



LINUX COMMAND LINE

Hal Pomeranz



THE STORY OF HAL

Became a Unix admin in the 1980s

Job history? Let's just say I've been around...

blog.commandlinekungfu.com

Currently – independent DFIR consultant, expert witness

Slides here – deer-run.com/~hal/cldojo/

ALL SORTS OF SORTS

```
$ ps -ef | awk '{print $1}' | sort | uniq -c | sort -n
```

...

1 nobody

1 polkitd

1 rtkit

1 UID

2 avahi

2 postfix

57 hal

153 root

Alphabetic sort

Numeric sort

SORT ON DELIMITED FIELDS

```
$ sort -t: -k3,3 -n passwd
```

```
evilyn:x:0:0:Evilyn Admin:/usr/share/doc/.evillair:/bin/bash
```

```
root:x:0:0:root:/root:/bin/bash
```

```
bin:x:1:1:bin:/bin:/sbin/nologin
```

```
daemon:x:2:2:daemon:/sbin:/sbin/nologin
```

```
adm:x:3:4:adm:/var/adm:/sbin/nologin
```

```
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
```

```
sync:x:5:0:sync:/sbin:/bin/sync
```

```
...
```

MULTIPLE KEYS

```
$ sort -t. -k1,1 -k2,2 -k3,3 -k4,4 -n ip-addrs
```

```
2.72.83.54
```

```
2.85.146.242
```

```
2.90.13.196
```

```
3.33.224.46
```

```
3.54.228.54
```

```
3.105.95.4
```

```
3.109.196.161
```

```
3.124.109.232
```

```
3.137.148.37
```

```
...
```

MAKING TEST DATA IS FUN (1)

\$RANDOM

gives a random value from 0 to 32K

\$ ((\$RANDOM % 255))

changes range to 0 to 254

Do it four times with dots in between for a random IP address

Put it in a loop and you have...

MAKING TEST DATA IS FUN! (2)

```
$ for i in {1..1000}; do
    echo $(( $RANDOM % 255 )) . $(( $RANDOM % 255 )) . $(( $RANDOM % 255 ))
done >ip-addr
```

FINDING YOUR WAY AROUND

```
find /dev -type f
```

There really shouldn't be many regular files under /dev

```
find / -xdev -type d -name .\*
```

Dot directory names uncommon outside of user home dirs

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

Yeah they could alter timestamps, but mostly they don't

IT'S COMPLICATED

```
find / -xdev -type f \( -perm -04000 -o -perm -02000 \)
```

Do you know where all your SUID/SGID progs are?

```
find / -xdev -type d
```

```
\( -name bin -o -name sbin -o -name lib \) | wc -l
```

How many do you think we'll find?

EXECS ARE SLOW

```
find /var/log -type f -mtime +7 -exec gzip {} \;
```

*Works, but slow because **-exec** spawns a shell for each file found*

```
find /var/log -type f -mtime +7 | xargs gzip
```

xargs** collects file names into groups to limit number of calls to **gzip

WHICH IS FASTER?

```
grep -r1 PATH_MAX /usr/include
```

```
find /usr/include -type f -exec grep -l PATH_MAX {} \;
```

```
find /usr/include -type f | xargs grep -l PATH_MAX
```

THE HAEMER HAMMER

Directories named `<version>-<date>`:

```
$ ls
```

```
1.2.00.00_devel-20190903  2.0.00.00_devel-20190906  
1.2.00.00_devel-20190905  2.0.00.00_devel-20190907  
1.2.00.00_devel-20190907  2.0.00.00_devel-20190908  
1.2.00.00_devel-20190909  2.0.00.00_devel-20190909  
1.2.00.00_devel-20190910  2.0.00.00_devel-20190910
```

Remove all but the *two most recent* copies of each version group



WORK IT IN PIECES

Get the version numbers

Filter out the two most recent directories

Remove the directories

PIECE WORK

Get the version numbers

```
$ ls | cut -d- -f1 | uniq  
1.2.00.00_devel  
2.0.00.00_devel
```

Filter out the two most recent directories with a clever trick

```
$ ls -rd 2.0.00.00_devel* | tail -n +3  
2.0.00.00_devel-20100908  
2.0.00.00_devel-20100907  
2.0.00.00_devel-20100906
```

PIÈCE DE RÉSISTANCE

```
for dir in $(ls | cut -d- -f1 | uniq); do  
    ls -rd $dir* | tail -n +3  
done | xargs rm -rf
```



SPEND MORE TIME IN THE DOJO

One-day lab-based online training is coming!

We welcome your questions and suggestions!

Hal Pomeranz

hal@deer-run.com

@hal_pomeranz