puppet 报告系统 Dashboard 部署及配置详解

Puppet Dasshboard 是由支持 Puppet 开发的公司 Puppetlabs 创建的，是 Ruby on Rails 程序。可以作为一个 ENC（外部节点分类器）以及一个报告工具，并且正在逐渐成为一个包含许多 Puppet 新功能的集成界面，例如审计和资源管理功能。 Puppet Dashboard 是一个 Ruby on Rails 程序，用于显示 Puppet master 和 agent 的相关信息。它允许你查看从一个或多个 Puppet master 汇总的图形和报告数据。它同时从一个或者多个 Puppet master 上收集来自于 Puppet agent 的资产数据（主机的 Fact 和其他信息）。最后，它能作为一个 ENC 来配置 Puppet 节点，并指定这些节点上的类和参数。

1 前期准备工作

Puppet Dashboard（1.2.3）程序目前版本只能安装在 Ruby 1.8.x（Dashboard 还不能工作在 1.9.x 下或者更新的版本下），只支持 MySQL 作为数据库后端。

```
Rake version 0.8.3 or newer
MySQL database server version 5.x
Ruby-MySQL bindings version 2.7.x or 2.8.x
```

备注：更多详细信息请参考：http://docs.puppetlabs.com/dashboard/

2 安装相关软件包

```
[root@puppetserver nodes]# yum install  ruby-mysql mysql-server puppet-dashboard
```

3 配置 Dashboard（包括与数据库的结合部分）

3.1 创建管理 Dashboard 的 MySQL 数据库账号并授权

```
[root@puppetserver rpms]# /etc/rc.d/init.d/mysqld restart
[root@puppetserver ~]# chkconfig mysqld on
[root@puppetserver rpms]# mysqladmin -uroot password 123.com
[root@puppetserver rpms]# mysql –p123.com
mysql> create database dashboard character set utf8;
mysql> grant all on dashboard.* to 'dashboard'@'localhost' identified by "123.com";
mysql> flush privileges;
[root@puppetserver rpms]# mysql -udashboard -p123.com #测试账号是否创建成功
…
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql>
```

3.2 优化数据库配置文件 my.cnf

```
[root@puppetserver rpms]# vim /etc/my.cnf
[mysqld]
# Allowing 32MB allows an occasional 17MB row with plenty of spare room
max_allowed_packet = 32M
…
[root@puppetserver rpms]# /etc/rc.d/init.d/mysqld restart #重启 MySQL 生效
Stopping mysqld:                                       [  OK  ]
Starting mysqld:                                       [  OK  ]
```

3.3 编辑 dashboard YAML 配置文件（`database.yml`）来指定数据库

```
[root@puppetserver rpms]# vim /usr/share/puppet-dashboard/config/database.yml
production:
  database: dashboard
  username: dashboard
  password: 123.com
  encoding: utf8
  adapter: mysql
…
```

3.4 填充数据库

```
[root@puppetserver ~]# cd /usr/share/puppet-dashboard/
[root@puppetserver puppet-dashboard]# rake gems:refresh_specs
[root@puppetserver puppet-dashboard]# rake RAILS_ENV=production db:migrate #环境变量
RAILS_ENV=production 告诉 Ruby on Rails 我们工作在生产环境。每次你运行一个 rake 命令都需要
使用合适的环境值来设置 RAILS_ENV 环境变量
```

3.5 查看是否导入成功

```
[root@puppetserver puppet-dashboard]# mysql -udashboard -p123.com
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 5
Server version: 5.1.66 Source distribution
Copyright (c) 2000, 2012, Oracle and/or its affiliates. All rights reserved.
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql> use dashboard;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A
Database changed
mysql> show tables;
+-------------------------------+
| Tables_in_dashboard           |
+-------------------------------+
| delayed_job_failures          |
| delayed_jobs                 |
| metrics                      |
| node_class_memberships       |
| node_classes                 |
| node_group_class_memberships |
| node_group_edges             |
| node_group_memberships       |
| node_groups                  |
| nodes                        |
| old_reports                  |
| parameters                   |
| report_logs                  |
| reports                      |
| resource_events              |
| resource_statuses            |
| schema_migrations            |
| timeline_events              |
+------------------------------+
18 rows in set (0.00 sec)
```

4 启动并运行 Dashboard（WEBrick 方式）

WEBrick 有助于快速使用 Dashboard，不过它不能很好地进行扩展，并且当有许多 Puppet agent 向 Dashboard 进行报告时，它的性能会非常差，因此不推荐使用。

