**Puppet with Foreman [ Configuration Management Tool & Lifecycle Management Tool ]**

### **Working of Puppet with Foreman ( Puppet Server, Puppet Agent, PuppetDB, Puppet Board )**

1. **Initial Setup**
   * Puppet Server (Puppet Master) is installed on a central server.
   * Puppet Agent is installed on nodes (client machines) that need to be managed.
   * Foreman is used for node classification and provisioning, providing a user-friendly interface for managing infrastructure.
   * PuppetDB is installed to store facts, catalogs, and reports from Puppet runs.
   * Puppet Board provides a web interface for visualizing data stored in PuppetDB.
2. **Puppet Agent to Puppet Master Communication**
   * The Puppet Agent installed on a client node sends a Certificate Signing Request (CSR) to the Puppet Master.
   * The Puppet Master manually signs the CSR, allowing the agent to authenticate.
   * The agent sends facts (node attributes like OS, memory, hostname) to the Puppet Master during its first run.
3. **Catalog Compilation**
   * The Puppet Master receives the facts from the Puppet Agent and compiles a catalog.
   * The catalog contains the configuration details and desired state for the node.
   * Foreman is used to assign Puppet classes to nodes based on their facts. These facts are then retrieved from puppetDB. Foreman interacts with the Puppet Master to manage which classes should be applied to which nodes.
   * Puppet Master sends the catalog back to the Puppet Agent.
4. **Configuration Application**
   * The Puppet Agent applies the catalog to the node, ensuring the node's configuration matches the desired state.
   * For example, if a file is missing or a service is not running, the Puppet Agent will enforce the configuration to bring the node to the desired state.
5. **Puppet Agent Sends Report to Puppet Master**
   * After applying the configuration, the Puppet Agent sends a report back to the Puppet Master.
   * The report includes the status of the applied changes, any errors, and a summary of actions taken by the Puppet Agent.
6. **Storing Data in PuppetDB**
   * The Puppet Master stores the report in PuppetDB.
   * PuppetDB also stores facts about the node (e.g., OS type, architecture) and the catalog describing the desired configuration.
7. **Foreman and PuppetDB Integration**
   * Foreman can pull data from PuppetDB to provide a graphical interface for managing nodes and viewing detailed reports.
   * Foreman classifies nodes based on facts received from PuppetDB, and the configuration is managed accordingly.
   * If there are updates to the configuration (e.g., new classes assigned to nodes), Foreman will trigger a new Puppet run.
8. **Puppet Board for Viewing Data**:
   * Puppet Board connects to PuppetDB and provides a web interface for visualizing data stored in PuppetDB.
   * It shows reports of Puppet runs, including successes, failures, and detailed changes made during each run.
   * Puppet Board allows users to search and filter facts and reports based on different criteria (e.g., node name, environment, etc.).

**Steps to create Puppet architecture with Foreman**

1. **Install Puppet Master**

*apt-get install puppetserver*

1. **Configure Puppet Master and Start Puppet Master**

*[main]*

*certname = <server\_hostname>*

*server = <server\_hostname>*

*report = true*

*reports = foreman,puppetdb*

*(and)*

*systemctl start puppetserver*

1. **Install Puppet Agent**

*apt-get install puppet-agent*

1. **Configure Puppet Agent and Start Puppet Agent**

*[main]*

*certname = <agent\_certificatename>*

*server = <server\_hostname>*

*(and)*

*systemctl start puppet*

1. **Install Foreman**

*apt-get install -y foreman*

1. **Install the Foreman-Puppet plugin**

*apt-get install -y foreman-plugin-puppet*

1. **Configure Foreman**
2. **Install PuppetDB**

*apt-get install puppetdb*

1. **Configure PuppetDB (/etc/puppet/puppet.conf) on master and Start PuppetDB**

*[main]*

*storeconfigs\_backend = puppetdb*

*reports = foreman,puppetdb*

*(and)*

*systemctl start puppetdb*

1. **Install Puppetboard**

*apt-get install -y python3-pip python3-dev libpq-dev*

*pip3 install puppetboard*

1. **Configure Puppetboard (/etc/puppetboard/config.py) and Start Puppetboard**

*DATABASES = {*

*'default': {*

*'NAME': 'puppetdb',*

*'USER': '<username\_of\_postgresql)',*

*'PASSWORD': '<password>',*

*'HOST': 'localhost',*

*'PORT': '5432',*

*'ENGINE': 'django.db.backends.postgresql',*

*}*

*}*

*(and)*

*puppetboard (To start puppetboard)*

*[ http://<puppetboard-server-ip>:8000]*