Deploy

Please follow the below guide to setup the AMS user

Prerequisite Setup

List of pre-requisites
1. Home Brew Installer

```
/bin/bash -c "$(curl -fsSLhttps://raw.githubusercontent.com/Homebrew/install/master/install.sh)"
```

2. brew install pass
3. brew install gh
4. brew install gpg
5. brew reinstall gnupg
6. brew install cask osxfuse
7. sudo reboot
8. brew install encfs
9. pip3 install fabric
10. pip3 install fabric
11. pip3 install fabric2
12. pip3 install fabric3
13. pip3 install boto3 --user
14. pip3 install boto3 --user
15. pip3 install pyyaml
16. pip3 install virtualenv
17. brew install pyenv-virtualenvwrapper

Installation

18. gpg --gen-key
You will see the result below

```
gpg (GnuPG) 2.2.27; Copyright (C) 2021 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Note: Use "gpg --full-generate-key" for a full featured key generation dialog.

GnuPG needs to construct a user ID to identify your key.

Real name: anju
Name must be at least 5 characters long
Real name: anjukm
Email address: anju@intelegencia.com
You selected this USER-ID:
    "anjukm <anju@intelegencia.com>"

Change (N)ame, (E)mail, or (O)kay/(Q)uit? O
We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
gpg: /Users/anju/.gnupg/trustdb.gpg: trustdb created
```

```
gpg: key 8E655BDBF93870EB marked as ultimately trusted
gpg: directory '/Users/anju/.gnupg/openpgp-revocs.d' created
gpg: revocation certificate stored as '/Users/anju/.gnupg/openpgp-revocs.d/-
A3BE6DE787F2FF7B0B4CEBB48E655BDBF93870EB.rev'
public and secret key created and signed.

pub   rsa3072 2021-03-10 [SC] [expires: 2023-03-10]
      A3BE6DE787F2FF7B0B4CEBB48E655BDBF93870EB
uid                      anjukm <anju@intelegencia.com>
sub   rsa3072 2021-03-10 [E] [expires: 2023-03-10]
```

19. gpg --export-secret-keys pd@intelegencia.com > pdprivate.key
After running above command you will see the pdprivate.key in your directory
20. add your private ke in the system.

```
gpg --export --armor anju@intelegencia.com > anjukmpubkey.asc
```

22. Git Authentication

```
gh auth login
Select Github.com
Select Web browser
Copy One time token
Launch browser
Authenticate and approve github
```

22. Follow the below link to add ssh key to git hub

```
https://docs.github.com/en/github/authenticating-to-github/generating-a-new-ssh-key-and-adding-it-
to-the-ssh-agent
```

23. Clone repo ops, server and messanger-web
24. navigate to ops repo and create your branch
25. pass init anjukm : you will see belwo result

```
Password store initialized for anjukm
```

26. Export your public key in to pubkey file so that you can upload it to git.

```
gpg --export -a anjukm > /Users/kalyaan/dev/ams/ops/sec/pass/pubkeys/anjukm.pub
```

27. Then user need to commit the changes to git and admin will sync the same to his system and run the belwo command to modyfy the public keys. so that the user can start accessing the bastion and other servers.
open directory : /ops/sec/pass and then run the below command:

```
./modify-pubkeys
```

28. Admin will commit changes after completing the step 27.
29. User will sycn the changesd repo to his system.
30. user needs to run the below command after syncing the repos. to (trust) all the pubkeys of all the users that are using the password database.

```
./import-pubkeys
```

31. pull out master password. in the ops/sec/pass directory do run the below command.

```
./cpass dev/credentials-encfs
```

above command **should show the master password for encfs**.
32. Temporarily hold this ENCFS password
33. Connect any ubuntu 18.04 machine with SSH and create new password to access the bastion servers, use below command :

```
a) apt install whois
b) mkpasswd -m sha-512
```

Abve command needs to be run 2 time for administrator. you will create 2 password for Live and all play environment access.
copy/save the password hash
34. **mount the disk** - in /dev/ams/ops/ec2

```
fab credentials.mount
```

35. then edit:

```
vi credentials/bastion_creds.json
```

I used Text Editor, Nano and VI both break the MacInCloud machines. Cureatr recommends VI
NOTE: you can not use tabs to align your brackets in bastion_creds.json

either way there are two sections: "Play" and "Live", add the user accounts as necessary to each one, copy the existing entries, update the user id and insert your password hash from step 33. **Leave the group assignment for your new entry as "cureatr". This matches an existing Linux group. any change here will break the auth processes.**

