<> Code · ⊙ **Issues** 20 · ⊗ Pull requests 9 · ▷ Actions · ▦ Projects · ⊘ Security · •••
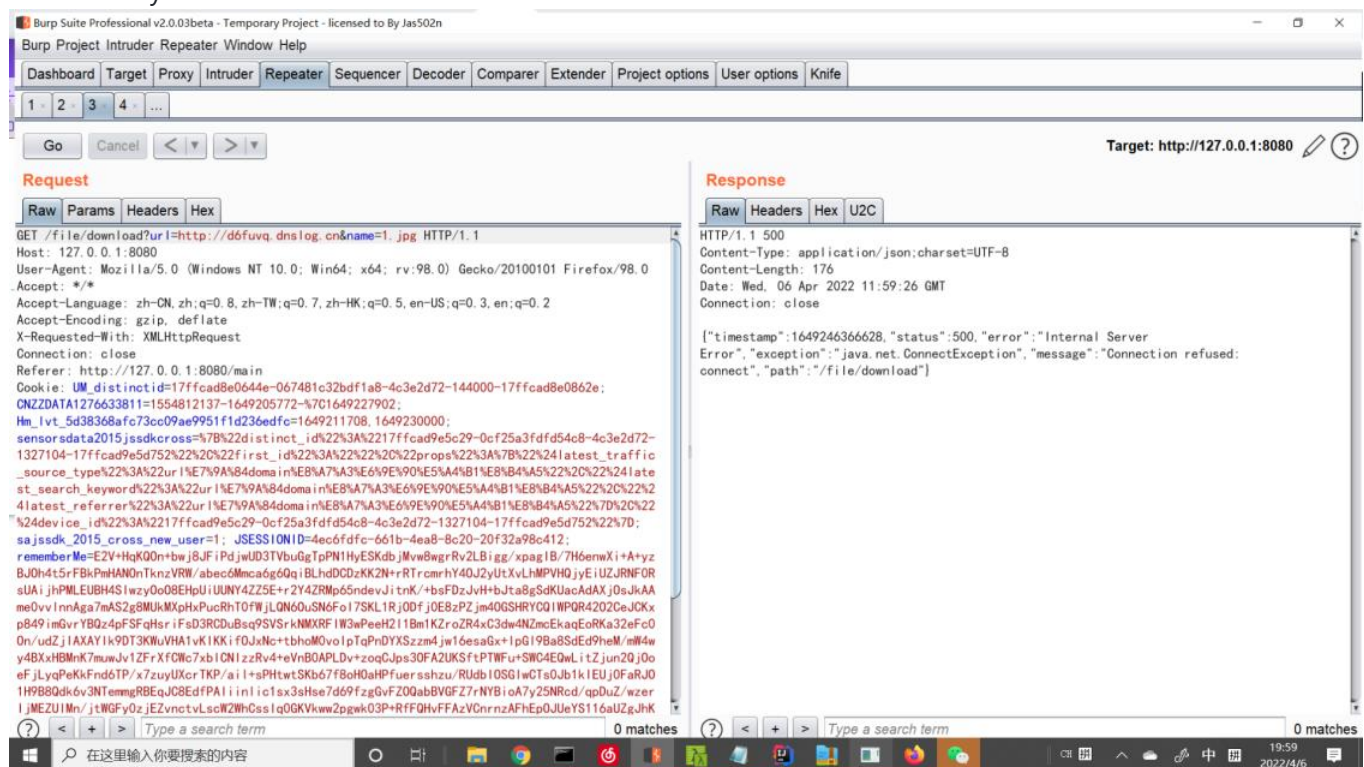
New issue

