

Administración de Servidores con Puppet

Administración de Servidores con Puppet

Tony García

tonyg@tonyskapunk.net

www.tonyskapunk.net

Que es Puppet?

FOSS, Software Libre^[1] y Gratuito^[2], Distribuido bajo GPLv2 Apache 2.0.

Puppet es una **plataforma** escrita en **ruby** enfocado en **automatización** y **configuración** de sistemas.

Utiliza un Lenguaje **Declarativo** que hace que sea sencillo de aprender y esta pensado para "**describir**" la configuracion de **uno o varios** sistemas.

[1]Codigo disponible en github.

[2] A partir del primero de Febrero de 2011 se anuncio una version Empresarial la cual tiene un costo.

Quién esta detras de Puppet?

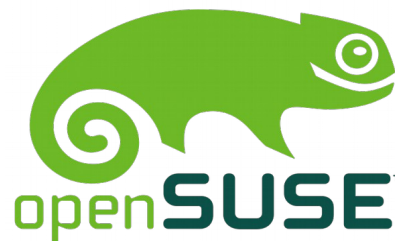
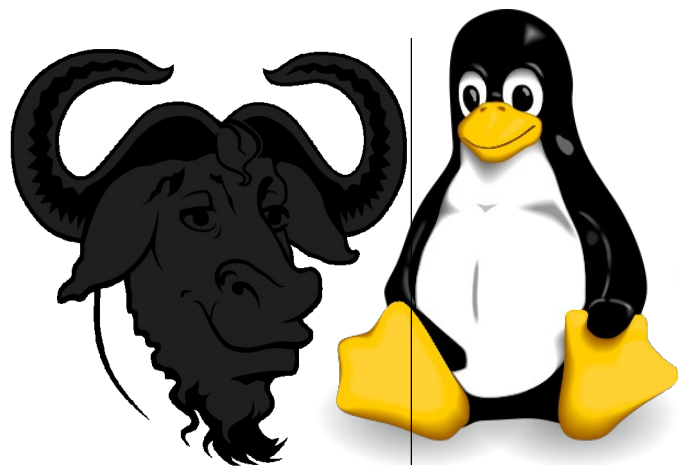
La compañía privada Puppet Labs, antes conocida como Reductive Labs, localizadas en Portland OR. Fundada en 2005 por el actual CEO Luke Kanies



En qué plataformas o sistemas operativos se puede utilizar?

En prácticamente todo lo que soporte ruby.

Linux(Debian, RedHat), BSDs, MAC, Unix(HPUX, AIX, Solaris), Windoze.



Breve historia de Puppet...

Luke Kanies fundador y CEO de la compañía es graduado de la Universidad de Reed en **Portland, OR**, tiene una Ingeniería en Química, trabajo varios años como Unix SysAdmin y con el paso de los años se dio cuenta de que mantener múltiples nodos en diferentes Data Centers de manera **manual** era **tonto** e innecesario, empezó a considerar la opción de trabajar de una manera proactiva más que reactiva, esto a **principios de 2000**, En **2001** gracias a LISA (Large Installation System Administrator) conoció al autor de **ISconf**, una herramienta que ayuda a automatizar instalaciones, asegurándose de **seguir el mismo orden** de pasos para tratar de llegar a un estado consistente, esta idea le pareció muy interesante e incluso participó en código de ISconf, con el paso del tiempo se dio cuenta de las limitaciones de esta herramienta pero fue una gran influencia para Luke.

Poco tiempo después fue el turno de **Cfengine** una herramienta de administración de sistemas, en el cual el **orden no lo es todo** como en el caso de ISConf, aquí lo interesante fue la **mezcla** de los dos sistemas para llevar a cabo su cometido "**Administrar Sistemas**", en donde el **resultado** final fue terminar usando **Cfengine** pero esto lo ayudó a darse cuenta de la falta de control en actividades a nivel de OS como control de usuarios, paquetes entre otros, además de tener una gran **dificultad** al manejar diferentes tipos de **plataformas**.

