puppet 资源 service 详细介绍（附案例）

一、资源介绍

Description

Manage running services. Service support unfortunately varies widely by platform — some platforms have very little if any concept of a running service, and some have a very codified and powerful concept. Puppet’s service support is usually capable of doing the right thing, but the more information you can provide, the better behaviour you will get.

Puppet 2.7 and newer expect init scripts to have a working status command. If this isn’t the case for any of your services’ init scripts, you will need to set hasstatus to false and possibly specify a custom status command in the status attribute.

Note that if a service receives an event from another resource, the service will get restarted. The actual command to restart the service depends on the platform. You can provide an explicit command for restarting with the restart attribute, or you can set hasrestart to true to use the init script’s restart command; if you do neither, the service’s stop and start commands will be used.

```
service { 'resource title':
  name       => # (namevar) The name of the service to run.  This name is...
  ensure     => # Whether a service should be running.  Valid...
  binary     => # The path to the daemon.  This is only used for...
  control    => # The control variable used to manage services...
  enable     => # Whether a service should be enabled to start at...
  hasrestart => # Specify that an init script has a `restart...
  hasstatus  => # Declare whether the service's init script has a...
  manifest   => # Specify a command to config a service, or a path
  path       => # The search path for finding init scripts....
  pattern    => # The pattern to search for in the process table...
  provider   => # The specific backend to use for this `service...
  restart    => # Specify a *restart* command manually.  If left...
  start      => # Specify a *start* command manually.  Most...
  status     => # Specify a *status* command manually.  This...
  stop       => # Specify a *stop* command...
  # ...plus any applicable metaparameters.
}
```

1、实现功能

1.1 服务处于运行状态

1.2 服务能够在配置文件更改的情况下自动重启

二、系统环境

1、puppet 服务端

```
Release：RHEL6.4
HOSTNAME: puppetserver.kisspuppet.com
TCP/IP: 172.16.200.100/24
Packages：
puppet-server-2.7.21-1.el6.noarch
mcollective-client-2.2.4
activemq-5.5.0
```

2、puppet 节点

```
Release: RHEL5.8
HOSTNAME: agent1.kisspuppet.com
TCP/IP: 172.16.200.101/24
Packages:
puppet-2.7.21-1.el5
mcollective-2.2.4-1.el5
```

3、puppet 节点

```
Release: RHEL6.4
HOSTNAME: agent3.kisspuppet.com
TCP/IP: 172.16.200.103/24
Packages:
puppet-2.7.21-1.el6
mcollective-2.2.4-1.el6
```

三、支持参数

```
ensure => running|stopped 指定服务的目标状态

enable => true|false 指定服务是否开机自启动(修改 chkconfig 部分)，并非对所有均有效
```

```
name => "service name",该资源的 namevar, 服务的名字,通常就是在/etc/init.d/目录下的名字,默认与 title 相同

hasstatus => true|false, 指出管理脚本是否支持 status 参数,puppet 用 status 参数来判断服务是否已经在运行了,如果不支持 status 参数,puppet 利用查找运行进程列表里面是否有服务名来判断服务是否在运行.

hasrestart => true|false, 指出管理脚本是否支持 restart 参数,如果不支持,就用 stop 和 start 实现 restart 效果.

path => "/etc/rc.d/init.d", 启动脚本的搜索路径,可以用冒号分割多个路径,或者用数组指定

provider => base|daemontools|init,   默认为 init
```

四、资源示例

1、示例一

```
1.1 实现功能
*要求系统启动后，sshd 服务自动启动
*要求通过系统进程方式查看 sshd 服务运行状态
*要求服务关闭后能够自动重启
*要求配置文件被更改后服务能够执行 restart 动作
```

```
1.2 配置说明
class ssh::service{
  service { $ssh::params::ssh_service_name:
    ensure     => running,
    hasstatus  => false,
    hasrestart => true,
    enable     => true,
    subscribe  => Class["ssh::config"],
  }
}
或
class ssh::service{
  service { $ssh::params::ssh_service_name:
    ensure     => stopped,
    hasstatus  => false,
    hasrestart => true,
    enable     => true,
    subscribe  => Class["ssh::config"],
    }
  }
```

```
1.3 客户端 agent3 测试
[root@agent3 ~]# puppet agent --test
info: Retrieving plugin
info: Loading facts in /var/lib/puppet/lib/facter/backup_date.rb
info: Loading facts in /var/lib/puppet/lib/facter/my_apply1.rb
info: Loading facts in /var/lib/puppet/lib/facter/my_apply3.rb
info: Loading facts in /var/lib/puppet/lib/facter/my_apply2.rb
info: Caching catalog for agent3.kisspuppet.com
info: Applying configuration version '1378281102'
notice: Finished catalog run in 0.38 seconds
[root@agent3 ~]#
[root@agent3 ~]# puppet agent --test
info: Retrieving plugin
info: Loading facts in /var/lib/puppet/lib/facter/backup_date.rb
info: Loading facts in /var/lib/puppet/lib/facter/my_apply1.rb
info: Loading facts in /var/lib/puppet/lib/facter/my_apply3.rb
info: Loading facts in /var/lib/puppet/lib/facter/my_apply2.rb
info: Caching catalog for agent3.kisspuppet.com
info: Applying configuration version '1378280861'
notice: /Stage[main]/Ssh::Service/Service[sshd]/ensure: ensure changed 'running' to
'stopped'
notice: Finished catalog run in 0.54 seconds

测试结果：ensure => running, hasstatus => false,的情况下服务如果被关闭是起不来的，
ensure => stopped, hasstatus => false,的情况下服务如果是运行的可以被关闭
建议：写 SysV 脚本放到/etc/init.d 目录下来实现
```

