The Return of Unix Command-Line Kung Fu

Forensic Edition

Who Is Hal Pomeranz?

- Independent consultant
- SANS Faculty Fellow, "oldest" surviving SANS instructor
  - Author, track lead for Sec506: Linux/Unix Security
  - Instructor for SANS Forensics classes
- Did I mention the blogs?
  - commandlinekungfu.com (w/ Ed Skoudis and Tim Medin)
  - http://blogs.sans.org/computer-forensics/

Mighty `file` Fu

```
# file core
core: ELF 64-bit LSB core file x86-64, version 1 (SYSV),
  SVR4-style, from 'ssh localhost'
# file dev_sda2.dd
dev_sda2.dd: Linux rev 1.0 ext3 filesystem data,
  UUID=0bdec38a-b63d-11d7-9522-87905b54ba45 (needs journal
  recovery)
# file 'The Return of Unix Command-Line Kung Fu.ppt'
The Return of Unix Command-Line Kung Fu.ppt: CDF V2
  Document, Little Endian, Os: Windows, Version 5.1, Code
  page: 1252, Title: Unix Command-Line Kung Fu, Author:  ,
  Template: Origin, Last Saved By: Hal Pomeranz, Revision
  Number: 132, Name of Creating Application: Microsoft
  Office PowerPoint, Total Editing Time: 1d+17:01:36,
  Create Time/Date: Tue Jan 15 14:08:20 2008, Last Saved
  Time/Date: Mon Mar 8 01:55:54 2010, Number of Words: 926
```

Time (Zones) Are Relative

```
$ date
Wed Mar  3 10:04:56 PST 2010
$ ls -l /etc/passwd
-rw-r--r-- 1 root root 1656 2009-12-05 10:31 /etc/passwd
$ export TZ=EST5EDT
$ date
Wed Mar  3 13:05:11 EST 2010
$ ls -l /etc/passwd
-rw-r--r-- 1 root root 1656 2009-12-05 13:31 /etc/passwd
```

- Very useful when looking at a system image from a different time zone than your analysis workstation

Timestamps too!

- Use **touch** (as root) to manipulate timestamps at will:

```
# touch -t 201001010000 /tmp/testing
# stat /tmp/testing
  File: `/tmp/testing'
  Size: 0               Blocks: 0          IO Block: 4096
Device: fc01h/64513d Inode: 5603        Links: 1
Access: (0644/-rw-r--r--)  Uid: (0/root)   Gid: (0/root)
Access: 2010-01-01 00:00:00.000000000 -0800
Modify: 2010-01-01 00:00:00.000000000 -0800
Change: 2010-03-08 16:42:33.993133369 -0800
```

Use Your `touch` for Good, Not Evil

- "find … -mtime …" only works on one-day intervals:

```
find / -mtime -7
```

- Combine touch with "find … -newer …" and do better:

```
touch -t 201003021337 /tmp/timestamp
find / -newer /tmp/timestamp
```

Since We're Talking Timestamps…

- Sort directory entries by last modified time (-t):

```
ls -lrt
```

- Add **-u** option to sort by last access time

What About ctime?

- Hacking ctime values generally requires specialized tool
- Enter **`debugfs`** on Linux EXT file systems:

```
# df -h /tmp/testing
Filesystem            Size  Used Avail Use% Mounted on
/dev/mapper/elk-root  961M  665M  247M  73% /
# debugfs -w -R 'set_inode_field /tmp/testing
                 ctime 201001012222' /dev/mapper/elk-root
# debugfs -R 'stat /tmp/testing' /dev/mapper/elk-root
[…]
 ctime: 0x4b3ee608:ecc80ce4 -- Fri Jan  1 22:22:00 2010
 atime: 0x4b3dab80:00000000 -- Fri Jan  1 00:00:00 2010
 mtime: 0x4b3dab80:00000000 -- Fri Jan  1 00:00:00 2010
[…]
```

What's Going on Here?

```
# stat /tmp/testing | tail -3
Access: 2010-01-01 00:00:00.000000000 -0800
Modify: 2010-01-01 00:00:00.000000000 -0800
Change: 2010-03-08 16:42:33.993133369 -0800
# echo 2 >/proc/sys/vm/drop_caches
# stat /tmp/testing | tail -3
Access: 2010-01-01 00:00:00.000000000 -0800
Modify: 2010-01-01 00:00:00.000000000 -0800
Change: 2010-01-01 22:22:00.993133369 -0800
```

- Flushing caches also useful when doing perf analysis

Wiping Clean

- Single File:

```
shred -u myfile
```

- Entire Device:

```
dd if=/dev/zero of=/dev/sdb bs=1M
```

- Unallocated space in file system:

```
dd if=/dev/zero of=junk bs=1M; rm junk
```

Fun With FIFOs

- You want to capture command output with **`script`**
- **`script`** wants to write to a local file
- This is bad from a forensic perspective
- Use a FIFO instead!

```
# mkfifo /tmp/fifo
# cat </tmp/fifo >/dev/tcp/192.168.1.1/8001 &
[1] 3066
# script -f /tmp/fifo
Script started, file is /tmp/fifo
```

More Fun With FIFOs

- You have a large disk image file
- You want to dump both ASCII and Unicode strings
- You don't want to have to read the image twice
- Use tee command with FIFO:

```
# strings -a -t d -e l </tmp/fifo | \
    gzip > strings.uni.gz &
[1] 23281
# cat ntfs.img | tee /tmp/fifo | \
    strings -a -t d | gzip >strings.ascii.gz
```

Finishing Up

- Any final questions?
- Thanks for participating!
- Please fill out your surveys

http://commandlinekungfu.com

http://blogs.sans.org/computer-forensics/

http://www.deer-run.com/~hal/