

## ***USB Heaven & Points South***

Clint Tinsley, IPCUG/April 2008

After watching a presentation on the building of bootable USB drive using Ubuntu 7.10, a number of questions were raised in my mind. After that presentation, I went on a quest of sorts in pursuit of things you can do with a USB drive.

Before I get into the details, first let me describe the “deliverables” that you will get from reading this article:

1. USB “thumb” drives are great, very inexpensive, but do come with some risks.
2. Portable Applications that you can run from a USB drive that can be very handy in that you can run them on any Windows computer without installing them.
3. Two Bootable USB implementations for Windows Vista and Windows XP. Incidentally, Vista has some very nice tools for working with USB drives and creating bootable USB drives.

I have been using USB drives for about 2 years now and I have seen the good, the bad, and ugly when it comes to these USB devices which tout a number of names such as flash drive (no they don't flash, and they aren't fast “as in a flash”), thumb drive (only if you have hit your thumb with a hammer is there a resemblance), and stick, as in stick it in your computer. Other monikers include JumpDrive (Lexar, never seen one jump and beans were not included), Firefly (doesn't light up if taken outside at night, only flies if you throw it), the Vault by Sony and the list goes on.

The good, bad, and the ugly. All USB's drives are not created equal, not even close. Beware the “freebies” received from purveyors of hardware and software. One might say you get what you pay for and since you didn't pay for these... I have three that I have received over the years, a 32 MB from a major corporation known for the color blue, completely useless, most computers won't even recognize it; a 128 MB from local computer manufacturer which sort of works, and recently, a 1 GB from a software vendor, which does work but is a recalcitrant child when you try to change it and is slow as molasses.

The good: USB are wonderful devices because they are portable, fit in your pocket, you don't have to worry about formatting them, and they are rewritable so you can store and update files on them, delete files no longer needed,

and carry them from computer to computer. They can be very secure when security software is used. The really bad - They can be easily lost or recycled in a washing machine, gives true meaning to water in the drive, spin dry. Finally, I compare USB drives to a level surface that collects stuff and where stuff is buried, who knows where?

The really good: USB drives are very inexpensive and you can buy a 4 GB unit for under \$20 (on sale). USB 4 GB and 8 GB sizes makes these almost weapons of mass storage, and also of destruction (WMD) if lost.

The ugly: They are slow, CD's are faster even though you can't put them in your pocket, and the potential for problems with inserting them into a computer and removing them is great. To prevent data loss, you should always have Windows “remove” the device first from the system before you physically remove it from the computer. Windows Vista makes this easy as you can simply open your computer, right click on the drive letter, and select safely remove. XP not so easy, you have to click on the funny icon in the system tray, select the device, and say okay to stop. Also, many times, stopping/removing is not clean, Windows thinks the device is busy and it maybe. I have found that is usually easiest and best to just shut down the system before removing. I say best because there is a higher probability of all the data being properly written to the drive and files closed plus I have a certain preference for removing the device from the computer in a powered down state. I don't trust “hot pluggable” devices, anytime you plug and unplug something under power, the potential for damage exists.

### **Portable Applications**

Portable applications that you can run from a USB drive run the gantlet from single executable programs to full copy of OpenOffice. I even ran the XAMPP Apache web server from a USB drive which included the MySQL Database server and PHP Administration tool for MySQL. Wow! And even better, all these applications are free!

The not pretty: Portable Applications for PowerToGo by Lexar and Ceedo. First off, the PowerToGo USB software which runs the Portable Applications, is only free if you have a Lexar Lightning, Mercury or 360 drive. For other Lexar drives, you can get the 30 day trial version which costs \$29.95 for a licensed copy. And

*(Continued next page...)*

## USB Heaven & Points South (Continued from pg. 10)



secondly, many of the applications offered are freeware and can be obtained directly from the publisher as “standalone” applications that are portable by their nature in being fully self contained. You don’t need PowerToGo to run them!

