

TASK 1

SHELL SCRIPT

```
#!/bin/bash
```

```
echo $(seq 10) | tr " " "\n" | shuf
```

Build Instruction:

copy the random.sh file to linux machine and make sure you have changed the permission to executable by running `chmod +x random.sh`

Usage:

To execute the code run `sh random.sh`, script will randomly display the numbers from 1 to 10 each time.

Description:

In simple word it will order the set of things in different ways. so here we are echoing the sequence of numbers from 1 to 10 and we are piping the result to translate command it will list down the numbers instead of single line, once this is done using `shuf` command we are displaying the numbers randomly.

TASK 2

SERVER DETAILS:

OS: Ubuntu 20.04

WORDPRESS: 5.5

MARIA-DB : 10.3.31-MariaDB-0ubuntu0.20.04.1

APACHE: 2.4.41

CPU DETAILS:

```
root@ip-172-31-32-159:~# lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:            Little Endian
Address sizes:         46 bits physical, 48 bits virtual
CPU(s):                16
On-line CPU(s) list:   0-15
Thread(s) per core:    2
Core(s) per socket:    8
Socket(s):             1
NUMA node(s):         1
Vendor ID:             GenuineIntel
CPU family:            6
Model:                85
Model name:            Intel(R) Xeon(R) Platinum 8259CL CPU @ 2.50GHz
Stepping:              7
CPU MHz:               2499.998
BogoMIPS:              4999.99
Hypervisor vendor:     KVM
Virtualization type:   full
L1d cache:             256 KiB
L1i cache:             256 KiB
L2 cache:              8 MiB
L3 cache:              35.8 MiB
NUMA node0 CPU(s):    0-15
```

MEMORY DETAILS:

```
root@ip-172-31-32-159:~# free -h
               total        used         free       shared    buff/cache   available
Mem:           62Gi        440Mi        60Gi         23Mi         718Mi        60Gi
Swap:           0B           0B           0B
```

root@ip-172-31-32-159:~# |

DISK DETAILS:

```

root@ip-172-31-32-159:~# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        97G   2.2G   95G   3% /
devtmpfs         32G    0    32G   0% /dev
tmpfs            32G    0    32G   0% /dev/shm
tmpfs            6.3G   900K   6.3G   1% /run
tmpfs            5.0M    0   5.0M   0% /run/lock
tmpfs            32G    0    32G   0% /sys/fs/cgroup
/dev/nvme1n1     1.9T   77M   1.8T   1% /data
/dev/loop1       34M   34M    0 100% /snap/amazon-ssm-agent/3552
/dev/loop0       56M   56M    0 100% /snap/core18/1997
/dev/loop2       33M   33M    0 100% /snap/snapd/11588
/dev/loop3       71M   71M    0 100% /snap/lxd/19647
tmpfs            6.3G    0   6.3G   0% /run/user/1000
root@ip-172-31-32-159:~# |

```

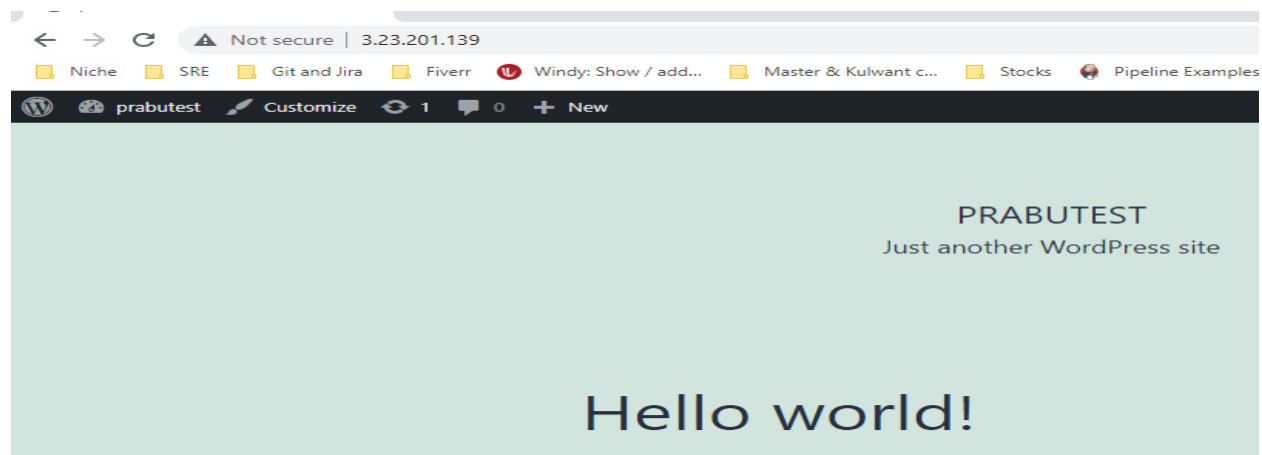
NIC DETAILS:

```

root@ip-172-31-32-159:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens5: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc mq state UP group default qlen 1000
    link/ether 0a:0e:7a:ac:24:12 brd ff:ff:ff:ff:ff:ff
    inet 172.31.32.159/20 brd 172.31.47.255 scope global dynamic ens5
        valid_lft 2519sec preferred_lft 2519sec
    inet6 fe80::80e:7aff:feac:2412/64 scope link
        valid_lft forever preferred_lft forever
root@ip-172-31-32-159:~# |

