1. **App or DB server accessible only from Botree network IP Addresses**
    a. **Tata IP: 115.160.248.91**
    b. **Reliance: 115.249.202.98**

2. **History command with timestamp, need to execute this command after create the new user name**

    $ echo 'HISTTIMEFORMAT="%F-- %T--"' >> ~/.bashrc

3. **CHANGE Time to IST**

    $ timedatectl set-timezone Asia/Kolkata

4. **Disable IPv6**
    a. **# vi /etc/ssh/sshd_config**

        Add this line

        AddressFamily inet

    b. # systemctl restart sshd
    c. # vi /etc/sysctl.conf

        Add this line

```
net.ipv6.conf.all.disable_ipv6 = 1
net.ipv6.conf.default.disable_ipv6 = 1
```

    d. # sysctl -p

    e. If you get problems with starting postfix after disabling IPv6, edit **/etc/postfix/main.cf** file;

```
# vi /etc/postfix/main.cf
```

        and comment out the localhost part of the config and use ipv4 loopback.

```
#inet_interfaces = localhost
inet_interfaces = 127.0.0.1
```

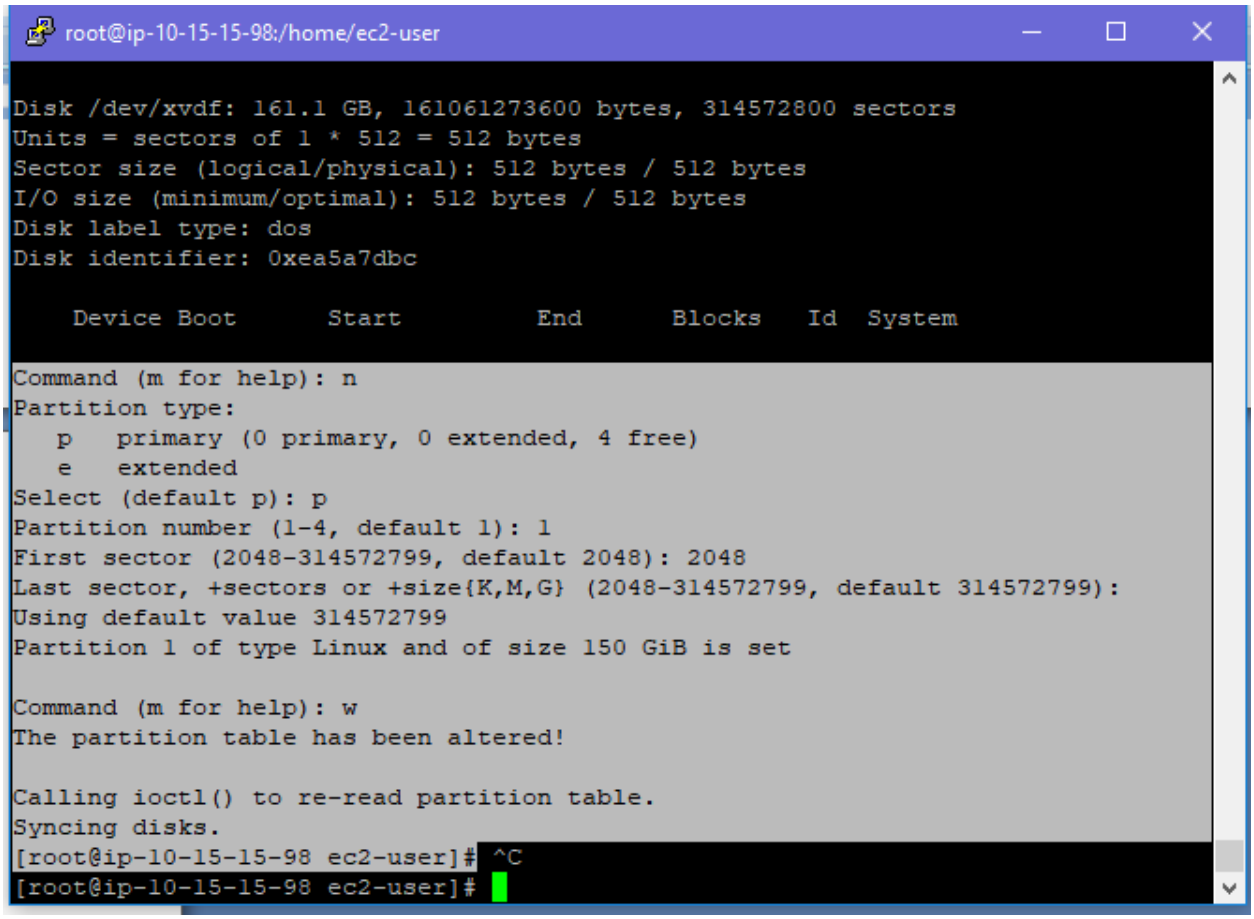
    f. Then, restart ssh to reflect the changes.

```
# systemctl restart sshd
```

