Experiment No.1.8

Student Name: Gaurav Kumar   **UID:** 22MCC20177
Branch: MCA–CCD   **Section/Group:** MCD-1/A
Semester: III   **Date of Performance:** 06th Nov 23
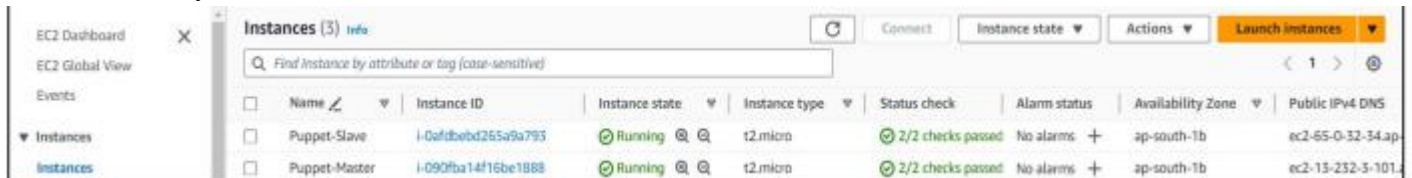**Subject Name:** DevOps Process Automation Lab   **Subject Code:** 22CAP-745

1. Aim/Overview of the practical:

a)   Install and configure Puppet (Master and Slave).
b)   Generate, Send and Sign SSL certificate by Puppet Slave and Verify SSL Certificate on Puppet Master.

2. Steps for practical: (a)

Step 1 :   Go to AWS and Create two Ec2 instances (I'm using Ubuntu 20.04) and Access both by Putty.

Step 2 :   Once both machines are accessed, First go to Puppet-Master and download the Puppet package.
`wget` https://apt.puppetlabs.com/puppet-release-bionic.deb.

Step 3 :   Now unpack the package `sudo dpkg -I puppet-release-bionic.deb`.

```
root@ip-172-31-12-227: ~                                          —   □   ×

root@ip-172-31-12-227:~# sudo dpkg -i puppet-release-bionic.deb
Selecting previously unselected package puppet-release.
(Reading database ... 62002 files and directories currently installed.)
Preparing to unpack puppet-release-bionic.deb ...
Unpacking puppet-release (1.0.0-28bionic) ...
Setting up puppet-release (1.0.0-28bionic) ...
root@ip-172-31-12-227:~#
```

Step 4 :   Install the Puppet package `sudo apt-get install puppetmaster`.

```
root@ip-172-31-12-227: ~                                          —   □   ×

root@ip-172-31-12-227:~# sudo apt-get install puppetmaster
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  augeas-lenses debconf-utils facter fonts-lato hiera javascript-common libaugeas0
  libboost-filesystem1.71.0 libboost-locale1.71.0 libboost-log1.71.0
  libboost-program-options1.71.0 libboost-regex1.71.0 libboost-thread1.71.0 libcpp-hocon0.1.7
  libfacter3.11.0 libjs-jquery libleatherman1.4.2 libruby2.7 libyaml-cpp0.6 puppet
  puppet-master rake ruby ruby-augeas ruby-deep-merge ruby-minitest ruby-net-telnet
  ruby-power-assert ruby-selinux ruby-shadow ruby-sync ruby-test-unit ruby-xmlrpc ruby2.7
  rubygems-integration unzip zip
```

Step 5 :   Edit puppet master configuration file.
`sudo vim /etc/default/puppet-master Add JAVA_ARGS="-Xms512m -Xmx512m".`

```
root@ip-172-31-12-227: ~                                          —   □   ×

root@ip-172-31-12-227:~# sudo vim /etc/default/puppet-master
```

```
root@ip-172-31-12-227: ~                                          —   □   ×

# Defaults for puppetmaster - sourced by /etc/init.d/puppet-master
JAVA_ARGS="-Xms512m -Xmx512m"
# Start puppetmaster on boot?
START=yes

# Startup options.
DAEMON_OPTS=""
~
```

Step 6 :   Enable and start or restart the puppet-master services.
`sudo systemctl restart puppet-master.service` also allow 8140/tcp connection.

Step 7 :   Now go to Puppet-Slave machine and download the Puppet package.
`wget https://apt.puppetlabs.com/puppet-release-bionic.deb`

Step 8 :   Now Unpack the package and install puppet.
`sudo apt-get install puppet`

Step 9 :   Go to hosts file of Puppet-Master machine and config IP address of Puppet-Master machine.

