**Настройка PUPPET**

Официальная документация: <https://docs.puppet.com/puppet/4.8/>

1) Настройка службы времени:

sudo apt-get -y install ntp

По желанию можно добавить свои серверы в /etc/ntp.conf и перезапустить службу

sudo service ntp restart

2) Установка репозитория и самого пакета puppetserver:

cd ~ && wget <https://apt.puppetlabs.com/puppetlabs-release-pc1-trusty.deb>

sudo dpkg -i puppetlabs-release-pc1-trusty.deb

sudo apt-get update

sudo apt-get install –y puppetserver

3) Для определения количества ОЗУ доступного серверу puppet редактируется файл /etc/default/puppetserver а именно параметры JAVA\_ARGS

4) Для указания агенту puppet на том же сервере адреса сервера редактируется файл /etc/puppetlabs/puppet/puppet.conf, а именно в него добавлется секция:

[main]

certname = devops

server = devops

5) Запускаю службы сервера и агента puppet:

sudo /opt/puppetlabs/bin/puppet resource service puppetserver ensure=running enable=true

sudo /opt/puppetlabs/bin/puppet resource service puppet ensure=running enable=true

6) Проверяю корректную работу агента:

sudo /opt/puppetlabs/bin/puppet agent --test

------

Официальное руководство: [**https://docs.puppet.com/puppet/latest/install\_linux.html**](https://docs.puppet.com/puppet/latest/install_linux.html)

7) Устанавливаю информацию о репозитории puppet и сам puppet-agent на Ubuntu (в моем случае 16, Xenial):

wget <https://apt.puppetlabs.com/puppetlabs-release-pc1-xenial.deb>

sudo dpkg -i puppetlabs-release-pc1-wheezy.deb

sudo apt-get update

sudo apt-get install puppet-agent

8) Делаю то же самое на Centos или RedHat (у меня Centos 5.11):

wget <https://yum.puppetlabs.com/puppetlabs-release-pc1-el-5.noarch.rpm>rpm -Uvh puppetlabs-release-pc1-el-5.noarch.rpm

yum install puppet-agent

9) На обоих системах прописываю в /etc/hosts инфонмацию о сервере (в моем случае 10.0.1.6 devops) и настаиваю службу точного времени аналогично шагу №1

10) Редактирую файл конфигурации puppet на агентах: /etc/puppetlabs/puppet/puppet.conf , добавляя туда сведения о сервере:

[agent]

server = devops

11) Запускаю и ставлю в автозапуск puppet на агентах командой:

sudo /opt/puppetlabs/bin/puppet resource service puppet ensure=running enable=true

12) На сервере подписываю все сертификаты:

sudo /opt/puppetlabs/bin/puppet cert sign –all

13) На клиентах проверяю связь с сервером:

sudo /opt/puppetlabs/bin/puppet agent –test

--------

14) Создаю тестовую среду (<https://docs.puppet.com/puppet/4.8/environments_creating.html> ):

cd /etc/puppetlabs/code/environments

sudo cp -R production test

sudo mv test/environment.conf test/test.conf

15) Создаю манифест для тестовой среды (<https://docs.puppet.com/puppet/latest/types/package.html> ):

sudo vim /etc/puppetlabs/code/environments/test/manifests/test.pp

Со сценарием установки пакета "tree":

package { ‘tree’:

ensure => installed,

}

16) На узлах прописываю тестовую среду по умолчанию в файле **/etc/puppetlabs/puppet/puppet.conf** (<https://docs.puppet.com/puppet/latest/configuration.html> ):

environment = test

И запускаю принудительную связь с сервером puppet:  
sudo /opt/puppetlabs/bin/puppet agent --test

---------

17) Устаналиваю пакет whois и создаю хэш пароля "12345":

sudo apt-get install whois

mkpasswd

12345

18) Переименовываю основной манифест (<https://docs.puppet.com/puppet/latest/dirs_manifest.html>) и добавляю в него код создания нового пользователя (отдельно для каждого из двух узлов):

cd /etc/puppetlabs/code/environments/test/manifests

sudo mv test.pp site.pp

sudo vim sitee.pp

Добавляю следующие строки (<https://docs.puppet.com/puppet/latest/types/user.html>):

*node 'ubsrv' {*

*user { 'engin':*

*name => 'engineer',*

*home => '/home/engineer',*

*managehome => true,*

*shell => '/bin/bash',*

*ensure => present,*

*password => 'OMbJkRx6Ocyok',*

*groups => sudo,*

*}*

*}*

*node 'cnt3.vbox' {*

*user { 'engin':*

*name => 'engineer',*

*home => '/home/engineer',*

*managehome => true,*

*shell => '/bin/bash',*

*ensure => present,*

*password => 'OMbJkRx6Ocyok',*

*groups => wheel,*

*}*

*}*

--------

19) Создаю конфиг файлового сервера (файл **/etc/puppetlabs/puppet/fileserver.conf** по мануалу <https://docs.puppet.com/puppet/latest/config_file_fileserver.html>) с текстом:

*[files]*

*path /etc/puppetlabs/code/files*

*allow \**

20) Создаю новый каталог

mkdir -p /etc/puppetlabs/code/files/test

21) В нем создаю файл **.bashrc** со следующим содержимым:

*# managed by puppet*

*echo "$(tput setaf 2)====== Привет ! ======"$(tput sgr 0)*

*echo "Точное время: $(date)"*

*echo "Место на дисках:"*

*df -h | sed -n '/^\/dev/p'*

*echo "Имя хоста: $(tput setaf 3)$(hostname -f)$(tput sgr 0)"*

22) Дописываю в главный манифест **/etc/puppetlabs/code/environments/test/manifests/site.pp** текст (<https://docs.puppet.com/puppet/latest/types/file.html>):

*file { 'bashrc':*

*ensure => file,*

*path => '/home/engineer/.bashrc',*

*source => 'puppet:///files/test/.bashrc',*

*owner => 'engineer',*

*group => 'engineer',*

*mode => '0644',*

*}*

--------