# Hidden Disk Areas: HPA and DCO

Mayank R. Gupta
Michael D. Hoeschele
Marcus K. Rogers
Purdue University

## Abstract

This paper focuses on certain manufacturer hidden areas of a hard disk, specifically Host Protected Areas (HPA) and Device Configuration Overlays (DCO). These areas can be problematic for computer forensic investigators, since many of the common industry tools cannot detect the presence of the HPA and DCO. A review of the ATA specifications and recent white papers indicate that these areas can be accessed, modified, and written to by end users using specific open source and freely available tools, allowing data to be stored and/or hidden in these areas. This greatly increases the risk that image acquisitions may not be a true copy of the physical drive in question. This also could result in the obfuscation of data, leading to incomplete or erroneous investigative conclusions. The paper provides an introduction to these commonly used manufacturer areas and discusses their implication to the computer forensics investigative process. Suggestions for future study and testing are also provided.

## Introduction

The Host Protected Area (HPA) as defined is a reserved area on a Hard Disk Drive (HDD) (T13, 2001). It was designed to store information in such a way that it cannot be easily modified, changed, or accessed by the user, BIOS, or the OS. This area can contain information ranging from HDD utilities, to diagnostic tools, as well as boot sector code. An additional hidden area on many of today's HDDs is the Device Configuration Overlay (DCO). The DCO allows system vendors to purchase HDDs from different manufacturers with potentially different sizes, and then configure all HDDs to have the same number of sectors. An example of this would be using DCO to make an 80 Gigabyte HDD appear as a 60 Gigabyte HDD to both the OS and the BIOS.

Usually when information is stored in either the DCO or HPA area, it is not accessible by the BIOS, OS, or the user. However, certain tools can be used to modify the HPA or DCO. Given the potential to place data in these hidden areas, this is an area of concern for computer forensics investigators. An additional issue for forensic investigators is imaging the HDD that has the HPA and or DCO on it. While certain vendors claim that their tools are able to both properly detect and image the HPA, they are either silent on the handling of the DCO or indicate that this is beyond the capabilities of their tool.

Due to the secretive nature of most HDD manufacturers, the HPA and DCO may not be familiar to some investigators. In order to rectify this situation, the next sections will describe the characteristics of both the HPA and the DCO in detail.

## Host Protected Area (HPA)

The HPA was first introduced in the ATA-4 standard (T13, 2001). The SET MAX ADDRESS command compartmentalizes the HDD into the user accessible and protected areas. The starting address of the protected area is the maximum user address +1 sector. For example, if the maximum addressable user area is 2343 sectors in size, then the protected area starts from sector 2344. As mentioned, the HPA commonly contains diagnostic utilities, as well as the boot sector code; the exact content depends upon the manufacturer. For the purposes of this discussion, the description of how the HPA is organized will be restricted to drives using only the Phoenix BIOS. According to the Protected Area Run-Time Interface Extension Services (PARTIES) document, the information about the HPA is contained in the Boot Engineering Extension Record (BEER) (T13, 2000). The BEER itself is a part of PARTIES and is a record that contains non-volatile configuration information about the HDD and is stored in the native maximum address, which is the last sector on the HDD. The BEER also contains information about the user addressable sectors, start of the reserved area, and the code for the boot area. The BEER has a header that is 128 bytes long and immediately following the header is the directory of services with 64 bytes of information. It is here in the directory of services where the diagnostic utilities are stored.

As mentioned, the primary function of the HPA is to store diagnostic utilities as well as a boot record; this is useful when it is not possible to boot from the primary partition. One can use the SET MAX ADDRESS command to reset the HPA to the maximum user addressable sectors, and then boot from what was the HPA. If the volatile bit is also set then the HDD retains the new values on power up or reboot.

Any HDD that supports the HPA will also support the commands READ NATIVE MAX ADDRESS and SET MAX, as described in the working draft of ATA-6 interface. In addition to the commands mentioned above, if the device supports HPA and 48-bit addressing, then the device will also support the two additional commands READ NATIVE MAX ADDRESS EXT and SET MAX ADDRESS EXT.

