
**2000 University Ave**
**Dubuque, IA 52001**
**(563)589-3233**
**http://www.dbq.edu**

# Preventing “Oh Shit” Moments for €20 or Less

*By Philip A. Polstra, Sr.*

**March 2012**

## Contents



## Introduction

Since its introduction USB has quickly taken over the PC peripheral market as the standard interconnect standard. USB flash drives have replaced CD-ROM/DVD-ROMs and floppy drives as a means of exchanging data and providing booting alternatives.

While everyone uses USB devices, few understand how they work. A family of live Linux distributions has become popular among security practitioners. Forensic investigators are extremely likely to encounter evidentiary flash drives during the course of their work. In this paper, an inexpensive device for blocking USB write operations is discussed.

## Problem Statement

Plugging a USB flash drive into a computer can result in its alteration. In the case of forensic investigation this can destroy evidence. For information security professionals, having tools stored on a flash drive deleted by anti-virus programs is bothersome. As a result, a practical method of blocking write operations to USB flash drives is desirable.

## Previous Options

Some older flash drives have write protect switches, but such a feature is rare today. Commercial write blockers exist, but are too expensive for everyday use.

## Proposed Solution

An open source USB flash drive write blocker based on the FTDI VNC2 microcontroller is presented.

## Benefits of Proposed Solution

### *Low Cost*

The devices presented here can be constructed for €20 or less. Commercial write blocker devices currently available cost several hundred euros.

### *Ease of Construction*

Two different packages are presented. One package is completed implemented in software. The other package requires the builder to solder four wires from a USB cable to the circuit board. Alternatively, a dual port FTDI development board could be used with a USB A-to-A gender changer cable. Devices are easily programmed using a USB-based debug board.

### *Open Source Solution*

Full source code is freely available. Redistribution and modification is permitted in accordance with the GNU GPLv3.

## Implementation

### *How USB Flash Drives Work*

In contrast to most USB devices which process commands using the control endpoint, flash drives receive commands and transfer data exclusively on bulk endpoints. Each transaction involves 2 or 3 phases: Command Block Wrapper (CBW), Data (optional), and Command Status Wrapper (CSW). Because they use bulk endpoints exclusively, USB flash (and hard disk) drives are sometimes called Bulk Only Mass Storage (BOMS) or BBB devices.

### *How Does the Write Blocker Work*

Our device impersonates a BOMS device. We intercept the CBW and look at the command (first byte in the command block itself). If the command appears on a white list of safe commands that won't alter our drive it is forwarded to the drive and data (if applicable) and CSW phases are bridged between the write blocker and the drive. Depending on the command, we either fake a successful transaction or report failure. The reason for faking some transactions is that some operating systems, namely Windows, handle certain failures poorly.

### *What Platform Will Be Used for the Write Blocker*

The FTDI Vinculum II (VNC2) microcontroller will be used for this project. FTDI is well known for their USB chips (such as the FT232 chip found in some Arduino boards). The VNC2 chip supports two USB ports which may be configured to operate as a host or slave. FTDI offers several different development modules to VNC2 developers. One

package, the Vinco, has an Arduino form factor and has one slave and one host USB port. The Vinco allows the write blocker to be implemented completely in software. A more compact device can be constructed using the V2DIP1-32 module. This does require soldering four wires to connect a USB cable to the V2DIP1-32.

### *How is the Device Constructed*

The Vinco solution is completely implemented in software. If the V2DIP1-32 is used, do the following. First, cut the USB B connector off of a USB cable and strip the four wires. Second, unsolder the header pins from the V2DIP1-32. Finally, solder the four wires as follows: Red-V50, Black-GND, Green-U1P, and White-U1M.

### *Which Commands are Permitted*

The following are passed on to the drive: INQUIRY=0x12, MODE_SELECT6=0x15, MODE_SELECT10=0x55, MODE_SENSE6=0x1A, MODE_SENSE10=0x5A, READ6=0x08, READ10=0x28, READ12=0xA8, READ_CAPACITY10=0x25, READ_FORMAT_CAPACITIES=0x23, REPORT_LUNS=0xA0, REQUEST_SENSE=0x03, SEND_DIAGNOSTIC=0x1D, START_STOP_UNIT=0x1B, WRITE6=0x0A, WRITE10=0x2A, WRITE12=0xAA.

### *Getting the Source Code*

The source code is available on the conference DVD. Alternatively, the author will happily provide you with a copy if you request it by e-mail.

### *Known Issues and Limitations*

The VNC2 operates at USB 2.0 full speed (12 Mbps). High speed USB (480 Mbps) is not supported. In addition to the lower bandwidth, full speed endpoints utilize smaller packets (64 bytes) than high speed endpoints (512 bytes) so the transmission overhead is higher as well. Using this write blocker is noticeably slower than directly connecting a flash drive to a computer, but the device is still quite usable. Because flash drives are typically slower than hard disk drives, this device is not recommended for use when performing a full backup of an external hard drive. The author is currently working on a family of devices to facilitate forensic duplication of larger devices at high speed.