36. Push the changes to git and raise the PR and get it approved with existing admin.

37. Then notify someone who is ALREADY an admin. they will need to process the next steps listed here and admin will run this command on Live and Play bastion :

```
• fab role.auto deploy.sync_server deploy.sync_web deploy.sync_ops
• On the bastion run fab role.auto admin.ssh_setup_totp.
• Do not replace existing users TOTP when prompted.
• Provide each user with their TOTP secret from ~USER/.google_authenticator_instructions
```

You will see the result :

38.Copy each of the three URLs that are a part of this output and bring each up in a web browser.

39. open your MFA Authenticator app (I'm using Microsoft Authenticator on my phone) and add an account and scan each QR Code

40. Receive the TOTP file from another admin after incorporating your signature

41. unecrypt the TOTP file using:

```
gpg2 --decrypt totp-sumeet.txt.asc
```

40.step 38 and step 39, you can skip by sending the details directly to user.

41. looked up the passphrases for Live and RS Backup by doing:

```
in the /ops/sec/pass folder
./cpass ssh/live
```

42. looked up the passphrases for Play by doing:

```
in the /ops/sec/pass folder
./cpass ssh/play
```

43. **add the private keys to the Mac key chain**

```
in /ops/ec2 folder
fab credentials.addkeys
```

when you run the above command you will see the below result:

```
IITC-MM-02:ec2 anju$ fab credentials.addkeys
[localhost] local: chmod 400 /Users/anju/dev/ams/ops/ec2/ssh/id_rsa_vpc_live && /usr/bin/ssh-add -
K /Users/anju/dev/ams/ops/ec2/ssh/id_rsa_vpc_live
Enter passphrase for /Users/anju/dev/ams/ops/ec2/ssh/id_rsa_vpc_live: here you have to put the
passwprd from live from step 41
Identity added: /Users/anju/dev/ams/ops/ec2/ssh/id_rsa_vpc_live (aw@jello.local)
[localhost] local: chmod 400 /Users/anju/dev/ams/ops/ec2/ssh/id_rsa_vpc_play && /usr/bin/ssh-add -
K /Users/anju/dev/ams/ops/ec2/ssh/id_rsa_vpc_play
Enter passphrase for /Users/anju/dev/ams/ops/ec2/ssh/id_rsa_vpc_play:  here you have to enter the
password from play from step 42
Identity added: /Users/anju/dev/ams/ops/ec2/ssh/id_rsa_vpc_play (aw@jello.local)
[localhost] local: chmod 400 /Users/anju/dev/ams/ops/ec2/ssh/id_rsa_backup && /usr/bin/ssh-add -K /
Users/anju/dev/ams/ops/ec2/ssh/id_rsa_backup
Enter passphrase for /Users/anju/dev/ams/ops/ec2/ssh/id_rsa_backup: here you have to enter the
password of live form step : 41
Identity added: /Users/anju/dev/ams/ops/ec2/ssh/id_rsa_backup (aw@jello.local)
```

```
Done.
```

44. dit local ssh config
nano ~/.ssh/config
add content from here:
https://github.com/AMSConnect/server/blob/master/dev/ssh-config
Save your config file
You will see the result :
IITC-MM-02:ec2 anju$ ssh-add -K /Users/anju/.ssh/id_ed25519
Identity added: /Users/anju/.ssh/id_ed25519 ("anju@intelegencia.com")

45. T**ry logging into the bastion(s)**
ssh playbastion-public.play.internal.aws.amsconnectapp.com
ssh playbastion-public.play-dev1.internal.aws.amsconnectapp.com
ssh bastion1-public.live.internal.aws.amsconnectapp.com
ssh playbastion-public.play-qa2.internal.aws.amsconnectapp.com

46. Users can add readonly or root mongodb user accounts for direct database access

45.1.Add a root user fab role.auto configure.add_individual_db_user:True
45.2.Add a readonly user fab role.auto configure.add_individual_db_user
45.3.login to database from any db server by running # /site/mongoauth USER

