HPA缩容问题

Subtitle

2022/09/27

Table of Contents

HPA缩容问题

问题描述见：conn_reuse_mode副作用

可能的原因

- terminationGracePeriodSeconds 对nginx没有效果：nginx being misbehaved
- ipvs gracefull termination

terminationGracePeriodSeconds

supervisor配置nginx和php-fpm gracefully shutdown

```
[program:nginx]
command = nginx -g 'daemon off;'
stdout_logfile=/dev/fd/1
stdout_logfile_maxbytes=0
stderr_logfile=/dev/fd/2
stderr_logfile_maxbytes=0
stopsignal=QUIT
stopwaitsecs=30

[program:php-fpm]
command = php-fpm5
stdout_logfile=/dev/fd/1
stdout_logfile_maxbytes=0
stderr_logfile=/dev/fd/2
stderr_logfile_maxbytes=0
stopsignal=QUIT
stopwaitsecs=35
```

效果不明显，缩容仍然会变慢

docker退出信号

以上过程有问题，有可能使用sh运行脚本启动supervisor的方式，pid 1的进程为/bin/sh，可能没有正确处理SIGTERM

- https://hynek.me/articles/docker-signals/
- https://stackoverflow.com/questions/33117068/use-of-supervisor-in-docker
- https://medium.com/@gchudnov/trapping-signals-in-docker-containers-7a57fdda7d86

目前CMD [“sh”, “/init.sh”] 方式，pstree看到的是

```
1fa221ba6ede:~# pstree
sh---supervisord-+-nginx---4*[nginx]
                  `-php-fpm5---30*[php-fpm5]
```

docker stop 命令执行慢的原因：不能正确处理SIGTERM，等10s后docker发送SIGKILL信号后才退出

So if you've ever wondered why your docker stop takes so long – this might be the reason: you didn't listen for SIGTERM and the signal bounced off your process because it's PID 1. No cleanup, slow shutdown.

docker处理signal的最佳实践

- Use the exec/JSON array form of ENTRYPOINT.
- Use exec in shell entrypoints.(shell的内建命令exec将并不启动新的shell，而是用要被执行命令替换当前的shell进程，并且将老进程的环境清理掉，而且exec命令后的其它命令将不再执行。)
- Don't pipe your application's output.
- Avoid being PID 1.(use a init system)
- Listen for SIGTERM or set STOPSIGNAL in your Dockerfile.

使用dumb-init管理多进程

```
RUN chmod +x /init.sh
ENTRYPOINT ["/usr/bin/dumb-init", "-v", "--rewrite", "15:3", "--"]
CMD ["/init.sh"]
```

init.sh脚本中可以做一些初始化操作，比如修改配置文件。需要运行多个程序时，前面的程序需要后台运行，最后一个程序使用exec调用，并在前台执行

```
php-fpm5 -D
exec nginx
```

preStop

Deployment滚动更新过程中流量负载均衡异常,会出现丢失请求的情况 原因：Pod Terminating过程中，有些机器的Iptable还未刷新，导致部分流量仍然请求到Terminating的Pod上，导致请求出错。详情参见：
https://github.com/kubernetes/kubernetes/issues/47597
https://github.com/kubernetes/kubernetes/issues/43576
https://github.com/kubernetes/kubernetes/issues/70747#issuecomment-440573161

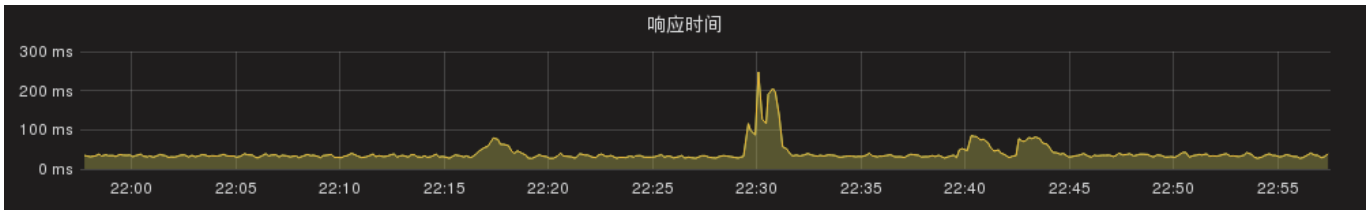
解决方案：利用Kubernetes的preStop特性为每个Pod设置一个退出时间，让每个Pod收到退出信号时时默认等待一段时间再退出。

第一次测试

```
        "lifecycle": {
          "preStop": {
            "exec": {
```

```
            "command": [
              "sleep",
          "30"
        ]
      }
    }
  },
```

结果



考虑到

> but kube-proxy need time to flush the rules, iptables mode should also have this problem.
> https://github.com/kubernetes/kubernetes/issues/70747#issuecomment-440573161

查看kube-proxy选项，ipvs-sync-period默认为30s因此考虑增加preStop时间到45s再次测试

```
--ipvs-sync-period duration                    The maximum interval of how often ipvs rules are refreshed (e.g. '5s', '1m', '2h22m'). Must be greater than 0. (default 30s)
```

第二次测试

改为45，仍未解决

```
        "lifecycle": {
          "preStop": {
            "exec": {
              "command": [
                "sleep",
          "45"
        ]
      }
    }
  },
```

