# 秒杀场景下的运维架构

# 自我介绍

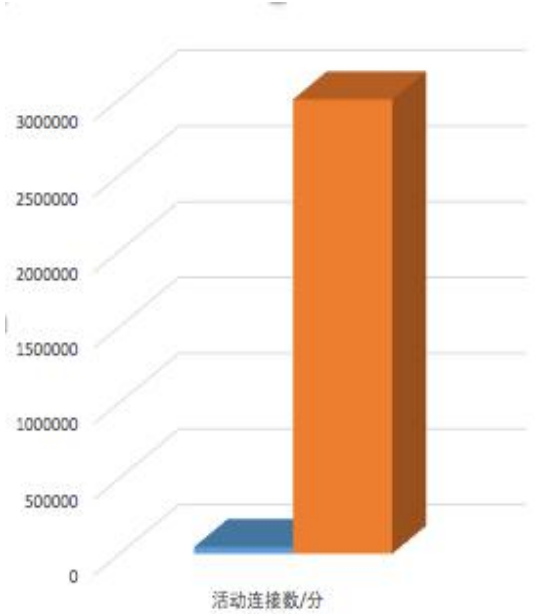
聚美优品高级运维经理

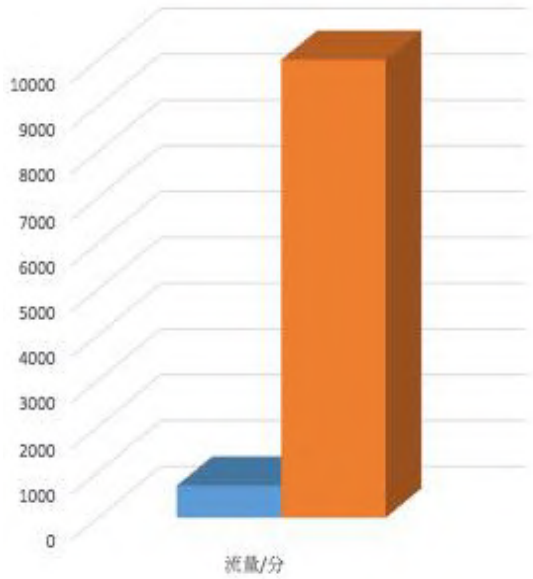
03年加入聚美，期间主要负责过运维监控系统，运维自动化系统，网站系统架构的优化与重构。