46. TMUX Session Brief and Creation of shared pass

```
Admin will run this command :then: tmux -S /tmp/tmux/pair attach

enter passphrase: 'It will prompt the existing Admin for their existing passphrase - this is the
shareserver passphrase that was entered by the Admin previous when they were setup. This is common
between Live and Play



Enter a New passphrase: this is a new pass phrase you will use for live and play so enter a secure
passphrase. It must begin with a letter not already used by an existing password so don't start it
with a, b or z. (this list has now grown as we have added additional admins), additionally it's not
case sensitive, so A and a are the same as far as the restriction for starting with a letter
goes. After that we will need to add another one for ops: USE the same passphrase for both" This is
a live screen share where the other admin is having you type your password into the shared account
for your use. They will prompt you to add the passphrase 4 times

this is the serverside steps from: http://github.com/AMSConnect/ops/blob/-
0c1ce79b6155ea710fd16c628d4c5e414ee2e59a/ec2/docs/users.md#aws
```

```
Admin run the following command on the bastion 1 :
admin needs to run this command from his user directory
cd /home/pd/
git clone git@github.com:AMSConnect/protected-live.git
then after run the below mentioned command:
mkdir /tmp/tmux-anju (you can keep only tmux or you can add user it up to you. to keep you aware in
which folder you have to work.)
chgrp cureatr /tmp/tmux-anju
chmod g+ws /tmp/tmux-anju
then come to created directory
cd /tmp/tmux-anju/
run the below command to create the session in tmux
tmux -S /tmp/tmux-anju/pair : then after admin will be in to thux session and needs to run the
below mentioned command in  D:
and the same time then after new user will run the below mentioned command from his user directory :
```

```
so if anju is the user, he need to go in to the below mentioned directory:
cd /home/anju
then then run the below command to attach the session :
tmux -S /tmp/tmux-anju/pair attach

D: then admin needs to run the below command from tmux-anju folder :
cd /home/re/cureatr_server
PYTHONPATH=. tools/protected.py --share ~/protected-live/protected_web
PYTHONPATH=. tools/protected.py --share ~/protected-live/protected_ops


process should be like :
(venv) pd@bastion1:/tmp/tmux-anju$ cd /home/re/cureatr_server/
[tags/deploy/live/1603809159^0] (venv) pd@bastion1:/home/re/cureatr_server$ PYTHONPATH=. tools/-
protected.py --share ~/protected-live/protected_ops
Share passphrase: admin share passphrase
Adding new share
New share username: [pd]   hit enter button
New share passphrase: new user will put his password
New share passphrase (again):  again new user will put his password
[tags/deploy/live/1603809159^0] (venv) pd@bastion1:/home/re/cureatr_server$
[tags/deploy/play-qa2/1612205033^0] (venv) pd@playbastion:/home/re/cureatr_server$ PYTHONPATH=.
tools/protected.py --share ~/protected-live/protected_ops

Share passphrase: existing admin will enter his password




Adding new share

New share username: [pd] hit enter button

New share passphrase: new user password

this share pass phrase used at the time of creating the ams users and live environment deployment
time.
```

46. on every mac reboot you need to do
ssh-add -A

47. Port mapping to test Nagios and Kibana:server/dev/portmap-env.sh live in mac laptop

NOTE:- Please make sure you have chrome browser to access these localhost

then goto http://localhost:8001 for live for nagios

(Get password of nagios by running command in ops/sec/pass path $./cpass Dev/nagios will display password and username)

http://localhost:5601 for kibana live
for play its on 28001 and 25601 port
for play-staging 45601 and 48001
nagios will ask you to authenticate
kibana will not

Note : if getting error for ssh into bastion like Hostname domain is too long for unix daemon or domain get latest config from " server/☐ dev/ssh-config "

48. **DB BACKUP**
Rsync Backup :- https://www.rsync.net/
DB Backup Day And Time :- Sunday 11:30pm EST
yes, see https://github.com/AMSConnect/ops/blob/master/ec2/fabfile/admin.py#L338 - you run `fab role.auto admin.db_secondary_host admin.db_backup_dump` each week to backup to rsync.net. And also see https://github.com/☐ AMSConnect/ops/blob/master/.github/workflows/backup.yml which backs up git repos to rsync.net

49. Install docker

brew install docker

50. User access part in AWS - Users.md

```
python3 -m venv venv
 source venv/bin/activate
 Navigate to ops/ec2
  ./bootstrap-fab.sh
  venv/bin/fab credentials.update_aws_credentials:ALIAS
  alias you have to enter the alias name.
  then commit the same in the git.
 git branch
 git add .
 git commit -m "Added Anju Key into awscredentials.json"
 git push origin anju
```

51. How to give access to an user in newly created environment.

52.