X-Raying Files: Using grep for Fame and Profit

Aaron Burgemeister
Identity / Security / Linux Consultant

aburgemeister@gmail.com
https://www.linkedin.com/in/aaronburgemeister

Keyboards and Mice

- Mice are slow and imprecise
  - Rely on relative position
  - Couple of buttons
  - Couple of fingers
- Keyboards are fast and have many options
  - Rely on absolute position
  - 100+ buttons
  - Ten digits

Linux Philosophy

- Do one thing, and do it well
- Makes commands chainable
- Great power, responsibility, etc.

Poll

- Experience w/Linux
  - Command line?
  - Scripting
  - Regular Expressions / Perl Compatible (PCRE)
    - Know what they are
    - Can make them
    - Can read them

Philosophy in Information Technology

- Computers vs. Humans
  - Humans are good at creativity.
  - Computers are good at brute forcing specific tasks.
- Your value to your organization
  - Your ability to do things
  - Your ability to communicate with others
  - “Others” in the IT world is not limited to mere mortals
  - Computers are not great at figuring out IT things, but we are, so it’s up to us to communicate on computers’ levels.

Computer Communication: grep

- Tool used by experts in IT, CS, and others since at 1974
- Designed to look at file contents, or contents of streams
  - Not focused on metadata like file names, sizes, etc.
    - The droid you are seeking there is called “find”
- Uses Regular Expressions (regexes)
  - Fancy strings to match data patterns
    - String: a set of bytes, usually printable
  - Some regexes are not fancy strings
    - Every word in every language could be counted

More grep Information

- grep
  - -i = case-insensitive
  - -R = recursive in a filesystem directory structure
  - -v = inverse (only returned unmatched lines)
  - -e 'pattern' -e 'anotherpattern' -e 'yetanother'
  - -l = only print the file name of a matching file
  - -L = only print the file name of a non-matching file
  - -n = Print the line number along with the match
  - ^ = Match start of line
  - $ = Match end of line
  - . = Match any non-newline character

‘For the Win’ == Bash + Grep

- Command Line Power
  - Know which commands
  - Know how to mix commands
  - Learn more options of commands (man pages, info, Google)
- Pipelines
  - command | othercommand | othercommand
  - Each command, being specialized to do its one task very well, adds to the total value of the implementation
- In the human world combining multiple specialists who work well together is called a profitable company, not a pipeline, but probably for historical reasons only.

Wordle: What's grep Got To Do With It?

- Wordle
  - Game created to see if a person can guess a daily five-letter word in six tries with a few clues from input words
  - https://www.nytimes.com/games/wordle/index.html
  - Makes it easy to share personal successes with others
    - Bragging without giving away the answer.
- In the end, Wordle is solved using patterns
  - e.g. What five-letter word starts with L, ends with Y, contains a K, but does not have any of these:
    - AEIOURSTN

Wordle: What's grep Got To Do With It?

- Wordle Winning Requirements:
  - Know five-letter words
  - Be able to retrieve those words with letters in the right places, or invalid letters omitted, or valid letters in any place,
  - Helpful hint: having a good starting word, or a few, helps lead to clues.
- Aaron's Philosophy on Computers vs. Humans
  - Human: Come up with creative solution to…
  - Computer: Do brute force calculations

Word(le) List

- All of you probably have one on your Linux/Unix-based servers.
  - rpm -qf /usr/share/dict/american
    - Symlinked from /usr/share/dict/words
  - Otherwise: zypper install words
  - 307,993 “words” for you to grep (on my laptop).
    - Not all are five characters long
    - Not all are common words used by Wordle

Wordle grep Examples

Find the five-character strings:

> grep '^.....$' /usr/share/dict/words

Find strings without wrong characters:
> grep -v -e '[wrongchars]' /usr/share/dict/words

Find strings with right characters:
> grep -e '[rightchars]' /usr/share/dict/words

Combine the Fun: Create a Script

- wordlesolver.sh
- Works with the output from Wordle to narrow down choices
- Lets you specify:
  - Known positional characters
    - e.g. first character is L, last character is Y
  - Known wrong characters: e.g. AEIORSTN
  - Known valid characters: e.g. K

  > L...Y:aeiorstn:k