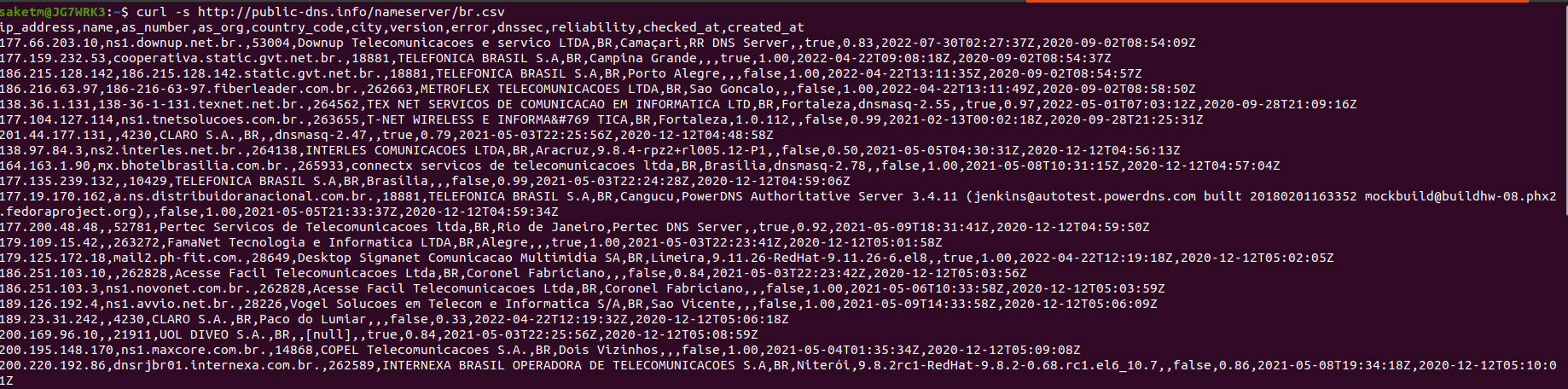
Given command

curl -s <http://public-dns.info/nameserver/br.csv> | cut -d, -f1 | shuf | tail -n 50 | xargs -i timeout 1 ping -c1 -w 1 {} | grep "time=" | awk '{print substr($7, 6, length($7)) " " substr($4, 1, length($4) -1)}' | sort -n | awk '{print $2 " " $1 "ms"}' | head -n 10

1: curl -s <http://public-dns.info/nameserver/br.csv>

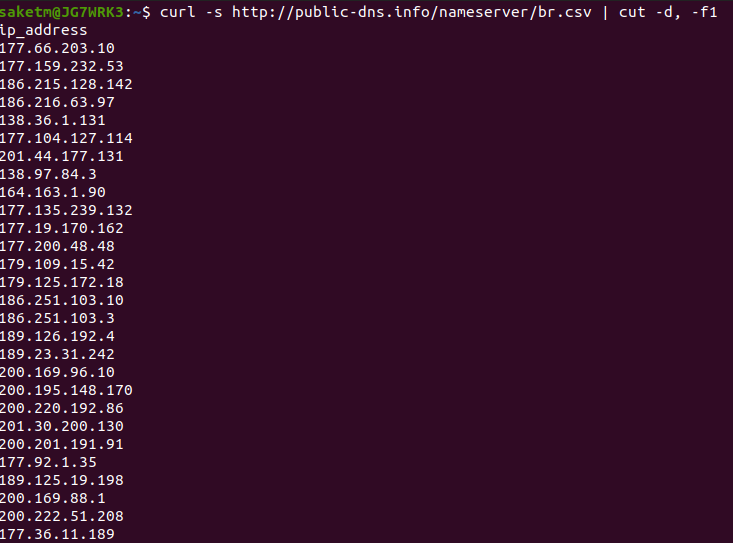
cURL (client URL) is one of the most used commands to automate the process of sending and receiving data to or from a server



2. cut -d, -f1

The cut command is a command-line utility that allows you to cut out sections of a specified file or piped data and print the result to standard output. The command cuts parts of a line by field, delimiter, byte position, and character.

This command splits the data by comma delimiter and outputs the first field of each line



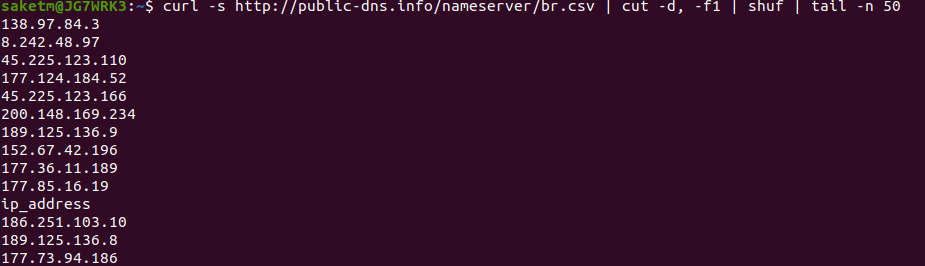
3. Shuf

The shuf command in Linux writes a random permutation of the input lines to standard output. It pseudo randomizes an input in the same way as the cards are shuffled.



