一、需求

使用 Kafka 做日志收集。

需要收集的信息：
1、用户 ID（user_id）
2、时间（act_time）
3、操作（action，可以是：点击：click，收藏：job_collect，投简历：cv_send，上传简历：cv_upload）
4、对方企业编码（job_code）

二、架构：

HTML+Nginx+ngx_kafka_module+Kafka

三、搭建过程

1、安装配置 zookeeper 和 kafka 集群

```
# 复用学习时的环境，不再赘述~
```

2、安装配置 nginx

```
#下载 Nginx
cd /opt/lagou/softwares
wget http://nginx.org/download/nginx-1.12.2.tar.gz
#解压 nginx
tar zxvf nginx-1.18.0.tar.gz -C /opt/lagou/server
#安装 ngx_kafka_module 依赖 librdkafka
yum install librdkafka-devel
#下载 ngx_kafka_module
git clone https://github.com/brg-liuwei/ngx_kafka_module
#编译安装 nginx
cd /opt/lagou/server/nginx-1.12.2
./configure --add-module=/opt/lagou/softwares/ngx_kafka_module
make && make install
```

```
#将可执行文件 nginx 软链接到 PATH 下：
ln -s /usr/local/nginx/sbin/nginx /usr/sbin/
#修改 nginx 配置文件：
vim /usr/local/nginx/conf/nginx.conf
#user  nobody;
worker_processes  1;
#error_log  logs/error.log;
#error_log  logs/error.log  notice;
#error_log  logs/error.log  info;
```

```
#pid        logs/nginx.pid;
events {
    worker_connections  1024;
}
http {
    include       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  logs/access.log  main;
    sendfile        on;
    #tcp_nopush     on;
    #keepalive_timeout  0;
    keepalive_timeout  65;
    #gzip  on;
# 引入 kafka 模块
    kafka;
    # 指定 broker 地址
    kafka_broker_list linux121:9092 linux122:9092 linux123:9092;
    server {
        listen       80;
        server_name  localhost;
        access_log  logs/access.log;
        location / {
            root   html;
            index  index.html index.htm;
        }
        location = /homework/action {
                # 指定发送的 topic
         kafka_topic topic_homework;
        }
    }
    server {
        listen       80;
        server_name  localhost;
        #charset koi8-r;
        #access_log  logs/host.access.log  main;
        location / {
            root   html;
            index  index.html index.htm;
        }
        #error_page  404              /404.html;
        # redirect server error pages to the static page /50x.html
```

```
        #
        error_page   500 502 503 504  /50x.html;
        location = /50x.html {
            root   html;
        }
        # proxy the PHP scripts to Apache listening on 127.0.0.1:80
        #
        #location ~ \.php$ {
        #    proxy_pass   http://127.0.0.1;
        #}
        # pass the PHP scripts to FastCGI server listening on 127.0.0.1:9000
        #
        #location ~ \.php$ {
        #    root           html;
        #    fastcgi_pass   127.0.0.1:9000;
        #    fastcgi_index  index.php;
        #    fastcgi_param  SCRIPT_FILENAME  /scripts$fastcgi_script_name;
        #    include        fastcgi_params;
        #}
        # deny access to .htaccess files, if Apache's document root
        # concurs with nginx's one
        #
        #location ~ /\.ht {
        #    deny  all;
        #}
    }
    # another virtual host using mix of IP-, name-, and port-based configuration
    #
    #server {
    #    listen       8000;
    #    listen       somename:8080;
    #    server_name  somename  alias  another.alias;
    #    location / {
    #        root   html;
    #        index  index.html index.htm;
    #    }
#}
    # HTTPS server
    #
    #server {
    #    listen       443 ssl;
    #    server_name  localhost;
    #    ssl_certificate      cert.pem;
    #    ssl_certificate_key  cert.key;
```

```
    #    ssl_session_cache    shared:SSL:1m;
    #    ssl_session_timeout  5m;
    #    ssl_ciphers  HIGH:!aNULL:!MD5;
    #    ssl_prefer_server_ciphers  on;
    #    location / {
    #        root   html;
    #        index  index.html index.htm;
    #    }
    #}
}
```

