## CWE翻译计划

**贡献人员：** *Mardan & Leslie Shang & LJIJCJ & Michael Tan*

## CWE - 5 J2EE Misconfiguration: Data Transmission Without Encryption

中文：**J2EE配置错误：没有加密的数据传输**

### Description

* 通过网络发送的信息在传输过程中可能会受到影响。如果数据以明文发送或者是弱加密，则攻击者可能能够读取或修改内容。

### Extended Description

* 没有详细描述

## CWE - 6 J2EE Misconfiguration: Insufficient Session-ID Length

中文：**J2EE配置错误：会话ID长度不足**

### Description

* J2EE应用程序配置为使用不足的会话ID长度。

### Extended Description

* 如果攻击者可以猜测或窃取会话ID，那么他们就可以接管用户的会话（称为会话劫持）。可能的会话ID的数量随着会话ID长度的增加而增加，使得猜测或窃取会话ID变得更加困难。

## CWE - 7 J2EE Misconfiguration: Missing Custom Error Page

中文：**J2EE配置错误：缺少自定义错误页面**

### Description

* Web应用程序的默认错误页面不应显示有关软件系统的敏感信息。

### Extended Description

* Web应用程序必须为4xx错误（例如404），5xx（例如500）错误定义默认错误页面并捕获java.lang.Throwable异常以防止攻击者从应用程序容器的内置错误响应中挖掘信息。 当攻击者探索寻找漏洞的网站时，该网站提供的信息量对于任何企图攻击的最终成功或失败至关重要。

## CWE - 8 J2EE Misconfiguration: Entity Bean Declared Remote

中文：**J2EE配置错误：实体Bean声明为远程**

### Description

* 当应用程序公开实体bean的远程接口时，它还可能公开获取或设置bean数据的方法。可以利用这些方法来读取敏感信息，或以违反应用程序期望的方式更改数据，从而可能导致其他漏洞。

### Extended Description

* 没有详细描述

## CWE - 9 J2EE Misconfiguration: Weak Access Permissions for EJB Methods

中文：**J2EE配置错误：EJB方法的弱访问权限**

### Description

* 如果将提升的访问权限分配给EJB方法，则攻击者可以利用这些权限来利用软件系统。

### Extended Description

* 如果EJB部署描述符包含一个或多个授予对特殊ANYONE角色的访问权限的方法权限，则表示尚未完全考虑应用程序的访问控制，或者应用程序的结构是以合理的访问控制限制为不可能。

## CWE - 11 ASP.NET Misconfiguration: Creating Debug Binary

中文：**ASP.NET配置错误：创建调试二进制文件**

### Description

* 调试消息可帮助攻击者了解系统并规划一种攻击形式。

### Extended Description

* 可以将ASP .NET应用程序配置为生成调试二进制文件。这些二进制文件提供详细的调试消息，不应在生产环境中使用。调试二进制文件旨在用于开发或测试环境，如果将它们部署到生产环境中，则可能会带来安全风险。

## CWE - 12 ASP.NET Misconfiguration: Missing Custom Error Page

中文：**ASP.NET配置错误：缺少自定义错误页面**

### Description

* ASP .NET应用程序必须启用自定义错误页面，以防止攻击者从框架的内置响应中挖掘信息。

### Extended Description

* 没有详细描述

## CWE - 13 ASP.NET Misconfiguration: Password in Configuration File

中文：**ASP.NET配置错误：配置文件中的密码**

### Description

* 在配置文件中存储明文密码允许任何能够读取文件的人访问受密码保护的资源，使其成为攻击者的轻松目标。

### Extended Description

* 没有详细描述

## CWE - 14 Compiler Removal of Code to Clear Buffers

中文：**编译器删除代码以清除缓冲区**

### Description

* 根据源代码清除敏感内存，但编译器优化会在不再次读取内存时保持内存不变，即“死存储删除”。

### Extended Description

* 发生以下编译器优化错误：

1.秘密数据存储在内存中。 2.通过覆盖其内容从内存中清除秘密数据。 3.使用优化编译器编译源代码，该编译器识别并删除将内容覆盖为死存储的函数，因为随后不使用该存储器。

## CWE - 15 External Control of System or Configuration Setting

中文：**系统外部控制或配置设置**

### Description

* 一个或多个系统设置或配置元素可以由用户外部控制。

### Extended Description

* 允许外部控制系统设置可能会中断服务或导致应用程序以意外和潜在的恶意方式运行。

## CWE - 20 Improper Input Validation

中文：**输入验证不正确**

### Description

* 产品不验证或错误验证可能影响程序的控制流或数据流的输入。

### Extended Description

* 当软件未正确验证输入时，攻击者能够以应用程序其余部分不期望的形式制作输入。这将导致系统的某些部分接收到非预期的输入，这可能导致控制流的改变，资源的任意控制或任意代码执行。

## CWE - 22 Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')

中文：**限制目录的路径名的不正确限制（“路径遍历”）**

### Description

* 该软件使用外部输入来构造路径名，该路径名用于标识位于受限父目录下的文件或目录，但该软件未正确中和路径名中可能导致路径名解析到的位置的特殊元素在受限制的目录之外。

### Extended Description

* 许多文件操作都是在受限目录中进行的。通过使用诸如“..”和“/”分隔符之类的特殊元素，攻击者可以在受限制的位置之外逃逸，以访问系统中其他位置的文件或目录。最常见的特殊元素之一是“../”序列，在大多数现代操作系统中，它被解释为当前位置的父目录。这被称为相对路径遍历。路径遍历还包括使用绝对路径名，例如“/ usr / local / bin”，这在访问意外文件时也很有用。这被称为绝对路径遍历。 在许多编程语言中，注入空字节（0或NUL）可能允许攻击者截断生成的文件名以扩大攻击范围。例如，软件可以将“.txt”添加到任何路径名，从而将攻击者限制为文本文件，但空注入可以有效地消除此限制。

## CWE - 23 Relative Path Traversal

中文：**相对路径遍历**

### Description

* 该软件使用外部输入来构造应该在受限目录内的路径名，但它不能正确地中和可以解析到该目录之外的位置的序列，例如“..”。

### Extended Description

* 这允许攻击者遍历文件系统以访问受限目录之外的文件或目录。

## CWE - 24 Path Traversal: '../filedir'

中文：**路径遍历：'../filedir'**

### Description

* 该软件使用外部输入来构造应该在受限目录内的路径名，但它不能正确地中和可以解析到该目录之外的位置的“../”序列。

### Extended Description

* 这允许攻击者遍历文件系统以访问受限目录之外的文件或目录。 “../”操作是对使用“/”作为目录分隔符的操作系统的规范操作，例如基于UNIX和Linux的系统。在某些情况下，在支持“/”但不支持主分隔符的环境中绕过保护方案很有用，例如Windows，它使用“\”但也可以接受“/”。

## CWE - 25 Path Traversal: '/../filedir'

中文：**路径遍历：'/../filedir'**

### Description

* 该软件使用外部输入来构造应该在受限目录内的路径名，但它不能正确地中和可以解析到该目录之外的位置的“/../”序列。

### Extended Description

* 这允许攻击者遍历文件系统以访问受限目录之外的文件或目录。 有时程序会在输入开头检查“../”，因此“/../”可以绕过该检查。

## CWE - 26 Path Traversal: '/dir/../filename'

中文：**路径遍历：'/ dir /../filename'**

### Description

* 该软件使用外部输入来构造应该在受限目录内的路径名，但它不能正确地中和可以解析到该目录之外的位置的“/dir/../filename”序列。

### Extended Description

* 这允许攻击者遍历文件系统以访问受限目录之外的文件或目录。 '/dir/../filename'操作对于绕过某些路径遍历保护方案很有用。有时程序只在输入开头检查“../”，因此“/../”可以绕过该检查。

## CWE - 27 Path Traversal: 'dir/../../filename'

中文：**路径遍历：'dir /../../ filename'**

### Description

* 该软件使用外部输入来构造应该在受限目录内的路径名，但它不能正确地中和可以解析到该目录之外的位置的多个内部“../”序列。

### Extended Description

* 这允许攻击者遍历文件系统以访问受限目录之外的文件或目录。 '目录/../../ filename'操作对于绕过某些路径遍历保护方案很有用。有时程序只删除一个“../”序列，因此多个“../”可以绕过该检查。或者，此操作可用于绕过路径名开头的“../”检查，向上移动多个目录级别。

## CWE - 28 Path Traversal: '..\filedir'

中文：**路径遍历：'.. \ filedir'**

### Description

* 该软件使用外部输入来构造应该在受限目录内的路径名，但它不能正确地中和可以解析到该目录之外的位置的“..”序列。

### Extended Description

* 这允许攻击者遍历文件系统以访问受限目录之外的文件或目录。 '.. \'操作是对使用“\”作为目录分隔符（如Windows）的操作系统的规范操作。但是，它也可用于绕过仅假设“/”分隔符有效的路径遍历保护方案。

## CWE - 29 Path Traversal: '..\filename'

中文：**路径遍历：'\ .. \ filename'**

### Description

* 该软件使用外部输入来构造应该在受限目录内的路径名，但它不能正确地中和可以解析到该目录之外的位置的'\ .. \ filename'（前导反斜点点）序列。

### Extended Description

* 这允许攻击者遍历文件系统以访问受限目录之外的文件或目录。 这类似于CWE-25，除了使用“\”而不是“/”。有时程序会在输入的开头检查“.. \”，因此“\ .. \”可以绕过该检查。它也可用于绕过仅假设“/”分隔符有效的路径遍历保护方案。

## CWE - 30 Path Traversal: '\dir..\filename'

中文：**路径遍历：'\ dir \ .. \ filename'**

### Description

* 该软件使用外部输入来构造一个应该在受限目录内的路径名，但是它没有正确地中和'\ dir \ .. \ filename'（前导反斜点点）序列，这些序列可以解析到该区域之外的位置目录。

### Extended Description

* 这允许攻击者遍历文件系统以访问受限目录之外的文件或目录。 这类似于CWE-26，除了使用“\”而不是“/”。 '\ dir \ .. \ filename'操作对于绕过某些路径遍历保护方案很有用。有时程序只在输入的开头检查“.. \”，因此“\ .. \”可以绕过该检查。

## CWE - 31 Path Traversal: 'dir....\filename'

中文：**路径遍历：'dir \ .. \ .. \ filename'**

### Description

* 该软件使用外部输入来构造一个应该在受限目录内的路径名，但是它没有正确地中和'dir \ .. \ .. \ filename'（多个内部反斜杠点点）序列，这些序列可以解析为一个位置。在该目录之外。

### Extended Description

* 这允许攻击者遍历文件系统以访问受限目录之外的文件或目录。 'dir \ .. \ .. \ filename'操作对于绕过某些路径遍历保护方案很有用。有时程序只删除一个“..”序列，因此多个“.. \”可以绕过该检查。或者，此操作可用于绕过路径名开头的“.. \”检查，向上移动多个目录级别。

## CWE - 32 Path Traversal: '...' (Triple Dot)

中文：**路径遍历：'...'（三点）**

### Description

* 该软件使用外部输入来构造应该在受限目录内的路径名，但它不能正确地中和可以解析到该目录之外的位置的“...”（三点）序列。

### Extended Description

* 这允许攻击者遍历文件系统以访问受限目录之外的文件或目录。 “...”操作对于绕过某些路径遍历保护方案很有用。在某些Windows系统上，它等同于“.. \ ..”并可能绕过假设只有两个点有效的检查。不完整的过滤，例如去除“./”序列，最终会产生有效的“..”序列，因为它们会陷入不安全的值（CWE-182）。

## CWE - 33 Path Traversal: '....' (Multiple Dot)

中文：**路径遍历：'....'（多点）**

### Description

* 该软件使用外部输入来构造应该在受限目录内的路径名，但它不能正确地中和可以解析到该目录之外的位置的“......”（多点）序列。

### Extended Description

* 这允许攻击者遍历文件系统以访问受限目录之外的文件或目录。 '....'操作对于绕过某些路径遍历保护方案很有用。在某些Windows系统上，它相当于“.. \ .. \ ..”并可能绕过假设只有两个点有效的检查。不完整的过滤，例如去除“./”序列，最终会产生有效的“..”序列，因为它们会陷入不安全的值（CWE-182）。

## CWE - 34 Path Traversal: '....//'

中文：**路径遍历：'.... //'**

### Description

* 该软件使用外部输入来构造一个应该在受限目录内的路径名，但是它没有正确地中和可以解析到该目录之外的位置的“.... //”（加倍点点斜杠）序列。 。

### Extended Description

* 这允许攻击者遍历文件系统以访问受限目录之外的文件或目录。 '.... //'操作对于绕过某些路径遍历保护方案很有用。如果“../”按顺序方式过滤，如某些正则表达式引擎所做，则“.... //”可以折叠为“../”不安全值（CWE-182）。如果操作系统将“//”和“/”视为等效，则删除“..”时也可能有用。

## CWE - 35 Path Traversal: '.../...//'

中文：**路径遍历：'... / ... //'**

### Description

* 该软件使用外部输入来构造一个应该在受限目录内的路径名，但是它没有正确地中和可以解析到外部位置的'... / ... //'（加倍的三点斜杠）序列那个目录。

### Extended Description

* 这允许攻击者遍历文件系统以访问受限目录之外的文件或目录。 '... / ... //'操作对于绕过某些路径遍历保护方案很有用。如果“../”按顺序方式过滤，如某些正则表达式引擎所做，则“... / ... //”可以折叠为“../”不安全值（CWE-182）。删除第一个“../”会产生“.... //”;第二次删除产生“../”。根据算法，软件可能对CWE-34敏感，但对CWE-35不敏感，反之亦然。

## CWE - 36 Absolute Path Traversal

中文：**绝对路径遍历**

### Description

* 该软件使用外部输入来构造应该在受限目录内的路径名，但它不能正确地中和绝对路径序列，例如“/ abs / path”，它可以解析到该目录之外的位置。

### Extended Description

* 这允许攻击者遍历文件系统以访问受限目录之外的文件或目录。

## CWE - 37 Path Traversal: '/absolute/pathname/here'

中文：**路径遍历：'/ absolute / pathname / here'**

### Description

* 在没有适当验证的情况下以斜杠绝对路径（'/ absolute / pathname / here'）的形式接受输入的软件系统可以允许攻击者将文件系统遍历到非预期的位置或访问任意文件。

### Extended Description

* 没有详细描述

## CWE - 38 Path Traversal: '\absolute\pathname\here'

中文：**路径遍历：'\ absolute \ pathname \ here'**

### Description

* 在没有经过适当验证的情况下以反斜杠绝对路径（'\ absolute \ pathname \ here'）的形式接受输入的软件系统可以允许攻击者将文件系统遍历到非预期的位置或访问任意文件。

### Extended Description

* 没有详细描述

## CWE - 39 Path Traversal: 'C:dirname'

中文：**路径遍历：'C：dirname'**

### Description

* 攻击者可以将驱动器号或Windows卷号（“C：dirname”）注入软件系统，以潜在地重定向对非预期位置或任意文件的访问。

### Extended Description

* 没有详细描述

## CWE - 40 Path Traversal: '\UNC\share\name\' (Windows UNC Share)

中文：**路径遍历：'\ UNC \ share \ name \'（Windows UNC Share）**

### Description

* 攻击者可以将Windows UNC共享（'\ UNC \ share \ name'）注入软件系统，以潜在地重定向对非预期位置或任意文件的访问。

### Extended Description

* 没有详细描述

## CWE - 41 Improper Resolution of Path Equivalence

中文：**路径等价的不正确解决方案**

### Description

* 系统或应用程序容易受到路径等效的文件系统内容泄露。路径等效涉及在文件和目录名称中使用特殊字符。相关的操作旨在为同一对象生成多个名称。

### Extended Description

* 通常采用路径等价来规避使用不完整的文件名或文件路径表示表示的访问控制。这与路径遍历不同，其中执行操作以生成不同对象的名称。

## CWE - 42 Path Equivalence: 'filename.' (Trailing Dot)

中文：**路径等价：'文件名。' （尾随点）**

### Description

* 在没有适当验证的情况下接受尾随点（'filedir。'）形式的路径输入的软件系统可能导致模糊的路径解析，并允许攻击者将文件系统遍历到非预期的位置或访问任意文件。

### Extended Description

* 没有详细描述

## CWE - 43 Path Equivalence: 'filename....' (Multiple Trailing Dot)

中文：**路径等价：'filename ....'（多个尾随点）**

### Description

* 在没有适当验证的情况下接受多个尾随点（'filedir ....'）形式的路径输入的软件系统可能导致模糊的路径解析，并允许攻击者将文件系统遍历到非预期的位置或访问任意文件。

### Extended Description

* 没有详细描述

## CWE - 44 Path Equivalence: 'file.name' (Internal Dot)

中文：**路径等价：'file.name'（内部点）**

### Description

* 在没有适当验证的情况下接受内部点（'file.ordir'）形式的路径输入的软件系统可能导致模糊的路径解析，并允许攻击者将文件系统遍历到非预期的位置或访问任意文件。

### Extended Description

* 没有详细描述

## CWE - 45 Path Equivalence: 'file...name' (Multiple Internal Dot)

中文：**路径等价：'文件...名称'（多个内部点）**

### Description

* 在没有适当验证的情况下接受多个内部点（'file ... dir'）形式的路径输入的软件系统可能导致模糊的路径解析，并允许攻击者将文件系统遍历到非预期的位置或访问任意文件。

### Extended Description

* 没有详细描述

## CWE - 46 Path Equivalence: 'filename ' (Trailing Space)

中文：**路径等价：'文件名'（尾随空格）**

### Description

* 在没有适当验证的情况下接受尾随空间（'filedir'）形式的路径输入的软件系统可能导致模糊的路径解析，并允许攻击者将文件系统遍历到非预期的位置或访问任意文件。

### Extended Description

* 没有详细描述

## CWE - 47 Path Equivalence: ' filename' (Leading Space)

中文：**路径等价：'文件名'（前导空格）**

### Description

* 在没有适当验证的情况下接受前导空间（'filedir'）形式的路径输入的软件系统可能导致模糊的路径解析，并允许攻击者将文件系统遍历到非预期的位置或访问任意文件。

### Extended Description

* 没有详细描述

## CWE - 48 Path Equivalence: 'file name' (Internal Whitespace)

中文：**路径等价：'文件名'（内部空白）**

### Description

* 在没有经过适当验证的情况下接受内部空间（'文件（SPACE）名称'）形式的路径输入的软件系统可能导致模糊的路径解析，并允许攻击者将文件系统遍历到非预期的位置或访问任意文件。

### Extended Description

* 没有详细描述

## CWE - 49 Path Equivalence: 'filename/' (Trailing Slash)

中文：**路径等价：'filename /'（尾随斜线）**

### Description

* 在没有适当验证的情况下以尾部斜杠（'filedir /'）的形式接受路径输入的软件系统可能导致模糊的路径解析，并允许攻击者将文件系统遍历到非预期的位置或访问任意文件。

### Extended Description

* 没有详细描述

## CWE - 50 Path Equivalence: '//multiple/leading/slash'

中文：**路径等价：'// multiple / leading / slash'**

### Description

* 在没有适当验证的情况下以多个前导斜杠（'// multiple / leading / slash'）的形式接受路径输入的软件系统可能导致模糊的路径解析并允许攻击者将文件系统遍历到非预期的位置或访问任意文件。

### Extended Description

* 没有详细描述

## CWE - 51 Path Equivalence: '/multiple//internal/slash'

中文：**路径等价：'/ multiple // internal / slash'**

### Description

* 在没有适当验证的情况下以多个内部斜杠（'/ multiple // internal / slash /'）的形式接受路径输入的软件系统可能导致模糊路径解析并允许攻击者将文件系统遍历到非预期的位置或任意访问文件。

### Extended Description

* 没有详细描述

## CWE - 52 Path Equivalence: '/multiple/trailing/slash//'

中文：**路径等价：'/ multiple / trailing / slash //'**

### Description

* 在没有适当验证的情况下以多个尾部斜杠（'/ multiple / trailing / slash //'）的形式接受路径输入的软件系统可能导致模糊的路径解析并允许攻击者将文件系统遍历到非预期的位置或任意访问文件。

### Extended Description

* 没有详细描述

## CWE - 53 Path Equivalence: '\multiple\internal\backslash'

中文：**路径等价：'\ multiple \ internal \ backslash'**

### Description

* 在没有经过适当验证的情况下以多个内部反斜杠（'\ multiple \ trailing \ slash'）的形式接受路径输入的软件系统可能导致模糊的路径解析并允许攻击者将文件系统遍历到非预期的位置或访问任意文件。

### Extended Description

* 没有详细描述

## CWE - 54 Path Equivalence: 'filedir\' (Trailing Backslash)

中文：**路径等价：'filedir'（尾随反斜杠）**

### Description

* 在没有适当验证的情况下以尾随反斜杠（'filedir \'）的形式接受路径输入的软件系统可能导致模糊的路径解析，并允许攻击者将文件系统遍历到非预期的位置或访问任意文件。

### Extended Description

* 没有详细描述

## CWE - 55 Path Equivalence: '/./' (Single Dot Directory)

中文：**路径等价：'/。/'（单点目录）**

### Description

* 在没有经过适当验证的情况下接受单点目录漏洞（'/./'）形式的路径输入的软件系统可能导致模糊的路径解析，并允许攻击者将文件系统遍历到非预期的位置或访问任意文件。

### Extended Description

* 没有详细描述

## CWE - 56 Path Equivalence: 'filedir\*' (Wildcard)

中文：**路径等价：'filedir \*'（通配符）**

### Description

* 在没有经过适当验证的情况下接受星号通配符（'filedir \*'）形式的路径输入的软件系统可能导致模糊的路径解析，并允许攻击者将文件系统遍历到非预期的位置或访问任意文件。

### Extended Description

* 没有详细描述

## CWE - 57 Path Equivalence: 'fakedir/../realdir/filename'

中文：**路径等价：'fakedir /../ realdir / filename'**

### Description

* 该软件包含限制对“realdir / filename”的访问的保护机制，但它使用“fakedir /../ realdir / filename”形式的外部输入构造路径名，这些输入不由这些机制处理。这允许攻击者对目标文件执行未经授权的操作。

### Extended Description

* 没有详细描述

## CWE - 58 Path Equivalence: Windows 8.3 Filename

中文：**路径等效：Windows 8.3文件名**

### Description

* 该软件包含一个保护机制，限制在Windows操作系统上访问长文件名，但该软件没有正确限制对等效的短“8.3”文件名的访问。

### Extended Description

* 在以后的Windows操作系统中，文件可以具有“长名称”和与旧Windows文件系统兼容的短名称，文件名最多8个字符，扩展名最多3个字符。因此，这些“8.3”文件名充当具有长名称的文件的备用名称，因此它们是有用的路径名等效操作。

## CWE - 59 Improper Link Resolution Before File Access ('Link Following')

中文：**文件访问前的链接解析不正确（'链接跟随'）**

### Description

* 软件会尝试根据文件名访问文件，但不能正确阻止该文件名识别解析为非预期资源的链接或快捷方式。

### Extended Description

* 没有详细描述

## CWE - 61 UNIX Symbolic Link (Symlink) Following

中文：**UNIX符号链接（符号链接）以下**

### Description

* 打开文件或目录时，该软件无法充分说明文件是何时解析为目标控制范围之外的目标的符号链接。这可能允许攻击者使软件对未经授权的文件进行操作。

### Extended Description

* 允许UNIX符号链接（符号链接）作为路径的一部分的软件系统，无论是在内部代码中还是通过用户输入，都允许攻击者欺骗符号链接并将文件系统遍历到非预期的位置或访问任意文件。符号链接可以允许攻击者读取/写入/损坏他们最初没有访问权限的文件。

## CWE - 62 UNIX Hard Link

中文：**UNIX硬链接**

### Description

* 当打开文件或目录时，该软件不足以说明名称何时与目标控制范围之外的目标的硬链接相关联。这可能允许攻击者使软件对未经授权的文件进行操作。

### Extended Description

* 系统检查硬链接失败可能导致不同类型的攻击。例如，如果特权程序使用的文件被替换为敏感文件的硬链接（例如/ etc / passwd），则攻击者可以升级其权限。当进程打开文件时，攻击者可以承担该进程的权限。

## CWE - 64 Windows Shortcut Following (.LNK)

中文：**Windows快捷方式（.LNK）**

### Description

* 打开文件或目录时，如果文件是目标位于预期控制范围之外的Windows快捷方式（.LNK），则该软件无法充分处理。这可能允许攻击者使软件对未经授权的文件进行操作。

