

Session-2

Nice and Renice

nice is used to start process with specific priority

renice is used to change the priority of existing process

1. each process has a nice value ranging from -20(highest Priority) to 19 (lowest priority)
2. by default new process starts with a nice value of 0.

Introduction to Nice and Renice

```
top - 09:47:28 up 4 min, 1 user, load average: 0.01, 0.01, 0.00
Tasks: 23 total, 1 running, 22 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.1 sy, 0.0 ni, 99.9 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 7884.0 total, 7273.5 free, 578.1 used, 260.8 buff/cache
MiB Swap: 2048.0 total, 2048.0 free, 0.0 used, 7306.0 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1	root	20	0	21760	12820	9496	S	0.0	0.2	0:01.10	systemd
2	root	20	0	2776	1924	1796	S	0.0	0.0	0:00.01	init-systemd(ub
7	root	20	0	2776	132	132	S	0.0	0.0	0:00.00	init
55	root	19	-1	50428	17328	16240	S	0.0	0.2	0:00.31	systemd-journal
98	root	20	0	23992	6016	4788	S	0.0	0.1	0:00.35	systemd-udev
110	systemd+	20	0	21452	11828	9636	S	0.0	0.1	0:00.26	systemd-resolve
111	systemd+	20	0	91020	8512	5608	S	0.0	0.1	0:00.18	systemd-timesyn
168	root	20	0	4236	2660	2432	S	0.0	0.0	0:00.01	cron
169	message+	20	0	9620	5236	4548	S	0.0	0.1	0:00.10	dbus-daemon
177	root	20	0	17976	8180	7156	S	0.0	0.1	0:00.33	systemd-logind
180	root	20	0	1755840	15996	9400	S	0.0	0.2	0:00.31	wsl-pro-service
195	root	20	0	3160	1188	1100	S	0.0	0.0	0:00.04	agetty
204	root	20	0	3116	1180	1096	S	0.0	0.0	0:00.03	agetty
210	syslog	20	0	22508	7268	4436	S	0.0	0.1	0:00.17	rsyslogd
220	root	20	0	136996	22472	13096	S	0.0	0.3	0:00.17	unattended-upgr
310	root	20	0	2780	208	80	S	0.0	0.0	0:00.00	SessionLeader
311	root	20	0	2780	212	80	S	0.0	0.0	0:00.01	Relay(312)
312	nikunj	20	0	6072	5312	3596	S	0.0	0.1	0:00.08	bash
313	root	20	0	6692	4480	3696	S	0.0	0.1	0:00.01	login
364	nikunj	20	0	20260	11348	9268	S	0.0	0.1	0:00.22	systemd
365	nikunj	20	0	21140	1720	0	S	0.0	0.0	0:00.00	(sd-pam)
378	nikunj	20	0	6072	5124	3484	S	0.0	0.1	0:00.05	bash
391	nikunj	20	0	9272	5280	3140	R	0.0	0.1	0:00.21	top

- higher the value means lower priority(the process is "nicer" to others)
- Lower nice value means Highers Priority (requires root access for negative values)
- you can start a particular process by using priority as

```
nice -n 10 myscript.sh
```

```
renice 10 -p <PID>
```

create script

```
#!/bin/bash
echo "Starting script with priority $(nice)"
```

```
sleep 100 # Simulate a process running
echo "Script completed."
```

start the script

```
bash script.sh
```

now open the wsl and run the above nice and renice commands and check the priority (PI) and nice (NI) using top command

CPU Bottlenecks

- High CPU Usage (nearly 100%)
- Slow Application response Time
- High Load Average(uptime , top , htop to monitor this)

How to Deal With IT?

Use ps to list the CPU Consuming processes

```
ps -eo pid,ppid,cmd,%mem,%cpu --sort=-%mem | head -10
```

also u can check using sysstat

```
sudo apt install mpstat # if its ask to install install using this command
mpstat -P ALL 1
```

Solution: Optimize the code, add caching, or Scale the servers HORIZINTALLY by adding More and More Servers

MEMORY BOTTLENECKS

- System slowdown Frequent Swapping
- High Ram usage (you can check this free -m or vmstat)
- Out of Memory (OOM error)

how to deal with it?

```
free -h # used to check free memory usage
```

```
ps -eo pid,ppid,cmd,%mem,%cpu --sort=-%mem | head -10 # find process consuming
the most of the memories
```

I/O Bottlenecks (Disk & Network)

- slow file read / write operation
- High Disk utilization
- High Network latency or dropped packets

How to Identify

```
# monitor disk I/O
iostat
```

```
iostat -dx
```

```
iostat -dx -1
```

```
#to find the input and output i/o consuming process
sudo iotop
```

A database query performing full table scan instead of using indexes can cause excessive disk I/O.

Solution: Optimize the queries , add indexes or caching

Introduction to SAR

(System Activity Report)

- it is command line tool that collects, reports, and saves system performance data.
- it is part of sysstat package
- it provide insight of CPU Usage, Memory Utilization, Network Activity...

How to Install?

