MOSIP Sandbox-v2 v1.1.2 On-Premise Deployment Guide

by

CyLab-Africa

This guide is based on the official MOSIP deployment instructions and adapted from this and this cloudlab deployment guides. We recommend that you skim through the official guide to gain context before following this deployment guide.

Note:

A. This guide assumes that the installation should not be internet-facing and that it can only be accessed over VPN or the internal network.
B. It also assumes that self-signed SSL/TLS certificates will be used.

1. Hardware Setup

a. Create 7 Virtual Machines (VMs) and install CentOS 7 on all of them.
    i. The VMs should be created with the following compute resources:

Component	Number of VMs	Configuration	Storage
Console	1	4 VCPU*, 16 GB RAM	128 GB SSD*
K8s MZ master	1	4 VCPU, 8 GB RAM	32 GB SSD
K8s MZ workers	3	4 VCPU, 16 GB RAM	32 GB SSD
K8s DMZ master	1	4 VCPU, 8 GB RAM	32 GB SSD
K8s DMZ workers	1	4 VCPU, 16 GB RAM	32 GB SSD

* VCPU: Virtual CPU

*SSD: Solid State Drive

b. Assign the following hostnames to your VMs using the command: sudo hostnamectl set-hostname <hostname>
    i. console.sb
    ii. mzmaster.sb
    iii. mzworker0.sb
    iv. mzworker1.sb

v. mzworker2.sb
vi. dmzmaster.sb
vii. dmzworker0.sb

c. Enable Internet connectivity on all machines.

2. Setting up the machine environments for MOSIP

a. Create a new user on the console machine
   i. Connect to the shell of the console machine.
   ii. Create the mosipuser account
      1. `sudo useradd mosipuser`
      2. `sudo passwd mosipuser`
   iii. Add mosipuser to the sudoers
      1. `sudo usermod -aG wheel mosipuser`
   iv. Open the sudoers file using
      1. `sudo visudo`
   v. And append these lines to it to give mosipuser unlimited access and prevent applications to prompt for a password
      1. `mosipuser ALL=(ALL) ALL`
      2. `%mosipuser ALL=(ALL) NOPASSWD:ALL`

3. Give the user ssh permissions as root to all other machines

a. Keep running these commands on the console machines:
   i. Switch to the mosipuser account using the password you created for it.
      1. su - mosipuser
   ii. Generate the ssh keys (just tap return three times)
      1. ssh-keygen -t rsa
   iii. And copy the ssh public key to clipboard manually (ctrl+c; copying using mouse or browser causes issues later on when pasting, so don't do it!)
      1. cat .ssh/id_rsa.pub
   iv. Store the ssh keys in the authorized keys of the console VM and the root of all other VMs:
      1. Run this on the console machine and add the copied ssh public key to the file
         a. nano .ssh/authorized_keys
      2. Then change the permissions
         a. chmod 644 .ssh/authorized_keys
      3. Run this on all other machines and add the copied ssh public key to the file
         a. sudo nano /root/.ssh/authorized_keys

v. Test if the ssh keys were shared correctly by running the below commands on the console machine.
    1. ssh mosipuser@console
    2. ssh root@[all other hosts]

4. Disable the firewall and set time to UTC on all machines

a. Run the below commands on all machines to disable their firewall
    i. sudo systemctl stop firewalld
    ii. sudo systemctl disable firewalld
b. Set the date and time of the VMs to the correct UTC time
    i. sudo yum install ntp ntpdate -y && sudo systemctl enable ntpd && sudo ntpdate -u -s 0.centos.pool.ntp.org 1.centos.pool.ntp.org 2.centos.pool.ntp.org && sudo systemctl restart ntpd && sudo timedatectl