### Extended Description

* 快捷方式（扩展名为.lnk的文件）可以允许攻击者读取/写入他们最初没有访问权限的文件。

## CWE - 65 Windows Hard Link

中文：**Windows硬链接**

### Description

* 当打开文件或目录时，该软件在名称与目标控制范围之外的目标的硬链接相关联时无法充分处理。这可能允许攻击者使软件对未经授权的文件进行操作。

### Extended Description

* 系统检查硬链接失败可能导致不同类型的攻击。例如，如果特权程序使用的文件被替换为敏感文件的硬链接（例如AUTOEXEC.BAT），则攻击者可以升级其权限。当进程打开文件时，攻击者可以承担该进程的权限，或阻止程序准确处理数据。

## CWE - 66 Improper Handling of File Names that Identify Virtual Resources

中文：**识别虚拟资源的文件名处理不当**

### Description

* 产品不处理或错误处理标识“虚拟”资源的文件名，该文件名未在与文件名关联的目录中直接指定，导致产品对非资源上的资源执行基于文件的操作文件。

### Extended Description

* 虚拟文件名称表示为普通文件名，但它们实际上是其他资源的别名，这些资源的行为与普通文件不同。根据其功能，它们可以是备用实体。它们不一定列在目录中。

## CWE - 67 Improper Handling of Windows Device Names

中文：**Windows设备名称处理不当**

### Description

* 该软件从用户输入构造路径名，但它不处理或错误地处理包含Windows设备名称（如AUX或CON）的路径名。当应用程序尝试将路径名作为常规文件处理时，这通常会导致拒绝服务或信息泄露。

### Extended Description

* 未正确处理虚拟文件名（例如AUX，CON，PRN，COM1，LPT1）可能导致不同类型的漏洞。在某些情况下，攻击者可以通过在URL中注入虚拟文件名来请求设备，这可能导致导致拒绝服务的错误或显示敏感信息的错误页面。允许设备名称绕过过滤的软件系统会冒着攻击者在具有设备名称的文件中注入恶意代码的风险。

## CWE - 69 Improper Handling of Windows ::DATA Alternate Data Stream

中文：**Windows :: DATA备用数据流处理不当**

### Description

* 该软件无法正确阻止对备用数据流（ADS）的访问或检测其使用。

### Extended Description

* 攻击者可以使用ADS从系统或文件浏览器工具（如Windows资源管理器）和命令行实用程序中的“dir”隐藏有关文件的信息（例如，大小，进程名称）。或者，攻击者可能能够绕过关联数据分叉的预期访问限制。

## CWE - 71 DEPRECATED: Apple '.DS\_Store'

中文：**弃用：Apple'.DS\_Store'**

### Description

* 此条目已被弃用，因为它表示UNIX Hard Link弱点类型的特定观察示例，而不是其自身的个别弱点类型。请参阅CWE-62。

### Extended Description

* 没有详细描述

## CWE - 72 Improper Handling of Apple HFS+ Alternate Data Stream Path

中文：**Apple HFS +备用数据流路径处理不当**

### Description

* 该软件无法正确处理可能识别HFS +文件系统上文件的数据或资源分支的特殊路径。

### Extended Description

* 如果软件根据文件名选择要采取的操作，则如果攻击者提供数据或资源分支，则软件可能会采取意外操作。此外，如果软件打算限制对文件的访问，则攻击者仍可能通过请求该文件的数据或资源分支来绕过预期的访问限制。

## CWE - 73 External Control of File Name or Path

中文：**文件名或路径的外部控制**

### Description

* 该软件允许用户输入来控制或影响文件系统操作中使用的路径或文件名。

### Extended Description

* 这可能允许攻击者访问或修改对应用程序至关重要的系统文件或其他文件。 满足以下两个条件时会发生路径操作错误：

1.攻击者可以指定文件系统上的操作中使用的路径。 2.通过指定资源，攻击者获得了不允许的功能。

例如，程序可能使攻击者能够覆盖指定的文件或使用受攻击者控制的配置运行。

## CWE - 74 Improper Neutralization of Special Elements in Output Used by a Downstream Component ('Injection')

中文：**下游组件使用的输出中特殊元素的中和不当（'注入'）**

### Description

* 该软件使用来自上游组件的外部影响输入构造命令，数据结构或记录的全部或部分，但它不会中和或不正确地中和可能修改它被发送到下游组件。

### Extended Description

* 软件对数据和控制的构成分别有一定的假设。缺乏对用户控制输入的这些假设的验证导致注入问题。注射问题包含各种各样的问题 - 所有问题都以非常不同的方式得到缓解，并且通常是为了改变过程的控制流程。因此，讨论这些弱点的最有效方法是注意将它们归类为注入弱点的独特特征。需要注意的最重要的问题是所有注入问题都有一个共同点 - 即，它们允许将控制平面数据注入用户控制的数据平面。这意味着可以通过合法数据通道发送代码来改变进程的执行，而不使用其他机制。虽然缓冲区溢出和许多其他缺陷涉及使用一些进一步的问题来获得执行，但注入问题仅需要解析数据。这类弱点的最经典的实例是SQL注入和格式字符串漏洞。

## CWE - 75 Failure to Sanitize Special Elements into a Different Plane (Special Element Injection)

中文：**未将特殊元素消毒到不同的平面（特殊元素注入）**

### Description

* 对于具有控制意义的特殊元素，该软件不能充分过滤用户控制的输入。

### Extended Description

* 没有详细描述

## CWE - 76 Improper Neutralization of Equivalent Special Elements

中文：**等效特殊元素的中和不当**

### Description

* 该软件适当地中和了某些特殊元素，但它不正确地中和了等效的特殊元素。

### Extended Description

* 该软件可能具有其认为完整的特殊字符的固定列表。但是，可能存在具有相同含义的替代编码或表示。例如，软件可以过滤掉前导斜杠（/）以防止绝对路径名称，但不会考虑后跟用户名的波浪号（〜），在某些\* nix系统上可以将其扩展为绝对路径名。或者，软件可能在调用外部程序时过滤危险的“-e”命令行开关，但它可能不会考虑“--exec”或具有相同语义的其他开关。

## CWE - 77 Improper Neutralization of Special Elements used in a Command ('Command Injection')

中文：**命令中使用的特殊元素的不正确中和（'命令注入'）**

### Description

* 该软件使用来自上游组件的外部影响输入构造全部或部分命令，但它不会中和或不正确地中和可能在将命令发送到下游组件时修改预期命令的特殊元素。

### Extended Description

* 命令注入漏洞通常在以下情况下发生：

1.数据从不受信任的来源进入应用程序。 2.数据是应用程序作为命令执行的字符串的一部分。 3.通过执行该命令，应用程序为攻击者提供了攻击者不具备的特权或能力。

命令注入是包装器程序的常见问题。

## CWE - 78 Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')

中文：**OS命令中使用的特殊元素的中和不正确（'OS命令注入'）**

### Description

* 该软件使用来自上游组件的外部影响输入构造全部或部分OS命令，但它不会中和或不正确地中和可能在将其发送到下游组件时修改预期OS命令的特殊元素。

### Extended Description

* 这可能允许攻击者直接在操作系统上执行意外的危险命令。这种弱点可能导致攻击者无法直接访问操作系统的环境中存在漏洞，例如在Web应用程序中。或者，如果弱点发生在特权程序中，则可能允许攻击者指定通常无法访问的命令，或者使用攻击者没有的特权来调用备用命令。如果受损的进程不遵循最小权限原则，则问题会更加严重，因为攻击者控制的命令可能会以特殊的系统权限运行，从而增加损害的数量。 OS命令注入至少有两种子类型：

该应用程序打算执行一个由其自己控制的单个固定程序。它打算使用外部提供的输入作为该程序的参数。例如，程序可能使用系统（“nslookup [HOSTNAME]”）来运行nslookup并允许用户提供HOSTNAME，该HOSTNAME用作参数。攻击者无法阻止nslookup执行。但是，如果程序没有从HOSTNAME参数中删除命令分隔符，则攻击者可以将分隔符放入参数中，这允许它们在nslookup完成执行后执行自己的程序。 应用程序接受一个输入，用于完全选择要运行的程序以及要使用的命令。应用程序只是将整个命令重定向到操作系统。例如，程序可能使用“exec（[COMMAND]）”来执行用户提供的[COMMAND]。如果COMMAND受攻击者控制，则攻击者可以执行任意命令或程序。如果使用exec（）和CreateProcess（）等函数执行命令，则攻击者可能无法在同一行中将多个命令组合在一起。

从弱点的角度来看，这些变体代表了不同的程序员错误。在第一个变体中，程序员明确地希望来自不可信方的输入将成为要执行的命令中的参数的一部分。在第二个变体中，程序员不打算让任何不受信任的方可以访问该命令，但程序员可能没有考虑恶意攻击者可以提供输入的替代方式。

## CWE - 79 Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')

中文：**网页生成期间输入的中和不正确（“跨站点脚本”）**

### Description

* 在将用户可控输入放入用作提供给其他用户的网页的输出之前，该软件不会中和或不正确地中和用户可控输入。

### Extended Description

* 跨站点脚本（XSS）漏洞发生在：

1.不受信任的数据通常来自Web请求进入Web应用程序。

1. Web应用程序动态生成包含此不受信任数据的网页。 3.在页面生成期间，应用程序不会阻止数据包含Web浏览器可执行的内容，例如JavaScript，HTML标记，HTML属性，鼠标事件，Flash，ActiveX等。 4.受害者通过Web浏览器访问生成的Web页面，该Web浏览器包含使用不受信任的数据注入的恶意脚本。 5.由于脚本来自Web服务器发送的网页，因此受害者的Web浏览器会在Web服务器的域中执行恶意脚本。 6.这实际上违反了Web浏览器的同源策略的意图，该策略指出一个域中的脚本不应该能够访问资源或在不同的域中运行代码。

XSS有三种主要类型：

类型1：反射的XSS（或非持久性） - 服务器直接从HTTP请求读取数据并将其反映在HTTP响应中。当攻击者导致受害者向易受攻击的Web应用程序提供危险内容时，会发生反射的XSS攻击，然后将其反射回受害者并由Web浏览器执行。传递恶意内容的最常见机制是将其作为参数包含在公开发布或直接通过电子邮件发送给受害者的URL中。以这种方式构建的URL构成了许多网络钓鱼方案的核心，攻击者诱使受害者访问引用易受攻击网站的URL。在站点将攻击者的内容反映回受害者之后，内容由受害者的浏览器执行。

类型2：存储的XSS（或持久） - 应用程序将危险数据存储在数据库，消息论坛，访问者日志或其他可信数据存储中。稍后，危险数据随后被读回应用程序并包含在动态内容中。从攻击者的角度来看，注入恶意内容的最佳位置是在向许多用户或特别有趣的用户显示的区域中。有趣的用户通常在应用程序中具有提升的权限，或者与对攻击者有价值的敏感数据进行交互。如果这些用户之一执行恶意内容，则攻击者可能能够代表用户执行特权操作或访问属于该用户的敏感数据。例如，攻击者可能会将XSS注入日志消息，当管理员查看日志时，可能无法正确处理。

类型0：基于DOM的XSS - 在基于DOM的XSS中，客户端执行将XSS注入页面;在其他类型中，服务器执行注入。基于DOM的XSS通常涉及发送到客户端的服务器控制的可信脚本，例如在用户提交表单之前对表单执行完整性检查的Javascript。如果服务器提供的脚本处理用户提供的数据，然后将其注入网页（例如使用动态HTML），则可以使用基于DOM的XSS。

一旦注入恶意脚本，攻击者就可以执行各种恶意活动。攻击者可以将受害者机器上的私人信息（例如可能包含会话信息的cookie）传输给攻击者。攻击者可以代表受害者向网站发送恶意请求，如果受害者具有管理该网站的管理员权限，则可能对该网站特别危险。网络钓鱼攻击可用于模拟受信任的网站，并欺骗受害者输入密码，允许攻击者破坏受害者在该网站上的帐户。最后，该脚本可以利用Web浏览器本身的漏洞，可能接管受害者的计算机，有时也称为“偷袭黑客”。 在许多情况下，攻击可以在受害者甚至没有意识到的情况下发起。即使有细心的用户，攻击者也经常使用各种方法来编码攻击的恶意部分，例如URL编码或Unicode，因此请求看起来不那么可疑。

## CWE - 80 Improper Neutralization of Script-Related HTML Tags in a Web Page (Basic XSS)

中文：**网页中与脚本相关的HTML标记的中和不当（基本XSS）**

### Description

* 该软件接收来自上游组件的输入，但它不会中和或不正确地中和特殊字符，例如“<”，“>”和“＆”，当它们被发送到下游组件时可被解释为Web脚本元素处理网页。

### Extended Description

* 这可以允许将这些字符视为控制字符，其在用户会话的上下文中在客户端执行。虽然这可以归类为注入问题，但更相关的问题是在将这些特殊字符显示给用户之前将这些特殊字符不正确地转换到相应的上下文相关实体。

## CWE - 81 Improper Neutralization of Script in an Error Message Web Page

中文：**错误消息网页中脚本的中和不正确**

### Description

* 该软件接收来自上游组件的输入，但它不会中和或不正确地中和可能在发送到错误页面时被解释为Web脚本元素的特殊字符。

### Extended Description

* 错误页面可能包括自定义的403 Forbidden或404 Not Found页面。 当攻击者可以触发包含未中和输入的错误时，可能会发生跨站点脚本攻击。

## CWE - 82 Improper Neutralization of Script in Attributes of IMG Tags in a Web Page

中文：**网页中IMG标记属性中脚本的中和不当**

### Description

* Web应用程序不会中和或错误地中和HTML IMG标记属性中的脚本元素，例如src属性。

### Extended Description

* 攻击者可以将XSS攻击嵌入到流式传输然后在受害者浏览器中执行的IMG属性（例如SRC）的值中。请注意，当页面加载到用户的浏览器中时，漏洞会自动执行。

## CWE - 83 Improper Neutralization of Script in Attributes in a Web Page

中文：**网页属性中脚本的中和不正确**

### Description

* 该软件不会中和或错误地中和“javascript：”或来自标签内危险属性的其他URI，例如onmouseover，onload，onerror或style。

### Extended Description

* 没有详细描述

## CWE - 84 Improper Neutralization of Encoded URI Schemes in a Web Page

中文：**网页中编码URI方案的中和不正确**

### Description

* Web应用程序不正确地中和了用URI编码伪装的可执行脚本的用户控制输入。

### Extended Description

* 没有详细描述

## CWE - 85 Doubled Character XSS Manipulations

中文：**双字符XSS操作**

### Description

* Web应用程序不会过滤用户控制的输入，因为可执行脚本使用相关字符加倍来伪装。

### Extended Description

* 没有详细描述

## CWE - 86 Improper Neutralization of Invalid Characters in Identifiers in Web Pages

中文：**网页中标识符中无效字符的中和不正确**

### Description

* 该软件不会中和或错误地中和标记名称，URI方案和其他标识符中间的无效字符或字节序列。

### Extended Description

* 某些Web浏览器可能会删除这些序列，从而导致输出可能具有意外的控制含义。例如，软件可能会尝试删除“javascript：”URI方案，但“java％00script：”URI可能会绕过此检查，并且仍会被某些浏览器呈现为活动javascript，从而允许XSS或其他攻击。

## CWE - 87 Improper Neutralization of Alternate XSS Syntax

中文：**备用XSS语法的中和不正确**

### Description

* 该软件不会中和或错误地中和用户控制的输入以获得备用脚本语法。

### Extended Description

* 没有详细描述

## CWE - 88 Argument Injection or Modification

中文：**参数注入或修改**

### Description

* 该软件没有充分划分传递给另一个控件领域中的组件的参数，允许提供备用参数，从而导致潜在的安全相关更改。

### Extended Description

* 没有详细描述

## CWE - 89 Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')

中文：**SQL命令中使用的特殊元素的中和不正确（'SQL注入'）**

### Description

* 该软件使用来自上游组件的外部影响输入构造全部或部分SQL命令，但它不会中和或不正确地中和可能在发送到下游组件时修改预期SQL命令的特殊元素。

### Extended Description

* 如果没有在用户可控输入中充分删除或引用SQL语法，则生成的SQL查询可以将这些输入解释为SQL而不是普通用户数据。这可用于更改查询逻辑以绕过安全检查，或插入修改后端数据库的其他语句，可能包括执行系统命令。 SQL注入已经成为数据库驱动的网站的常见问题。该漏洞很容易被检测到，并且容易被利用，因此，任何具有最小用户群的站点或软件包都可能遭受此类攻击。这个缺陷取决于SQL在控制平面和数据平面之间没有真正区别的事实。

## CWE - 90 Improper Neutralization of Special Elements used in an LDAP Query ('LDAP Injection')

中文：**LDAP查询中使用的特殊元素的中和不正确（'LDAP注入'）**

### Description

* 该软件使用来自上游组件的外部影响输入构造全部或部分LDAP查询，但它不会中和或不正确地中和可能在将其发送到下游组件时修改预期LDAP查询的特殊元素。

### Extended Description

* 没有详细描述

## CWE - 91 XML Injection (aka Blind XPath Injection)

中文：**XML注入（又称盲注XPath注入）**

### Description

* 该软件无法正确中和XML中使用的特殊元素，允许攻击者在终端系统处理之前修改XML的语法，内容或命令。

### Extended Description

* 在XML中，特殊元素可以包括保留字或字符，例如“<”，“>”，“”和“＆”，然后可以用于添加新数据或修改XML语法。

## CWE - 92 DEPRECATED: Improper Sanitization of Custom Special Characters

中文：**弃用：自定义特殊字符的不当消毒**

### Description

* 此条目已被弃用。它最初来自PLOVER，它有时定义“其他”和“杂项”类别，以满足分类法的详尽性要求。在CWE的上下文中，在映射情况下优选使用更抽象的条目。 CWE-75是更合适的映射。

### Extended Description

* 没有详细描述

## CWE - 93 Improper Neutralization of CRLF Sequences ('CRLF Injection')

中文：**CRLF序列的中和不当（'CRLF注射'）**

### Description

* 该软件使用CRLF（回车换行）作为特殊元素，例如，分隔行或记录，但它不会中和或错误地中和输入中的CRLF序列。

### Extended Description

* 没有详细描述

## CWE - 94 Improper Control of Generation of Code ('Code Injection')

中文：**代码生成控制不当（'代码注入'）**

### Description

* 该软件使用来自上游组件的外部影响输入构造全部或部分代码段，但它不会中和或不正确地中和可能修改预期代码段的语法或行为的特殊元素。

### Extended Description

* 当软件允许用户的输入包含代码语法时，攻击者可能以这样的方式制作代码，使其改变软件的预期控制流程。这种改变可能导致任意代码执行。 注射问题包含各种各样的问题 - 所有问题都以非常不同的方式得到缓解。因此，讨论这些弱点的最有效方法是注意将它们归类为注入弱点的独特特征。需要注意的最重要的问题是所有注入问题都有一个共同点 - 即，它们允许将控制平面数据注入用户控制的数据平面。这意味着可以通过合法数据通道发送代码来改变进程的执行，而不使用其他机制。虽然缓冲区溢出和许多其他缺陷涉及使用一些进一步的问题来获得执行，但注入问题仅需要解析数据。这类弱点的最经典的实例是SQL注入和格式字符串漏洞。

## CWE - 95 Improper Neutralization of Directives in Dynamically Evaluated Code ('Eval Injection')

中文：**动态评估代码中指令的不正确中和（'Eval Injection'）**

### Description

* 软件接收来自上游组件的输入，但在动态评估调用（例如“eval”）中使用输入之前，它不会中和或不正确地中和代码语法。

### Extended Description

* 这可能允许攻击者执行任意代码，或者至少修改可以执行的代码。

## CWE - 96 Improper Neutralization of Directives in Statically Saved Code ('Static Code Injection')

中文：**静态保存代码中指令的不正确中和（'静态代码注入'）**

### Description

* 该软件接收来自上游组件的输入，但在将输入插入可执行资源（例如库，配置文件或模板）之前，它不会中和或不正确地中和代码语法。

### Extended Description

* 没有详细描述

## CWE - 97 Improper Neutralization of Server-Side Includes (SSI) Within a Web Page

中文：**网页中服务器端包含（SSI）的不正确中和**

### Description

* 该软件生成一个网页，但不会中和或不正确地中和可被解释为服务器端包含（SSI）指令的用户可控输入。

### Extended Description

* 没有详细描述

## CWE - 98 Improper Control of Filename for Include/Require Statement in PHP Program ('PHP Remote File Inclusion')

中文：**PHP程序中Include / Require语句的文件名控制不当（'PHP远程文件包含'）**

### Description

* PHP应用程序从上游组件接收输入，但在“require”，“include”或类似函数中使用之前，它不会限制或错误地限制输入。

### Extended Description

* 在PHP的某些版本和配置中，这可以允许攻击者指定远程位置的URL，软件将从该位置获取要执行的代码。在与路径遍历相关联的其他情况下，攻击者可以指定可能包含可由PHP解析的可执行语句的本地文件。

## CWE - 99 Improper Control of Resource Identifiers ('Resource Injection')

中文：**资源标识符控制不当（'资源注入'）**

### Description

* 软件接收来自上游组件的输入，但在将输入用作可能在预期控制范围之外的资源的标识符之前，它不限制或不正确地限制输入。

### Extended Description

* 满足以下两个条件时会发生资源注入问题：

攻击者可以指定用于访问系统资源的标识符。例如，攻击者可能能够指定要打开的文件的名称的一部分或要使用的端口号。 通过指定资源，攻击者获得了不允许的功能。例如，程序可能使攻击者能够覆盖指定的文件，使用受攻击者控制的配置运行，或者将敏感信息传输到第三方服务器。

这可以使攻击者能够访问或修改受保护的系统资源。

## CWE - 102 Struts: Duplicate Validation Forms

中文：**Struts：重复的验证表单**

### Description

* 应用程序使用多个具有相同名称的验证表单，这可能会导致Struts验证程序验证程序员不期望的表单。

### Extended Description

* 如果两个验证表单具有相同的名称，Struts Validator会任意选择其中一个表单用于输入验证并丢弃另一个表单。这个决定可能与程序员的期望不符，可能导致最终的弱点。此外，它表明验证逻辑不是最新的，并且可以指示存在其他更微妙的验证错误。

## CWE - 103 Struts: Incomplete validate() Method Definition

中文：**Struts：不完整的validate（）方法定义**

### Description

* 应用程序具有验证器表单，该表单既不定义validate（）方法，也不定义validate（）方法但不调用super.validate（）。

### Extended Description

* 如果不调用super.validate（），则验证框架无法根据验证表单检查表单的内容。换句话说，将针对给定表单禁用验证框架。

## CWE - 104 Struts: Form Bean Does Not Extend Validation Class

中文：**Struts：Form Bean不会扩展验证类**

### Description

* 如果表单bean没有扩展Validator框架的ActionForm子类，它可以将应用程序暴露给与输入验证不足相关的其他弱点。

### Extended Description

* 没有详细描述

## CWE - 105 Struts: Form Field Without Validator

中文：**Struts：没有验证器的表单字段**

### Description

* 应用程序的表单字段未通过相应的验证表单进行验证，这可能会引入与输入验证不足相关的其他弱点。

### Extended Description

* 即使是单个输入字段省略验证也可能为攻击者提供破坏应用程序所需的余地。尽管J2EE应用程序通常不容易受到内存损坏攻击，但如果J2EE应用程序与不执行数组边界检查的本机代码接口，则攻击者可能能够在J2EE应用程序中使用输入验证错误来启动缓冲区溢出攻击。

## CWE - 106 Struts: Plug-in Framework not in Use

中文：**Struts：未使用的插件框架**

### Description

* 当应用程序不使用输入验证框架（如Struts Validator）时，引入与输入验证不足相关的弱点的风险更大。

### Extended Description

* 未经检查的输入是J2EE应用程序中漏洞的主要原因。未经检查的输入会导致跨站点脚本，进程控制和SQL注入漏洞等。 尽管J2EE应用程序通常不容易受到内存损坏攻击，但如果J2EE应用程序与不执行数组边界检查的本机代码接口，则攻击者可能能够在J2EE应用程序中使用输入验证错误来启动缓冲区溢出攻击。