5. **yum update**
6. **Create the Volume EBS**

    a. fdisk -l (Disk /dev/xvdf: 161.1 GB, 161061273600 bytes, 314572800 sectors)
    b. fdisk /dev/xvdf
    c. Command(m for help):

d.



e. Fdisk –l
   i. New disk name /dev/xvdf1

f. `mkfs.xfs /dev/xvdf1`
g. `mkdir /mnt/ebs`
h. `mount /dev/xvdf1 /mnt/ebs/`
i. `#blkid`
   i. `/dev/xvdf1: UUID="3de4431c-a3d4-4457-a059-33bb60529cad" TYPE="ext4" or "xfs"`

j. Make an entry in **/etc/fstab** file for permanent mount at boot time.
   i. Edit fstab
   ii. Add the line -
   UUID=3de4431c-a3d4-4457-a059-33bb60529cad /mnt/ebs/ xfs defaults 1 2
   iii. Save and exit

7.

8. Swap memory

	total	used	free	shared	buffers	cached
Mem:	491	460	30	164	30	228
-/+ buffers/cache:		202	289			
Swap:	0	0	0			

Here's the free memory after

	total	used	free	shared	buffers	cached
Mem:	491	485	5	122	21	323
-/+ buffers/cache:		140	350			
Swap:	511	0	511			

Here are the steps I took to enable swap space

```
$ dd if=/dev/zero of=/swapfile bs=1M count=4096
$ mkswap /swapfile
$ chmod 600 /swapfile
$ swapon /swapfile
```

You also have to edit your fstab file so the swap is available after reboot

```
vi /etc/fstab
```

Add the following to the bottom of the file - I made sure it lined up with the existing entries, but that's probably not necessary.

```
/swapfile swap swap defaults 0 0
```

To disable your existing swap,

```
# swapoff -a
```

9. **Java Installation**

```
a. wget --no-check-certificate --no-cookies --header "Cookie: oraclelicense=accept-
   securebackup-cookie" http://download.oracle.com/otn-pub/java/jdk/8u181-
   b13/96a7b8442fe848ef90c96a2fad6ed6d1/jdk-8u181-linux-x64.tar.gz
b. tar -zxvf jdk-8u*-linux-x64.tar.gz
c. mv jdk1.8.*/ /usr/
d. Install Oracle Java 8
     i. alternatives --install /usr/bin/java java /usr/jdk1.8.*/bin/java 2
    ii. alternatives --config java
```

e. Setup Environmental Variable

```
     i. export JAVA_HOME=/usr/jdk1.8.0_*
    ii. export JRE_HOME=/usr/jdk1.8.0_*/jre/
   iii. export PATH=$JAVA_HOME/bin:$PATH
```

f. To set it as a permanent, place the above three commands in the "**/etc/profile**" (All Users) or "**~/.bash_profile** " (Single User)

10.

11.

12. **Install wget**

```
$ sudo yum install wget
```

13. check server crash log
14. Install Apache2

```
# yum -y install httpd
```

systemctl enable httpd or # systemctl disable httpd

systemctl start httpd, systemctl status httpd

15. Install epel.7 ((Pre-request for PHP, redis-server,)

```
$ sudo rpm -ivh https://dl.fedoraproject.org/pub/epel/epel-release-latest-
7.noarch.rpm
$ sudo yum update
```