5. Installing dependencies and downloading the MOSIP repo

a. Follow these instructions on the Console VM:
    i. Install Git
        1. sudo yum install -y git
    ii. Clone the mosip-infra repo and switch to the appropriate branch
        1. $ cd ~/
        2. $ git clone https://github.com/mosip/mosip-infra
        3. $ cd mosip-infra
        4. $ git checkout 1.1.2
        5. $ cd mosip-infra/deployment/sandbox-v2
    iii. Change ownership of the cloned repo (if not the owner)
        1. sudo chown -R mosipuser mosip-infra/
    iv. Install Ansible and create shortcuts:
        1. ./preinstall.sh
        2. source ~/.bashrc

6. Configuring and Installing MOSIP

a. Update hosts.ini as per your setup. Make sure the machine names and IP addresses match your setup.
b. Follow these instructions on the Console VM
    i. Open group_vars/all.yml using nano mosip-infra/deployment/sandbox-v2/group_vars/all.yml and replace the following values as below:

```
sandbox_domain_name: '{{inventory_hostname}}'
site:
  sandbox_public_url: 'https://{{sandbox_domain_name}}'
  ssl:
    ca: 'selfsigned'   # The ca to be used in this deployment
```

ii. Open both the files below

nano mosip-infra/deployment/sandbox-v2/group_vars/mzcluster.yml

nano mosip-infra/deployment/sandbox-v2/group_vars/dmzcluster.yml

and replace the value of network_interface found in both files with 'enp0s3' or the configured network interface on the CentOS VMs.

iii. Run the ansible scripts that will install MOSIP
    1. cd mosip-infra/deployment/sandbox-v2/
    2. an site.yml

iv. The main MOSIP web interface can be accessed by typing the console VMs IP address / hostname into a web browser.

v. To access the Pre-Registration UI after the installation is complete, use the below link:
    1. <your console hostname>/pre-registration-ui
    2. To avoid issues with the pre-registration page not loading properly. Make sure you access the page using the domain name of the console VM and not its IP Address. If you do not have a DNS server on your network to translate the console VM domain name to its IP Address, you can add a static DNS mapping of the console machine's domain name and IP Address on your machine. In Linux/MAC, this mapping can be done in the `/etc/hosts` file. In Windows this can be done in the `C:\Windows\System32\drivers\etc\hosts` file

vi. While testing:
    1. You can connect to it using OTP and the static OTP value: `111111`
    2. You can use this fake valid postal code: `14022`

7. Ansible vault

a. All secrets (passwords) used by the MOSIP installation are stored in Ansible vault file secrets.yml. The default password to access the file is 'foo'. It is recommended that you change this password with following command:
    i. av rekey secrets.yml

b. You may view and edit the contents of secrets.yml:
    i. av view secrets.yml
    ii. av edit secrets.yml

8. Windows Registration Client Setup

a. Go through the official MOSIP Guide located here: `https://docs.mosip.io/platform/modules/registration-client/registration-client-setup` to familiarize yourself with the registration client functionality and installation process.

b. Set "mosip.hostname" environment variable on your machine with the host name of the console VM.
c. On the console VM, copy the maven-metadata.xml file from /home/mosipuser/mosip-infra/deployment/sandbox-v2/roles/reg-client-prep/templates/ to /usr/share/nginx/html/
d. Login to the console VM and change the configs of the file: /home/mosipuser/mosip-infra/deployment/sandbox-v2/tmp/registration/registration/registration-libs/src/main/resources/props/mosip-application.properties to the below configuration:

```
mosip.reg.healthcheck.url=https\://<console VM
hostname>/v1/authmanager/actuator/health
mosip.reg.rollback.path=../BackUp
mosip.reg.cerpath=/cer//mosip_cer.cer
mosip.reg.db.key=bW9zaXAxMjM0NQ\=\=
mosip.reg.xml.file.url=https\://<console VM hostname>/maven-
metadata.xml
mosip.reg.app.key=bBQX230Wskq6XpoZ1c+Ep1D+znxfT89NxLQ7P4KFkc
4\=
mosip.reg.client.tpm.availability=N
mosip.reg.env=qa
mosip.reg.dbpath=db/reg
mosip.reg.logpath=../logs
mosip.reg.mdm.server.port=8080
mosip.reg.version=1.1.2-rc2
mosip.reg.packetstorepath=../PacketStore
mosip.reg.client.url=https\://console VM
hostname/registration-client/1.1.2/reg-client.zip
```