4.1 关闭 httpd 服务

```
[root@puppetserver puppet-dashboard]# /etc/rc.d/init.d/httpd stop #之前配置过使用httpd 运行 puppetmaster，需要关闭
Stopping httpd:                                            [  OK  ]
```

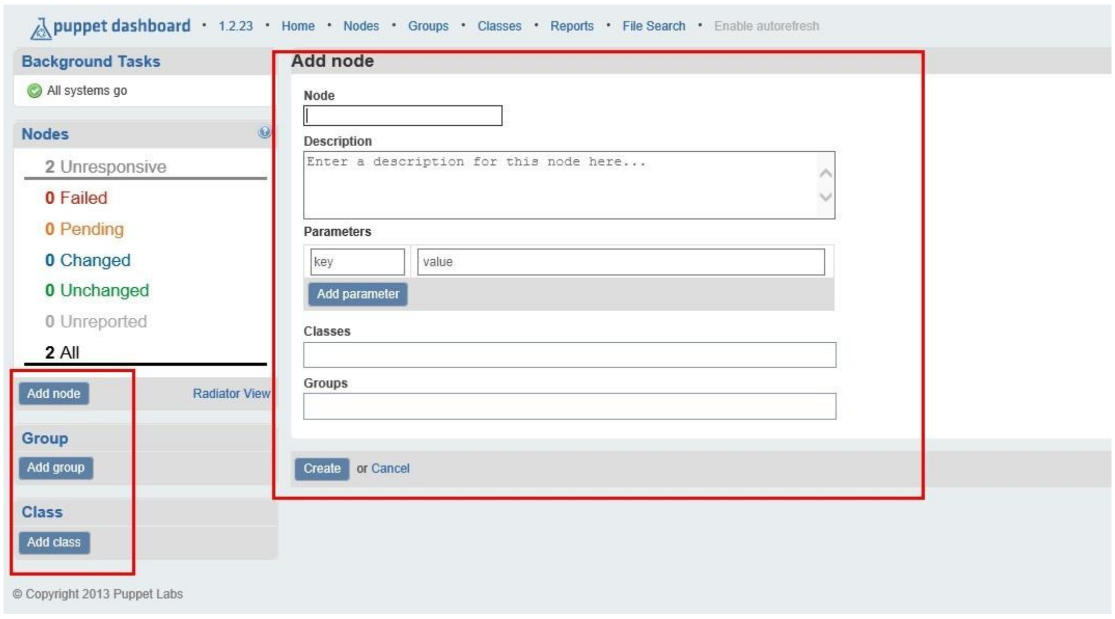
4.2 启动 puppetmaster 服务

```
[root@puppetserver puppet-dashboard]# /etc/rc.d/init.d/puppetmaster start
Starting puppetmaster:                                     [  OK  ]
```

4.3 启动 puppet-dashboard 服务

```
[root@puppetserver puppet-dashboard]# /etc/rc.d/init.d/puppet-dashboard start #启动dashboard
Starting Puppet Dashboard: => Booting WEBrick
=> Rails 2.3.17 application starting on http://0.0.0.0:3000
                                                         [  OK  ]
```

4.4 通过浏览器访问 http://192.168.100.110:3000



5 启动并运行 Dashboard（Passenger 方式）

5.1 使用 Ruby Gem 安装 Passenger

```
[root@puppetserver etc]# yum install ruby-devel ruby-libs rubygems libcurl-devel
[root@puppetserver etc]# yum install httpd httpd-devel apr-util-devel apr-devel
mod_ssl
[root@puppetserver repos]# gem install --local passenger-4.0.19.gem #自动解决依赖关
系，进入 gem 包目录进行安装
Building native extensions.  This could take a while...
Successfully installed rake-10.0.1
Successfully installed daemon_controller-1.1.5
Successfully installed rack-1.5.2
Successfully installed passenger-4.0.19
```

5.2 配置虚拟主机和 passenger

```
[root@puppetserver puppet-dashboard]# vim /etc/httpd/conf.d/passenger.conf
LoadModule passenger_module /usr/lib/ruby/gems/1.8/gems/passenger-
4.0.19/buildout/apache2/mod_passenger.so
<IfModule mod_passenger.c>
   PassengerRoot /usr/lib/ruby/gems/1.8/gems/passenger-4.0.19
   PassengerRuby /usr/bin/ruby
   PassengerHighPerformance on
   PassengerMaxPoolSize 12
   PassengerPoolIdleTime 1500
   PassengerStatThrottleRate 120
 # RailsAutoDetect On
</IfModule>
```

