Sam Revitch
5842 SE Taylor St.
Portland, OR 97215
sam.revitch@gmail.com

Objective

To work in the field of system software, on technically interesting kernel-mode or network serving projects. I enjoy working with file systems, device drivers, palmtop environments, and complex networked applications.

Skills

- Linux kernel environment; VFS, SCSI, videodev, and USB driver stacks.
- Windows NT kernel environment; IFS driver stack.
- Programming Languages: C, C++, Perl, x86/amd64/xscale assembly.
- Debugging and troubleshooting kernel and user mode software.
- Reverse engineering

Work Experience

April 2001 – Present: Sr. Software Engineer, Hewlett-Packard StorageWorks NAS (formerly PolyServe, Inc.)

I contributed to the design and implementation of the B-tree manipulation library, metadata journal, and software snapshots features of a new and yet unreleased Linux clustered filesystem product.

I worked on the PolyServe File System (PSFS) shared-storage file system driver. PSFS is designed to install and run on several servers attached to the same block level shared storage device, and supports concurrent, coherent access to a single disk file system across all servers.

PSFS involves fine grain cross-server exclusion. When one or more servers fail, it is able to restore the filesystem to a consistent state and continue service on remaining servers without interruption. It runs in kernel mode yet defers some responsibilities to user-level processes, and runs in both the Linux and Windows kernel environments.

My specific responsibilities included:

- Metadata journal design and implementation
- Block allocator design and implementation
- Code separation to support common, Windows-specific, and Linux-specific routines, following a difficult Windows port
- Cross-platform kernel thread and synchronization primitives library
- Linux and Windows user-data write path implementations
- Windows rename implementation
- Windows cross-server directory change notification design and kernel side implementation
- On-going product support and maintenance tasks

Summer 1997 – Software Engineering Intern, Sequent Computer Systems

I worked on the Dynix/PTX kernel, including a reimplementation of the sleep system call handler.

Summer 1996 – Software Engineering Intern, Sequent Computer Systems

My responsibilities included porting various open-source software packages to the Dynix/PTX operating system.

Education

BS, Computer Science, University of Washington, 2002.

I did two “senior projects” here. One was a research project in natural language processing with a graduate student. The other was a multi-player real time strategy game in a team of five, for which I implemented the networked game server component in C++.

Open Source Projects

I implemented a software Bluetooth hands-free stack for Linux, including control, audio handling, and a simple GUI. This project is released under an open-source license and can be obtained from http://nohands.sf.net/

I implemented a Linux driver for Ricoh USB built-in laptop webcams, the final version of which supported seven different devices. I was forced to discontinue this project, despite the fact that it was done in my spare time, due to a denied request for approval from the HP open-source review board.

I occasionally contribute hardware support patches to various projects, including ALSA and the Linux ACPI and IEEE1394 subsystems.

I own a Motorola EZX Linux phone and developed a partial SDK for the Motorola environment, based on reverse engineering parts of the Motorola environment and patching freely available Trolltech Qt sources.