Step 10 :   Enable and start the puppet services.
`sudo systemctl start puppet.`
`sudo systemctl enable puppet.`

Step 11 :   Your Puppet-Master and Puppet-Slave machine are configured till now.
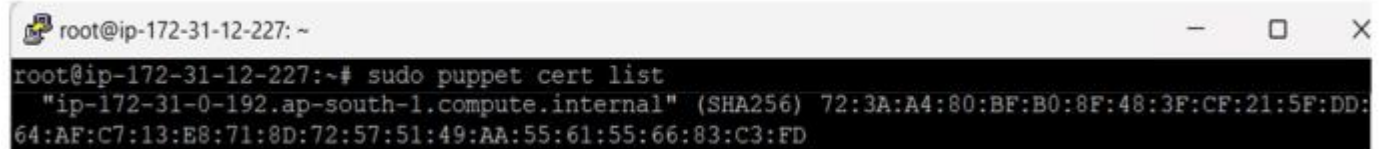
UNIVERSITY INSTITUTE of COMPUTING
Asia's Fastest Growing University

NAAC GRADE A+
ACCREDITED UNIVERSITY

CU
CHANDIGARH
UNIVERSITY

3. **Steps for practical: (b)**

   Generate, Send and Sign SSL certificate by Puppet Slave and Verify SSL Certificate on Puppet Master.

   Step 1 : Once Puppet-Master and Puppet-Slave machine are configured and connected Puppet-slave machine generate and send the SSL certificate to Puppet-Master certificate.

   Step 2 : Now go to Puppet-Master machine and list the SSL certificate.
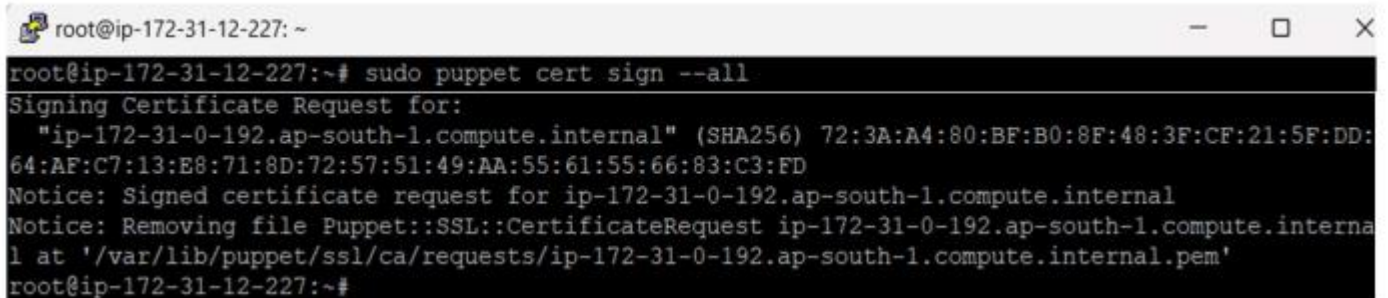
   ```
   sudo puppet cert list
   ```

   ```
   root@ip-172-31-12-227: ~                                          —   □   ×
   root@ip-172-31-12-227:~# sudo puppet cert list
    "ip-172-31-0-192.ap-south-1.compute.internal" (SHA256) 72:3A:A4:80:BF:B0:8F:48:3F:CF:21:5F:DD:
   64:AF:C7:13:E8:71:8D:72:57:51:49:AA:55:61:55:66:83:C3:FD
   ```

   Step 3 : Confirm the IP address by checking the SSL certificate request.

   Step 4 : Now, Sign the SSL certificate request one by one manually or all (here I'm sign all certificate by –all flag).

   ```
   sudo puppet cert sign -all
   ```

   ```
   root@ip-172-31-12-227: ~                                          —   □   ×
   root@ip-172-31-12-227:~# sudo puppet cert sign --all
   Signing Certificate Request for:
    "ip-172-31-0-192.ap-south-1.compute.internal" (SHA256) 72:3A:A4:80:BF:B0:8F:48:3F:CF:21:5F:DD:
   64:AF:C7:13:E8:71:8D:72:57:51:49:AA:55:61:55:66:83:C3:FD
   Notice: Signed certificate request for ip-172-31-0-192.ap-south-1.compute.internal
   Notice: Removing file Puppet::SSL::CertificateRequest ip-172-31-0-192.ap-south-1.compute.interna
   l at '/var/lib/puppet/ssl/ca/requests/ip-172-31-0-192.ap-south-1.compute.internal.pem'
   root@ip-172-31-12-227:~#
   ```

   Step 5 : SSL certificate signed successfully.