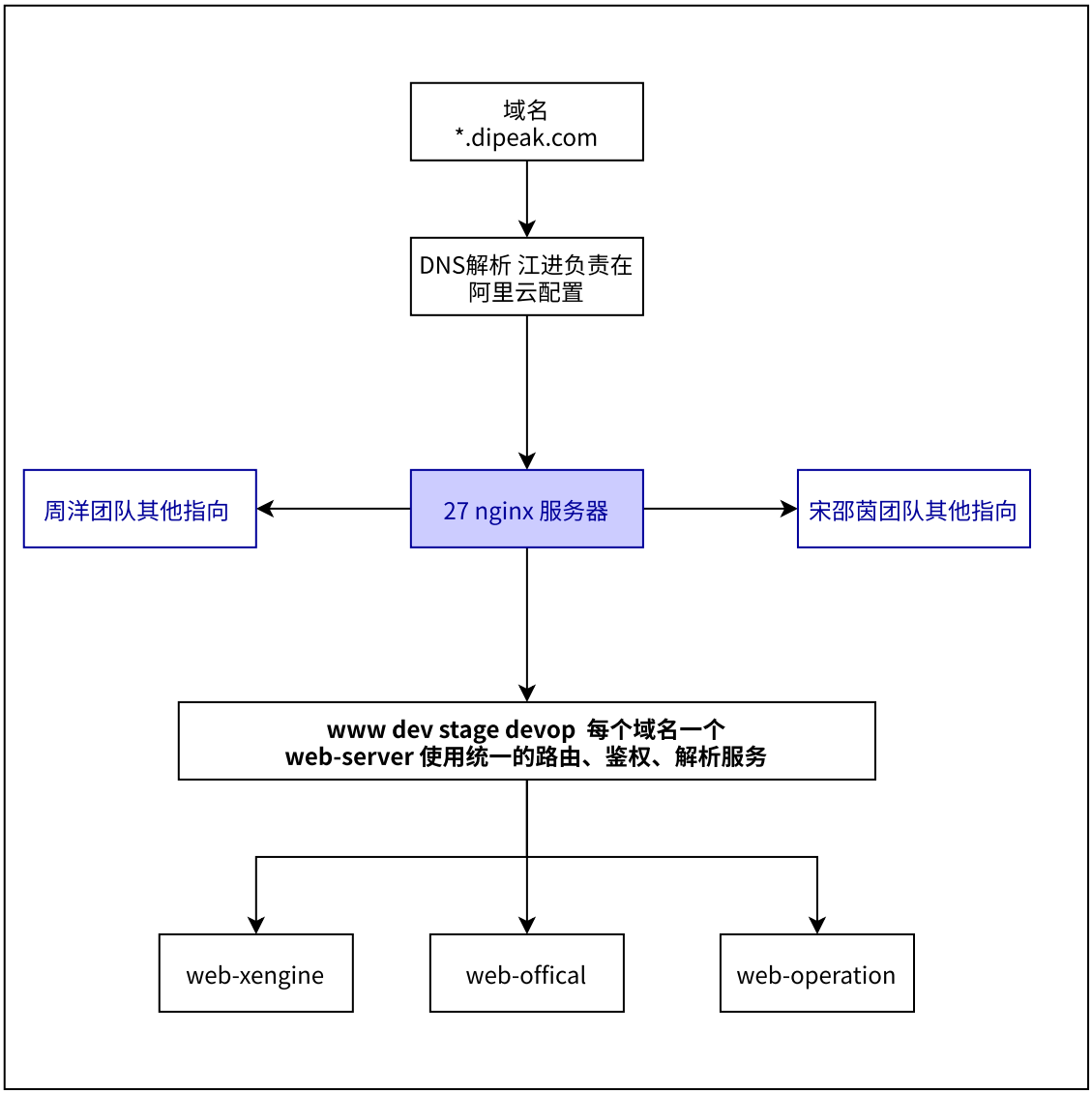
从 0 开始了解前端部署链路

架构图



Nginx 篇

```
1 登录 27服务器
2 ssh lijiacheng@192.168.110.27
```

```
1 查找nginx 所在容器
```

```
docker ps | grep nginx
```


```
进入容器
docker exec -it fa3dcb1751f4 bash
cd /etc/nginx/          #进入 nginx 配置文件 这里面有两个文件(夹)比较重要 conf.d
nginx.conf
                        #conf.d 存放 proxy 相关的配置
                        #nginx.conf 存放负载均衡配置
```

/etc/nginx/nginx.conf 存放负载均衡配置 文件内容如下

```
root@fa3dcb1751f4:/etc/nginx# cat /etc/nginx/nginx.conf

user  nginx;
worker_processes  auto;

error_log  /var/log/nginx/error.log notice;
pid        /var/run/nginx.pid;


events {
    worker_connections  1024;
}


http {
    include       /etc/nginx/mime.types;
    default_type  application/octet-stream;

    log_format  main  '$remote_addr - $remote_user [$time_local] "$request" '
                      '$status $body_bytes_sent "$http_referer" '
                      '"$http_user_agent" "$http_x_forwarded_for"';

    access_log  /var/log/nginx/access.log  main;

    sendfile        on;
    #tcp_nopush     on;

    keepalive_timeout  65;

    #gzip  on;

    client_max_body_size 8g;

    include /etc/nginx/conf.d/*.conf;

    upstream devops_node_servers {
        server 192.168.110.27:32316;
    }
    upstream dev_node_servers {
        #server 192.168.110.27:18080;
        server 192.168.110.27:32321;
    }
    upstream www_node_servers {
        #server 192.168.110.26:8089;
        server 192.168.110.27:8089;
        server 192.168.110.27:32318;
        ip_hash;
    }
}
root@fa3dcb1751f4:/etc/nginx#
```