Ante esta situación y la **falta de desarrolladores** en Cfengine en **2004** decidió hacer un **fork** del proyecto y llamarlo **cnfg**, el cual **no sobrevivió** por mucho tiempo, con todas estas ideas Luke decidió crear un pequeño **prototipo** que las mantendría vivas llamándolo **Blink**.

Después de algún tiempo y tratar de compartir sus ideas y encontrar la solución perfecta, sin que **nadie lo tomara** mucho en **cuenta**, decidió continuar con su prototipo y empezar con las ideas básicas que hoy Puppet tiene, Luke siendo un desarrollador de **Perl** se dio cuenta de las limitaciones que enfrentaría, **Python** era la otra opción viable pero no era lo suyo, un amigo recomendó **Ruby** y decidió darse una oportunidad, en cuestión de **horas** pudo finalmente tener un prototipo **funcional** y como él mismo lo dice nunca se ha arrepentido de esta decisión.

En **Marzo de 2005** después de varios **problemas económicos y laborales**, Luke decidió fundar **Reductive Labs**, cambió el nombre de Blink a **Puppet** y así fue como nació la empresa; el dominio reductivelabs.com fue registrada el 24 de Febrero de 2005.

Breve Historia de Puppet...

2006- ???

2007



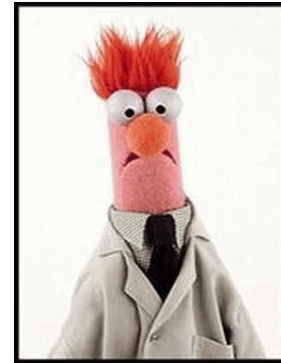
29 Enero 2007 - 0.22.1 (kermit)

19 Marzo 2007 - 0.22.2 (grover)



18 Junio 2007 - 0.23.0 (fozzie)

20 Julio 2007 - 0.23.1 (beaker)



13 Diciembre 2007 - 0.24.0 (misspiggy)

2008



Enero 2008 - Pulling Strings with Puppet
James Turnbull

7 Julio 2008 - 0.24.5 (scooter)



22 Octubre 2008 - 0.24.6 (camila)



15 Diciembre 2008 - 0.24.7 (gonzo)

Breve Extensa Historia de Puppet...