Portable Apps (<http://portableapps.com>) has almost 50 applications that you can run as “portable” including huge applications like OpenOffice and the Gimp photo editor (think free Adobe Photoshop), web server, or even a MacIntosh 128 or Plus on a stick (some assembly required). Portable Apps is very nice because they provide an integrated menu system that comes up when you “hot plug” your USB drive into the computer, an “autoplay” function and they have done a lot of work in integrating the applications. Most applications are “play anywhere” in Windows but applications like the web server install to a “drive letter” and if that changes from computer to computer, it may not run without some tweaking and editing of configuration files.

Another site, makeuserof.com (<http://www.makeuseof.com/>)

[tag/portable-software-usb/](http://tag/portable-software-usb/)), lists 100 Portable applications for your USB Stick (both for Mac and Win).

Once you have your portable applications and are happy with those, the next thing to be considered is would you like to run them from Windows running off a bootable USB drive where you don’t have to rely on the computer’s copy of Windows (although in most cases, you would, for printing, Internet, etc.)

### Bootable USB Drives

Before you go down this path, beware your mileage may vary. Example, I have a USB device which boots great in my newer systems but won’t boot in one of my slightly older systems, an AMD AthlonXP 2200 box even though it is recognized and available on the boot menu. Also after considering the presentation on creating bootable USB drive from a LiveCD such as Ubuntu 7.10, basically making a copy of it to the USB drive. The instructions for doing this can be found at <http://www.pendrivelinux.com/2007/09/28/usb-ubuntu-710-gutsy-gibbon-install/> and I have one built which I use for system testing. However, I strongly advise against Linux “persistent” installs where basically a CD image is copied to the USB drive, for a number of reasons:

1. The potential for data loss and operating system corruption is great. I trashed my Ubuntu 7.10 drive twice trying to do just regular software updates so such installations are not maintainable, and once trashed, all your settings and data are pretty much gone.
2. These “LiveCD” installs are not designed for regular use even though you can store your settings and data files on them.
3. Such solutions are simultaneously good, bad, and ugly. Good because they are easy to build and use, bad because they are slow to load, sometimes with the entire operating system or image being loaded into memory (which makes them fast after they have loaded), and really ugly when they corrupt and/or self-destruct, which generally doesn’t happen with a LiveCD. In my quest, I looked at one “bootable USB” solution where the entire image of the CD was stored on USB drive from which it was booted and the amount of memory required to load the “image” was exactly the same as the size (650 MB). This really slowed the load time and also meant it wouldn’t be usable in any computer with less than 650 MB of RAM.

Continuing my quest for building a bootable USB drive, I

(Continued next page...)

**USB Heaven & Points South** (Continued from pg. 11)

have two success stories to tell, one for Windows Vista and one for Windows XP, both booting off a 1 GB USB stick. The finished “product” – the USB drives are pretty impressive and I learned something about Vista and Windows PE along the way including something called BartPE and how it works. All the software is free for the downloading, even the Vista AIK from Microsoft. While you do have to have a licensed copy of Vista to build it, the USB image is not separately “licensed” and will never ask for a license number or activation.

Before you start, you might want to test your USB drive and system for compatibility by first following the steps found at this site: <http://www.pendrivelinux.com/2007/09/17/testing-your-system-for-usb-boot-compatibility/>.

It should be noted that some of the instructions given here in building a bootable USB drive are a bit bewildering on reading but in practice, as a step-by-step, they will become clear.

### The Vista Story

I actually had this running the night of our user group meeting where Ubuntu on a USB drive was demonstrated and the Portable Apps I demonstrated were loaded on the Vista stick. In building my bootable Vista USB drive, I took my cue from an article entitled “Pimp My USB Stick...” by Mikael Nyström of Norway.

Vista has some very nice tools to aid you in creating a USB bootable drive. The first one is called Diskpart and I found this useful in also creating my WindowsXP bootable drive as well.