/etc/nginx/conf.d 文件夹内容如下 存放 proxy 逻辑 和 https 证书文件位置配置

```
root@fa3dcb1751f4:/etc/nginx/conf.d# ls
askbi.conf   default.conf           devops.conf               fanwei.dipeak.com.conf  htpasswd2  htpasswd4  smartbot.conf
askdoc.conf  default.conf.back.0908  devops.conf.back.20230915  htpasswd                htpasswd3  htpasswd5  smartbot.conf.back.0720
root@fa3dcb1751f4:/etc/nginx/conf.d#
```

查看 devops.conf 配置 如下

```
server {
        listen       443 ssl http2;
        server_name  devops.dipeak.com;
        ssl_certificate /etc/letsencrypt/live/dipeak.com/fullchain.pem;
        ssl_certificate_key /etc/letsencrypt/live/dipeak.com/privkey.pem;
```

```
6
7           location /.well-known/ {
8               root /usr/share/nginx/html;
9               tcp_nodelay     on;
10              proxy_set_header Host            $host;
11              proxy_set_header X-Real-IP       $remote_addr;
12              proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
13              proxy_read_timeout 1500;  # 秒
14          }
15          location / {
16              # 检查请求的URI是否是根路径 "/"
17              if ($request_uri = "/") {
18                  # 如果是根路径，则执行重定向
19                  return 302
    https://devops.dipeak.com/xengine/devops/operation/home;
20              }
21              proxy_pass http://devops_node_servers;
22              tcp_nodelay     on;
23              proxy_set_header Host            $host;
24              proxy_set_header X-Real-IP       $remote_addr;
25              proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
26              proxy_read_timeout 1500;  # 秒
27          }
28  }
29
30  server {
31          listen       443 ssl http2;
32          server_name  *.devops.dipeak.com;
33          ssl_certificate /etc/letsencrypt/live/dipeak.com/fullchain.pem;
34          ssl_certificate_key /etc/letsencrypt/live/dipeak.com/privkey.pem;
35
36
37          if ($http_host ~* "^(.*?)\.devops\.dipeak\.com$") {     #正则表达式
38                  set $domain $1;                         #设置变量
39          }
40
41          location /.well-known/ {
42              root /usr/share/nginx/html;
43              tcp_nodelay     on;
44              proxy_set_header Host            $host;
45              proxy_set_header X-Real-IP       $remote_addr;
46              proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
47              proxy_read_timeout 1500;  # 秒
48              #以上三行，目的是将代理服务器收到的用户的信息传到真实服务器上
49
50          }
51
```

```
52
53          location / {
54              root /usr/share/nginx/html;
55
56              if ($domain ~* "grafana") {
57                  proxy_pass http://192.168.110.27:30030;
58              }
59              if ($domain ~* "elk") {
60                  proxy_pass http://192.168.110.27:30560;
61              }
62              if ($domain ~* "jenkins") {
63                  proxy_pass http://192.168.110.21:8090;
64              }
65
66              tcp_nodelay     on;
67              proxy_set_header Host             $host;
68              proxy_set_header X-Real-IP        $remote_addr;
69              proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
70              proxy_read_timeout 1500;  # 秒
71              #以上三行，目的是将代理服务器收到的用户的信息传到真实服务器上
72          }
73  }
```

进行 https 签名

- certbot certonly --webroot -w /usr/share/nginx/html -d dipeak.com -d dev.dipeak.com -d test.dipeak.com -d sentry.dipeak.com -d pufa.dipeak.com -d staging.dipeak.com -d smartbot.dipeak.com -d smartbot-dev.dipeak.com -d smartbot-test.dipeak.com -d www.dipeak.com -d askbi.dipeak.com -d askdoc.dipeak.com -d devops.dipeak.com -d askbi2.dipeak.com -d askbi3.dipeak.com -d askbi4.dipeak.com -d askbi5.dipeak.com -d askdoc2.dipeak.com -d askdoc3.dipeak.com -d grafana.devops.dipeak.com -d elk.devops.dipeak.com -d jenkins.devops.dipeak.com -d fanwei.dipeak.com 这里加上你的域名

- 然后修改 nginx
- nginx -t 测试 nginx 文件配置是否正确
- 最后 nginx -s reload 否则不会重载证书，会提示证书不信任导致没有效果
- nginx gitlab ci 里面执行 nginx -t -> 打单独的镜像 部署镜像 treafk k8s

---Nginx 篇完结

k8s 篇

创建k8s 容器

k8s 创建新的容器环境

获取 pod

登录27机器

kubectl get pods -n frontend-nodejs

```
lijiacheng@T440-27 ~ % kubectl get pods -n frontend-nodejs
NAME                                   READY   STATUS    RESTARTS   AGE
dev-application-6c6fd49b59-f8pw2       1/1     Running   0          22h
devops-application-79d4b74b54-ppwd5    1/1     Running   0          64m
mvp-application-7f594674c5-qckjd       1/1     Running   0          4d3h
staging-application-6b97685fb9-xk6rg   1/1     Running   0          22h
www-application-6748876695-574cd       1/1     Running   0          3d
```

需求

- [ ] 泛域名 => 代理机制
  - [ ] x.member.dipeak.com => 按照普通签名的形式签泛域名
  - [ ] x.dipeak.com => 自定义关系 askmeta xengine distore ditest
- [ ] 添加域名 每人一个域名 x.member.dipeak.com =>
- [ ] 开 8 个域名 普通形式搞

- [ ] 每个项目再开一个域名 askmeta xengine distore ditest
- [ ] dev 保持最新
- [ ] pod 每个同学自己搞
- [ ] 新搞一个ns

- [ ] Nginx 代码放仓库

- [ ] 启动新域名　　经过旧nginx　到新的　　nginx