4. tail -n 50

It is the complementary of [head](https://www.geeksforgeeks.org/head-command-linux-examples/) command.The tail command, as the name implies, print the last N number of data of the given input. By default it prints the last 10 lines of the specified files



5. curl -s <http://public-dns.info/nameserver/br.csv> | cut -d, -f1 | shuf | tail -n 50 | xargs -i timeout 1 ping -c1 -w 1 {}

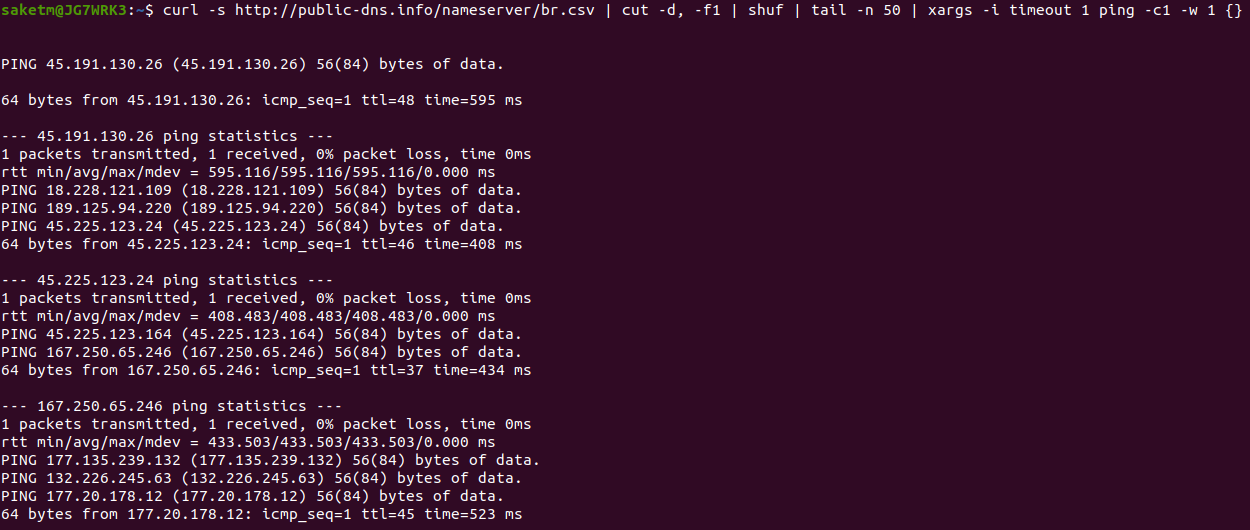
-> The xargs command builds and executes commands provided through the standard input. It takes the input and converts it into a command argument for another command.

-> timeout allows you to run a command with a time limit.

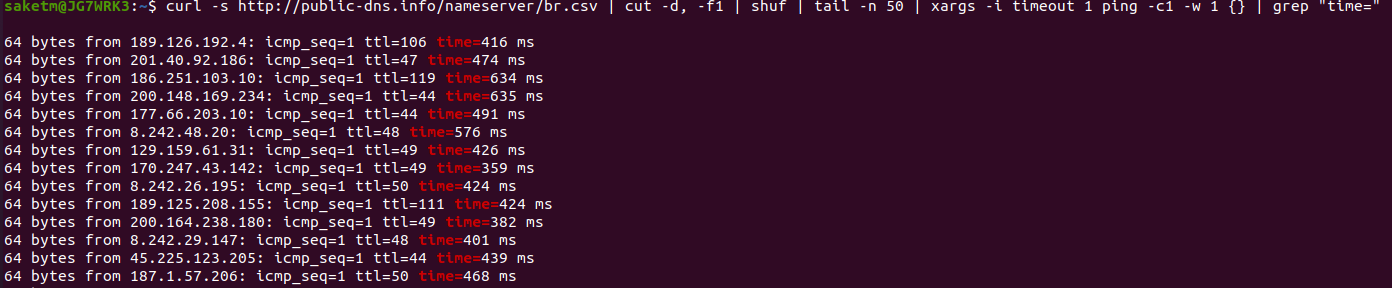
Xargs takes input and passes it to ping command with a timeout for 1 sec each