## CWE - 107 Struts: Unused Validation Form

中文：**Struts：未使用的验证表单**

### Description

* 未使用的验证表单表明验证逻辑不是最新的。

### Extended Description

* 当开发人员删除或重命名动作表单映射时，很容易忘记更新验证逻辑。验证逻辑未得到正确维护的一个迹象是存在未使用的验证表单。

## CWE - 108 Struts: Unvalidated Action Form

中文：**Struts：未经验证的行动表格**

### Description

* 每个行动表格都必须有相应的验证表格。

### Extended Description

* 如果Struts Action Form Mapping指定了一个表单，它必须在Struts Validator下定义一个验证表单。

## CWE - 109 Struts: Validator Turned Off

中文：**Struts：验证器关闭**

### Description

* 已关闭通过Struts bean自动筛选，这会禁用Struts Validator和自定义验证逻辑。这使应用程序暴露于与输入验证不足相关的其他弱点。

### Extended Description

* 没有详细描述

## CWE - 110 Struts: Validator Without Form Field

中文：**Struts：没有表单字段的验证器**

### Description

* 未出现在与其关联的表单中的验证字段表示验证逻辑已过期。

### Extended Description

* 开发人员在更改ActionForm类时很容易忘记更新验证逻辑。验证逻辑未得到正确维护的一个迹象是操作表单和验证表单之间的不一致。 尽管J2EE应用程序通常不容易受到内存损坏攻击，但如果J2EE应用程序与不执行数组边界检查的本机代码接口，则攻击者可能能够在J2EE应用程序中使用输入验证错误来启动缓冲区溢出攻击。

## CWE - 111 Direct Use of Unsafe JNI

中文：**直接使用不安全的JNI**

### Description

* 当Java应用程序使用Java本机接口（JNI）来调用用另一种编程语言编写的代码时，它可以将应用程序暴露给该代码中的弱点，即使这些弱点不能在Java中出现。

### Extended Description

* 程序员可能认为理所当然的许多安全功能根本不适用于本机代码，因此您必须仔细检查所有此类代码是否存在潜在问题。用于实现本机代码的语言可能更容易受到缓冲区溢出和其他攻击的影响。本机代码不受运行时环境强制执行的安全功能的保护，例如强类型和数组边界检查。

## CWE - 112 Missing XML Validation

中文：**缺少XML验证**

### Description

* 该软件接受来自不受信任来源的XML，但不会根据正确的模式验证XML。

### Extended Description

* 大多数成功的攻击始于违反程序员的假设。通过接受XML文档而不针对DTD或XML模式验证它，程序员为攻击者留下了一扇门，以便提供意外，不合理或恶意的输入。

## CWE - 113 Improper Neutralization of CRLF Sequences in HTTP Headers ('HTTP Response Splitting')

中文：**HTTP标头中CRLF序列的不正确中和（'HTTP响应拆分'）**

### Description

* 软件从上游组件接收数据，但在数据包含在传出HTTP标头中之前，不会中和或错误地中和CR和LF字符。

### Extended Description

* 在HTTP标头中包含未经验证的数据允许攻击者指定浏览器呈现的整个HTTP响应。当HTTP请求包含意外CR（回车，也由％0d或\ r \ n）和LF（换行，也由％0a或\ n给出）字符时，服务器可能会响应一个被解释为两个不同的输出流HTTP响应（而不是一个）。攻击者可以控制第二个响应并加载攻击，例如跨站点脚本和缓存中毒攻击。 在以下情况下可能存在HTTP响应拆分弱点：

数据通过不受信任的来源进入Web应用程序，最常见的是HTTP请求。 数据包含在发送给Web用户的HTTP响应标头中，而不会对恶意字符进行验证。

## CWE - 114 Process Control

中文：**过程控制**

### Description

* 从不受信任的源或不受信任的环境执行命令或加载库可能导致应用程序代表攻击者执行恶意命令（和有效负载）。

### Extended Description

* 进程控制漏洞有两种形式：1。攻击者可以更改程序执行的命令：攻击者明确控制命令的内容。 2.攻击者可以更改命令执行的环境：攻击者隐式控制命令的含义。当数据从不受信任的源进入应用程序并且数据用作表示应用程序执行的命令的字符串的一部分时，会发生第一种类型的进程控制漏洞。通过执行该命令，应用程序为攻击者提供了攻击者无法拥有的特权或能力。

## CWE - 115 Misinterpretation of Input

中文：**对输入的误解**

### Description

* 该软件以安全相关的方式错误解释了来自攻击者或其他产品的输入。

### Extended Description

* 没有详细描述

## CWE - 116 Improper Encoding or Escaping of Output

中文：**输出的编码或转义不正确**

### Description

* 该软件准备用于与另一个组件通信的结构化消息，但是数据的编码或转义要么丢失要么不正确。结果，不保留消息的预期结构。

### Extended Description

* 不正确的编码或转义可能允许攻击者更改发送到另一个组件的命令，而是插入恶意命令。 大多数软件遵循某种协议，该协议使用结构化消息进行组件之间的通信，例如查询或命令。这些结构化消息可以包含散布有元数据或控制信息的原始数据。例如，“GET /index.html HTTP / 1.1”是一个结构化消息，其中包含一个带有单个参数（“/index.html”）的命令（“GET”）和有关正在使用哪个协议版本的元数据（“HTTP / 1.1" ）。 如果应用程序使用攻击者提供的输入来构造结构化消息而没有正确编码或转义，则攻击者可以插入特殊字符，这些特征将导致数据被解释为控制信息或元数据。因此，接收输出的组件将执行错误的操作，或以其他方式错误地解释数据。

## CWE - 117 Improper Output Neutralization for Logs

中文：**日志的输出中和不正确**

### Description

* 该软件不会中和或错误地中和写入日志的输出。

### Extended Description

* 这可能允许攻击者伪造日志条目或将恶意内容注入日志。 在以下情况下发生日志伪造漏洞：

数据从不受信任的来源进入应用程序。 数据将写入应用程序或系统日志文件。

## CWE - 118 Incorrect Access of Indexable Resource ('Range Error')

中文：**可索引资源的不正确访问（“范围错误”）**

### Description

* 该软件不限制或不正确地限制使用索引或指针（例如内存或文件）访问的资源边界内的操作。

### Extended Description

* 没有详细描述

## CWE - 119 Improper Restriction of Operations within the Bounds of a Memory Buffer

中文：**内存缓冲区内的操作限制不当**

### Description

* 软件对内存缓冲区执行操作，但它可以读取或写入缓冲区预期边界之外的内存位置。

### Extended Description

* 某些语言允许直接寻址内存位置，并且不会自动确保这些位置对正在引用的内存缓冲区有效。这可以导致对可能与其他变量，数据结构或内部程序数据相关联的存储器位置执行读或写操作。 因此，攻击者可能能够执行任意代码，更改预期的控制流，读取敏感信息或导致系统崩溃。

## CWE - 120 Buffer Copy without Checking Size of Input ('Classic Buffer Overflow')

中文：**缓冲区复制而不检查输入大小（'经典缓冲区溢出'）**

### Description

* 程序将输入缓冲区复制到输出缓冲区，而不验证输入缓冲区的大小是否小于输出缓冲区的大小，从而导致缓冲区溢出。

### Extended Description

* 当程序试图将更多数据放入缓冲区而不是它可以容纳时，或者当程序试图将数据放入缓冲区边界之外的内存区域时，就会出现缓冲区溢出情况。最简单的错误类型和缓冲区溢出的最常见原因是“经典”情况，程序复制缓冲区而不限制复制的数量。存在其他变体，但经典溢出的存在强烈暗示程序员甚至不考虑最基本的安全保护。

## CWE - 121 Stack-based Buffer Overflow

中文：**基于堆栈的缓冲区溢出**

### Description

* 基于堆栈的缓冲区溢出条件是被覆盖的缓冲区被分配在堆栈上的条件（即，是局部变量，或者很少是函数的参数）。

### Extended Description

* 没有详细描述

## CWE - 122 Heap-based Buffer Overflow

中文：**基于堆的缓冲区溢出**

### Description

* 堆溢出条件是缓冲区溢出，其中可以覆盖的缓冲区在内存的堆部分中分配，通常意味着缓冲区是使用诸如malloc（）之类的例程分配的。

### Extended Description

* 没有详细描述

## CWE - 123 Write-what-where Condition

中文：**写什么地方条件**

### Description

* 攻击者能够将任意值写入任意位置的任何情况，通常是缓冲区溢出的结果。

### Extended Description

* 没有详细描述

## CWE - 124 Buffer Underwrite ('Buffer Underflow')

中文：**缓冲区保护（'缓冲下溢'）**

### Description

* 软件使用索引或指针写入缓冲区，该索引或指针在缓冲区开始之前引用内存位置。

### Extended Description

* 这通常发生在指针或其索引递减到缓冲区之前的位置，指针算术导致有效内存位置开始之前的位置或使用负索引时。

## CWE - 125 Out-of-bounds Read

中文：**越界阅读**

### Description

* 软件在预期缓冲区的结束或开始之前读取数据。

### Extended Description

* 通常，这可能允许攻击者从其他内存位置读取敏感信息或导致崩溃。当代码读取可变数量的数据并假定存在用于停止读取操作的标记（例如字符串中的NUL）时，可能会发生崩溃。预期的哨兵可能不会位于越界内存中，导致过多的数据被读取，从而导致分段错误或缓冲区溢出。软件可以修改索引或执行引用超出缓冲区边界的存储器位置的指针算术。随后的读操作会产生未定义或意外的结果。

## CWE - 126 Buffer Over-read

中文：**缓冲区过度读取**

### Description

* 软件使用缓冲区访问机制从缓冲区读取，例如在目标缓冲区之后引用内存位置的索引或指针。

### Extended Description

* 这通常发生在指针或其索引递增到超出缓冲区边界的位置时，或者当指针算法导致有效内存位置之外的位置时，这一点通常会发生。这可能导致敏感信息的暴露或可能导致崩溃。

## CWE - 127 Buffer Under-read

中文：**缓冲区欠读**

### Description

* 软件使用缓冲区访问机制从缓冲区读取，例如在目标缓冲区之前引用内存位置的索引或指针。

### Extended Description

* 这通常发生在指针或其索引递减到缓冲区之前的位置，指针算术导致有效内存位置开始之前的位置，或者使用负索引时。这可能导致敏感信息的暴露或可能导致崩溃。

## CWE - 128 Wrap-around Error

中文：**环绕错误**

### Description

* 只要值增加超过其类型的最大值就会发生错误，因此会“绕回”到非常小的，负的或未定义的值。

### Extended Description

* 没有详细描述

## CWE - 129 Improper Validation of Array Index

中文：**数组索引的不正确验证**

### Description

* 在计算或使用数组索引时，产品使用不受信任的输入，但产品不验证或错误地验证索引以确保索引引用数组中的有效位置。

### Extended Description

* 没有详细描述

## CWE - 130 Improper Handling of Length Parameter Inconsistency

中文：**长度参数不一致的处理不当**

### Description

* 该软件解析格式化的消息或结构，但它不处理或错误地处理与关联数据的实际长度不一致的长度字段。

### Extended Description

* 如果攻击者可以操纵与输入相关联的长度参数，使其与输入的实际长度不一致，则可以利用这一点来使目标应用程序以意外的，可能是恶意的方式运行。这样做的一个可能动机是将任意大量的输入传递给应用程序。另一个可能的动机是通过为应用程序的后续属性包含无效数据来修改应用程序状态。这些弱点通常会导致诸如缓冲区溢出和任意代码执行之类的攻击。

## CWE - 131 Incorrect Calculation of Buffer Size

中文：**缓冲区大小的计算不正确**

### Description

* 软件无法正确计算分配缓冲区时要使用的大小，这可能导致缓冲区溢出。

### Extended Description

* 没有详细描述

## CWE - 132 DEPRECATED (Duplicate): Miscalculated Null Termination

中文：**弃用（重复）：计算的空终止**

### Description

* 此条目已被弃用，因为它与CWE-170重复。所有内容均已转移至CWE-170。

### Extended Description

* 没有详细描述

## CWE - 134 Use of Externally-Controlled Format String

中文：**使用外部控制的格式字符串**

### Description

* 该软件使用一个接受格式字符串作为参数的函数，但格式字符串来自外部源。

### Extended Description

* 当攻击者可以修改外部控制的格式字符串时，这可能导致缓冲区溢出，拒绝服务或数据表示问题。 应该注意的是，在某些情况下，例如国际化，格式字符串集是由设计外部控制的。如果这些格式字符串的来源是可信的（例如，仅包含在只能由系统管理员修改的库文件中），则外部控件本身可能不会构成漏洞。

## CWE - 135 Incorrect Calculation of Multi-Byte String Length

中文：**多字节字符串长度的计算不正确**

### Description

* 该软件无法正确计算可包含宽字节或多字节字符的字符串长度。

### Extended Description

* 没有详细描述

## CWE - 138 Improper Neutralization of Special Elements

中文：**特殊元素的中和不当**

### Description

* 该软件接收来自上游组件的输入，但它不会中和或不正确地中和特殊元素，这些元素在被发送到下游组件时可被解释为控制元素或语法标记。

### Extended Description

* 大多数语言和协议都有自己的特殊元素，如字符和保留字。这些特殊元素可以带来控制意义。如果软件不阻止外部控制或影响包含这些特殊元素，则程序的控制流程可能会改变。例如，Unix和Windows都将符号<（“小于”）解释为“从文件读取输入”。

## CWE - 140 Improper Neutralization of Delimiters

中文：**分隔符的中和不正确**

### Description

* 该软件不会中和或错误地中和分隔符。

### Extended Description

* 没有详细描述

## CWE - 141 Improper Neutralization of Parameter/Argument Delimiters

中文：**参数/参数分隔符的中和不正确**

### Description

* 软件接收来自上游组件的输入，但它不会中和或不正确地中和可能在发送到下游组件时被解释为参数或参数分隔符的特殊元素。

### Extended Description

* 在解析数据时，注入/缺失/格式错误的分隔符可能会导致进程执行意外操作。

## CWE - 142 Improper Neutralization of Value Delimiters

中文：**价值分隔符的中和不当**

### Description

* 该软件接收来自上游组件的输入，但它不会中和或不正确地中和在将它们发送到下游组件时可被解释为值分隔符的特殊元素。

### Extended Description

* 在解析数据时，注入/缺失/格式错误的分隔符可能会导致进程执行意外操作。

## CWE - 143 Improper Neutralization of Record Delimiters

中文：**记录分隔符的中和不正确**

### Description

* 该软件接收来自上游组件的输入，但它不会中和或不正确地中和在将它们发送到下游组件时可被解释为记录分隔符的特殊元素。

### Extended Description

* 在解析数据时，注入/缺失/格式错误的分隔符可能会导致进程执行意外操作。

## CWE - 144 Improper Neutralization of Line Delimiters

中文：**线分隔符的中和不正确**

### Description

* 该软件接收来自上游组件的输入，但它不会中和或不正确地中和在将它们发送到下游组件时可被解释为行分隔符的特殊元素。

### Extended Description

* 在解析数据时，注入/缺失/格式错误的分隔符可能会导致进程执行意外操作。

## CWE - 145 Improper Neutralization of Section Delimiters

中文：**截面分隔符的不正确中和**

### Description

* 该软件接收来自上游组件的输入，但它不会中和或不正确地中和在将它们发送到下游组件时可被解释为段分隔符的特殊元素。

### Extended Description

* 在解析数据时，注入/缺失/格式错误的分隔符可能会导致进程执行意外操作。 区段分隔符的一个示例是多部分MIME消息中的边界字符串。在许多情况下，双线分隔符可以用作节分隔符。

## CWE - 146 Improper Neutralization of Expression/Command Delimiters

中文：**表达式/命令分隔符的中和不正确**

### Description

* 该软件接收来自上游组件的输入，但它不会中和或不正确地中和在将它们发送到下游组件时可被解释为表达式或命令分隔符的特殊元素。

### Extended Description

* 在解析数据时，注入/缺失/格式错误的分隔符可能会导致进程执行意外操作。

## CWE - 147 Improper Neutralization of Input Terminators

中文：**输入终结器的中和不正确**

### Description

* 该软件接收来自上游组件的输入，但它不会中和或不正确地中和在将它们发送到下游组件时可被解释为输入终止符的特殊元素。

### Extended Description

* 例如，“。”在SMTP中表示邮件消息数据的结束，而空字符可用于字符串的结尾。

## CWE - 148 Improper Neutralization of Input Leaders

中文：**输入领导者的中和不当**

### Description

* 当主要字符或序列（“领导者”）丢失或格式错误时，或者当只允许一个领导者使用多个领导者时，应用程序无法正确处理。

### Extended Description

* 没有详细描述

## CWE - 149 Improper Neutralization of Quoting Syntax

中文：**引用语法的中和不正确**

### Description

* 注入应用程序的引号可用于危害系统。在解析数据时，引用的注入/不存在/重复/格式错误的使用可能导致进程采取意外操作。

### Extended Description

* 没有详细描述

## CWE - 150 Improper Neutralization of Escape, Meta, or Control Sequences

中文：**转义，元或控制序列的中和不正确**

### Description

* 该软件接收来自上游组件的输入，但它不会中和或不正确地中和可能被解释为转义，元或控制字符序列的特殊元素，当它们被发送到下游组件时。

### Extended Description

* 在解析数据时，注入/缺失/格式错误的分隔符可能会导致进程执行意外操作。

## CWE - 151 Improper Neutralization of Comment Delimiters

中文：**评论分隔符的不正当中和**

### Description

* 软件接收来自上游组件的输入，但它不会中和或错误地中和特殊元素，这些元素在被发送到下游组件时可以被解释为注释分隔符。

### Extended Description

* 没有详细描述

## CWE - 152 Improper Neutralization of Macro Symbols

中文：**宏符号的中和不正确**

### Description

* 软件接收来自上游组件的输入，但它不会中和或不正确地中和可能在被发送到下游组件时被解释为宏符号的特殊元素。

### Extended Description

* 没有详细描述

## CWE - 153 Improper Neutralization of Substitution Characters

中文：**替换字符的中和不当**

### Description

* 该软件接收来自上游组件的输入，但它不会中和或不正确地中和在将它们发送到下游组件时可被解释为替换字符的特殊元素。

### Extended Description

* 没有详细描述

## CWE - 154 Improper Neutralization of Variable Name Delimiters

中文：**变量名称分隔符的不正确中和**

### Description

* 该软件接收来自上游组件的输入，但它不会中和或不正确地中和在将它们发送到下游组件时可被解释为变量名称分隔符的特殊元素。

### Extended Description

* 在解析数据时，注入的分隔符可能会导致进程采取导致攻击的意外操作。示例：“$”表示环境变量。

## CWE - 155 Improper Neutralization of Wildcards or Matching Symbols

中文：**对通配符或匹配符号的不正确中和**

### Description

* 软件接收来自上游组件的输入，但它不会中和或不正确地中和可能被解释为通配符或匹配符号的特殊元素，当它们被发送到下游组件时。

### Extended Description

* 在解析数据时，注入的元素可能会导致进程执行意外操作。

## CWE - 156 Improper Neutralization of Whitespace

中文：**空白中和不当**

### Description

* 该软件接收来自上游组件的输入，但它不会中和或不正确地中和在将它们发送到下游组件时可被解释为空白的特殊元素。

### Extended Description

* 这可以包括空格，标签等。

## CWE - 157 Failure to Sanitize Paired Delimiters

中文：**未能消除配对分隔符**

### Description

* 该软件无法正确处理用于标记一组实体的开头和结尾的字符，例如括号，括号和大括号。

### Extended Description

* 配对分隔符可能包括：

<和>尖括号 （和）括号 {和}大括号 [和]方括号 “ “ 双引号 ''单引号

## CWE - 158 Improper Neutralization of Null Byte or NUL Character

中文：**空字节或NUL字符的中和不正确**

### Description

* 软件接收来自上游组件的输入，但是当它们被发送到下游组件时，它不会中和或不正确地中和NUL字符或空字节。

### Extended Description

* 在解析数据时，注入的NUL字符或空字节可能导致软件认为输入比实际更早地终止，或者导致输入被误解。然后，这可以用于注入在空字节之后发生的潜在危险输入，或者绕过验证例程和其他保护机制。

## CWE - 159 Failure to Sanitize Special Element

中文：**未能消毒特殊元素**

### Description

* 这种以攻击为重的类别中的弱点无法正确过滤和解释用户控制输入中的特殊元素，这些元素可能会对软件行为和完整性产生负面影响。

### Extended Description

* 没有详细描述

## CWE - 160 Improper Neutralization of Leading Special Elements

中文：**领导特殊元素的不正当中和**

### Description

* 该软件接收来自上游组件的输入，但它不会中和或不正确地中和可能在发送到下游组件时以意外方式解释的前导特殊元素。

### Extended Description

* 在解析数据时，不正确处理前导特殊元素可能会导致进程采取导致攻击的意外操作。

## CWE - 161 Improper Neutralization of Multiple Leading Special Elements

中文：**多种主要特殊元素的中和不当**

### Description

* 该软件接收来自上游组件的输入，但它不会中和或不正确地中和多个可能在发送到下游组件时以意外方式解释的前导特殊元素。

### Extended Description

* 在解析数据时，不正确地处理多个前导特殊元素可能会导致进程采取导致攻击的意外操作。

## CWE - 162 Improper Neutralization of Trailing Special Elements

中文：**尾随特殊元素的中和不当**

### Description

* 该软件接收来自上游组件的输入，但它不会中和或不正确地中和可能在发送到下游组件时以意外方式解释的尾随特殊元素。

### Extended Description

* 在解析数据时，不正确处理尾随特殊元素可能会导致进程采取导致攻击的意外操作。

## CWE - 163 Improper Neutralization of Multiple Trailing Special Elements

中文：**多尾随特殊元素的中和不当**

### Description

* 该软件接收来自上游组件的输入，但它不会中和或不正确地中和多个尾随特殊元素，这些特殊元素在被发送到下游组件时可能以意外方式解释。

### Extended Description

* 在解析数据时，不正确地处理多个尾随特殊元素可能会导致进程采取导致攻击的意外操作。

## CWE - 164 Improper Neutralization of Internal Special Elements

中文：**内部特殊元素的中和不当**

### Description

* 该软件接收来自上游组件的输入，但它不会中和或不正确地中和内部特殊元素，这些特殊元素在被发送到下游组件时可能以意外方式解释。

### Extended Description

* 在解析数据时，不正确地处理内部特殊元素可能会导致进程采取导致攻击的意外操作。

## CWE - 165 Improper Neutralization of Multiple Internal Special Elements

中文：**多个内部特殊元素的中和不当**

### Description

* 该软件接收来自上游组件的输入，但它不会中和或不正确地中和多个内部特殊元素，这些特殊元素在被发送到下游组件时可能以意外方式解释。

### Extended Description

* 在解析数据时，不正确地处理多个内部特殊元素可能会导致进程采取导致攻击的意外操作。

## CWE - 166 Improper Handling of Missing Special Element

中文：**丢失特殊元素的处理不当**

### Description

* 该软件接收来自上游组件的输入，但是当缺少预期的特殊元素时，它不会处理或错误处理。

### Extended Description

* 没有详细描述

## CWE - 167 Improper Handling of Additional Special Element

中文：**附加特殊元素的处理不当**

### Description

* 该软件接收来自上游组件的输入，但是当缺少其他意外特殊元素时，它不会处理或错误处理。

### Extended Description

* 没有详细描述

## CWE - 168 Improper Handling of Inconsistent Special Elements

中文：**对不一致的特殊元素的处理不当**

### Description

* 当两个或多个特殊字符或保留字之间存在不一致时，软件不会处理。

### Extended Description

* 此问题的一个示例是，如果配对字符以错误的顺序出现，或者特殊字符未正确嵌套。

## CWE - 170 Improper Null Termination

中文：**不适当的空终止**

### Description

* 该软件不会终止或错误地终止具有空字符或等效终结符的字符串或数组。

### Extended Description

* 空终止错误经常以两种不同的方式发生。逐个错误可能导致空值被写入超出边界，从而导致溢出。或者，程序可能会错误地使用strncpy（）函数调用，这会阻止添加空终止符。其他情况也是可能的。

