The given command is

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

This command prints ip addresses present in <http://public-dns.info/nameserver/br.csv> with their respective pings in the order of their ping value.

We are piping one command to another so that output from one command is taken as input to next command.

Going through entire command step-by-step:

**curl -s** [**http://public-dns.info/nameserver/br.csv**](http://public-dns.info/nameserver/br.csv)**:**

This command loads the data from <http://public-dns.info/nameserver/br.csv> on to the terminal.In this case,the data from the server is given to the next command.

**cut -d, -f1:**

It cuts out a section from each line based on the delimeter provided.In this case, the delimeter is “,”.-d is used to set delimeter.-f1 is used to select first field from every line.

Here,the first field from each line is an IP address.

**shuf:**

It shuffles all the lines of the output.

**tail -n 50:**

It is used to take bottom 50 lines from the entire data.

**xargs -i timeout 1 ping -c1 -w1 {}:**

Xargs is used to read streams of data from input and execute commands on them.In this case xargs reads lines of input from previous 50 lines.Ping and timeout commands are executed on each line.

Timeout 1 sets the time limit 1s for ping command.

Ping is used to check network connectivity between host and server.In this case,network connectivity is checked for all the IP addresses generated before.

-c1 sets the number of packets sent to “1”.

-w1 makes the host to stop pinging after 1 second.

**grep “time=”:**

It is used to match the pattern “time=” to the lines produced above.

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

Awk is scripting language used to manipulate data.Here we use awk to print substrings from lines generated above.

Print substr($7,6,length($7)) returns “length($7)” number of characters from 7th field of each line starting from position 6.In other words,it returns ping values for all Ip addresses.

Similarly substr($4,1,length($4)-1) returns ip addresses from each line.

**sort -n:**

It sorts all the lines based on their ping values in increasing order.

**awk ‘{print $2 “ “ $1 “ms”}:**

It returns the lines in the following format.

IPaddress ping value ms

**head -n 10:**

It prints the first 10 lines from the input generated before.