The device should work with flash drives with block sizes larger than 512 bytes. This functionality is untested, however, as the author could not locate flash drives with larger size blocks.

The device has been tested using Windows XP and OpenSuse 12.1 and seems to work. It is possible that other operating systems could cause the device to crash. It is unlikely that the drive will be altered even if the device were to crash.

Occasionally, if the drive is not connected shortly after it is detected, the drive will enter suspend mode and the device will hang. Removing and reinserting the device should allow the drive to be accessed. Mount the drive in a timely manner the second time!

If you are doing a forensic investigation as opposed to trying to protect your tools from a Windows box, the use of Linux is recommended. This is due to limitations of Windows and how it handles multiple LUNs and non-FAT filesystems.

## Summary

The device described here can be constructed in little time at a low cost. A USB write blocker has applications in both forensics and information security. The same concepts and code used in this device could likely be applied to more expensive high performance devices as well.

## About the Author

Phil cleaned out his savings at age 8 in order to buy a TI99-4A computer for the sum of $450. Two years later he learned 6502 assembly and has been hacking computers and electronics ever since.

Phil currently works as a professor at the University of Dubuque in Dubuque, Iowas. He teaches computer security and forensics. His current research focus involves use of microcontrollers and small embedded computers for forensics and pentesting. Prior to entering academia, Phil held several high level positions at well-known US companies. He holds a couple of the usual certs one might expect for someone in his position.

Phil is also an accomplished aviator with several thousand hours of flight time. He holds 12 ratings including instructor, commercial pilot, mechanic, inspector, and avionics tech. When not working, he likes to spend time with his family, fly, hack electronics, and has been known to build airplanes.

Phil has a Baccalaureate degree in Physics/Math from Calvin College (the number one physics undergraduate institution in its class during Phil's tenure there), a Master's degree in low-temperature condensed matter physics from Purdue University (ABD PhD), and should complete a PhD in business administration with a concentration in computer and information security from Northcentral University this spring.

In addition to teaching his normal classes, Phil serves as an advisor to the University of Dubuque Computer and Technology Club. Under his leadership, the club has attended several information security conferences around the USA, constructed a 3-d printer (RepRap Mendel), built a number of computers, been introduced to the wonderful world of Linux and open source software, and has helped numerous people in the community through PC tuneup events. Phil has also developed some new and exciting classes at University of Dubuque including cyber-forensics, microcontrollers, and ethical hacking. Phil has also been instrumental in developing online training offerings at the university.

Phil spoke at a number of conferences in 2011 including NetSecure '11, The Tri-state Computing Symposium (TriCS), Makerfaire Detroit, and the inaugural 44Con. During the spring of 2012 Phil is scheduled to speak at Black Hat Europe 2012, B-sides Iowa, the Midwest Instructional Computer Symposium (MICS), and ForenSecure '12. He anticipates further engagements during the summer and fall.

Phil may be contacted via e-mail at ppolstra@gmail.com or ppolstra@dbq.edu. You can also follow him on Twitter at @ppolstra. His blog lives at http://ppolstra.blogspot.com.

## About University of Dubuque

The University of Dubuque is a small, private university affiliated with the Presbyterian Church (U.S.A.) offering undergraduate, graduate, and theological seminary programs. The University is comprised of individuals from the region, the nation, and the world.

As a community, the University practices its Christian faith by educating students and pursuing excellence in scholarship. Therefore, the University of Dubuque is committed to:

- The Presbyterian tradition;
- Excellence in academic inquiry and professional preparation;
- Relationships which encourage intellectual, spiritual, and moral development;
- Community where diversity is appreciated and Christian love is practiced;
- Stewardship of all God's human and natural resources;
- Zeal for life-long learning and service.

## References


1. USB Complete: The Developers Guide (4th ed.) by Jan Axelson provides an excellent overview on all things USB.
2. USB Mass Storage: Designing and Programming Devices and Embedded Hosts by Jan Axelson has a great summary of the specifics of how BOMS devices work.
3. Jan Axelson's website at http://lvr.com has lots of good up to date information on USB and also files for her books.
4. The official USB documentation can be found at http://www.usb.org. This can be some very dry reading, but it is also authoritative.
5. See http://www.ftdichip.com for more on the VNC2 microcontroller.
6. See http://seagate.com for SCSI references unless you want to purchase documents from http:://t10.org.
7. Embedded USB Design by Example by John Hyde is a good book on doing USB stuff with the VNC2 family of microcontrollers. This book is available for download from http://ftdichip.com. In this book, Mr. Hyde walks through a couple of examples including a USB key logger.
8. Phil's 44Con USB Flash Drive Forensics Video http://www.youtube.com/watch?v=CIVGzG0W-DM provides a quick overview on USB and also discusses forensics of USB and how to build some cheap

forensic duplicators based on the VNC2. Slides from this presentation are available at http://www.slideshare.net/ppolstra/44con.