## CWE - 172 Encoding Error

中文：**编码错误**

### Description

* 该软件无法正确编码或解码数据，从而导致意外值。

### Extended Description

* 没有详细描述

## CWE - 173 Improper Handling of Alternate Encoding

中文：**替代编码的处理不当**

### Description

* 当输入使用对发送输入的控制范围有效的备用编码时，软件无法正确处理。

### Extended Description

* 没有详细描述

## CWE - 174 Double Decoding of the Same Data

中文：**双重解码相同的数据**

### Description

* 该软件对相同的输入进行两次解码，这可能限制在解码操作之间发生的任何保护机制的有效性。

### Extended Description

* 没有详细描述

## CWE - 175 Improper Handling of Mixed Encoding

中文：**混合编码的处理不当**

### Description

* 当相同的输入使用多种不同（混合）编码时，软件无法正确处理。

### Extended Description

* 没有详细描述

## CWE - 176 Improper Handling of Unicode Encoding

中文：**Unicode编码处理不当**

### Description

* 当输入包含Unicode编码时，软件无法正确处理。

### Extended Description

* 没有详细描述

## CWE - 177 Improper Handling of URL Encoding (Hex Encoding)

中文：**URL编码处理不当（十六进制编码）**

### Description

* 当全部或部分输入已经过URL编码时，软件无法正确处理。

### Extended Description

* 没有详细描述

## CWE - 178 Improper Handling of Case Sensitivity

中文：**案例敏感性处理不当**

### Description

* 在访问或确定资源属性时，软件无法正确解释区分大小写的差异，从而导致结果不一致。

### Extended Description

* 处理不当的区分大小写的数据可能会导致一些可能的后果，包括：

不区分大小写的密码减少了密钥空间的大小，使得暴力攻击更容易 使用备用名称绕过过滤器或访问控制 使用备用名称的多个解释错误。

## CWE - 179 Incorrect Behavior Order: Early Validation

中文：**行为顺序不正确：早期验证**

### Description

* 软件在应用修改输入的保护机制之前验证输入，这可能允许攻击者通过仅在修改后出现的危险输入绕过验证。

### Extended Description

* 在数据经过规范化和清理后，软件需要在适当的时间验证数据。早期验证易受各种操作的影响，这些操作会导致标准化和清洁产生的危险输入。

## CWE - 180 Incorrect Behavior Order: Validate Before Canonicalize

中文：**行为顺序不正确：在Canonicalize之前验证**

### Description

* 软件在规范化之前验证输入，这阻止软件检测在规范化步骤之后变为无效的数据。

### Extended Description

* 攻击者可以使用此方法绕过验证并发起攻击，从而暴露可能会被阻止的弱点，例如注入。

## CWE - 181 Incorrect Behavior Order: Validate Before Filter

中文：**行为顺序不正确：在过滤前验证**

### Description

* 软件在过滤之前验证数据，这可以防止软件检测到过滤步骤后变为无效的数据。

### Extended Description

* 攻击者可以使用此方法绕过验证并发起攻击，从而暴露可能会被阻止的弱点，例如注入。

## CWE - 182 Collapse of Data into Unsafe Value

中文：**数据崩溃成不安全的价值**

### Description

* 该软件以某种方式过滤数据，导致数据被缩减或“折叠”为违反预期安全属性的不安全值。

### Extended Description

* 没有详细描述

## CWE - 183 Permissive Whitelist

中文：**允许的白名单**

### Description

* 应用程序使用可接受值的“白名单”，但白名单包含至少一个不安全值，从而导致产生的弱点。

### Extended Description

* 没有详细描述

## CWE - 184 Incomplete Blacklist

中文：**不完整的黑名单**

### Description

* 应用程序使用禁止值的“黑名单”，但黑名单不完整。

### Extended Description

* 如果将不完整的黑名单用作安全机制，则软件可能允许非预期的值传递到应用程序逻辑中。

## CWE - 185 Incorrect Regular Expression

中文：**正则表达式不正确**

### Description

* 该软件以导致数据不正确匹配或比较的方式指定正则表达式。

### Extended Description

* 当正则表达式用于过滤或验证等保护机制时，这可能允许攻击者绕过对传入数据的预期限制。

## CWE - 186 Overly Restrictive Regular Expression

中文：**过度限制性的正则表达式**

### Description

* 正则表达式过于严格，可以防止检测到危险值。

### Extended Description

* 这个弱点与正则表达式的复杂性无关。而是关于正则表达式与所有预期的值不匹配。考虑使用正则表达式将可接受的值列入白名单或将不需要的术语列入黑名单。过度限制性的正则表达式错过了一些潜在的安全相关值，导致误报*或*假阴性，这取决于在代码中如何使用正则表达式。考虑表达式/ [0-8] /其中意图是/ [0-9] /。这个表达式并不“复杂”，但是当程序员计划检查它时，值“9”不匹配。

## CWE - 187 Partial String Comparison

中文：**部分字符串比较**

### Description

* 该软件执行比较，该比较仅在确定是否存在匹配（例如子串）之前检查因子的一部分，从而导致产生的弱点。

### Extended Description

* 例如，攻击者可能通过提供与较大的正确密码的关联部分匹配的小密码来成功进行身份验证。

## CWE - 188 Reliance on Data/Memory Layout

中文：**依赖于数据/内存布局**

### Description

* 该软件对协议数据或内存如何在较低级别组织进行无效假设，从而导致意外的程序行为。

### Extended Description

* 更改平台或协议版本时，内存中的数据组织可能会以非预期的方式发生变化。例如，某些体系结构可能将局部变量A和B放在彼此旁边，A顶部;有些人可能会把它们放在一起，B顶在上面;和其他人可能会添加一些填充。填充大小可以变化以确保每个变量与适当的字大小对齐。 在协议实现中，通常计算相对于另一个字段的偏移量以挑选出特定的数据。特殊情况（通常涉及新协议版本）可能会添加以不寻常方式更改数据布局的极端情况。结果可能是实现访问数据包中的非预期字段，将一种类型的数据视为另一种类型的数据。

## CWE - 190 Integer Overflow or Wraparound

中文：**整数溢出或环绕**

### Description

* 当逻辑假定结果值始终大于原始值时，软件执行可产生整数溢出或环绕的计算。当计算用于资源管理或执行控制时，这可能引入其他弱点。

### Extended Description

* 当整数值递增到太大而无法存储在关联表示中的值时，会发生整数溢出或回绕。发生这种情况时，该值可能会换行变为非常小或负数。虽然这可能是在依赖包装的情况下的预期行为，但如果包装是意外的，则会产生安全性后果。如果可以使用用户提供的输入触发整数溢出，则尤其如此。当结果用于控制循环，做出安全决策或确定行为（例如内存分配，复制，连接等）中的偏移量或大小时，这将成为安全关键。

## CWE - 191 Integer Underflow (Wrap or Wraparound)

中文：**整数下溢（换行或环绕）**

### Description

* 该产品从另一个值中减去一个值，使得结果小于允许的最小整数值，这会产生一个不等于正确结果的值。

### Extended Description

* 这可能发生在签名和未签名的情况下。

## CWE - 192 Integer Coercion Error

中文：**整数强制误差**

### Description

* 整数强制是指与原始数据类型的类型转换，扩展或截断有关的一组缺陷。

### Extended Description

* 几个缺陷属于整数强制误差的范畴。在大多数情况下，这些错误本身只会导致可用性和数据完整性问题。但是，在某些情况下，它们可能会导致其他更复杂的安全相关缺陷，例如缓冲区溢出情况。

## CWE - 193 Off-by-one Error

中文：**一个错误**

### Description

* 产品计算或使用的错误最大值或最小值比正确值多1或少1。

### Extended Description

* 没有详细描述

## CWE - 194 Unexpected Sign Extension

中文：**意外的标志扩展**

### Description

* 该软件对一个数字执行操作，使其在转换为更大的数据类型时进行符号扩展。当原始数字为负数时，这会产生意外的值，从而导致产生的弱点。

### Extended Description

* 没有详细描述

## CWE - 195 Signed to Unsigned Conversion Error

中文：**签名到无符号转换错误**

### Description

* 该软件使用带符号的基元并对无符号基元执行转换，如果无法使用无符号基元表示有符号基元的值，则可以产生意外值。

### Extended Description

* 依赖有符号和无符号数之间的隐式强制转换是危险的，因为结果可能会产生意外的值并违反程序所做的假设。 通常，函数将返回负值以指示失败。当函数的结果用作大小参数时，使用这些负返回值可能会产生意外结果。例如，如果将负大小值传递给标准内存副本或分配函数，则它们将隐式转换为大的无符号值。这可能导致可利用的缓冲区溢出或下溢情况。

## CWE - 196 Unsigned to Signed Conversion Error

中文：**无符号转换为有符号转换错误**

### Description

* 该软件使用无符号基元并对有符号基元执行强制转换，如果无符号基元的值无法使用有符号基元表示，则可能会产生意外值。

### Extended Description

* 尽管问题比签名到无符号转换的频率低，但无符号到签名的转换可能是危险缓冲区保护条件的完美前提，这些条件允许攻击者向下移动堆栈，否则它们可能无法在正常的缓冲区溢出条件下访问。当大的无符号值转换为有符号值，然后用作缓冲区的索引或指针算术时，缓冲区承保经常发生。

## CWE - 197 Numeric Truncation Error

中文：**数字截断错误**

### Description

* 当基元转换为较小尺寸的基元并且数据在转换中丢失时，会发生截断错误。

### Extended Description

* 当基元被转换为较小的基元时，大值的高位比特在转换中丢失，可能导致意外的值不等于原始值。可能需要此值作为缓冲区，循环迭代器或简单必需状态数据的索引。在任何情况下，该值都不可信，系统将处于未定义状态。尽管可以有效地使用该方法来隔离值的低位，但这种用法很少，并且截断通常意味着发生了实现错误。

## CWE - 198 Use of Incorrect Byte Ordering

中文：**使用不正确的字节顺序**

### Description

* 软件接收来自上游组件的输入，但在处理输入时不考虑字节排序（例如，大端和小端），导致使用不正确的数字或值。

### Extended Description

* 没有详细描述

## CWE - 200 Information Exposure

中文：**信息曝光**

### Description

* 信息暴露是有意或无意地向未明确授权访问该信息的行为者披露信息。

### Extended Description

* 信息要么：

在产品自身的功能中被视为敏感的，例如私人消息;要么 提供有关产品或其环境的信息，这些信息可能在攻击中有用，但攻击者通常无法使用，例如可远程访问的产品的安装路径。

许多信息暴露是结果（例如，PHP脚本错误揭示了程序的完整路径），但它们也可能是主要的（例如加密中的时间差异）。有许多不同类型的问题涉及信息曝光。它们的严重程度可以根据所揭示的信息类型而广泛。

## CWE - 201 Information Exposure Through Sent Data

中文：**通过已发送数据的信息曝光**

### Description

* 通过发送数据意外暴露敏感信息是指数据的传输，这些数据本身是敏感的，或者在通过标准数据信道进一步利用系统时是有用的。

### Extended Description

* 没有详细描述

## CWE - 202 Exposure of Sensitive Data Through Data Queries

中文：**通过数据查询暴露敏感数据**

### Description

* 在尝试保密信息时，攻击者通常可以使用统计信息来推断某些信息。

### Extended Description

* 在数据不应该绑定到单个用户的情况下，但是大量用户应该能够进行“擦除”用户身份的查询，则可以获得关于用户的信息 - 例如，通过指定搜索已知对该用户而言唯一的术语。

## CWE - 203 Information Exposure Through Discrepancy

中文：**信息暴露通过差异**

### Description

* 产品的行为不同或以暴露与产品状态相关的安全相关信息的方式发送不同的响应，例如特定操作是否成功。

### Extended Description

* 没有详细描述

## CWE - 204 Response Discrepancy Information Exposure

中文：**响应差异信息暴露**

### Description

* 该软件以允许参与者确定该参与者的控制范围之外的系统状态信息的方式对传入请求提供不同的响应。

### Extended Description

* 此问题在身份验证期间经常发生，其中失败登录消息的差异可能允许攻击者确定用户名是否有效。这些暴露可能是无意的（错误的）或有意的（设计）。

## CWE - 205 Information Exposure Through Behavioral Discrepancy

中文：**通过行为差异进行信息曝光**

### Description

* 产品的行为表明基于（1）产品的内部状态或（2）与同一类别中的其他产品的差异的重要差异。

### Extended Description

* 例如，OS指纹识别等攻击在很大程度上依赖于行为和响应差异。

## CWE - 206 Information Exposure of Internal State Through Behavioral Inconsistency

中文：**通过行为不一致信息暴露内部状态**

### Description

* 产品中的两个单独操作会导致产品以可被攻击者观察到的方式表现不同，并显示有关产品内部状态的安全相关信息，例如特定操作是否成功。

### Extended Description

* 没有详细描述

## CWE - 207 Information Exposure Through an External Behavioral Inconsistency

中文：**信息暴露通过外部行为不一致**

### Description

* 该产品的行为与其他类似产品的行为不同，其方式是攻击者可以观察到并暴露与使用哪种产品相关的安全相关信息。

### Extended Description

* 没有详细描述

## CWE - 208 Information Exposure Through Timing Discrepancy

中文：**信息暴露通过时间差异**

### Description

* 产品中的两个单独的操作需要不同的时间来完成，其方式是对于演员可观察到并且揭示关于产品状态的安全相关信息，例如特定操作是否成功。

### Extended Description

* 没有详细描述

## CWE - 209 Information Exposure Through an Error Message

中文：**信息通过错误消息曝光**

### Description

* 该软件会生成一条错误消息，其中包含有关其环境，用户或关联数据的敏感信息。

### Extended Description

* 敏感信息本身可能是有价值的信息（例如密码），或者对于发起其他更致命的攻击可能是有用的。如果攻击失败，攻击者可能会使用服务器提供的错误信息来启动另一个更集中的攻击。例如，尝试利用路径遍历弱点（CWE-22）可能会产生已安装应用程序的完整路径名。反过来，这可以用于选择适当数量的“..”序列以导航到目标文件。使用SQL注入（CWE-89）的攻击最初可能不会成功，但错误消息可能会显示格式错误的查询，这会暴露查询逻辑甚至是查询中使用的密码或其他敏感信息。

## CWE - 210 Information Exposure Through Self-generated Error Message

中文：**通过自生错误消息进行信息曝光**

### Description

* 软件识别错误情况并创建自己的包含敏感信息的诊断或错误消息。

### Extended Description

* 没有详细描述

## CWE - 211 Information Exposure Through Externally-Generated Error Message

中文：**通过外部生成的错误消息进行信息暴露**

### Description

* 该软件执行触发外部诊断或错误消息的操作，该消息不是由软件直接生成的，例如由软件使用的编程语言解释器生成的错误。该错误可能包含敏感的系统信息。

### Extended Description

* 没有详细描述

## CWE - 212 Improper Cross-boundary Removal of Sensitive Data

中文：**不正确的跨境删除敏感数据**

### Description

* 该软件使用包含敏感数据的资源，但在与其他控制领域中的actor存储，传输或共享资源之前，它不会正确删除该数据。

### Extended Description

* 可能包含敏感数据的资源包括文档，数据包，消息，数据库等。虽然此数据可能对共享资源的单个用户或小组用户有用，但可能需要在资源可以在外部共享之前将其删除受信任的组织。去除过程有时称为清洁或擦洗。 例如，用于编辑文档的软件可能不会删除敏感数据，例如审阅者注释或存储文档的本地路径名。或者，在向Internet站点发出传出请求之前，代理可能不会从标头中删除内部IP地址。

## CWE - 213 Intentional Information Exposure

中文：**故意信息曝光**

### Description

* 产品的设计或配置明确要求发布可被管理员视为敏感的信息。

### Extended Description

* 没有详细描述

## CWE - 214 Information Exposure Through Process Environment

中文：**通过流程环境进行信息曝光**

### Description

* 使用敏感参数，环境变量或操作系统上其他进程可以看到的其他元素调用进程。

### Extended Description

* 许多操作系统允许用户列出有关其他用户拥有的进程的信息。此信息可能包括命令行参数或环境变量设置。当此数据包含敏感信息（如凭据）时，可能允许其他用户对软件或相关资源发起攻击。

## CWE - 215 Information Exposure Through Debug Information

中文：**通过调试信息暴露信息**

### Description

* 该应用程序包含调试代码，可以将敏感信息暴露给不受信任的各方。

### Extended Description

* 没有详细描述

## CWE - 216 Containment Errors (Container Errors)

中文：**遏制错误（容器错误）**

### Description

* 这试图涵盖在“容器”中包含不正确数据的各种问题。

### Extended Description

* 没有详细描述

## CWE - 217 DEPRECATED: Failure to Protect Stored Data from Modification

中文：**已弃用：未能保护存储数据不被修改**

### Description

* 这种弱点已被弃用，因为它融合了许多弱点并使其混淆。以前在这个弱点中涵盖的问题可以在CWE-766和CWE-767中找到。

### Extended Description

* 没有详细描述

## CWE - 218 DEPRECATED (Duplicate): Failure to provide confidentiality for stored data

中文：**已弃用（重复）：未对存储的数据提供机密性**

### Description

* 这个弱点已被弃用，因为它与CWE-493重复。所有内容均已转移至CWE-493。

### Extended Description

* 没有详细描述

## CWE - 219 Sensitive Data Under Web Root

中文：**Web Root下的敏感数据**

### Description

* 应用程序将敏感数据存储在Web文档根目录下，访问控制不足，这可能使不受信任的各方可以访问它们。

### Extended Description

* 没有详细描述

## CWE - 220 Sensitive Data Under FTP Root

中文：**FTP根目录下的敏感数据**

### Description

* 应用程序将敏感数据存储在FTP文档根目录下，访问控制不足，这可能使不受信任方可以访问它。

### Extended Description

* 没有详细描述

## CWE - 221 Information Loss or Omission

中文：**信息丢失或遗漏**

### Description

* 该软件不会记录或不正确地记录导致错误决策或妨碍以后分析的安全相关信息。

### Extended Description

* 这可能是结果，例如，缓冲区溢出可能会在产品记录事件之前触发崩溃。

## CWE - 222 Truncation of Security-relevant Information

中文：**截断安全相关信息**

### Description

* 应用程序以可能模糊攻击来源或性质的方式截断安全相关信息的显示，记录或处理。

### Extended Description

* 没有详细描述

## CWE - 223 Omission of Security-relevant Information

中文：**省略与安全相关的信息**

### Description

* 应用程序不会记录或显示对于识别攻击的来源或性质或确定某个操作是否安全非常重要的信息。

### Extended Description

* 没有详细描述

## CWE - 224 Obscured Security-relevant Information by Alternate Name

中文：**通过替代名称隐藏安全相关信息**

### Description

* 该软件根据受影响实体的备用名称而不是规范名称记录与安全相关的信息。

### Extended Description

* 没有详细描述

## CWE - 225 DEPRECATED (Duplicate): General Information Management Problems

中文：**弃用（重复）：一般信息管理问题**

### Description

* 这个弱点可以在CWE-199找到。

### Extended Description

* 没有详细描述

## CWE - 226 Sensitive Information Uncleared Before Release

中文：**敏感信息在发布前未清除**

### Description

* 在将该资源提供给另一个控制领域的一方之前，该软件不能完全清除数据结构，文件或其他资源中以前使用的信息。

### Extended Description

* 这通常是由新数据产生的，这些新数据不像旧数据那样长，这使得旧数据的一部分仍然可用。在数据长度可变但相关数据结构不可变的其他情况下，可能会发生等效错误。如果在使用后未清除内存，则可能允许非预期的actor在重新分配内存时读取数据。

## CWE - 228 Improper Handling of Syntactically Invalid Structure

中文：**句法无效结构的处理不当**

### Description

* 产品不处理或错误处理与相关规范相关的语法不完整的输入。

### Extended Description

* 没有详细描述

## CWE - 229 Improper Handling of Values

中文：**不正确的价值处理**

### Description

* 如果输入中未提供参数，字段或参数的预期数量的值，或者这些值未定义，则软件无法正确处理。

### Extended Description

* 没有详细描述

## CWE - 230 Improper Handling of Missing Values

中文：**错误值的处理不当**

### Description

* 指定参数，字段或参数名称时，软件不处理或错误处理，但缺少关联值，即它为空，空白或空。

### Extended Description

* 没有详细描述

## CWE - 231 Improper Handling of Extra Values

中文：**对额外值的处理不当**

### Description

* 当提供的值多于预期时，软件不会处理或错误处理。

### Extended Description

* 没有详细描述

## CWE - 232 Improper Handling of Undefined Values

中文：**未定义值的处理不当**

### Description

* 如果未为相关参数，字段或参数名称定义或支持值，则软件不会处理或错误处理。

### Extended Description

* 没有详细描述

## CWE - 233 Improper Handling of Parameters

中文：**参数处理不当**

### Description

* 如果输入中未提供预期数量的参数，字段或参数，或者这些参数未定义，则软件无法正确处理。

### Extended Description

* 没有详细描述

## CWE - 234 Failure to Handle Missing Parameter

中文：**无法处理缺失的参数**

### Description

* 如果向函数发送的参数太少，该函数仍会从堆栈中弹出预期数量的参数。潜在地，可以在函数中耗尽可变数量的参数。

### Extended Description

* 没有详细描述

## CWE - 235 Improper Handling of Extra Parameters

中文：**额外参数的处理不当**

### Description

* 当具有相同名称的参数，字段或参数的数量超过预期量时，软件不会处理或错误处理。

### Extended Description

* 没有详细描述

## CWE - 236 Improper Handling of Undefined Parameters

中文：**未定义参数的处理不当**

### Description

* 当产品未定义或支持特定参数，字段或参数名称时，软件不会处理或错误处理。

### Extended Description

* 没有详细描述

## CWE - 237 Improper Handling of Structural Elements

中文：**结构元素处理不当**

### Description

* 该软件不处理或错误处理与复杂结构相关的输入。

### Extended Description

* 没有详细描述

## CWE - 238 Improper Handling of Incomplete Structural Elements

中文：**对不完整结构元素的处理不当**

### Description

* 当未完全指定特定结构元素时，软件不处理或错误处理。

### Extended Description

* 没有详细描述

## CWE - 239 Failure to Handle Incomplete Element

中文：**未能处理不完整的元素**

### Description

* 如果未完全指定特定元素，则软件无法正确处理。

### Extended Description

* 没有详细描述

## CWE - 240 Improper Handling of Inconsistent Structural Elements

中文：**对不一致结构元素的处理不当**

### Description

* 当两个或多个结构元素应该一致时，软件不会处理或错误处理，但不是。

### Extended Description

* 没有详细描述

## CWE - 241 Improper Handling of Unexpected Data Type

中文：**不正确处理意外数据类型**

### Description

* 当特定元素不是预期类型时，软件不处理或错误处理，例如它需要一个数字（0-9），但提供一个字母（A-Z）。

### Extended Description

* 没有详细描述

## CWE - 242 Use of Inherently Dangerous Function

中文：**使用固有危险的功能**

### Description

* 该程序调用一个永远不能保证安全工作的函数。

### Extended Description

* 无论如何使用，某些功能都会以危险的方式运行。通常在不考虑安全问题的情况下实施此类别中的功能。 gets（）函数是不安全的，因为它不对其输入的大小执行边界检查。攻击者可以轻松地将任意大小的输入发送到gets（）并溢出目标缓冲区。类似地，在读取静态分配的字符数组时，>> operator不安全，因为它不对输入的大小执行边界检查。攻击者可以轻松地将任意大小的输入发送到>>运算符并溢出目标缓冲区。

## CWE - 243 Creation of chroot Jail Without Changing Working Directory

中文：**在不改变工作目录的情况下创建chroot监狱**

### Description

* 该程序使用chroot（）系统调用来创建一个jail，但之后不会更改工作目录。这不会阻止访问jail之外的文件。

### Extended Description

* chroot（）的不当使用可能允许攻击者逃离chroot监狱。 chroot（）函数调用不会更改进程的当前工作目录，因此在调用chroot（）之后，相对路径仍可能引用chroot jail之外的文件系统资源。

## CWE - 244 Improper Clearing of Heap Memory Before Release ('Heap Inspection')

中文：**释放前不正确清除堆内存（'堆检查'）**

### Description

* 使用realloc（）来调整存储敏感信息的缓冲区可能会使敏感信息暴露于攻击，因为它不会从内存中删除。

### Extended Description

* 如果未从内存中删除敏感数据（如密码或加密密钥），则可能会使用“堆检查”攻击向攻击者公开，该攻击使用内存转储或其他方法读取敏感数据。 realloc（）函数通常用于增加已分配内存块的大小。此操作通常需要将旧存储器块的内容复制到新的更大的块中。此操作使原始块的内容保持不变但程序无法访问，从而阻止程序从内存中擦除敏感数据。如果攻击者以后可以检查内存转储的内容，则可能会暴露敏感数据。

## CWE - 245 J2EE Bad Practices: Direct Management of Connections

中文：**J2EE不良做法：直接管理连接**

### Description

* J2EE应用程序直接管理连接，而不是使用容器的连接管理工具。

### Extended Description

* J2EE标准禁止直接管理连接。它要求应用程序使用容器的资源管理工具来获取与资源的连接。每个主要Web应用程序容器都提供池化数据库连接管理作为其资源管理框架的一部分。在应用程序中复制此功能很困难且容易出错，这是J2EE标准禁止的部分原因。

## CWE - 246 J2EE Bad Practices: Direct Use of Sockets

中文：**J2EE不良做法：直接使用套接字**

### Description

* J2EE应用程序直接使用套接字而不是使用框架方法调用。

### Extended Description

* 当没有更高级别的协议可用时，J2EE标准仅允许将套接字用于与遗留系统通信的目的。编写自己的通信协议需要解决棘手的安全问题。 如果没有安全专家的严格审查，自定义通信协议可能会遇到安全问题。许多相同的问题适用于标准协议的自定义实现。虽然通常有更多资源可用于解决与实施标准协议相关的安全问题，但攻击者也可以使用这些资源。

