Resume of Karl D Butz

650 274-7482 :: KarlDButz@EndlessSkye.com

PROFESSIONAL EXPERIENCE

GNOMORUM   SAN MATEO, CA
COMPUTER SECURITY CONSULTANT   6/2016-Present

Spun off security consulting business from Gnomorum in San Bruno, CA. Provide penetration testing, reverse engineering and security fixes for zero day bugs on demand to clients that can't be disclosed. Provide detailed documentation for a technical audience and custom software for ad hoc bug fixes.

Reconstructed C source code used to rebuild a legacy in house i686 Windows application using tools like IDA Pro Disassembler.

ENDLESS SKYE SOFTWARE   HALF MOON BAY, CA
PRINCIPAL SOFTWARE ENGINEERING CONSULTANT   1/1995 -2/2016

Gnomorum, San Bruno CA. 5/16-2/17. Primary role to fine tune, test and verify code correctness and robustness principally through automated testing. Proficient in such scripting languages as Python and Perl particularly for the development and execution of test automation on multiple, multi-platform small footprint embedded devices. Expert in writing highly optimized C code. Knowledge of x86 and ARM assembly. Significant experience with video codecs. Experience working on many aspects of image, video and display processing. Siginficant experience with Windows and Linux.

Genentech, South San Francisco CA. 12/15-2/16. Wrote Secure Socket Layer (SSL) layer for a wireless CE 5 device (the Motorola MK500 micro-kiosk) in C used to communicate securely with a remote gateway server over a cellular modem. Wrote prototype application in C/C++ that read an RFID badge, scanned a bar code, and then formatted the data in JSON format for secure transmission to the remote server. The host development environment ran in XP SP3 in a virtual machine. The target software ran under CE5.

0xc0d3, Los Altos CA. 4/15-10/15. Worked on specialized stealthy WiFi drivers for Windows and Linux including proprietary protocol drivers in kernel mode. Developed code framework in C and Python for use in end user design. Wrote Windows Native Mode control application in C to control and modify parameters of stealthy WiFi subsystem before Win32 was loaded during system boot.

GE Oil and Gas, Lewistown PA, 12/14-3/15. Worked on WiFi driver for proprietary Windows embedded device used in real time image analysis. Analyzed and solved issues of WiFi (802.11b) and SD Bus transport drivers, and investigated potential hardware problems. C/C++, Windows network drivers, Windows network stack and x86 assembler. Developed socket-based programs for use in analysis.

Ring Zero Tools, Sunnyvale CA, 2/14 -8/14 Developed low level penetration testing and diagnostic tools using my expert knowledge of Windows internals and Intel x86 architecture. Also worked on de-compilation tools from binary to C++/MFC source and Wine-like sandbox environment for forensic investigators. C/C++, Python, and x86 assembler. Used Ida Pro Disasembler, Bochs and in-house developed tools.

Microsoft, Mountain View CA, 11/12-9/13. Xbox One. Activities included GPU, low-level graphics and video subsystem support. Used C/C++ under Visual Studio 2010 and assembler language specific to the SOC. Developed highly multi-threaded programs to analyze performance in near real-time using iPerf and similar utilities. Wrote Perl scripts to analyze results and also for use by black box testers.

For Intel in Folsom, CA I worked on graphics drivers using the Sandy Bridge chipset on Windows 7 and Windows 8. Special emphasis was on optimum performance of DirectX Video Acceleration (DXVA) and DirectX/Direct3D 11. Development environment included C/C++ using Visual Studio 2008 and Windbg.

At Sony Ericsson in Redwood City, CA I worked on the optimization of key device drivers and system level components for an ARM based smart phone. This effort required an extensive knowledge of Windows Mobile 6.5 operating system internals and a highly detailed knowledge of the processor and device hardware. In addition to such standard tools as Kernel Tracker, ad hoc analysis tools were written to provide detailed performance profiles. Principal areas of investigation were in the graphics driver and the NAND flash file system and driver. In addition to the standard tools available in the Windows Mobile 6.5 AKU, I used the IDA Pro Disassembler to explore system areas where source was not available.

