UNIVERSIDAD LUTERANA SALVADOREÑA
FACULTAD DE CIENCIAS DEL HOMBRE Y LA NATURALEZA
LICENCIATURA EN CIENCIAS DE LA COMPUTACIÓN

ESPECIALIZACIÓN ADMINISTRACIÓN DE SERVIDORES 2017

"CONFIGURACIÓN DE UN SERVIDOR PUPPET"

N°	APELLIDOS	NOMBRES	CARNET
1	CARRANZA MUNGUIA	CRUZ AZAELA	CM01121584
2	MANZANARES QUIJANO	JUAN FRANCISCO	MQ02110994

DOCENTE: ING. MANUEL VILLATORO FLORES

San Salvador, 20 de Enero del 2018.

ÍNDICE

Índice de Imágenes

INTRODUCCIÓN

El presente documento contiene todos los procesos y caminos que se tomaron para poder desarrollar el proyecto de configuración de Puppet proyecto elegido en la especializaron de la carrera de licenciatura en ciencias de la computación.

Se demuestra en este documento la importancia de un servidor como Puppet ya que nos ayuda a gestionar y tener el control sobre varios ordenadores y agilizar procesos ahorrando tiempo en poder administrar de un nodo a la vez. En el contenido de este documento esta plasmado las alternativas para poder darle un mejor uso al software y controlar mediante ordenes los ordenadores.

El proyecto tiene como objetivos aprender a utilizar Puppet como una herramienta para administrar la configuración de varios servidores de una forma sencilla y automatizada.

1.OBJETIVOS

1.1 Objetivo general

Administrar ordenadores usando Puppet , teniendo un servidor máster y poder comunicarse con los clientes ejecutando recetas en ellos y así mantener un mejor control.

1.2 Objetivos específicos

- Lograr comunicación con los nodos.
- Emitir recetas recetas en los ordenadores.
- Controlar varios ordenadores que estén en comunicación con el máster.

2. MARCO TEÓRICO

Puppet es una herramienta desarrollada por puppetlabs para administrar la configuración de sistemas Unix y Windows de forma declarativa esto quiere decir que no le decimos a la maquina que es lo que tiene que ejecutar , si no el estado final en el que queremos que se encuentre.

Para entenderlo de una forma mas clara:

```
package    {' apache2':
    ensure =>  installed,
}
```

con esta simple declaración Puppet es capaz de instalar apache2, en ningún caso se le ha indicado que tiene que utilizar apt, para debian y yum para centos , solo le indicamos que este seguro que el paquete este instalado.

En resumen, Puppet nos permite realizar la configuración de forma abstracta, especificando el estado en el que queremos que se encuentre la maquina y no las ordenes que tiene que ejecutar, esto nos permite que con la declaración anterior, instalar el paquete independientemente del gestor de paquetes que tenga el sistema.

Componentes y Características

Puppet esta desarrollado en ruby, actualmente existen dos versiones una libre y otra enterprise

Puppet open source, bajo licencia Apache 2.0 para las versiones 2.7 en adelante y GNU GPL para las anteriores.

Puppet enterprise, es la solución de pago que ofrece PuppetLabs, aunque puedes probar el producto con un limite de 10 nodos.

¿Para qué me sirve Puppet?

Si bien, los administradores de sistemas pueden hacer uso de Puppet para obtener una gestión de sistemas consistente, transparentes y muy flexibles, permitiendo centrarnos en una sola infraestructura como cliente – servidor donde el servidor Puppetmaster se encarga de centralizar y ordenar todas las configuraciones para los clientes o servidores. Puppet lleva a cabo tareas de manera centralizada, es decir, despliega la misma configuración y el mismo software.

Site.pp: Contiene la ruta del manifiesto.

Init.pp: Contiene una definición de clase el nombre de esta clase debe coincidir con el nombre del modulo.

Fileserver.conf. Archivo donde se configura y se adhieren las rutas e Ip de los nodos

Puppet.conf: Archivo de configuración principal en los clientes y en el servidor.