16. PHP 7.2 install

Install epel after start install PHP

```
$ sudo yum -y install epel-release
$ sudo yum-config-manager --enable remi-php72
$ sudo yum update
$ sudo yum install php
```

```
# rpm -Uvh https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
# wget http://rpms.remirepo.net/enterprise/remi-release-7.rpm
# rpm -Uvh remi-release-7.rpm epel-release-latest-7.noarch.rpm
# subscription-manager repos --enable=rhel-7-server-optional-rpms
# yum install yum-utils
```

17. Create user with ssh(pem file)

a. Refer the document \\Bsipl0011\h$\IT home\KB\Generate_the_Keypar_in_AWS_EC2_linux.pdf

Memory & Disk Monitoring

b. Amazon Linux 2 and Amazon Linux AMI

1. **# sudo yum install -y perl-Switch perl-DateTime perl-Sys-Syslog perl-LWP-Protocol-https perl-Digest-SHA.x86_64**

c. Red Hat Enterprise Linux 6.9 & Centos 7

i. **# sudo yum install perl-DateTime perl-CPAN perl-Net-SSLeay perl-IO-Socket-SSL perl-Digest-SHA gcc -y**

ii. **#sudo yum install zip unzip**

iii. #sudo cpan

iv. #cpan[1]>

1. cpan[1]> install YAML
2. cpan[2]> install LWP::Protocol::https
3. cpan[3]> install Sys::Syslog
4. cpan[4]> install Switch

d. Red Hat Enterprise Linux 7.4

i. **#sudo yum install perl-Switch perl-DateTime perl-Sys-Syslog perl-LWP-Protocol-https perl-Digest-SHA --enablerepo="rhui-REGION-rhel-server-optional" -y**

ii. **#sudo yum install zip unzip**

e. *Ubuntu Server*

i. **#sudo apt-get update**
ii. **#sudo apt-get install unzip**
iii. **#sudo apt-get install libwww-perl libdatetime-perl**

Getting Started for all server in the above heading

$ cd /

$ mkdir monitor ----- creat the folder in the root directory (/)

$ cd monitor

Download monitoring script

```
$curl https://aws-cloudwatch.s3.amazonaws.com/downloads/CloudWatchMonitoringScripts-1.2.2.zip -O
```

Extract the zip folder

```
$ unzip CloudWatchMonitoringScripts-1.2.2.zip
$ cd aws-scripts-mon
```

IAM role (instance profile) with your instance

- cloudwatch:PutMetricData
- cloudwatch:GetMetricStatistics
- cloudwatch:ListMetrics
- ec2:DescribeTags

copy the `awscreds.template` file included with the monitoring scripts to `awscreds.conf`

```
$ cp awscreds.template awscreds.conf
```

Add the following content to the awscreds.conf file

```
AWSAccessKeyId=my-access-key-id
AWSSecretKey=my-secret-access-key
```

To perform a simple test run without posting data to CloudWatch

```
$ ./mon-put-instance-data.pl --mem-util --verify --verbose
```

To collect all available memory metrics and send them to CloudWatch, counting cache and buffer memory as used

```
$ ./mon-put-instance-data.pl --mem-used-incl-cache-buff --mem-util --mem-used --mem-avail
```

To set a cron schedule for metrics reported to CloudWatch

1. Start editing the crontab using the following command:

```
$ crontab -e
```

2. Add the following command to report memory and disk space utilization to CloudWatch every five minutes:

```
*/5 * * * * ~/aws-scripts-mon/mon-put-instance-data.pl --mem-used-incl-cache-buff --mem-util --disk-space-util --disk-path=/ --from-cron       (note: for root (/)

*/5 * * * * ~/aws-scripts-mon/mon-put-instance-data.pl --mem-used-incl-cache-buff --mem-util --disk-space-util --disk-path=/ --disk-path=/mnt/ebs/ --from-cron
                                                    (note: for addition partition /mnt/ebs/)
```

If the script encounters an error, the script will write the error message in the system log.

To collect aggregated metrics for an Auto Scaling group and send them to Amazon CloudWatch without reporting individual instance metrics

```
$ ./mon-put-instance-data.pl --mem-util --mem-used --mem-avail --auto-scaling=only
```

To collect aggregated metrics for instance type, AMI ID and region, and send them to Amazon CloudWatch without reporting individual instance metrics

```
$ ./mon-put-instance-data.pl --mem-util --mem-used --mem-avail --aggregated=only
```