

# Unit08 快速就业课

## 如何检查错误

思路：

- 找到日志文件，检查日志内容
- 将根据错误分析原因
- 利用搜索引擎检查错误原因
- 设计解决方案，测试，有问题返回 1

检查日志： /var/log/XXX

检查日志命令（经典面试题目）：

- cat 显示文本文件的内容
- tac 倒序显示文件内容
- head 头 查看头部
  - head -n 10 查看头10行
- tail 尾巴 查看文件尾部
  - tail -n 10 查看文件的最后十行
- more 分屏显示文件的内容
  - more 文件
  - tac 文件 | more 将两个命令拼接起来，第一个名结果交给第二命令继续处理
- vim
- grep 搜索结果
  - cat 文件 | grep 正则 利用正则表达式搜索第一个命令的结果

实验, 显示elasticsearch的日志文件：

```
cd /var/log/elasticsearch
head -n 10 elasticsearch.log
head elasticsearch.log
cat elasticsearch.log
tac elasticsearch.log
tail -n 10 elasticsearch.log
more elasticsearch.log
tac elasticsearch.log | more
cat elasticsearch.log | grep ERROR
```

解决Elasticsearch6.8 启动问题：

找到问题关键字：

```
java.security.AccessControlException: access denied ("java.io.FilePermission"
"/etc/pki/java/cacerts" "read") elasticsearch
```

搜索得到结果：

<https://discuss.elastic.co/t/elasticsearch-is-incompatible-with-new-versions-of-openjdk-from-redhat/318928>

文章说明：JDK1.8 U345 版本以后有兼容问题：

- 解决办法，添加一个文件 /home/elasticsearch/.java.policy 文件

```
grant {  
    Permission java.io.FilePermission "/etc/pki/java/cacerts", "read";  
};
```

步骤：

```
mkdir /home/elasticsearch  
vim /home/elasticsearch/.java.policy # 文件内容略  
vim /etc/passwd # 设置 elasticsearch 的主目录  
chown elasticsearch:elasticsearch /home/elasticsearch -R # 修改文件的拥有者
```

/etc/passwd 文件内容：

```
elasticsearch:x:986:980:elasticsearch user:/home/elasticsearch:/sbin/nologin
```

## 面试题：你部署项目时候出现的难题是什么？

1. 遇到过ES软件和JDK的兼容性问题
2. 部署ES 6.8.0 在 RH8.6 或者 RockyLinux 8.6，时候JDK8出现了兼容问题  
日志中出现 证书文件读取权限不足的错误
3. 将日志错误在网上搜索，找到这个原因，在ES用户目录下添加了一个 权限文件 java.policy 文件，开放了证书文件的权限，就解决了。

## 部署简单的SpringBoot程序：

编写静态/resources/static/index.html

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
    <meta charset="UTF-8">  
    <title>Title</title>  
</head>  
<body>  
    <h1>Hello world!</h1>  
</body>  
</html>
```

控制器 /controller/DemoController.java

```
package cn.tedu.hello.controller;

import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RestController;

@RestController
public class DemoController {

    @GetMapping("/demo")
    public String demo(){
        return "Hello Demo";
    }
}
```

配置POM文件，设置可以自动执行：

参考文件：<https://docs.spring.io/spring-boot/docs/current/reference/html/deployment.html#deployment.installing.nix-services>

```
<plugin>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-maven-plugin</artifactId>
  <!-- 设置Spring Boot 打包以后可以自动启动 -->
  <configuration>
    <executable>true</executable>
  </configuration>
</plugin>
```

打包：

```
mvn package
```

打包后得到文件：hello-0.0.1-SNAPSHOT.jar

上传到服务器。

登录服务器以后使用java命令启动服务：

```
java -jar hello-0.0.1-SNAPSHOT.jar
```

启动以后要开放网络 8080 端口，才能用浏览器访问

关闭终端窗口（或者Ctrl + C），服务就会停止。

如何把 jar 文件安装为系统服务：（作业，尝试一下）

<https://docs.spring.io/spring-boot/docs/current/reference/html/deployment.html#deployment.installing.nix-services>