e. Download the client zip file from https://<your console hostname>/registration-client/1.1.2/reg-client.zip
f. Unzip the downloaded client
g. Execute the run.bat file inside the unzipped folder.
h. Once the above file is execute, certain keys are generated and stored under this file: C:\Users\<Your User Name>\.mosipkeys\readme
i. Copy the machine name, public key, and key index values together with other details about your machine such as MAC Address, Serial Number, and IP address and append them to this file: /home/mosipuser/mosip-infra/deployment/sandbox-v2/tmp/commons/db_scripts/mosip_master/dml/master-machine_master.csv located on the MOSIP console VM.
j. cd to /home/mosipuser/mosip-infra/deployment/sandbox-v2/test/regclient and run the script: ./update_masterdb.sh

`/home/mosipuser/mosip-infra/deployment/sandbox-v2/tmp/commons/db_scripts/mosip_master`

k. After doing the above, you can login to the Windows client using the username `11011` and password `mosip`. You will see an application restart prompt. Close the application and rerun the run.bat file. This time, login with the username `110118` and password `Techno@123`

9. Appendix - Known Installation Issues

a. Error 1

i. Output

```
TASK [k8scluster/cni : Create flannel network daemonset]
*********************************************************************
************************

fatal: [dmzmaster.sb -> 172.29.108.22]: FAILED! => {"changed": true, "cmd":
["kubectl", "apply", "--kubeconfig=/etc/kubernetes/admin.conf", "-f",
"/etc/kubernetes/network/"], "delta": "0:00:00.083852", "end": "2021-04-27
13:42:54.099146", "msg": "non-zero return code", "rc": 1, "start": "2021-04-27
13:42:54.015294", "stderr": "The connection to the server 172.29.108.22:6443
was refused - did you specify the right host or port?", "stderr_lines": ["The
connection to the server 172.29.108.22:6443 was refused - did you specify the
right host or port?"], "stdout": "", "stdout_lines": []}
```

ii. Fix

Run the following commands on dmzmaster.sb (or on the node where the error happened):

- systemctl stop docker && systemctl stop kubelet
- kubeadm reset
- rm -rf /etc/cni/net.d
- rm -rf $HOME/.kube/config

b. Error2

i. Output

```
TASK [packages/helm-cli: Add stable repo]

fatal: [console]: FAILED! => {"changed": true, "cmd":
"/home/mosipuser/bin/helm repo add stable https://kubernetes-
charts.storage.googleapis.com", "delta": "0:00:00.238782", "end": "2021-04-21
09:54:07.660850", "msg": "non-zero return code", "rc": 1, "start": "2021-04-21
09:54:07.422068", "stderr": "Error: looks like \"https://kubernetes-
charts.storage.googleapis.com\" is not a valid chart repository or cannot be
reached: failed to fetch https://kubernetes-
charts.storage.googleapis.com/index.yaml : 403 Forbidden", "stderr_lines":
["Error: looks like \"https://kubernetes-charts.storage.googleapis.com\" is not a
valid chart repository or cannot be reached: failed to fetch https://kubernetes-
charts.storage.googleapis.com/index.yaml : 403 Forbidden"], "stdout": "",
"stdout_lines": []}
```

ii. Fix

Refer to: https://stackoverflow.com/a/65404574/15117449

In roles/packages/helm-cli/tasks/main.yml, replace the stable repo https://kubernetes-charts.storage.googleapis.com with https://charts.helm.sh/stable

c. Error 3

i. Output

TASK [packages/crypto : Install python3 cryptography]