## CWE - 247 DEPRECATED (Duplicate): Reliance on DNS Lookups in a Security Decision

中文：**弃用（重复）：依赖于安全决策中的DNS查找**

### Description

* 此条目已被弃用，因为它与CWE-350重复。所有内容均已转移至CWE-350。

### Extended Description

* 没有详细描述

## CWE - 248 Uncaught Exception

中文：**未捕获的异常**

### Description

* 从函数抛出异常，但它没有被捕获。

### Extended Description

* 未捕获异常时，可能会导致程序崩溃或泄露敏感信息。

## CWE - 249 DEPRECATED: Often Misused: Path Manipulation

中文：**弃用：经常被滥用：路径操纵**

### Description

* 由于名称混淆以及多个弱点的偶然组合，此条目已被弃用。其大部分内容已转移至CWE-785。

### Extended Description

* 没有详细描述

## CWE - 250 Execution with Unnecessary Privileges

中文：**执行不必要的权限**

### Description

* 该软件以高于所需最低级别的权限级别执行操作，这会产生新的弱点或放大其他弱点的后果。

### Extended Description

* 可能会暴露新的弱点，因为使用额外的特权（例如root或Administrator）运行可能会禁用操作系统或周围环境执行的正常安全检查。如果在提升权限下运行时，其他预先存在的弱点可能会变成安全漏洞。 权限管理功能可以以一些不太明显的方式运行，并且它们在不同平台上具有不同的怪癖。如果您从一个非root用户转换到另一个非root用户，则这些不一致性尤其明显。信号处理程序和生成的进程在拥有进程的特权下运行，因此如果在信号触发或执行子进程时进程以root身份运行，则信号处理程序或子进程将以root权限运行。

## CWE - 252 Unchecked Return Value

中文：**未选中的返回值**

### Description

* 软件不检查方法或函数的返回值，这可以防止它检测到意外的状态和条件。

### Extended Description

* 两个常见的程序员假设是“这个函数调用永远不会失败”和“这个函数调用失败并不重要”。如果攻击者可以强制该函数失败或以其他方式返回不期望的值，则后续程序逻辑可能导致漏洞，因为该软件不处于程序员假定的状态。例如，如果程序调用函数来删除权限但不检查返回代码以确保成功删除权限，则程序将继续以更高权限运行。

## CWE - 253 Incorrect Check of Function Return Value

中文：**函数返回值检查错误**

### Description

* 软件错误地检查函数的返回值，这会阻止软件检测错误或异常情况。

### Extended Description

* 重要和常见的功能将为其行动的成功返回一些价值。这将提醒程序是否处理由该功能引起的任何错误。

## CWE - 256 Unprotected Storage of Credentials

中文：**不受保护的凭证存储**

### Description

* 以明文存储密码可能会导致系统受损。

### Extended Description

* 当密码以纯文本形式存储在应用程序的属性或配置文件中时，会出现密码管理问题。在配置文件中存储明文密码允许任何能够读取该文件的人访问受密码保护的资源。

## CWE - 257 Storing Passwords in a Recoverable Format

中文：**以可恢复的格式存储密码**

### Description

* 以可恢复格式存储密码使得它们受到恶意用户的密码重用攻击。实际上，应该注意的是，可恢复的加密密码与明文密码相比没有明显的好处，因为它们不仅可以被恶意攻击者重用，还可以被恶意内部人员重用。如果系统管理员可以直接恢复密码，或对可用信息使用强力搜索，则管理员可以在其他帐户上使用密码。

### Extended Description

* 没有详细描述

## CWE - 258 Empty Password in Configuration File

中文：**配置文件中的空密码**

### Description

* 使用空字符串作为密码是不安全的。

### Extended Description

* 没有详细描述

## CWE - 259 Use of Hard-coded Password

中文：**使用硬编码密码**

### Description

* 该软件包含一个硬编码密码，用于自己的入站身份验证或外部组件的出站通信。

### Extended Description

* 硬编码密码通常会导致严重的身份验证失败，系统管理员很难检测到。一旦检测到，就很难修复，因此管理员可能会被迫完全禁用该产品。主要有两种变化：

入站：该软件包含一个检查硬编码密码的身份验证机制。 出站：软件连接到另一个系统或组件，它包含用于连接到该组件的硬编码密码。

在入站变体中，将创建默认管理帐户，并将简单密码硬编码到产品中并与该帐户关联。此硬编码密码对于产品的每次安装都是相同的，并且系统管理员通常无法在不手动修改程序或修补软件的情况下更改或禁用密码。如果密码被发现或发布（在Internet上很常见），那么任何知道此密码的人都可以访问该产品。最后，由于软件的所有安装都具有相同的密码，即使在不同的组织中，也可以实现诸如蠕虫之类的大规模攻击。 Outbound变体适用于使用后端服务进行身份验证的前端系统。后端服务可能需要一个可以轻松发现的固定密码。程序员可以简单地将这些后端凭证硬编码到前端软件中。该程序的任何用户都可以提取密码。具有硬编码密码的客户端系统构成了更大的威胁，因为从二进制文件中提取密码通常非常简单。

## CWE - 260 Password in Configuration File

中文：**配置文件中的密码**

### Description

* 该软件将密码存储在配置文件中，该配置文件可能对不知道密码的演员可访问。

### Extended Description

* 这可能导致使用密码的系统受到损害。攻击者可以访问此文件并了解存储的密码，或者更糟糕的是，将密码更改为他们选择的密码。

## CWE - 261 Weak Cryptography for Passwords

中文：**密码弱密码学**

### Description

* 使用简单编码隐藏密码不会保护密码。

### Extended Description

* 没有详细描述

## CWE - 262 Not Using Password Aging

中文：**不使用密码时效**

### Description

* 如果没有适当的机制来管理密码老化，用户将无法及时更新密码。

### Extended Description

* 没有详细描述

## CWE - 263 Password Aging with Long Expiration

中文：**长期过期的密码时效**

### Description

* 允许密码老化未经检查可能导致密码完整性降低。

### Extended Description

* 正如忽略包含管理密码老化的功能是危险的，因此允许密码老化继续未选中。密码必须具有最长寿命，之后用户需要使用新的不同密码进行更新。

## CWE - 266 Incorrect Privilege Assignment

中文：**权限分配不正确**

### Description

* 产品错误地将特权分配给特定的actor，为该actor创建一个非预期的控制范围。

### Extended Description

* 没有详细描述

## CWE - 267 Privilege Defined With Unsafe Actions

中文：**使用不安全操作定义的权限**

### Description

* 特定的权限，角色，功能或权限可用于执行非预期的不安全操作，即使将其分配给正确的实体也是如此。

### Extended Description

* 没有详细描述

## CWE - 268 Privilege Chaining

中文：**特权链接**

### Description

* 可以以允许实体执行不允许没有该组合的不安全动作的方式组合两个不同的特权，角色，功能或权限。

### Extended Description

* 没有详细描述

## CWE - 269 Improper Privilege Management

中文：**权限管理不当**

### Description

* 该软件未正确分配，修改，跟踪或检查演员的权限，从而为该演员创建了一个非预期的控制范围。

### Extended Description

* 没有详细描述

## CWE - 270 Privilege Context Switching Error

中文：**特权上下文切换错误**

### Description

* 当软件在具有不同权限或控制范围的不同上下文之间切换时，该软件无法正确管理权限。

### Extended Description

* 没有详细描述

## CWE - 271 Privilege Dropping / Lowering Errors

中文：**权限下降/降低错误**

### Description

* 在将资源控制权交给没有这些权限的actor之前，该软件不会删除权限。

### Extended Description

* 在某些情况下，使用提升权限执行的系统将切换进程/文件/等。到另一个进程或用户。如果未降低实体的权限，则提升的权限会在整个系统中传播，并可能传播给攻击者。

## CWE - 272 Least Privilege Violation

中文：**最低权限违规**

### Description

* 执行操作后，应立即删除执行chroot（）等操作所需的提升权限级别。

### Extended Description

* 没有详细描述

## CWE - 273 Improper Check for Dropped Privileges

中文：**不正确检查已删除的权限**

### Description

* 该软件尝试删除权限，但不检查或错误检查以查看删除是否成功。

### Extended Description

* 如果删除失败，软件将继续使用提升的权限运行，这可能会为非特权用户提供额外的访问权限。

## CWE - 274 Improper Handling of Insufficient Privileges

中文：**权利不足的处理不当**

### Description

* 当软件没有足够的权限执行操作时，软件不会处理或错误处理，从而导致产生的缺陷。

### Extended Description

* 没有详细描述

## CWE - 276 Incorrect Default Permissions

中文：**默认权限不正确**

### Description

* 安装后，该软件为将其公开给非预期的actor的对象设置了不正确的权限。

### Extended Description

* 没有详细描述

## CWE - 277 Insecure Inherited Permissions

中文：**不安全的继承权限**

### Description

* 产品定义一组不安全的权限，这些权限由程序创建的对象继承。

### Extended Description

* 没有详细描述

## CWE - 278 Insecure Preserved Inherited Permissions

中文：**不安全的保留继承权限**

### Description

* 产品继承了一组对象的不安全权限，例如从存档文件复制时，没有用户意识或参与。

### Extended Description

* 没有详细描述

## CWE - 279 Incorrect Execution-Assigned Permissions

中文：**执行分配的权限不正确**

### Description

* 在执行时，软件以违反用户指定的预期权限的方式设置对象的权限。

### Extended Description

* 没有详细描述

## CWE - 280 Improper Handling of Insufficient Permissions or Privileges

中文：**对权限或权限不足的处理不当**

### Description

* 当应用程序没有足够的权限访问其权限所指定的资源或功能时，应用程序不会处理或错误处理。这可能导致它遵循可能使应用程序处于无效状态的意外代码路径。

### Extended Description

* 没有详细描述

## CWE - 281 Improper Preservation of Permissions

中文：**权限保护不当**

### Description

* 在复制，还原或共享对象时，该软件不保留权限或错误地保留权限，这可能导致它们具有比预期更少的限制权限。

### Extended Description

* 没有详细描述

## CWE - 282 Improper Ownership Management

中文：**所有权管理不当**

### Description

* 该软件分配了错误的所有权，或者没有正确验证对象或资源的所有权。

### Extended Description

* 没有详细描述

## CWE - 283 Unverified Ownership

中文：**未经证实的所有权**

### Description

* 该软件未正确验证关键资源是否由适当的实体拥有。

### Extended Description

* 没有详细描述

## CWE - 284 Improper Access Control

中文：**访问控制不当**

### Description

* 该软件不限制或不正确地限制对未授权演员的资源访问。

### Extended Description

* 访问控制涉及使用多种保护机制，例如：

身份验证（证明演员的身份） 授权（确保给定的actor可以访问资源），以及 问责制（跟踪已执行的活动）

当任何机制未应用或以其他方式失败时，攻击者可以通过获取权限，读取敏感信息，执行命令，逃避检测等来危害软件的安全性。 有两种不同的行为可能会引入访问控制缺陷：

规范：为用户或资源明确指定了不正确的权限，权限，所有权等（例如，将密码文件设置为可全局写入，或为访客用户提供管理员功能）。此操作可由程序或管理员执行。 强制执行：该机制包含的错误会阻止它正确执行指定的访问控制要求（例如，允许用户指定自己的权限，或允许语法错误的ACL产生不安全的设置）。此问题发生在程序本身内，因为它实际上并未强制执行管理员指定的预期安全策略。

## CWE - 285 Improper Authorization

中文：**授权不当**

### Description

* 当actor尝试访问资源或执行操作时，该软件不执行或错误地执行授权检查。

### Extended Description

* 假设具有给定身份的用户，授权是基于用户的权限和适用于资源的任何权限或其他访问控制规范来确定该用户是否可以访问给定资源的过程。 当访问控制检查未一致地应用 - 或根本不应用 - 用户能够访问数据或执行不应被允许执行的操作。这可能导致各种各样的问题，包括信息泄露，拒绝服务和任意代码执行。

## CWE - 286 Incorrect User Management

中文：**用户管理不正确**

### Description

* 该软件无法在其环境中正确管理用户。

### Extended Description

* 可以将用户分配给错误的权限组（类），从而导致对敏感对象的非预期访问权限。

## CWE - 287 Improper Authentication

中文：**身份验证不正确**

### Description

* 当演员声称拥有特定身份时，该软件不会证明或证明该声明是正确的。

### Extended Description

* 没有详细描述

## CWE - 288 Authentication Bypass Using an Alternate Path or Channel

中文：**使用备用路径或通道进行身份验证旁路**

### Description

* 产品需要身份验证，但产品具有不需要身份验证的备用路径或通道。

### Extended Description

* 没有详细描述

## CWE - 289 Authentication Bypass by Alternate Name

中文：**身份验证绕过备用名称**

### Description

* 该软件基于正在访问的资源的名称或执行访问的actor的名称执行身份验证，但它没有正确检查该资源或actor的所有可能名称。

### Extended Description

* 没有详细描述

## CWE - 290 Authentication Bypass by Spoofing

中文：**通过欺骗进行身份验证绕过**

### Description

* 这种以攻击为中心的弱点是由不正确实施的受欺骗攻击的身份验证方案引起的。

### Extended Description

* 没有详细描述

## CWE - 291 Reliance on IP Address for Authentication

中文：**依赖于身份验证的IP地址**

### Description

* 该软件使用IP地址进行身份验证。

### Extended Description

* IP地址很容易被欺骗。攻击者可以伪造他们发送的数据包的源IP地址，但响应数据包将返回伪造的IP地址。要查看响应数据包，攻击者必须嗅探受害者计算机和伪造IP地址之间的流量。为了完成所需的嗅探，攻击者通常会尝试将自己定位在与受害者计算机相同的子网上。攻击者可以通过使用源路由来规避此要求，但是今天在大部分Internet上都禁用了源路由。总之，IP地址验证可以是身份验证方案的有用部分，但它不应该是身份验证所需的单一因素。

## CWE - 292 DEPRECATED (Duplicate): Trusting Self-reported DNS Name

中文：**DEPRECATED（重复）：信任自我报告的DNS名称**

### Description

* 此条目已被弃用，因为它与CWE-350重复。所有内容均已转移至CWE-350。

### Extended Description

* 没有详细描述

## CWE - 293 Using Referer Field for Authentication

中文：**使用Referer字段进行身份验证**

### Description

* HTTP请求中的referer字段可以很容易地修改，因此不是消息完整性检查的有效方法。

### Extended Description

* 没有详细描述

## CWE - 294 Authentication Bypass by Capture-replay

中文：**身份验证绕过Capture-replay**

### Description

* 当软件设计使恶意用户可以通过将其重播到相关服务器以获得与原始消息相同的效果（或稍作更改）来嗅探网络流量并绕过身份验证时，存在捕获重放缺陷。

### Extended Description

* 捕获重放攻击很常见，如果没有加密技术，很难打败。它们是网络注入攻击的一个子集，它依赖于观察先前发送的有效命令，然后在必要时稍微更改它们并将相同的命令重新发送到服务器。

## CWE - 295 Improper Certificate Validation

中文：**证书验证不正确**

### Description

* 该软件不验证或错误验证证书。

### Extended Description

* 当证书无效或恶意时，它可能允许攻击者通过使用中间人（MITM）攻击欺骗可信实体。该软件可能连接到恶意主机，同时认为它是可信主机，或者软件可能被欺骗接受看似来自可信主机的欺骗数据。

## CWE - 296 Improper Following of a Certificate's Chain of Trust

中文：**不正确地遵循证书的信任链**

### Description

* 该软件不遵循或错误地遵循证书的信任链回到受信任的根证书，从而导致与该证书关联的任何资源的不正确信任。

### Extended Description

* 如果系统不遵循证书对根服务器的信任链，则证书将失去作为信任度量的所有有用性。从本质上讲，从证书中获得的信任来自信任链 - 在该列表的末尾有一个信誉良好的可信实体。最终用户必须信任该信誉良好的来源，并且这个信誉良好的来源必须通过证书媒介担保相关资源。 在某些情况下，这种信任遍历了几个互相担保的实体。最终用户信任的实体位于此信任链的一端，而证书挥舞资源位于链的另一端。如果用户在其中一个信任链的末尾收到证书，然后继续仅检查链中的第一个链接，则不会导出真正的信任，因为必须遍历整个链到可信来源以验证证书。 可以通过多种方式破坏信任链，包括但不限于：

链中的任何证书都是自签名的，除非它是根。 并非每个中间证书都会被检查，从原始证书一直到根证书。 中间的CA签名证书没有预期的基本约束或其他重要扩展。 根证书已被泄露或授权给错误的一方。

## CWE - 297 Improper Validation of Certificate with Host Mismatch

中文：**主机不匹配的证书验证不正确**

### Description

* 该软件与提供证书的主机通信，但该软件未正确确保证书实际与该主机关联。

### Extended Description

* 即使证书格式良好，签名并遵循信任链，它也可能只是与软件交互的站点不同的站点的有效证书。如果未正确检查证书的特定于主机的数据（例如主题中的公用名（CN）或X.509证书的主题备用名称（SAN）扩展名），则可能会出现重定向或欺骗攻击允许具有有效证书的恶意主机提供数据，模拟可信主机。为了确保数据完整性，证书必须有效，并且必须与正在访问的站点相关。 即使软件试图检查主机名，仍然可能错误地检查主机名。例如，攻击者可以创建一个名称以可信名称开头，后跟NUL字节的证书，这可能导致某些基于字符串的比较仅检查包含受信任名称的部分。 即使软件使用证书固定，如果软件在固定证书时未验证主机名，也会出现此弱点。

## CWE - 298 Improper Validation of Certificate Expiration

中文：**证书过期的验证不正确**

### Description

* 证书过期未经过验证或未正确验证，因此可能会将信任分配给因年龄而被放弃的证书。

### Extended Description

* 如果不考虑证书的到期，则不一定通过它传达信任。因此，无法验证证书的有效性，并且证书的所有好处都将丢失。

## CWE - 299 Improper Check for Certificate Revocation

中文：**不正确的检查证书撤销**

### Description

* 该软件不检查或错误地检查证书的撤销状态，这可能导致它使用已被泄露的证书。

### Extended Description

* 对证书撤销进行不正确的检查是一个比相关证书失败更严重的缺陷。这是因为使用任何撤销的证书几乎肯定是恶意的。证书撤销的最常见原因是有问题的系统遭到破坏，结果是没有合法的服务器将使用撤销的证书，除非它们非常不同步。

## CWE - 300 Channel Accessible by Non-Endpoint ('Man-in-the-Middle')

中文：**非端点可访问的频道（'中间人'）**

### Description

* 该产品不能充分验证通信信道两端的参与者的身份，或者不能充分确保信道的完整性，从而允许信道被非端点的参与者访问或影响。

### Extended Description

* 为了在双方之间建立安全通信，充分验证通信信道每端的实体身份通常很重要。验证不充分或不一致可能导致通信实体的识别不充分或不正确。这可能会产生负面影响，例如信道另一端的实体信任错位。攻击者可以通过插入通信实体和伪装成原始实体来利用它。在没有足够的身份验证的情况下，这样的攻击者可以窃听并可能修改原始实体之间的通信。

## CWE - 301 Reflection Attack in an Authentication Protocol

中文：**身份验证协议中的反射攻击**

### Description

* 如果恶意用户可以使用目标计算机模拟受信任的用户，则简单身份验证协议会受到反射攻击。

### Extended Description

* 相互认证协议要求每一方通过用预共享密钥加密来响应另一方的随机质询。然而，这些协议通常使用相同的预共享密钥来与多个不同实体进行通信。恶意用户或攻击者可以通过对协议采用反射攻击来轻松地破坏此协议，而无需拥有正确的密钥。

## CWE - 302 Authentication Bypass by Assumed-Immutable Data

中文：**认证绕过假定不可变数据**

### Description

* 身份验证方案或实现使用假定为不可变的关键数据元素，但可由攻击者控制或修改。

### Extended Description

* 没有详细描述

## CWE - 303 Incorrect Implementation of Authentication Algorithm

中文：**验证算法的不正确实现**

### Description

* 软件的要求决定了使用已建立的认证算法，但算法的实现是不正确的。

### Extended Description

* 这种不正确的实现可能允许绕过身份验证。

## CWE - 304 Missing Critical Step in Authentication

中文：**缺少身份验证的关键步骤**

### Description

* 该软件实现了一种身份验证技术，但它跳过了削弱技术的步骤。

### Extended Description

* 身份验证技术应遵循精确定义它们的算法，否则可以绕过身份验证或更容易受到暴力攻击。

## CWE - 305 Authentication Bypass by Primary Weakness

中文：**主要弱点的身份验证旁路**

### Description

* 验证算法是合理的，但是实现的机制可以被旁路，因为单独的弱点是认证错误的主要缺陷。

### Extended Description

* 没有详细描述

## CWE - 306 Missing Authentication for Critical Function

中文：**缺少关键功能的身份验证**

### Description

* 该软件不对需要可证明的用户身份或消耗大量资源的功能执行任何身份验证。

### Extended Description

* 没有详细描述

## CWE - 307 Improper Restriction of Excessive Authentication Attempts

中文：**过度认证尝试的不当限制**

### Description

* 该软件没有实施足够的措施来防止在短时间内发生多次失败的身份验证尝试，使其更容易受到暴力攻击。

### Extended Description

* 没有详细描述

## CWE - 308 Use of Single-factor Authentication

中文：**使用单因素身份验证**

### Description

* 与双因素身份验证方案的优势相比，使用单因素身份验证可能会导致不必要的危害风险。

### Extended Description

* 虽然使用多种身份验证方案只是在身份验证之上进行了更多的复杂化，但拥有这种冗余度量是非常有价值的。在互联网上使用弱，重用和通用密码是猖獗的。如果没有多个身份验证方案的额外保护，单个错误可能会导致帐户泄露。因此，如果可能有多种方案且易于使用，则应实施和要求它们。

## CWE - 309 Use of Password System for Primary Authentication

中文：**使用密码系统进行主要验证**

### Description

* 使用密码系统作为主要的认证手段可能存在若干缺陷或缺点，每个缺陷或缺点都会降低机制的有效性。

### Extended Description

* 没有详细描述

## CWE - 311 Missing Encryption of Sensitive Data

中文：**缺少敏感数据的加密**

### Description

* 在存储或传输之前，该软件不会加密敏感或关键信息。

### Extended Description

* 缺乏适当的数据加密传递了正确实施加密所传达的机密性，完整性和责任性的保证。

## CWE - 312 Cleartext Storage of Sensitive Information

中文：**明文存储敏感信息**

### Description

* 应用程序将敏感信息以明文形式存储在可能可供另一个控制领域访问的资源中。

### Extended Description

* 由于信息以明文形式存储，因此攻击者可能会阅读它。即使以非人类可读的方式对信息进行编码，某些技术也可以确定正在使用哪种编码，然后对信息进行解码。

## CWE - 313 Cleartext Storage in a File or on Disk

中文：**文件或磁盘上的明文存储**

### Description

* 应用程序将敏感信息以明文形式存储在文件或磁盘上。

### Extended Description

* 敏感信息可由具有文件访问权限的攻击者读取，或者通过对原始磁盘的物理或管理员访问权限读取。即使以非人类可读的方式对信息进行编码，某些技术也可以确定正在使用哪种编码，然后对信息进行解码。

## CWE - 314 Cleartext Storage in the Registry

中文：**注册表中的明文存储**

### Description

* 应用程序将敏感信息以明文形式存储在注册表中。

### Extended Description

* 攻击者可以通过访问注册表项来读取信息。即使以非人类可读的方式对信息进行编码，某些技术也可以确定正在使用哪种编码，然后对信息进行解码。

## CWE - 315 Cleartext Storage of Sensitive Information in a Cookie

中文：**明文在Cookie中存储敏感信息**

### Description

* 应用程序将敏感信息以明文形式存储在cookie中。

### Extended Description

* 攻击者可以使用广泛使用的工具来查看cookie并读取敏感信息。即使以非人类可读的方式对信息进行编码，某些技术也可以确定正在使用哪种编码，然后对信息进行解码。

## CWE - 316 Cleartext Storage of Sensitive Information in Memory

中文：**明文中存储敏感信息的明文**

### Description

* 应用程序将敏感信息以明文形式存储在内存中。

### Extended Description

* 敏感内存可能会保存到磁盘，存储在核心转储中，或者如果应用程序崩溃仍未清除，或者程序员在释放内存之前未正确清除内存。 可以说，这些问题通常只有具有管理员权限的人才可以利用。但是，交换可能会导致内存写入磁盘并使之后可以进行物理攻击。核心转储文件可能具有不安全的权限，或存储在不受信任的人可访问的归档文件中。或者，由于另一个弱点，未清除的敏感内存可能会无意中暴露给攻击者。