To prep a USB drive for booting Windows Vista or other system that boots from an active partition, here are the steps:

Insert the USB drive you on which you intend to install USB Bootable Vista in the computer.

Open a cmd or command box, be sure run “cmd” as an administrator, by right clicking on “run” and selecting, run as administrator (the latter is very important as you have to be administrator for the commands to work) before you type in cmd in the run box.

Start the Diskpart tool from the command line by entering Diskpart. Then enter the following commands in the

command box or window (note that the commands are case sensitive):

1. List disk - Make sure you know which disk number is your USB drive which can be verified by removing your drive, then doing a List disk again (your drive should not be in the list), reinserting the drive, and then List disk one more time, making sure you know which drive the your USB drive. The penalty here is that you can erase a drive you had data on; there are no warnings, if you select the wrong disk number from the list.
2. Sel Disk 1 (assuming your disk is 1, most likely it will not be).
3. clean
4. cre par pri (if this fails and I have seen it fail, usually not enough room, just remove the drive and reinsert it, and start from step 1, cre par pri must be successful).
5. sel par 1
6. active
7. format fs=ntfs (this can also be FAT, FAT32)
8. assign
9. exit

Now you can remove the USB drive and set it aside while you build your boot system.

If you are going to build a bootable USB drive with Vista on it then you need to download something called WAIK from Microsoft. It is fairly big and will take a little time to download. Note that there are now two versions, one for Vista without SP1 and one with SP1. You will see links for both at <http://www.microsoft.com/downloads/results.aspx?pocId=&freetext=WAIK&DisplayLang=en> Note, if you download the one for SP1, be sure to download and install Vista SP1 if you have not already it installed. I recommend that you just use the original WAIK, released in February of 2007. It works fine and it is what I used. After downloading it, you will need to burn the “img” file too a CD, and install it from the CD to Windows Vista.

Process Overview - Note that there actually two phases to this as you go through the build process:

Phase 1:

1. Create a WinPE folder structure.
2. Mount that WinPE wim file.
3. Add features to it.

(Continued next page...)

**USB Heaven & Points South** (Continued from pg. 12)

Step-by-step (Phase 1)

4. Add your own features and tools to it.

Phase 2:

5. Add the Imagex tool to the image. Imagex is the tool you use to capture and deploy images
6. Add the offline servicing stack tool for Vista. With this tool you can add drivers and updates to a Vista machine without booting the OS. Pretty cool.
7. Add the BOOTSECT tool. This tool will help you create the correct master boot code on the disk. /NT52 for Windows XP and /NT60 for Windows Vista.
8. Prep the image to make it smaller.
9. Un-mount the image.
10. Create an ISO image.
11. Load/copy the image onto the USB stick.