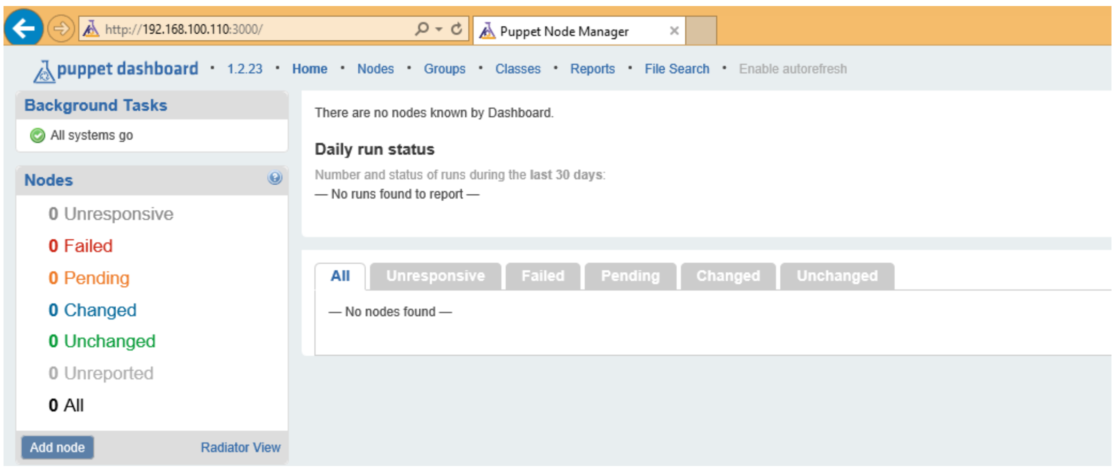
```
Listen 8141
<VirtualHost *:8141>
        DocumentRoot "/usr/share/puppet-dashboard/public/"
        <Directory "/usr/share/puppet-dashboard/public/">
                Options None
                AllowOverride AuthConfig
                Order allow,deny
                allow from all
        </Directory>
        ErrorLog /var/log/httpd/dashboard.error.log
        LogLevel warn
        CustomLog /var/log/httpd/dashboard.access.log combined
</VirtualHost>
```

5.3 启动相关服务

```
[root@puppetserver ~]# /etc/rc.d/init.d/puppetmaster stop  #停掉 puppetmaster 服务
Stopping puppetmaster:                                     [  OK  ]
[root@puppetserver ~]# /etc/rc.d/init.d/httpd restart
```

5.4 通过浏览器访问测试

http://192.168.100.110:8141/


http://192.168.100.110:3000/
Puppet Node Manager
puppet dashboard • 1.2.23 • Home • Nodes • Groups • Classes • Reports • File Search • Enable autorefresh
Background Tasks
All systems go
Nodes
Add node
Radiator View
All
Unresponsive
Failed
Pending
Changed
Unchanged

0 Unresponsive
0 Failed
0 Pending
0 Changed
0 Unchanged
0 Unreported
0 All

There are no nodes known by Dashboard.