According to the ATA-6 working draft by T13 (2001), the device shall set the "10$^{th}$ bit of word 82 to indicate that the host protected feature set is supported" (p. 44)**.** The use of the SET MAX ADDRESS or SET MAX ADDRESS EXT command is prohibited if the removable media feature set is implemented. The removable feature set, as described by Stephens (1997), prevents the loss of data by locking the HDD until completion of a cached write. The SET MAX ADDRESS or the SET MAX ADDRESS EXT command will always be preceded by the READ NATIVE MAX ADDRESS or the READ NATIVE MAX ADDRESS EXT commands. These are the two commands that are used to determine the original HDD capacity. After the SET MAX ADDRESS command has been issued, the HDD will indicate that a HPA has been configured. According to Landis (personal communication, May 27, 2005), "If bit 10 of word 85 is set to one, the Host Protected Area feature set is enabled." The volatile bit in the sector count register specifies whether the new address set has to be preserved across power-on or hardware reset

cycles. If the volatile bit is set to 0 then the drive will revert back to the last address that was set on the non-volatile SET MAX ADDRESS command. Any read or write access attempts to addresses greater than that specified by SET MAX ADDRESS or SET MAX ADDRESS EXT command will result in an ID Not Found Error (IDNF).

If the numbers of sectors on the HDD are greater than 268,435,455, then the HDD will support both 48-bit addressing and HPA (McLean, 2000a). In this case the SET MAX ADDRESS EXT command will be used to set and reset the HPA. The READ NATIVE MAX ADDRESS EXT command will return the HDD capacity. If the READ NATIVE MAX ADDRESS command is used, then the maximum capacity returned is 268,435,454. As mentioned above, HPA can be set and removed using the SET MAX ADDRESS commands. If the HPA was configured using the SET MAX ADDRESS command, then it can be removed only using the SET MAX ADDRESS command and not SET MAX ADDRESS EXT command. The reverse rule also applies for SET MAX ADDRESS EXT command. If the SET MAX ADDRESS EXT command was used to configure the HPA, then SET MAX ADDRESS cannot be used to again change the drive capacity.

The output of the Identify Device command depends upon the prior SET MAX ADDRESS command issued. If only 28-bit addressing is supported, any change that is made by the SET MAX ADDRESS command will be indicated in the words 60-61, and the words 100-103 will contain 0.0. The value reported by the words 60-61 give the actual capacity of the hard disk. The identify device command is issued by the host to receive parameter information about the device (T13, 2001). Examples of information these parameters provide are the total number of sectors on the drive, the ATA standards the device supports, and whether the drive supports HPA / DCO or not. HPA can also be detected by comparing the value in the words 60-61 with the actual specification of the hard disk. It is also important to note that any HDD that supports 48 bit addressing also supports 28 bit addressing. If 48-bit addressing is also supported and the SET MAX ADDRESS EXT command was used with the number of sectors greater than 268,435,455, then the new capacity of the HDD will be indicated by words 100 -103 and the value in the words 60-61 will be 268,435,455. However, if the SET MAX ADDRESS command is used and the number of sectors is less than 268,435,455, then the drive capacity will be indicated in words 60-61 and 100-103. Any attempt to access the sectors beyond the values specified by the identify device command will result in an Identify Device Not Found (IDNF) error.

### Address offset

In order to boot from the reserved area and to utilize the disk utilities, address-offset boot method was proposed in the address offset feature proposal (Colegrove, 1998). This method allows HDD to boot from its reserved area. If the device supports the address-offset feature, then it will do so by setting bit 7 of identify device word 83. The typical use would be to first set the HPA using the non-volatile SET MAX ADDRESS command, and then issue the SET FEATURES command to the HDD. This will result

in changing the location of the first sector (LBA 0), to the start of the protected area that was set using the non-volatile SET MAX ADDRESS command. Due to this change, the former user area now becomes the reserved area. Once the SET FEATURES command has been issued, the device will set bit 7 of the Identify device word 86 and indicate that the device is in the offset mode.

These commands are executed during the boot process so that the BIOS can boot from the reserved area instead of the user area. However, if the HPA is not established on the HDD, then the SET FEATURES command will result in an abort error. Upon entering the address-offset mode, the size of the HDD returned by the identify device command will be the size of the former HPA. A subsequent SET MAX ADDRESS command using the values returned by the READ NATIVE MAX ADDRESS command will allow access to the entire drive. The SET FEATURES command is volatile and is reset on the next power cycle. Therefore, on restart/hardware reset, the drive will be restored to the state it was in before entering the address-offset mode. This command is not valid if the HDD capacity is reduced by the DEVICE CONFIGURATION SET command and HPA is not set.

**Device Configuration Overlays**

The DCO feature was first introduced in ATA-6 standard. The DCO was proposed to allow device manufacturers to build one version of the product to satisfy all the customer needs and to allow the PC manufacturers to purchase different HDD of almost the same capacity; then make every drive have the exact same number of sectors (McLean, 2000b). The DCO can also be used to enable and disable features on the HDD.