Certificados SSL

Un certificado SSL es el elemento principal del protocolo de seguridad SSL, y se utiliza para que las transmisiones de información entre dos elementos conectados a través de Internet sea totalmente segura, siendo el caso más extendido el del protocolo HTTPS que sirve para conectar de forma segura con un servidor web. Los certificados SSL constan de dos partes, una pública y una privada. La parte pública será la encargada de cifrar la información, y la privada de descifrarla.

Certificado SSL en google.es

Al ser un protocolo abierto, puede ser empleado con la mayoría de los servicios de comunicación más extendidos de Internet, como pueden ser HTTP, FTP, SMTP, IMAP o POP3, entre otros, a los que SSL le añade una capa de seguridad (HTTP → HTTPS, FTP → FTPS, etc.)

Certificados SSL para implementar el protocolo HTTPS

En este caso, un certificado SSL aporta seguridad al visitante de un sitio web. Su misión consiste en identificar un servidor ante cualquier visitante, y en aportar un cifrado en la comunicación que permita preservar la confidencialidad y la integridad de los datos. Está formado por varios códigos que componen diferentes ficheros y sirven para implementar el protocolo SSL/TLS en un sitio web, y de esta forma establecer el protocolo HTTPS en la comunicación entre el cliente y el servidor.

De esta manera, un certificado SSL permite que la comunicación no pueda ser interceptada ni modificada por elementos no autorizados. Está formado por tres partes principales:

- **El certificado SSL**: Se trata de la parte pública. Está compuesto por un fichero, y cuando alguien se conecta a un servidor a través de HTTPS, es lo que recibirá en primer lugar. Contiene el nombre del dominio, y acredita que efectivamente se está comunicando con quien dice ser.
- **La clave privada**: Es fundamental que se almacene de forma segura y que bajo ningún concepto se dé a conocer. Funciona como un sello, y con ella se "sella" la comunicación, y se acredita que el servidor es, efectivamente, quien dice ser.
- **Certificados intermedios**: Las autoridades certificadoras o CA son terceras partes de confianza (por parte de los sistemas operativos y navegadores), y son las encargadas de expedir los certificados, y sólo lo harán si pueden verificar que la persona u organización que lo solicita es propietaria del dominio que desea certificar. Para ello se emiten certificados intermedios que son los que firmarán el certificado del servidor.

Capas del modelo TCP/IP , los protocolos y puertos involucrados en el proyecto.

CAPAS MODELO TCP/IP	PUERTOS INVOLUCRADOS	COMANDOS
Capa de acceso a Red	Ip, ARP	ping
Capa de acceso a Internet	Ip v4	Ip add
Capa Transporte	TCP/IP	Netstat (-a,putona)
Capa de Aplicacion	w.w.w, otros	Telnet,FTP.SMTP. DNS,NFS ,RIP, OTROS

Capa de acceso a Red: proporcionan al sistema los medios para enviar los datos a otros dispositivos conectados a la red. Definir del datagrama, que es la unidad básica de transmisión en Internet.

Capa de acceso a Internet: Definir el esquema de direccionamiento de Internet. Mover los datos entre la capa de acceso a red y la capa de transporte Encauzar los datagramas hacia sistemas remotos. (Routing) Realizar la fragmentación y re-ensamblaje de los datagramas.

Capa de transporte: encarga de los servicios de envío de datos con detección y corrección de errores.

Capa de aplicación: En esta capa se incluyen los procesos que usan los protocolos de la capa de transporte, tales como: Telnet,FTP.SMTP. DNS,NFS ,RIP, OTROS

