

## **Adjust**

# **Network Monitoring Engineer Take Home Task**

## **Question 1**

-Script that writes the numbers 1-10 in random order. Each number appears once.  
numbers2.sh file uploaded.

### **Build Instructions**

- the script is a bash script.
- script created using text editor vim

### **Usage**

- runs on a UNIX system with bash shell installed.

### **Script Description**

- Script that prints the numbers 1-10 in random order.
- Uses the awk command which selects data based on the text provided.
- rand () function generates random numbers and the srand () function sets the initial point for generating the pseudo-random numbers.
- script compares all previously generated numbers which are stored in an array called prev and prints any number between 1-10 without repeating previously printed numbers.

```
gmapuranga@testsysadmin2:~$ bash
gmapuranga@testsysadmin2:~$ ls -l
total 12
-rwxr--r-- 1 gmapuranga gmapuranga 324 Jun 14 18:00 numbers2.sh
-rwxr--r-- 1 gmapuranga gmapuranga 207 Jun 14 17:28 randomnumber.sh
-rw-rw-r-- 1 gmapuranga gmapuranga 11 Jun 14 17:47 randomnumbers.file
gmapuranga@testsysadmin2:~$ cat numbers2.sh

#!/bin/bash
#This script generates numbers from 1-to 10 in random order.
#Author: Gwendolyn Mapuranga
#Date:14 June 2022

awk -v loop=10 -v range=10 'BEGIN{
  srand()
  do {
    numb = 1 + int(rand() * range)
    if (!(numb in prev)) {
      print numb
      prev[numb] = 1
      count++
    }
  } while (count<loop)
}'
gmapuranga@testsysadmin2:~$ ./numbers2.sh
2
9
10
4
8
1
3
7
6
5
gmapuranga@testsysadmin2:~$
```

## Limitations and Bugs

- The script will only print numbers up to 10.
- No bugs.

## Question 2

Important metrics to monitor on the server are below.

CPU utilization  
Memory utilization  
Disk utilization  
Network utilization

### How to monitor the metrics

Assuming the server is a Linux server. Following commands can be used to monitor metrics on the server.

Memory and CPU utilization

- **top** – displays a list of all processes currently running. It will give a readout of users, tasks, CPU load, and memory usage. To command produces an ordered list of running

processes selected by user-specified criteria and updates it periodically. By default, ordering is by CPU usage, and it shows processes that consume maximum CPU.

```
top - 19:01:03 up 1:40, 1 user, load average: 0.08, 0.02, 0.01
Tasks: 137 total, 1 running, 136 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.2 us, 0.0 sy, 0.0 ni, 99.8 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 7953.8 total, 6116.4 free, 319.5 used, 1517.8 buff/cache
MiB Swap: 0.0 total, 0.0 free, 0.0 used, 7354.6 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
890	root	20	0	401680	29224	10504	S	0.3	0.4	0:09.38	python3
986	omsagent	20	0	316240	63536	12080	S	0.3	0.8	0:16.58	omsagent
1	root	20	0	170632	13576	8400	S	0.0	0.2	0:04.69	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_gp
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_par_gp
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/0:0H-kblockd
9	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	mm_percpu_wq
10	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcu_tasks_rude_
11	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcu_tasks_trace
12	root	20	0	0	0	0	S	0.0	0.0	0:00.18	ksoftirqd/0
13	root	20	0	0	0	0	I	0.0	0.0	0:00.70	rcu_sched
14	root	rt	0	0	0	0	S	0.0	0.0	0:00.03	migration/0
15	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/0
16	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/1
17	root	rt	0	0	0	0	S	0.0	0.0	0:00.52	migration/1
18	root	20	0	0	0	0	S	0.0	0.0	0:00.35	ksoftirqd/1
20	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/1:0H-events_highpri
21	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kdevtmpfs
22	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	netns
23	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	inet_frag_wq
24	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kauditd
26	root	20	0	0	0	0	S	0.0	0.0	0:00.00	khungtaskd
27	root	20	0	0	0	0	S	0.0	0.0	0:00.00	oom_reaper
28	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	writeback
29	root	20	0	0	0	0	S	0.0	0.0	0:00.22	kcompactd0
30	root	25	5	0	0	0	S	0.0	0.0	0:00.00	ksmd
31	root	39	19	0	0	0	S	0.0	0.0	0:00.13	khugepaged
77	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kintegrityd
78	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kblockd
79	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	blkcg_punt_bio
80	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	tpm_dev_wq
81	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	ata_sff
82	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	md
83	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	edac-poller
84	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	hv_vmbus_con
85	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	hv_pri_chan
86	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	hv_sub_chan
87	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	devfreq_wq
88	root	-51	0	0	0	0	S	0.0	0.0	0:00.00	watchdogd
90	root	0	-20	0	0	0	I	0.0	0.0	0:00.09	kworker/1:1H-kblockd
92	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kswapd0
93	root	20	0	0	0	0	S	0.0	0.0	0:00.00	ecryptfs-kthrea
95	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kthrotld
96	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	nfit
98	root	20	0	0	0	0	S	0.0	0.0	0:00.00	scsi_ah_0
99	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	nvme-wq
100	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	scsi_tmf_0
101	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	scsi_ah_1

- **vmstat**- also shows memory and CPU utilization without showing the process running.