fatal: [console]: FAILED! => {"changed": false, "cmd": ["/bin/pip3", "install", "cryptography"], "msg": "stdout: Collecting cryptography\n Downloading https://files.pythonhosted.org/packages/9b/77/461087a514d2e8ece1c975d8216bc03f7048e6090c5166bc34115afdaa53/cryptography-3.4.7.tar.gz (546kB)\n   Complete output from command python setup.py egg_info:\n   \n   ============================DEBUG ASSISTANCE==========================\n   If you are seeing an error here please try the following to\n   successfully install cryptography:\n   \n   Upgrade to the latest pip and try again.\n   This will fix errors for most\n   users. See: https://pip.pypa.io/en/stable/installing/#upgrading-pip\n   ============================DEBUG ASSISTANCE==========================\n\n   Traceback (most recent call last):\n   File \"<string>\", line 1, in <module>\n   File \"/tmp/pip-build-ifd1g9v2/cryptography/setup.py\", line 14, in <module>\n   from setuptools_rust import RustExtension\n   ModuleNotFoundError: No module named 'setuptools_rust'\n   \n   --\n\n:stderr: WARNING: Running pip install with root privileges is generally not a good idea. Try `pip3 install --user` instead.\nCommand \"python setup.py egg_info\" failed with error code 1 in /tmp/pip-build-ifd1g9v2/cryptography/\n"}

ii. Fix

pip3 install --upgrade pip

pip3 install cryptography

a. .source /home/mosipuser/.venv-py3/bin/activate

b. python3 -m pip install setuptools_rust

c. pip install --upgrade pip

d. python3 -m pip install certbot

e. Deactivate

d. Error 4

i. Output

TASK [k8scluster/kubernetes/master : Init Kubernetes cluster]

fatal: [mzmaster]: FAILED! => {"changed": true, "cmd": "kubeadm init --service-cidr 10.96.0.0/12 --kubernetes-version v1.19.0 --pod-network-cidr 10.244.0.0/16 --token b0f7b8.8d1767876297d85c --apiserver-advertise-address 172.17.33.3 \n", "delta": "0:00:00.476187", "end": "2021-04-21 11:09:42.787299", "msg": "non-zero return code", "rc": 1, "start": "2021-04-21 11:09:42.311112", "stderr": "this version of kubeadm only supports deploying clusters with the control plane version >= 1.20.0. Current version: v1.19.0\nTo see the stack trace of this error execute with --v=5 or higher", "stderr_lines": ["this version of kubeadm only supports deploying clusters with the control plane version >= 1.20.0. Current version: v1.19.0", "To see the stack trace of this error execute with --v=5 or higher"], "stdout": "", "stdout_lines": []}

ii. Fix

Replace package names with package-name-1.19.0 in roles/k8scluster/kubernetes/node/meta/main.yml and roles/k8scluster/kubernetes/master/meta/main.yml, e.g., "kubeadm-1.19.0", then add allow_downgrade: true to the apt section of RHEL/Centos pkg install in roles/k8scluster/commons/pre-install/tasks/pkg.yml

