based platform. Debugged network related issues in the field.

2007 – 2008:

**Radware:** *Development, integration*

Unix systems programming, scripting, building, integration, delivery, packaging. Part of the Insite 5 management business unit. Developed the CLI (C on Unix). Implemented job control, piping and shell built-ins, improved terminal handling, rewrote most commands. Integrated between Linux, Oracle DB and Java-based product on the appliance. Designed and implemented a build system using make, Ant and Maven. Implemented upgrade procedure.

2006:

**Formula Telecom Solutions:** *Configuration management, packaging, scripting*

Building, packaging and deploying the software product. Setting up and maintaining build and test environments. Worked with Solaris, HP/UX; Makefiles, Perl, awk, C shell.

2004 – 2006:

*iMesh Ltd.: Unix server administration, scripting*

Installing, upgrading and maintaining the company's servers. Worked with Linux (Debian, RedHat/Fedora/CentOS), load ballancing and high availability (LinuxHA, Ultra Monkey), TUX, Apache, proprietary daemontools-based services.

2004 – 2005:

*T.N. Traffic Ltd.: Unix system and network administration, scripting, security*

Installing and maintaining the company's servers. Re-designing and upgrading office infrastructure. Worked with Debian Linux, OpenBSD, Apache, Postfix, Cisco Catalyst switches, Cisco PIX, OpenBSD pf packet filter, Linux iptables, CARP.

2001 – 2003:

*School of Engineering, Hebrew University of Jerusalem: Unix System administration, scripting, programming, user support*

2000 – 2001:

*Freelancer: Network design, Unix system administration, programming*

Clients included Intel Network Group of Jerusalem, Ness Technologies, Airport Authority of Israel, XChain Ltd., GoldFire Ltd., Saiki-Tech Co. Ltd., Volcani Institute.

1997 – 1999:

*Institute of Computer Science, Hebrew University of Jerusalem: System administration, programming*

## Contributions

- *Go qr package by Russ Cox*: Added industry standard features: Kanji and ECI modes, Structured Append, segmentation (shorter encoding), automatic mask selection; as well as Kanji mode encoding with UTF-8 input, Micro QR code generation. Improved PNG encoding, achieving **30–65× faster and up to 28% more efficient compression** compared to the standard encoder.
- *OpenBSD's cwm window manager*: contributed features and bugfixes.
- *Heatshrink embedded compression library*: simplified encoder and decoder state machines, redefined stream format in backwards compatible manner, improved compression ratio, fixed bugs.
- *ihex*: Wrote a feature-complete fully documented Intel HEX package for Go for use in production.
- *The Go programming language standard library*: added CRAM-MD5 authentication to the net/smtp package, added conversion between Latin-1 and UTF-8 to compress/gzip, minor contributions to other packages.
- *forego*: Wrote a 32-bit virtual machine in Go running a minimal self-hosted standard-compliant Forth system (compiler, assembler, parser and interpreter) supporting full CORE wordset as a learning project. During bootstrap phase, wrote a standalone assembler for the VM.
- *SSH 1.2.26, OpenSSH 1.2–2.3*: Ported to BSD/OS 3.1–4.1, adding BSD authentication.
- *Debian GNU/Linux 1.0–1.3*: Maintained software packages.
- Developed a suite allowing BSD/OS users to build custom installation floppies.
- Contributed patches to OpenBSD, BSD/OS, Exim and other software.
- Contributed to the OpenBSD ports project (porting software to OpenBSD 2.2–2.4).