```

WEBSITE for load test



Server Load before starting the load test:

ALL CPU showing 0.0 due to no requests web server and Memory utilization is only 418M out of 64G everything looks fine before starting the load test

```
1 [ 0.0%] 5 [ 0.0%] 9 [ 0.0%] 13 [ 0.0%]
2 [ 0.0%] 6 [ 0.0%] 10 [ 0.0%] 14 [ 0.0%]
3 [ 0.0%] 7 [ 0.0%] 11 [ 0.0%] 15 [ 0.0%]
4 [ 0.0%] 8 [ 0.0%] 12 [ 0.0%] 16 [ 0.0%]
Mem[ 417M/62.0G] Tasks: 58, 124 thr; 1 running
Swp[ 0K/0K] Load average: 0.01 0.66 0.97
Uptime: 02:24:16
```

PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
24916	root	20	0	8784	4404	3128	R	0.7	0.0	0:00.04	htop
1	root	20	0	168M	12764	3424	S	0.0	0.0	0:05:47	/sbin/init
270	root	19	-1	59688	22684	21508	S	0.0	0.0	0:00.19	/lib/systemd/systemd-journald
307	root	20	0	8720	5152	3696	S	0.0	0.0	0:00.10	/lib/systemd/systemd-udev
453	root	RT	0	273M	17944	8204	S	0.0	0.0	0:00.03	/sbin/multipathd -d -s
454	root	RT	0	273M	17944	8204	S	0.0	0.0	0:00.00	/sbin/multipathd -d -s
455	root	RT	0	273M	17944	8204	S	0.0	0.0	0:00.00	/sbin/multipathd -d -s
456	root	RT	0	273M	17944	8204	S	0.0	0.0	0:00.18	/sbin/multipathd -d -s
457	root	RT	0	273M	17944	8204	S	0.0	0.0	0:00.00	/sbin/multipathd -d -s
458	root	RT	0	273M	17944	8204	S	0.0	0.0	0:00.00	/sbin/multipathd -d -s
452	root	RT	0	273M	17944	8204	S	0.0	0.0	0:00.36	/sbin/multipathd -d -s
528	systemd-t	20	0	90236	6128	5360	S	0.0	0.0	0:00.00	/lib/systemd/systemd-timesyncd
499	systemd-t	20	0	90236	6128	5360	S	0.0	0.0	0:00.06	/lib/systemd/systemd-timesyncd
554	systemd-n	20	0	26744	7460	6600	S	0.0	0.0	0:00.06	/lib/systemd/systemd-networkd
557	systemd-r	20	0	23904	12028	6104	S	0.0	0.0	0:00.07	/lib/systemd/systemd-resolved
593	root	20	0	231M	7404	6564	S	0.0	0.0	0:00.07	/usr/lib/accounts-service/accounts-daemon
702	root	20	0	231M	7404	6564	S	0.0	0.0	0:00.00	/usr/lib/accounts-service/accounts-daemon
592	root	20	0	231M	7404	6564	S	0.0	0.0	0:00.09	/usr/lib/accounts-service/accounts-daemon
595	root	20	0	2548	792	724	S	0.0	0.0	0:00.00	/usr/sbin/acpid
601	root	20	0	8544	2896	2620	S	0.0	0.0	0:00.01	/usr/sbin/cron -f
605	messagebu	20	0	7540	4600	3948	S	0.0	0.0	0:00.18	/usr/bin/dbus-daemon --system --address=systemd: --nofork --nopidfile --systemd-activation --syslog-only
625	root	20	0	81916	3884	3540	S	0.0	0.0	0:00.00	/usr/sbin/irqbalance --foreground
619	root	20	0	81916	3884	3540	S	0.0	0.0	0:00.23	/usr/sbin/irqbalance --foreground
621	root	20	0	23284	17088	10324	S	0.0	0.0	0:00.05	/usr/bin/python3 /usr/bin/networkd-dispatcher --run-startup-triggers
696	syslog	20	0	219M	4992	3932	S	0.0	0.0	0:00.01	/usr/sbin/rsyslogd -n -iNONE
697	syslog	20	0	219M	4992	3932	S	0.0	0.0	0:00.00	/usr/sbin/rsyslogd -n -iNONE