DIAGRAMA DE ARQUITECTURA DEL PROYECTO

Ilustración 1: nodos que se ejecutan

DIAGRAMA DE RED FÍSICO

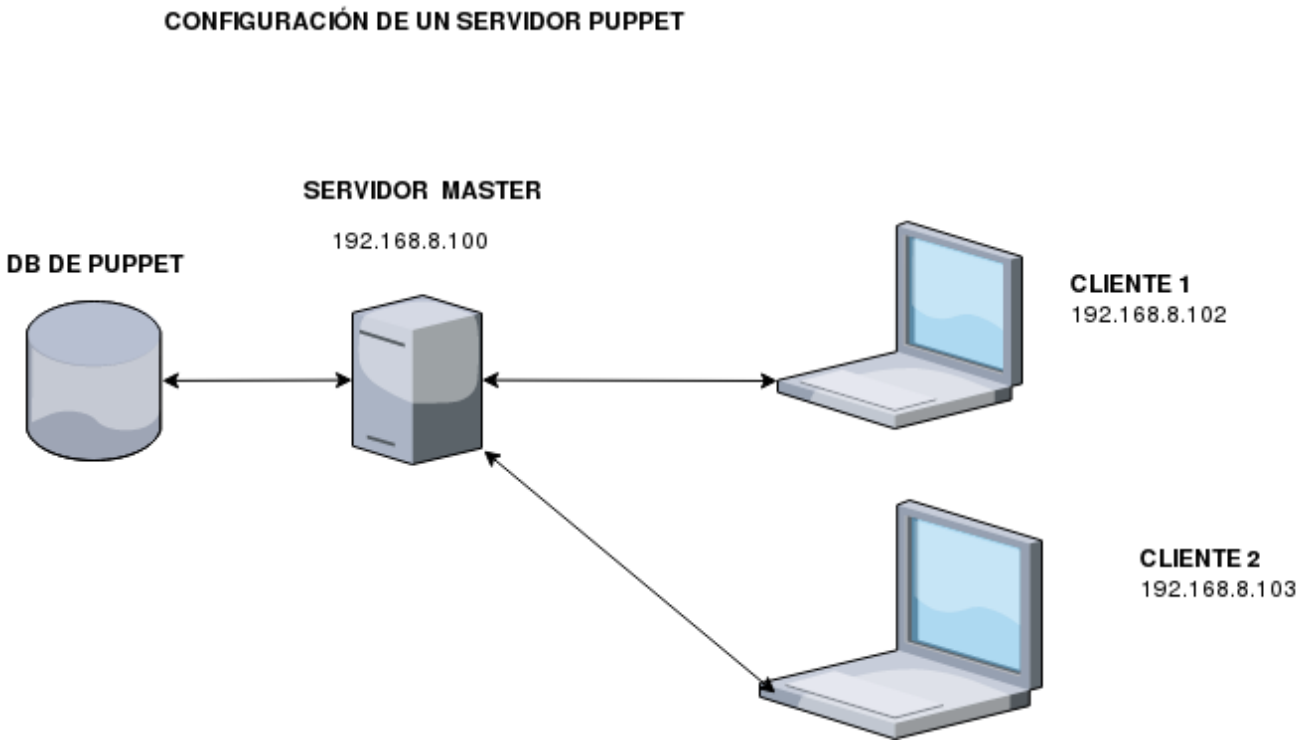


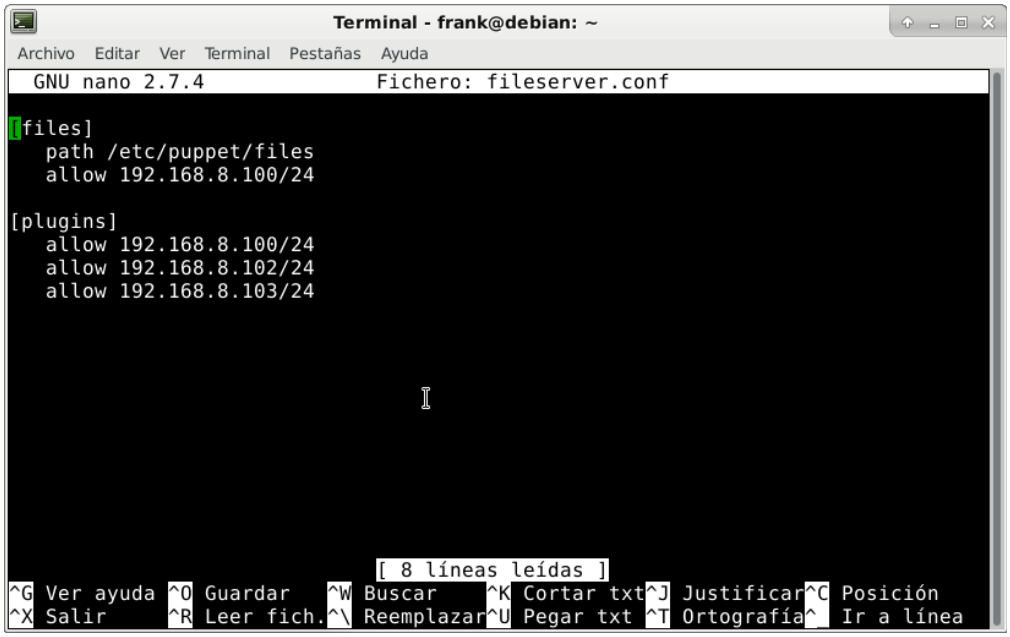
Ilustración 2: Diagrama de red físico

3. CONSTRUCCIÓN DEL PROYECTO

3.1Manual de Instalación de puppet