## CWE - 317 Cleartext Storage of Sensitive Information in GUI

中文：**GUI中敏感信息的明文存储**

### Description

* 应用程序在GUI中以明文形式存储敏感信息。

### Extended Description

* 通过使用API直接访问GUI对象（如窗口和菜单），攻击者通常可以从GUI获取数据，即使是隐藏的。即使以非人类可读的方式对信息进行编码，某些技术也可以确定正在使用哪种编码，然后对信息进行解码。

## CWE - 318 Cleartext Storage of Sensitive Information in Executable

中文：**在可执行文件中明确存储敏感信息**

### Description

* 应用程序将敏感信息以明文形式存储在可执行文件中。

### Extended Description

* 攻击者可以对二进制代码进行逆向工程以获取秘密数据当明文是纯ASCII时，这一点尤其容易。即使以非人类可读的方式对信息进行编码，某些技术也可以确定正在使用哪种编码，然后对信息进行解码。

## CWE - 319 Cleartext Transmission of Sensitive Information

中文：**明文传播敏感信息**

### Description

* 该软件以通信渠道的明文形式传输敏感或安全关键数据，这些数据可能被未经授权的参与者嗅探。

### Extended Description

* 在数据传输期间，攻击者可以“嗅探”许多通信信道。例如，任何有权访问网络接口的攻击者都可以嗅探网络流量。这大大降低了攻击者利用的难度。

## CWE - 321 Use of Hard-coded Cryptographic Key

中文：**使用硬编码密码密钥**

### Description

* 使用硬编码的加密密钥显着增加了可以恢复加密数据的可能性。

### Extended Description

* 没有详细描述

## CWE - 322 Key Exchange without Entity Authentication

中文：**没有实体身份验证的密钥交换**

### Description

* 该软件与演员进行密钥交换，而不验证该演员的身份。

### Extended Description

* 执行密钥交换将保持两个实体之间发送的信息的完整性，但这并不能保证实体是他们声称的实体。这可能会导致一系列“中间人”攻击。通常，这涉及受害客户端，该客户端联系冒充受信任服务器的恶意服务器。如果客户端跳过身份验证或忽略身份验证失败，则恶意服务器可以从用户请求身份验证信息。然后，恶意服务器可以使用此身份验证信息使用受害者的凭据登录受信任的服务器，嗅探受害者和受信任服务器之间的流量等。

## CWE - 323 Reusing a Nonce, Key Pair in Encryption

中文：**在加密中重用Nonce，密钥对**

### Description

* Nonce应该用于当前场合并且只能使用一次。

### Extended Description

* 没有详细描述

## CWE - 324 Use of a Key Past its Expiration Date

中文：**在过期日期之后使用密钥**

### Description

* 该产品使用加密密钥或密码超过其到期日期，通过增加针对该密钥的破解攻击的时间窗口，显着降低了其安全性。

### Extended Description

* 虽然密钥的到期并不一定确保它们被泄露，但是长时间使用的密钥具有降低的完整性概率是一个重要的问题。因此，在与其强度成比例的时间段内更换按键非常重要。

## CWE - 325 Missing Required Cryptographic Step

中文：**缺少必需的加密步骤**

### Description

* 该软件没有在加密算法中实现所需的步骤，导致加密比该算法所公布的更弱。

### Extended Description

* 加密实现应遵循精确定义它们的算法，否则加密可能比预期的要弱。

## CWE - 326 Inadequate Encryption Strength

中文：**加密强度不足**

### Description

* 该软件使用理论上合理的加密方案存储或传输敏感数据，但不足以达到所需的保护级别。

### Extended Description

* 弱加密方案可能会遭受暴力攻击，这些攻击有可能成功使用当前的攻击方法和资源。

## CWE - 327 Use of a Broken or Risky Cryptographic Algorithm

中文：**使用破碎或危险的密码算法**

### Description

* 使用破坏或有风险的加密算法是不必要的风险，可能导致敏感信息的暴露。

### Extended Description

* 使用非标准算法是危险的，因为确定的攻击者可能能够破坏算法并破坏任何受保护的数据。可能存在众所周知的技术来破坏算法。

## CWE - 328 Reversible One-Way Hash

中文：**可逆的单向哈希**

### Description

* 该产品使用散列算法生成哈希值，该哈希值可用于确定原始输入，或查找可生成相同哈希的输入，比蛮力技术更有效。

### Extended Description

* 当哈希用于需要单向属性保持的安全算法时，这种弱点尤其危险。例如，如果身份验证系统获取传入密码并生成哈希值，则将哈希值与其身份验证数据库中存储的另一个哈希值进行比较，然后创建冲突的能力可能允许攻击者提供生成的备用密码相同的目标哈希，绕过身份验证。

## CWE - 329 Not Using a Random IV with CBC Mode

中文：**不使用具有CBC模式的随机IV**

### Description

* 不使用具有密码块链接（CBC）模式的随机初始化向量（IV）导致算法易受字典攻击。

### Extended Description

* 没有详细描述

## CWE - 330 Use of Insufficiently Random Values

中文：**使用不充分的随机值**

### Description

* 软件可能在安全上下文中使用不完全随机的数字或值，这取决于不可预测的数字。

### Extended Description

* 当软件在需要不可预测性的上下文中生成可预测值时，攻击者可能猜测将生成的下一个值，并使用此猜测来模拟其他用户或访问敏感信息。

## CWE - 331 Insufficient Entropy

中文：**熵不足**

### Description

* 该软件使用产生不足熵的算法或方案，留下比其他更可能发生的模式或值集群。

### Extended Description

* 没有详细描述

## CWE - 332 Insufficient Entropy in PRNG

中文：**PRNG中的熵不足**

### Description

* 伪随机数发生器（PRNG）可用或使用的熵的缺乏可能是稳定性和安全性威胁。

### Extended Description

* 没有详细描述

## CWE - 333 Improper Handling of Insufficient Entropy in TRNG

中文：**TRNG中熵不足的处理不当**

### Description

* 真随机数发生器（TRNG）通常具有有限的熵源，因此可能失败或阻塞。

### Extended Description

* 可以生成真随机数的速率是有限的。重要的是，只有在安全需要时才使用它们。

## CWE - 334 Small Space of Random Values

中文：**随机值的小空间**

### Description

* 可能的随机值的数量小于产品所需的数量，使其更容易受到强力攻击。

### Extended Description

* 没有详细描述

## CWE - 335 Incorrect Usage of Seeds in Pseudo-Random Number Generator (PRNG)

中文：**伪随机数发生器（PRNG）中种子的使用不正确**

### Description

* 该软件使用伪随机数发生器（PRNG），不能正确管理种子。

### Extended Description

* 没有详细描述

## CWE - 336 Same Seed in Pseudo-Random Number Generator (PRNG)

中文：**伪随机数发生器（PRNG）中的相同种子**

### Description

* 伪随机数发生器（PRNG）在每次初始化产品时使用相同的种子。

### Extended Description

* 如果攻击者可以猜测（或知道）种子，则攻击者可能能够确定将从PRNG生成的随机数。

## CWE - 337 Predictable Seed in Pseudo-Random Number Generator (PRNG)

中文：**伪随机数发生器（PRNG）中的可预测种子**

### Description

* 伪随机数发生器（PRNG）从可预测的种子初始化，例如进程ID或系统时间。

### Extended Description

* 可预测种子的使用显着减少了攻击者需要测试的可能种子的数量，以便预测PRNG将生成哪些随机numnbers。

## CWE - 338 Use of Cryptographically Weak Pseudo-Random Number Generator (PRNG)

中文：**使用密码弱伪随机数发生器（PRNG）**

### Description

* 该产品在安全上下文中使用伪随机数生成器（PRNG），但PRNG的算法在加密方面不强。

### Extended Description

* 当在加密上下文中使用非加密PRNG时，它可以将加密暴露给某些类型的攻击。 通常，伪随机数发生器（PRNG）不是为加密而设计的。对于使用随机数的算法，有时候平庸的随机性来源是充分的或者更可取的。弱发电机通常需要较少的处理能力和/或不在系统上使用宝贵的有限熵源。虽然这些PRNG可能具有非常有用的功能，但这些相同的功能可用于打破加密。

## CWE - 339 Small Seed Space in PRNG

中文：**PRNG的小种子空间**

### Description

* PRNG使用相对较小的种子空间。

### Extended Description

* 没有详细描述

## CWE - 340 Predictability Problems

中文：**可预测性问题**

### Description

* 此类别中的弱点与生成数字或标识符的方案相关，这些数字或标识符比应用程序所需的更可预测。

### Extended Description

* 没有详细描述

## CWE - 341 Predictable from Observable State

中文：**从可观察国家可预测**

### Description

* 根据攻击者可以对系统或网络的状态（例如时间，进程ID等）进行的观察，可以预测数字或对象。

### Extended Description

* 没有详细描述

## CWE - 342 Predictable Exact Value from Previous Values

中文：**先前值的可预测精确值**

### Description

* 通过观察先前的值可以精确地预测精确值或随机数。

### Extended Description

* 没有详细描述

## CWE - 343 Predictable Value Range from Previous Values

中文：**先前值的可预测值范围**

### Description

* 该软件的随机数发生器产生一系列值，当观察到这些值时，可用于推断可能产生的下一个值的相对小范围的可能性。

### Extended Description

* 根据对先前值的观察，不应预测随机数发生器的输出。在某些情况下，攻击者无法预测接下来会产生的确切值，但可以显着缩小可能性。这减少了执行暴力攻击的努力量。例如，假设产品生成1到100之间的随机数，但它总是产生一个更大的值，直到达到100.如果生成器产生80，那么攻击者知道下一个值将介于81和100之间。在100种可能性中，攻击者只需要考虑20种。

## CWE - 344 Use of Invariant Value in Dynamically Changing Context

中文：**在动态变化的上下文中使用不变值**

### Description

* 该产品使用常量值，名称或引用，但此值可以（或应该）在不同的环境中变化。

### Extended Description

* 没有详细描述

## CWE - 345 Insufficient Verification of Data Authenticity

中文：**数据真实性验证不足**

### Description

* 该软件无法充分验证数据的来源或真实性，导致其接受无效数据。

### Extended Description

* 没有详细描述

## CWE - 346 Origin Validation Error

中文：**原点验证错误**

### Description

* 该软件未正确验证数据源或通信是否有效。

### Extended Description

* 没有详细描述

## CWE - 347 Improper Verification of Cryptographic Signature

中文：**密码签名的不正确验证**

### Description

* 该软件不会验证或错误地验证数据的加密签名。

### Extended Description

* 没有详细描述

## CWE - 348 Use of Less Trusted Source

中文：**使用较少可信源**

### Description

* 该软件具有相同数据或信息的两个不同来源，但它使用的源代码对验证的支持较少，信任度较低，或者抵御攻击的能力较弱。

### Extended Description

* 没有详细描述

## CWE - 349 Acceptance of Extraneous Untrusted Data With Trusted Data

中文：**使用可信数据接受外部不受信任的数据**

### Description

* 该软件在处理可信数据时，接受可信数据中包含的任何不受信任的数据，将不受信任的数据视为可信数据。

### Extended Description

* 没有详细描述

## CWE - 350 Reliance on Reverse DNS Resolution for a Security-Critical Action

中文：**依赖反向DNS解决方案来实现安全关键行动**

### Description

* 该软件对IP地址执行反向DNS解析以获取主机名并做出安全决策，但它无法正确确保IP地址与主机名真正关联。

### Extended Description

* 由于DNS名称很容易被欺骗或误报，并且软件可能难以检测可信DNS服务器是否已被泄露，因此DNS名称不构成有效的认证机制。 当软件对IP地址执行反向DNS解析时，如果攻击者控制服务器获取该IP地址，则攻击者可以使服务器返回任意主机名。因此，攻击者可能绕过身份验证，导致错误的主机名记录在日志文件中以隐藏活动或执行其他攻击。 攻击者可以通过（1）破坏DNS服务器并修改其记录（有时称为DNS缓存中毒）或（2）对与其IP地址关联的DNS服务器进行合法控制来欺骗DNS名称。

## CWE - 351 Insufficient Type Distinction

中文：**类型区别不足**

### Description

* 该软件无法以导致不安全行为的方式正确区分不同类型的元素。

### Extended Description

* 没有详细描述

## CWE - 352 Cross-Site Request Forgery (CSRF)

中文：**跨站请求伪造（CSRF）**

### Description

* Web应用程序没有或不能充分验证提交请求的用户是否有意提供了格式良好，有效，一致的请求。

### Extended Description

* 当Web服务器被设计为从客户端接收请求而没有任何机制来验证它是否被故意发送时，攻击者可能会欺骗客户端向Web服务器发出无意的请求，该请求将被视为真实的要求。这可以通过URL，图像加载，XMLHttpRequest等来完成，并且可能导致数据暴露或意外的代码执行。

## CWE - 353 Missing Support for Integrity Check

中文：**缺少对完整性检查的支持**

### Description

* 该软件使用传输协议，该协议不包括用于在传输期间验证数据完整性的机制，例如校验和。

### Extended Description

* 如果从协议中省略完整性检查值或“校验和”，则无法确定数据在传输中是否已损坏。协议中缺少校验和功能会删除可以使用的数据的第一个应用程序级别检查。端到端的检查理念指出，完整性检查应该在可以完全实现的最低级别执行。除了应用程序执行的进一步的健全性检查和输入验证之外，协议的校验和是最重要的校验和级别，因为它可以比任何先前级别更完整地执行并且考虑整个消息，而不是单个数据包。

## CWE - 354 Improper Validation of Integrity Check Value

中文：**对完整性检查值的不正确验证**

### Description

* 该软件不验证或错误地验证消息的完整性检查值或“校验和”。这可以防止它检测数据是否在传输中被修改或损坏。

### Extended Description

* 使用前对校验和的不正确验证会导致不必要的风险，可以轻松减轻。协议规范描述了用于计算校验和的算法。然后，实现计算并验证计算的校验和与接收的校验和匹配是一件简单的事情。对计算的校验和和接收的校验和的不正确验证可能导致更大的后果。

## CWE - 356 Product UI does not Warn User of Unsafe Actions

中文：**产品UI不会警告不安全操作的用户**

### Description

* 在代表该用户执行不安全操作之前，该软件的用户界面不会警告用户。这使攻击者更容易诱骗用户对其系统造成损害。

### Extended Description

* 软件系统应警告用户，如果用户继续进行，可能会发生潜在危险的操作。例如，如果用户从未知来源下载文件并尝试在其计算机上执行该文件，则应用程序的GUI可以指示该文件不安全。

## CWE - 357 Insufficient UI Warning of Dangerous Operations

中文：**危险操作的UI警告不足**

### Description

* 用户界面向用户提供有关危险或敏感操作的警告，但警告不足以引起注意。

### Extended Description

* 没有详细描述

## CWE - 358 Improperly Implemented Security Check for Standard

中文：**未正确实施标准安全检查**

### Description

* 该软件未实现或错误地实现由标准化算法，协议或技术的设计指定的一个或多个安全相关检查。

### Extended Description

* 没有详细描述

## CWE - 359 Exposure of Private Information ('Privacy Violation')

中文：**私人信息的曝光（'隐私违规'）**

### Description

* 该软件无法正确防止私人数据（如信用卡号）被（1）未明确授权访问数据或（2）未获得数据所针对的人的默许相关的。

### Extended Description

* 错误处理私人信息（例如客户密码或社会安全号码）可能会损害用户隐私，而且通常是非法的。私人信息的曝光并不一定会妨碍软件正常工作，事实上它可能是开发人员想要的，但对于与此私人信息相关的人来说，它仍然是不受欢迎的（或法律明确禁止）。 在以下情况下可能发生隐私

私人用户信息进入该程序。 数据将写入外部位置，例如控制台，文件系统或网络。

私人数据可以通过多种方式进入程序：

直接来自用户的密码或个人信息的形式 应用程序从数据库或其他数据存储访问 间接来自合作伙伴或其他第三方

某些类型的私人信息包括：

政府标识符，例如社会安全号码 联系信息，如家庭住址和电话号码 地理位置 - 用户所在的位置 工作经历 财务数据 - 例如信用卡号，工资，银行账户和债务 图片，视频或音频 行为模式 - 例如网上冲浪历史，执行某些活动时等。 与他人的关系（和关系类型） - 家人，朋友，联系人等。 通讯 - 电子邮件地址，私人电子邮件，短信，聊天记录等。 健康 - 医疗条件，保险状况，处方记录 凭证，例如密码，可用于访问其他信息。

这些信息中的一些可以被表征为PII（个人可识别信息），受保护的健康信息（PHI）等。私人信息的类别可以基于特定行业的预期用途或政策和实践而重叠或变化。 根据其所在地，所经营的业务类型以及其处理的任何私人数据的性质，组织可能需要遵守以下一项或多项联邦和州法规： - 安全港隐私框架[REF-340 ] - Gramm-Leach Bliley Act（GLBA）[REF-341] - 健康保险流通与责任法案（HIPAA）[REF-342] - 加州SB-1386 [REF-343]。 有时，未标记为私有的数据可能在不同的上下文中具有隐私含义。例如，学生识别号码通常不被视为私人，因为没有明确且公开可用的映射到个别学生的个人信息。但是，如果学校根据学生的社会安全号码生成识别号码，则识别号码应视为私密号码。 安全和隐私问题似乎经常相互竞争。从安全角度来看，应记录所有重要操作，以便以后可以识别任何异常活动。但是，当涉及私人数据时，这种做法实际上可能会产生风险。尽管有许多方法可以不安全地处理私人数据，但共同的风险源于错误的信任。程序员通常信任程序运行的操作环境，因此认为可以接受在文件系统，注册表或其他本地控制的资源中存储私有信息。但是，即使限制对某些资源的访问，也不能保证可以信任具有访问权限的个人。

## CWE - 360 Trust of System Event Data

中文：**信任系统事件数据**

### Description

* 基于事件位置的安全性是不安全的，可能是欺骗性的。

### Extended Description

* 事件是消息传递系统，其可以向监听事件的程序提供控制数据。事件通常没有任何类型的身份验证框架，以允许从受信任的来源验证它们。 Windows中的任何应用程序都可以在给定桌面上向同一桌面上的任何窗口发送消息。这些消息没有身份验证框架。因此，如果进程未检查这些消息的有效性和安全性，则可以使用任何消息来操作桌面上的任何进程。

## CWE - 362 Concurrent Execution using Shared Resource with Improper Synchronization ('Race Condition')

中文：**使用具有不正确同步的共享资源并发执行（'竞争条件'）**

### Description

* 该程序包含可与其他代码并发运行的代码序列，并且代码序列需要对共享资源的临时，独占访问，但存在时序窗口，其中共享资源可由另一个并发操作的代码序列修改。

### Extended Description

* 当预期的同步在安全关键代码中时，这可能具有安全隐患，例如记录用户是否经过身份验证或修改不应受到局外人影响的重要状态信息。 竞争条件发生在并发环境中，并且实际上是代码序列的属性。根据上下文，代码序列可以是函数调用，少量指令，一系列程序调用等形式。 竞争条件违反了这些与之密切相关的属性：

排他性 - 给予代码序列对共享资源的独占访问权，即，在原始序列完成执行之前，没有其他代码序列可以修改共享资源的属性。 原子性 - 代码序列是行为原子的，即，没有其他线程或进程可以同时针对相同资源执行相同的指令序列（或子集）。

当“干扰代码序列”仍然可以访问共享资源时，存在竞争条件，违反了排他性。程序员可以假设某些代码序列执行得太快而不受干扰代码序列的影响;如果不是，那就违反了原子性。例如，单个“x ++”语句在代码层可能看起来是原子的，但它在指令层实际上是非原子的，因为它涉及读取（x的原始值），然后是计算（x + 1） ），然后写入（将结果保存到x）。 干扰代码序列可以是“可信任的”或“不可信的”。可信的干扰码序列发生在程序内;它不能被攻击者修改，只能间接调用。不受信任的干扰代码序列可以由攻击者直接创作，通常它位于易受攻击的程序外部。

## CWE - 363 Race Condition Enabling Link Following

中文：**竞争条件启用链接跟随**

### Description

* 软件在访问之前检查文件或目录的状态，这会产生竞争条件，在执行访问之前可以用链接替换文件，从而导致软件访问错误的文件。

### Extended Description

* 虽然开发人员可能期望在检查时间和使用时间之间存在非常窄的时间窗口，但仍存在竞争条件。攻击者可能导致软件速度变慢（例如，内存消耗），导致时间窗口变大。或者，在某些情况下，攻击者可以通过执行大量攻击来赢得比赛。

## CWE - 364 Signal Handler Race Condition

中文：**信号处理器竞争条件**

### Description

* 该软件使用引入竞争条件的信号处理程序。

### Extended Description

* 竞争条件经常发生在信号处理程序中，因为信号处理程序支持异步动作。这些种族条件有各种根本原因和症状。攻击者可能能够利用信号处理程序竞争条件导致软件状态被破坏，可能导致拒绝服务甚至代码执行。 当信号处理程序中发生非重入函数或状态敏感操作时，会发生这些问题，可以随时调用它们。这些行为可能违反被中断的“常规”代码或可能被调用的其他信号处理程序所做出的假设。如果在不合适的时刻调用这些函数 - 例如在非重入函数已经运行时 - 可能会发生内存损坏，这可能会被代码执行利用。通常发现的另一种信号竞争条件发生在信号处理程序中调用free时，导致双重释放，从而导致write-what-where条件。即使给定指针在释放后设置为NULL，在释放内存和指针设置为NULL之间仍然存在争用条件。如果为多个信号设置了相同的信号处理程序，则这尤其成问题 - 因为这意味着可以重新输入信号处理程序本身。 有几个与信号处理程序相关的已知行为已经收到“信号处理程序竞争条件”的标签：

信号处理程序和“常规”代码都可访问的共享状态（例如全局数据或静态变量） 信号处理程序和其他信号处理程序之间的共享状态 在信号处理程序中使用非重入功能 - 这通常意味着正在使用共享状态。例如，malloc（）和free（）是不可重入的，因为它们可能使用全局或静态数据结构来管理内存，并且它们被无辜的看似函数间接使用，例如syslog（）;这些函数可能被用于内存损坏，可能还有代码执行。 将相同的信号处理函数与多个信号相关联 - 这可能意味着共享状态，因为访问了相同的代码和资源。例如，这可能是双重免费和使用后免费弱点的来源。 使用setjmp和longjmp，或阻止信号处理程序将控制权返回到原始功能的其他机制 虽然从技术上讲不是竞争条件，但是一些信号处理程序被设计为最多被调用一次，并且被调用不止一次会引入安全问题，即使没有任何并发调用信号处理程序。这可能是双重免费和使用后免费弱点的来源。

信号处理程序漏洞通常基于缺少特定保护机制进行分类，尽管CWE不鼓励这种分类，因为程序员通常可以选择几种不同的机制来解决这些弱点。这种保护机制可以保留对共享资源的访问权限，以及相关代码的行为原子性：

避免共享状态 在信号处理程序中使用同步 在常规代码中使用同步 禁用或屏蔽其他信号，提供原子性（有效确保排他性）

## CWE - 365 Race Condition in Switch

中文：**交换机中的竞争条件**

### Description

* 该代码包含一个switch语句，其中可以在交换机仍在执行时修改切换变量，从而导致意外行为。

### Extended Description

* 这个问题在涉及直通式案例陈述的switch语句中尤为重要 - 即那些不以break结尾的语句。如果交换机测试的变量在执行过程中发生变化，这可能会改变交换机的预期逻辑，以至于它将进程置于矛盾状态，在某些情况下甚至可能导致内存损坏。

## CWE - 366 Race Condition within a Thread

中文：**线程中的竞争条件**

### Description

* 如果两个执行线程同时使用资源，则存在可能在无效时使用资源的可能性，从而使得执行状态未定义。

### Extended Description

* 没有详细描述

## CWE - 367 Time-of-check Time-of-use (TOCTOU) Race Condition

中文：**检查时间（TOCTOU）竞赛条件**

### Description

* 软件在使用该资源之前检查资源的状态，但资源的状态可以在检查和使用之间以一种使检查结果无效的方式发生变化。这可能导致软件在资源处于意外状态时执行无效操作。

### Extended Description

* 当攻击者可以在检查和使用之间影响资源状态时，这种弱点可能与安全相关。这可能发生在共享资源（如文件，内存，甚至是多线程程序中的变量）中。

## CWE - 368 Context Switching Race Condition

中文：**上下文切换竞争条件**

### Description

* 产品执行一系列非原子操作以在跨越特权或其他安全边界的上下文之间切换，但竞争条件允许攻击者在切换期间修改或歪曲产品的行为。

### Extended Description

* 这通常出现在Web浏览器漏洞中，攻击者可以在浏览器从受信任域转换到不受信任域时执行某些操作，反之亦然，浏览器使用信任级别和资源在一个域上执行操作。其他域名。

## CWE - 369 Divide By Zero

中文：**除以零**

### Description

* 产品将值除以零。

### Extended Description

* 当向产品提供意外值时，或者如果发生未正确检测到的错误，通常会发生此弱点。它经常发生在涉及物理尺寸的计算中，例如尺寸，长度，宽度和高度。

## CWE - 370 Missing Check for Certificate Revocation after Initial Check

中文：**初始检查后缺少检查证书撤销**

### Description

* 软件在初始撤销检查后不检查证书的撤销状态，这可能导致软件即使在稍后撤销证书后也执行特权操作。

### Extended Description

* 如果在每个需要权限的操作之前未检查证书的撤销状态，则系统可能会受到竞争条件的影响。如果在初始检查后撤销证书，则与撤销证书的所有者一起执行的所有后续操作都将失去证书保证的所有权益。实际上，几乎可以肯定使用撤销证书表示恶意活动。