```
root@testsysadmin2:/home/gmapuranga# free -m
              total        used        free      shared  buff/cache   available
Mem:           7953          343         6077           0         1533         7329
Swap:              0              0              0

root@testsysadmin2:/home/gmapuranga# vmstat
procs -----memory----- --swap-- -----io----- -system-- -----cpu-----
 r  b   swpd   free   buff  cache   si   so    bi   bo   in   cs us sy id wa st
  0  0       0 6223884 29120 1540916    0    0    78  123   34  148  1  1 98  0  0
root@testsysadmin2:/home/gmapuranga#
```

## Disk utilization

- **df** - a command displays the information of device name, total blocks, total disk space, used disk space, available disk space, and mount points on a file system.

```
gmapuranga@testsysadmin2:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        29G  6.6G   23G   23% /
devtmpfs         3.9G     0   3.9G    0% /dev
tmpfs            3.9G     0   3.9G    0% /dev/shm
tmpfs            796M  960K   795M    1% /run
tmpfs            5.0M     0   5.0M    0% /run/lock
tmpfs            3.9G     0   3.9G    0% /sys/fs/cgroup
/dev/loop0       62M   62M     0 100% /snap/core20/1494
/dev/loop2       68M   68M     0 100% /snap/lxd/22526
/dev/loop3       45M   45M     0 100% /snap/snapd/15904
/dev/loop4       68M   68M     0 100% /snap/lxd/22753
/dev/sda15       105M  5.2M   100M    5% /boot/efi
/dev/sdb1        16G   45M   15G    1% /mnt
/dev/loop6       47M   47M     0 100% /snap/snapd/16010
/dev/loop5       62M   62M     0 100% /snap/core20/1518
tmpfs            796M     0   796M    0% /run/user/1001
gmapuranga@testsysadmin2:~$ df
Filesystem      1K-blocks    Used Available Use% Mounted on
/dev/root       30309264 6919132  23373748   23% /
devtmpfs        4068768      0   4068768    0% /dev
tmpfs           4072336      0   4072336    0% /dev/shm
tmpfs           814468      960    813508    1% /run
tmpfs           5120        0      5120    0% /run/lock
tmpfs           4072336      0   4072336    0% /sys/fs/cgroup
/dev/loop0       63488   63488      0 100% /snap/core20/1494
/dev/loop2       69632   69632      0 100% /snap/lxd/22526
/dev/loop3       45824   45824      0 100% /snap/snapd/15904
/dev/loop4       69504   69504      0 100% /snap/lxd/22753
/dev/sda15       106858    5321    101537    5% /boot/efi
/dev/sdb1       16446332 45080  15546112    1% /mnt
/dev/loop6       48128   48128      0 100% /snap/snapd/16010
/dev/loop5       63488   63488      0 100% /snap/core20/1518
tmpfs           814464      0    814464    0% /run/user/1001
gmapuranga@testsysadmin2:~$ █
```

- **iostat** - report the disk read/write rates and counts for an interval continuously. It collects disk statistics, waits for the given amount of time, collects them again, and displays the difference.

