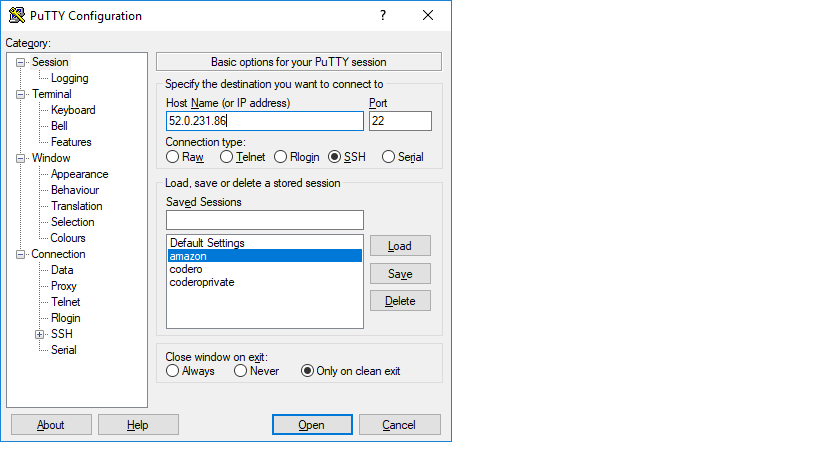
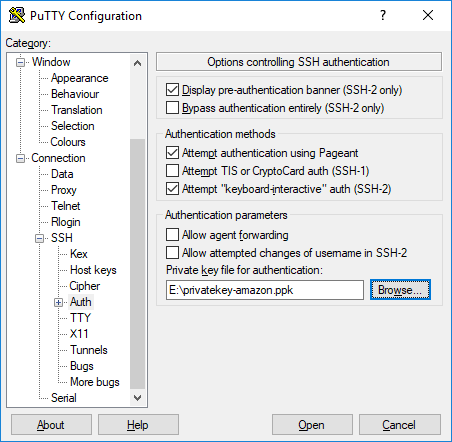
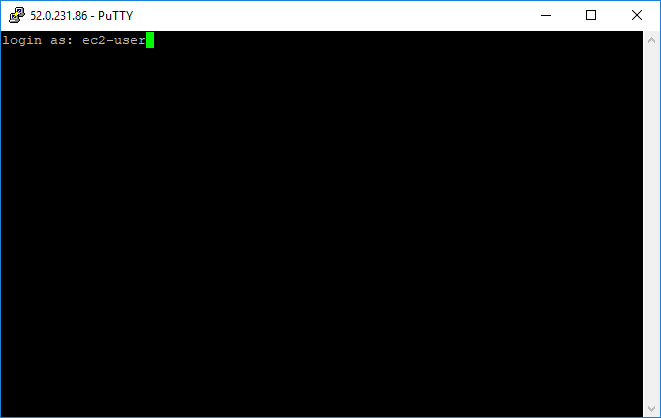
AWS Login Screenshots:



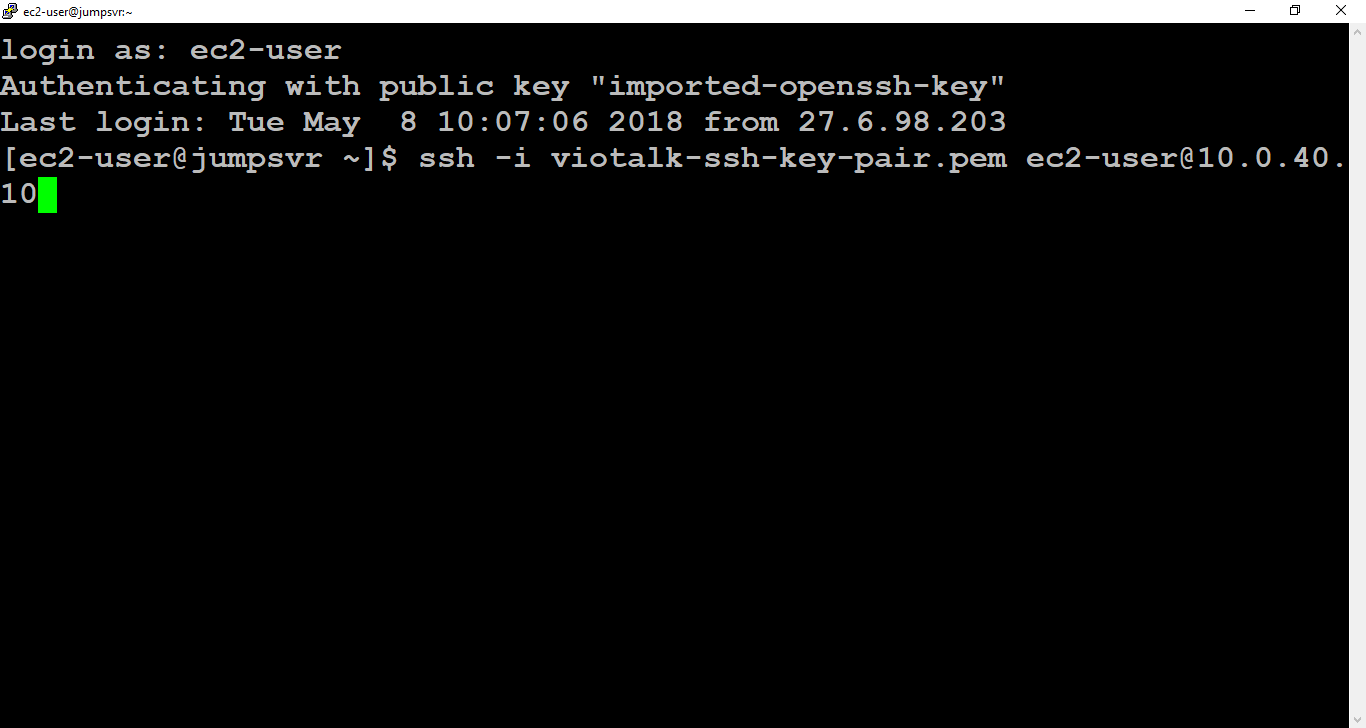
We need to mention ppk file for amazon key from our local machine as shown below screenshot



After that click on open to connect which shown below screenshot and login as ec2-user:

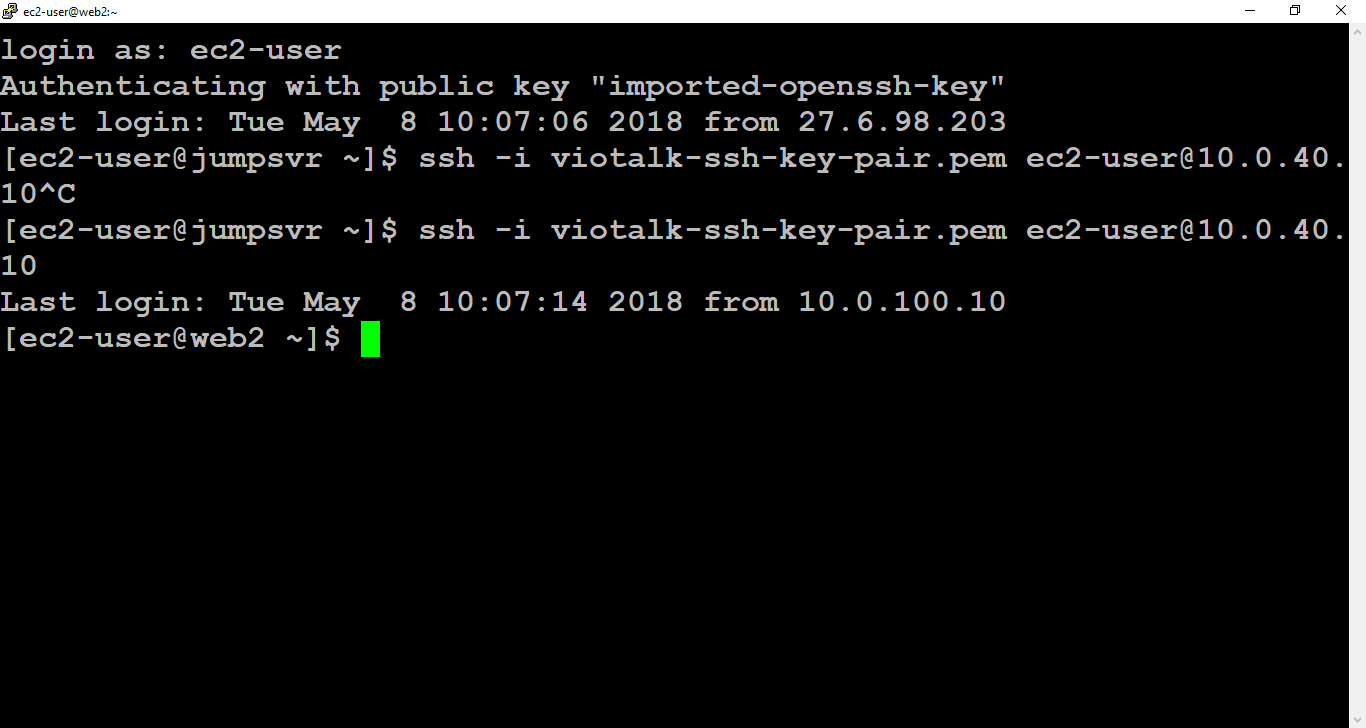


Then we connect to web2 as shown below screenshot:

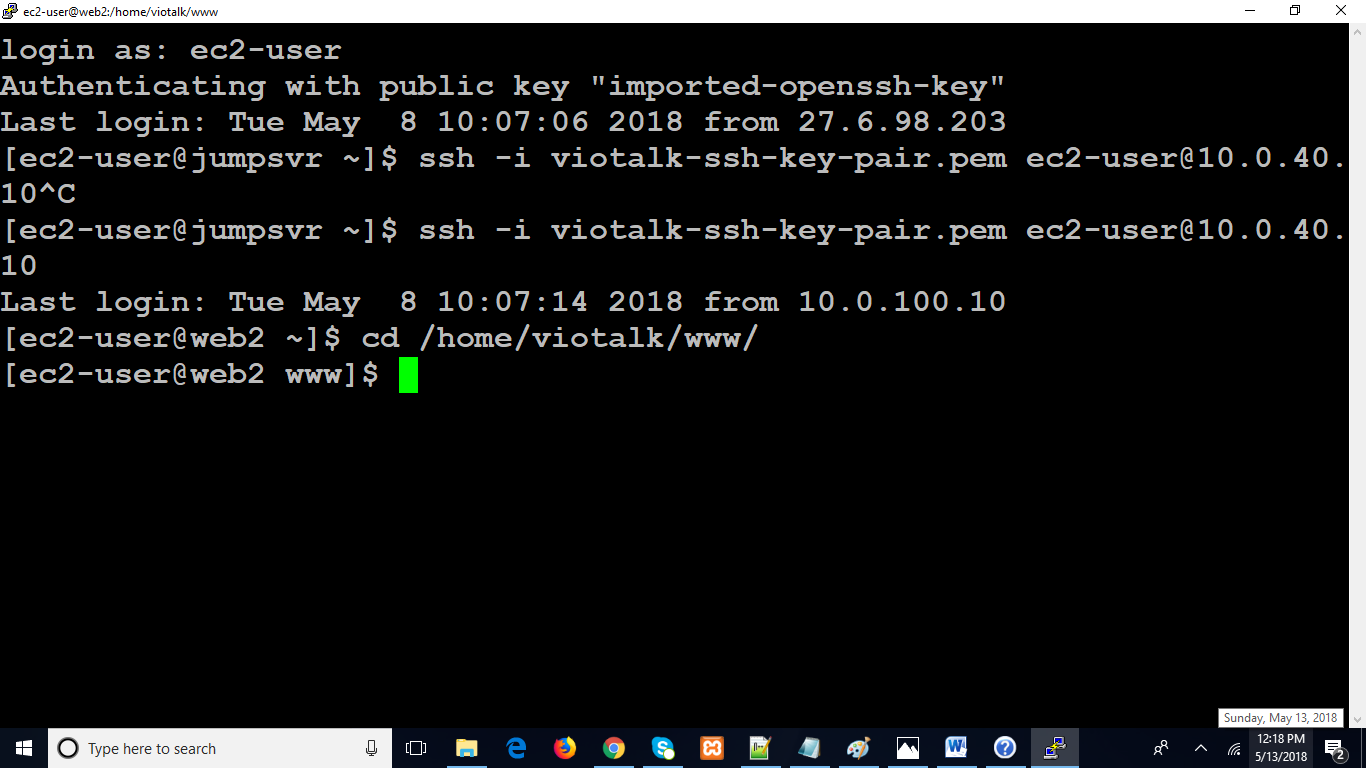


ssh -i viotalk-ssh-key-pair.pem [ec2-user@10.0.40.10](mailto:ec2-user@10.0.40.10)

