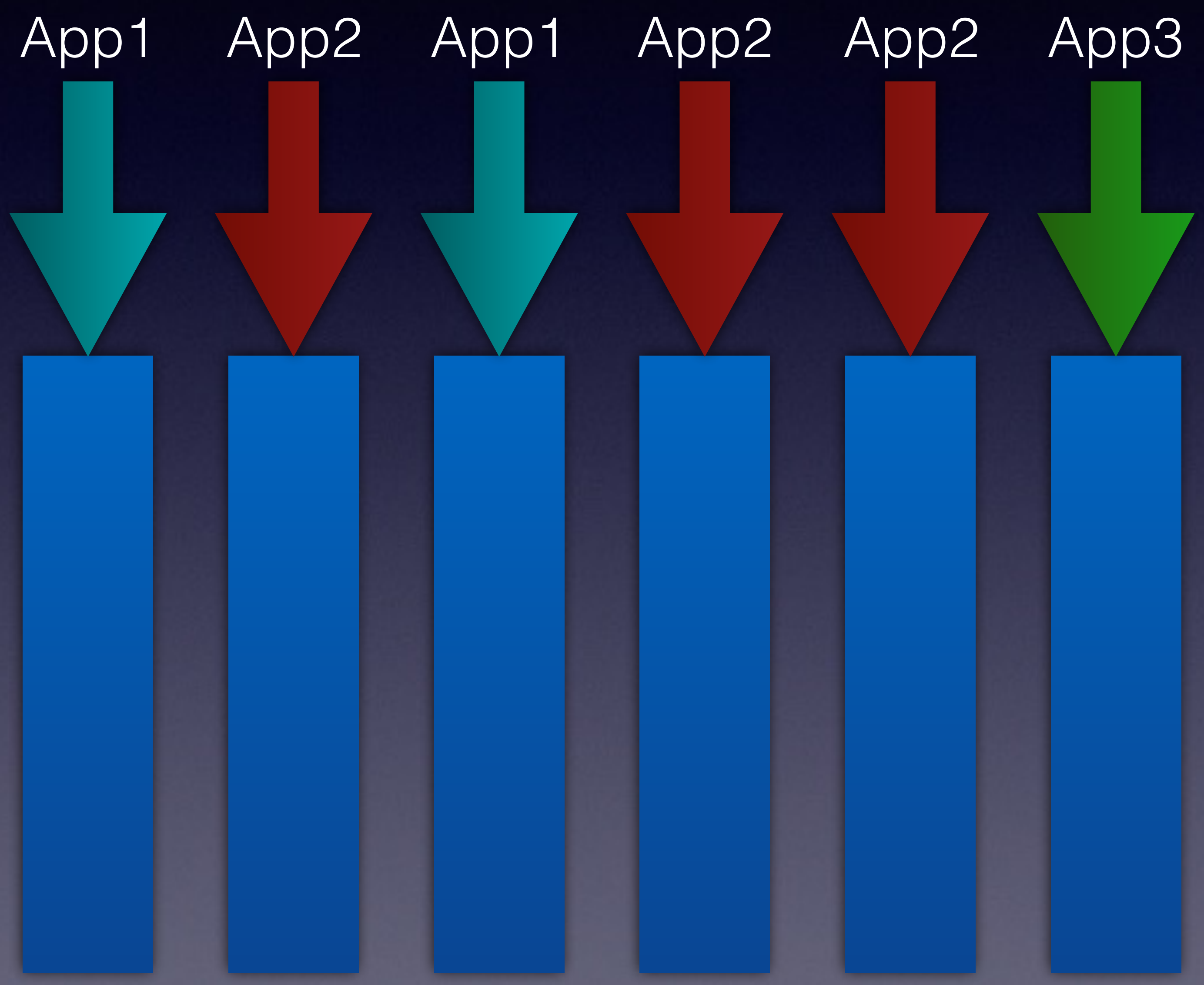
基于Kubernetes打造SAE容器云

丛磊
2016.1.24

目前SAE基于请求的架构



HTTP Request Pool

目前SAE基于请求的架构

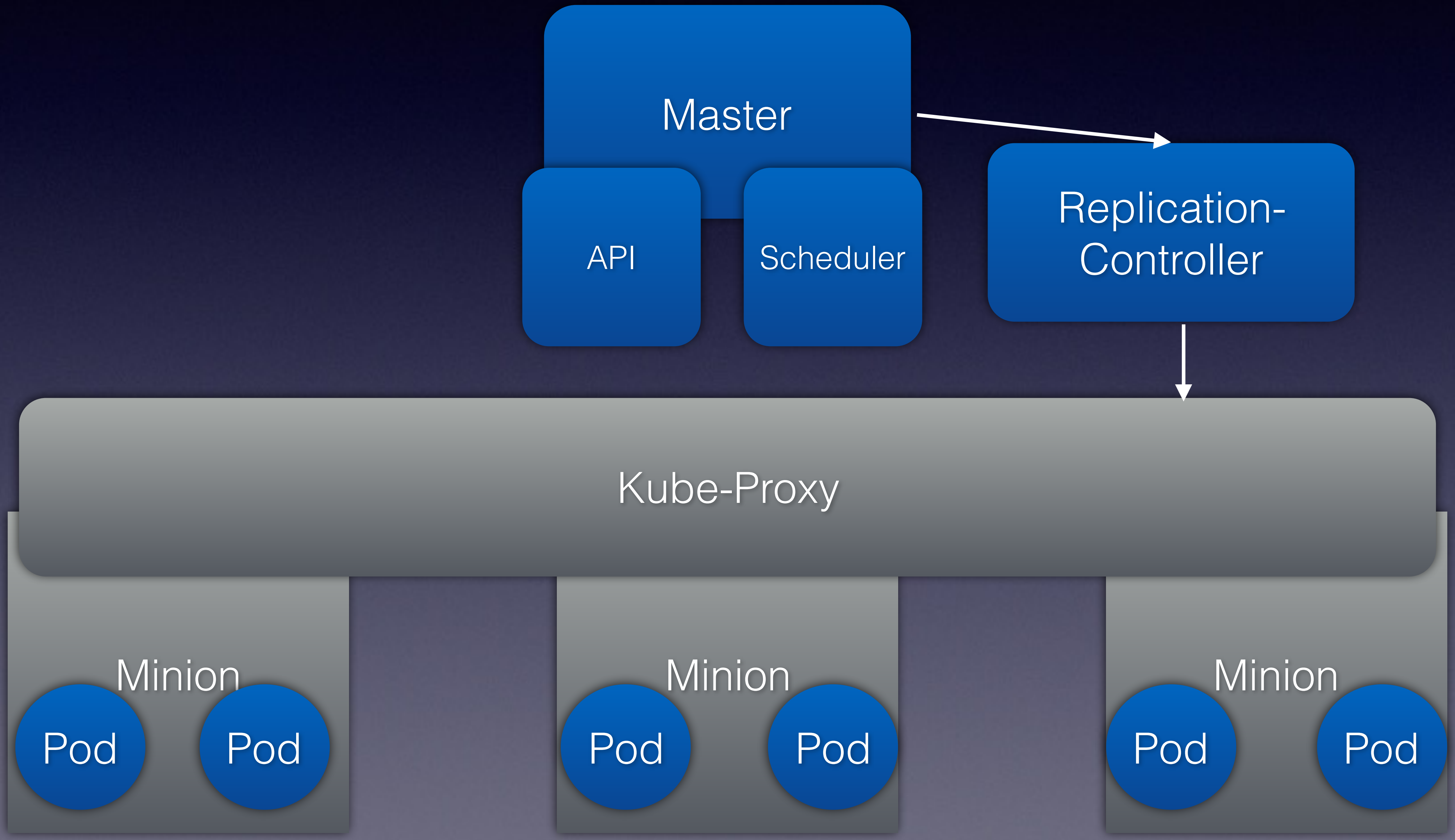
- 优点
  - 进程内隔离，消耗资源最小
  - 无感扩容&缩容，用户无成本
  - Health&Redispatch，升级切换无成本

