

nginx代理vue

一、nginx代理vue的意义

vue打包后,因为开发环境设置代理 `proxy` 的原因,使得访问打包页面 `dist/index.html` 请求数据出现跨域。

nginx代理vue的意义在于把开发环境中的代理 `proxy: {'/api': ...` 的/api,换成生产环境中的地址。

二、下载nginx

下载并解压,推荐使用稳定版本 (nginx/Windows-1.18.0)

nginx: download				
Mainline version				
CHANGES	nginx-1.19.0	pgp	nginx/Windows-1.19.0	pgp
Stable version				
CHANGES-1.18	nginx-1.18.0	pgp	nginx/Windows-1.18.0	pgp
Legacy versions				
CHANGES-1.16	nginx-1.16.1	pgp	nginx/Windows-1.16.1	pgp
CHANGES-1.14	nginx-1.14.2	pgp	nginx/Windows-1.14.2	pgp
CHANGES-1.12	nginx-1.12.2	pgp	nginx/Windows-1.12.2	pgp
CHANGES-1.10	nginx-1.10.3	pgp	nginx/Windows-1.10.3	pgp

三、vue打包

```
1 npm run build // 打包文件
```

(默认)生成文件夹 `dist`

四、放入nginx中

把生成的 `dist` 下所有文件 放入压缩后nginx中 `html` 文件夹下

名称	修改日期	类型	大小
conf	2020/5/27 10:37	文件夹	
contrib	2020/5/27 10:37	文件夹	
docs	2020/5/27 10:37	文件夹	
html	2020/5/27 11:03	文件夹	
logs	2020/5/27 10:37	文件夹	
temp	2020/5/27 10:37	文件夹	
nginx.exe	2020/4/21 21:26	应用程序	3,630 KB

五、配置nginx

进入nginx的 `conf`

文件夹中, 打开 `nginx.conf` 文件进行代理配置

› nginx-1.18.0 <code>conf</code>			
名称	修改日期	类型	大小
fastcgi.conf	2020/4/21 22:12	CONF 文件	2 KB
fastcgi_params	2020/4/21 22:12	文件	2 KB
koi-utf	2020/4/21 22:12	文件	3 KB
koi-win	2020/4/21 22:12	文件	3 KB
mime.types	2020/4/21 22:12	TYPES 文件	6 KB
nginx.conf	2020/5/27 10:39	CONF 文件	4 KB
scgi_params	2020/4/21 22:12	文件	1 KB
uwsgi_params	2020/4/21 22:12	文件	1 KB
win-utf	2020/4/21 22:12	文件	4 KB

`nginx` 配置文件

```
1
2 #user nobody;
3 worker_processes 1;
4
5 #error_log logs/error.log;
6 #error_log logs/error.log notice;
7 #error_log logs/error.log info;
8
9 #pid logs/nginx.pid;
10
11
12 events {
13     worker_connections 1024;
14 }
15
16
17 http {
18     include mime.types;
19     default_type application/octet-stream;
20
21     #log_format main '$remote_addr - $remote_user [$time_local] "$request" '
22     # '$status $body_bytes_sent "$http_referer" '
```

```
23 # "$http_user_agent" "$http_x_forwarded_for";
24
25 #access_log logs/access.log main;
26
27 sendfile on;
28 #tcp_nopush on;
29
30 #keepalive_timeout 0;
31 keepalive_timeout 65;
32
33 #gzip on;
34
35 server {
36     listen 80;
37     server_name 127.0.0.1;
38
39     #charset koi8-r;
40
41     #access_log logs/host.access.log main;
42
43     location / {
44         root html;
45         index index.html index.htm;
46     }
47
48     location /api/ {
49         rewrite ^.+api/?(.*)$ /$1 break;
50         proxy_pass http://192.168.1.110:8081/;
51         proxy_redirect off;
52         proxy_set_header Host $host;
53         proxy_set_header X-Real-IP $remote_addr;
54         proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
55     }
56
57     #error_page 404 /404.html;
58
59     # redirect server error pages to the static page /50x.html
60     #
61     error_page 500 502 503 504 /50x.html;
62     location = /50x.html {
63         root html;
64     }
65
66     # proxy the PHP scripts to Apache listening on 127.0.0.1:80
67     #
68     #location ~ \.php$ {
69     #     proxy_pass http://127.0.0.1;
70     #}
71
72     # pass the PHP scripts to FastCGI server listening on 127.0.0.1:9000
73     #
74     #location ~ \.php$ {
75     #     root html;
76     #     fastcgi_pass 127.0.0.1:9000;
77     #     fastcgi_index index.php;
78     #     fastcgi_param SCRIPT_FILENAME /scripts$fastcgi_script_name;
79     #     include fastcgi_params;
80     #}
```

```

81
82     # deny access to .htaccess files, if Apache's document root
83     # concurs with nginx's one
84     #
85     #location ~ /\.ht {
86     #     deny  all;
87     #}
88 }
89
90
91     # another virtual host using mix of IP-, name-, and port-based configuration
92     #
93     #server {
94     #     listen      8000;
95     #     listen      somename:8080;
96     #     server_name  somename  alias  another.alias;
97
98     #     location / {
99     #         root     html;
100    #         index  index.html index.htm;
101    #     }
102    #}
103
104
105    # HTTPS server
106    #
107    #server {
108    #     listen      443 ssl;
109    #     server_name localhost;
110
111    #     ssl_certificate      cert.pem;
112    #     ssl_certificate_key  cert.key;
113
114    #     ssl_session_cache    shared:SSL:1m;
115    #     ssl_session_timeout  5m;
116
117    #     ssl_ciphers  HIGH:!aNULL:!MD5;
118    #     ssl_prefer_server_ciphers  on;
119
120    #     location / {
121    #         root     html;
122    #         index  index.html index.htm;
123    #     }
124    #}
125 }

```

- 默认监听端口为 **80**，可自行更改
- 默认监听地址为 **localhost**，可自行更改

```

1  server {
2      listen 80;
3      server_name localhost;
4      ...

```







添加代理

- 在 `server` 中配置;
- `/api/` 为vue中代理的值;
- `proxy_pass` 需要代理的IP地址, **必填**, 其他可忽略

```
1  location /api/ {
2      # 可省略开始
3      rewrite ^.+api/?(.*)$ /$1 break;    // 重定向
4      proxy_redirect off;
5      proxy_set_header Host $host;
6      proxy_set_header X-Real-IP $remote_addr;
7      proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
8      # 可省略结束
9
10     # 必填
11     proxy_pass http://192.168.1.10:8888/;    #node api server 即需要代理的IP地址
12 }
```

六、启动Nginx

双击 `nginx.exe` 进行启动 (一闪而过, 若出现命令行界面, 则说明配置有误)

名称	修改日期	类型	大小
 conf	2020/5/27 10:37	文件夹	
 contrib	2020/5/27 10:37	文件夹	
 docs	2020/5/27 10:37	文件夹	
 html	2020/5/27 11:03	文件夹	
 logs	2020/5/27 10:37	文件夹	
 temp	2020/5/27 10:37	文件夹	
 nginx.exe	2020/4/21 21:26	应用程序	3,630 KB

七、访问打包后的页面

打开浏览器, 访问 [第五节](#) 设置的地址和端口 (此处为localhost:80)

至此, nginx代理vue打包结束。