第三次测试

- https://github.com/kubernetes/kubernetes/issues/71358

以上保持不变，设置net.ipv4.vs.expire*nodest*conn=1，再次测试

结果：**没有变化**

第四次测试

测试环境

- kube-proxy 1.11.4 with out gracefull termination
- service with only 1 pod
- lifecycle preStop sleep 45s;
- pod will return 302 if reachable

delete pod

```
# date; time kubectl -n dev delete pod kube-dash-proxy-7b5b6647ff-s2lz4 ;date
Sun Nov 25 14:55:20 CST 2018
pod "kube-dash-proxy-7b5b6647ff-s2lz4" deleted

real    0m48.476s
user    0m0.186s
sys 0m0.031s
Sun Nov 25 14:56:09 CST 2018
```

list rs

```
# while true;do dt=`date +%M:%S`;echo -n "$dt ";ipvsadm -ln |awk '{print $2}' |tr '\n' ' ' |sed 's/169.169/\n169.169/g' |grep 169.169.122.37;sleep 1;done
...
55:19 169.169.122.37:80 172.20.45.6:80
55:20 169.169.122.37:80 172.20.45.6:80
55:21 169.169.122.37:80
55:22 169.169.122.37:80
...
56:16 169.169.122.37:80
56:17 169.169.122.37:80
56:18 169.169.122.37:80 172.20.58.5:80
56:19 169.169.122.37:80 172.20.58.5:80
...
```

curl pod ip

```
# while true;do dt=`date +%M:%S`;echo -n "$dt ";curl -s http://172.20.45.6 -o /dev/null -w "%{http_code} %{time_connect} %{time_total}\n";sleep 1;done
...
55:18 302 0.001 0.001
55:19 302 0.001 0.002
55:20 302 0.001 0.001
55:21 302 0.001 0.034
...
56:04 302 0.001 0.002
56:05 302 0.001 0.001
56:06 000 0.001 0.033
56:07 000 0.000 0.000
...
```

curl svc ip

```
#  while true;do dt=`date +%M:%S`;echo -n "$dt ";curl -s http://169.169.122.37 -o /dev/null -w "%{http_code} %{time_connect} %{time_total}\n";sleep 1;done
...
55:19 302 0.001 0.002
```

```
55:20 302 0.001 0.001
55:21 000 0.000 1.008
55:23 000 0.000 1.008
...
56:18 302 1.011 1.012
56:20 302 0.001 0.002
...
```

第五次测试

- kube-proxy 1.11.4 with out gracefull termination
- **service with 3 pod**
- lifecycle preStop sleep 45s;
- pod will return 302 if reachable

delete pod

```
]# date; time kubectl -n dev delete pod kube-dash-proxy-7b5b6647ff-s2c6l ;date
Sun Nov 25 15:11:05 CST 2018
pod "kube-dash-proxy-7b5b6647ff-s2c6l" deleted

real    0m47.500s
user    0m0.186s
sys 0m0.025s
Sun Nov 25 15:11:53 CST 2018
```

list rs

```
# while true;do dt=`date +%M:%S`;echo -n "$dt ";ipvsadm -ln |awk '{print $2}' |tr '\n' ' ' |sed 's/169.169/\n169.169/g' |grep 169.169.122.37;sleep 1;done
...
11:04 169.169.122.37:80 172.20.36.9:80 172.20.45.7:80 172.20.58.5:80
11:05 169.169.122.37:80 172.20.36.9:80 172.20.45.7:80 172.20.58.5:80
11:06 169.169.122.37:80 172.20.36.9:80 172.20.58.5:80
11:07 169.169.122.37:80 172.20.36.9:80 172.20.58.5:80
...
12:00 169.169.122.37:80 172.20.36.9:80 172.20.48.7:80 172.20.58.5:80
12:01 169.169.122.37:80 172.20.36.9:80 172.20.48.7:80 172.20.58.5:80
```

curl pod ip

```
# while true;do dt=`date +%M:%S`;echo -n "$dt ";curl -s http://172.20.45.7 -o /dev/null -w "%{http_code} %{time_connect} %{time_total}\n";sleep 1;done
...
11:04 302 0.001 0.001
11:05 302 0.001 0.001
...
11:49 302 0.001 0.001
11:50 302 0.001 0.002
11:51 302 0.001 0.002
11:52 000 0.000 0.000
11:53 000 0.000 0.000
...
```

curl svc ip

```
#  while true;do dt=`date +%M:%S`;echo -n "$dt ";curl -s http://169.169.122.37 -o /dev/null -w "%{http_code} %{time_connect} %{time_total}\n";sleep 1;done
10:42 302 0.001 0.002
10:43 302 0.001 0.003
...
11:04 302 0.001 0.001
11:05 302 0.001 0.001
11:06 302 0.001 0.002
...
11:52 302 0.001 0.002
11:53 302 0.001 0.001
11:54 302 0.001 0.002
...
```

curl svc ip no sleep

> 因为要压测，先扩容为2核，防止cpu不够用对结果造成的干扰

```
ab -n1000000 -t65 -c5 http://169.169.122.37/
# keep alive
ab -c5 -t65 -n1000000 -r -k http://169.169.122.37/
```

analyze

```
不删pod,完成256k请求，删pod，完成63k请求
keepalive，不删pod完成700k删pod完成832k ？？？居然还多了
```

Printed on: 2022/09/27 02:13

Convert to img Failed!