```
root@testsysadmin2:/home/gmapuranga# iostat
Linux 5.13.0-1025-azure (testsysadmin2)      06/14/22      _x86_64_      (2 CPU)

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           0.87    0.16    0.62    0.20    0.00   98.15

Device            tps    kB_read/s    kB_wrtn/s    kB_dscd/s    kB_read    kB_wrtn    kB_dscd
loop0              0.04         0.38         0.00         0.00       2464         0         0
loop1              0.01         0.06         0.00         0.00        364         0         0
loop2              0.01         0.19         0.00         0.00       1230         0         0
loop3              0.17         6.59         0.00         0.00      42954         0         0
loop4              0.01         0.19         0.00         0.00       1246         0         0
loop5              0.02         0.13         0.00         0.00        872         0         0
loop6              0.09         3.08         0.00         0.00      20060         0         0
loop7              0.00         0.00         0.00         0.00         14         0         0
sda                13.43        165.09        194.48       3655.28     1076425     1268045     23833017
sdb                 0.20         2.20         50.93       5088.28      14333     332104     33176420

root@testsysadmin2:/home/gmapuranga#
```

- **iotop** - displaying real-time disk activity. It can list the processes that are performing I/O, along with the disk bandwidth being used.

## Network utilization

- **nload** - monitors network traffic. By reporting in traffic and out traffic. The disadvantage of nload is that it doesn't display traffic by PID or socket.

```
Device enP31892s1 (1/3):
=====
Incoming:

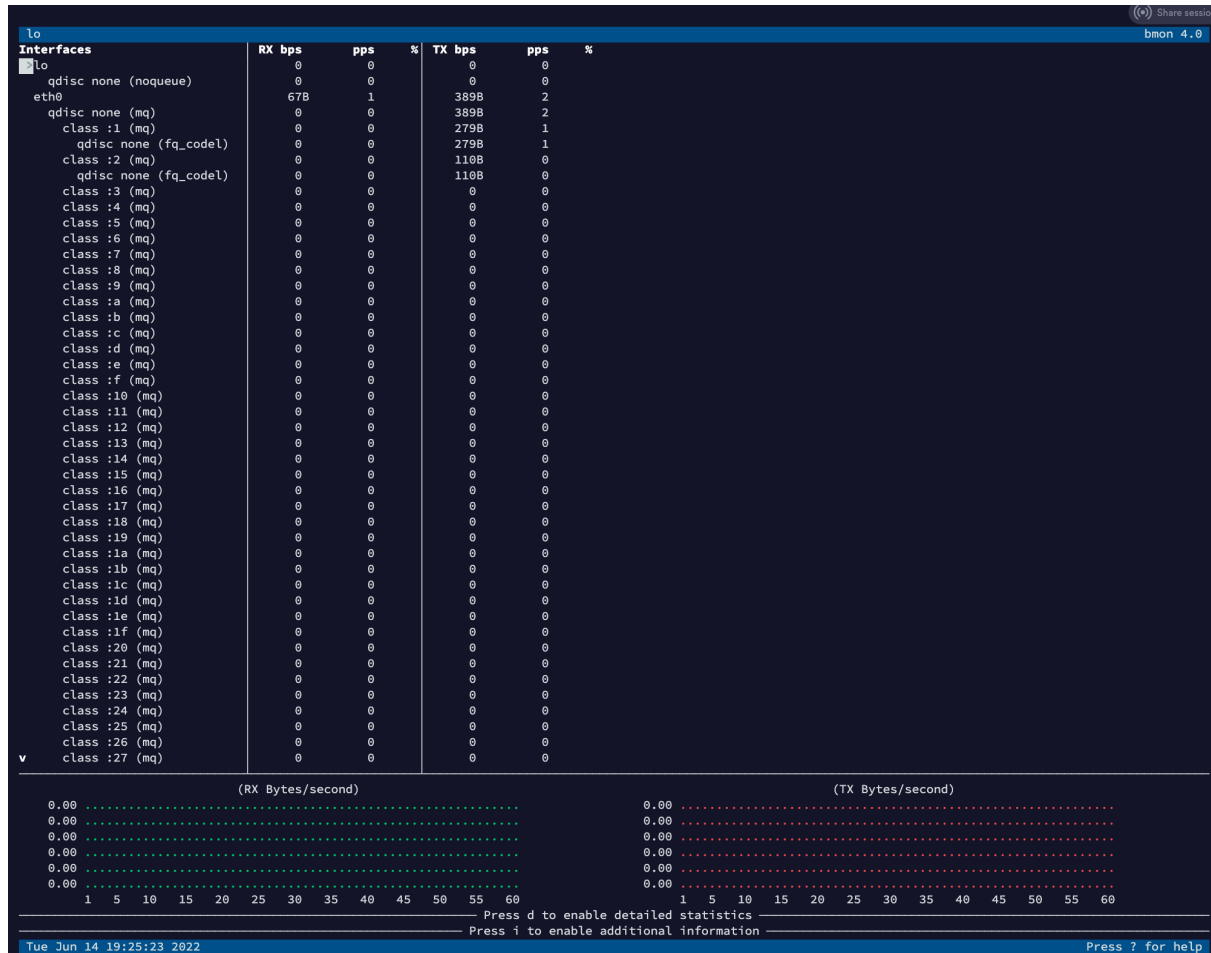
Outgoing:

