Session – 8 : Discussion on Linux: Network Ports, Application the awk command
23-08-2019, 08:00 – 09:00 Hrs, RJN 302

[1] Network Ports

Read up about TCP / UDP ports from Wikipedia at the following URL:

https://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers

To know if a system is up and reachable, we can use the command “ping” which uses the Internet Control Message Protocol (ICMP) for communication. It is possible that some networks or machines disable ICMP. This means that if a machine replies to ping, it is up but if there is no reply, it does not mean the machine is down.

The system reserved ports that are for popular services are as follows:

Port	**Service**	**Remarks**
21	ftp	To transfer files in an unencrypted manner.
22	sctp	Secure shell connection for ssh login, scp, sftp
23	telnet	Terminal connection in an unencrypted manner
25	smtp	Email communication in an unencrypted manner
80	http	To view web pages in an unencrypted manner
443	https	Secure protocol to view web pages
465	smtps	Secure email communication
587	msa	Secure email communication after authentication
631	cupsd	Printer protocol in Linux

If a software that needs communication does not work here are few steps that should be checked:

- The server is reachable over network. Check using ping or ssh or any other service that is known to be available on the server.
- Check if the port is open using telnet <ip address> <port number>. You can quit this command using control+] and typing quit.
- If the port is not reachable, check on the server if the port is open locally.
- Check if the port is blocked by an anti-virus software in your client machine.

- There are possibilities that communication over that port is blocked in the network by the switches. Check that out by taking the client machine to different parts of the physical network and trying.

Licensed software check for license over the network using some port number that their vendor decides. This may change from version to version as well. If you are not able to run a licensed software, check if the issue is about port or about the number of licenses or an entry in the allowed hosts on the license server.

Tools such as "`nmap`" and websites such as "`mxtoolbox.com`" help in debugging some of these issues. There is a dedicated linux flavour "kali linux at `http://www.kali.org`" that is used by network security experts for audit and monitoring of machines.

[2] Application of awk in processing lots of data.

Spreadsheet software such as Microsoft Excel or LibreOffice Calc can load only 1 million rows in a sheet. That comes to 2^{20} = 1048576 rows. If you have more rows than that in a file, you will need tools that can work on them line by line or in an efficient manner.

The following example illustrates the case shown in the class.

[2.1] Creating a file with 2 million rows of data, each row containing two fields – the second is a square of the first. The shell script is as follows.

Create a file "m.sh" with the following content.

```
#!/bin/bash
i=0;
while [ $i -lt 2000000 ]
do
    echo $i, ",", $(( $i*$i ));
    i=$(( $i+1 ));
done
```

Execute this script using the command:

```
./m.sh > input.csv
```

[2.2] Check the input data for its content.

Check the number of lines in the file using the command:

```
wc -l input.csv
```

Check the first few lines of the file using the command:

```
head input.csv
```

Check the last few lines of the file using the command:

```
tail input.csv
```

[2.3] Try opening this file in LibreOffice Calc and see the error. Scroll to the bottom of the sheet to see how many rows of data could be read by the spread sheet software.

[2.4] The task is to add the two columns and create the third column. This can be done using the following awk script.

Create a file “m.awk” with the following content.

```
#!/usr/bin/gawk -f
BEGIN{
  FS=”,”;
  n=0;
}
{
  a=$1;
  b=$2;
  c=a+b;
  print a, “,”, b, “,”, c;
  n++;
{
END{
  print “Rows processed:”, n;
}
```

Execute the script using the command to store the output in a file:

```
cat input.csv | awk -f m.awk > output.csv
```

If you wish to check how long it took for awk to do this processing, time the process using the command:

```
time cat input.csv | awk -f m.awk > output.csv
```

[2.5] Follow the steps in 2.2 to check the content of the file output.csv.

Homework:

1. Write a code in awk that can list `n` number of Fibonacci numbers where `n` is a command line input for the script.