- 缺点
  - 无法提供独立的namespace
  - 无法Build&Ship&Run

用户的需求

- 面对代码 vs 面向容器
- 定义一切
- run anywhere
- 无感扩容/缩容

为什么选择Kubernetes



为什么选择Kubernetes

- Pod
- Replication
- Go
- Easy for CentOS6

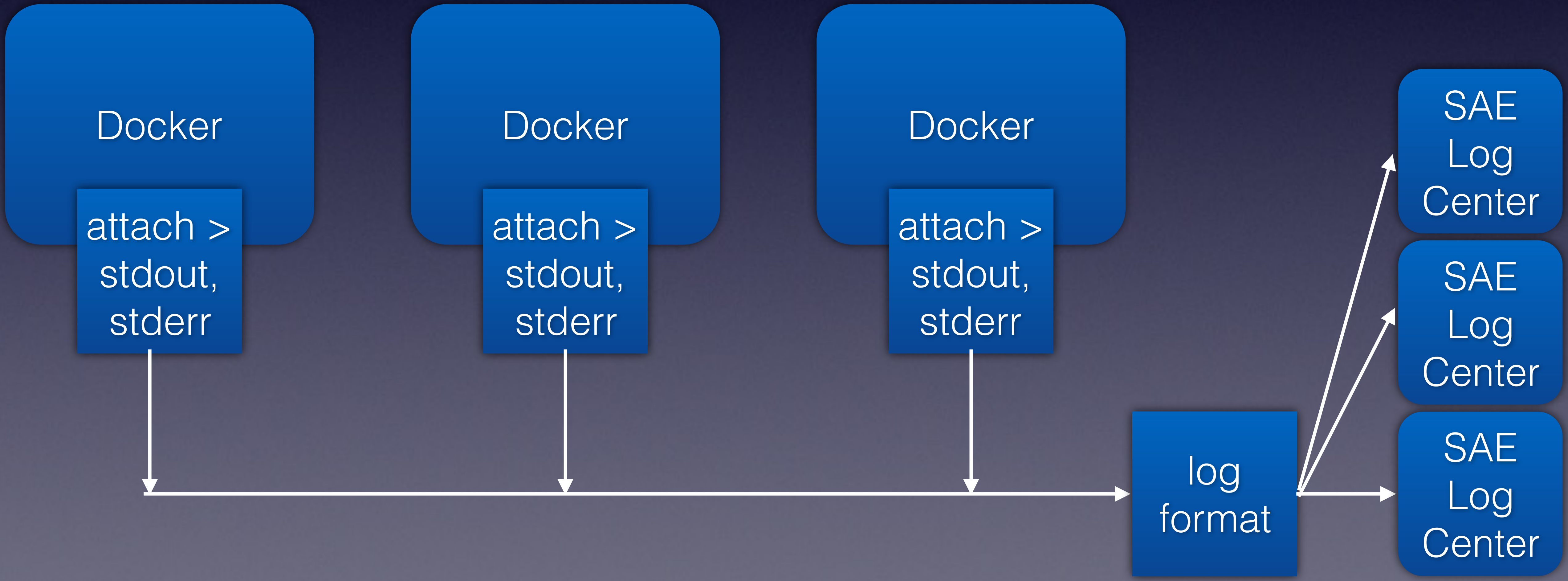
为什么要改进Kubernetes

- 不足之处:
  - 无感扩容
  - 监控

- 不适合SAE之处:
  - Kube-Proxy&VIP
  - Etcd

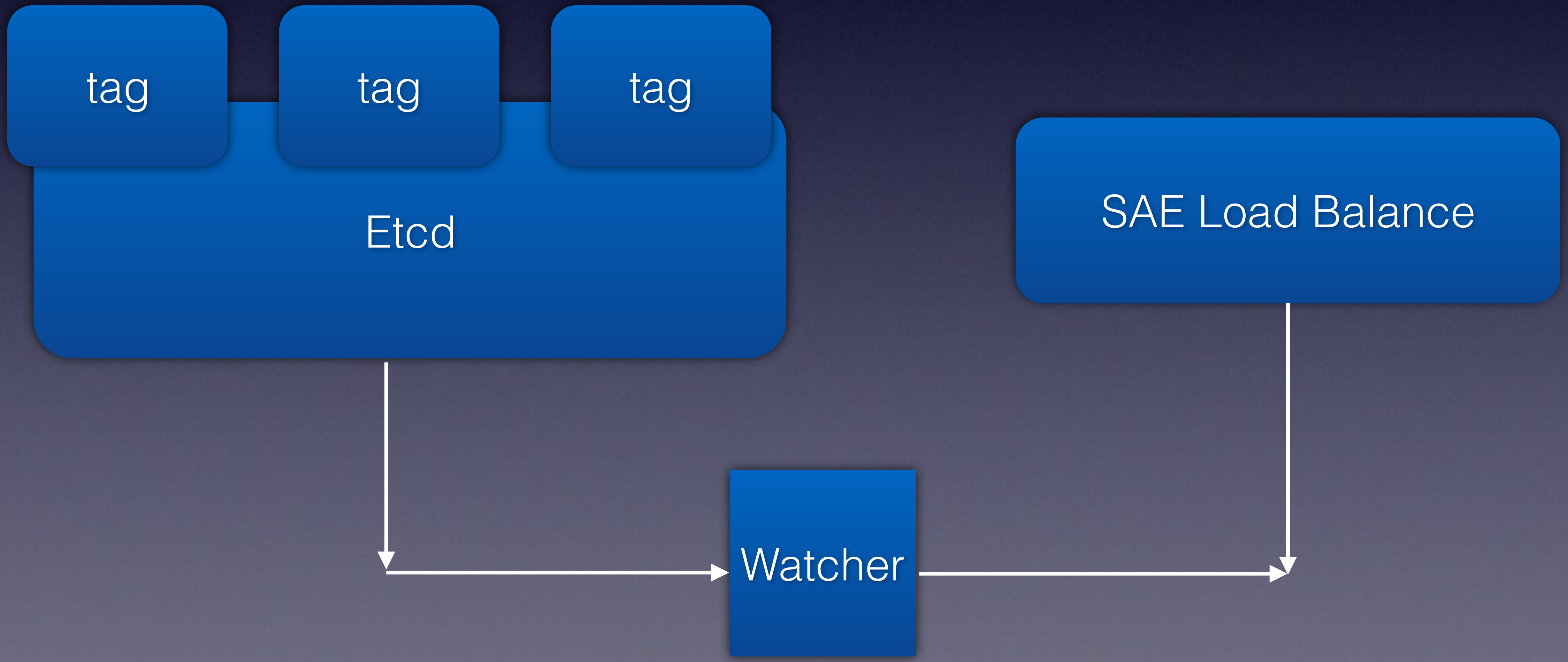
改进Kubernetes

- 日志系统

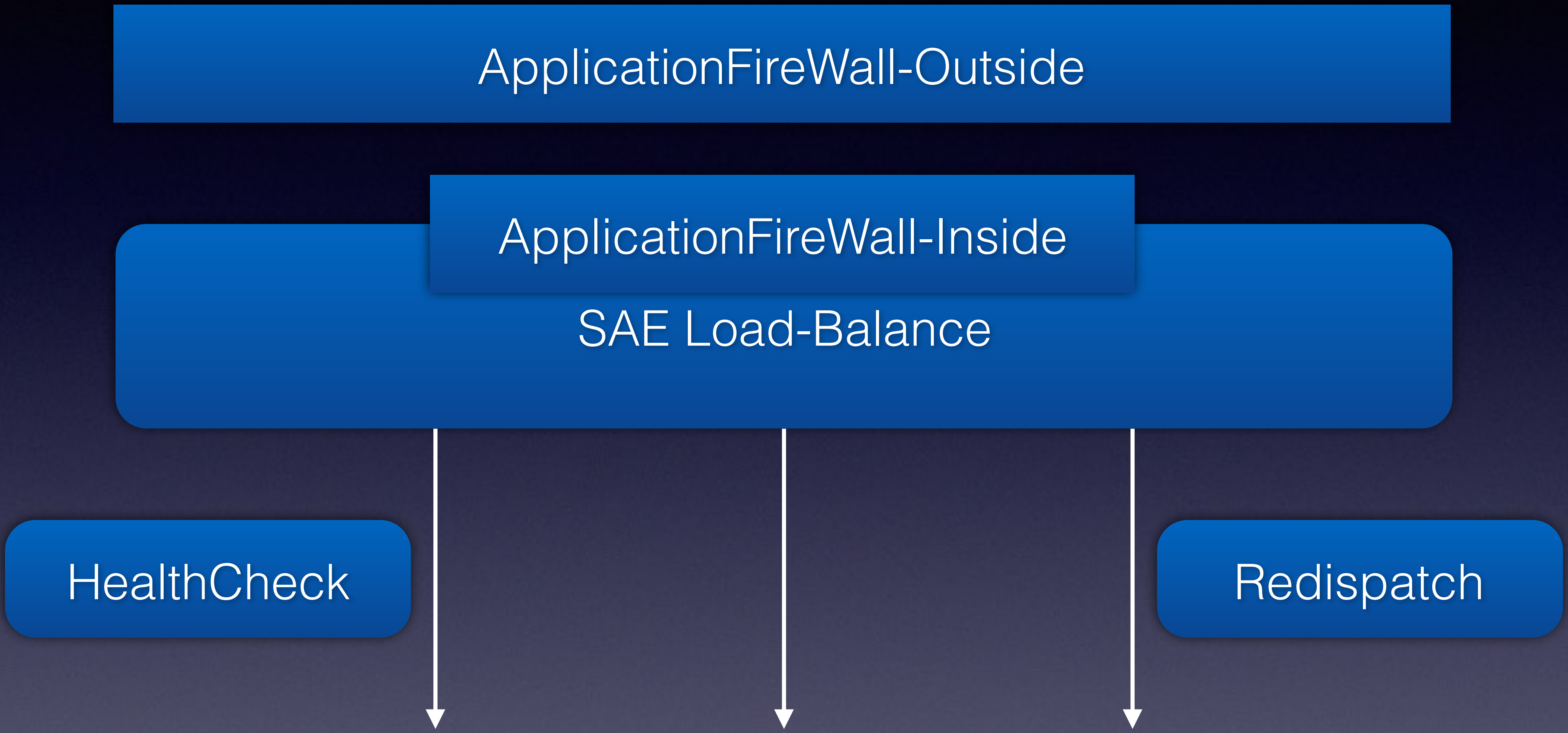