Daily run status

Number and status of runs during the last 30 days:
— No runs found to report —

— No nodes found —

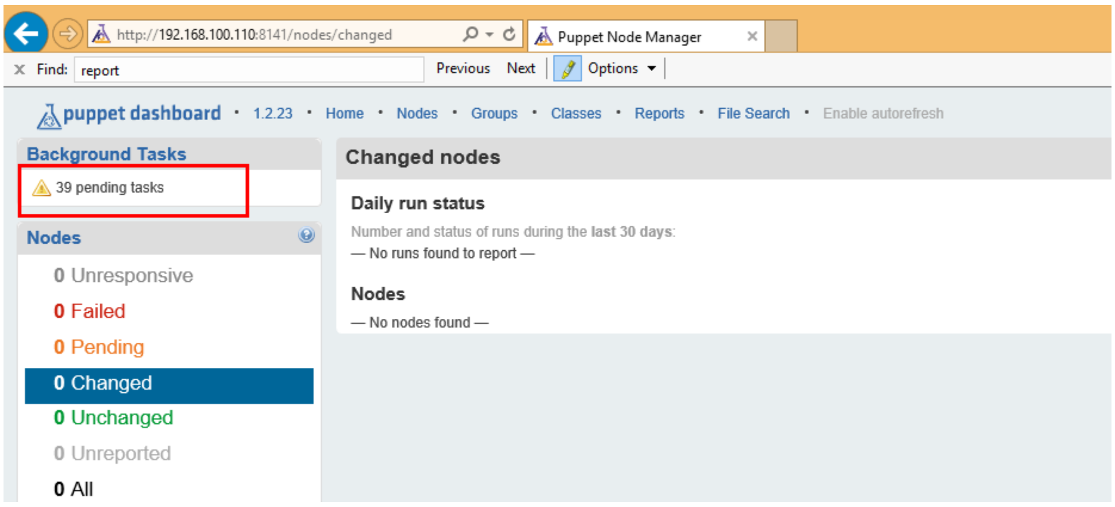
6 集成 Puppet Dashboard

6.1 手工导入现有的报告（方式一）

```
[root@puppetserver ~]# cd /usr/share/puppet-dashboard/
[root@puppetserver puppet-dashboard]# rake RAILS_ENV=production reports:import  #导
入已经存在的报告
Importing 39 reports from /var/lib/puppet/reports in the background
```

```
Importing:     100%
|##################################################################################
#| Time: 00:00:00
39 of 39 reports queued
```

备注：默认节点报告会在`/var/lib/puppet/reports/`产生，如果路径发生变化，导入报告时需要在后面加上`“REPORT_DIR=report 路径”`，reports 更改路径可在`puppet.conf`中设置参数“reportdir = 新路径”，这种方式不够实时。



Changed nodes

Daily run status

Number and status of runs during the last 30 days:

— No runs found to report —

Nodes

— No nodes found —

6.2 配置实施汇总 puppet 报告（方式二）

```
[root@agent1 ~]# vim /etc/puppet/puppet.conf  #配置 agent 节点自动发送报告
 [agent]
report = true  #从 2.7.0 版本开始，报告系统会默认开启，不需要配置
…
[root@puppetserver puppet-dashboard]# vim /etc/puppet/puppet.conf
[main]
    reports = http  #定义为 http 报告处理器，除此之外还有 store，log，tagmail，rrdgraph 等
报告处理器
reporturl = http://172.16.200.100:8141/reports #http 报告处理器将 puppet 报告发送到一个
HTTP URL 和端口（Dashboard 位置）。Puppet 报告以被转储为 HTTP Poort 形式的 YAML 格式进行发
送。
…
[root@puppetserver public]# /etc/rc.d/init.d/httpd restart
```

6.3 开启后台处理报告进程

```
[root@puppetserver puppet-dashboard]# rake RAILS_ENV=production jobs:work &  #运行
“Delayed Job Workers”，使其在后台为我们处理报告日志
[1] 28651
[root@puppetserver puppet-dashboard]# [Worker(host:puppetserver.kisspuppet.com
pid:28651)] Starting job worker
```

```
[Worker(host:puppetserver.kisspuppet.com pid:28651)] Report.create_from_yaml_file
completed after 0.2674
```

```
[Worker(host:puppetserver.kisspuppet.com pid:28651)] Report.create_from_yaml_file
completed after 0.1725
[Worker(host:puppetserver.kisspuppet.com pid:28651)] Report.create_from_yaml_file
completed after 0.1345
[Worker(host:puppetserver.kisspuppet.com pid:28651)] Report.create_from_yaml_file
completed after 0.1772
[Worker(host:puppetserver.kisspuppet.com pid:28651)] Report.create_from_yaml_file
completed after 0.1397
…
[Worker(host:puppetserver.kisspuppet.com pid:28651)] 42 jobs processed at 5.9487
j/s, 0 failed ...
```