698	syslog	20	0	219M	4992	3932	S	0.0	0.0	0:00.01	/usr/sbin/rsyslogd -n -iNONE
622	syslog	20	0	219M	4992	3932	S	0.0	0.0	0:00.03	/usr/sbin/rsyslogd -n -iNONE
1019	root	20	0	1784M	23032	12820	S	0.0	0.0	0:00.06	/snap/amazon-ssm-agent/3552/amazon-ssm-agent
1020	root	20	0	1784M	23032	12820	S	0.0	0.0	0:00.00	/snap/amazon-ssm-agent/3552/amazon-ssm-agent
1021	root	20	0	1784M	23032	12820	S	0.0	0.0	0:00.00	/snap/amazon-ssm-agent/3552/amazon-ssm-agent
1022	root	20	0	1784M	23032	12820	S	0.0	0.0	0:00.00	/snap/amazon-ssm-agent/3552/amazon-ssm-agent
1023	root	20	0	1784M	23032	12820	S	0.0	0.0	0:00.00	/snap/amazon-ssm-agent/3552/amazon-ssm-agent
1064	root	20	0	1784M	23032	12820	S	0.0	0.0	0:00.00	/snap/amazon-ssm-agent/3552/amazon-ssm-agent
1066	root	20	0	1784M	23032	12820	S	0.0	0.0	0:00.02	/snap/amazon-ssm-agent/3552/amazon-ssm-agent

I am going to use the Apache benchmark tool for load testing, using this we can figure out the exact load when sending 20000 concurrent request to the server.

NOTE: we can setup only 20000 Concurrent requests in AB tool due to this limitation I am not going to setup 25000 request per second which is mentioned in the TASK Document.

We can do the monitoring by using so many open-source tools like nagios, Zabbix open source but more understanding I am going to do this manually by checking all the possible metrics using Linux tools. So, this will give us brief understanding of what happening in server and how we can troubleshoot the issue in future.

Screenshot of benchmark testing

```
root@ip-172-31-24-153:~# ab -r -n 1000000 -c 20000 -g out.data http://3.23.201.139/
This is ApacheBench, Version 2.3 <$Revision: 1843412 $>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking 3.23.201.139 (be patient)
```

SERVER METRICS:

UPTIME:

```
root@ip-172-31-32-159:~# uptime
07:54:17 up 4:10, 2 users, load average: 1.66, 1.25, 0.53
root@ip-172-31-32-159:~# uptime
07:54:22 up 4:10, 2 users, load average: 1.68, 1.27, 0.54
root@ip-172-31-32-159:~# uptime
07:54:29 up 4:10, 2 users, load average: 1.97, 1.34, 0.57
root@ip-172-31-32-159:~# uptime
07:54:31 up 4:10, 2 users, load average: 1.97, 1.34, 0.57
root@ip-172-31-32-159:~# uptime
07:54:35 up 4:10, 2 users, load average: 2.29, 1.42, 0.60
root@ip-172-31-32-159:~# uptime
07:54:43 up 4:10, 2 users, load average: 2.57, 1.50, 0.64
root@ip-172-31-32-159:~#
```

```
root@ip-172-31-32-159:~# uptime
08:23:33 up 4:39, 3 users, load average: 28.28, 8.91, 4.11
root@ip-172-31-32-159:~# http
```

This is the quick way to view the load average and processes want to run this will give high level idea of resource load in server

This value counted based on 1, 5, 15 mins interval, so we can assume something is cooking up in CPU.

