



## 自定义403处理方案

使用 Spring Security 时经常会看见 403（无权限），默认情况下显示的效果如下：

### Whitelabel Error Page

This application has no explicit mapping for /error, so you are seeing this as a fallback.

Tue Jul 16 16:44:08 CST 2019

There was an unexpected error (type=Forbidden, status=403).  
Forbidden

而在实际项目中可能都是一个异步请求，显示上述效果对于用户就不是特别友好了。Spring Security 支持自定义权限受限。

### 新建类

新建类实现 AccessDeniedHandler

```
package com.xxxx.springsecuritydemo.handler;

import org.springframework.security.access.AccessDeniedException;
import org.springframework.security.web.access.AccessDeniedHandler;
import org.springframework.stereotype.Component;

import javax.servlet.ServletException;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import java.io.IOException;
import java.io.PrintWriter;

/**
 * @author zhoubin
 * @since 1.0.0
 */
@Component
public class MyAccessDeniedHandler implements AccessDeniedHandler {

    @Override
    public void handle(HttpServletRequest request, HttpServletResponse response,
        AccessDeniedException accessDeniedException) throws IOException, ServletException {
        response.setStatus(HttpServletResponse.SC_FORBIDDEN);
        response.setHeader("Content-Type", "application/json;charset=utf-8");
        PrintWriter out = response.getWriter();
        out.write("{\"status\":\"error\",\"msg\":\"权限不足，请联系管理员!\"}");
        out.flush();
        out.close();
    }
}
```

## 修改配置类

配置类中重点添加异常处理器。设置访问受限后交给哪个对象进行处理。

myAccessDeniedHandler 是在配置类中进行自动注入的。

```
//异常处理
http.exceptionHandling()
    .accessDeniedHandler(myAccessDeniedHandler);
```