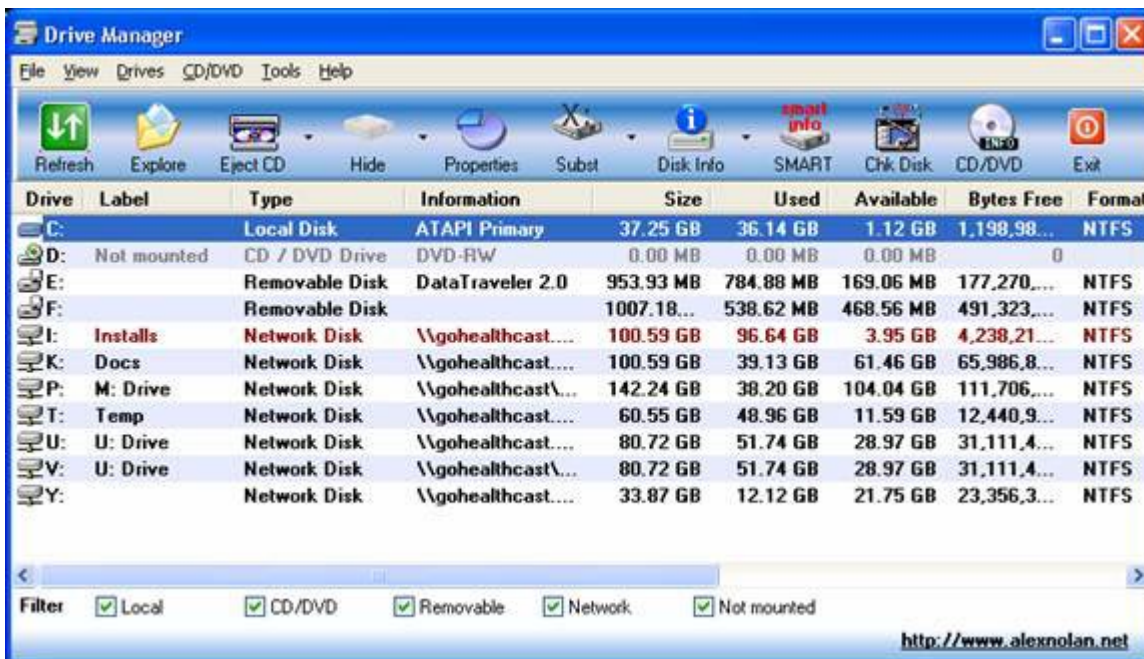
1. Use copye to build the vimfile and the catalog: copye x86 c:\winpe\_x86
2. Use Imagex to mount the vimfile for read and writing: imagex /mountrw c:\winpe\_x86\winpe.wim 1 c:\winpe\_x86\mount
3. Use the tool called "peimg" to add support for HTA, Scripting, XML support, Database support and WMI support
4. peimg /install=\*HTA\* c:\winpe\_x86\mount\windows
5. peimg /install=\*Script\* c:\winpe\_x86\mount\windows
6. peimg /install=\*XML\* c:\winpe\_x86\mount\windows
7. peimg /install=\*MDAC\* c:\winpe\_x86\mount\windows
8. peimg /install=\*WMI\* c:\winpe\_x86\mount\windows

**The Process**

Again, we enter that thing called "command prompt." and be sure to run as Administrator. It's very important that you start the correct command prompt - you'll find it under "All Programs - Windows AIK - Windows PE Tools Command Prompt." (Right click - run as Administrator.) That prompt adds some environment variables, and if you run these variables on Vista, then there should be an elevated command prompt as well. All the following commands are done in the command prompt window.

Now, you'll want to consider adding tools to the "image." There are a couple of considerations in doing this I found. You can put all the tools such as the portable apps in the image but if you do so, then they will only be available while running Vista or you can add them to the USB drive after finishing the boot image system. I did the latter for most of my portable apps so I could use them from the USB drive with regular Windows plus it speeds the startup time for Vista from the USB drive as it reduces

the size of the image. However, the most useful utility I have found to have in the image is DriveMan.exe because it helps me navigate the various drives and drive letters. Driveman.exe, which is available at <http://www.alexnolan.net/software/driveman.htm>. Another program that I would include in the image is



(Continued next page...)

**USB Heaven & Points South** (Continued from pg. 13)

SystemSpec, a utility which provides a significant amount of information about the hardware. It is available at <http://www.alexnolan.net/software/sysspec.htm>.

The process for adding any applications and tools you want inside the image is to copy or install them to a folder someplace and then copy the folder contents with the following command:

```
Copy [path to files]*.* c:\winpe_x86\mount\windows\system32 /Y
```