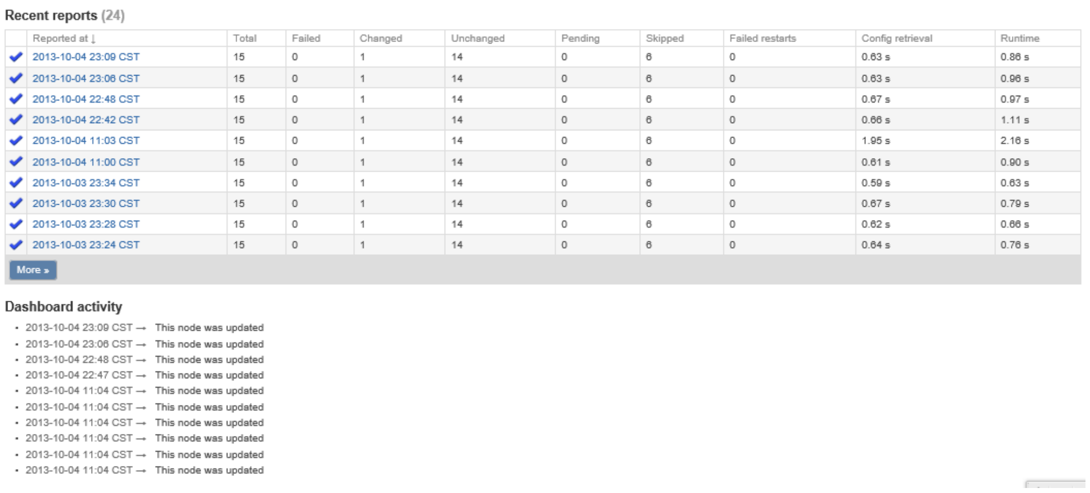
6.4 修改 dashboard 时区

Dashboard 默认时区为 UTC 格式，我们这里需要更改为 `CST（Asia/Shanghai）`格式

```
[root@puppetserver ~]# vim /usr/share/puppet-dashboard/config/settings.yml
time_zone: 'Asia/Shanghai'
…
**备注**：设置的 settings.yml 会覆盖掉 config/environment.rb 中对应的配置项
（config.time_zone = 'UTC'）
```

6.5 显示报告

通过 http://192.168.100.110:8141/ 及时查看节点更新的报告信息，可以看到两个节点 agent1 和 agent2，默认显示时间为 CST 格式，除此之外还可以看到某一个节点在某一个时刻的更新报告和运行曲线图。

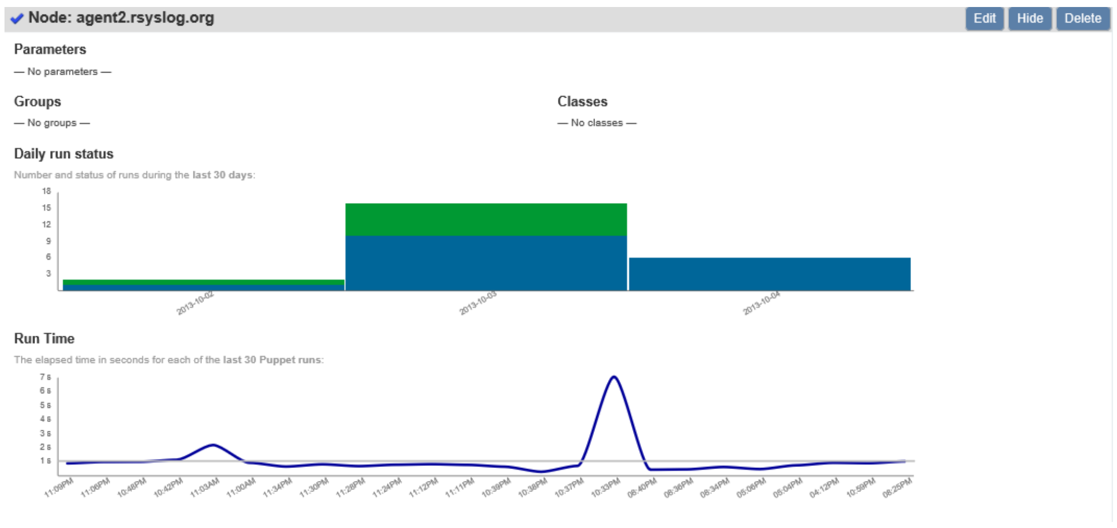


Recent reports (24)

	Reported at ↓	Total	Failed	Changed	Unchanged	Pending	Skipped	Failed restarts	Config retrieval	Runtime
✓	2013-10-04 23:09 CST	15	0	1	14	0	6	0	0.63 s	0.86 s
✓	2013-10-04 23:06 CST	15	0	1	14	0	6	0	0.63 s	0.96 s
✓	2013-10-04 22:48 CST	15	0	1	14	0	6	0	0.67 s	0.97 s
✓	2013-10-04 22:42 CST	15	0	1	14	0	6	0	0.66 s	1.11 s
✓	2013-10-04 11:03 CST	15	0	1	14	0	6	0	1.95 s	2.16 s
✓	2013-10-04 11:00 CST	15	0	1	14	0	6	0	0.61 s	0.90 s
✓	2013-10-03 23:34 CST	15	0	1	14	0	6	0	0.59 s	0.63 s
✓	2013-10-03 23:30 CST	15	0	1	14	0	6	0	0.67 s	0.79 s
✓	2013-10-03 23:28 CST	15	0	1	14	0	6	0	0.62 s	0.66 s
✓	2013-10-03 23:24 CST	15	0	1	14	0	6	0	0.64 s	0.76 s