Curr: 1.02 kBit/s
Avg: 3.70 kBit/s
Min: 1.02 kBit/s
Max: 38.67 kBit/s
Ttl: 102.74 MByte

Curr: 10.73 kBit/s
Avg: 23.11 kBit/s
Min: 6.56 kBit/s
Max: 66.70 kBit/s
Ttl: 38.52 MByte
```

Total DISK READ:			0.00 B/s	Total DISK WRITE:			53.79 K/s	
Current DISK READ:			0.00 B/s	Current DISK WRITE:			0.00 B/s	
TID	PRIO	USER	DISK READ	DISK WRITE	SWAPIN	IO>	COMMAND	
177	be/3	root	0.00 B/s	53.79 K/s	0.00 %	0.00 %	systemd-journald	
1	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	init	
2	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kthreadd]	
3	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[rcu_gp]	
4	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[rcu_par_gp]	
6	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kworker/0:0H-kblockd]	
9	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[mm_percpu_wq]	
10	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[rcu_tasks_rude_]	
11	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[rcu_tasks_trace]	
12	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[ksoftirqd/0]	
13	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[rcu_sched]	
14	rt/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[migration/0]	
15	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[cpuhp/0]	
16	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[cpuhp/1]	
17	rt/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[migration/1]	
18	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[ksoftirqd/1]	
20	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kworker/1:0H-events_highpri]	
21	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kdevtmpfs]	
22	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[netns]	
23	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[inet_frag_wq]	
24	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kauditd]	
26	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[khungtaskd]	
27	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[oom_reaper]	
28	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[writeback]	
29	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kcompactd0]	
30	be/5	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[ksmd]	
31	be/7	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[khugepaged]	
77	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kintegrityd]	
78	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kblockd]	