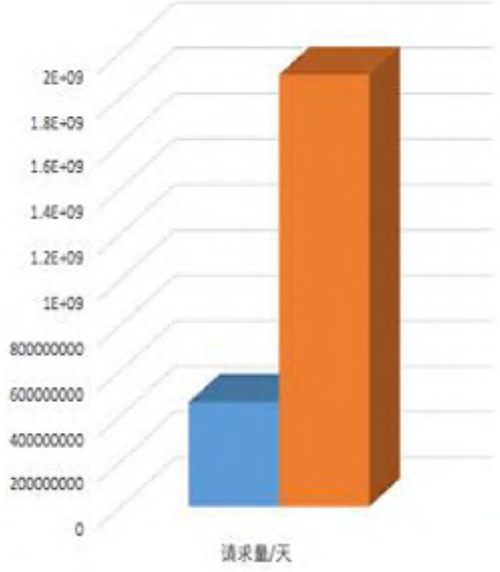
主导设计并参与整个运维平台建设与推广，带领运维团队完成了平台化的过渡。

# 面临的问题与挑战

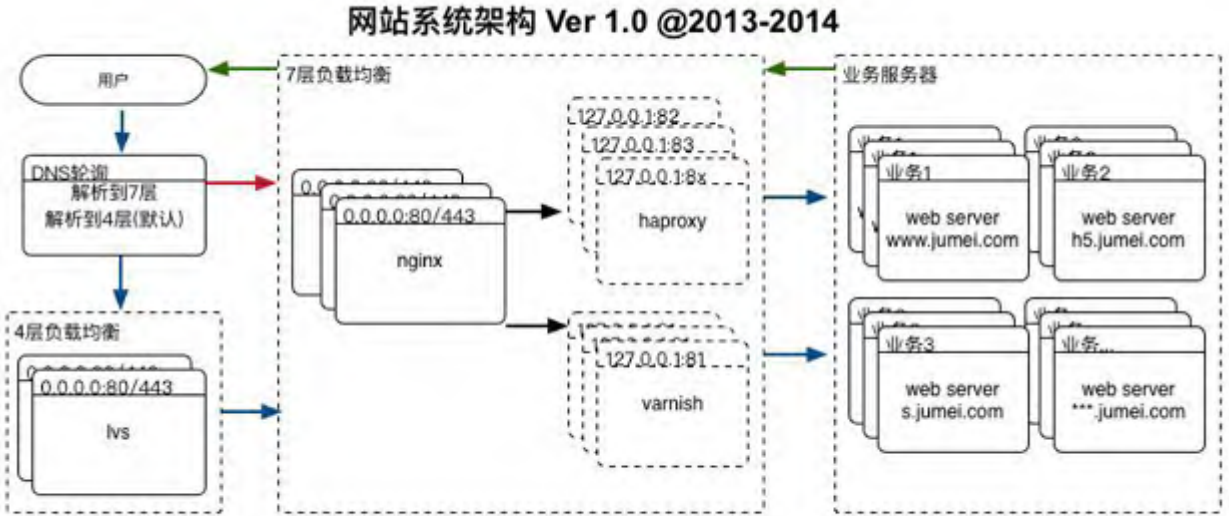
# 面临的问题与挑战


3000000
2500000
2000000
1500000
1000000
500000
0
活动连接数/分


10000
9000
8000
7000
6000
5000
4000
3000
2000
1000
0
流量/分

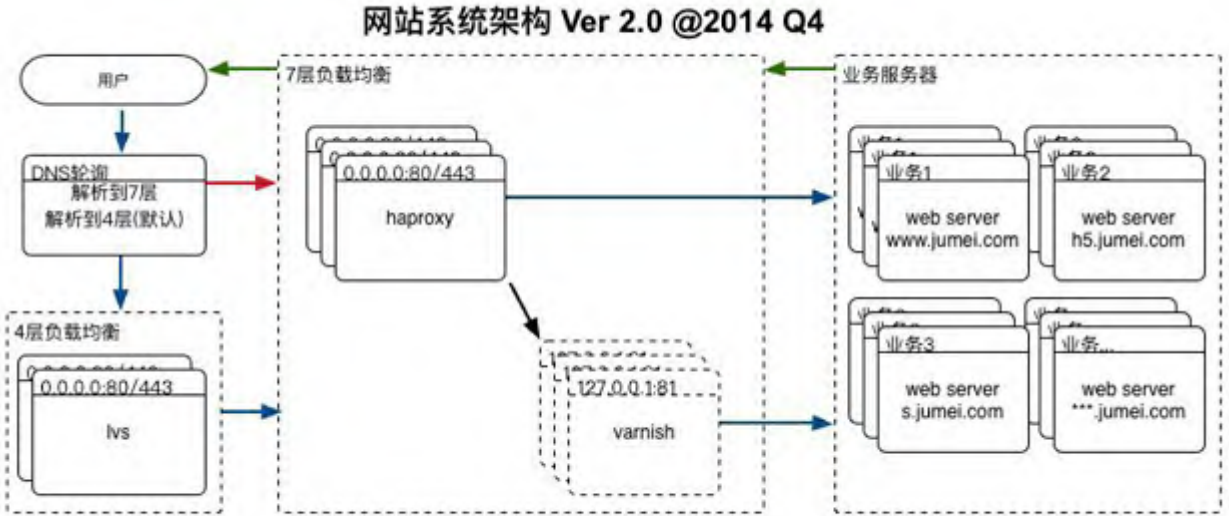

2E+09
1.8E+09
1.6E+09
1.4E+09
1.2E+09
1E+09
800000000
600000000
400000000
200000000
0
请求量/天

