

puppet 端口负载均衡配置（nginx+mongrel）

当 puppetmaster 管理的主机越来越多时，puppetmaster 本身性能会存在性能瓶颈问题，除了增加服务器扩充 puppetmaster 的数量增加 puppetmaster 整体性能外，也可以通过单台扩充 puppetmaster 的进程数来增加 puppetmaster 的性能。

以下是通过 nginx 负载均衡 puppetmaster 的进程，由 nginx 向所有 puppetagent 提供认证服务，除此之外的其他 puppetmaster 功能的实现由 nginx 转向 puppetmaster 其中一个进程去处理即可。而 nginx 的 upstream 字段里面所包含的地址填写为 127.0.0.1 指向 puppetmaster 进程，提高了安全性。

1.1 安装相关包

```
[root@puppetserver yum.repos.d]# yum install rubygem-mongrel nginx
```

1.2 修改 puppet 端口

```
[root@puppetserver yum.repos.d]# vim /etc/sysconfig/puppetmaster
PUPPETMASTER_PORTS=( 18140 18141 18142 18143 )
PUPPETMASTER_EXTRA_OPTS="--servertype=mongrel --
ssl_client_header=HTTP_X_SSL_SUBJECT"
```

1.3 配置 nginx

```
[root@puppetserver nginx]# vim nginx.conf
user          nginx nginx;
worker_processes 4;
error_log     /var/log/puppet/nginx-puppet.log notice;
pid           /var/run/nginx.pid;

events {
    worker_connections 1024;
}
http {
    default_type      application/octet-stream;

    sendfile          on;
    tcp_nopush        on;

    keepalive_timeout 65;
```

```
tcp_nodelay          on;

large_client_header_buffers 16    4k;
proxy_buffers         128        4k;

upstream puppetmaster {
    server 127.0.0.1:18140;
    server 127.0.0.1:18141;
    server 127.0.0.1:18142;
    server 127.0.0.1:18143;
}

server {
    listen 8140;
    root    /etc/puppet;

    ssl on;
    ssl_session_timeout 5m;
    ssl_certificate
/var/lib/puppet/ssl/certs/puppetserver.kisspuppet.com.pem;
    ssl_certificate_key
/var/lib/puppet/ssl/private_keys/puppetserver.kisspuppet.com.pem;
    ssl_client_certificate /var/lib/puppet/ssl/ca/ca.crt.pem;
    ssl_crl /var/lib/puppet/ssl/ca/ca.crl.pem;
    ssl_ciphers SSLv2:-LOW:-EXPORT:RC4+RSA;
    ssl_verify_client optional;

    location / {
        proxy_pass http://puppetmaster;
        proxy_redirect off;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Client-Verify $ssl_client_verify;
        proxy_set_header X-Client-DN $ssl_client_s_dn;
        proxy_set_header X-SSL-Subject $ssl_client_s_dn;
        proxy_set_header X-SSL-Issuer $ssl_client_i_dn;
        proxy_read_timeout 65;
    }
}
```

1.4 启动 puppetmaster 和 nginx 服务

```
[root@linuxmaster1poc ~]# /etc/rc.d/init.d/nginx restart
Stopping nginx: [FAILED]
Starting nginx: [ OK ]
[root@linuxmaster1poc ~]# /etc/rc.d/init.d/puppetmaster start
Starting puppetmaster:
Port: 18140 [ OK ]
Port: 18141 [ OK ]
Port: 18142 [ OK ]
Port: 18143 [ OK ]
```

1.5 查看监听端口

```
[root@linuxmaster1poc ~]# netstat -nlp | grep 814
tcp        0      0 0.0.0.0:8140          0.0.0.0:*             LISTEN
6224/nginx
tcp        0      0 127.0.0.1:18140       0.0.0.0:*             LISTEN
6271/ruby
tcp        0      0 127.0.0.1:18141       0.0.0.0:*             LISTEN
6312/ruby
tcp        0      0 127.0.0.1:18142       0.0.0.0:*             LISTEN
6351/ruby
tcp        0      0 127.0.0.1:18143       0.0.0.0:*             LISTEN
6390/ruby
```

1.6 通过日志/var/log/nginx/access.log 进行查看

```
[root@linuxmaster1poc nodes]# tailf /var/log/nginx/access.log
192.168.100.127 - - [25/Nov/2013:16:42:49 +0800] "POST
/production/catalog/agent2.kisspuppet.com HTTP/1.1" 200 570 "-" "-"
192.168.100.127 - - [25/Nov/2013:16:42:52 +0800] "PUT
/production/report/agent2.kisspuppet.com HTTP/1.1" 200 58 "-" "-"
192.168.100.126 - - [25/Nov/2013:16:42:54 +0800] "GET
/production/file_metadatas/plugins?links=manage&checksum_type=md5&&ignore=---
+%0A++-%22svn%22%0A++-+CVS%0A++-%22git%22&recurse=true HTTP/1.1" 404 56 "-" "-"
192.168.100.126 - - [25/Nov/2013:16:42:54 +0800] "GET
/production/file_metadata/plugins? HTTP/1.1" 404 36 "-" "-"
192.168.100.126 - - [25/Nov/2013:16:42:55 +0800] "POST
/production/catalog/agent1.kisspuppet.com HTTP/1.1" 200 570 "-" "-"
192.168.100.126 - - [25/Nov/2013:16:42:58 +0800] "PUT
/production/report/agent1.kisspuppet.com HTTP/1.1" 200 58 "-" "-"
192.168.100.125 - - [25/Nov/2013:16:43:07 +0800] "GET
```

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，投稿邮箱：admin@kisspuppet.com

微信公众号：“**puppet2014**”，可搜索加入，也可以扫描以下二维码