Dashboard activity

- 2013-10-04 23:09 CST → This node was updated
- 2013-10-04 23:06 CST → This node was updated
- 2013-10-04 22:48 CST → This node was updated
- 2013-10-04 22:47 CST → This node was updated
- 2013-10-04 11:04 CST → This node was updated
- 2013-10-04 11:04 CST → This node was updated
- 2013-10-04 11:04 CST → This node was updated
- 2013-10-04 11:04 CST → This node was updated
- 2013-10-04 11:04 CST → This node was updated
- 2013-10-04 11:04 CST → This node was updated



✓ Node: agent2.rsyslog.org

Parameters

— No parameters —

Groups

— No groups —

Classes

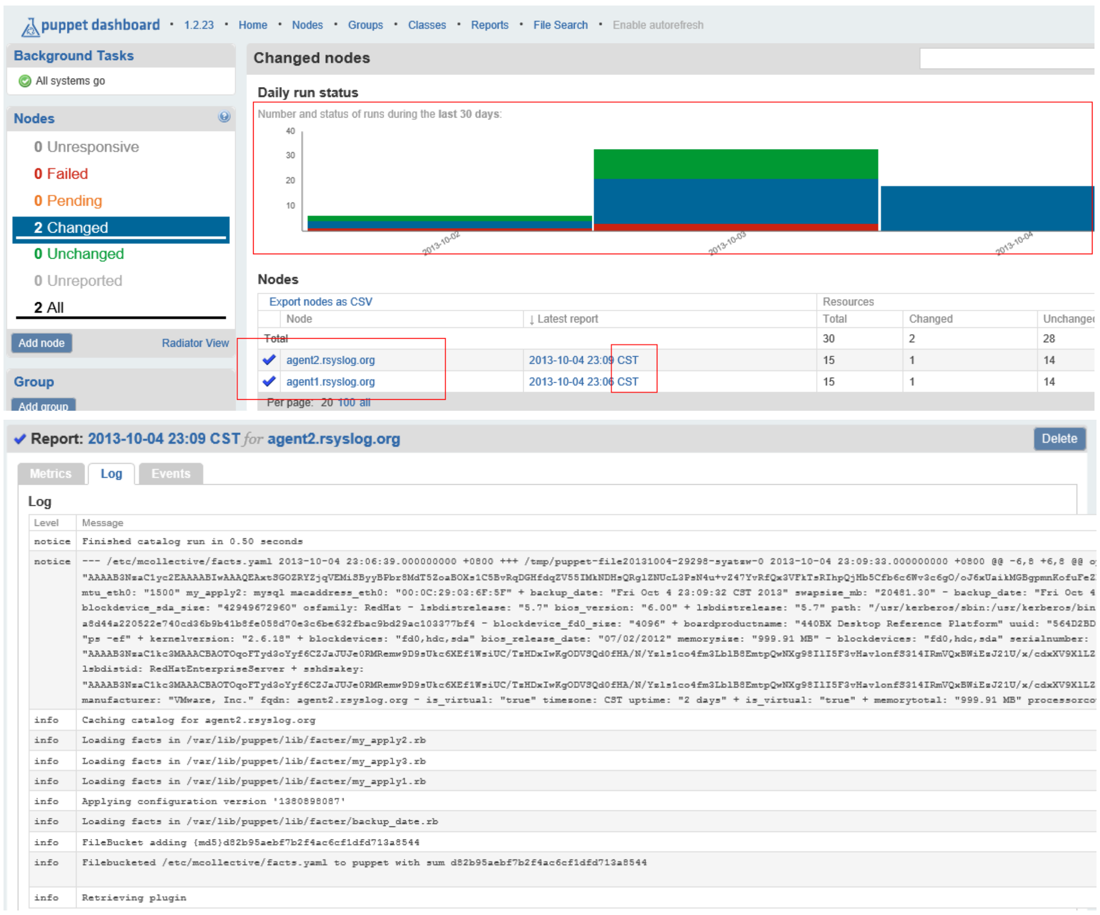
— No classes —

Daily run status

Number and status of runs during the **last 30 days**:

Run Time

The elapsed time in seconds for each of the **last 30 Puppet runs**:



	Node	↓ Latest report	Resources Total	Changed	Unchanged
Total			30	2	28
✓	agent2.rsyslog.org	2013-10-04 23:09 CST	15	1	14
✓	agent1.rsyslog.org	2013-10-04 23:06 CST	15	1	14