The DEVICE CONFIGURATION SET command can be used to reduce the capacity of the HDD by setting the device parameters, or LBA, to the desired value. If the HDD supports DCO, then it will do so by setting bit 11 of words 83 and 86 in the identify device command. In this case the identify device command will indicate the reduced capacity in words 60-61 and 100-103. Accessing the HDD beyond the newly specified sectors will result in an IDNF error. The device will return an error or command aborted if the HPA is already set using the SET MAX ADDRESS or the SET MAX ADDRESS EXT command. However, the device can execute the DCO set command if the HPA has been removed or the drive has been reset back to the factory settings. This can be done using the SET MAX ADDRESS or SET MAX ADDRESS EXT commands. The DCO can only be removed using the DEVICE CONFIGURATION RESTORE command. It is important to note that the DEVICE CONFIGURATION RESTORE command cannot be used to remove the HPA.

## Co-existence of HPA and DCO

The DCO and HPA can co-exist on the same HDD. However, in order for this to happen, the DCO must be set first using the DEVICE CONFIGURATION SET command and then the HPA is configured using the SET MAX ADDRESS or the SET MAX ADDRESS EXT command. The new capacity of the HDD will be determined by the parameters that are given to the SET MAX ADDRESS and DEVICE CONFIGURATION SET commands. In this case the output of the READ NATIVE MAX ADDRESS or READ NATIVE MAX ADDRESS EXT command will give the value that was set by the DEVICE CONFIGURATION SET command. Hence the BIOS will also report the size of the disk as reported by the READ NATIVE MAX ADDRESS commands. The reason for this is that during the boot process the BIOS issues only the READ NATIVE MAX ADDRESS commands and not the DEVICE CONFIGURATION IDENTIFY command, which is the only command that will report the true size of the hard disk when the DEVICE CONFIGURATION SET command is used.

## Investigative Significance

Since an end user can modify and write to the HPA and DCO, allowing them to potentially hide data, forensics investigators need to be aware of these a two areas. As mentioned earlier, the HPA and DCO are hidden from the OS, BIOS, and the user. It is also possible to create an HPA that is approximately the same size as the HDD. This means that the HPA, DCO, or the HPA and DCO combined can potentially store large amounts of information, invisible to the investigator and/or the acquisition and analysis tools.

Certain forensic tools can be used to detect the HPA[1]. These tools include Encase, Sleuth Kit and ATA forensic tool (see Table 1). Carrier (2004) lists three different methods for detecting HPA on Linux. The utilities used by these three methods are dmesg, hdparm, and disk_stat. The first two are built in utilities in Linux and the third one is a tool in the Sleuth Kit. It is also possible to reset the HPA until the next reset using the disk_sreset utility in Sleuth Kit (Carrier, 2005). According to Vidstrom (2005b), it is possible to set and reset HPA and DCO using the ATA forensics tool. The report also mentions that Encase v4.18a can detect HPA when booted in DOS with ATA mode. However, Encase v4.18a was not able to detect HPA in Windows mode. It is important to note that both Encase and Sleuth Kit cannot be used for detecting or setting DCO (Vidstrom, 2005b). Caution should be used when employing forensic tools that do not support the detection of both the HPA and DCO.

[1] The list of forensic tools mentioned in this paper is not at all inclusive

**Table 1: Forensic Tools**

| Tool | Programmer/Vendor | Version |
|---|---|---|
| The Sleuth Kit | Brian Carrier | 2.02 |
| ATA Forensics Tool | Arne Vidstrom | 1.1 |
| Encase | Guidance Software | 4.20 |

As mentioned earlier, Encase in windows mode cannot properly recognize the combination of HPA/DCO. This can be problematic. Consider an HDD with the HPA configured. If two images are acquired form this HDD, one using Encase in Windows mode and the other using Sleuth Kit in Linux, the MD5 checksums may not be equivalent. According to Scalet (2005), the best evidence copy is usually the first copy that is made from the evidence. In order to prove that the integrity of the original evidence has not been compromised, an MD5 checksum of the image should match with the MD5 of the original HDD. In the example given above, because of the possibility of the two MD5's not matching, the integrity or fidelity of the image is questionable.

Although current automated tools may not deal with HDDs containing the HPA/DCO in a forensically sound matter, if at all, investigators have other options. Investigators using a Linux system to acquire the suspect's HDD can use several open source command line tools to assist them in dealing with these hidden areas. The use of these tools appears to be consistent with the forensic axiom of preventing or minimizing changes to the scene or in this case the original HDD. For example, using setmax.c it is possible to modify the HPA in order to look for hidden data (Brouwer, n.d.). The program issues the SET MAX ADDRESS command that results in changing the values of the words 60-61, 100, 101, 102 and 103. The SET MAX ADDRESS command does not change data in any sectors of the hard disk[2]. During restore, resizing, or when the BIOS issues a READ NATIVE MAX ADDRESS command, there is also no change to the user data; it is not changed either in the user addressable area or in the reserved area. In addition, according to the drive manufacturers, setting and resetting DCO does not modify the user data, thus the integrity of the evidence should not be affected.