```
sudo apt update sudo apt install sysstat -y
```

Enable and Start Data Collection

```
sudo systemctl enable sysstat
```

```
sudo systemctl start sysstat
```

lets start collecting data

```
sar -u 5 5
```

here -u : CPU usage Report

5 5 : Collects data every 5 seconds for 5 iteration

%user : CPU time Spent on user process

%system : CPU time Spent on Syatem / Kernel

%lOwait : Time Waiting for I/O operations

%idle : Available CPU time.

lets start collecting data

```
sar -S 5 5
```

here -S : Swap Usage STATISTICS

5 5 : Collects data every 5 seconds for 5 iteration

```
sar -d 5 5
```

here -d : disk Memory

5 5 : Collects data every 5 seconds for 5 iteration

PROMETHEUS

it is open source monitoring system

pull based model to collect system metrics from systems

How to Install ?

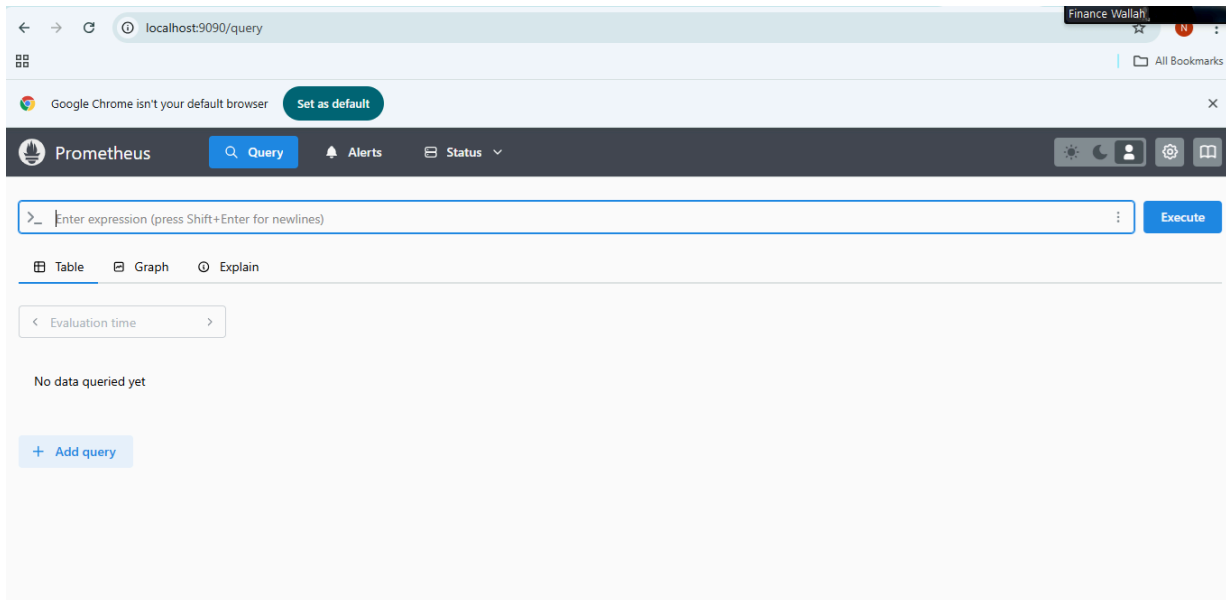
download from : <https://prometheus.io/download/>

Extract it to the folder

Start the Prometheus

```
./prometheus --config.file=prometheus.yml
```

goto the browser and open: localhost:9090



GRAFANA

- it is an interactive tool for Data Visualization

Link: <https://grafana.com/grafana/download>

```
sudo apt-get install -y adduser libfontconfig1 musl
wget https://dl.grafana.com/enterprise/release/grafana-
enterprise_11.5.2_amd64.deb
sudo dpkg -i grafana-enterprise_11.5.2_amd64.deb
```

once it is installed

```
sudo systemctl daemon-reload
sudo systemctl start grafana-server
sudo systemctl status grafana-server
sudo systemctl enable grafana-server.service
```

- open the Grafna UI-> http:localhost:3000
- LOGIN: admin /admin
- Configure DataSource: Configuration-> DataSource
- Select: Prometheus and enter the URL: <http://localhost:9090>
- Save and TEST