# 网站系统架构的演进



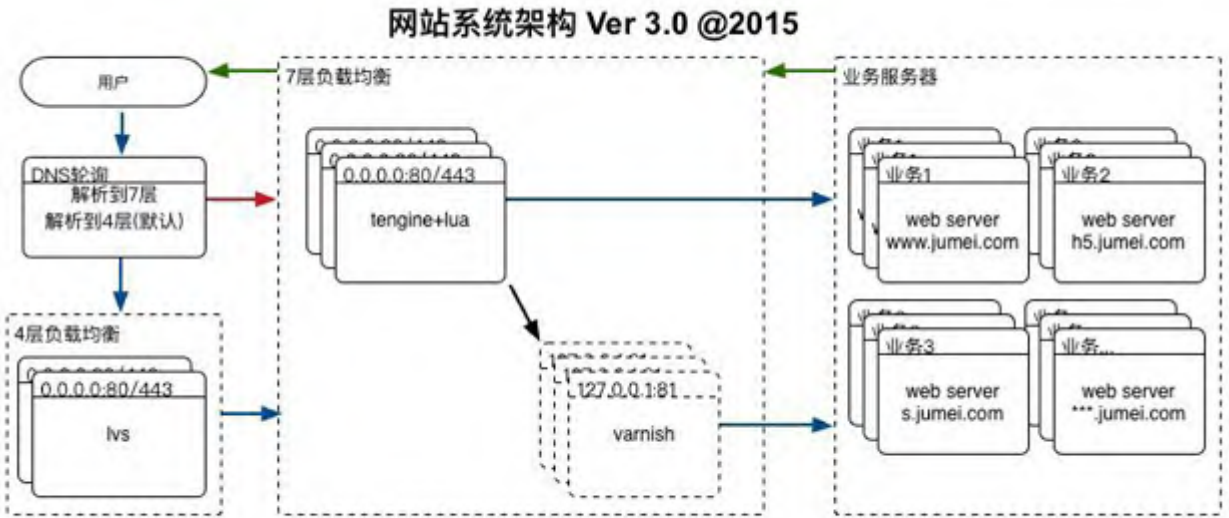

- Nginx
  - 安全策略
  - 业务策略
  - 缓存策略
- Haproxy
  - 业务检查基于单元用例
  - 调度算法
- 面临的一些问题
  - 两次7层负载带来的开销
  - 超时连接比较多
  - 维护成本高

# 网站系统架构的演进




- Haproxy
  - 安全策略
  - 业务策略
  - 缓存策略
- 优化及改进
  - 单机连接数40w
  - 超时连接减少1半
- 面临的一些问题
  - 安全策略没有完全实现
  - 维护成本仍较高