## Conclusions

The paper describes the HPA and DCO and introduces a scenario where the HPA and DCO can co-exist on the same HDD. The HPA and DCO are hidden from the OS and BIOS and can potentially contain evidence, as research indicates the end user can

[2] Readers interested in a more detailed discussion of using these tools for investigations should see Carrier (2004) and Carrier (2005).

modify these areas. The ability to obfuscate evidence makes the HPA and DCO a concern for investigators. When imaging an HDD, the investigator must be aware that any HDD that supports ATA-6 and above can contain HPA and or DCO. However, if the HDD supports ATA-4 or 5, it only has the potential to contain HPA. Given that the majority of current HDDs support ATA-6 and above, it is extremely important for current forensics practitioners to be aware of the HPA and DCO and use appropriate tools.

## Future Research

Future research should focus on the methods by which forensics tools detect the HPA and or DCO (as well as under what constraints), and what affects these methods might have on the acquisition process. An additional area of interest is whether the HDD can be "booted" from DCO in the presence of HPA. Finally, subsequent research should conduct general testing of the HPA and DCO to determine what effects creating and removing these areas has on the integrity of data on a HDD.



## About the Authors:

Marcus Rogers is an Associate Professor of Computer and Information Technology at Purdue University, West Lafayette, Indiana. His research interests include applied digital forensics, psychological crime scene analysis and information assurance. He can be reached at rogersmk@purdue.edu.

Mayank Gupta is an M.S. student in Computer and Information Technology at Purdue University, West Lafayette, Indiana. His research interests include Digital Forensics and Computer Networks. He can be reached at mrg81@purdue.edu.

Michael Hoeschele is an M.S. student in Computer and Information Technology at Purdue University, West Lafayette, Indiana. His research interests include social engineering, digital forensics and ethics. He can be reached at mhoesche@purdue.edu.

## References


Access Data. (2005). *Forensic tool kit*. 1.6. Retrieved January 25, 2006, from http://www.accessdata.com

Brouwer, A. E. (n.d). *Setmax.c*. Retrieved January 24, 2006, from http://www.win.tue.nl/~aeb/linux/setmax.c

Carrier, B. (2004). *Detecting host protected areas (HPA) in Linux*. Retrieved January 22, 2006, from http://www.sleuthkit.org/informer/sleuthkit-informer-17.html

Carrier, B. (2005). *Removing host Protected Areas (HPA) in Linux*. Retrieved January 22, 2006, from http://www.sleuthkit.org/informer/sleuthkit-informer-20.html

Colegrove, D. (1998). *The address offset feature proposal*. Retrieved October 5th, 2005, from http://www.t13.org/technical/d98123r1.pdf

Guidance Software. (2005). *Encase forensic edition 4.20*. Retrieved January 25, 2006, from http://www.encase.com

McLean, P. (2000a). *48-bit LBA proposal*. Retrieved October 5 2005, from http://www.t13.org/technical/e00101r0.pdf

McLean, P. (2000b). *Device configuration overlay proposal*. Retrieved October 5th, 2005, from http://www.t13.org/technical/e00140r0.pdf

Scalet, S. D. (2005). *How to keep a digital chain of custody*. Retrieved January 23, 2006, from http://enterprisesecurity.symantec.com/content.cfm?articleid=6313&EID=0

Stephens, R. (1997). *Removable media feature set*. Retrieved October 6th, 2005, from http://www.t13.org/technical/d97120r1.pdf

Technical Committee T13. (2000, September 30). *Protected area run-time Interface extension services*. Retrieved January 25 2005, from http://www.t13.org/project/d1367r1-PARTIES.pdf

Technical committee T13. (2001). *AT-Attachment with packet interface-6*. Retrieved October 5 2005, from http://pdos.csail.mit.edu/6.828/2005/readings/hardware/ATA-d1410r3a.pdf

Vidstrom, A. (2005a). *The ATA forensic tool 1.1*. Retrieved January 25, 2006, from http://www.vidstrom.net/stools/taft

Vidstrom, A. (2005b). *Computer forensics and the ATA interface*. Retrieved January 25, 2006, from http://www.foi.se/upload/rapporter/foi-computer-forensics.pdf