3、启动服务

#启动 zookeeper

#通过之前的脚本群起

```
cd /root
sh zk.sh start
```

#后台启动 kafka

```
kafka-server-start.sh -daemon /opt/lagou/servers/kafka_2.12-
1.0.2/config/server.properties"
```

#启动消费者消费主题：

```
kafka-console-consumer.sh --bootstrap-server linux43:9092 --topic topic_homework
```

#启动 nginx

```
nginx -s reload
```

4、编写 html 页面

#添加点击事件

```
<li ><button onclick="collect('click')">点击</button>
<li><button onclick="collect('collect')">收藏</button>
<li><button onclick="collect('send')">投简历</button>
<li><button onclick="collect('upload')">上传简历</button>
```

#点击事件 js 方法

```
 //收集操作日志
function collect(action) {
    let user_id = ['user1', 'user2', 'user3']
    let job_code = ['lagou', 'ali', 'tx'];
    $.ajax({
        url: 'http://linux121/homework/action',
        type: 'POST',
        contentType: 'application/json;charset=utf-8',
        dataType: 'json',
```

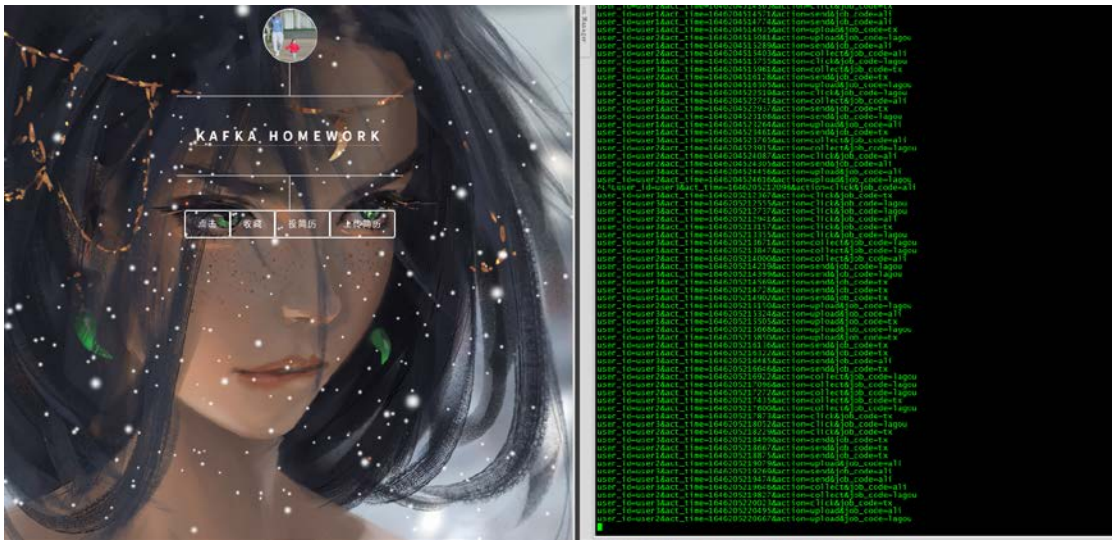
```
        data: {
            user_id: user_id[randomNum(0, 2)],
            act_time: new Date().getTime(),
            action: action,
            job_code: job_code[randomNum(0, 2)]
        },
        success: function(data) {


        }
    })
}


//生成从 minNum 到 maxNum 的随机数
function randomNum(minNum, maxNum) {
    switch (arguments.length) {
        case 1:
            return parseInt(Math.random() * minNum + 1, 10);
            break;
        case 2:
            return parseInt(Math.random() * (maxNum - minNum + 1) + minNum, 10);
            break;
        default:
            return 0;
            break;
    }
}
# 上传 html 至 nginx /usr/local/nginx/html/homework 目录下
```

四、操作验证

#访问 linux121/homework/index.html
#点击操作，可见消费者打印日志


KAFKA HOMEWORK
点击
收藏
投简历
上传简历