2、示例二

```
2.1 实现功能
*要求服务配置文件被改动后，服务能够自动 reload 而不是自动 restart（也可以通过 exec 资源实
现）
```

2.2 配置说明

```
class ssh::service{
  service { $ssh::params::ssh_service_name:
    ensure     => running,
    hasstatus  => true,
    hasrestart => true,
    enable     => true,
    subscribe  => Class["ssh::config"],
#   provider   => base|daemontools|init,
    provider   => init,
    path       => "/etc/rc.d/init.d",
    restart    => "/etc/rc.d/init.d/sshd reload",
    start      => "/etc/rc.d/init.d/sshd start",
    stop       => "/etc/rc.d/init.d/sshd stop",
  }
}

class ssh::config{
  file { $ssh::params::ssh_service_config:
    ensure  => present,
    owner   => 'root',
    group   => 'root',
    mode    => 0640,
    source  => "puppet:///modules/ssh/etc/ssh/sshd_config",
#   backup  => ".$backup_date.bak",
    backup  => 'main',
    require => Class["ssh::install"],
    notify  => Class["ssh::service"],  #等同于 class ssh::service 中的 subscribe
  }
}
```

2.3 客户端 agent3 测试

可修改 agent3 端/etc/rc.d/init.d/sshd 进行测试

```
case "$1" in
        start)
                start
                ;;
        stop)
                stop
                ;;
        restart)
                stop
                start
                echo "sshd restart stop-start" >>/tmp/sshd_status
                ;;
        reload)
                reload
                echo "sshd reload" >>/tmp/sshd_status
                ;;
测试部分
[root@agent3 ~]# puppet agent --test
info: Retrieving plugin
info: Loading facts in /var/lib/puppet/lib/facter/backup_date.rb
info: Loading facts in /var/lib/puppet/lib/facter/my_apply1.rb
info: Loading facts in /var/lib/puppet/lib/facter/my_apply3.rb
info: Loading facts in /var/lib/puppet/lib/facter/my_apply2.rb
info: Caching catalog for agent3.kisspuppet.com
info: Applying configuration version '1378279503'
notice: /File[/etc/ssh/sshd_config]/content:
--- /etc/ssh/sshd_config    2013-09-04 15:27:22.177863699 +0800
+++ /tmp/puppet-file20130904-19622-yy8g9o-0 2013-09-04 15:29:47.791863671 +0800
@@ -5,8 +5,6 @@
 Protocol 2
 #AddressFamily any

-# HostKey for protocol version 1
-
 # Lifetime and size of ephemeral version 1 server key
 #KeyRegenerationInterval 1h
 #ServerKeyBits 768

info: FileBucket adding {md5}43f5b3f207a1b6fb35e3bd779b83c3f8
info: /File[/etc/ssh/sshd_config]: Filebucketed /etc/ssh/sshd_config to main with
sum 43f5b3f207a1b6fb35e3bd779b83c3f8
notice: /File[/etc/ssh/sshd_config]/content: content changed
'{md5}43f5b3f207a1b6fb35e3bd779b83c3f8' to '{md5}df197ccd4957217616b62a82b890ed98'
info: /File[/etc/ssh/sshd_config]: Scheduling refresh of Class[Ssh::Service]
info: Class[Ssh::Config]: Scheduling refresh of Service[sshd]
info: Class[Ssh::Service]: Scheduling refresh of Service[sshd]
notice: /Service[sshd]: Triggered 'refresh' from 2 events
notice: Finished catalog run in 0.97 seconds
[root@agent3 ~]# cat /tmp/sshd_status
sshd reload   #可以看到服务是 reload 而不是 restart
[root@agent3 ~]#
```

为了能够和大家更好的交流和学习 Puppet，本人 2014 年又新开辟了微信公众号进行交流学习，目前已经有 300 多人同时收听，喜欢 Puppet 的大神们可自行加入哦。

如果你有好的有关 Puppet 的咨询也可以给我投稿，投稿地址：admin@kisspuppet.com

微信公众号：“puppet2014”，可搜索加入，也可以扫描以下二维码