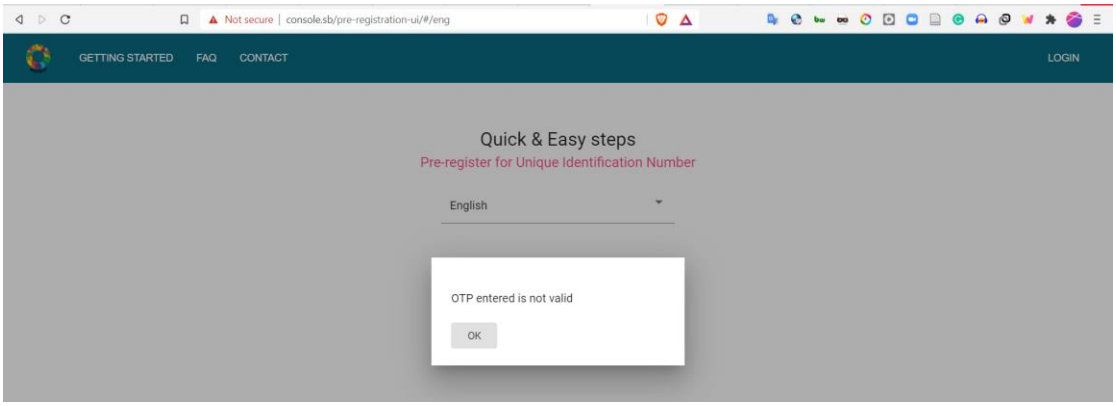
e. Error 5

i. The admin helm release fails to deploy.

ii. Fix: The docker image version for the admin playbook is incorrect. Find the relevant docker images in versions.yml, replace 1.1.3 with 1.1.2

f. Error 6

i. Default OTP of 111111 is not valid on the pre-registration-ui as shown below



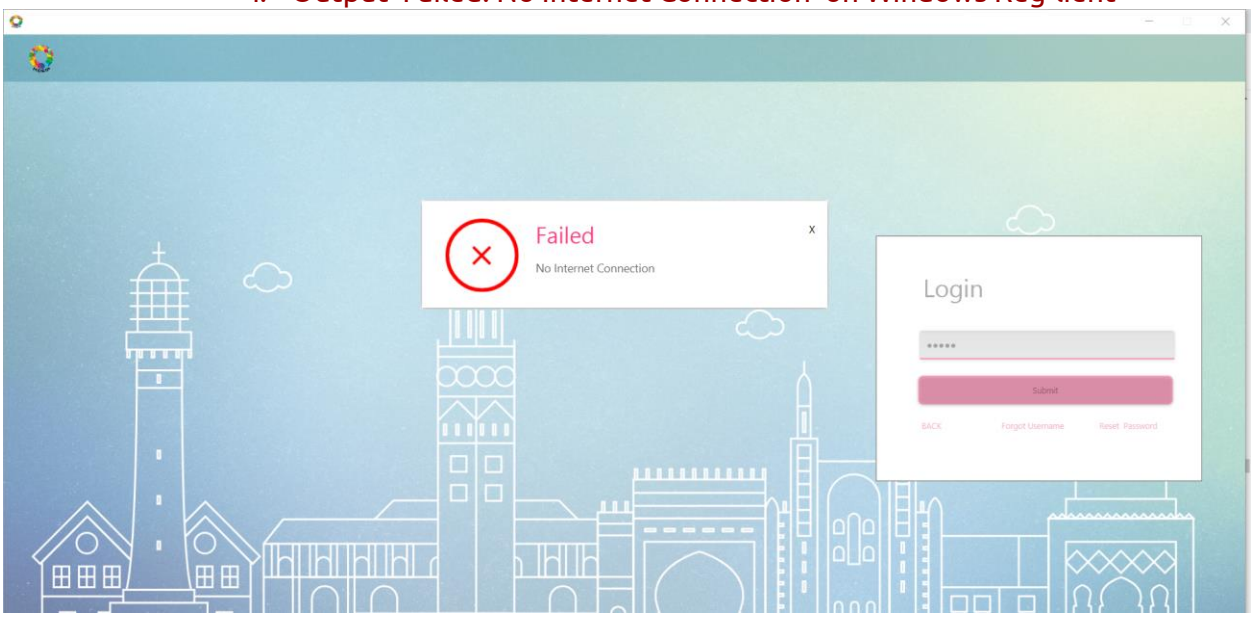
ii. Fix

1. Make sure all the VM clocks are synchronized and set to the correct UTC date and time.

2. If the above does not work, Reinstall the Keycloak helm release by running helm1 delete keycloak and then an playbooks/keycloak.yml