✓ Report: 2013-10-04 23:09 CST *for* agent2.rsyslog.org

Log

Level	Message
notice	Finished catalog run in 0.50 seconds
notice	--- /etc/mcollective/facts.yaml 2013-10-04 23:06:39.000000000 +0800 +++ /tmp/puppet-file20131004-29298-syatsw-0 2013-10-04 23:09:33.000000000 +0800 @@ -6,8 +6,8 @@ o "AAAAB3NzaC1yc2EAAAABIwAAAQEAxtSGOZRYZjqVEMiSByyBPbr8MdT5ZoaBOXs1C5BvRqDGHfdqZV55IMkNDHsQRglZNUcL3PsN4u+vZ47YvRfQx3VFkTsRIhpQjHb5Cfb6c6Wv3c6gO/oJ6xUaikMGBgpmnKofuFeZ mtu_eth0: "1500" my_apply2: mysql macaddress_eth0: "00:0C:29:03:6F:5F" + backup_date: "Fri Oct 4 23:09:32 CST 2013" swapsize_mb: "20481.30" - backup_date: "Fri Oct 4 blockdevice_sda_size: "42949672960" osfamily: RedHat - lsbdistrelease: "5.7" bios_version: "6.00" + lsbdistrelease: "5.7" path: "/usr/kerberos/sbin:/usr/kerberos/bin a8d44a220522e740cd36b9b41b8fe058d70e3c6be632fbac9bd29ac103377bf4 - blockdevice_fd0_size: "4096" + boardproductname: "440BX Desktop Reference Platform" uuid: "564D2BD "ps -ef" + kernelversion: "2.6.18" + blockdevices: "fd0,hdc,sda" bios_release_date: "07/02/2012" memorysize: "999.91 MB" - blockdevices: "fd0,hdc,sda" serialnumber: "AAAAB3NzaC1kc3MAAACBAOTOqoFTyd3oYyf6CZJaJUJe0RMRemw9D9sUkc6XEf1WsiUC/TzHDxIwKgODVSQd0fHA/N/Yzls1co4fm3LblB8EmtpQwNXg98IlI5F3vHavlonfS314IRmVQxBWiEzJ21U/x/cdxXV9X1LZ lsbdistid: RedHatEnterpriseServer + sshdsakey: "AAAAB3NzaC1kc3MAAACBAOTOqoFTyd3oYyf6CZJaJUJe0RMRemw9D9sUkc6XEf1WsiUC/TzHDxIwKgODVSQd0fHA/N/Yzls1co4fm3LblB8EmtpQwNXg98IlI5F3vHavlonfS314IRmVQxBWiEzJ21U/x/cdxXV9X1LZ manufacturer: "VMware, Inc." fqdn: agent2.rsyslog.org - is_virtual: "true" timezone: CST uptime: "2 days" + is_virtual: "true" + memorytotal: "999.91 MB" processorco
info	Caching catalog for agent2.rsyslog.org
info	Loading facts in /var/lib/puppet/lib/facter/my_apply2.rb
info	Loading facts in /var/lib/puppet/lib/facter/my_apply3.rb
info	Loading facts in /var/lib/puppet/lib/facter/my_apply1.rb
info	Applying configuration version '1380898087'
info	Loading facts in /var/lib/puppet/lib/facter/backup_date.rb
info	FileBucket adding {md5}d82b95aebf7b2f4ac6cf1dfd713a8544
info	Filebucketed /etc/mcollective/facts.yaml to puppet with sum d82b95aebf7b2f4ac6cf1dfd713a8544
info	Retrieving plugin

7 自定义报告

7.1 编写外部报告处理器

使用现有的被存储的报告，就是那些 yaml 文件，可以通过设置 `puppet.conf` 中 `reports = store` 进行收集。然后编写一个外部的处理器来处理这些信息，例如绘图或者将他们存储在外部数据库。这也是 Puppet Dashboard 中的报告输入进程的工作原理。这些外部的报告处理器可以很简单地使用 Ruby 进行编写，以便使用 Ruby 反序列化 YAML 文件的能力以及使用生成的对象。你可以使用任何支持导入第三方 ymal 数据的工具。