3.1.2Preparando el Master

1- Instalación en el master
apt-get install puppet-master
2- Abrimos el fichero fileserver.conf



3- Editamos el fichero puppet.conf

4- Editamos el fichero nano/etc/hosts

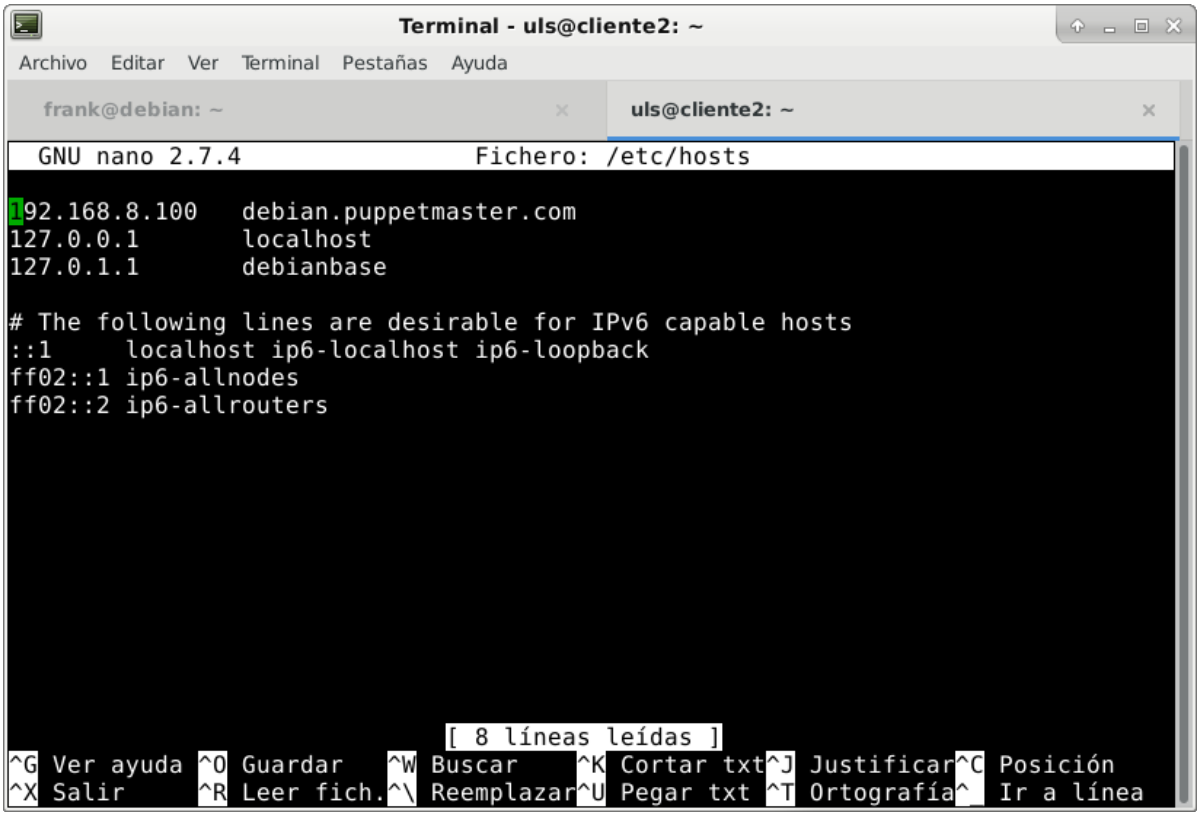
5- Editamos el fichero nano/etc/resolv.conf