79	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[blkcg_punt_bio]	
80	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[tpm_dev_wq]	
81	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[ata_sff]	
82	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[md]	
83	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[edac-poller]	
84	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[hv_vmbus_con]	
85	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[hv_pri_chan]	
86	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[hv_sub_chan]	
87	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[devfreq_wq]	
88	rt/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[watchdogd]	
90	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kworker/1:1H-kblockd]	
92	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kswapd0]	
93	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[ecryptfs-kthrea]	
95	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kthrotld]	
96	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[nfit]	
98	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[scsi_eh_0]	
99	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[nvme-wq]	
100	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[scsi_tmf_0]	
101	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[scsi_eh_1]	
102	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[scsi_tmf_1]	

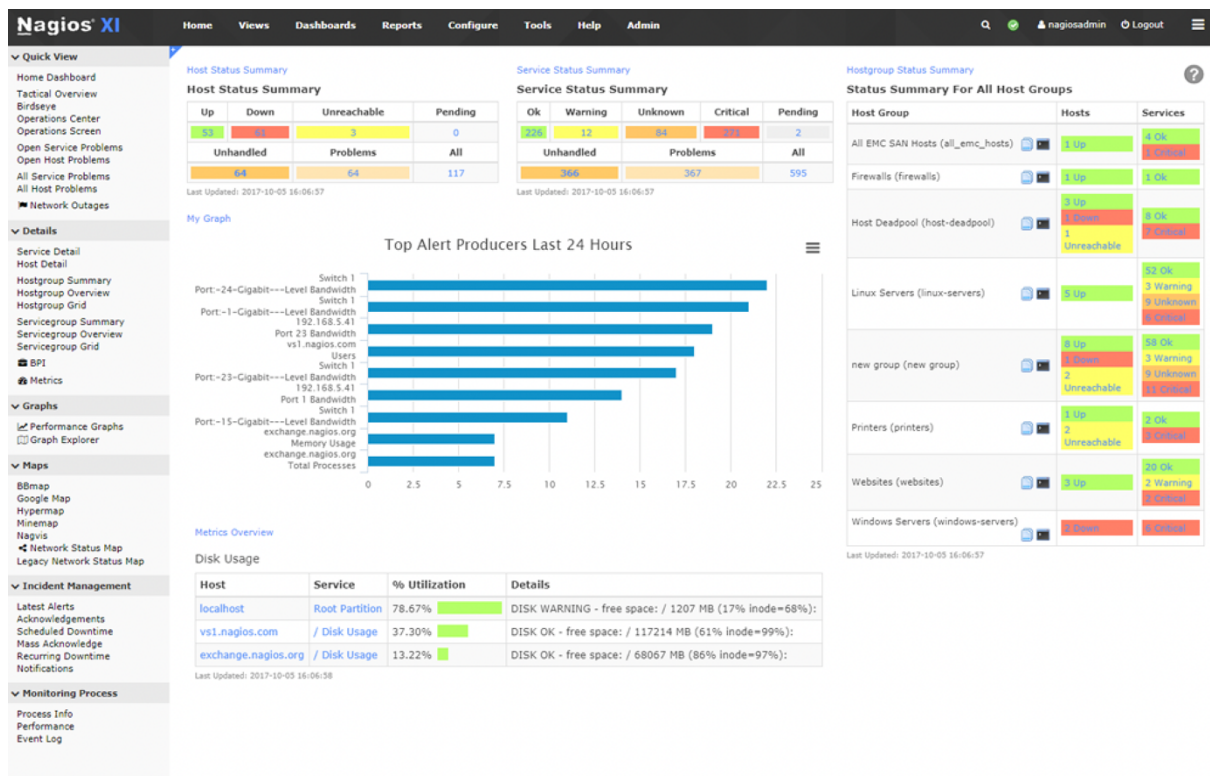
- **bmon**- monitors bandwidth utilization, along with keeping a running rate estimate. It breaks out usage by device, allowing you to track bandwidth across multiple network adapters.



### Challenges of monitoring this.

- Monitoring the server performance using Linux commands is too manual and does not provide automated alerts when the server exceeds set thresholds.
- Alternatively implementing monitoring tools such as Nagios and Zabbix will be able to set a threshold for utilization of server metrics. Once the server has exceeded a certain threshold alert will be sent via SMS or email ensuring a prompt response. Zabbix and Nagios offers dashboards such as 1 below easing the monitoring. All server metrics can be viewed from one screen.





- Monitoring metrics without any triggers or automated actions will result in system downtime. Using platforms such as AWS CloudWatch actions can be configured to scale vertically or horizontally when certain thresholds are exceeded during high peak levels.