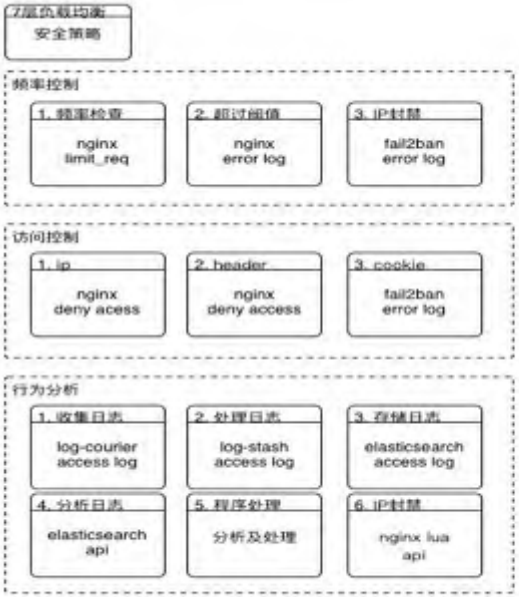
# 网站系统架构的演进

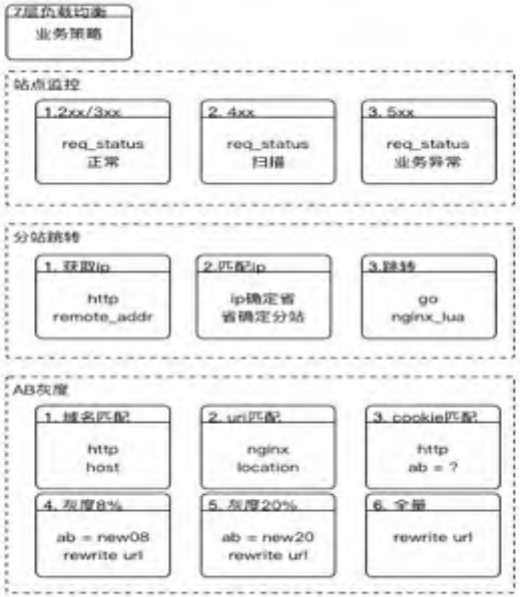


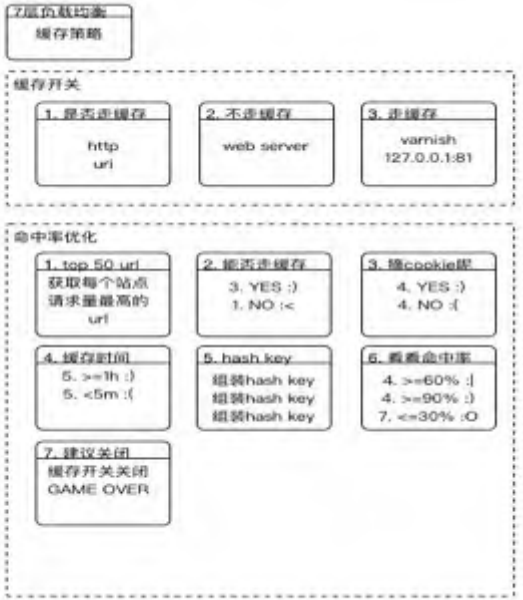

- Tengine
  - 安全策略
  - 业务策略
  - 缓存策略

- 优化及改进
  - 整体连接数减少1／3
  - cpu利用率降低
  - 业务检查过服务
  - 通过consul管理
  - WAF接口
  - 1台机器/1亿+请求/天

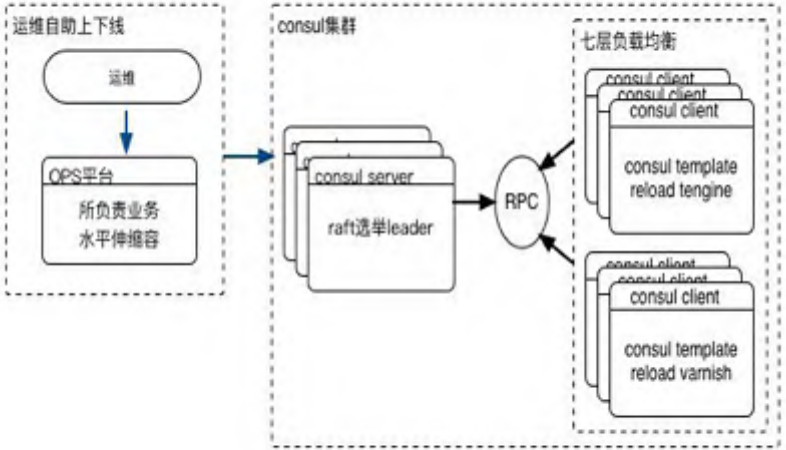
# 负载均衡系统核心功能


7层负载均衡
安全策略
频率控制
1. 频率检查
nginx
limit_req
2. 超过阀值
nginx
error log
3. IP封禁
fail2ban
error log
访问控制
1. ip
nginx
deny acess
2. header
nginx
deny access
3. cookie
fail2ban
error log
行为分析
1. 收集日志
log-courier
access log
2. 处理日志
log-stash
access log
3. 存储日志
elasticsearch
access log
4. 分析日志
elasticsearch
api
5. 程序处理
分析及处理
6. IP封禁
nginx lua
api

