

①

> puppet cert sign --all → signed certificate & now nodes are able to

> Simple Certificates to communicate with node
(host level certificate) [comm. level certificate]

→ what node to add & what node to Remove - all depends on certificate signing.

* vi manifest.pp (puppet manifest)

file { 'tmp/helloavi': // resources is file ()

ensure => present,

mode => '0644',

content => "It ... { yln", # Print the eth0 IP facts.

* Node:

cd tmp/helloavi (Node waits 30 minutes)

↳ puppet agent -t (-t means test)

> 'production'

> 'plogfacts'

> 'plugin'

/opt/etc/puppet/code/

environment/

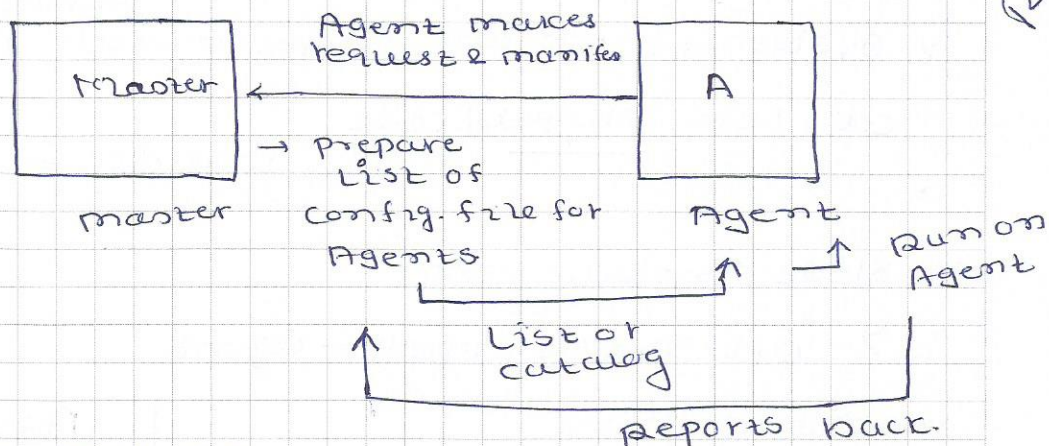
products/manifest.pp

Caching catalog for ip-172 (instruction for this node & applied)
APPY configuration '1587434704'

↳ vi site.pp (master file → Node - keep updating
exec { 'apt-update': — this file.]

* catalog → master & Agent communication take place.

[Keep on updating on node level, but keep only one file.]



* time Interval on Node level.

> /etc/puppet/puppet.cfg

ssldir = /var/lib/puppet/ssl

runinterval = 1m (Run test every one minute)

> systemctl restart puppet.

> apt-get remove apache2

> apt install rpm (Red-hat Package manager)

> puppet-master → www.log.

```
> node 'nodename' {
  class {'linux'}
}
```

ip address or
node name.

{ apt-get remove apache2 }

```
class linux {
```

```
}
```

> service apache2 start.

> service apache2 status.

>

/var/log/puppet → master http.log
(log)

```
$ soft = ['git', 'vim']
```

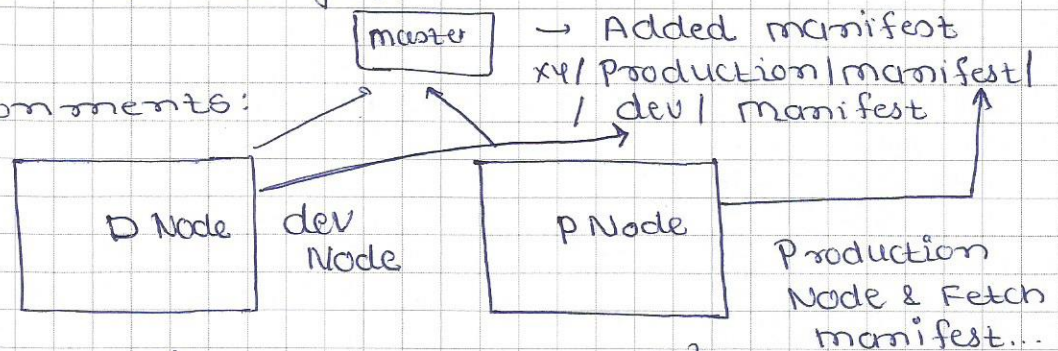
soft has a multiple option inside it.

> more puppet.conf

> apt-get remove git

> (15)

* Environments:



→ { production node & Dev. node }

* puppet.conf.

environment = dev.

} master
> mkdir dev
> cd dev
> vi newmanifest.pp

> environment, conditions, Loops.

* Puppet module & Roles in Ansible.

↳ [Like Roles in a Ansible.]

→

* terraform → cloud work on → kind of management

→ terraform } Hashicorp
test

```
main.tf
variable "myvar" {
```

```
}
```

→ Procure Server on cloud.
Ansible - copy services & all.
installing software.