**Step-by-step (Phase 2)**

1. copy "c:\program files\windows aik\tools\x86\imagex.exe"  
c:\winpe\_x86\mount\windows\system32
2. Imagex is a real cool tool that could be used to capture an image from a disk with the parameter/ capture and to deploy an image with the command/ deploy.
3. copy "c:\Program Files\Windows AIK\Tools\x86\Serviceing"  
c:\winpe\_x86\mount\windows\servicing /s /y
4. copy % windir%\system32\msxml6\*.dll  
c:\winpe\_x86\mount\windows\Serviceing
5. The servicing tools are tools for working with the offline servicing stack in Windows Vista.
6. copy "C:\Program Files\Windows AIK\Tools\PETools\x86\BootSect.exe"  
c:\winpe\_x86\mount\windows\system32
7. Finish steps is to prep that image, unmount it, put it in the right spot, and finally create an ISO file :
8. peimg /prep /f c:\winpe\_x86\mount\windows
9. imagex /unmount c:\winpe\_x86\mount /commit
10. copy c:\winpe\_x86\winpe.wim  
c:\winpe\_x86\ISO\sources\boot.wim /Y
11. osdimg - oscdimg -n -bc:\winpe\_x86\etfsboot.com  
c:\winpe\_x86\ISO c:\winpe\_x86\winpe\_x86.iso

DONE !! The last thing you need to do is re-insert (if you removed it) or make sure the USB drive where you are going to install Windows Vista is in the computer and then copy the files from your hard disk to the USB stick with the command: xcopy c:\winpe\_x86\iso\\*.\* /s /e /f h:\ (assume h: is the drive letter of your USB drive).

Now you have a bootable USB stick that not only boots Windows Vista but will run your portable applications in Vista while still allowing you to run them from the USB on a regular Windows computer as well. If you want to get



brave and explore the many capabilities of this Vista "stick" you will find you can capture and deploy images (backup / restore to and from the USB stick if it's big enough), edit files, copy files, add drivers, manage the operating system and do much, much more. The above graphic shows the directory structure of my Vista USB drive.

**Windows XP on USB**

Another tool in my CD case is the Universal Boot CD for Windows (UBCD), which I have been using since November of 2006. UBCD 4 Windows is a WindowsXP environment that contains many tools for both using and fixing WindowsXP (and other versions of Windows as well). If you want more information, details can be found on the website link below. With a USB implementation, I can add my portable applications and have room for data storage as well. For this project, I decided to build the latest version of UBCD 4 Windows, and install it to the USB stick, if possible. Let the adventure begin! Building the UBCD 4 Windows is a bit time consuming but all the instructions can be found at <http://www.ubcd4win.com/howto.htm>, so I won't repeat them here.

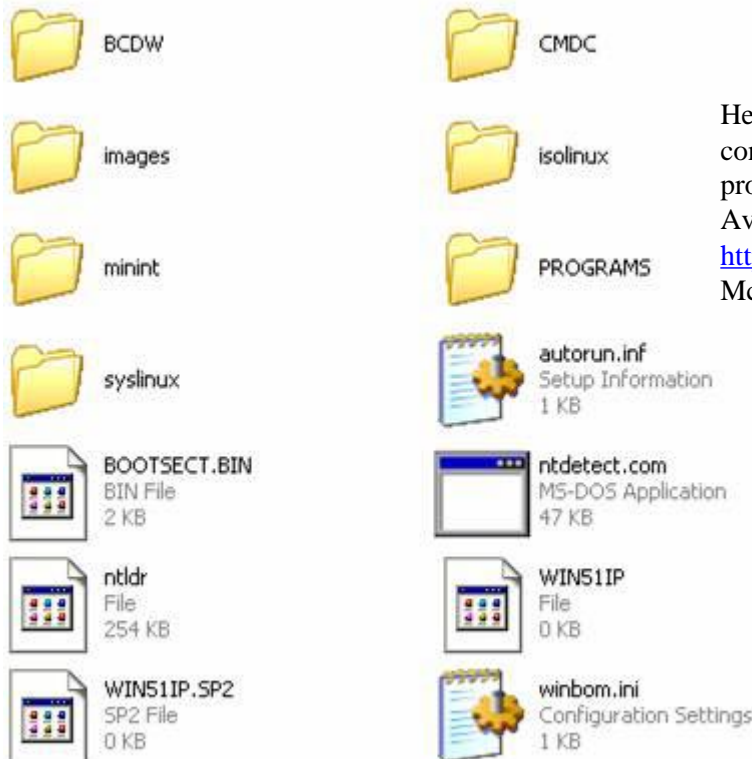
After you get it built, here are the instructions on installing it to the USB that works:

