WeBankPartners / **wecube-platform** Public

Code   Issues 20   Pull requests 9   Actions   Wiki   Security   ...

New issue   Jump to bottom

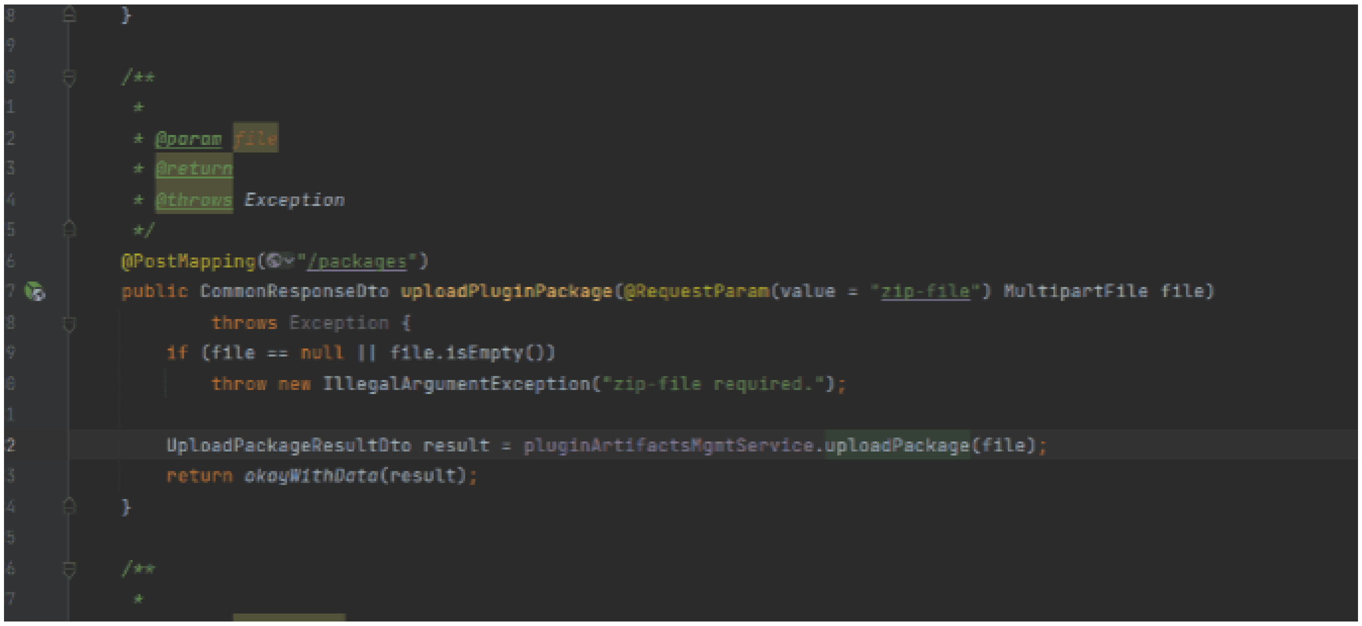
zip-slip #2324

Open **xiufengyue** opened this issue on May 30 · 0 comments

Labels   bug

xiufengyue commented on May 30 • edited

解压文件时未对 ../ 进行校验



```
    }

    /**
     *
     * @param file
     * @return
     * @throws Exception
     */
    @PostMapping("/packages")
    public CommonResponseDto uploadPluginPackage(@RequestParam(value = "zip-file") MultipartFile file)
            throws Exception {
        if (file == null || file.isEmpty())
            throw new IllegalArgumentException("zip-file required.");

        UploadPackageResultDto result = pluginArtifactsMgmtService.uploadPackage(file);
        return okayWithData(result);
    }

    /**
     *
```