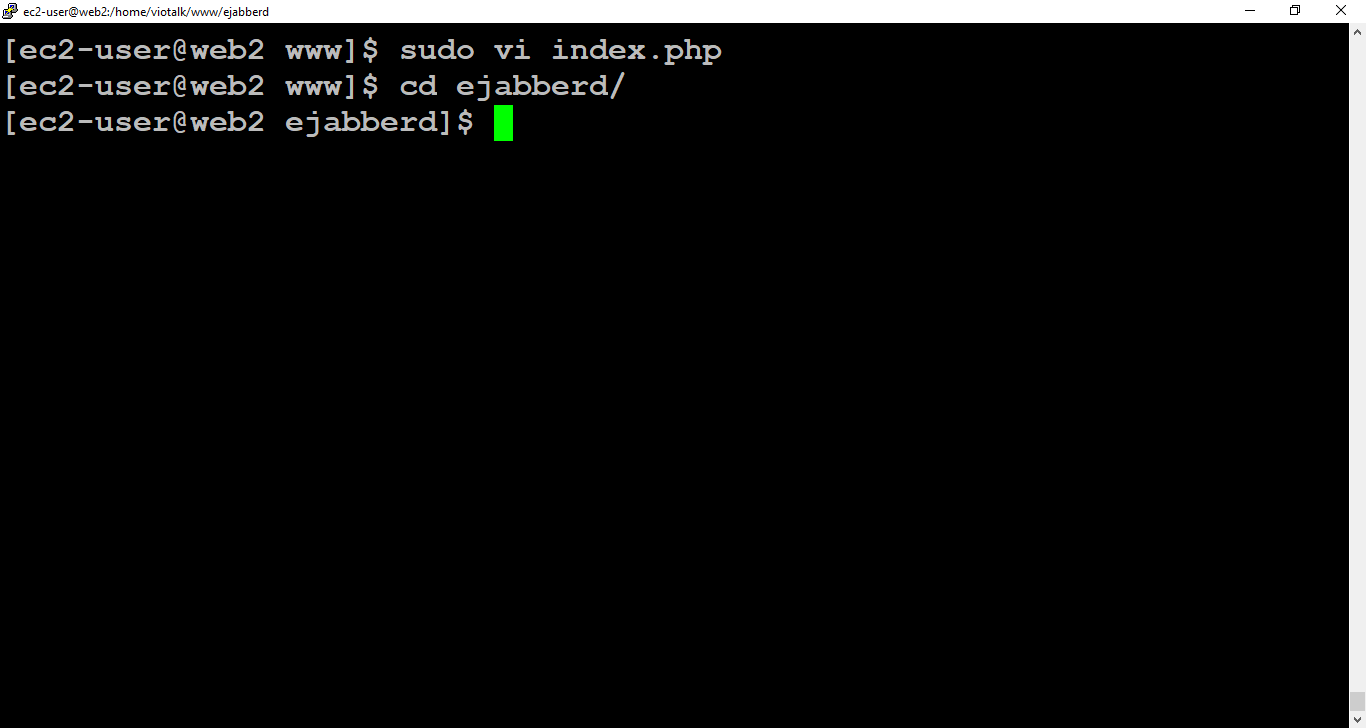
Now web2 is logged in as shown below command.



Now file directory for viotalk is as follows:



Now viotalk services for both android and ios:



For android services – cd ejabberd/android

For ios services – cd ejabberd/ios

Viotalkregister.php – used for registering onto app

Viotalkverifyotp.php – used to verify otp from email or sms(kapsystems).

Viotalkvoip.php – used for audio and video calling notifications mobile.

Viotalkiosjsonuploadv2.php – contacts upload only for viotalk

$params = array(

'credentials' => array(

'key' => 'AKIAJKH65D6FGZ62VRQQ',

'secret' => 'bXsJgDUNWY0R9VAs9qbkNYPqHSnBX1hMcoBlY2qP',

),

'region' => 'us-east-1', // < your aws from SNS Topic region

'version' => 'latest'

);

$sns = new \Aws\Sns\SnsClient($params);

$args = array(

"SenderID" => "VIOTALK",

"SMSType" => "Transactional",

"Message" => $sms,

"PhoneNumber" => $mobile

);

$result = $sns->publish($args);

viotalkofflinepush.php – used for sending notifications chat(firebase)..

<https://gist.github.com/samvermette/759564> -- reference for push notifications ios.

stream\_context\_set\_option($ctx, 'ssl', 'passphrase', ‘VioTalk123’);

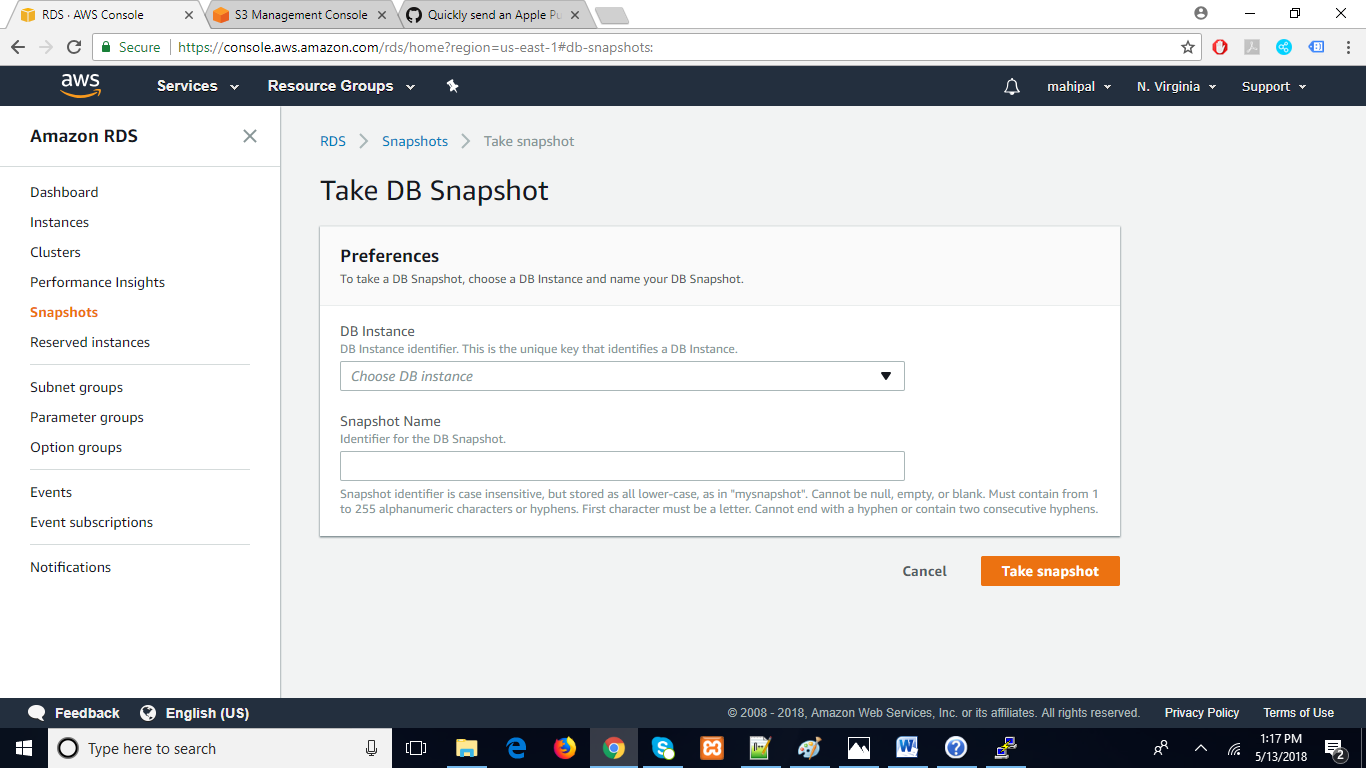
database name: viotalkdb.

Amazon AWS Mysql:

---------------------------

mysql -hviotalkdb.c2amfvktfbda.us-east-1.rds.amazonaws.com -uviotalkdb -pVioTlkRdS12 -P3306

mysql -hviotalkdb.c2amfvktfbda.us-east-1.rds.amazonaws.com -uviotalk -pV10talk -P3306



Roundcube mysql:

----------------------------

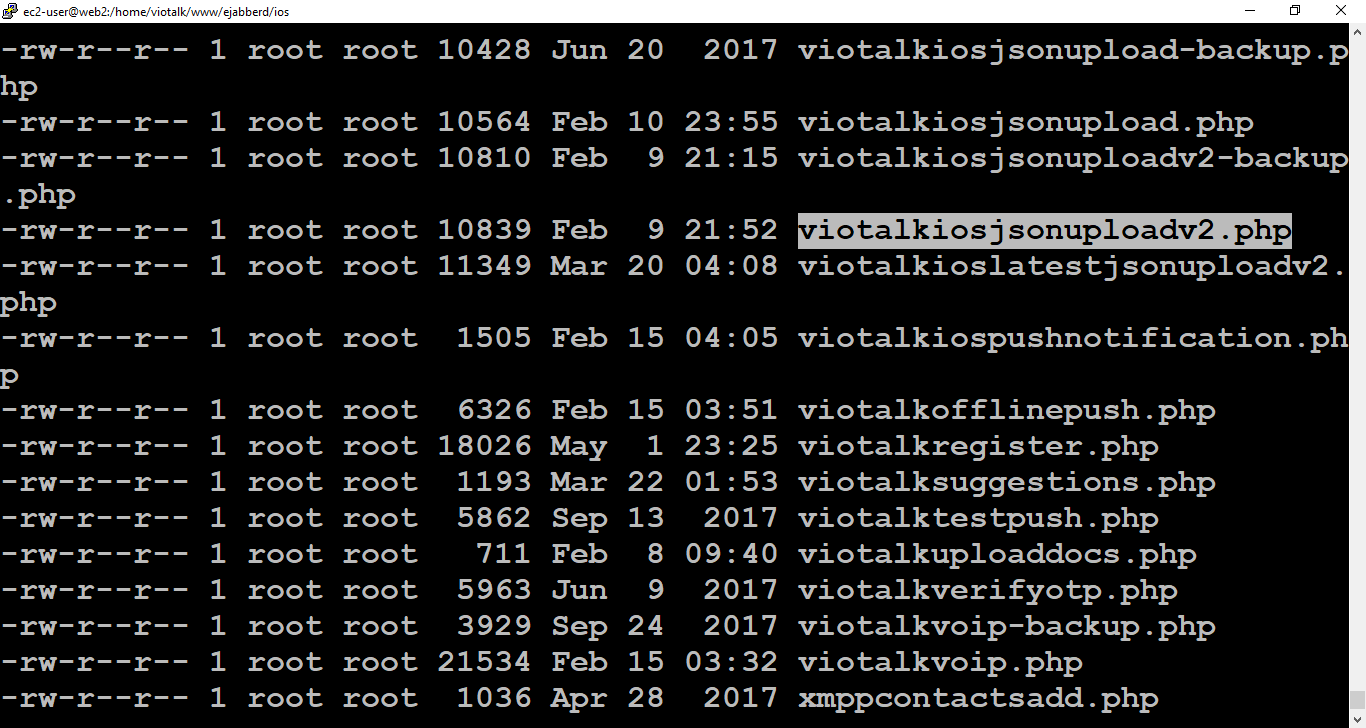
mysql -hroundcube.c2amfvktfbda.us-east-1.rds.amazonaws.com -P3306 -uroundcube -p'R2oundC8ubE$x' -Droundcube