g. Error 7

i. Output 'Failed: No Internet Connection' on Windows Reg-lient

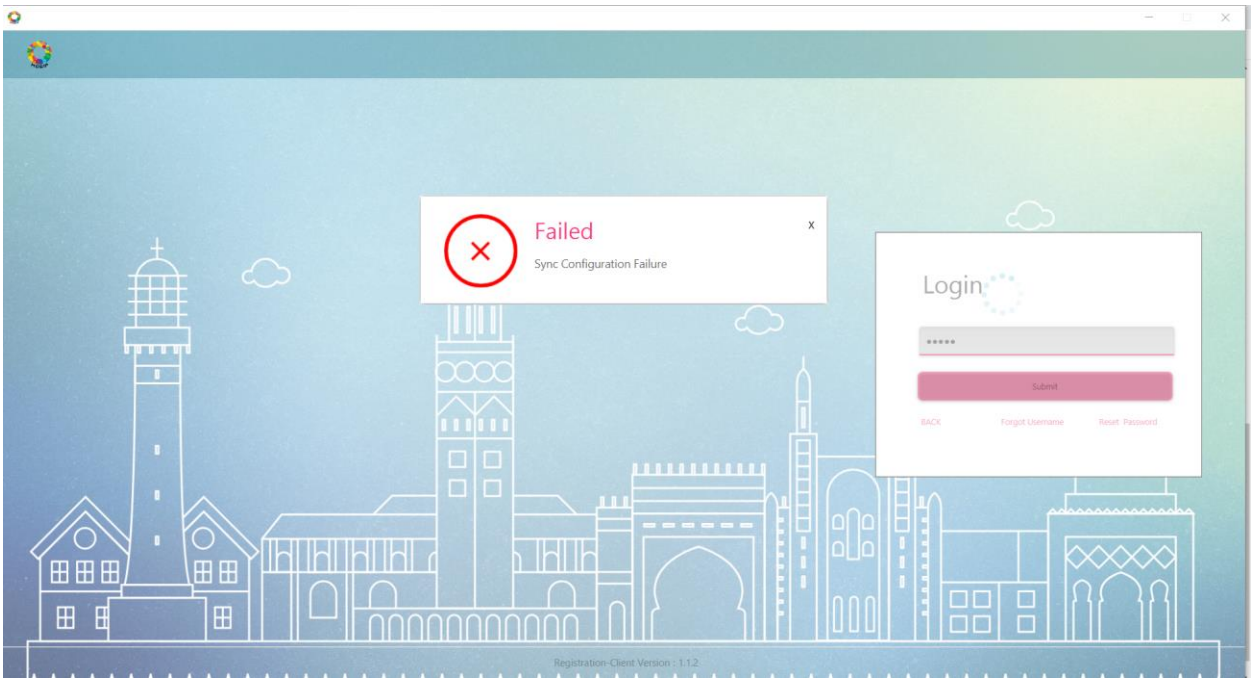


ii. Fix:

Generate a new self-signed certificate for nginx and adding `console.sb` as the certificate's `Common Name (CN)`. The reason being, by default, MOSIP uses the server's IP address as the `CN` when it is generating the self-signed certificate and as mentioned here: https://stackoverflow.com/questions/29157861/java-certificateexception-no-subject-alternative-names-matching-ip-address and here: https://web.archive.org/web/20160201235032/http://www.jroller.com/hasant/entry/no_subject_alternative_names_matching , JAVA has issues with using an IP address as a CN in certificates. Here is a link on how to generate a self-signed certificate for nginx: https://www.digitalocean.com/community/tutorials/how-to-create-a-self-signed-ssl-certificate-for-nginx-on-centos-7. This should not be an issue when using a CA-issued certificate since this is issued to the domain name registered under MOSIP and not the IP address.

h. Error 8

i. Output: Failed: Sync Configuration Failure



ii. Thi is related to your machine details not added to the `mosip_master` database. Add your machine details in the `master-machine_master.csv` file and ran the `update_masterdb.sh` script to update the details in the database. The reg-client application should restart and you should be able to login with the user `110118` and Password `Techno@123`.