改进Kubernetes

- 接入SAE Load Balance



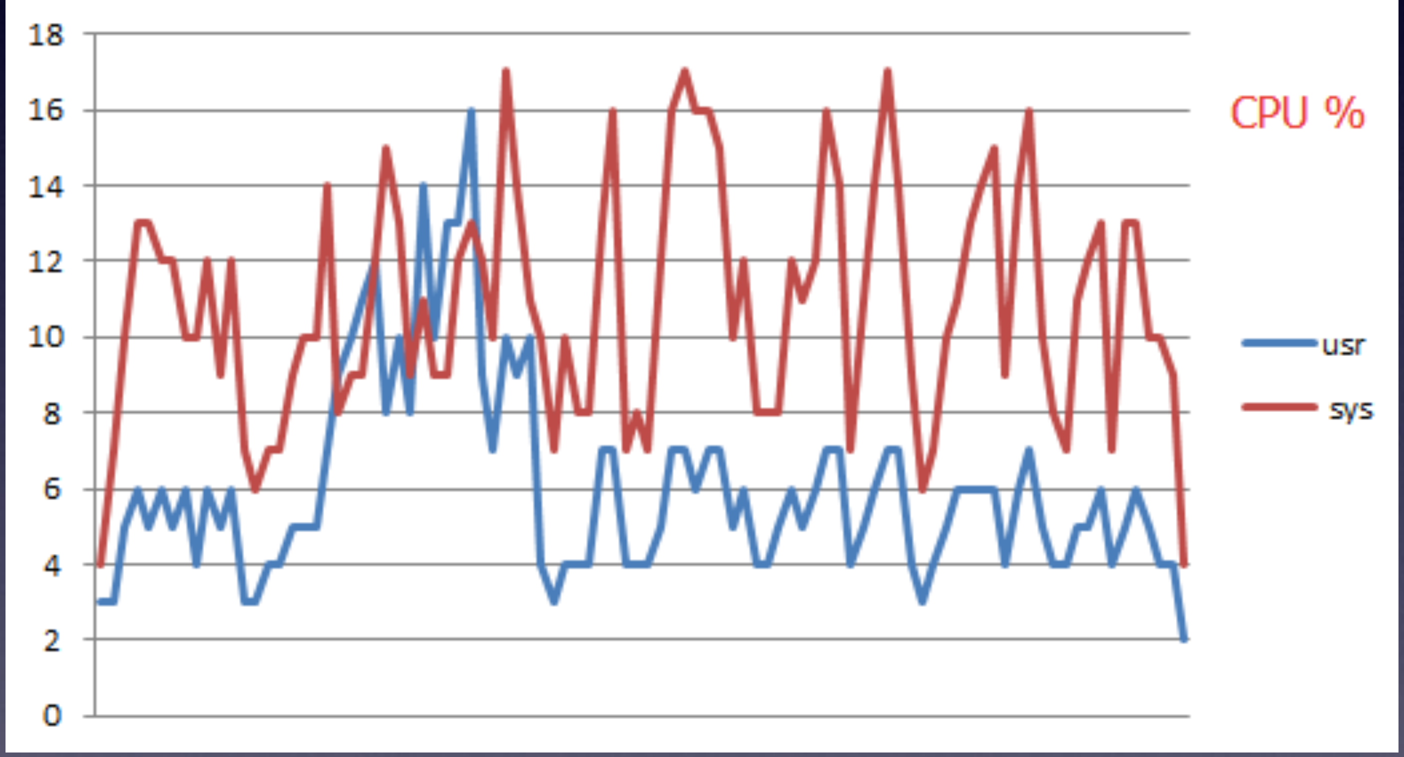
改进Kubernetes



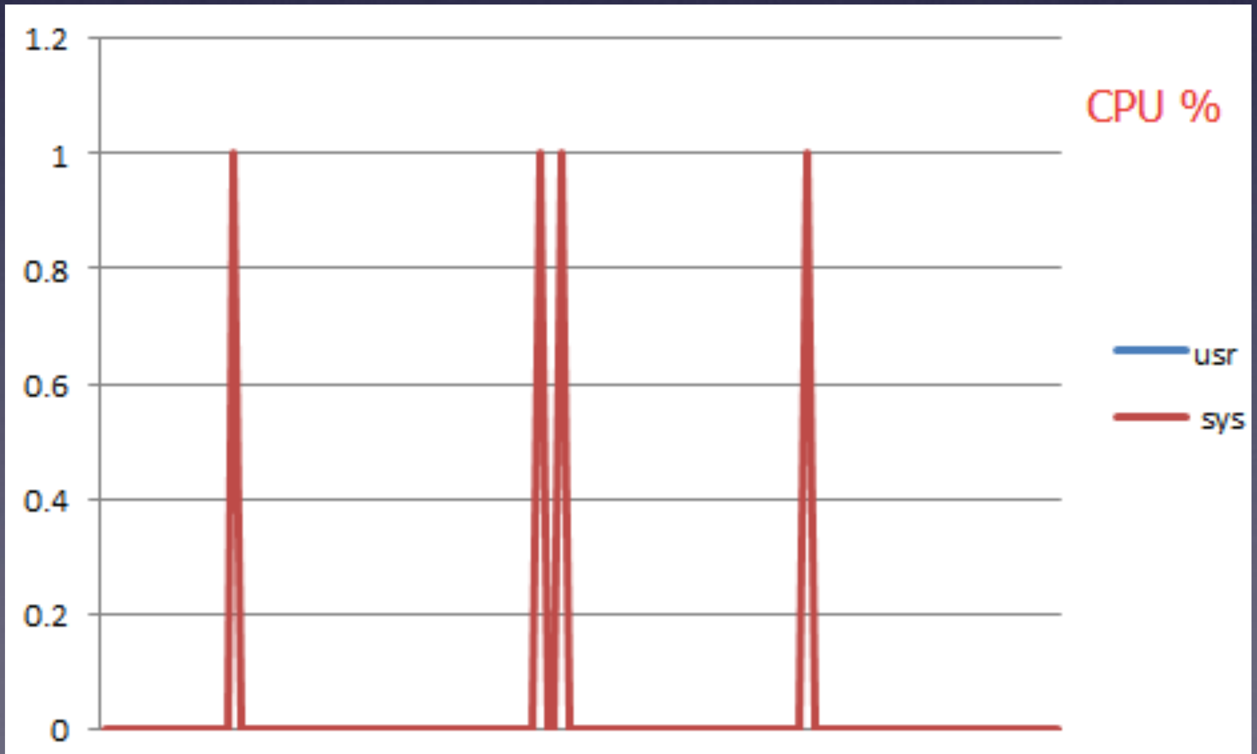
- Redispatch
- 403 vs 444
- Drop vs Reject
- Netlink-Queue