For a Neptune Technology Group in Tallassee AL I resolved board bring up issues on a Windows CE 6 based industrial single board computer. Issues included providing KITL support and device driver compatibility under CE 6.

At Broadcom in Sunnyvale, CA I developed a BSP and brought up mobile phones for Windows Mobile 6. As a key part of this development, I ported NAND flash code for the flash file system from a real time mobile phone kernel to Windows Mobile 6. This hardware was based on a prototype ARM 9/ARM 11 SOC mobile phone chip set. I converted big to little-endian code, especially in the bootloader and pre-bootloader authentication components. In addition, I wrote the real time clock, timer interrupt, I2C and other fundamental components in the OEM code. I wrote serial KITL and assisted in the development of USB KITL components. The development environment included JTAG, Platform Builder, ARM assembler and C/C++. Also, I used the IDA Pro Disassembler and serial output to identify and resolve development issues. JTAG was used to burn the crypto authentication, bootloader and OS images to flash memory.

At Microsoft in Mountain View, CA, I wrote bootloader components for IPTV STBs running CE5 on SH4-based SOCs. Developed and integrated bootloader components from partners including graphics, NAND flash and Ethernet. Used JTAG to burn the bootloader to flash memory. Used IDA Pro Disassembler and serial output to help solve development issues. Wrote IP fragment re-assembly code used by the bootloader's IP stack. This component is cross-platform and used across the entire family of IPTV STBs. Principal development is C/C++ with some SH4 assembler. Platform Builder 5 and Visual Studio 2005 constituted the principal development environment.

While at Wyse Technology in San Jose, CA, I enhanced and optimized CE 5.0 and Linux drivers and kernel on multiple hardware platforms for a leading thin client manufacturer. Optimized display and USB storage drivers running on thin clients connected to RDP and ICA. Wrote Direct Show filter driver for CE 5.0 in support of high performance ICA remote session. Optimized CE NDIS driver. Diagnosed and fixed such low level system bugs as DMA and IRQ conflicts during system boot and PCI enumeration without hardware support. Added functionality to CE kernel to enhance thin client capabilities. Principal development and debugging environment consisted of Platform Builder 5.0 and Embedded Visual C++ 4.0. Additional debugging efforts involved RedHat Linux and gdb over telnet. Strong C background with some highly optimized x86/MMX assembly language in time critical and heavily used routines.

During my tenure at Hand Held Products (formerly HHP) in Charlotte, NC, I optimized Pocket PC display drivers for wireless embedded devices used in large enterprise installations. Used C++/C and ARM assembler to achieve substantial performance gains in heavily used display driver routines relying on the Graphics Primitive Engine. Tuned and validated performance enhancements using custom developed tools.

In a private partnership I ported an embedded wireless (802.11g) Linux appliance (based on 2.5.21 Linux kernel) to a Windows CE .NET wireless appliance based on the ARM V4I. Development is mainly in C with some ARM assembler for low-level OS and time-critical routines. Used Embedded Visual C 4.0 and Platform   Builder   4.2.

At Intermec in Everett, WA, I helped the bring-up of an embedded wireless appliance running under Windows CE .NET and the ARM V4I microprocessor. Wrote and modified ISRs and OEM kernel code for the keyboard, touch screen, and display using C and some ARM assembler. Wrote diagnostics and test utilities for embedded wireless CE .NET devices connected to corporate intranet Virtual Private Networks. Used Platform Builder 4.2 and Embedded Visual C 4.0 to develop and debug over Ethernet and serial links.

Ported Windows NDIS drivers to Pocket PC drivers when at Symbol Technologies in San Jose CA for their hand-held bar scanners.

For Secret Seal in Palo Alto CA I wrote Win 32 API hook software using remote thread injection to target processes of interest for security software running under Windows 2000. In addition I wrote anti-debugging techniques and other tricks like dynamic code execution on the stack to thwart debuggers and disassemblers. C and x86 assembler.