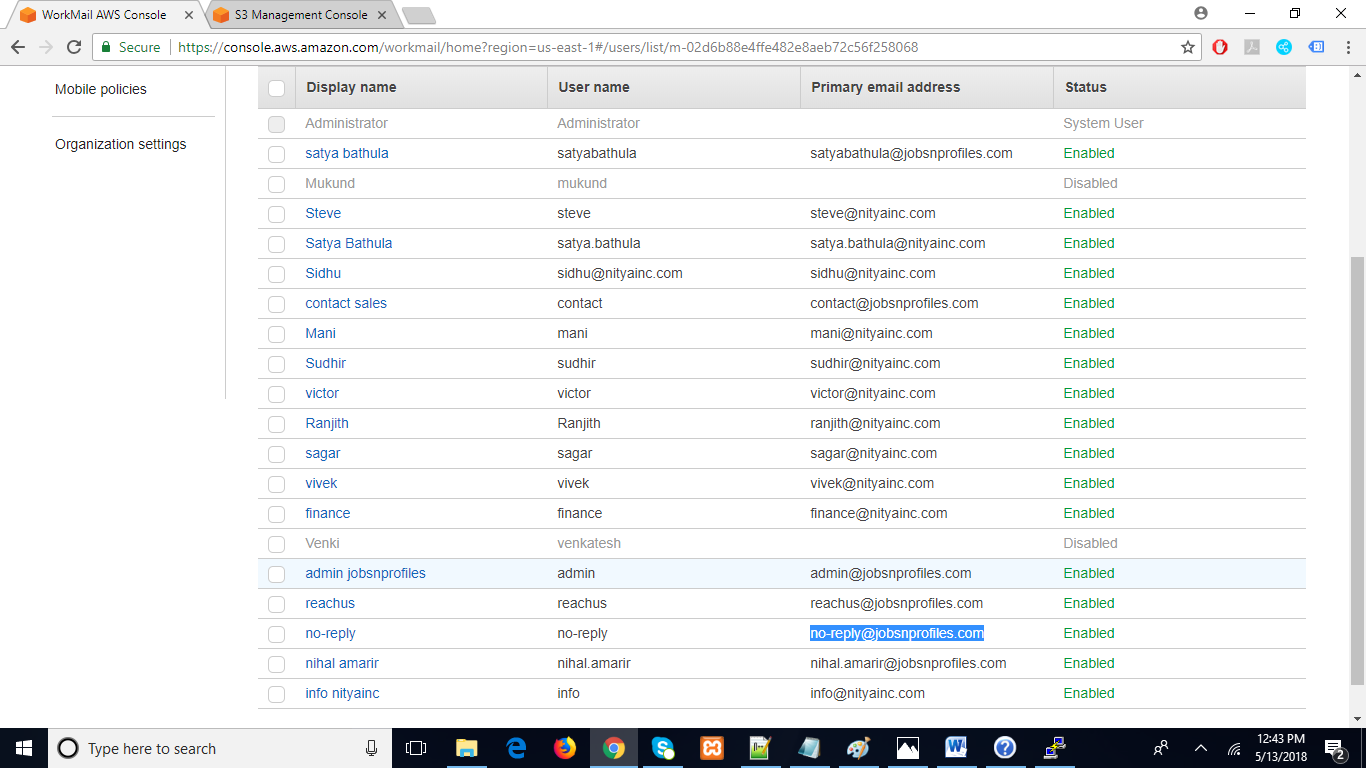
viotalkiospushnotification.php – used to store device token in a table called viotalkvoipiospush.

For storing voip badge count – viotalkiosbadgecount table.

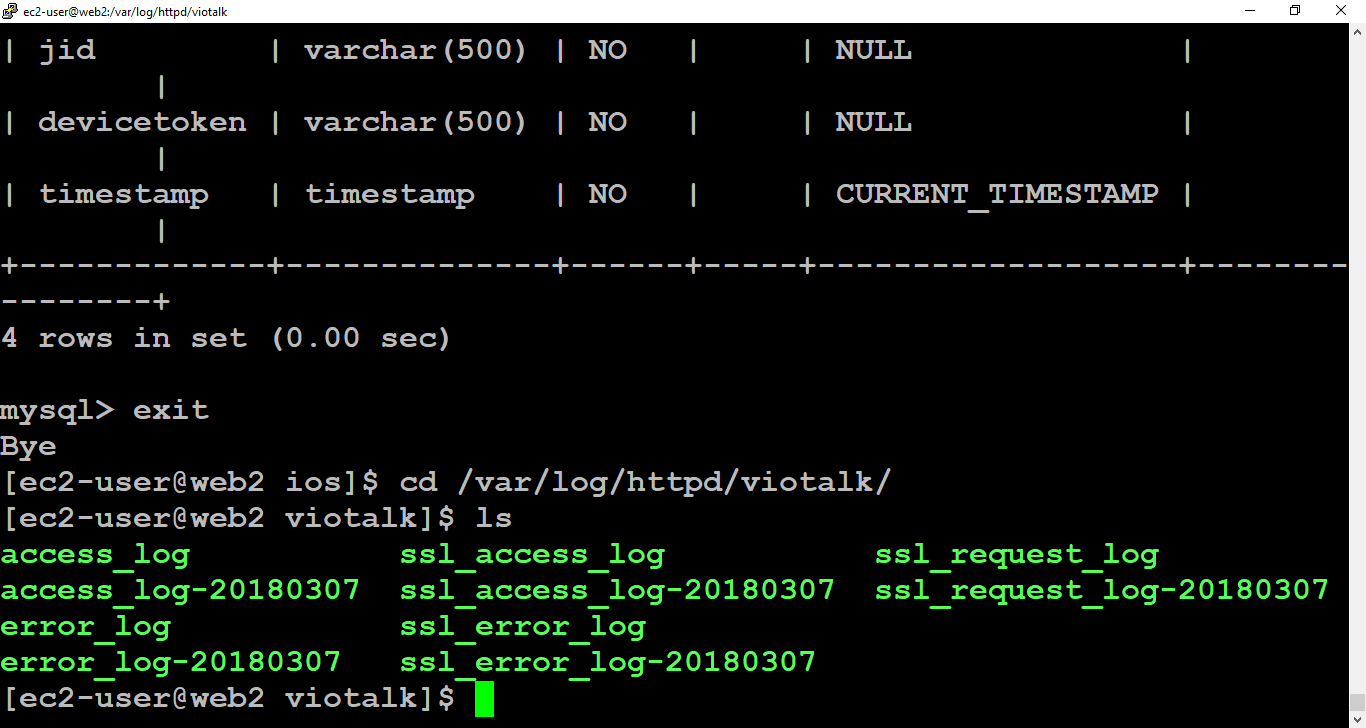
Viotalkiospush table – for sending normal push messages.

Viotalkbadge table – for sending normal push messages

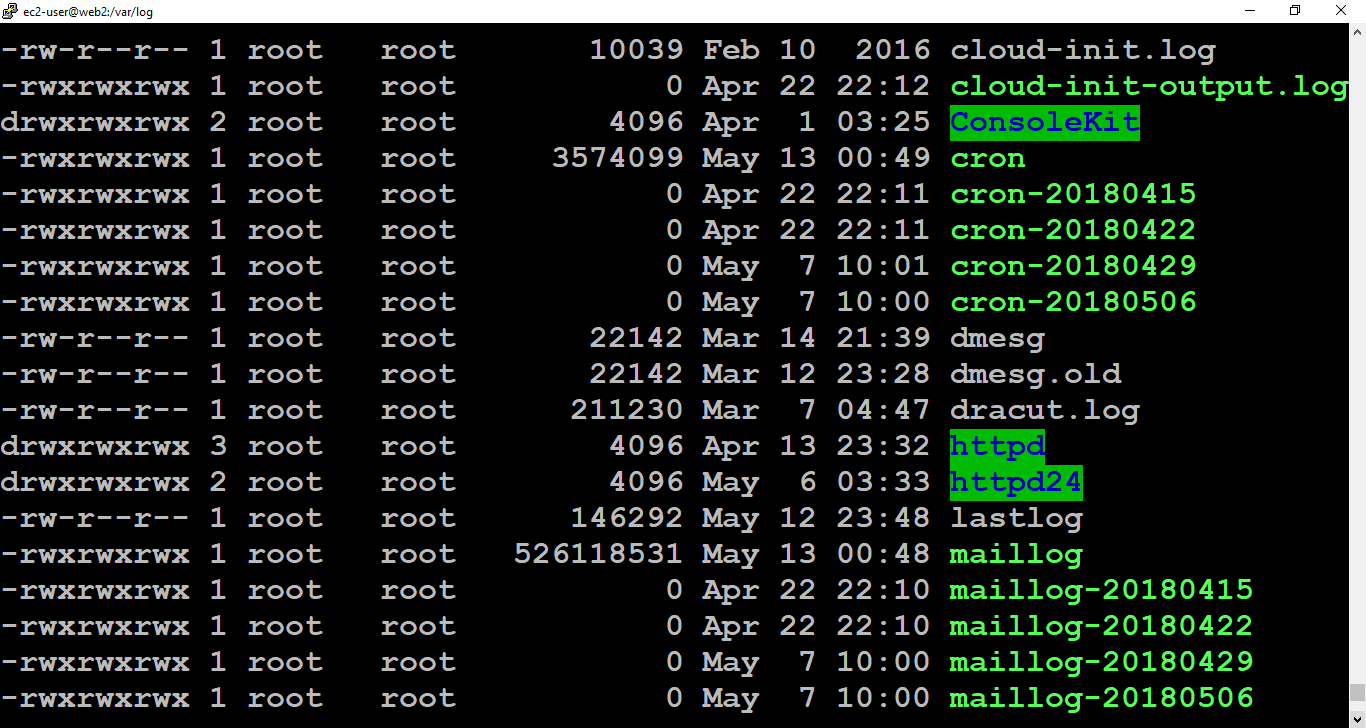




Viotalk log files:



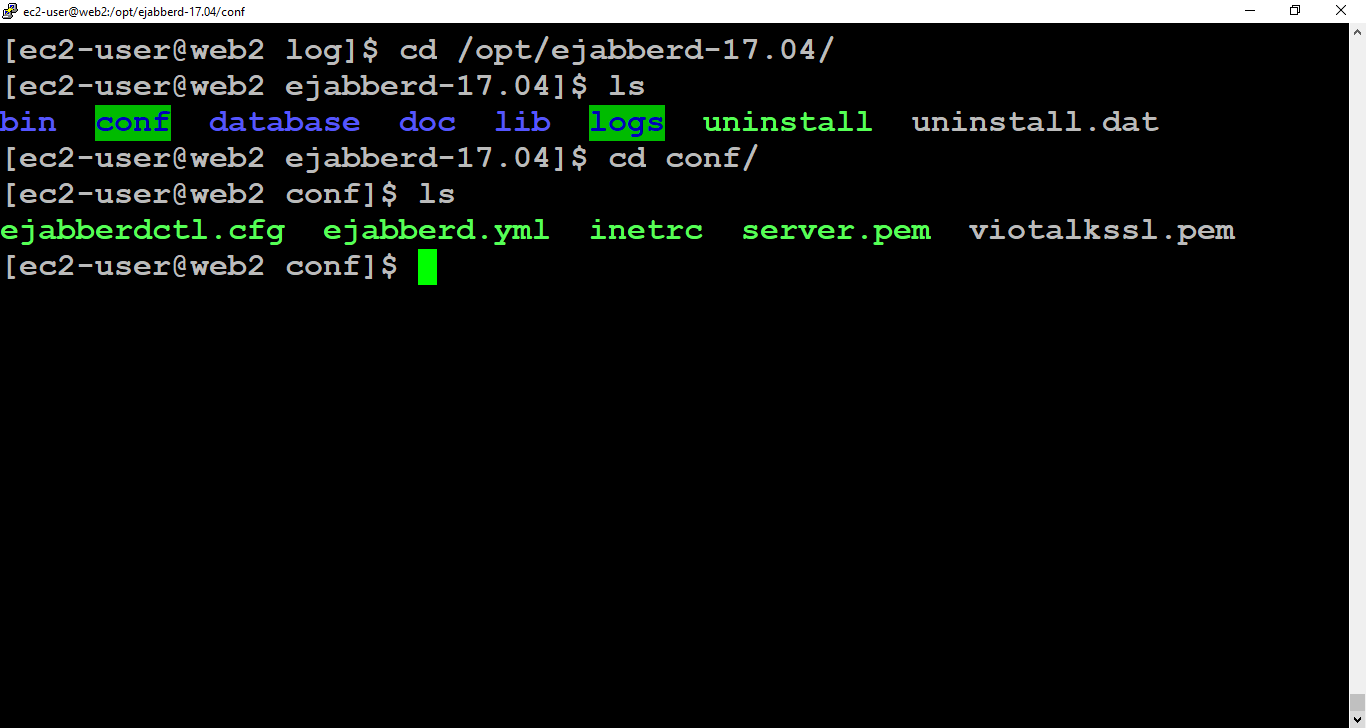
Cron and mail logs, sms:

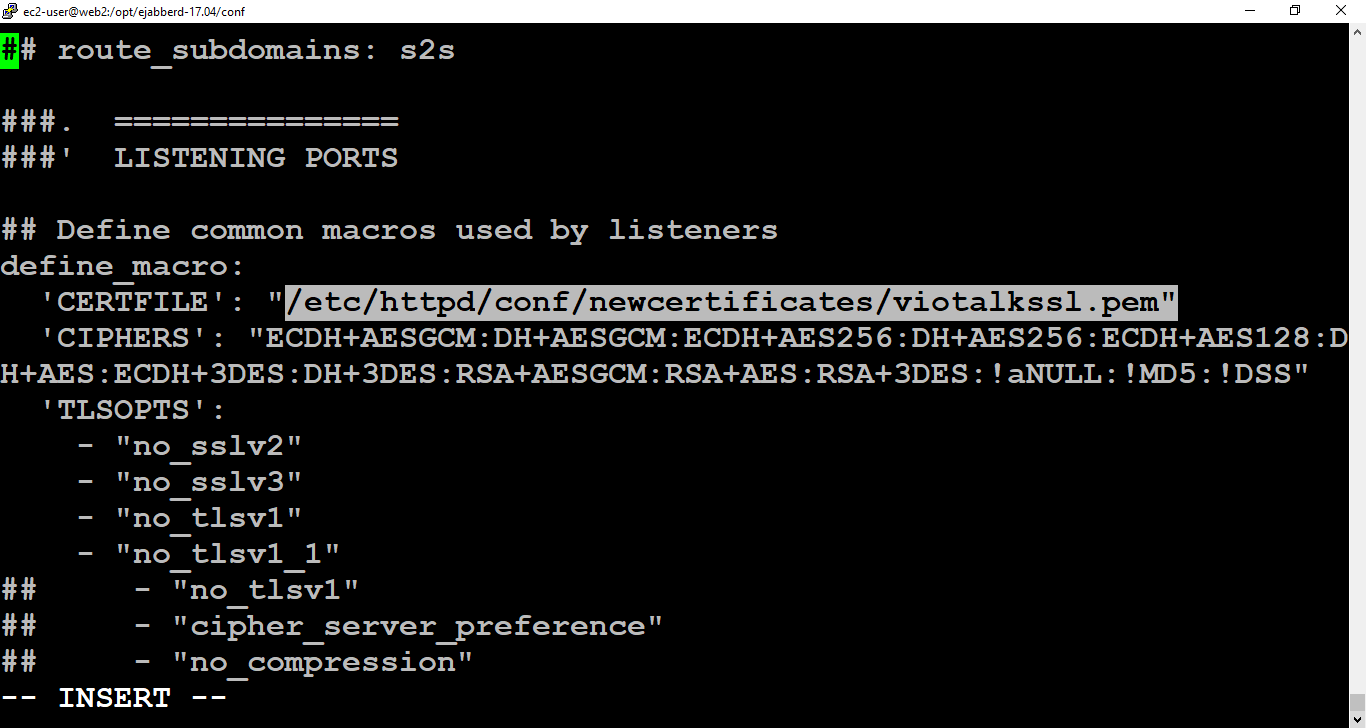


Ejabberd configuration and server certificate:

Sudo wget <https://www.process-one.net/downloads/downloads-action.php?file=/ejabberd/18.04/ejabberd-18.04-linux-x86_64-installer.run>

Sudo ./ ejabberd-18.04-linux-x86\_64-installer.run

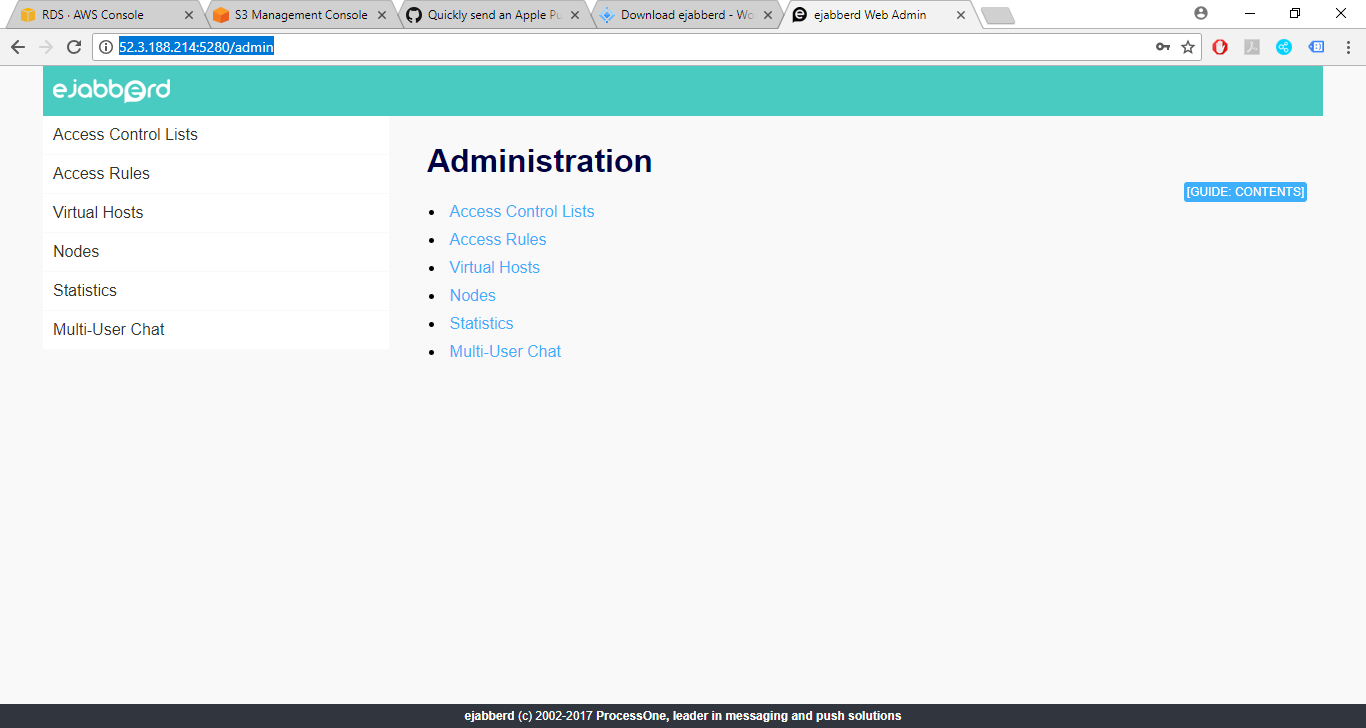




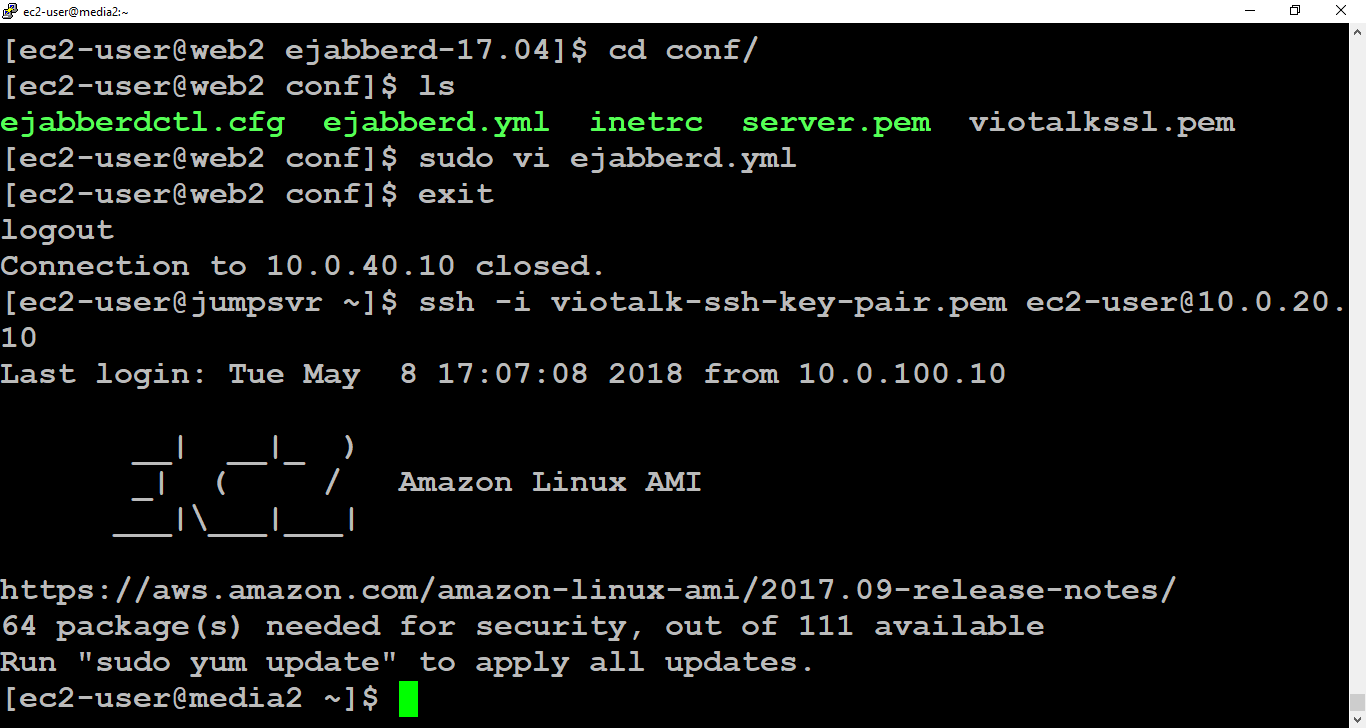
Ejabberd admin login:

<http://52.3.188.214:5280/admin> (viotalk admin)

username: [admin@viotalk.com](mailto:admin@viotalk.com) password: nitya@123

https://docs.ejabberd.im/admin/configuration/ 

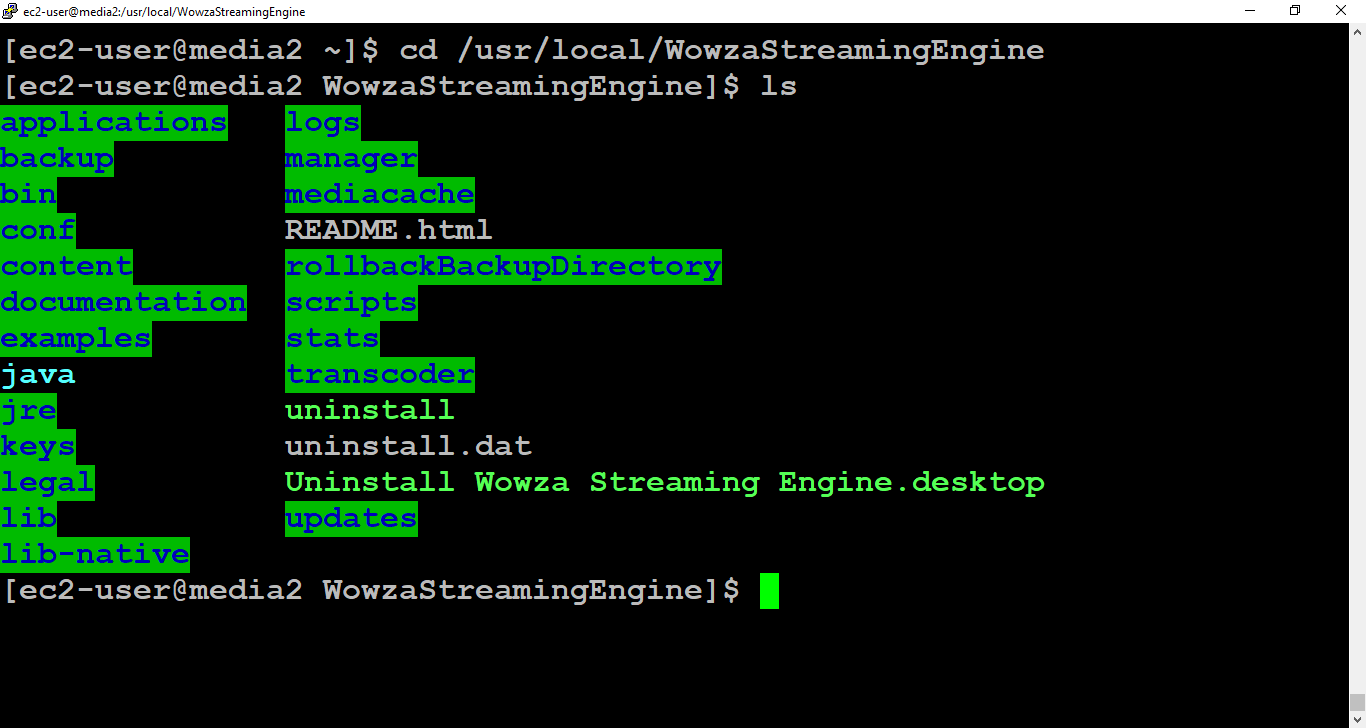
AWS media2 server (Wowza Server):

ssh -i viotalk-ssh-key-pair.pem ec2-user@10.0.20.10 

Wowza Streaming Engine installation:

<https://www.wowza.com/docs/how-to-install-and-configure-wowza-streaming-engine>

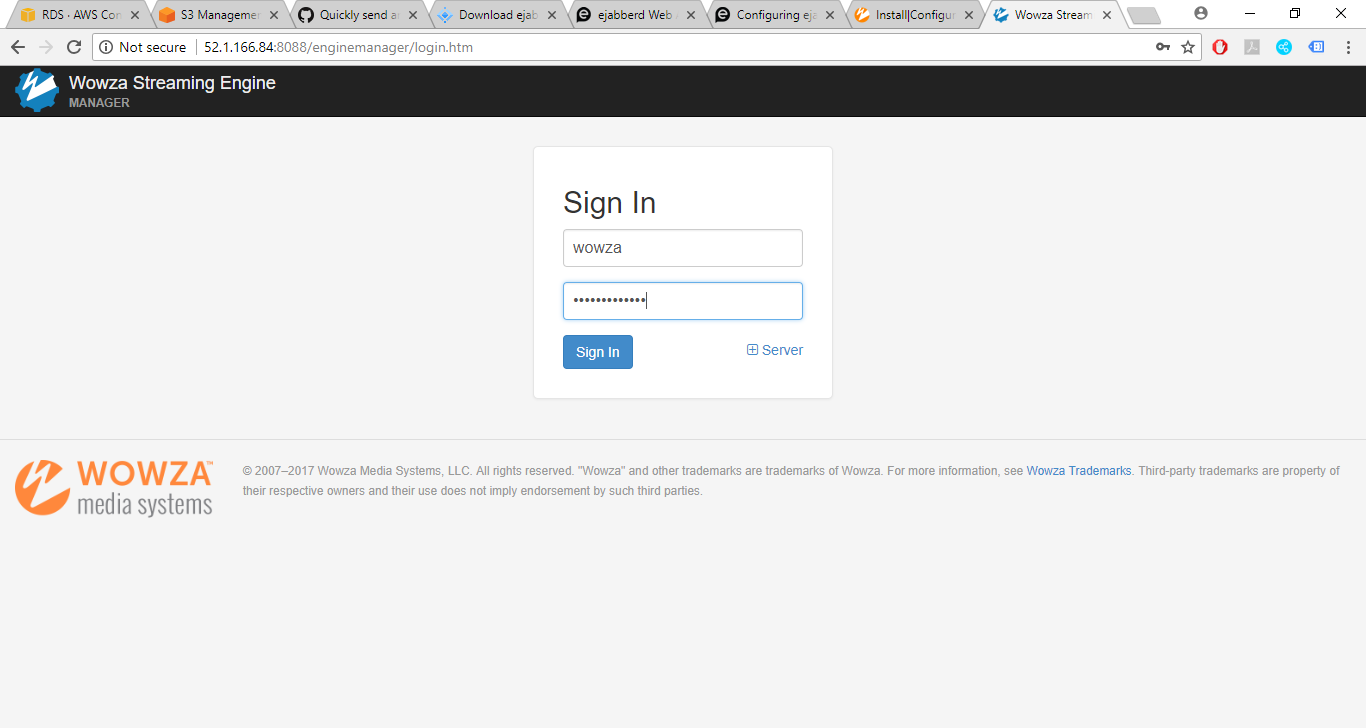
Wowza Streaming Engine file directory:

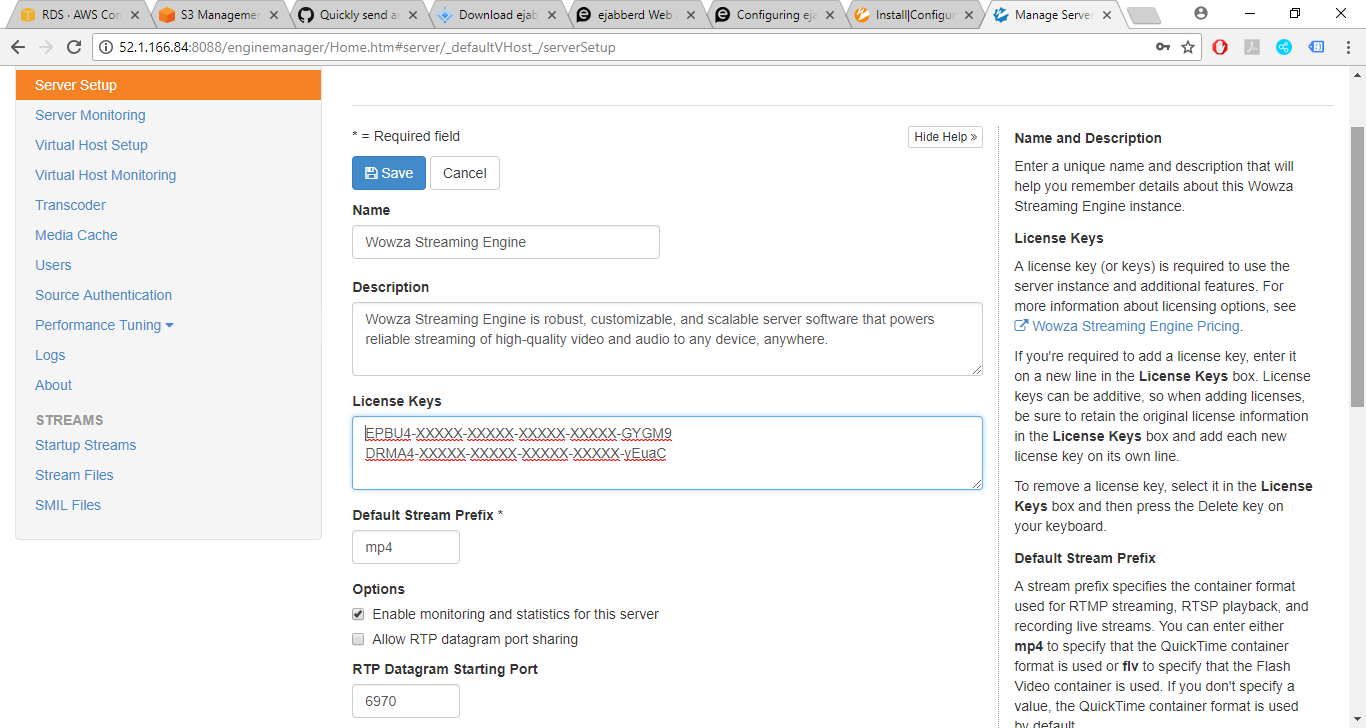


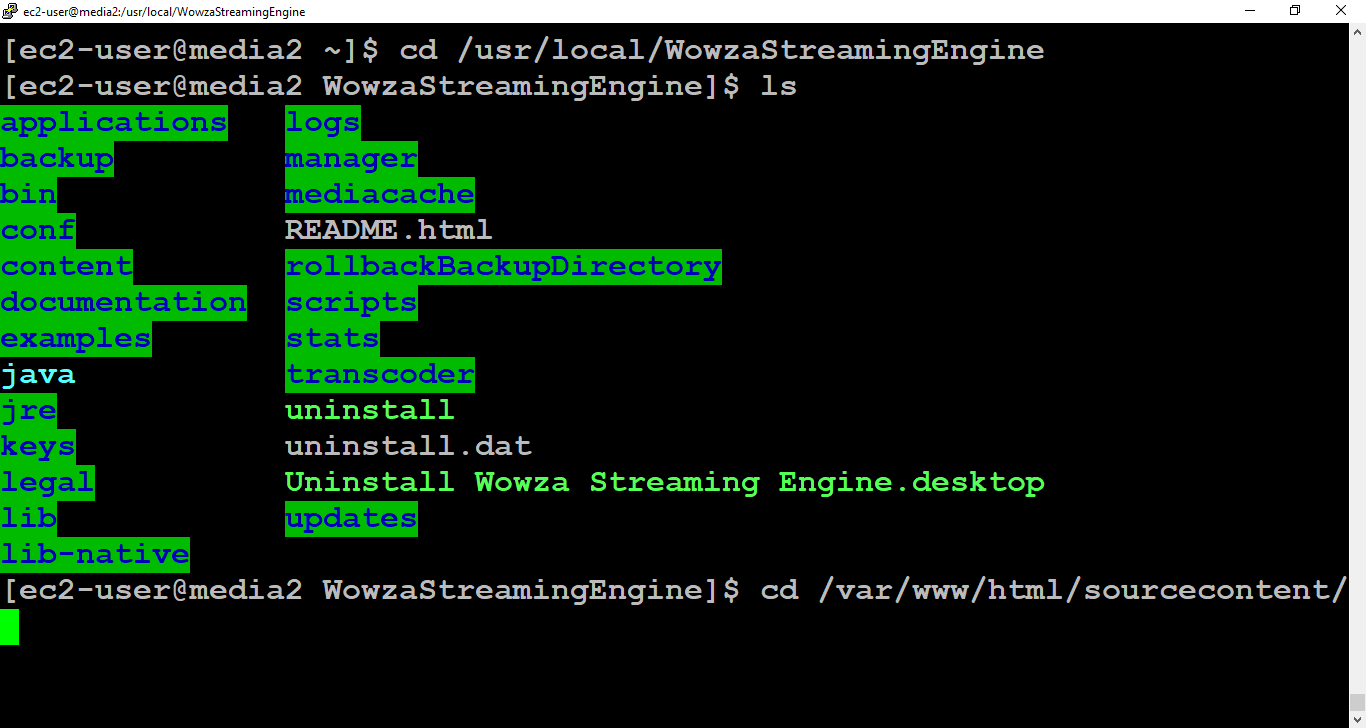
http://52.1.166.84:8088/

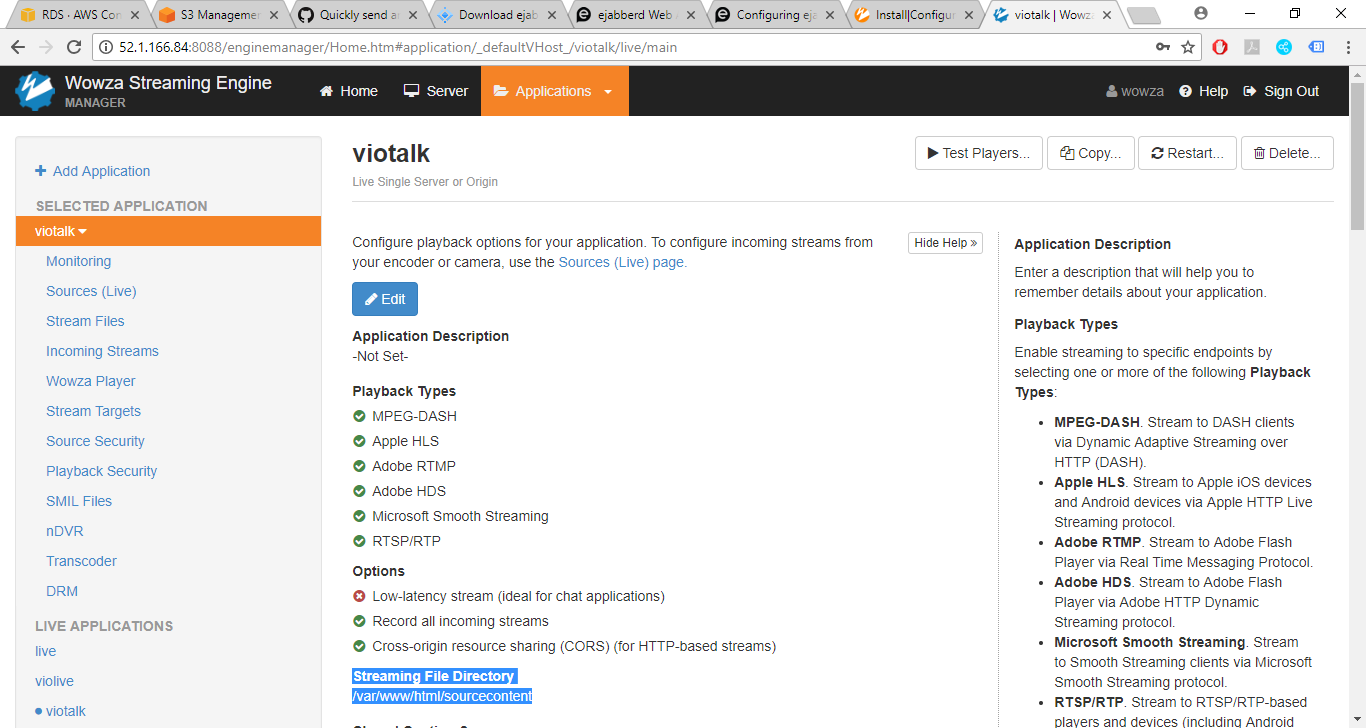
username: wowza

password: woWzav10t@lK@





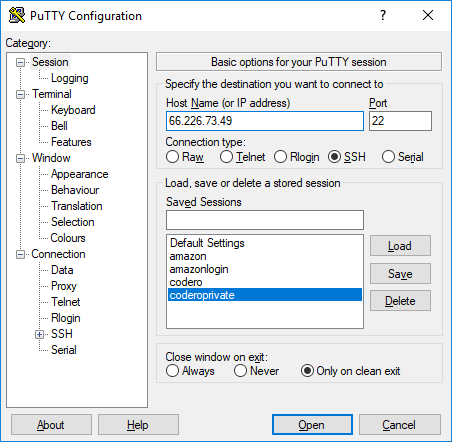




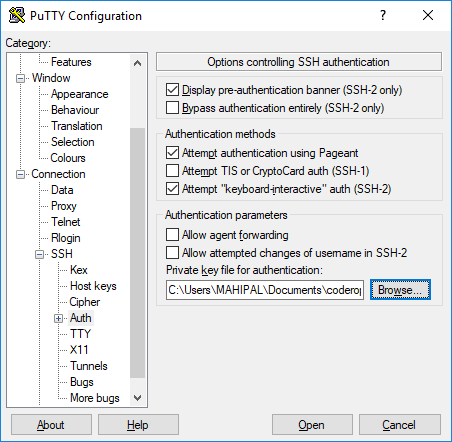
Media2 file directories for apache server(/var/www/html):



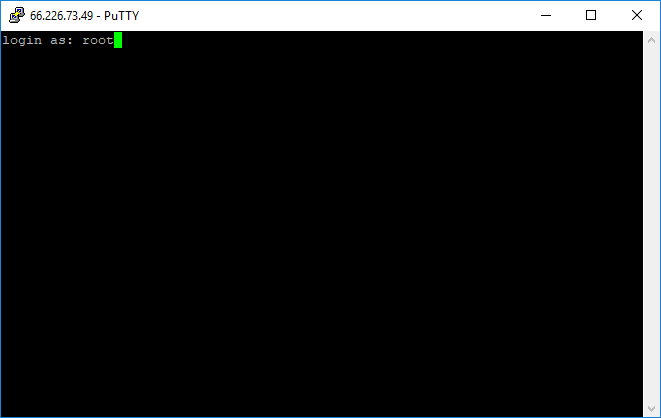
Codero using ssh:



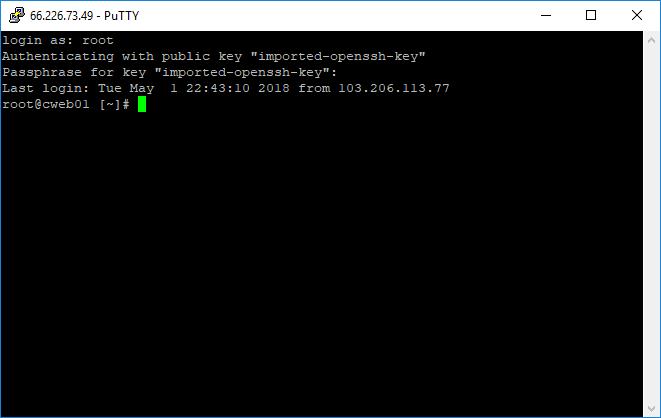
Coderoprivatekey upload – go to ssh 🡪 auth 🡪 browser ppk file(coderoprivatekey).



Login as root



Passphrase for ssh key: NityaInc@123

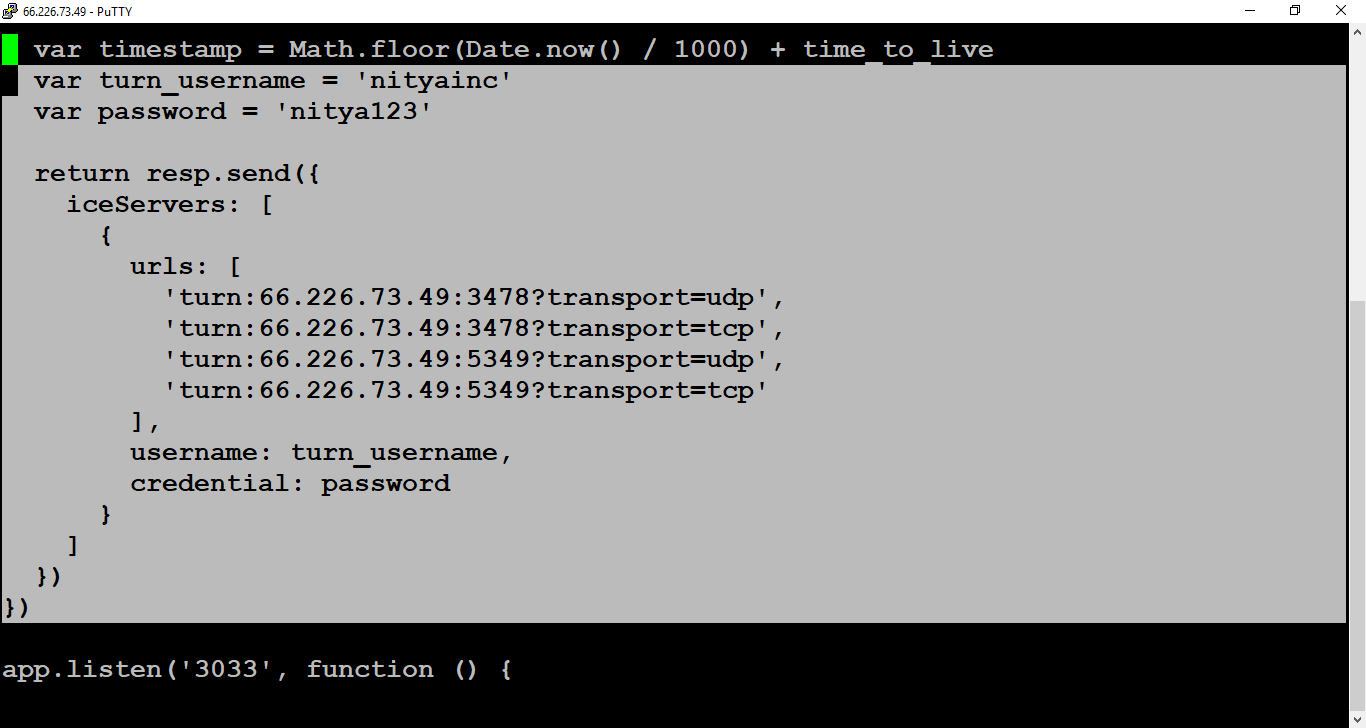


For Audio and Calling server:

To setup this server we need to install google cloud sdk with python > 2.5 first step..