2009

4 Septiembre 2009 - 0.25.0 (elmo)

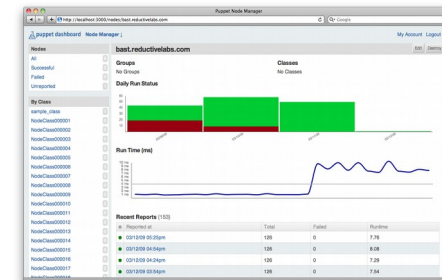


28 Febrero 2009 - 0.24.8 (beauregard)



1 Septiembre 2009 - 0.25.1 (zoot)

8 Diciembre 2009 - Puppet Dashboard



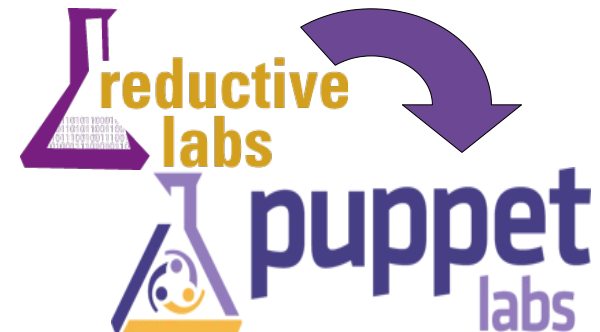
2010

1 Enero 2010 - Puppet Docs

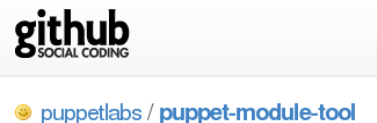


4 Enero 2010 - 0.25.2 (zoe)

4 Enero 2010 - 0.25.3 (clifford)



24 Marzo 2010



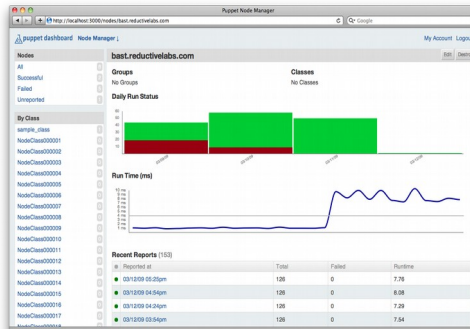
27 Mayo 2010 - Puppet Modules



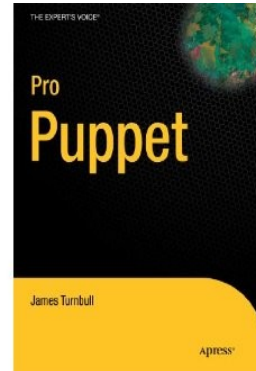
19 Julio 2010 - 2.6.0 (rowlf)

Breve Extensa (Zzz..) Historia de Puppet...

2011



4 Abril 2011 - Puppet Dashbard 1.1.0



4 Mayo 2011 – Pro Puppet



21 Junio 2011 - 2.7.0 (statler)

Y despues....



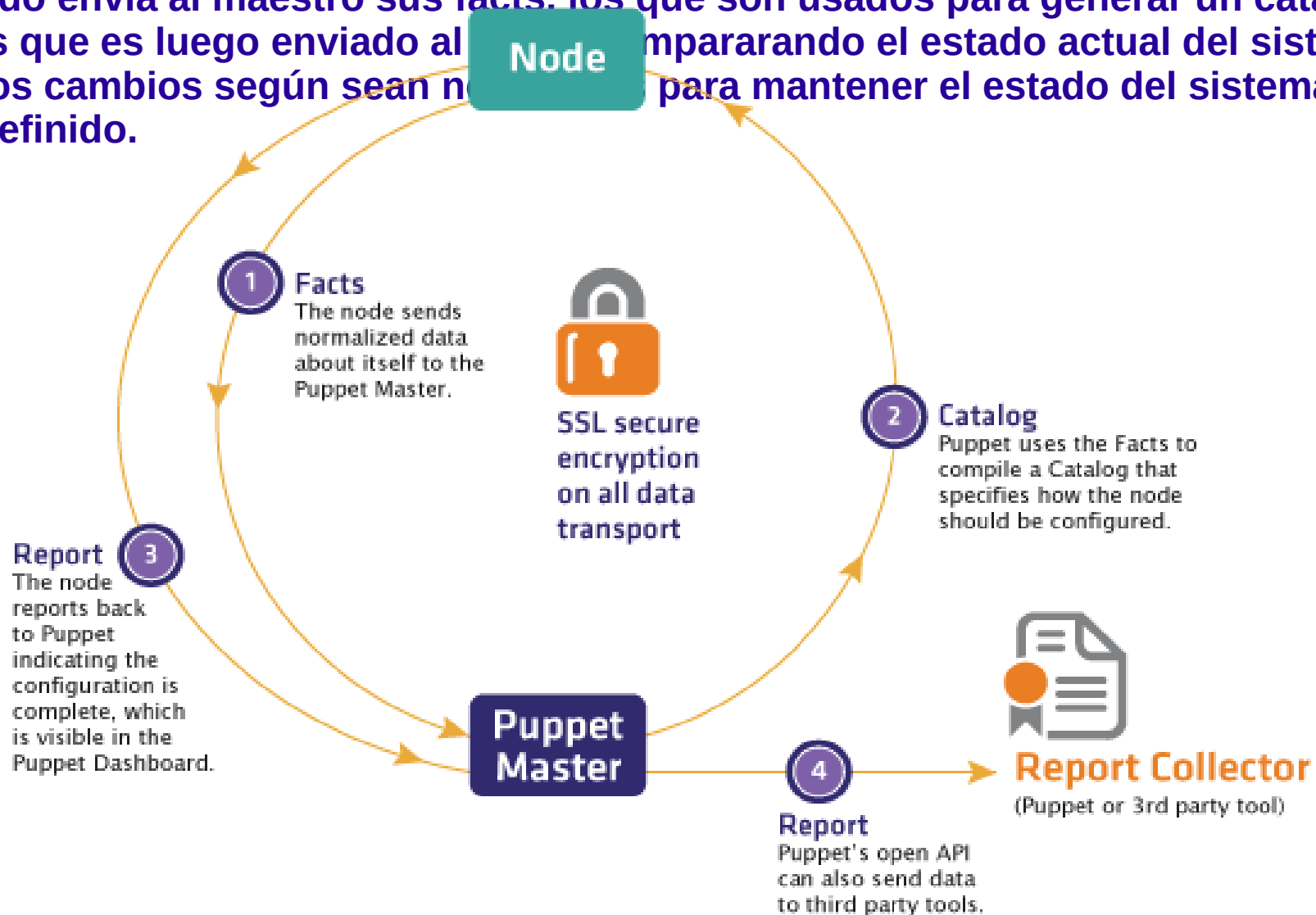
2.8.0 (telly)