改进Kubernetes

- LoadBalance抗CC攻击压力对比



Nginx 444



Iptables + Netlink Queue

改进Kubernetes

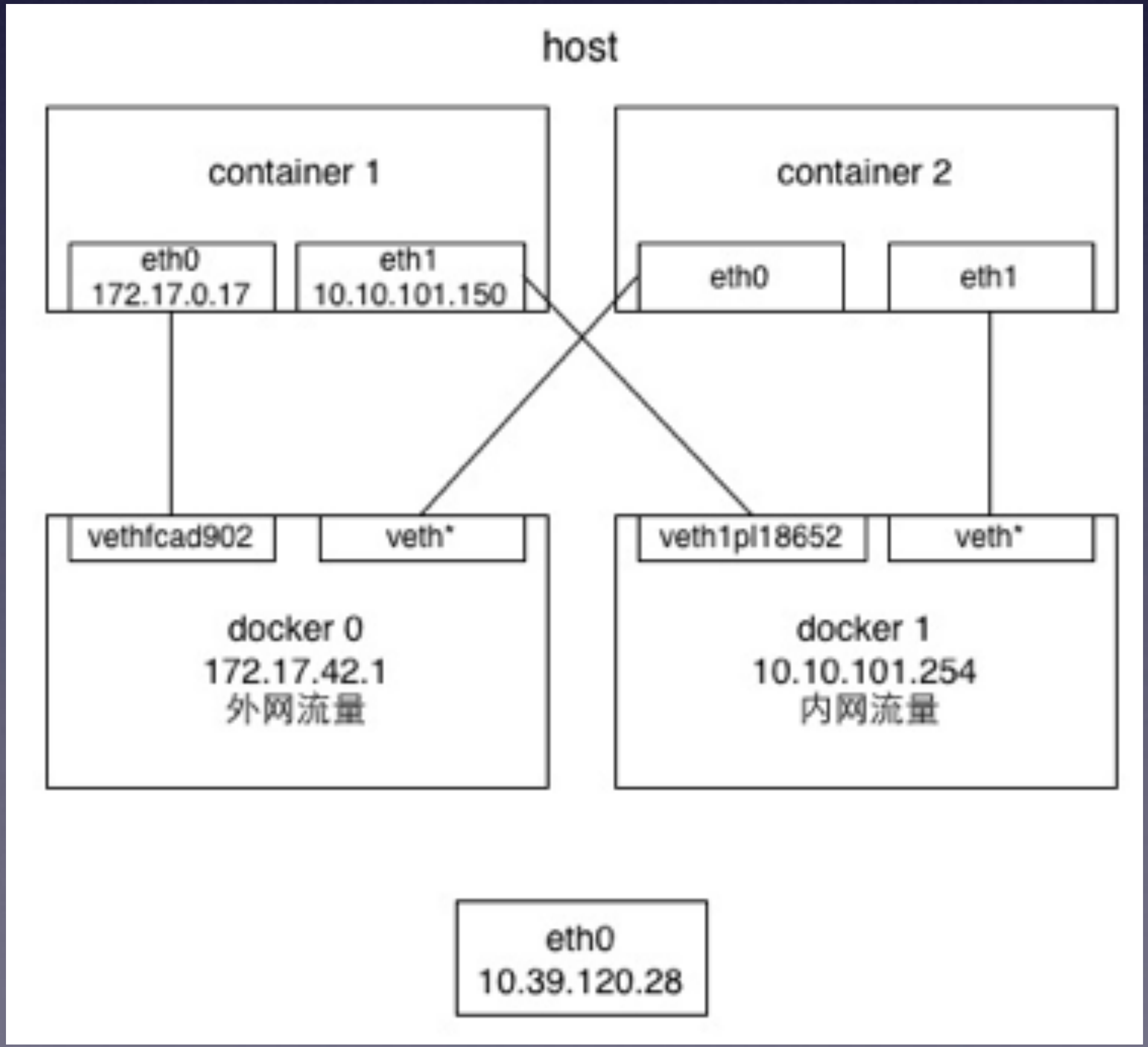
- PaaS SDN和IaaS SDN的区别

- 网络隔离
  - NAT
  - Bridge（更主流）

- 我们选择NAT
  - NAT提速

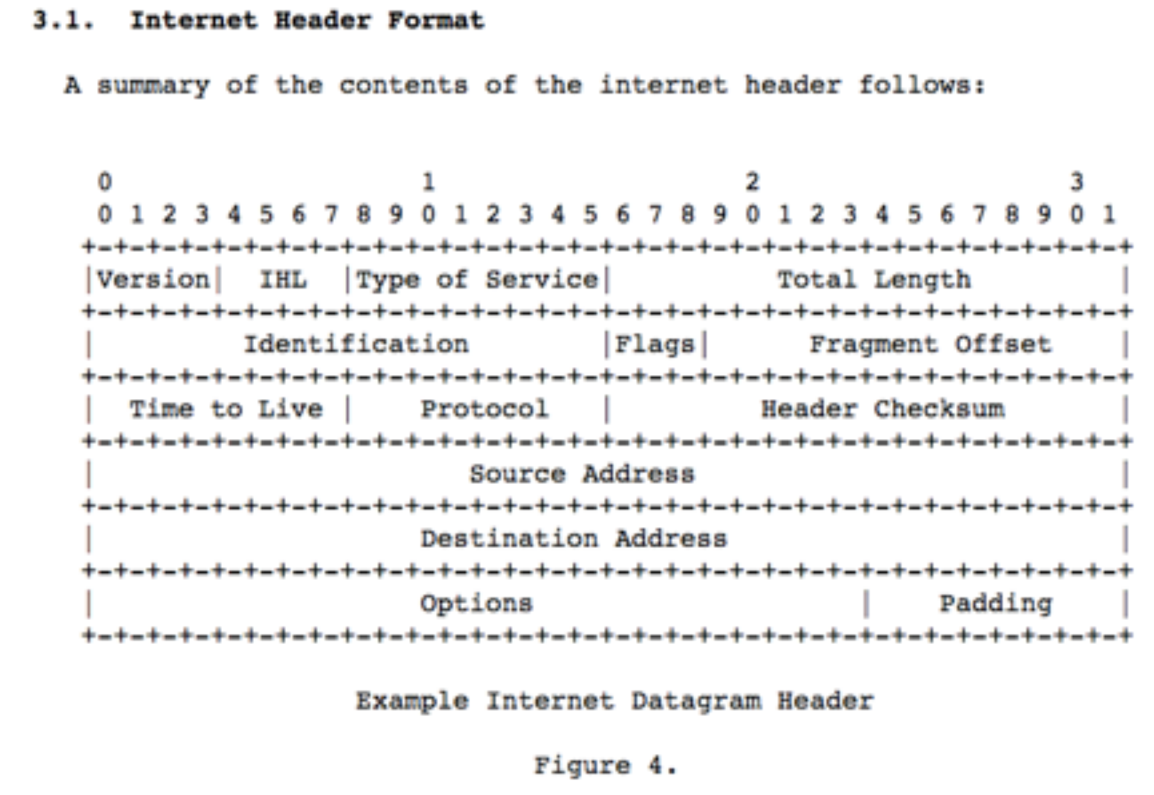
改进Kubernetes

- Simple Docker Network
- 内外网流量分开

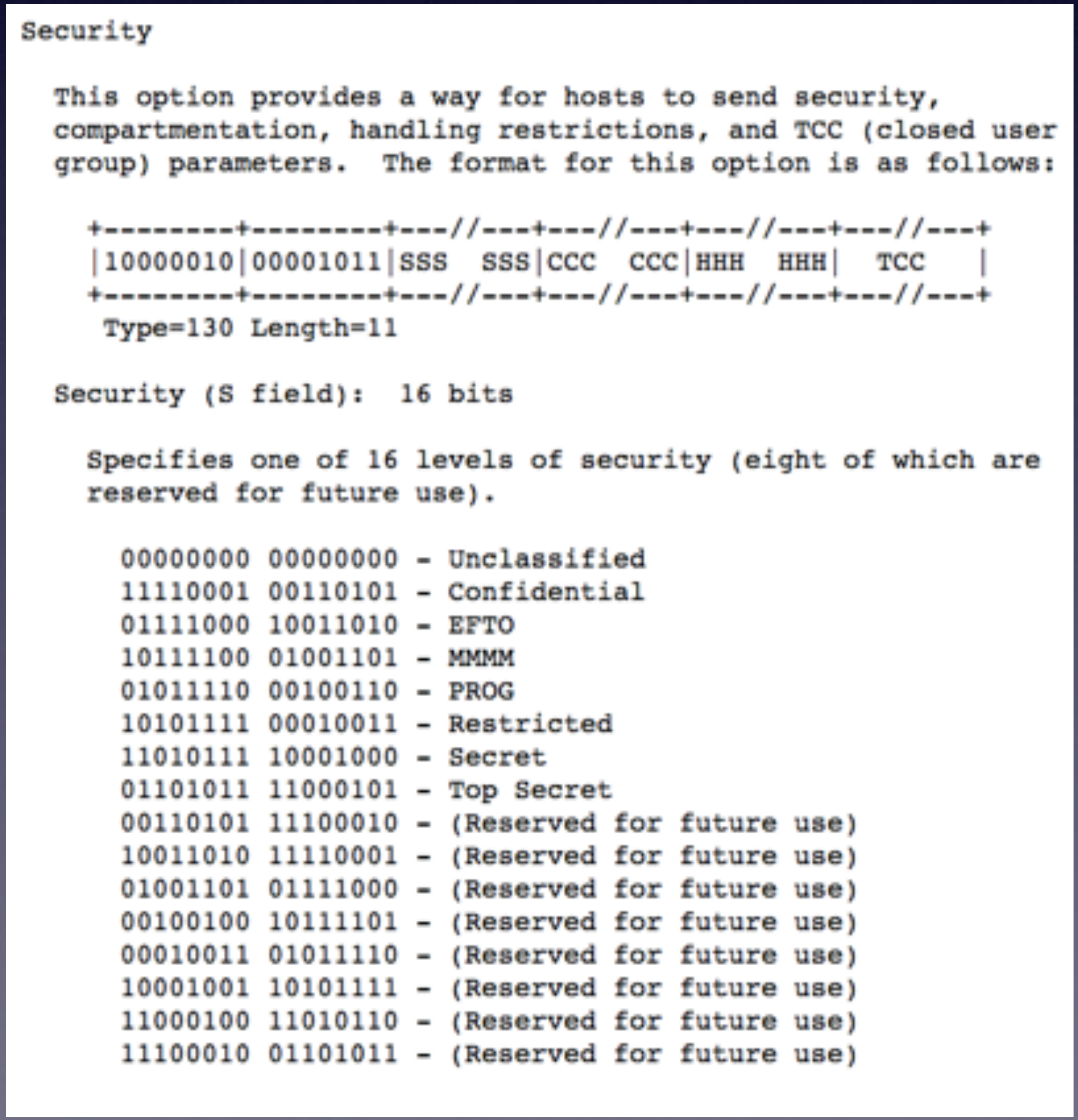


改进Kubernetes

- Simple Docker Network
- L3 tag



```
3.1.  Internet Header Format

  A summary of the contents of the internet header follows:


    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |Version|  IHL  |Type of Service|          Total Length         |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |         Identification        |Flags|      Fragment Offset    |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |  Time to Live |    Protocol   |         Header Checksum       |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                       Source Address                          |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                    Destination Address                        |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                    Options                    |    Padding    |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

                    Example Internet Datagram Header