Apprtc server reference link:

<https://github.com/webrtc/apprtc>



File directory for collider:

cd /home/vmailu/public\_html/apprtc/src/collider/collidermain

main.go



Start server both collider and apprtc:

This is starting apprtc server using iceconfig ports and domain

nohup ~/google-cloud-sdk/bin/dev\_appserver.py --host 66.226.73.49 --port 8000 ./out/app\_engine/ > my.log 2>&1 &

This is used for starting collider with our domain name matching ssl to connect while placing audio and video calls.

nohup /collider/collidermain -port=8089 -tls=true > /collider/collider.log 2>&1 &

for checking ports which are running using this command:

netstat –plnt

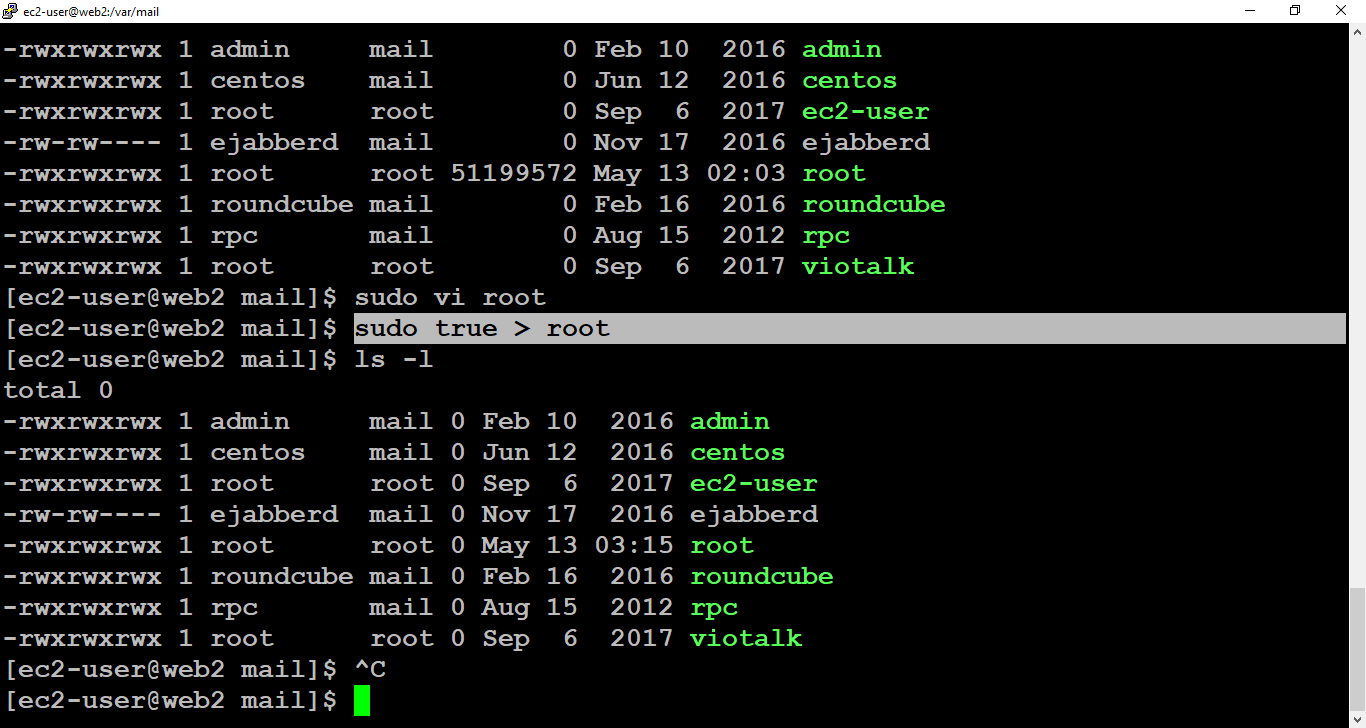
turn/stun server installation guide link:

turn installation steps to configure and set our public ip address.

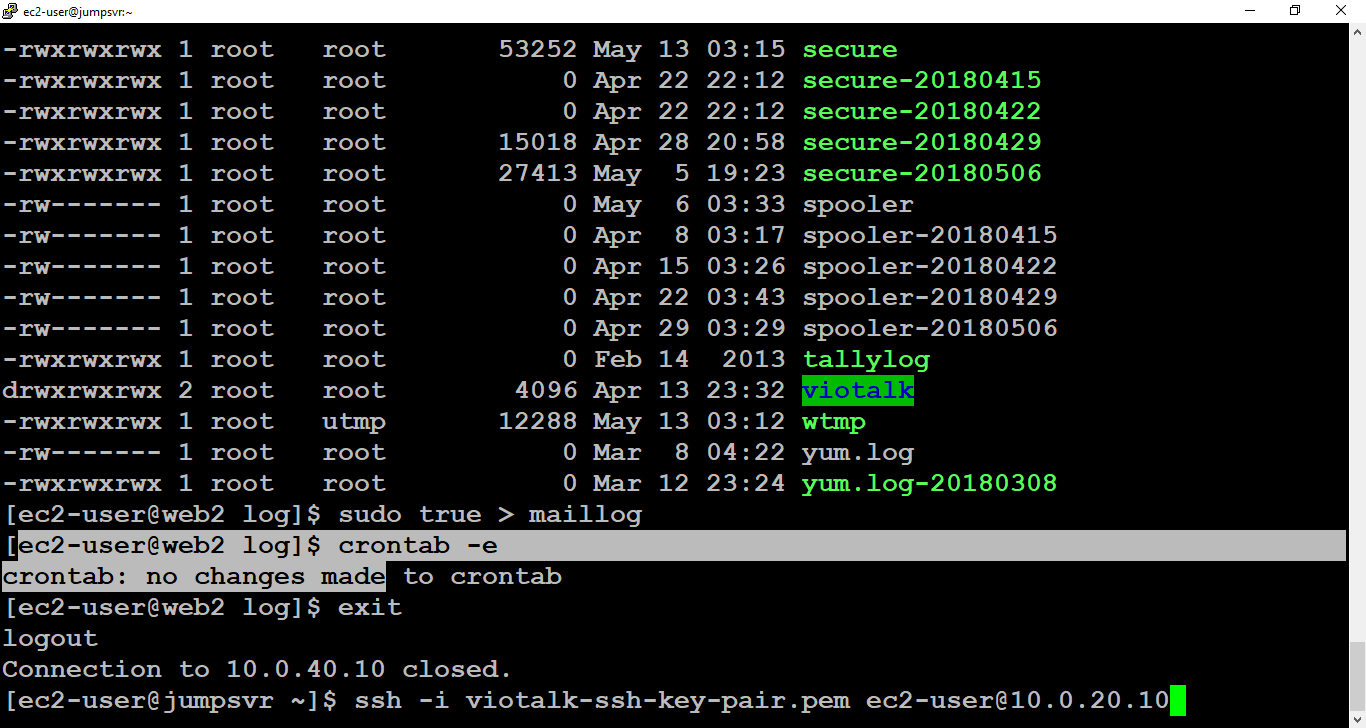
Go to Return installation on Ubuntu/centos using this below link:

<https://www.webrtc-experiment.com/docs/TURN-server-installation-guide.html>

service resiprocate-turn-server restart



Viotalk cron jobs(web2)



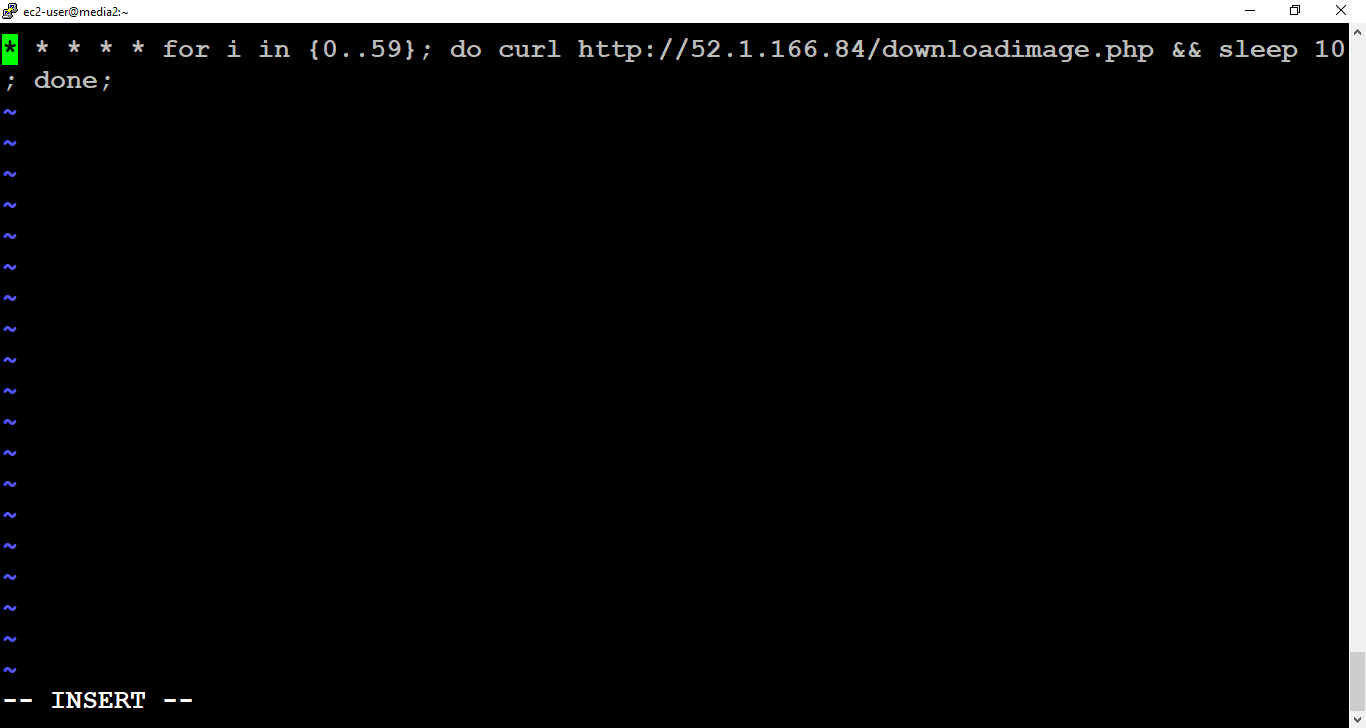
First cron is for sending push notifications for complete offline chat messages.