VMSTAT

```
root@ip-172-31-32-159:~# vmstat 1
procs -----memory----- --swap-- -----io----- -system-- -----cpu-----
r  b   swpd   free   buff  cache   si   so    bi   bo    in   cs  us  sy  id  wa  st
3  0     0 62551780 45528 1326224    0    0     3   77   77  42 116  3  0 96  0  0
5  0     0 62550808 45528 1326248    0    0     0    0 25490 30479 18  2 80  0  0
2  0     0 62550224 45528 1326268    0    0     0    0 24665 28634 16  2 82  0  0
1  0     0 62550304 45528 1326288    0    0     0    0 21652 24731 14  2 84  0  0
6  0     0 62550936 45528 1326308    0    0     0    0 21367 22758 13  1 86  0  0
3  0     0 62550304 45528 1326324    0    0     0    0 25618 29700 17  2 81  0  0
1  0     0 62550312 45528 1326352    0    0     0    0 26597 31307 18  2 80  0  0
1  0     0 62550520 45528 1326372    0    0     0    0 30663 37616 22  3 75  0  0
3  0     0 62550560 45528 1326392    0    0     0    0 27777 32497 19  3 79  0  0
3  0     0 62550912 45528 1326420    0    0     0    0 28428 33343 19  3 78  0  0
```

I have printed the vmstat values for every 1 second, there is nothing much happening on the memory side, you can see free, buff, cache everything, SI(swap in) SO(Swap out) everything looks good, but CPU value, cs,us,sy,id,wa,st looks benchmark process is stressing the CPU

PIDSTAT 1:

This command is similar to top command, but it will print a rolling summary, instead of clearing screen, this will help us to identify which process is taking more cpu, in below example we can see apache2 taking more cpu resources

```
root@ip-172-31-32-159:~# pidstat 1
```

Linux 5.11.0-1020-aws (ip-172-31-32-159) 10/27/21 _x86_64_ (16 CPU)									
08:26:57	UID	PID	%usr	%system	%guest	%wait	%CPU	CPU	Command
08:26:58	0	86	0.00	0.98	0.00	0.00	0.98	12	ksoftirqd/12
08:26:58	113	24769	28.43	36.27	0.00	0.00	64.71	8	mysqld
08:26:58	33	25876	10.78	0.98	0.00	0.00	11.76	0	apache2
08:26:58	33	25887	6.86	1.96	0.00	0.00	8.82	12	apache2
08:26:58	33	25902	13.73	1.96	0.00	3.92	15.69	15	apache2
08:26:58	33	26795	6.86	0.00	0.00	0.98	6.86	13	apache2
08:26:58	33	26802	9.80	0.98	0.00	0.00	10.78	10	apache2
08:26:58	33	26805	7.84	0.98	0.00	0.00	8.82	8	apache2
08:26:58	33	26810	7.84	0.98	0.00	0.98	8.82	9	apache2
08:26:58	33	28959	7.84	0.98	0.00	2.94	8.82	2	apache2
08:26:58	33	28960	8.82	0.98	0.00	0.00	9.80	12	apache2
08:26:58	33	28961	3.92	0.00	0.00	0.00	3.92	8	apache2
08:26:58	33	28962	12.75	0.00	0.00	0.98	12.75	5	apache2
08:26:58	33	28964	14.71	1.96	0.00	0.98	16.67	9	apache2
08:26:58	33	28966	1.96	0.00	0.00	0.00	1.96	8	apache2
08:26:58	33	28967	1.96	0.00	0.00	0.00	1.96	7	apache2
08:26:58	33	28970	15.69	1.96	0.00	0.98	17.65	0	apache2
08:26:58	33	28976	11.76	1.96	0.00	2.94	13.73	4	apache2
08:26:58	33	28977	2.94	0.00	0.00	0.00	2.94	1	apache2
08:26:58	33	28978	4.90	0.00	0.00	0.00	4.90	1	apache2