How Puppet Manages Data Flow

Cómo funciona Puppet?

for Individual Nodes

Cada nodo envía al maestro sus facts, los que son usados para generar un catalogo de recursos que es luego enviado al nodo comparando el estado actual del sistema y realiza los cambios según sean necesarios para mantener el estado del sistema que le fue predefinido.



Que es lo que puede controlar puppet?

Servicios

Paquetes

Archivos

Usuarios

Grupos

Cron

Mount(puntos de montajes)

Ejecución de comandos.

...

Que no puede hacer puppet?

- Cosas que no puedas definir como un estado, por ejemplo reiniciar el equipo.
- Tampoco puede controlar aquello que tu **no** has definido explícitamente, es decir no puedes asumir que puppet va a remover un paquete sólo porque no has definido ese paquete.
- Entonces Puppet sólo puede tener control sobre lo que tu defines.

Manos en Puppet...

A
i

B
d

C
j

D
i

E
j

F
d

G
i

H
d

I
j

J
i

K
i

L
d

M
j

N
d

O
i

P
j

R
i

S
i

T
d

U
j

V
i

W
j

Y
d

Z
d

Obtener Ips de cada equipo y crear entradas para cada uno en /etc/hosts e.g.

```
192.168.1.1 puppet.fslvallarta.org puppet
192.168.1.2 client.fslvallarta.org client
```

Instalar los repositorios necesarios.

```
# cd /etc/yum.repos.d
# wget --no-check-certificate \
https://raw.githubusercontent.com/tonyskapunk/repos/master/epel.repo

# wget --no-check-certificate \
https://raw.githubusercontent.com/tonyskapunk/repos/master/puppetlabs.repo

# wget --no-check-certificate \
https://raw.githubusercontent.com/tonyskapunk/repos/master/rpmforge.repo
```


Instalar GIT.

```
# yum install git -y --no-gpg
```

```
# apt-get install git -y
```

Instalar Puppet Server en el maestro.

```
# yum install puppet-server ruby-rdoc -y
```

```
# apt-get install puppetmaster
```

Ahora en los clientes.

```
# yum install puppet -y
```

```
# apt-get install puppet -y
```