Ping -c1: to send 1 packet

-w 1: to stop pinging after 1sec



6. grep "time=" : search and print the specified string for each line



7. awk '{print substr($7, 6, length($7)) " " substr($4, 1, length($4) -1)}'

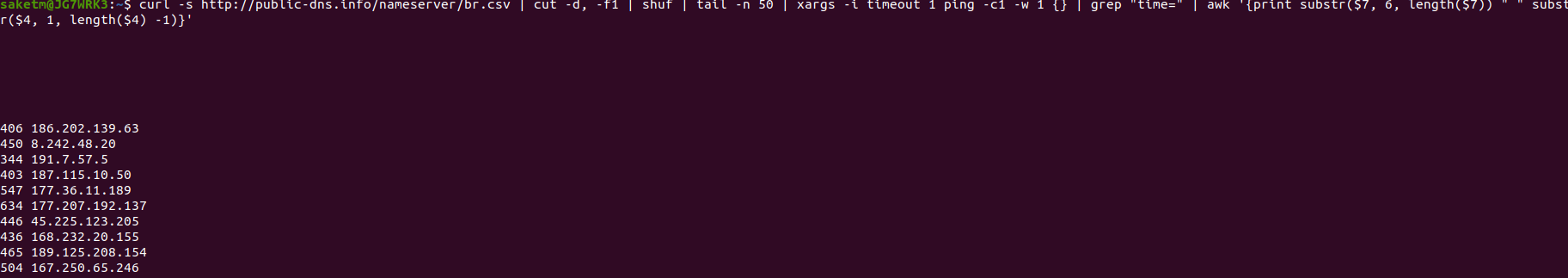
AWK Operations:

(a) Scans a file line by line

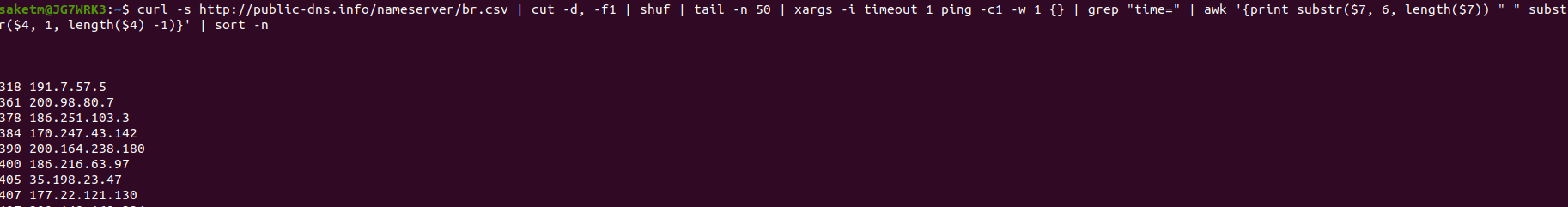
(b) Splits each input line into fields

(c) Compares input line/fields to pattern

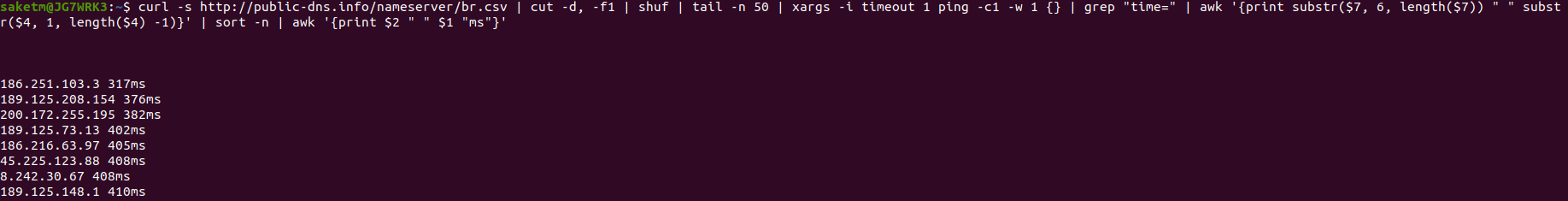
(d) Performs action(s) on matched lines



8. sort -n : To sort a file numerically used –n option.



9. curl -s http://public-dns.info/nameserver/br.csv | cut -d, -f1 | shuf | tail -n 50 | xargs -i timeout 1 ping -c1 -w 1 {} | grep "time=" | awk '{print substr($7, 6, length($7)) " " substr($4, 1, length($4) -1)}' | sort -n | awk '{print $2 " " $1 "ms"}'



10. head -n 10

The head command, as the name implies, print the top N number of data of the given input

-n num: Prints the first ‘num’ lines instead of first 10 lines