Demo:

```
 packagecom.sp.test; importorg.apache.commons.fileupload.FileItem;
importorg.apache.commons.fileupload.FileItemFactory;
importorg.apache.commons.fileupload.disk.DiskFileItemFactory; importorg.apache.commons.io.FileUtils;
importorg.springframework.web.multipart.MultipartFile; import
org.springframework.web.multipart.commons.CommonsMultipartFile; importorg.xml.sax.SAXException;
importjava.io.*; importjava.text.SimpleDateFormat; importjava.util.Date; importjava.util.Enumeration;
importjava.util.zip.ZipEntry; importjava.util.zip.ZipFile; publicclassDemo{
publicstaticvoidmain(String[]args){ MultipartFilefile=fileToMultipartFile(new
File("D:\\project\\os\\ne1111w.zip")); uploadPackage(file); }
publicstaticUploadPackageResultDtouploadPackage(MultipartFile pluginPackageFile){
//1.savepackagefiletolocal StringtmpFileName=new
SimpleDateFormat("yyyyMMddHHmmssSSS").format(newDate());
FilelocalFilePath=newFile(SystemUtils.getTempFolderPath() +tmpFileName+"/"); try{
UploadPackageResultDtoresult= performUploadPackage(pluginPackageFile,localFilePath); returnresult;
}finally{ if(localFilePath!=null&&localFilePath.exists()){ FileUtils.deleteQuietly(localFilePath); } }
} publicstaticbooleanvalidateUploadFilename(Stringfilename){
if(filename.contains("\\")||filename.contains("/")){ returnfalse; } returntrue; }
publicstaticUploadPackageResultDto performUploadPackage(MultipartFilepluginPackageFile,File
localFilePath){ StringpluginPackageFileName=pluginPackageFile.getName();
if(!validateUploadFilename(pluginPackageFileName)){ thrownewRuntimeException(); }
if(!localFilePath.exists()){ if(localFilePath.mkdirs()){ //log.info("Createdirectory[{}]successful",
localFilePath.getAbsolutePath()); }else{ StringerrMsg=String.format("Createdirectory[%s]
failed.",localFilePath.getAbsolutePath()); thrownewRuntimeException("3099"); } } //TODO
Filedest=newFile(localFilePath,"/"+ pluginPackageFileName); try{ //log.info("newfilelocation:
{},filename:{}, canonicalpath:{},canonicalfilename:{}", //dest.getAbsoluteFile(),dest.getName(),
dest.getCanonicalPath(),dest.getCanonicalFile().getName()); pluginPackageFile.transferTo(dest);
}catch(IOExceptione){ //log.error("errorstotransferuploadedfiles.",e);
thrownewRuntimeException("Failedtouploadpackage,dueto "+e.getMessage()); }
UploadPackageResultDtoresult=null; try{ result=parsePackageFile(dest,localFilePath);
//log.info("Packageuploadedsuccessfully."); }catch(Exceptione){
//log.error("Errorstoparseuploadedpackagefile.",e);
thrownewRuntimeException("Failedtouploadpackagedueto "+e.getMessage()); } returnresult; }
publicstaticUploadPackageResultDtoparsePackageFile(Filedest,
FilelocalFilePath)throwsIOException,SAXException{ //2.unziplocalpackagefile
unzipLocalFile(dest.getCanonicalPath(), localFilePath.getCanonicalPath()+"/"); returnnull; }
publicstaticvoidunzipLocalFile(StringsourceZipFile,String destFilePath)throwsIOException{
try(ZipFilezipFile=newZipFile(sourceZipFile)){ Enumerationentries=zipFile.entries();
while(entries.hasMoreElements()){ ZipEntryentry=(ZipEntry)entries.nextElement();
StringzipEntryName=entry.getName(); //if(entry.isDirectory() ||!ACCEPTED_FILES.contains(zipEntryName)){
//continue; //} if(entry.isDirectory()){ FiledirFile=newFile(destFilePath,zipEntryName);
if(!dirFile.exists()){ booleanmkDirRet=dirFile.mkdirs(); if(mkDirRet){
//log.info("Createnewtemporaryfile:{}", destFilePath+zipEntryName); }else{
//log.info("Failedtocreatenewtemporary file:{}",destFilePath+zipEntryName); } } continue; }
if(newFile(destFilePath+ zipEntryName).createNewFile()){ //log.info("Createnewtemporaryfile:{}",
destFilePath+zipEntryName); } try(BufferedInputStreaminputStream=new
BufferedInputStream(zipFile.getInputStream(entry)); OutputStreamoutputStream=new
FileOutputStream(destFilePath+zipEntryName,true)){ byte[]buf=newbyte[2048]; intlen;
while((len=inputStream.read(buf))>0){ outputStream.write(buf,0,len); } }catch(Exceptione){
//log.error("Readinputstreammeeterror:",e); } } } //log.info("Zipfilehasunzipped!"); }
publicstaticclassSystemUtils{ privatestaticStringTEMP_FOLDER_PATH=null;
publicstaticStringgetTempFolderPath(){ if(TEMP_FOLDER_PATH==null){ TEMP_FOLDER_PATH=
System.getProperty("java.io.tmpdir"); if(!TEMP_FOLDER_PATH.endsWith("/")
&&!TEMP_FOLDER_PATH.endsWith("\\")){ TEMP_FOLDER_PATH=TEMP_FOLDER_PATH+"/"; } } returnTEMP_FOLDER_PATH;
} } publicclassUploadPackageResultDto{ privateStringid; privateStringname; privateStringversion;
privateStringstatus; privateStringuploadTimestamp; privateBooleanuiPackageIncluded;
privateStringedition; publicStringgetId(){ returnid; } publicvoidsetId(Stringid){ this.id=id; }
publicStringgetName(){ returnname; } publicvoidsetName(Stringname){ this.name=name; }
publicStringgetVersion(){ returnversion; } publicvoidsetVersion(Stringversion){ this.version=version; }
publicStringgetStatus(){ returnstatus; } publicvoidsetStatus(Stringstatus){ this.status=status; }
publicStringgetUploadTimestamp(){ returnuploadTimestamp; }
publicvoidsetUploadTimestamp(StringuploadTimestamp){ this.uploadTimestamp=uploadTimestamp; }
publicBooleangetUiPackageIncluded(){ returnuiPackageIncluded; }
publicvoidsetUiPackageIncluded(BooleanuiPackageIncluded){ this.uiPackageIncluded=uiPackageIncluded; }
publicStringgetEdition(){ returnedition; } publicvoidsetEdition(Stringedition){ this.edition=edition; }
} publicstaticMultipartFilefileToMultipartFile(Filefile){ FileItemfileItem=createFileItem(file);
MultipartFilemultipartFile=new CommonsMultipartFile(fileItem); returnmultipartFile; }
privatestaticFileItemcreateFileItem(Filefile){ FileItemFactoryfactory=newDiskFileItemFactory(16,null);
FileItemitem=factory.createItem("textField","text/plain", true,file.getName()); intbytesRead=0;
byte[]buffer=newbyte[8192]; try{ FileInputStreamfis=newFileInputStream(file);
OutputStreamos=item.getOutputStream(); while((bytesRead=fis.read(buffer,0,8192))!=-1){
os.write(buffer,0,bytesRead); } os.close(); fis.close(); }catch(IOExceptione){ e.printStackTrace(); }
returnitem; } }
```

xiufengyue added the bug label on May 30

Assignees

No one assigned

Labels

bug

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

1 participant