## CWE - 372 Incomplete Internal State Distinction

中文：**不完整的内部状态区别**

### Description

* 软件没有正确地确定它所处的状态，导致它假设它处于状态X，而实际上它处于状态Y，导致它以安全相关的方式执行不正确的操作。

### Extended Description

* 没有详细描述

## CWE - 373 DEPRECATED: State Synchronization Error

中文：**DEPRECATED：状态同步错误**

### Description

* 此条目已弃用，因为它与竞争条件（CWE-362）和不正确同步（CWE-662）相同的概念重叠。

### Extended Description

* 没有详细描述

## CWE - 374 Passing Mutable Objects to an Untrusted Method

中文：**将可变对象传递给不受信任的方法**

### Description

* 程序将非克隆的可变数据作为参数发送给方法或函数。

### Extended Description

* 已调用的函数或方法可以更改或删除可变数据。这可能违反了调用函数对其状态所做的假设。在通过引用可变数据调用未知代码的情况下，此外部代码可以对发送的数据进行更改。如果先前未克隆此数据，则修改后的数据在执行上下文中可能无效。

## CWE - 375 Returning a Mutable Object to an Untrusted Caller

中文：**将可变对象返回给不受信任的调用者**

### Description

* 将非克隆的可变数据作为返回值发送可能导致该数据被调用函数更改或删除。

### Extended Description

* 在函数返回对可变数据的引用的情况下，调用该函数的外部代码可能会对发送的数据进行更改。如果此数据先前未被克隆，则该类将使用可能违反其内部状态假设的修改数据。

## CWE - 377 Insecure Temporary File

中文：**不安全的临时文件**

### Description

* 创建和使用不安全的临时文件会使应用程序和系统数据容易受到攻击。

### Extended Description

* 没有详细描述

## CWE - 378 Creation of Temporary File With Insecure Permissions

中文：**使用不安全的权限创建临时文件**

### Description

* 在没有适当措施或控制的情况下打开临时文件可能会使文件，其内容以及任何易受攻击的功能受到攻击。

### Extended Description

* 没有详细描述

## CWE - 379 Creation of Temporary File in Directory with Incorrect Permissions

中文：**在具有不正确权限的目录中创建临时文件**

### Description

* 该软件在目录中创建一个临时文件，其权限允许非预期的actor确定文件是否存在或以其他方式访问该文件。

### Extended Description

* 在某些操作系统上，对于具有足够权限访问该目录的任何用户，临时文件存在的事实可能是显而易见的。由于文件可见，因此可以知道使用临时文件的应用程序。如果有权访问系统上的进程列表，则攻击者已获得有关用户当时正在执行的操作的信息。通过将此与用户运行的应用程序相关联，攻击者可能会发现用户的操作是什么。由此可以破坏更高级别的安全性。

## CWE - 382 J2EE Bad Practices: Use of System.exit()

中文：**J2EE不良做法：使用System.exit（）**

### Description

* J2EE应用程序使用System.exit（），它也会关闭其容器。

### Extended Description

* Web应用程序尝试关闭应用程序容器绝不是一个好主意。访问可以关闭应用程序的功能是拒绝服务（DoS）攻击的途径。

## CWE - 383 J2EE Bad Practices: Direct Use of Threads

中文：**J2EE不良做法：直接使用线程**

### Description

* 在某些情况下，Web应用程序中的线程管理是被禁止的，并且总是非常容易出错。

### Extended Description

* 在某些情况下，J2EE标准禁止Web应用程序中的线程管理，并且总是非常容易出错。管理线程很困难，并且可能以不可预测的方式干扰应用程序容器的行为。即使不干扰容器，线程管理通常也会导致难以检测和诊断的错误，如死锁，竞争条件和其他同步错误。

## CWE - 384 Session Fixation

中文：**会话固定**

### Description

* 在不使任何现有会话标识符无效的情况下认证用户或以其他方式建立新用户会话使攻击者有机会窃取经认证的会话。

### Extended Description

* 在以下情况下通常会出现这种情况：

1. Web应用程序在不首先使现有会话无效的情况下对用户进行认证，从而继续使用已经与该用户相关联的会话。 2.攻击者能够强制用户使用已知的会话标识符，以便一旦用户进行身份验证，攻击者就可以访问经过身份验证的会话。 3.应用程序或容器使用可预测的会话标识符。在会话固定漏洞的一般利用中，攻击者在Web应用程序上创建新会话并记录关联的会话标识符。然后，攻击者使用该会话标识符使受害者与服务器关联并可能进行身份验证，从而使攻击者通过活动会话访问用户的帐户。

## CWE - 385 Covert Timing Channel

中文：**隐蔽时间通道**

### Description

* 隐蔽定时信道通过随时间调整系统行为的某些方面来传达信息，使得接收信息的程序可以观察系统行为并推断受保护的信息。

### Extended Description

* 在某些情况下，知道何时在各方之间传输数据可以向恶意用户提供特权信息。此外，从外部监视操作的时间可能会泄露敏感数据。例如，如果执行操作所花费的时间根据状态而变化，则加密操作可以暴露其内部状态。 隐蔽信道通常被分类为存储或定时信道。隐蔽定时信道的一些示例是系统的寻呼速率，特定事务执行所需的时间以及获得对共享总线的访问所花费的时间。

## CWE - 386 Symbolic Name not Mapping to Correct Object

中文：**符号名称不映射到正确的对象**

### Description

* 使用对象的常量符号引用，即使引用可以随时间解析为不同的对象。

### Extended Description

* 没有详细描述

## CWE - 390 Detection of Error Condition Without Action

中文：**无动作的错误条件检测**

### Description

* 该软件检测到特定错误，但不采取任何操作来处理错误。

### Extended Description

* 没有详细描述

## CWE - 391 Unchecked Error Condition

中文：**未选中的错误条件**

### Description

* 忽略异常和其他错误条件可能允许攻击者引起意外行为而不被注意。

### Extended Description

* 没有详细描述

## CWE - 392 Missing Report of Error Condition

中文：**错误报告错误情况**

### Description

* 软件遇到错误但未提供状态代码或返回值以指示发生了错误。

### Extended Description

* 没有详细描述

## CWE - 393 Return of Wrong Status Code

中文：**返回错误的状态代码**

### Description

* 函数或操作返回不正确的返回值或状态代码，但不会指示错误，但会导致产品根据错误的结果修改其行为。

### Extended Description

* 这可能导致不可预测的行为。如果该函数用于制定安全关键决策或提供安全关键信息，则错误的状态代码可能导致软件认为某个操作是安全的，即使它不是。

## CWE - 394 Unexpected Status Code or Return Value

中文：**意外的状态代码或返回值**

### Description

* 软件无法正确检查函数或操作何时返回对函数合法的值，但软件不期望该值。

### Extended Description

* 没有详细描述

## CWE - 395 Use of NullPointerException Catch to Detect NULL Pointer Dereference

中文：**使用NullPointerException Catch来检测空指针解除引用**

### Description

* 捕获NullPointerException不应用作编程检查的替代方法，以防止取消引用空指针。

### Extended Description

* 程序员通常在三种情况下捕获NullPointerException：

该程序包含空指针取消引用。捕获生成的异常比修复底层问题更容易。 程序显式抛出NullPointerException来发出错误信号。 该代码是测试工具的一部分，为测试中的类提供意外输入。

在这三种情况中，只有最后一种是可以接受的。

## CWE - 396 Declaration of Catch for Generic Exception

中文：**通用例外的捕获声明**

### Description

* 捕获过于广泛的异常会促使复杂的错误处理代码更容易包含安全漏洞。

### Extended Description

* 多个catch块可能会变得丑陋和重复，但是通过捕获像Exception这样的高级类来“压缩”catch块会掩盖需要特殊处理的异常或者在程序中此时不应该捕获的异常。捕获过于宽泛的异常基本上会破坏Java类型异常的目的，如果程序增长并开始抛出新类型的异常，则会变得特别危险。新的异常类型不会受到任何关注。

## CWE - 397 Declaration of Throws for Generic Exception

中文：**通用例外的抛出声明**

### Description

* 抛出过于广泛的异常会促使复杂的错误处理代码更容易包含安全漏洞。

### Extended Description

* 声明抛出Exception或Throwable的方法会使调用者难以执行正确的错误处理和错误恢复。例如，Java的异常机制被设置为使调用者能够轻松地预测可能出错的内容并编写代码来处理每个特定的异常情况。声明方法抛出一般形式的异常会使该系统失效。

## CWE - 400 Uncontrolled Resource Consumption

中文：**不受控制的资源消耗**

### Description

* 该软件没有适当地控制有限资源的分配和维护，从而使行动者能够影响所消耗的资源量，最终导致可用资源的耗尽。

### Extended Description

* 有限的资源包括内存，文件系统存储，数据库连接池条目和CPU。如果攻击者可以触发这些有限资源的分配，但资源的数量或大小未受控制，则攻击者可能会导致拒绝服务，从而消耗所有可用资源。这将阻止有效用户访问该软件，并且可能对周围环境产生影响。例如，针对应用程序的内存耗尽攻击可能会降低应用程序及其主机操作系统的速度。 至少有三种不同的情况通常会导致资源耗尽：

对分配的资源数量缺乏限制 在到达关闭阶段之前丢失对资源的所有引用 处理后没有关闭/返回资源

资源耗尽问题通常是由于以下情况的错误实施导致的：

错误条件和其他特殊情况。 混淆程序的哪个部分负责释放资源。

## CWE - 401 Improper Release of Memory Before Removing Last Reference

中文：**删除最后一个引用之前内存释放不正确**

### Description

* 软件在使用后无法充分跟踪和释放已分配的内存，这会慢慢消耗剩余的内存。

### Extended Description

* 这通常是由于错误处理格式错误的数据或意外中断的会话而触发的。

## CWE - 402 Transmission of Private Resources into a New Sphere ('Resource Leak')

中文：**私人资源进入新领域（'资源泄漏'）**

### Description

* 当这些资源仅供软件访问时，该软件可为不受信任的各方提供资源。

### Extended Description

* 没有详细描述

## CWE - 403 Exposure of File Descriptor to Unintended Control Sphere ('File Descriptor Leak')

中文：**文件描述符暴露于非预期控制球（'文件描述符泄漏'）**

### Description

* 在调用子进程之前，进程不会关闭敏感文件描述符，这允许子进程使用这些描述符执行未经授权的I / O操作。

### Extended Description

* 分叉或执行新进程时，子进程继承任何打开的文件描述符。当子进程的权限少于父进程时，如果子进程可以访问文件描述符但没有访问关联文件的权限，则可能会引入漏洞。

## CWE - 404 Improper Resource Shutdown or Release

中文：**资源关闭或发布不当**

### Description

* 程序在可用于重用之前不会释放或错误地释放资源。

### Extended Description

* 创建或分配资源时，开发人员负责正确释放资源以及计算所有可能的到期或失效路径，例如设定的时间段或撤销。

## CWE - 405 Asymmetric Resource Consumption (Amplification)

中文：**不对称资源消耗（扩增）**

### Description

* 不能适当监视或控制资源消耗的软件可能导致系统性能下降。

### Extended Description

* 如果软件允许恶意用户或攻击者消耗比访问级别允许的更多资源，则这种情况会被放大。利用这种弱点可能导致不对称的资源消耗，有助于对系统或网络的放大攻击。

## CWE - 406 Insufficient Control of Network Message Volume (Network Amplification)

中文：**网络消息量控制不足（网络扩容）**

### Description

* 该软件不能充分监视或控制传输的网络流量，因此参与者可以使软件传输的流量超过该参与者应允许的流量。

### Extended Description

* 在缺乏限制非对称资源消耗的策略的情况下，应用程序或系统无法区分合法传输和旨在用作对目标系统的放大攻击的流量。系统通常可以配置为根据客户端的来源或访问级别限制代表客户端发送的流量。这通常在资源分配策略中定义。在没有跟踪传输的机制的情况下，系统或应用程序可以容易地被滥用以传输比应该允许的请求或客户端更不对称更大的流量。

## CWE - 407 Algorithmic Complexity

中文：**算法复杂性**

### Description

* 产品中的算法具有低效的最坏情况计算复杂度，可能对系统性能有害并且可以由攻击者触发，通常使用精心设计的操作来确保达到最坏的情况。

### Extended Description

* 没有详细描述

## CWE - 408 Incorrect Behavior Order: Early Amplification

中文：**行为顺序不正确：早期扩增**

### Description

* 该软件允许实体在进行认证或授权之前执行合法但昂贵的操作。

### Extended Description

* 没有详细描述

## CWE - 409 Improper Handling of Highly Compressed Data (Data Amplification)

中文：**高压缩数据处理不当（数据放大）**

### Description

* 该软件无法处理或错误处理具有非常高压缩比的压缩输入，从而产生大量输出。

### Extended Description

* 数据放大的一个例子是“解压缩炸弹”，一种小型ZIP文件，在解压缩时可以产生大量数据。

## CWE - 410 Insufficient Resource Pool

中文：**资源池不足**

### Description

* 软件的资源池不足以处理峰值需求，这允许攻击者通过使用（相对）大量的资源请求来阻止其他人访问资源。

### Extended Description

* 通常结果是连接或会话的“泛滥”。

## CWE - 412 Unrestricted Externally Accessible Lock

中文：**无限制的外部可访问锁**

### Description

* 软件正确地检查是否存在锁，但是锁可以由外部控制或受到在预期控制范围之外的行为者的影响。

### Extended Description

* 这可以防止软件对相关资源起作用或执行由锁的存在控制的其他行为。相关锁可能包括独占锁或互斥锁，或修改被视为锁的共享资源。如果锁定可以无限期保留，那么拒绝服务可能是永久性的。

## CWE - 413 Improper Resource Locking

中文：**资源锁定不当**

### Description

* 当软件必须具有对资源的独占访问权限时，软件不会锁定或未正确锁定资源。

### Extended Description

* 当资源未正确锁定时，攻击者可以在软件操作资源时修改资源。这可能违反了软件假设资源不会改变，可能导致意外行为的假设。

## CWE - 414 Missing Lock Check

中文：**缺少锁定检查**

### Description

* 在对资源执行敏感操作之前，产品不会检查是否存在锁定。

### Extended Description

* 没有详细描述

## CWE - 415 Double Free

中文：**双免费**

### Description

* 该产品在同一内存地址上调用free（）两次，可能导致意外内存位置的修改。

### Extended Description

* 当程序使用相同的参数调用free（）两次时，程序的内存管理数据结构将被破坏。这种损坏可能导致程序崩溃，或者在某些情况下导致两个以后的malloc（）调用返回相同的指针。如果malloc（）返回相同的值两次，并且程序稍后让攻击者控制写入这个双重分配的内存的数据，则该程序容易受到缓冲区溢出攻击。

## CWE - 416 Use After Free

中文：**免费使用**

### Description

* 在释放内存后引用内存可能导致程序崩溃，使用意外值或执行代码。

### Extended Description

* 使用先前释放的内存可能会产生任何数量的不利后果，从有效数据的损坏到任意代码的执行，具体取决于缺陷的实例化和时间安排。可能发生数据损坏的最简单方法是系统重用已释放的内存。释放后使用错误有两个常见且有时重叠的原因：

错误条件和其他特殊情况。 混淆程序的哪个部分负责释放内存。

在这种情况下，有问题的内存在释放后的某个时刻有效地分配给另一个指针。再次使用指向释放内存的原始指针，并指向新分配中的某个位置。随着数据的改变，它会破坏有效使用的内存;这会导致过程中出现未定义的行为。 如果新分配的数据有机会保存类，例如在C ++中，各种函数指针可能分散在堆数据中。如果用有效shellcode的地址覆盖这些函数指针之一，则可以实现任意代码的执行。

## CWE - 419 Unprotected Primary Channel

中文：**不受保护的主要频道**

### Description

* 该软件使用主要通道进行管理或限制功能，但它不能正确保护通道。

### Extended Description

* 没有详细描述

## CWE - 420 Unprotected Alternate Channel

中文：**不受保护的替代频道**

### Description

* 该软件保护主要通道，但它不对备用通道使用相同级别的保护。

### Extended Description

* 没有详细描述

## CWE - 421 Race Condition During Access to Alternate Channel

中文：**访问备用频道期间的竞争条件**

### Description

* 该产品打开一个备用频道与授权用户进行通信，但其他演员可以访问该频道。

### Extended Description

* 这会创建一个竞争条件，允许攻击者在授权用户之前访问该通道。

## CWE - 422 Unprotected Windows Messaging Channel ('Shatter')

中文：**不受保护的Windows消息传递通道（'Shatter'）**

### Description

* 在以提升的权限运行时，软件无法正确验证Windows Messaging System中的消息来源，从而创建了一个备用通道，攻击者可以通过该通道直接向产品发送消息。

### Extended Description

* 没有详细描述

## CWE - 423 DEPRECATED (Duplicate): Proxied Trusted Channel

中文：**弃用（重复）：代理可信通道**

### Description

* 此条目已被弃用，因为它与CWE-441重复。所有内容均已转移至CWE-441。

### Extended Description

* 没有详细描述

## CWE - 424 Improper Protection of Alternate Path

中文：**替代路径的保护不当**

### Description

* 该产品不足以保护用户可以用来访问受限功能或资源的所有可能路径。

### Extended Description

* 没有详细描述

## CWE - 425 Direct Request ('Forced Browsing')

中文：**直接请求（'强制浏览'）**

### Description

* Web应用程序未对所有受限制的URL，脚本或文件充分执行适当的授权。

### Extended Description

* 易受直接请求攻击影响的Web应用程序通常会假设这样的资源只能通过给定的导航路径到达，因此只能在路径中的某些点应用授权。

## CWE - 426 Untrusted Search Path

中文：**不受信任的搜索路径**

### Description

* 应用程序使用外部提供的搜索路径搜索关键资源，该搜索路径可以指向不在应用程序直接控制之下的资源。

### Extended Description

* 这可能允许攻击者执行自己的程序，访问未经授权的数据文件或以意外方式修改配置。如果应用程序使用搜索路径来定位关键资源（如程序），则攻击者可以修改该搜索路径以指向恶意程序，然后目标应用程序将执行该程序。问题扩展到应用程序信任的任何类型的关键资源。 一些最常见的不受信任的搜索路径变体是：

在各种基于UNIX和Linux的系统中，可以查询PATH环境变量以定位可执行程序，并且可以使用LD\_PRELOAD来定位单独的库。 在各种基于Microsoft的系统中，如果在搜索顺序中较早出现的其他路径中找不到DLL，则查询PATH环境变量以定位DLL。

## CWE - 427 Uncontrolled Search Path Element

中文：**不受控制的搜索路径元素**

### Description

* 该产品使用固定或受控的搜索路径来查找资源，但该路径中的一个或多个位置可能受非预期参与者的控制。

### Extended Description

* 虽然这种弱点可能出现在任何类型的资源中，但是当产品使用目录搜索路径查找可执行文件或代码库时，它经常被引入，但路径包含可由攻击者修改的目录，例如“/ tmp”或当前的工作目录。 在基于Windows的系统中，当使用不包含完全限定路径的DLL名称调用LoadLibrary或LoadLibraryEx函数时，该函数遵循包含两个可能不受控制的路径元素的搜索顺序：

加载程序的目录 当前的工作目录。

在某些情况下，攻击可以远程进行，例如使用SMB或WebDAV网络共享时。 在一些基于Unix的系统中，可能会创建一个包含空元素的PATH，例如通过将空变量拼接到PATH中。此空元素可以解释为等效于当前工作目录，该目录可能是不受信任的搜索元素。

## CWE - 428 Unquoted Search Path or Element

中文：**不带引号的搜索路径或元素**

### Description

* 该产品使用包含不带引号的元素的搜索路径，其中元素包含空格或其他分隔符。这可能导致产品访问父路径中的资源。

### Extended Description

* 如果恶意个人有权访问文件系统，则可以通过插入“C：\ Program.exe”这样的文件来提升权限，以便由使用WinExec的特权程序运行。

## CWE - 430 Deployment of Wrong Handler

中文：**部署错误的处理程序**

### Description

* 分配了错误的“处理程序”来处理对象。

### Extended Description

* 部署错误处理程序的一个示例是调用servlet来显示.JSP文件的源代码，或者自动“确定”对象的类型，即使它与明确指定的类型相矛盾。

## CWE - 431 Missing Handler

中文：**缺少处理程序**

### Description

* 处理程序不可用或未实现。

### Extended Description

* 当抛出异常并且未捕获异常时，该过程放弃了决定给定故障或事件是否值得执行更改的机会。

## CWE - 432 Dangerous Signal Handler not Disabled During Sensitive Operations

中文：**敏感操作期间未禁用危险信号处理程序**

### Description

* 应用程序使用与其他信号处理程序共享状态的信号处理程序，但在原始信号处理程序仍在运行时，它无法正确屏蔽或阻止调用这些信号处理程序。

### Extended Description

* 在执行信号处理程序期间，当发送不同的信号时，它可以被另一个处理程序中断。如果两个处理程序共享状态 - 例如全局变量 - 那么攻击者可以通过在第一个处理程序完成执行之前发送另一个信号来破坏状态。

## CWE - 433 Unparsed Raw Web Content Delivery

中文：**Unparsed Raw Web Content Delivery**

### Description

* 该软件将原始内容或支持代码存储在Web文档根目录下，其扩展名不是由服务器专门处理的。

### Extended Description

* 如果代码存储在扩展名为“.inc”或“.pl”的文件中，并且Web服务器没有该扩展名的处理程序，则服务器可能会将该文件的内容直接发送给请求者没有预期的预处理。当该文件包含敏感信息（如数据库凭据）时，这可能允许攻击者破坏应用程序或关联的组件。

## CWE - 434 Unrestricted Upload of File with Dangerous Type

中文：**无限制上传危险类型的文件**

### Description

* 该软件允许攻击者上传或传输可在产品环境中自动处理的危险类型的文件。

### Extended Description

* 没有详细描述

## CWE - 435 Improper Interaction Between Multiple Correctly-Behaving Entities

中文：**多个正确行为实体之间的不正确交互**

### Description

* 当两个实体在彼此独立运行时具有正确的行为时会发生交互错误，但是当它们作为组件集成到更大的系统或进程中时，它们会引入可能导致结果弱点的错误行为。

### Extended Description

* 当系统或流程组合多个独立组件时，这通常会在系统级别产生新的紧急行为。但是，如果这些组成部分之间的相互作用没有得到充分考虑，那么一些紧急行为可能是不正确的，甚至是不安全的。

## CWE - 436 Interpretation Conflict

中文：**解释冲突**

### Description

* 产品A处理与产品B不同的输入或步骤，这导致A根据其对B状态的感知执行不正确的操作。

### Extended Description

* 这通常存在于代理，防火墙，防病毒软件和其他中间设备中，这些设备根据客户端或服务器的行为方式监控，允许，拒绝或修改流量。

## CWE - 437 Incomplete Model of Endpoint Features

中文：**不完整的端点特征模型**

### Description

* 产品充当两个或多个端点之间的中介或监视器，但它没有端点的功能，行为或状态的完整模型，可能导致产品基于此不完整的模型执行不正确的操作。

### Extended Description

* 没有详细描述

## CWE - 439 Behavioral Change in New Version or Environment

中文：**新版本或环境中的行为改变**

### Description

* A的行为或功能随着新版本的A或新环境而变化，而新环境由B未知（或可管理）。

### Extended Description

* 没有详细描述

## CWE - 440 Expected Behavior Violation

中文：**预期的行为违规**

### Description

* 产品使用的功能，API或功能的行为与产品预期的不同。

### Extended Description

* 没有详细描述

## CWE - 441 Unintended Proxy or Intermediary ('Confused Deputy')

中文：**非预期的代理人或中介人（“困惑的代理人”）**

### Description

* 软件从上游组件接收请求，消息或指令，但在将请求转发给软件控制范围之外的外部参与者之前，软件不能充分保留请求的原始源。这导致软件看起来是请求的来源，使其充当上游组件和外部参与者之间的代理或其他中介。