At VPNet in Milpitas CA I worked on NDIS drivers used in their virtual private network software.

For Uppercase in Santa Clara CA I wrote a proprietary operating system extension under Windows 98 and NT4.0 that facilitated document capture for a Windows CE hand-held device based on an embedded SH3-based CE appliance.

When at Logitech in Fremont CA I wrote early stage keyboard device drivers and mice device drivers.

Wrote low level utilities mostly in assembler but some C used in the control and analysis of in-house hardware while at Tandem Computer in Cupertino CA.

While at EnterPlay/ShareWave in Rancho Cordova CA/El Dorado Hills, CA I worked on remote wireless multi-monitor support for the PC in the home entertainment market. Designed, developed and wrote specialized and proprietary Windows 95 graphics and Direct Draw drivers, DLLs, VXDs and thunks that redirected Windows and selected applications onto a second and independent graphics display. Required highly extensive knowledge of kernel, GDI, VMM and Direct Draw internals.

At CompCore Multimedia in Santa Clara CA I optimized MPEG driver using Intel's Vtune in conjunction with hand-optimized assembler x86 included the then new MMX instruction set.

Designed and developed a turnkey image processing system for a private company in Los Altos Hills, CA. It was used in an industrial distributed environment running on multiple embedded x86-Windows CE appliances. Developed an image processing system for use in the real-time monitoring of image quality on large industrial printing presses.

Developed network-independent API for network profiling and diagnostic tools. Wrote tools in C++ and network support for NetBIOS, IPX and TCP/IP as Windows 3.1 DLLs in C and 80x86 assembler.

When at Compaq Computer in San Bruno CA I wrote a Windows 95 File System Driver (FSD.) Designed and wrote several Windows 95 VxDs and DLLs as part of a set of utilities used in the manufacturing, setup and diagnosis of consumer computers.

QUATERNIONICS   HALF MOON BAY, CA
SOFTWARE ENGINEERING CONSULTANT   10/1991 - 1/1995

While at Weitek in Sunnyvale CA I designed the Weitek 5X86 graphics controller device driver development kit. Wrote sample programs in C and 80x86 assembler. Wrote user and API manuals for inclusion with the commercially available graphics board.

Wrote the MIL (Machine Interface Layer) documentation for use by developers on the PenPoint operating system of GO Corporation in Foster City CA. Interviewed developers, reviewed in-house documents and source code in C and x86 assembler to create this documentation.

SOFTWARE PUBLISHING CORPORATION   MOUNTAIN VIEW, CA
SENIOR SOFTWARE ENGINEER   9/1989 - 9/1991

Designed and implemented the HARVARD GRAPHICS 3.0 device driver development kit for ISVs. Designed, specified and wrote a driver development environment using C and 80x86 assembler under PC/MS-DOS. Wrote a comprehensive user manual and prepared sample device drivers.

MEDIA CYBERNETICS, INC.   SILVER SPRING, MD
GRAPHICS TOOLS MANAGER & PRINCIPAL SOFTWARE ENGINEER   11/1986 - 9/1989

Principal software engineer and manager of the Halo Graphics Tools product line. Responsible for design and architecture of new versions of Halo. Architected and led the migration of the HALO graphics kernel to the Texas Instruments TMS34010 graphics microprocessor in 34010 assembler. Wrote highly optimized 2D graphics routines in 80x86 assembler for the HALO graphics kernel and display, scanner and printer drivers.

Performed project management and scheduling. Managed a staff of six software engineers.

INTERACTIVE MICROWARE, INC.   STATE COLLEGE, PA
APPLICATIONS PROGRAMMER/ANALYST   12/1984 - 11/1986

Designed and developed data acquisition and display software for laboratory instruments. Managed the SCIENTIFIC PLOTTER-PC+™ product through the software development life cycle. Developed powerful yet small and fast code in C, 8086 and 6502 assembler.

EDUCATION

B.S., Environmental Engineering. Pennsylvania State University. March, 1981.