IOSTAT -XZ 1

root@ip-172-31-32-159:~# iostat -xz 1

Linux 5.11.0-1020-aws (ip-172-31-32-159) 10/27/21 _x86_64_ (16 CPU)

avg-cpu:	%user	%nice	%system	%iowait	%steal	%idle														
	4.91	0.00	0.66	0.01	0.00	94.42														
Device	r/s	rkB/s	rrqm/s	%rrqm	r_await	rareq-sz	w/s	wkB/s	wrqm/s	%wrqm	w_await	wareq-sz	d/s	dkB/s	drqm/s	%drqm	d_await	dareq-sz	aqu-sz	%util
loop0	0.02	0.13	0.00	0.00	0.21	8.30	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
loop1	0.03	0.75	0.00	0.00	0.11	27.92	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01
loop2	0.00	0.09	0.00	0.00	0.62	19.38	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
loop3	0.00	0.06	0.00	0.00	0.45	14.99	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
loop4	0.03	1.05	0.00	0.00	0.03	32.45	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01
loop5	0.00	0.02	0.00	0.00	0.00	6.32	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
loop6	0.01	0.14	0.00	0.00	0.02	9.16	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
loop7	0.02	0.53	0.00	0.00	0.03	28.62	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
loop8	0.00	0.06	0.00	0.00	0.01	15.38	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
nvme0n1	0.64	33.79	0.17	20.67	1.16	53.04	1.13	60.85	1.38	54.86	3.84	53.62	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.17
nvme1n1	0.02	0.25	0.00	0.00	0.47	10.85	4.01	1025.45	0.02	0.39	2.70	255.88	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.22
avg-cpu:	%user	%nice	%system	%iowait	%steal	%idle														
	90.18	0.00	9.19	0.00	0.00	0.63														
Device	r/s	rkB/s	rrqm/s	%rrqm	r_await	rareq-sz	w/s	wkB/s	wrqm/s	%wrqm	w_await	wareq-sz	d/s	dkB/s	drqm/s	%drqm	d_await	dareq-sz	aqu-sz	%util
avg-cpu:	81.08	0.00	8.36	0.00	0.00	10.56														
Device	r/s	rkB/s	rrqm/s	%rrqm	r_await	rareq-sz	w/s	wkB/s	wrqm/s	%wrqm	w_await	wareq-sz	d/s	dkB/s	drqm/s	%drqm	d_await	dareq-sz	aqu-sz	%util
avg-cpu:	%user	%nice	%system	%iowait	%steal	%idle														
	88.17	0.00	9.39	0.00	0.00	2.44														
Device	r/s	rkB/s	rrqm/s	%rrqm	r_await	rareq-sz	w/s	wkB/s	wrqm/s	%wrqm	w_await	wareq-sz	d/s	dkB/s	drqm/s	%drqm	d_await	dareq-sz	aqu-sz	%util
nvme0n1	0.00	0.00	0.00	0.00	0.00	0.00	2.00	24.00	4.00	66.67	0.50	12.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.80
avg-cpu:	%user	%nice	%system	%iowait	%steal	%idle														
	88.97	0.00	10.09	0.00	0.00	0.94														
Device	r/s	rkB/s	rrqm/s	%rrqm	r_await	rareq-sz	w/s	wkB/s	wrqm/s	%wrqm	w_await	wareq-sz	d/s	dkB/s	drqm/s	%drqm	d_await	dareq-sz	aqu-sz	%util

ac

This tool will help us to the disks both workload and performance, %util value will tell us how much disk is busy, more than 60% typically lead to poor performance but here we have only 2% hence we can conclude that there is not much load in the disk.

Free -h

```
root@ip-172-31-32-159:~# free -h
              total        used        free      shared  buff/cache   available
Mem:           62Gi        1.1Gi        59Gi         17Mi        1.4Gi        60Gi
Swap:           0B           0B           0B
```