                               Figure 4.
```

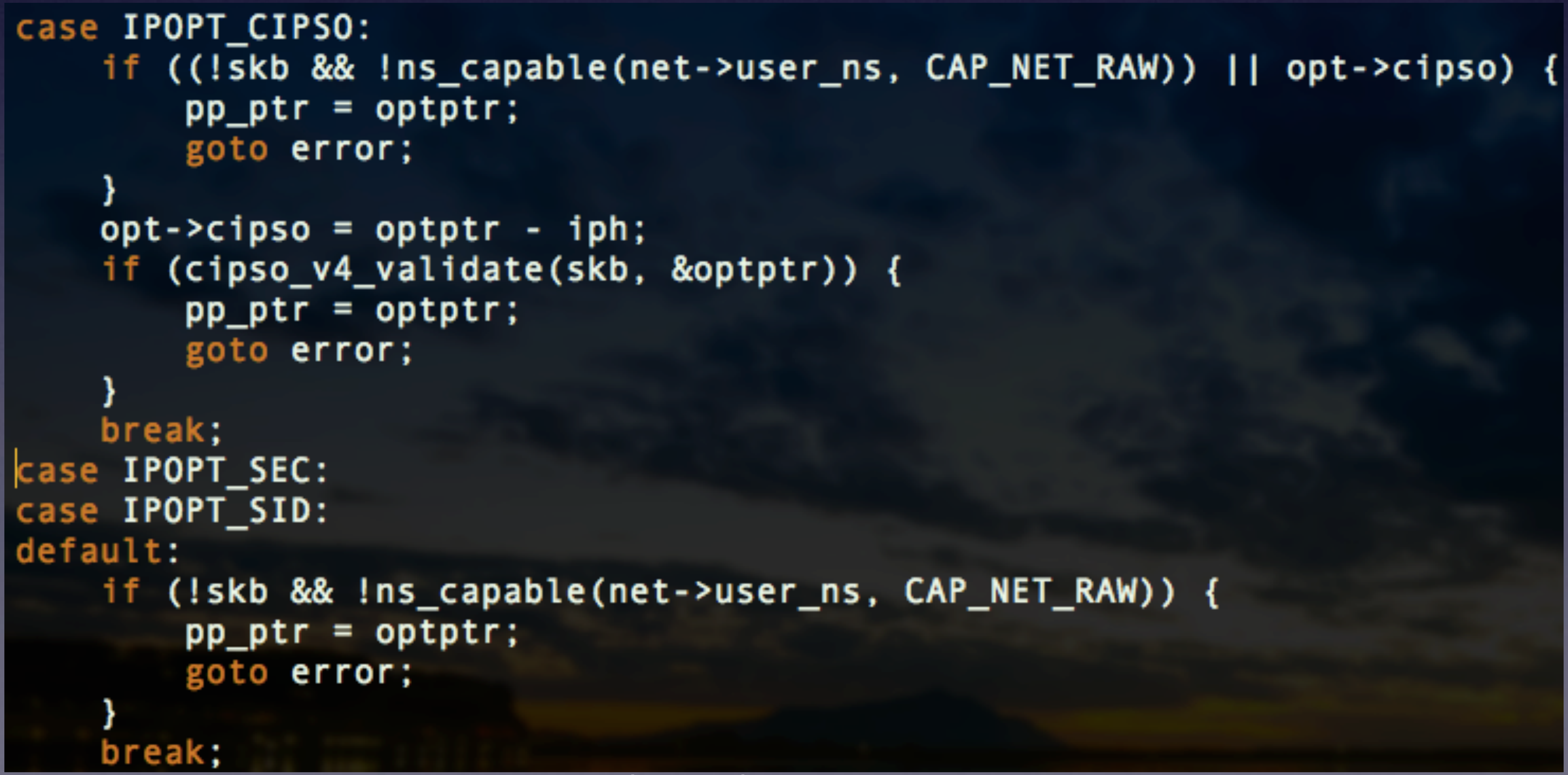


```
Security

  This option provides a way for hosts to send security,
  compartmentation, handling restrictions, and TCC (closed user
  group) parameters.  The format for this option is as follows:

    +--------+--------+---//---+---//---+---//---+---//---+
    |10000010|00001011|SSS  SSS|CCC  CCC|HHH  HHH|  TCC   |
    +--------+--------+---//---+---//---+---//---+---//---+
     Type=130 Length=11

  Security (S field):  16 bits

    Specifies one of 16 levels of security (eight of which are
    reserved for future use).

      00000000 00000000 - Unclassified
      11110001 00110101 - Confidential
      01111000 10011010 - EFTO
      10111100 01001101 - MMMM
      01011110 00100110 - PROG
      10101111 00010011 - Restricted
      11010111 10001000 - Secret
      01101011 11000101 - Top Secret
      00110101 11100010 - (Reserved for future use)
      10011010 11110001 - (Reserved for future use)
      01001101 01111000 - (Reserved for future use)
      00100100 10111101 - (Reserved for future use)
      00010011 01011110 - (Reserved for future use)
      10001001 10101111 - (Reserved for future use)
      11000100 11010110 - (Reserved for future use)
      11100010 01101011 - (Reserved for future use)
```