4.Preparando los clientes

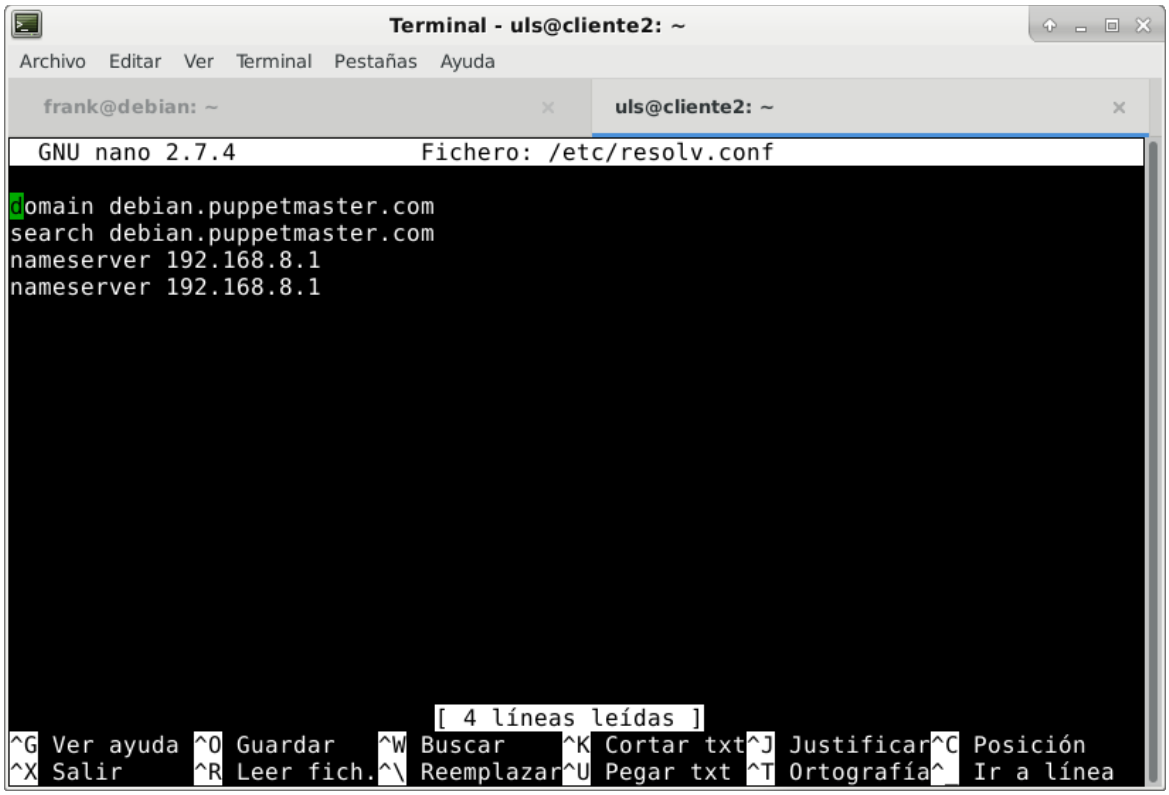
4.1-Instalación de Puppet

apt-get install Puppet

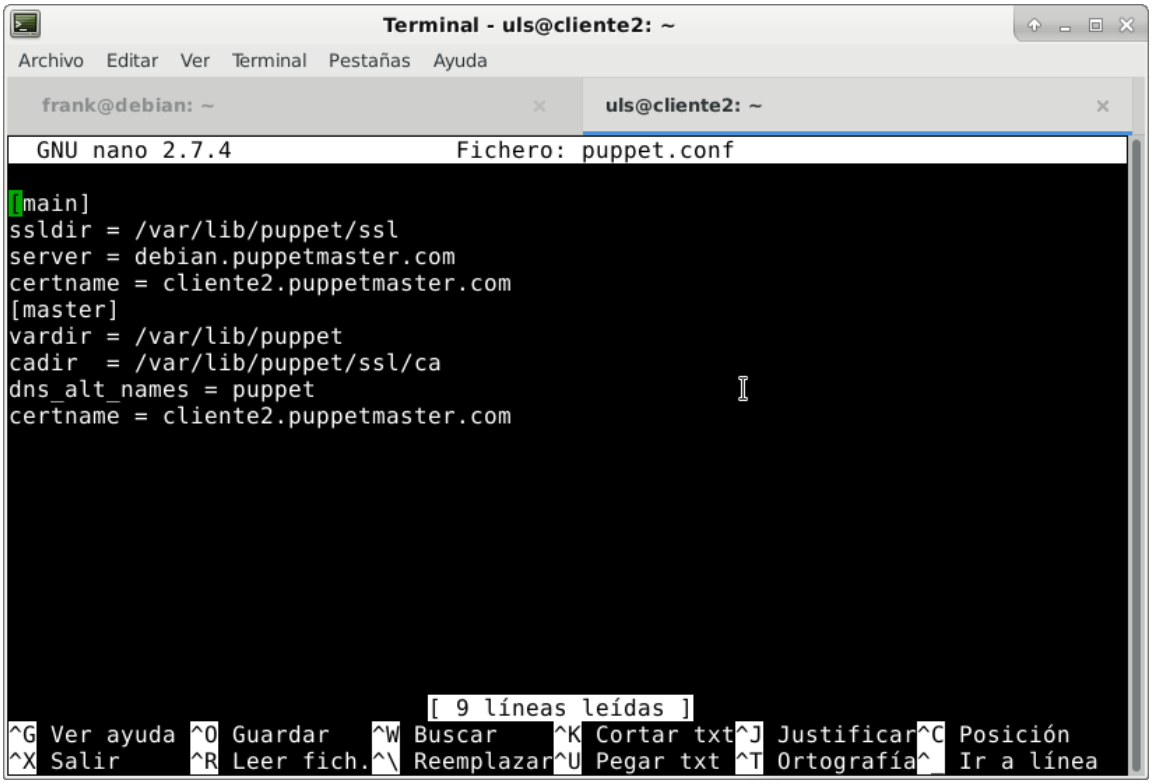
4.1.2- Editamos el archivos nano /etc/hosts de los clientes



4.1.3- Editamos en nano /etc/resolv.conf



4.1.4 Editamos el archivo puppet.conf del cliente1



5. Generar Certificados

5.1 Cliente

Puppet agent --server debian.puppetmaster.com --waitforcert 60 --test –verbose

puppet agent -vt

5.2 Master

Autoridad certificadora cert ca list

Listar certificados puppet cert –list

Listamos los certificados ya firmados puppet cert –list –all

5.3 Eliminar certificados

Detener el servidor en ambos /etc/init.d/puppet-master stop

cliente /etc/init.d/puppet stop

Reiniciar servidor en el cliente y en el master /etc/init.d/puppet-master restart

cliente /etc/init.d/puppet restart

Verificar el estado en ambos

/etc/init.d/puppet-master status

cliente /etc/init.d/puppet status