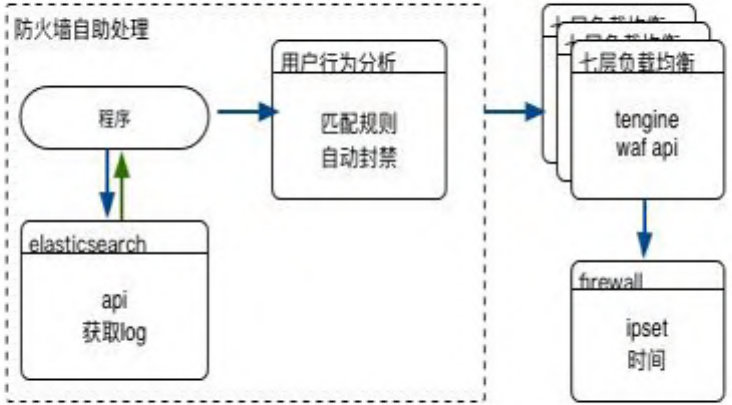

7层负载均衡
业务策略
站点监控
1. 2xx/3xx
req_status
正常
2. 4xx
req_status
扫描
3. 5xx
req_status
业务异常
分站跳转
1. 获取ip
http
remote_addr
2. 匹配ip
ip确定省
省确定分站
3. 跳转
go
nginx_lua
AB灰度
1. 域名匹配
http
host
2. uri匹配
nginx
location
3. cookie匹配
http
ab = ?
4. 灰度8%
ab = new08
rewrite url
5. 灰度20%
ab = new20
rewrite url
6. 全量
rewrite url


7层负载均衡
缓存策略
缓存开关
1. 是否走缓存
http
uri
2. 不走缓存
web server
3. 走缓存
varnish
127.0.0.1:81
命中率优化
1. top 50 url
获取每个站点
请求量最高的
url
2. 能否走缓存
3. YES :)
1. NO :<
3. 换cookie配
4. YES :)
4. NO :(
4. 缓存时间
5. >=1h :)
5. <5m :(
5. hash key
组装hash key
组装hash key
组装hash key
6. 看看命中率
4. >=60% :|
4. >=90% :)
7. <=30% :O
7. 建议关闭
缓存开关关闭
GAME OVER

# 如何做到弹性管理

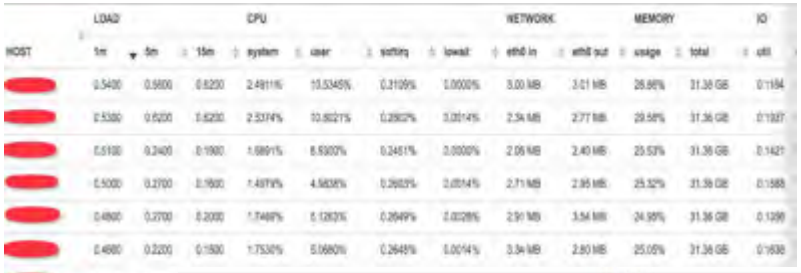

运维自助上下线
运维
OPS平台
所负责业务
水平伸缩容
consul集群
consul server
raft选举leader
RPC
七层负载均衡
consul client
consul template
reload tengine
consul client
consul template
reload varnish