1. Download PEtoUSB (<http://gocoding.com/page.php?al=petousb>). You can extract it (it is a zip file) to anywhere you want but I do recommend you put it in a folder and remember where you put it.
2. Create your UBCD4win load using the UBCD4Win

(Continued next page...)

**USB Heaven & Points South** (Continued from pg. 14)  
 program as described in the howto. No ISO is needed just the created BARTPE folder with all the files for loading to the USB drive.  
 3. Open PEtoUSB and ensure Destination Drive is set to your Flash Drive and as USB Removeable  
 4. Check the box "Enable Disk Format" and for "Source Path to Built BARTPE" just point it to the location of the BARTPE folder that UBCD4win created in your build. **ALSO ENSURE YOU CHECK BOX "ENABLE FILE COPY"** and have it set to overwrite always.  
 5. Once you click start the program it will format the drive and then copy the files it needs to boot from.

I have verified this process, building UBCD4WIN according to the instructions found on the web site posted above and then following the 5 step procedure above. I have built 3 drives this way. I did experience a problem when trying to installing to a Kingston DataTraveler 2 GB USB drive and it turned out to be some sort of incompatibility between PeToUSB and the drive. This was very perplexing for me because I knew the USB drive was bootable but I could not get it to work no matter what I did. What I found was that I could create a 1 GB drive using PeToUSB but the Kingston 2GB drive would immediately fail on boot. I did come up with a workaround on this but that solution is outside the objectives of this



article. I am running my UBCD4WIN booting off that Kingston USB 2 GB drive! Here's the directory structure:

One other note: There have been reports that the USB boot may fail with a BSOD (Blue Screen of Death and a stop at 0x07b BSOD (Blue Screen of Death) before it loads Windows but I have not seen this. If this happens, you may need to replace all instances of NTDETECT.COM, SETUPPLDR.BIN, and RAMDISK.SY\_ with the same files from a Server 2003 CD to fix the problem. You can download the Server 2003 CD at <http://technet.microsoft.com/en-us/windowsserver/bb405947.aspx> but this fix too is beyond the objectives of this article.

A couple of links that may be helpful:  
 How To: Boot BartPE from a FAT32 formatted USB drive (Easy – Well, maybe not, use at your own pleasure.)  
<http://ubcd4win.com/forum/index.php?showtopic=9668>  
 PeToUSB - Boot BartPE From USB Utility Download version 3.0.7  
<http://gocoding.com/page.php?al=petousb>

Linux  
<http://www.pendrivelinux.com/2007/09/28/usb-ubuntu-710-gutsy-gibbon-install/>  
<http://www.pendrivelinux.com/2008/03/28/usb-suse-flash-drive-install/>  
[http://en.opensuse.org/Portable\\_SUSE](http://en.opensuse.org/Portable_SUSE)

**Other Portable Tools**

Here are some sytem and desktop tools you may want to consider adding to your USB drive as standalone programs.

- Avast 4  
<http://www.avast.com>
- McAfee Avert Stinger – detect and remove specific viruses.  
<http://vil.nai.com/vil/stinger/default.aspx>
- Drive Manager  
<http://www.alexnolan.net/software/driveman.htm>
- SystemSpec Utility  
<http://www.alexnolan.net/software/syssspec.htm>
- System Internals  
<http://technet.microsoft.com/en-us/sysinternals/default.aspx>
- Malicious Software Removal Tool  
<http://www.microsoft.com/security/malwareremove/default.aspx>
- Microsoft Security Tools  
<http://www.microsoft.com/technet/security/tools/default.aspx>