Iniciar el servicio de PuppetMaster...

```
# puppet master --verbose --no-daemonize
```

```
/usr/lib/ruby/site_ruby/1.8/puppet/type/file.rb:48: warning: parenthesize argument(s) for future version
```

```
info: Creating a new SSL key for ca
```

```
info: Creating a new SSL certificate request for ca
```

```
info: Certificate Request fingerprint (md5): EE:53:9C:35:F6:52:EF:15:3B:C1:C3:8A:D2:64:4D:F0
```

```
notice: Signed certificate request for ca
```

```
notice: Rebuilding inventory file
```

```
info: Creating a new certificate revocation list
```

```
info: Creating a new SSL key for gnucloud.tonyskapunk.net
```

```
info: Creating a new SSL certificate request for gnucloud.tonyskapunk.net
```

```
info: Certificate Request fingerprint (md5): 86:3D:4F:42:92:89:5D:28:FF:BC:64:2C:7E:D6:36:AE
```

```
notice: gnucloud.tonyskapunk.net has a waiting certificate request
```

```
notice: Signed certificate request for gnucloud.tonyskapunk.net
```

```
notice: Removing file Puppet::SSL::CertificateRequest gnucloud.tonyskapunk.net at
```

```
'/var/lib/puppet/ssl/ca/requests/gnucloud.tonyskapunk.net.pem'
```

```
notice: Removing file Puppet::SSL::CertificateRequest gnucloud.tonyskapunk.net at
```

```
'/var/lib/puppet/ssl/certificate_requests/gnucloud.tonyskapunk.net.pem'
```

```
notice: Starting Puppet master version 2.7.6
```

puppet master

SSL

/var/lib/puppet/ssl

/var/lib/puppet/ssl/ca

ca_crl.pem ca.crt.pem ca_key.pem ca_pub.pem **inventory.txt** **private** requests serial **signed**

/var/lib/puppet/ssl/certs

ca.pem gnucloud.tonyskapunk.net.pem

/var/lib/puppet/ssl/private_keys

gnucloud.tonyskapunk.net.pem

/var/lib/puppet/ssl/public_keys

gnucloud.tonyskapunk.net.pem

puppet cert

Configurar Puppet Master con dos ambientes (production, test)

1. Detener puppet master: **# *service puppetmaster stop***
2. Crear en el master: **# *mkdir -p /opt/puppet/{production,test}***
3. Bajar configuracion de prueba del repositorio en github: **# *mkdir /opt/git; pushd /opt/git; git clone git://github.com/tonyskapunk/puppet-fsl.git; popd***
4. Copiar configuraciones a ambos ambientes: **# *pushd /opt/git/puppet-fsl; cp manifests/* /opt/puppet/production/ -R; cp manifests/* /opt/puppet/test/ -R; popd***
5. Cambiar permisos para que puppet pueda leer: **# *chown puppet:puppet /opt/puppet -R***
6. Copiar config de puppet: **# *cp -f /opt/git/puppet-fsl/config/puppet.conf /etc/puppet/***
7. Iniciar puppet manual: **# *puppet master --verbose --no-daemonize***

Configuración de Puppet...

[main]

```
# The Puppet log directory.  
# The default value is '$vardir/log'.  
logdir = /var/log/puppet  
  
# Where Puppet PID files are kept.  
# The default value is '$vardir/run'.  
rundir = /var/run/puppet  
  
# Where SSL certificates are kept.  
# The default value is '$confdir/ssl'.  
ssldir = $vardir/ssl
```

[master]

```
# environments  
environments = production,test  
  
# Module Path  
modulepath = /opt/puppet/production/modules  
# Manifest Site (default $manifestdir/site.pp)  
manifest = /opt/puppet/production/site.pp
```

[production]

```
# Module Path  
modulepath = /opt/puppet/production/modules  
# Manifest Site (default $manifestdir/site.pp)  
manifest = /opt/puppet/production/site.pp
```

[test]

```
# Module Path  
modulepath = /opt/puppet/test/modules  
# Manifest Site (default $manifestdir/site.pp)  
manifest = /opt/puppet/test/site.pp
```

[agent]

```
# The file that stores a list of the classes  
classfile = $vardir/classes.txt
```

Puppet Agent...

```
# puppet agent --no-daemonize --environment=test
```

notice: foo.bar.com has a waiting certificate request

En el maestro ejecutar:

```
# puppet cert -l
```

Firmar el requerimiento:

```
# puppet ca sign foo.bar.com
```

notice: Signed certificate request for foo.bar.com

notice: Removing file Puppet::SSL::CertificateRequest foo.bar.com at '/var/lib/puppet/ssl/ca/requests/foo.bar.com.pem'

puppet agent

Puppet – Lenguaje y Estructura...