7.2 编写内部报告处理器

编写自定义报告处理器并将它添加到 Puppet。和 fact、函数、类型及提供者的插件不同，Puppet 没有提供一个自动分发自定义报告的方法。

7.2.1 现有报告处理器信息

```
[root@puppetserver ~] # ls /usr/lib/ruby/site_ruby/1.8/puppet/reports
http.rb  log.rb  rrdgraph.rb  store.rb  tagmail.rb
[root@puppetserver reports]# cat http.rb  #查看 http 报告处理器内容
require 'puppet'
require 'net/http'
require 'uri'
```

```
Puppet::Reports.register_report(:http) do

  desc <<-DESC
  Send report information via HTTP to the `reporturl`. Each host sends
  its report as a YAML dump and this sends this YAML to a client via HTTP POST.
  The YAML is the body of the request.
  DESC

  def process
    url = URI.parse(Puppet[:reporturl])
    req = Net::HTTP::Post.new(url.path)
    req.body = self.to_yaml
    req.content_type = "application/x-yaml"
    Net::HTTP.new(url.host, url.port).start {|http|
      response = http.request(req)
      unless response.kind_of?(Net::HTTPSuccess)
        Puppet.err "Unable to submit report to #{Puppet[:reporturl].to_s}
[#{response.code}] #{response.msg}"
      end
    }
  end
end
```

7.2.2 自定义摘要报告处理器

7.2.2.1 进入 reports 目录编写自定义 summary.rb 报告处理器

```
[root@puppetserver ~]# cd /usr/lib/ruby/site_ruby/1.8/puppet/reports
[root@puppetserver reports]# vim summary.rb
require 'puppet'
Puppet::Reports.register_report(:summary) do
  desc <<-DESC
  Send summary report information to the report directory.
  DESC
  def process
    client = self.host
    summary = self.summary
    dir = File.join(Puppet[:reportdir],client)
    client = self.host
    file = "summary.txt"
    destination = File.join(dir,file)
    File.open(destination,"w") do |f|
        f.write(summary)
      end
  end
```

```
end
```

7.2.2.2 将报告处理器的名字加入 puppet.conf 中，并重新启动 httpd 服务

```
[root@puppetserver ~]# vim /etc/puppet/puppet.conf
[main]
reports = http,summary
…
[root@puppetserver ~]# /etc/rc.d/init.d/httpd restart
Stopping httpd:                                            [  OK  ]
Starting httpd:                                            [  OK  ]
```

7.2.2.3 使用 mco 命令触发更新节点 agent1

```
[root@puppetserver ~]# mco puppet -v runonce  mco facts -v --with-fact
hostname='agent1'
Discovering hosts using the mc method for 2 second(s) .... 1
 * [ ============================================================> ] 1 / 1
agent1.kisspuppet.com                    : OK
    {:summary=>     "Started a background Puppet run using the 'puppet agent --
onetime --daemonize --color=false --splay --splaylimit 30' command"}
---- rpc stats ----
          Nodes: 1 / 1
     Pass / Fail: 1 / 0
      Start Time: Fri Oct 04 12:54:50 +0800 2013
  Discovery Time: 2005.27ms
      Agent Time: 1118.41ms
      Total Time: 3123.68ms
```

7.2.2.4 查看新生成的报告信息

```
[root@puppetserver ~]#  cd /var/lib/puppet/reports/agent1.kisspuppet.com/
[root@puppetserver agent1.kisspuppet.com]# cat summary.txt
Changes:
           Total: 1
Events:
           Total: 1
         Success: 1
Resources:
     Out of sync: 1
         Changed: 1
           Total: 15
         Skipped: 6
Time:
      Filebucket: 0.00
         Package: 0.00
            File: 0.11
         Service: 0.12
```

```
   Config retrieval: 1.29
             Total: 1.52
          Last run: 1380861882
Version:
            Config: 1380861878
            Puppet: 2.7.23
```

在整个报告处理器中，我们定义了一个叫做 process 的方法来承载处理器的核心逻辑。我们从报告中提取了一些信息：使用 `self.host` 方式提取了主机名，使用 summary 方式提取了变更的摘要。还可以使用 `self.logs` 和 `self.metrics` 方式来访问报告中的日子以及度量值。 我们同时还将报告的摘要输出了报告目录下对应的以 Puppet agent 主机名命名的目录中，报告目录的位置是由 reportdir 配置的值来指定的，默认在/var/lib/puppet/reports/目录下。

备注：更多报告处理器信息请访问

```
现有报告处理器 https://github.com/puppetlabs/puppet/tree/master/lib/puppet/reports
报告参考 http://docs.puppetlabs.com/references/latest/report.html#http
报告及报告系统 http://docs.puppetlabs.com/guides/reporting.html
```

——

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

——