Free output looks fine there is no much use in memory side its just only 1.1g and we have to consider buff/cache its almost 1.4Gi higher the value it will affect disk block device IO

sar -n DEV 1

```
root@ip-172-31-32-159:~# sar -n DEV 1
Linux 5.11.0-1020-aws (ip-172-31-32-159)      10/27/21      _x86_64_      (16 CPU)

08:47:54      IFACE  rxpck/s  txpck/s  rxkB/s  txkB/s  rxcmp/s  txcmp/s  rxmcst/s   %ifutil
08:47:55      lo      0.00    0.00    0.00    0.00    0.00    0.00    0.00      0.00
08:47:55      ens5   9874.00  9718.00  866.32 10066.00  0.00    0.00    0.00      0.00

08:47:55      IFACE  rxpck/s  txpck/s  rxkB/s  txkB/s  rxcmp/s  txcmp/s  rxmcst/s   %ifutil
08:47:56      lo      0.00    0.00    0.00    0.00    0.00    0.00    0.00      0.00
08:47:56      ens5  10052.00 10541.00  824.50 10916.60  0.00    0.00    0.00      0.00

08:47:56      IFACE  rxpck/s  txpck/s  rxkB/s  txkB/s  rxcmp/s  txcmp/s  rxmcst/s   %ifutil
08:47:57      lo      0.00    0.00    0.00    0.00    0.00    0.00    0.00      0.00
08:47:57      ens5   9157.00  9505.00  758.24  9643.81  0.00    0.00    0.00      0.00

08:47:57      IFACE  rxpck/s  txpck/s  rxkB/s  txkB/s  rxcmp/s  txcmp/s  rxmcst/s   %ifutil
08:47:58      lo      0.00    0.00    0.00    0.00    0.00    0.00    0.00      0.00
08:47:58      ens5   9035.00  9767.00  717.41 10160.11  0.00    0.00    0.00      0.00

08:47:58      IFACE  rxpck/s  txpck/s  rxkB/s  txkB/s  rxcmp/s  txcmp/s  rxmcst/s   %ifutil
08:47:59      lo      0.00    0.00    0.00    0.00    0.00    0.00    0.00      0.00
08:47:59      ens5  10682.00 10771.00  913.12 10892.07  0.00    0.00    0.00      0.00
```

This will check network interface throughput, rxkb txkb will measure workload

sar -n TCP, ETCP 1

```
root@ip-172-31-32-159:~# sar -n TCP,ETCP 1
Linux 5.11.0-1020-aws (ip-172-31-32-159)      10/27/21      _x86_64_      (16 CPU)

08:52:20   active/s  passive/s    iseg/s    oseg/s
08:52:21       0.00    234.00    3851.00    2763.00

08:52:20   atmpth/s  estres/s  retrans/s  isegerr/s  orsts/s
08:52:21       0.00       0.00    2089.00       0.00       0.00

08:52:21   active/s  passive/s    iseg/s    oseg/s
08:52:22       0.00    186.00    2927.00    2286.00

08:52:21   atmpth/s  estres/s  retrans/s  isegerr/s  orsts/s
08:52:22       0.00       0.00    1973.00       0.00       0.00

08:52:22   active/s  passive/s    iseg/s    oseg/s
08:52:23       0.00    223.00    3078.00    2656.00

08:52:22   atmpth/s  estres/s  retrans/s  isegerr/s  orsts/s
08:52:23       0.00       0.00    2014.00       0.00       0.00

08:52:23   active/s  passive/s    iseg/s    oseg/s
08:52:24       0.00    202.00    4285.00    2457.00

08:52:23   atmpth/s  estres/s  retrans/s  isegerr/s  orsts/s
08:52:24       0.00       0.00    1807.00       0.00       0.00
^C

Average:   active/s  passive/s    iseg/s    oseg/s
Average:       0.00    211.25    3535.25    2540.50
```

This will help us to find out the active and passive TCP connections in server which will give rough measure of server load.

HTOP:

Below is output of the htop command when the load was high on CPU side


```
[|||||100.0%] 5 [|||||100.0%] 9 [|||||100.0%] 13 [|||||100.0%]
[|||||99.4%] 6 [|||||100.0%] 10 [|||||100.0%] 14 [|||||100.0%]
[|||||100.0%] 7 [|||||100.0%] 11 [|||||100.0%] 15 [|||||99.3%]
[|||||100.0%] 8 [|||||100.0%] 12 [|||||100.0%] 16 [|||||99.3%]
em[|||||
wp[|||||
1.07G/62.0G Tasks: 203, 285 thr: 16 running
OK/OK Load average: 11.61 8.62 4.79
Uptime: 04:42:37

PID USER PR1 NI VIRT RES SHR S CPU% MEM% TIME+ Command
4769 mysql 20 0 8881M 99M 20096 S 78.8 0.2 6:02.35 /usr/sbin/mysqld
9024 www-data 20 0 192M 32220 24404 S 28.2 0.0 0:12.63 /usr/sbin/apache2 -k start
9091 www-data 20 0 192M 32240 24424 R 28.2 0.0 0:13.14 /usr/sbin/apache2 -k start
9057 www-data 20 0 192M 32236 24404 R 27.6 0.0 0:14.45 /usr/sbin/apache2 -k start
9012 www-data 20 0 192M 32220 24404 R 23.6 0.0 0:11.91 /usr/sbin/apache2 -k start
9092 www-data 20 0 192M 32136 24320 S 23.6 0.0 0:14.80 /usr/sbin/apache2 -k start
9323 www-data 20 0 192M 32156 24340 R 22.3 0.0 0:03.44 /usr/sbin/apache2 -k start
8969 www-data 20 0 192M 32216 24400 R 21.0 0.0 0:13.99 /usr/sbin/apache2 -k start
8968 www-data 20 0 192M 32216 24400 R 21.0 0.0 0:11.80 /usr/sbin/apache2 -k start
9047 www-data 20 0 192M 32156 24340 S 21.0 0.0 0:14.14 /usr/sbin/apache2 -k start
8991 www-data 20 0 192M 32216 24400 R 20.4 0.0 0:15.74 /usr/sbin/apache2 -k start
9049 www-data 20 0 192M 32192 24376 R 20.4 0.0 0:14.33 /usr/sbin/apache2 -k start
9044 www-data 20 0 192M 32156 24340 S 19.0 0.0 0:14.82 /usr/sbin/apache2 -k start
9318 www-data 20 0 192M 32040 24224 R 19.0 0.0 0:03.54 /usr/sbin/apache2 -k start
5902 www-data 20 0 192M 32208 24380 R 19.0 0.0 0:40.23 /usr/sbin/apache2 -k start
9327 www-data 20 0 192M 32136 24320 S 19.0 0.0 0:03.07 /usr/sbin/apache2 -k start
8970 www-data 20 0 192M 32224 24400 S 19.0 0.0 0:13.85 /usr/sbin/apache2 -k start
5809 www-data 20 0 192M 32220 24400 S 18.4 0.0 0:35.67 /usr/sbin/apache2 -k start
9004 www-data 20 0 192M 32220 24404 S 17.7 0.0 0:14.77 /usr/sbin/apache2 -k start
8980 www-data 20 0 192M 32216 24400 R 17.7 0.0 0:13.71 /usr/sbin/apache2 -k start
9022 www-data 20 0 192M 32220 24404 S 17.1 0.0 0:13.83 /usr/sbin/apache2 -k start
6810 www-data 20 0 192M 32224 24404 R 17.1 0.0 0:34.12 /usr/sbin/apache2 -k start
8965 www-data 20 0 192M 32068 24236 S 16.4 0.0 0:13.67 /usr/sbin/apache2 -k start
9088 www-data 20 0 192M 32240 24424 R 16.4 0.0 0:12.54 /usr/sbin/apache2 -k start
9083 www-data 20 0 192M 32188 24372 S 16.4 0.0 0:13.41 /usr/sbin/apache2 -k start
9069 www-data 20 0 192M 32176 24360 R 16.4 0.0 0:14.81 /usr/sbin/apache2 -k start
9093 www-data 20 0 192M 32184 24368 S 16.4 0.0 0:15.51 /usr/sbin/apache2 -k start
9074 www-data 20 0 192M 32240 24424 S 16.4 0.0 0:14.16 /usr/sbin/apache2 -k start
```

FOR SSL PROXY We can use nagios for monitoring the below metrics

- 1.Inbound Traffic
2. Outbound Traffic
3. Open Connections
4. New Connections per second
5. Closed Connections per second

Challenges:

1. Assuming we have proper monitoring tools in place, we can use above commands to figure out the exact problems happening in the server
2. For SSL PROXY we need proper setup of monitoring se we can figure out the issue well in advance
3. Implementing proper SSL handshake failure count script, this will help us to know whether we have any failures between application and Proxy