<http://docs.puppetlabs.com>

**/opt/puppet/test/
classes modules nodes site.pp**

**/opt/puppet/modules/
http load motd motdi**

**modules
files manifests templates**

**facts
lib/facter**

Puppet – Modulos y Facts

http load motd motdi

modules

files manifests templates

facts

lib/facter

Puppet – Resource

```
$ puppet resource package puppet
```

```
package { 'puppet':  
  ensure => '2.7.6-2.el5',  
}
```

```
$ puppet resource file /etc/motd
```

```
file { '/etc/motd':  
  ensure  => 'file',  
  content => '{md5}bec453e3e17f926c4ce5a24dac1af18c',  
  ctime   => 'Sat Oct 01 17:09:06 +0000 2011',  
  group   => '0',  
  mode    => '644',  
  mtime   => 'Sat Oct 01 17:09:06 +0000 2011',  
  owner   => '0',  
  type    => 'file',  
}
```

```
$ puppet resource service httpd
```

```
service { 'httpd':  
  ensure => 'running',  
  enable => 'true',  
}
```

Proyectos Relacionados

- **Foreman** – <http://theforeman.org>
- **Mcollective** – <http://puppetlabs.com/mcollective>

Puppet Enterprise:Cuál es la diferencia?

- Provee en un solo paquete facilitando la instalación. (Puppet servidor y agente, Dashboard, Module Tool, Facter, Ruby)
- Contiene paquetes para escalarlo(Apache, Phusion Passenger)
- Soporte con PuppetLabs.
- Precio, varia dependiendo del numero de Nodos:
 - 25 Nodos **\$1,995.00** usd.
 - Soporte Premium* **\$26,950.00** usd.
 - 2,500 Nodos* **\$185,625.00** usd.
 - Soporte Premium* **\$215,950.00** usd.

* Estos precios han reducido y no estan disponibles al publico.

Ventajas de Puppet?

- Estructuración por modulos, permitiendo reusar codigo.
- Provee una capa de transparencia para la administración de equipos, es lo mismo que si es un Debian, que si es un RedHat.
- Sobrecarga de codigo de puppet.
- Creación de tus propios "facts".

Mis comentarios de Puppet:

Yo recomiendo Puppet desde el uso de un par de maquinas, hasta miles de ellas.

Sus ventajas son el reuso de código, su flexibilidad con diferentes SOs,

El hacer uso de Puppet reduce considerablemente las tareas repetitivas de preparación de una maquina, de un ambiente o incluso de toda una estructura compleja de equipos.

Puppet no es la herramienta que resolvera todos tus problemas pero te permitirá tener más tiempo para enfocarte en arreglar otros problemas o bien mejorar otras areas.

Sus comentarios de Puppet ?

GRACIAS!!

annA Lee – Por su paciencia, comprensión y apoyo incondicional en lo que creo y quiero.

@chitopunk – Por su inspiración y por inducirme a GNU/Linux.

Contacto

Tony García

tonyg@tonyskapunk.net
www.tonyskapunk.net

Referencias

<http://docs.puppetlabs.com/>

http://projects.puppetlabs.com/projects/puppet/wiki/Cheat_Sheets

http://projects.puppetlabs.com/projects/puppet/wiki/Documentation_Start

<http://tmz.fedorapeople.org/repo/puppet/>

<http://devopscafe.org/show/2010/12/20/episode-17.html>

<http://www.linkedin.com/in/lukekanies>

<http://madstop.com/2008/11/01/a-short-history-of-puppet-pt>

<http://madstop.com/2008/11/06/a-short-puppet-history-pt-2-cfengine>

<http://www.isconf.org>

<http://cfng.tigris.org>

<https://github.com/puppetlabs>

<http://tmz.fedorapeople.org/repo/puppet/epel/puppet.repo>

http://projects.puppetlabs.com/projects/1/wiki/Downloading_Puppet

<http://docs.puppetlabs.com/references/latest/type.html>

http://stahnma.fedorapeople.org/puppetlabs/5/products/x86_64