6. COMANDOS ARTEFACTOS Y PROCEDIMIENTOS QUE SE REALIZARON

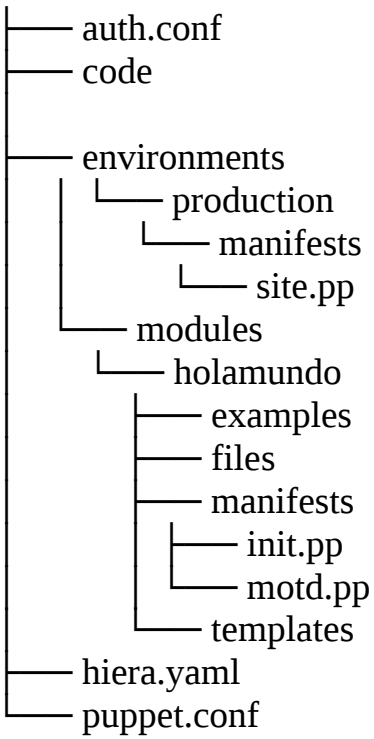
COMANDO	FUNCIÓN
Apt-get install	Instalación de paquetes
nano	Editor
Ip add	Muestra direcciones IP
reboot	reiniciar
Ls	Mostrar l que esta dentro de una carpeta
Mkdir	Crea una carpeta
cd	Entra en un direcctorio
Rm -rf	Borra un directorio
Config print	Imprimir configuracion
cp	copiar
Ps aux	Listar procesos
ping	Establecer comunicación atravez de ip
Puppet cert --list	Listar certificados en puppet
puppet cert --list --all	Lista certificados firmados

Tabla: 2 Comandos usados en el proyecto

7. ESCENARIO DE PRUEBAS

7.1 Creación de módulos y recetas

1- Se crean módulos en base a la estructura de carpetas



2- Creamos una serie de carpetas

```
cd code
cd environments
cd production
cd manifests
```

Donde de una sola ves incluimos los nodos donde se aplicara la receta que se trata de distribuir un fichero en los dos nodos.