An SSRF vulnerability exists in the system #43

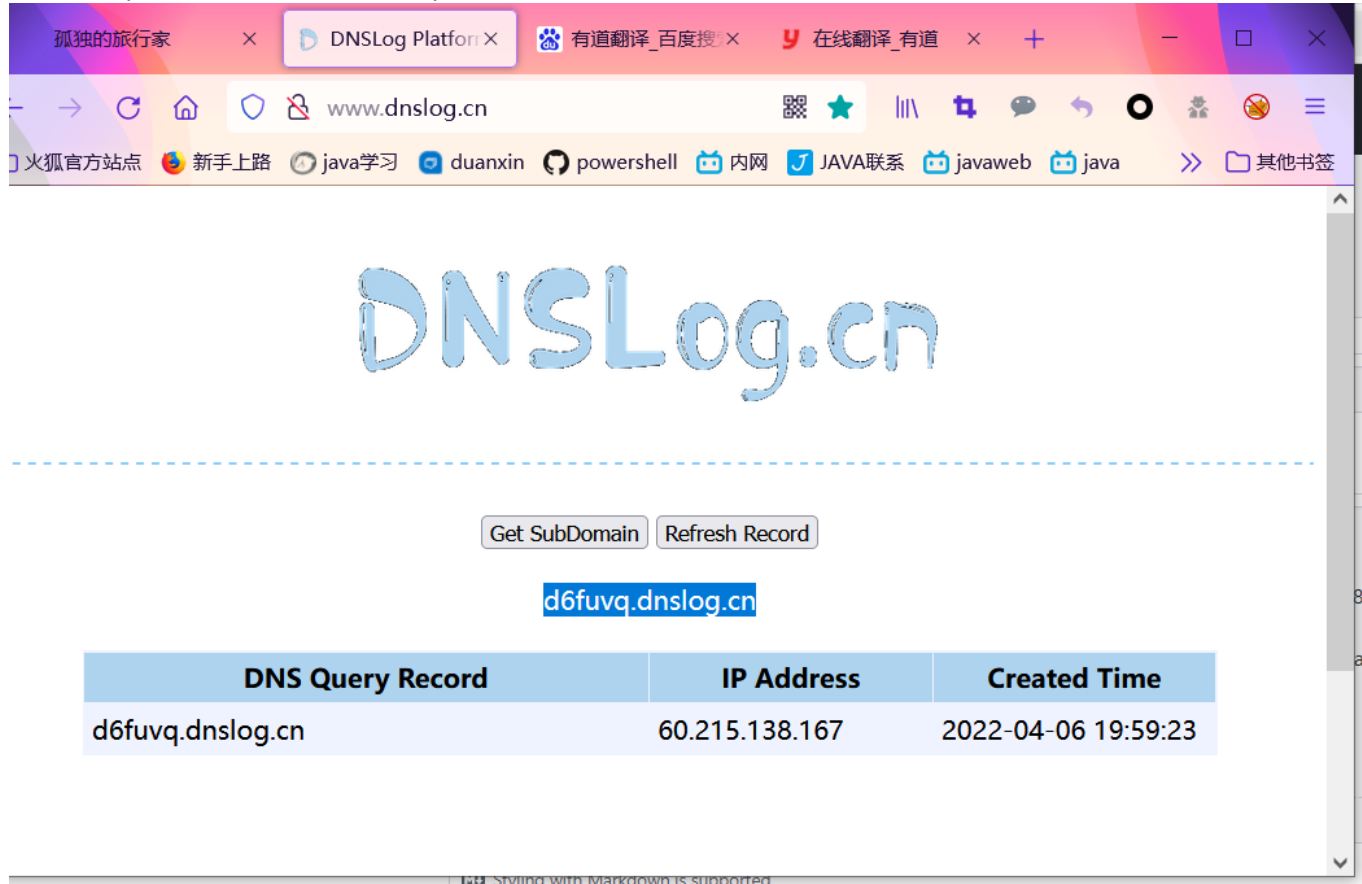⊙ Open · lanfei-4 opened this issue on Apr 6 · 0 comments

lanfei-4 commented on Apr 6

The problem code

Controllable URL parameters


```java
@GetMapping("download")
@ResponseBody
@SysLog("文件下载")
public RestResponse downFile(@RequestParam(value="url",required = false) String realurl,
                             @RequestParam(value="name",required = false) String name,
                             HttpServletResponse response) throws IOException {
    if(StringUtils.isBlank(realurl)){
        return RestResponse.failure("图片地址不能为空");
    }
    if(StringUtils.isBlank(name)){
        return RestResponse.failure("图片名称不能为空");
    }
    if("text/html".equals(ToolUtil.getContentType(name))){
        return RestResponse.failure("图片格式不正确");
    }
    name = new String(name.getBytes( charsetName: "GB2312"), charsetName: "ISO8859-1");
    URL url=new URL(realurl);
    HttpURLConnection conn=(HttpURLConnection)url.openConnection();
    conn.connect();
    BufferedInputStream br = new BufferedInputStream(conn.getInputStream());
    byte[] buf = new byte[1024];
    int len = 0;
    response.reset();
    response.setHeader( s: "Content-type", s1: "application/octet-stream");
    response.setHeader( s: "Content-Disposition", s1: "attachment; filename="+name);
    ServletOutputStream out = response.getOutputStream();
    while ((len = br.read(buf)) > 0) out.write(buf, off: 0, len);
    br.close();
    out.flush();
    out.close();
    conn.disconnect();
```

Vulnerability validation

The DNS platform received the request. Procedure

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

1 participant