Data Wrangling

Overview

- **Data Wrangling**: Converting data from one format to another.
- **Tools**: Utilizing various tools to manipulate and transform data.
- **Pipes** (`|`): A common data wrangling operation.

Starting with Data Wrangling

- **Need**: Data to wrangle and a tool to process it.
- **Example**: Investigating server logins.

```
ssh myserver journalctl
```

Filtering SSH Entries

- **Limiting to SSH**: Using `grep` to filter logs.

```
ssh myserver journalctl | grep sshd
```

Streamlining the Output

- **Refining our search**: Further filtering and using `less`.

```
ssh myserver 'journalctl | grep sshd | grep "Disconnected from"' | less
```

Saving Filtered Logs

- **Reducing network load**: Saving logs to a file.

```
$ ssh myserver 'journalctl | grep sshd | grep "Disconnected from"' > ssh.log
$ less ssh.log
```

Introducing `sed`

- **Stream Editor**: `sed` is a powerful tool for text manipulation.
- **Substitution**: Using `sed` for pattern replacement.

```
ssh myserver journalctl
 | grep sshd
 | grep "Disconnected from"
 | sed 's/.*Disconnected from //'
```

Regular Expressions

- **Powerful Pattern Matching**: Understanding regular expressions.
- **Common Patterns**:
  - `.`: Any single character
  - `*`: Zero or more of the preceding match
  - `+`: One or more of the preceding match
  - `[abc]`: Any one character of `a`, `b`, and `c`
  - `^`: The start of the line
  - `$`: The end of the line

Using `sed` to Clean Logs

- **Example**: Removing unwanted prefixes and suffixes.

```
| sed -E 's/.*Disconnected from (invalid |authenticating )?user .* [^ ]+ port [0-9]+( \[preauth\])?$//'
```

Capturing Groups in `sed`

- **Preserving Important Data**: Using capture groups to keep the username.

```
| sed -E 's/.*Disconnected from (invalid |authenticating )?user (.*) [^ ]+ port [0-9]+( \[preauth\])?$/\2/'
```

Regular Expressions for Email

- **Complexity**: Matching an email address is not straightforward.
- **Resources**:
  - Articles, discussions, tests, and regex debuggers.

Back to Data Wrangling

- **Sorting and Counting**: Identifying common usernames.

```
ssh myserver journalctl
 | grep sshd
 | grep "Disconnected from"
 | sed -E 's/.*Disconnected from (invalid |authenticating )?user (.*) [^ ]+ port [0-9]+( \[preauth\])?$/\2/'
 | sort | uniq -c
 | sort -nk1,1 | tail -n10
```

Generating Comma-Separated Lists

- **Combining `awk` and `paste`**: Transforming data for different uses.

```
ssh myserver journalctl
 | grep sshd
 | grep "Disconnected from"
 | sed -E 's/.*Disconnected from (invalid |authenticating )?user (.*) [^ ]+ port [0-9]+( \[preauth\])?$/\2/'
 | sort | uniq -c
 | sort -nk1,1 | tail -n10
 | awk '{print $2}' | paste -sd,
```

awk – Another Editor

- **Text Processing Language**: Basics of `awk`.

```
| awk '$1 == 1 && $2 ~ /^c[^ ]*e$/ { print $2 }' | wc -l
```

- **Using `awk` for calculations**:

```
BEGIN { rows = 0 }
$1 == 1 && $2 ~ /^c[^ ]*e$/ { rows += $1 }
END { print rows }
```

Analyzing Data with bc and R

- **Calculating**: Using `bc` for in-shell calculations.

```
| paste -sd+ | bc -l
```

- **Statistics**: Using `R` for data analysis.

```
ssh myserver journalctl
 | grep sshd
 | grep "Disconnected from"
 | sed -E 's/.*Disconnected from (invalid |authenticating )?user (.*) [^ ]+ port [0-9]+( \[preauth\])?$/\2/'
 | sort | uniq -c
 | awk '{print $1}' | R --no-echo -e 'x <- scan(file="stdin", quiet=TRUE); summary(x)'
```

Simple Plotting with gnuplot

- **Visualization**: Using `gnuplot` for plotting data.

```
ssh myserver journalctl
 | grep sshd
 | grep "Disconnected from"
 | sed -E 's/.*Disconnected from (invalid |authenticating )?user (.*) [^ ]+ port [0-9]+( \[preauth\])?$/\2/'
 | sort | uniq -c
 | sort -nk1,1 | tail -n10
 | gnuplot -p -e 'set boxwidth 0.5; plot "-" using 1:xtic(2) with boxes'
```

Data Wrangling for System Administration

- **Combining Tools**: Using `xargs` for bulk operations.

```
rustup toolchain list | grep nightly | grep -vE "nightly-x86" | sed 's/-x86.*//' | xargs rustup toolchain uninstall
```

Wrangling Binary Data

- **Binary Data**: Pipes are not just for text!

```
ffmpeg -loglevel panic -i /dev/video0 -frames 1 -f image2 -
 | convert - -colorspace gray -
 | gzip
 | ssh mymachine 'gzip -d | tee copy.jpg | env DISPLAY=:0 feh -'
```