1- Editamos

```
nano site.pp
  class prueba {
  file { "/root/ficherodeprueba":
  owner => 'root',
  group => 'root',
```

```
  mode  => '440'
,
  source => "puppet:///modules/prueba/ficherodeprueba"
  }
}

node 'cliente1' {
        include prueba
}

node 'cliente2' {
        include prueba
}
```

8. COSTO DEL PROYECTO

N°	Descripción	Costo
1	Pc Servidor	$300
2	Pc cliente1	$200
3	Pc cliente2	$200
4	Internet	$50
5	Transporte	$50
6	Alimentación	$40
7	Papelería	$5
8	Electricidad	$10
9	Oros gastos	$10
10	total	$875

9. BUENAS PRACTICAS PARA LA CONSTRUCCIÓN DEL PROYECTO.

Tomar en cuenta los siguientes aspectos para poder desarrollar e implementar puppet

1- Si elige la arquitectura estándar de agente / maestro, debe decidir qué servidor (s) actúa como el maestro de marionetas(y el servidor puppet master

2- Revise las versiones SO y requerimientos del sistema.

3- Consulta los requerimientos de sistemapara la versión de Puppet a instalar, considerando los siguientes puntos:

- Su Puppet master ha de ser capaz de gestionar el número total de clientes a servir.
- Los sistemas que proporcionamos paquetes oficiales tienen un path de instalación más fácil.

4- Compruebe la configuración de red.

- En despliegues de clientes/master, debes preparar tu red para soportar el tráfico de Puppet.
- Documentar errores y solución para poder manejar mejor el sistema y si se presenta algún error o no se recuerda de algo existe un respaldo para poder actual.

5- Si es primera experiencia en puppet realizar pruebas en maquinas virtuales usando sortware de virtualización.

10. CONCLUSIONES

La herramienta de puppet es muy importante poder implementarla porque encierra muchos aspectos para administración, orden y rapidez al momento de que el administrador desee aplicar una acción. Puppet es un software que permite realizar la configuración de forma abstracta, especificando el estado en el que queremos que se encuentre la maquinas en las cuales tenemos una vez ya instalado puppet porque solo teniendo puppet en el servidor y los nodos con las respectivas configuraciones se puede aplicar cualquier orden y tener el estado deseados en los equipos.

11.RECOMENDACIONES

Se deben tomar en cuenta muchos aspectos para el buen uso de la herramienta de puppet si se desea implementar conocer todo lo referente a puppet y poder implementarlo. También cabe mencionar que puppet nos sirve para poder gestionar un gran volumen de ordenadores lo que nos hace factible poder implementarlo no solo en aula informática si no en cualquier empresa donde halla mucho trafico de información en la red y poder solucionar de manera mas rápida cual tipo de ejecución que necesitemos en un determinado momento. Puppet una herramienta novedosa que sirve de mucho poder emplearla.

12. BIBLIOGRAFÍAS

Los artículos del blog Yo, admins is son creados por Francisco Jose Bejarano Melero y se les aplica la licencia Cretive Commons Reconocimiento-No comercial-Compartir bajo la licencia 3.0 españa

Bolg de Wordpress.com Juantrucupei https://www.digitalocean.com/community/tutorials/how-to-install-puppet-to-manage-your-server-infrastructure

Publicado por Esteban M. Navas Martin
Publicado en Puppet

Etiquetado configuracion, ine stalacion, script, softwarelibrhttps://laenredadera.net/configuracion-de-puppet-en-debian-jessie-y-creacion-de-un-modulo

Notas de ant 30
Esta entrada fue publicada en blog y etiquetada linux, puppet, software por ant30.

Creación de módulos para puppet

Publicado por Julian García-Sotoca