Second cron is used to send audio and video calls push notifications for messages.

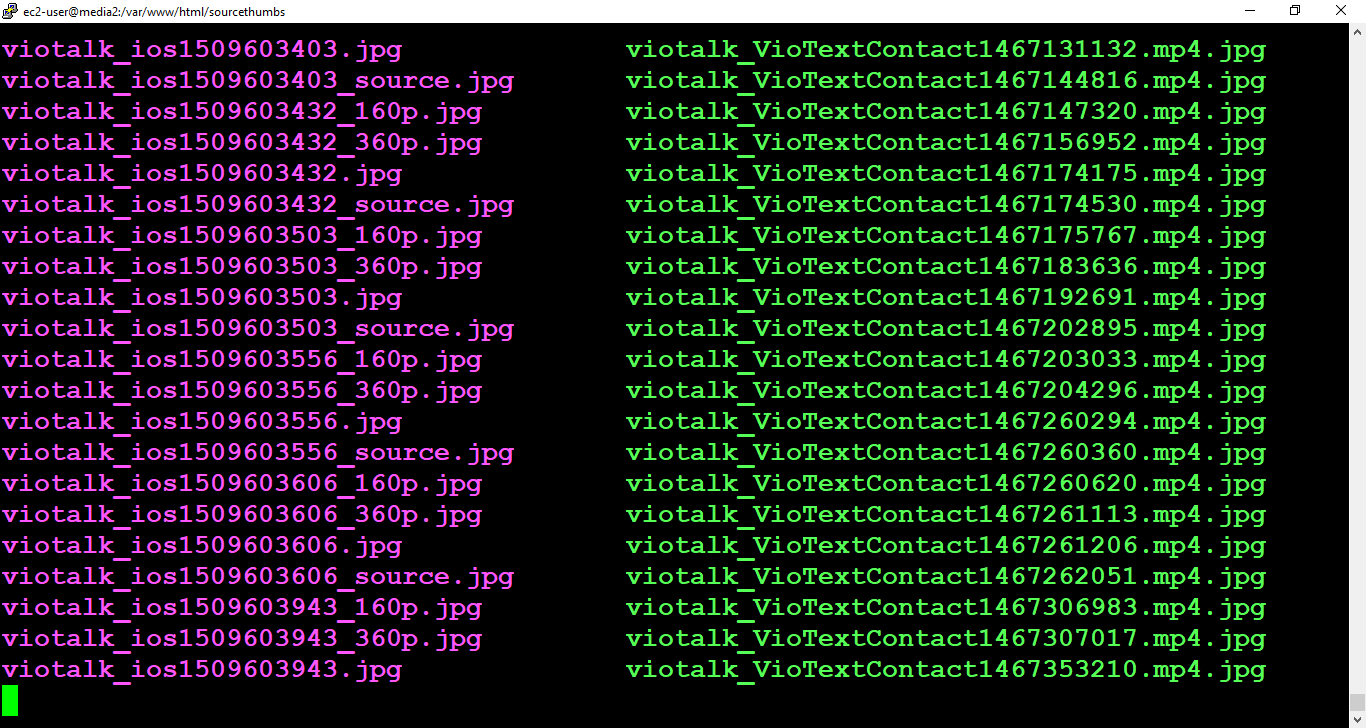


Media2 cron jobs(Wowza Media Server):

Thumbnail creation while video streaming with wowza.



Source thumbs folder where all video images are will get stored.(/var/www/html/sourcethumbs) in wowza media server.



Wowza streaming Engine start, stop, restart commands:

Service WowzaStreamingEngine start|restart|stop

<https://www.wowza.com/docs/how-to-install-and-configure-wowza-streaming-engine>

xmpp for web platform link like whatsapp web chatting:

<https://github.com/strophe/strophejs>

Ejabberd stopping server commands as follows:

Sudo killall epmd

Sudo killall beam.smp

To Start the ejabberd server:

/opt/ejabberd-17.04/bin/ejabberctl start

To know whether ejabberd server is running or not:

/opt/ejabberd-17.04/bin/ejabberctl status

In configuration ejabberd file, we need to add this port 5281 in order to connect xmpp.

-

port: 5281

ip: "::"

module: ejabberd\_http

request\_handlers:

"/websocket": ejabberd\_http\_ws

"/api": mod\_http\_api

## "/pub/archive": mod\_http\_fileserver

web\_admin: true

http\_bind: true

tls: true

certfile: 'CERTFILE'

## register: true

captcha: false

Apache Solr download link:

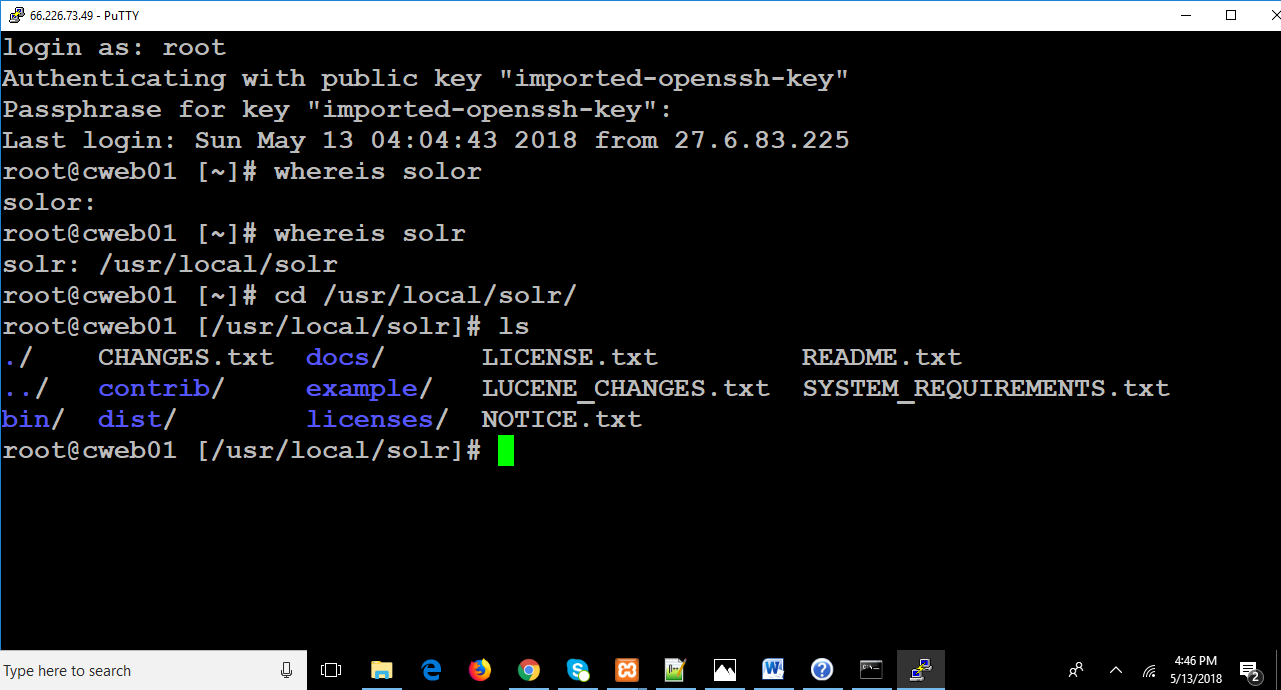
<http://www-eu.apache.org/dist/lucene/solr/7.3.0/>

JDk 1.8 need to be install in our system in order to download solr server

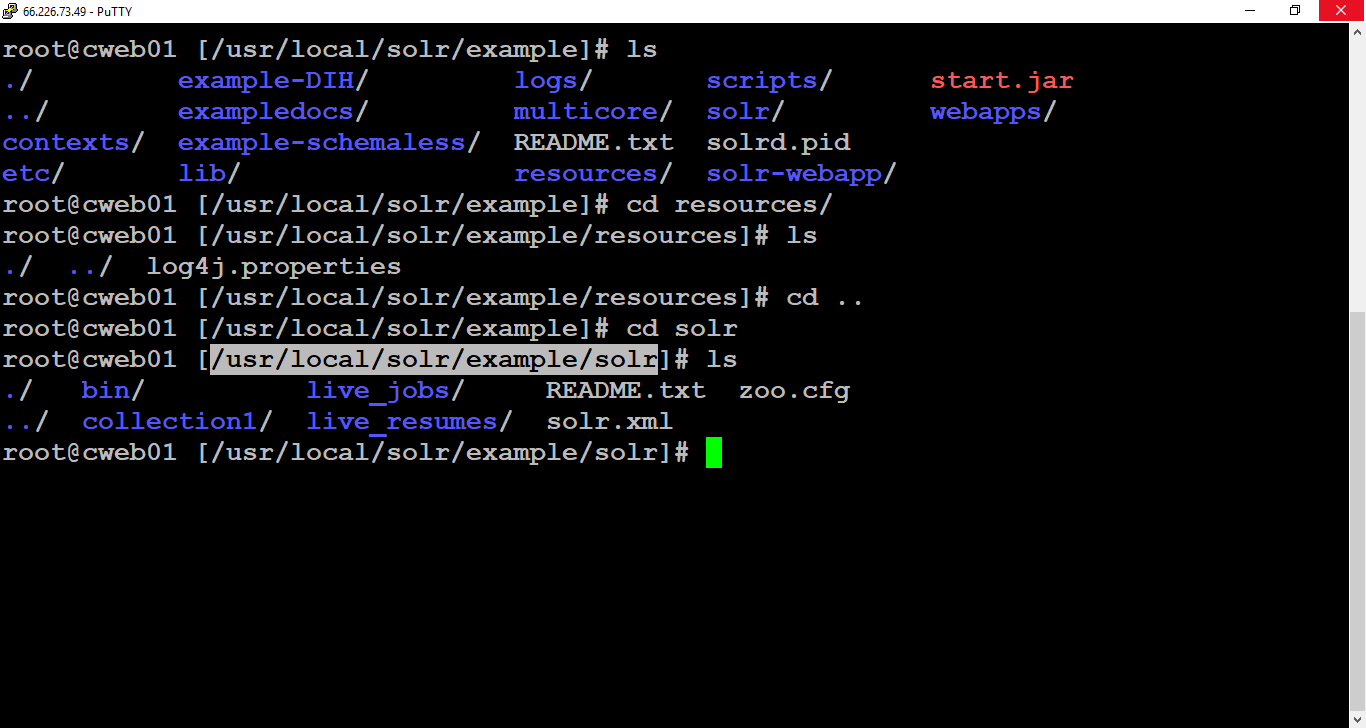
Solr directory files is under /usr/local/solr

For start & stop use bin folder

For creating projects under examples folder.



Under example/solr folder there are two projects: live\_jobs, live\_resumes.



Solr gui web link:

<http://66.226.73.49:8983/solr/>



Solr server service start, stop and restart:



Solr spec guide link:

<http://archive.apache.org/dist/lucene/solr/ref-guide/apache-solr-ref-guide-4.10.pdf>

jobsnprofiles search code model file path:

/home/jobsnprofiles/public\_html/components/com\_jobs/models/search.php

/home/jobsnprofiles/public\_html/components/com\_jobs/views/job\_search/tmpl/default.php