RFC791

改进Kubernetes

- Simple Docker Network
- 植入Tenant ID

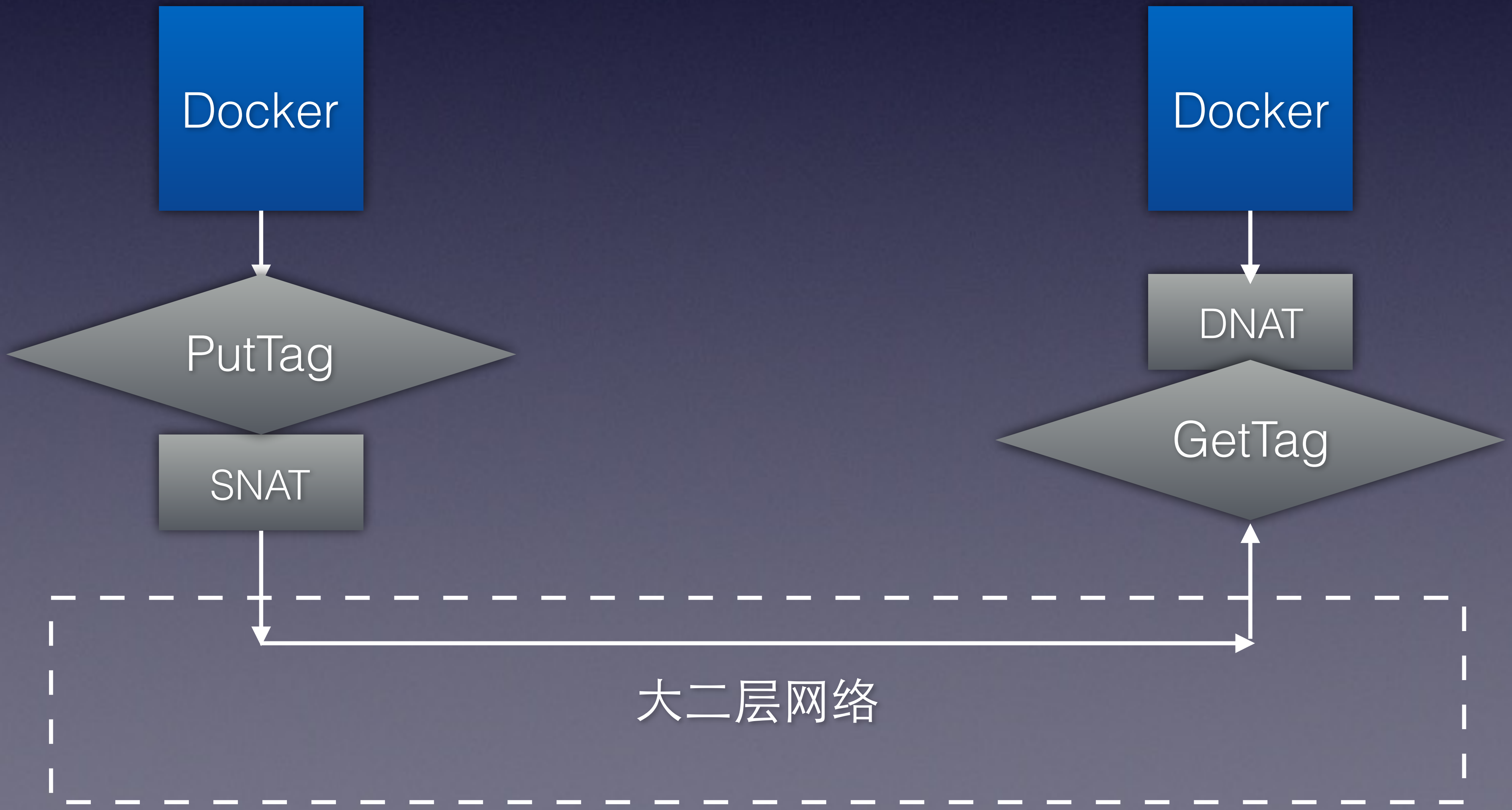


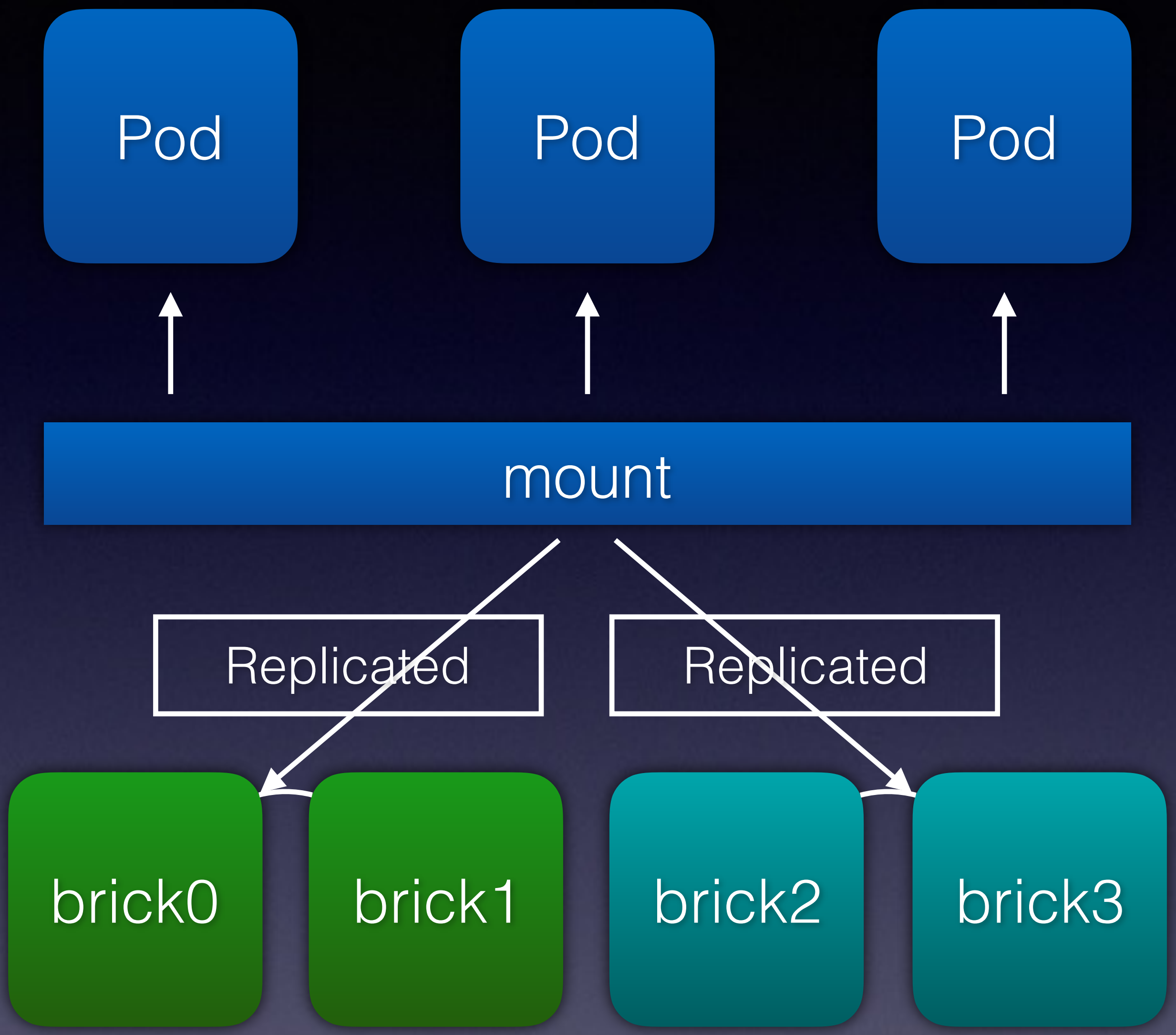
```
case IPOPT_CIPSO:
    if ((!skb && !ns_capable(net->user_ns, CAP_NET_RAW)) || opt->cipso) {
        pp_ptr = optptr;
        goto error;
    }
    opt->cipso = optptr - iph;
    if (cipso_v4_validate(skb, &optptr)) {
        pp_ptr = optptr;
        goto error;
    }
    break;
case IPOPT_SEC:
case IPOPT_SID:
default:
    if (!skb && !ns_capable(net->user_ns, CAP_NET_RAW)) {
        pp_ptr = optptr;
        goto error;
    }
    break;
```

net/ipv4/ip_options.c

改进Kubernetes