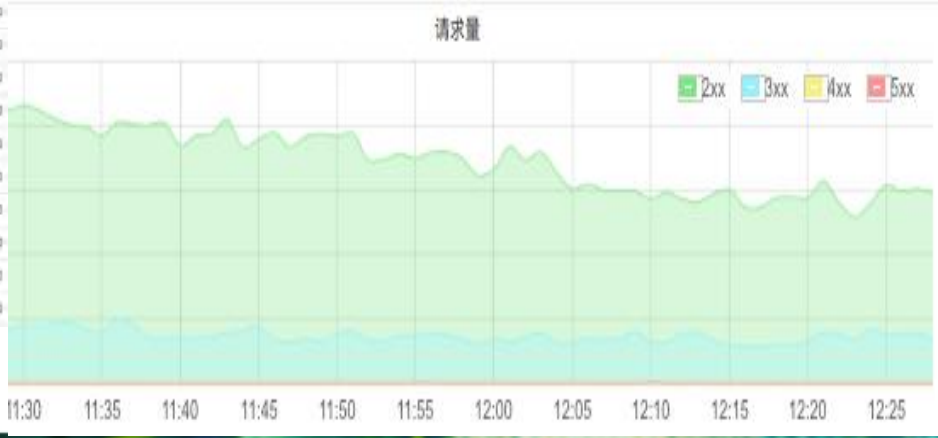
# 如何做到弹性管理

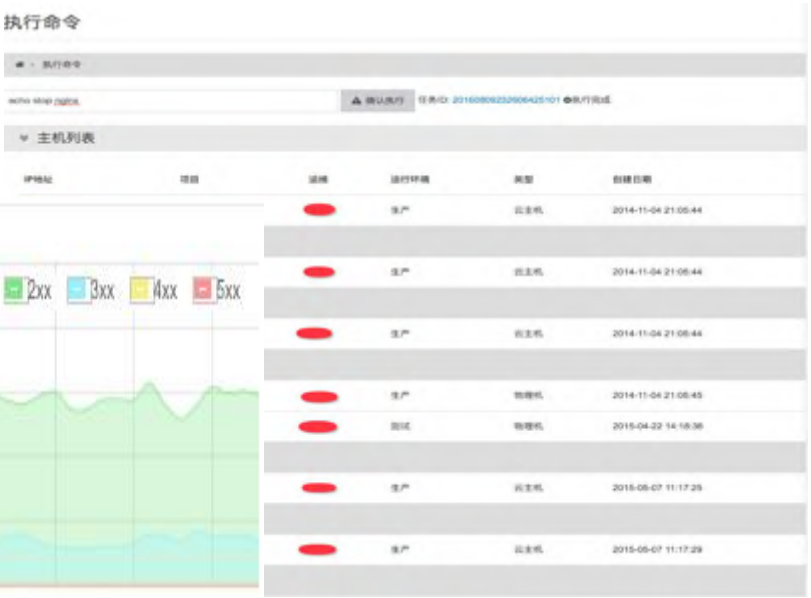

防火墙自助处理
程序
用户行为分析
匹配规则
自动封禁
七层负载均衡
tengine
waf api
elasticsearch
api
获取log
firewall
ipset
时间


防火墙策略

# 如何做到弹性管理


LOAD
CPU
NETWORK
MEMORY
IO
HOST
1m
5m
15m
system
user
softirq
iowait
eth0 in
eth0 out
usage
total
util

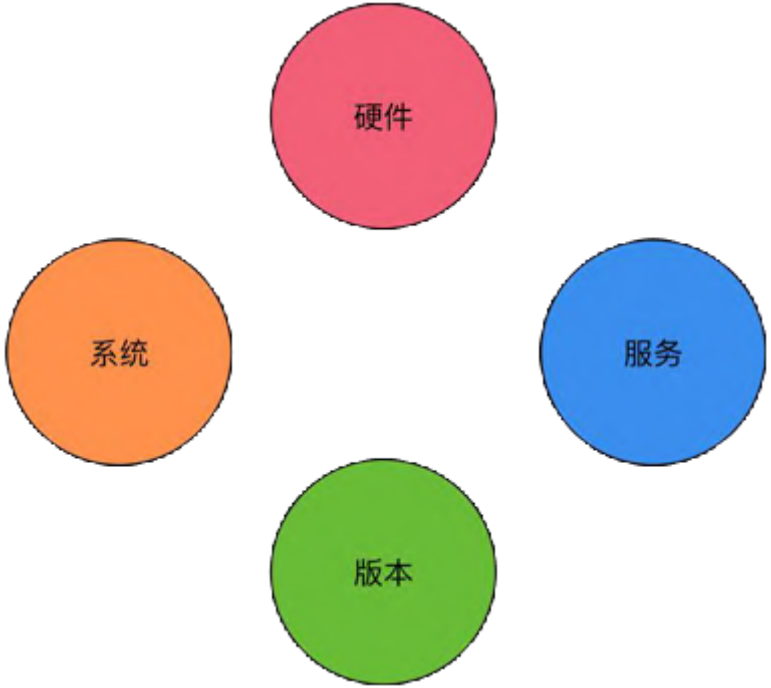

请求量
2xx
3xx
4xx
5xx
11:30
11:35
11:40
11:45
11:50
11:55
12:00
12:05
12:10
12:15
12:20
12:25


执行命令
主机列表

# 常见的几种应用场景

- 四层加七层，LVS + NGINX/HAPROXY/VARNISH
- 单四层，LVS
- 双四层，LVS + HAPROXY
- 单七层，HAPROXY/NGINX/VARNISH + HA
- 双七层，NGINX/HAPROXY + VARNISH

# 优化与建议




谢谢大家
Q&A