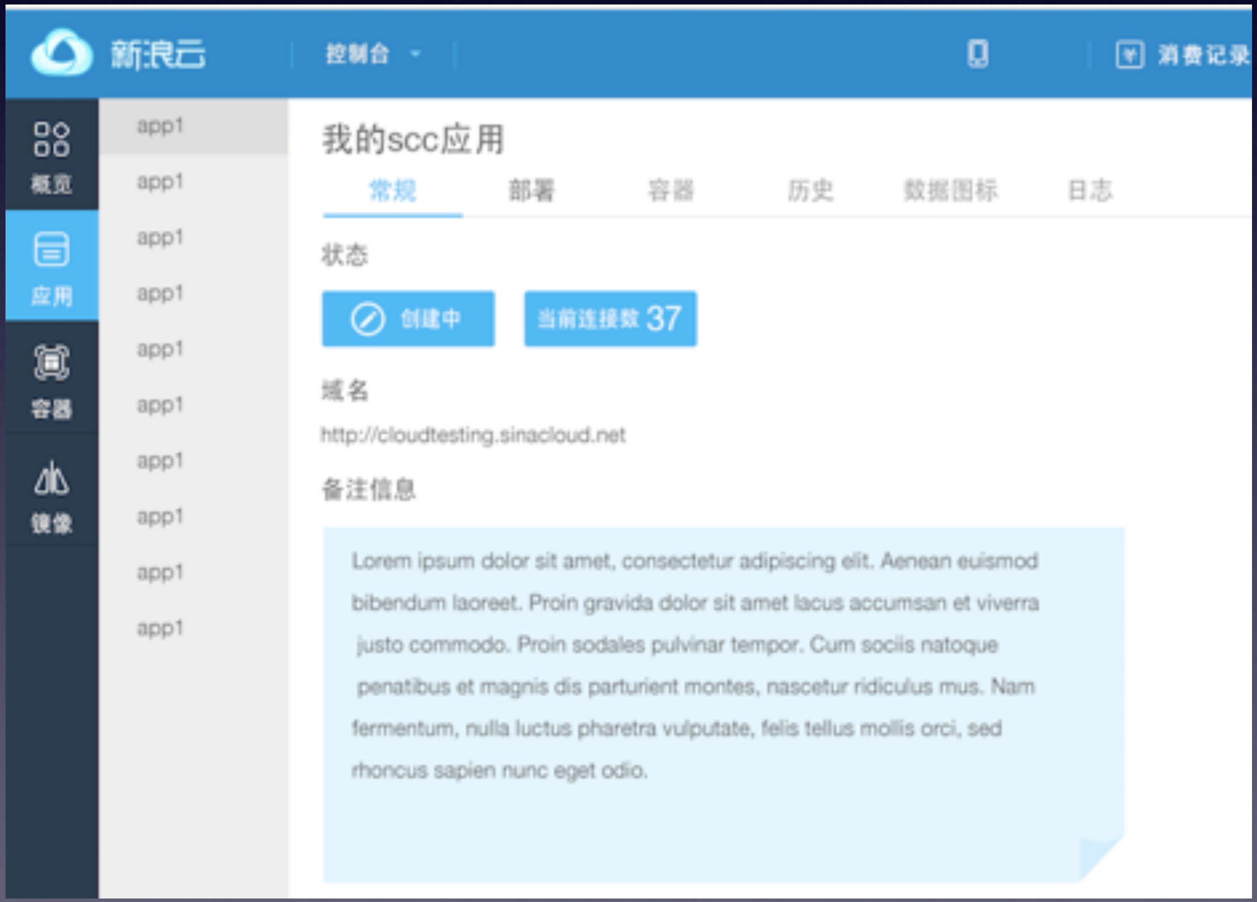
- Simple Docker Network




Pod
Pod
Pod
mount
Replicated
Replicated
brick0
brick1
brick2
brick3

SAE容器云

- 功能：
  - 镜像仓库
  - BuildPkg
  - 无感扩容
  - 共享存储
- 正式发布！



我的scc应用

状态

域名

http://cloudtesting.sinacloud.net

备注信息

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean euismod bibendum laoreet. Proin gravida dolor sit amet lacus accumsan et viverra justo commodo. Proin sodales pulvinar tempor. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Nam fermentum, nulla luctus pharetra vulputate, felis tellus mollis orci, sed rhoncus sapien nunc eget odio.

Q & A

丛磊
2015.11.6