### Extended Description

* 如果攻击者无法直接联系目标，但软件可以访问目标，则攻击者可以向软件发送请求并将其从目标转发。该请求似乎来自软件系统，而不是攻击者的系统。因此，攻击者可以绕过访问控制（例如防火墙）或隐藏恶意请求的来源，因为请求不会直接来自攻击者。 由于代理功能和邮件转发通常用于合法目的，因此在以下情况下此问题仅会成为漏洞：

软件以不同的权限运行或在不同的系统上运行，或者具有与上游组件不同的访问级别; 防止攻击者直接向目标发出请求;和 攻击者可以创建代理未明确打算代表请求者转发的请求。此类请求可能指向意外的主机名，端口号或服务。或者，请求可能会发送到允许的服务，但请求可能包含不允许的指令，命令或资源。

## CWE - 443 DEPRECATED (Duplicate): HTTP response splitting

中文：**DEPRECATED（Duplicate）：HTTP响应拆分**

### Description

* 这个弱点可以在CWE-113找到。

### Extended Description

* 没有详细描述

## CWE - 444 Inconsistent Interpretation of HTTP Requests ('HTTP Request Smuggling')

中文：**HTTP请求的不一致解释（'HTTP请求走私'）**

### Description

* 当用户和Web服务器（例如代理或防火墙）之间的数据流中的一个或多个实体解释格式错误或异常HTTP请求时，它们可能会被解释为不一致，从而允许攻击者将请求“偷偷”到一个设备没有其他设备意识到它。

### Extended Description

* 没有详细描述

## CWE - 446 UI Discrepancy for Security Feature

中文：**安全功能的UI差异**

### Description

* 用户界面未正确启用或配置安全功能，但界面提供的反馈使用户相信该功能处于安全状态。

### Extended Description

* 当用户界面没有正确反映用户询问的内容时，它可能会导致用户产生错误的安全感。例如，用户可能会选中一个框以启用安全选项以启用加密通信，但该软件实际上并未启用加密。或者，用户可能提供“限制所有”访问控制规则，但该软件仅实现“限制某些”。

## CWE - 447 Unimplemented or Unsupported Feature in UI

中文：**UI中未实现或不支持的功能**

### Description

* 似乎支持用于安全功能的UI功能，并向用户提供反馈，表明它受支持，但未实现基础功能。

### Extended Description

* 没有详细描述

## CWE - 448 Obsolete Feature in UI

中文：**UI中的过时功能**

### Description

* UI功能已过时，产品不会警告用户。

### Extended Description

* 没有详细描述

## CWE - 449 The UI Performs the Wrong Action

中文：**UI执行错误的操作**

### Description

* UI针对用户的请求执行错误的操作。

### Extended Description

* 没有详细描述

## CWE - 450 Multiple Interpretations of UI Input

中文：**UI输入的多重解释**

### Description

* UI具有用户输入的多种解释，但在选择较不安全的解释时不会提示用户。

### Extended Description

* 没有详细描述

## CWE - 451 User Interface (UI) Misrepresentation of Critical Information

中文：**用户界面（UI）错误表示关键信息**

### Description

* 用户界面（UI）不能正确地向用户表示关键信息，从而使信息（或其来源）变得模糊或欺骗。这通常是网络钓鱼攻击的一个组成部分。

### Extended Description

* 如果攻击者可能导致UI显示错误数据，或者说服用户显示看似来自可信来源的信息，则攻击者可能欺骗用户执行错误操作。这通常是网络钓鱼攻击的一个组成部分，但存在其他类型的问题。例如，如果UI用于监视系统或网络的安全状态，则省略或模糊重要指示符可能会阻止用户检测并响应安全关键事件。 用户界面虚假陈述可以采取多种形式：

指示不正确：显示不正确的信息，这会阻止用户了解软件的真实状态或软件正在监控的环境，尤其是潜在危险的情况或操作。这可以分解为几种不同的子类型。 叠加：显示区域用于提供关键信息，但另一个进程可以通过在显示屏上覆盖另一个元素来修改显示。用户没有与用户界面的预期部分交互。这是启用点击劫持攻击的问题，尽管存在涉及覆盖的许多其他类型的攻击。 图标操作：错误的图标或错误的颜色指示符可能会受到影响（例如使危险的.EXE可执行文件看起来像无害的.GIF） 时间安排：软件正在执行状态转换或上下文切换，并通过指示器呈现给用户，但竞争条件可能导致在产品完全切换上下文之前使用错误的指示符。如果攻击者可以触发错误，则可以无限期地延长竞赛窗口。 可视截断：重要信息可能会从显示中截断，例如带有危险扩展名的长文件名，因为恶意部分被截断，因此未在GUI中显示。使用过多的空格也会导致截断，或将潜在危险的指示符放在用户视野之外（例如“filename.txt .exe”）。当由于长度以外的原因而移除信息的一部分时会发生不同类型的截断，例如在输入的中间意外插入输入结束标记，例如C中的NUL字节式字符串。 视觉区分：视觉信息可能以一种使用户难以快速正确地区分显示器的关键和不重要部分的方式呈现。 同形函数：来自不同字符集，字体或语言的字母可能看起来非常相似（即可能在视觉上等效），导致人类用户误读文本（例如，进行网络钓鱼攻击以欺骗用户访问恶意网站，其名称与可信站点名称相似。这可以被视为一种视觉区分问题。

## CWE - 453 Insecure Default Variable Initialization

中文：**不安全的默认变量初始化**

### Description

* 默认情况下，软件使用不安全或不太安全的值初始化内部变量。

### Extended Description

* 没有详细描述

## CWE - 454 External Initialization of Trusted Variables or Data Stores

中文：**可信变量或数据存储的外部初始化**

### Description

* 该软件使用可由不信任的参与者修改的输入初始化关键内部变量或数据存储。

### Extended Description

* 软件系统应该不愿信任已在其信任边界之外初始化的变量，特别是如果它们由用户初始化。变量可能未正确初始化。如果攻击者可以初始化变量，那么他们可以影响易受攻击的系统将做什么。

## CWE - 455 Non-exit on Failed Initialization

中文：**初始化失败时不退出**

### Description

* 在初始化期间发生与安全相关的错误时，软件不会退出或以其他方式修改其操作，例如当配置文件具有格式错误时，这可能导致软件以不如管理员预期的安全方式执行。

### Extended Description

* 没有详细描述

## CWE - 456 Missing Initialization of a Variable

中文：**缺少变量的初始化**

### Description

* 该软件不会初始化关键变量，这会导致执行环境使用意外值。

### Extended Description

* 没有详细描述

## CWE - 457 Use of Uninitialized Variable

中文：**使用未初始化的变量**

### Description

* 代码使用尚未初始化的变量，从而导致不可预测或意外的结果。

### Extended Description

* 在某些语言（如C和C ++）中，默认情况下不会初始化堆栈变量。它们通常包含在调用函数之前具有堆栈内存的垃圾数据。攻击者有时可以控制或读取这些内容。在其他语言或条件中，未显式初始化的变量可以被赋予具有安全隐患的默认值，具体取决于程序的逻辑。存在未初始化的变量有时可能表示代码中存在排版错误。

## CWE - 458 DEPRECATED: Incorrect Initialization

中文：**DEPRECATED：初始化不正确**

### Description

* 这个弱点已被弃用，因为它的名称和描述不匹配。描述重复CWE-454，而名称提示更抽象的初始化问题。有关更抽象的问题，请参阅CWE-665。

### Extended Description

* 没有详细描述

## CWE - 459 Incomplete Cleanup

中文：**不完整的清理**

### Description

* 软件在使用后没有正确“清理”并删除临时或支持资源。

### Extended Description

* 没有详细描述

## CWE - 460 Improper Cleanup on Thrown Exception

中文：**抛出异常的不正确清理**

### Description

* 当抛出异常时，产品不会清除其状态或错误地清除其状态，从而导致意外状态或控制流。

### Extended Description

* 通常，当函数或循环变得复杂时，在整个执行过程中需要一定程度的资源清理。例外可能会扰乱代码流并阻止必要的清理。

## CWE - 462 Duplicate Key in Associative List (Alist)

中文：**关联列表中的重复键（Alist）**

### Description

* 关联列表中的重复键可能导致非唯一键被误认为是错误。

### Extended Description

* 重复键输入 - 如果alist设计正确 - 可以用作常量时间替换功能。但是，可能会错误地插入重复的键条目。由于这种歧义，不建议在关联列表中使用重复的键条目，不应该允许。

## CWE - 463 Deletion of Data Structure Sentinel

中文：**删除数据结构Sentinel**

### Description

* 意外删除数据结构标记可能会导致严重的编程逻辑问题。

### Extended Description

* 通常，数据结构标记用于标记数据结构的结构。一个常见的例子是字符串末尾的空字符。另一个常见的例子是链表，它可能包含一个标记列表末尾的标记。允许这种类型的控制数据易于访问是危险的。因此，重要的是防止在提供安全性的某些包装器接口之外的删除或修改。

## CWE - 464 Addition of Data Structure Sentinel

中文：**添加数据结构Sentinel**

### Description

* 意外添加数据结构标记会导致严重的编程逻辑问题。

### Extended Description

* 数据结构标记通常用于标记数据结构。一个常见的例子是字符串末尾的空字符或标记链表结尾的特殊标记。允许这种类型的控制数据易于访问是危险的。因此，保护免受哨兵的添加或修改是很重要的。

## CWE - 466 Return of Pointer Value Outside of Expected Range

中文：**超出预期范围的指针值的返回**

### Description

* 函数可以返回指向指针应该引用的缓冲区之外的内存的指针。

### Extended Description

* 没有详细描述

## CWE - 467 Use of sizeof() on a Pointer Type

中文：**在指针类型上使用sizeof（）**

### Description

* 代码在malloced指针类型上调用sizeof（），它总是返回wordsize / 8。如果程序员打算确定已分配了多少内存，则会产生意外结果。

### Extended Description

* 在指针上使用sizeof（）有时可以生成有用的信息。一个明显的例子是在平台上找出单词大小。通常，sizeof（指针）的外观表示错误。

## CWE - 468 Incorrect Pointer Scaling

中文：**指针缩放不正确**

### Description

* 在C和C ++中，由于数学运算被隐式缩放的语义，人们可能经常意外地引用错误的内存。

### Extended Description

* 没有详细描述

## CWE - 469 Use of Pointer Subtraction to Determine Size

中文：**使用指针减法来确定大小**

### Description

* 应用程序从另一个指针中减去一个指针以确定大小，但如果指针不存在于同一个内存块中，则此计算可能不正确。

### Extended Description

* 没有详细描述

## CWE - 470 Use of Externally-Controlled Input to Select Classes or Code ('Unsafe Reflection')

中文：**使用外部控制输入来选择类或代码（'不安全反射'）**

### Description

* 应用程序使用带反射的外部输入来选择要使用的类或代码，但它不足以防止输入选择不正确的类或代码。

### Extended Description

* 如果应用程序使用外部输入来确定要实例化的类或要调用的方法，则攻击者可以提供值以选择意外的类或方法。如果发生这种情况，则攻击者可以创建开发人员不想要的控制流路径。这些路径可能会绕过身份验证或访问控制检查，或以其他方式导致应用程序以意外方式运行。如果攻击者可以将文件上载到应用程序的类路径（CWE-427）上显示的位置，或者将新条目添加到应用程序的类路径（CWE-426），则这种情况将成为世界末日。在这些条件中的任何一种情况下，攻击者都可以使用反射将新的恶意行为引入应用程序。

## CWE - 471 Modification of Assumed-Immutable Data (MAID)

中文：**假定不可变数据的修改（MAID）**

### Description

* 该软件无法正确保护假定不可变元素不被攻击者修改。

### Extended Description

* 当一个特定的输入对于应用程序的运行至关重要而它根本不应该是可修改的时候就会发生这种情况，但确实如此。某些资源通常被认为是不可变的，例如Web应用程序中的隐藏表单字段，cookie和反向DNS查找。

## CWE - 472 External Control of Assumed-Immutable Web Parameter

中文：**假设不可变Web参数的外部控制**

### Description

* Web应用程序不能充分验证假定为不可变但实际上是外部可控的输入，例如隐藏的表单字段。

### Extended Description

* 如果Web产品未正确保护隐藏表单字段，参数，cookie或URL中的假定不可变值，则可能导致对关键数据的修改。 Web应用程序经常错误地假设在隐藏字段或cookie中传递给客户端的数据不容易被篡改。对用户可控制的数据进行不正确的验证可能会导致应用程序处理不正确，通常是恶意的输入。 例如，自定义cookie通常跨会话存储会话数据或持久数据。这种会话数据通常涉及服务器端的安全相关决策，例如用户认证和访问控制。因此，cookie可能包含敏感数据，例如用户凭据和权限。这是一种危险的做法，因为它通常会导致服务器端应用程序不正确地依赖客户端提供的cookie的价值。

## CWE - 473 PHP External Variable Modification

中文：**PHP外部变量修改**

### Description

* PHP应用程序无法正确防止从外部源修改变量，例如查询参数或cookie。这可能会使应用程序面临许多其他方面不存在的弱点。

### Extended Description

* 没有详细描述

## CWE - 474 Use of Function with Inconsistent Implementations

中文：**函数与不一致实现的使用**

### Description

* 该代码使用的函数跨操作系统和版本具有不一致的实现。

### Extended Description

* 当代码在程序员期望的不同环境下移植或构建时，使用不一致的实现可能会导致行为更改，这可能会在某些情况下导致安全问题。 许多功能的实现因平台而异，有时甚至是同一平台的不同版本。实施差异包括：

解释参数的方式略有不同会导致结果不一致。 该功能的某些实现具有重大的安全风险。 可能未在所有平台上定义该功能。 该函数可能会更改它可以提供的返回码，或更改其返回码的含义。

## CWE - 475 Undefined Behavior for Input to API

中文：**API输入的未定义行为**

### Description

* 除非将其控制参数设置为特定值，否则此函数的行为是未定义的。

### Extended Description

* 没有详细描述

## CWE - 476 NULL Pointer Dereference

中文：**NULL指针解除引用**

### Description

* 当应用程序取消引用它期望有效的指针但是为NULL时，会发生NULL指针解除引用，通常会导致崩溃或退出。

### Extended Description

* NULL指针解引用问题可能通过许多缺陷发生，包括竞争条件和简单的编程遗漏。

## CWE - 477 Use of Obsolete Function

中文：**使用过时功能**

### Description

* 该代码使用已弃用或过时的函数，这表明代码尚未被主动审查或维护。

### Extended Description

* 随着编程语言的发展，由于以下原因，函数偶尔会变得过时：

语言的进步 更好地理解如何有效和安全地执行操作 管理某些操作的约定的变化

被删除的函数通常被更新的对应函数替换，这些对应函数以某种不同的并且希望改进的方式执行相同的任务。

## CWE - 478 Missing Default Case in Switch Statement

中文：**缺少Switch语句中的默认情况**

### Description

* 代码在switch语句中没有默认情况，这可能导致复杂的逻辑错误和由此产生的弱点。

### Extended Description

* 该缺陷代表了软件开发中的常见问题，其中并非所有可能的变量值都由给定过程考虑或处理。因此，根据不良信息和级联故障结果做出进一步的决策。这种级联故障可能导致任何数量的安全问题，并构成系统中的重大故障。

## CWE - 479 Signal Handler Use of a Non-reentrant Function

中文：**信号处理程序使用非重入函数**

### Description

* 该程序定义了一个调用非重入函数的信号处理程序。

### Extended Description

* 非重入函数是在第一次调用完成之前无法安全地调用，中断和调用的函数，而不会导致内存损坏。这可能导致意外的系统状态具有不可预测的结果，并且具有取决于上下文的各种潜在后果，包括拒绝服务和代码执行。 许多函数不是可重入的，但如果在信号处理程序中使用它们中的一些可能会导致内存损坏。函数调用syslog（）就是一个例子。为了执行其功能，它将少量内存分配为“临时空间”。如果syslog（）被信号调用挂起并且信号处理程序调用syslog（），则这两个函数使用的内存将进入未定义且可能可利用的状态。 malloc（）和free（）的实现管理全局结构中的元数据，以便跟踪分配哪个内存与哪个内存可用，但它们是不可重入的。同时调用这些函数可能会导致元数据损坏。

## CWE - 480 Use of Incorrect Operator

中文：**使用不正确的操作员**

### Description

* 程序员意外地使用了错误的操作符，它以安全相关的方式更改应用程序逻辑。

### Extended Description

* 这些类型的错误通常是拼写错误的结果。

## CWE - 481 Assigning instead of Comparing

中文：**分配而不是比较**

### Description

* 当意图进行比较时，代码使用运算符进行赋值。

### Extended Description

* 在许多语言中，compare语句在外观上与赋值语句非常接近，并且经常被混淆。此错误通常是拼写错误的结果，通常会导致程序执行出现明显问题。如果比较在if语句中，则if语句通常会评估谓词右侧的值。

## CWE - 482 Comparing instead of Assigning

中文：**比较而不是分配**

### Description

* 在意图执行分配时，代码使用运算符进行比较。

### Extended Description

* 在许多语言中，compare语句与赋值语句的外观非常接近;他们经常感到困惑。

## CWE - 483 Incorrect Block Delimitation

中文：**块定界不正确**

### Description

* 代码未明确分隔旨在包含2个或更多语句的块，从而产生逻辑错误。

### Extended Description

* 在某些语言中，大括号（或其他分隔符）对于块是可选的。省略分隔符时，可以插入一个逻辑错误，其中一个语句被认为是在一个块中但不是。在某些情况下，逻辑错误可能会产生安全隐患。

## CWE - 484 Omitted Break Statement in Switch

中文：**Switch中省略的Break语句**

### Description

* 该程序在开关或类似构造中省略了break语句，导致与多个条件相关联的代码被执行。当程序员只打算执行与一个条件相关的代码时，这可能会导致问题。

### Extended Description

* 这可能导致关键代码在不应该执行的情况下执行。

## CWE - 486 Comparison of Classes by Name

中文：**按名称比较类**

### Description

* 该程序按名称比较类，当多个类具有相同的名称时，可能导致它使用错误的类。

### Extended Description

* 如果决定信任对象的方法和数据是基于类的名称，则恶意用户可能发送与受信任类同名的对象，从而获得对已知类和类型的信任。

## CWE - 487 Reliance on Package-level Scope

中文：**依赖于包级范围**

### Description

* Java包本身并不是封闭的;因此，依靠它们来实现代码安全并不是一种好的做法。

### Extended Description

* 包范围的目的是防止程序的其他部分意外访问。这是一种易于软件开发的功能，但不是安全功能。

## CWE - 488 Exposure of Data Element to Wrong Session

中文：**数据元素暴露于错误的会话**

### Description

* 该产品不足以强制执行不同会话状态之间的边界，从而导致数据被提供给错误会话或由错误会话使用。

### Extended Description

* 数据可以通过单个对象的成员变量（例如Servlet）和来自共享池的对象从一个会话“流出”到另一个会话。 对于Servlet，开发人员有时不明白，除非Servlet实现SingleThreadModel接口，否则Servlet是单例; Servlet只有一个实例，并且使用该实例并重新使用它来处理由不同线程同时处理的多个请求。一个常见的结果是开发人员使用Servlet成员字段，使得一个用户可能无意中看到另一个用户的数据。换句话说，在Servlet成员字段中存储用户数据会引入数据访问竞争条件。

## CWE - 489 Leftover Debug Code

中文：**剩余的调试代码**

### Description

* 可以使用可以创建非预期入口点的活动调试代码来部署应用程序。

### Extended Description

* 一种常见的开发实践是添加专门为调试或测试目的而设计的“后门”代码，该代码不打算随应用程序一起发布或部署。这些后门入口点会产生安全风险，因为它们在设计或测试期间不予考虑，并且超出了应用程序的预期运行条件。

## CWE - 491 Public cloneable() Method Without Final ('Object Hijack')

中文：**没有Final的公共cloneable（）方法（'Object Hijack'）**

### Description

* 类具有一个未声明为final的cloneable（）方法，它允许在不调用构造函数的情况下创建对象。这可能导致对象处于意外状态。

### Extended Description

* 没有详细描述

## CWE - 492 Use of Inner Class Containing Sensitive Data

中文：**使用包含敏感数据的内部类**

### Description

* 内部类被转换为可在包范围内访问的类，并且可能暴露程序员打算对攻击者保密的代码。

### Extended Description

* 内部类因其转换为Java字节码的方式而悄然引入了一些安全问题。在Java源代码中，似乎可以声明内部类只能由封闭类访问，但Java字节码没有内部类的概念，因此编译器必须将内部类声明转换为具有包级别的对等类访问原始的外部类。更隐蔽的是，由于内部类可以访问其封闭类中的私有字段，因此一旦内部类成为字节码中的对等类，编译器就会将内部类访问的私有字段转换为受保护字段。

## CWE - 493 Critical Public Variable Without Final Modifier

中文：**没有最终修饰符的关键公共变量**

### Description

* 该产品有一个非final的关键公共变量，它允许修改变量以包含意外值。

### Extended Description

* 如果某个字段是非final和public，则可以通过任何有权访问包含该字段的类的函数设置该值来更改该字段。如果程序的其他部分对该字段的内容做出假设，则可能导致漏洞。

## CWE - 494 Download of Code Without Integrity Check

中文：**下载没有完整性检查的代码**

### Description

* 该产品从远程位置下载源代码或可执行文件，并在不充分验证代码的来源和完整性的情况下执行代码。

### Extended Description

* 攻击者可以通过破坏主机服务器，执行DNS欺骗或修改传输中的代码来执行恶意代码。

## CWE - 495 Private Data Structure Returned From A Public Method

中文：**从公共方法返回的私有数据结构**

### Description

* 该产品有一个声明为public的方法，但返回对私有数据结构的引用，然后可以以意外的方式对其进行修改。

### Extended Description

* 没有详细描述

## CWE - 496 Public Data Assigned to Private Array-Typed Field

中文：**公共数据分配给私有阵列类型字段**

### Description

* 将公共数据分配给专用阵列等同于为该阵列提供公共访问权限。

### Extended Description

* 没有详细描述

## CWE - 497 Exposure of System Data to an Unauthorized Control Sphere

中文：**将系统数据暴露给未经授权的控制球体**

### Description

* 公开系统数据或调试信息有助于对手了解系统并形成攻击计划。

### Extended Description

* 当系统数据或调试信息通过输出流或记录功能离开程序时，会发生信息泄露，使未经授权的各方可以访问它。攻击者还可以通过向Web应用程序提交异常请求来导致错误。对这些错误的响应可以揭示详细的系统信息，拒绝服务，导致安全机制失败，以及使服务器崩溃。攻击者可以使用揭示技术，操作系统和产品版本的错误消息来调整针对这些技术中已知漏洞的攻击。应用程序可以使用提供重要实现细节的诊断方法，例如堆栈跟踪，作为其错误处理机制的一部分。

## CWE - 498 Cloneable Class Containing Sensitive Information

中文：**包含敏感信息的可克隆类**

### Description

* 代码包含一个包含敏感数据的类，但该类是可复制的。然后可以通过克隆类来访问数据。

### Extended Description

* 可克隆类是有效的开放类，因为数据不能隐藏在它们中。未明确拒绝克隆的类可以由任何其他类克隆，而无需运行构造函数。

## CWE - 499 Serializable Class Containing Sensitive Data

中文：**包含敏感数据的可序列化类**

### Description

* 代码包含一个包含敏感数据的类，但该类未明确拒绝序列化。可以通过另一个类序列化类来访问数据。

### Extended Description

* 可序列化类是有效的开放类，因为数据不能隐藏在它们中。未明确拒绝序列化的类可以由任何其他类序列化，然后可以依次使用存储在其中的数据。

## CWE - 500 Public Static Field Not Marked Final

中文：**公共静态字段未标记为最终**

### Description

* 对象包含未标记为final的公共静态字段，这可能允许以意外方式对其进行修改。

### Extended Description

* 公共静态变量可以在没有访问器的情况下读取，并且在没有应用程序中的任何类的mutator的情况下进行更改

## CWE - 501 Trust Boundary Violation

中文：**信任边界违规**

### Description

* 该产品在相同的数据结构或结构化消息中混合可信和不可信的数据。

### Extended Description

* 信任边界可以被认为是通过程序绘制的线。在该行的一侧，数据是不可信的。在该行的另一侧，数据被认为是值得信赖的。验证逻辑的目的是允许数据安全地跨越信任边界 - 从不受信任转移到受信任。当程序模糊信任信任和不信任信号之间的界线时，就会发生信任边界违规。通过在相同的数据结构中组合可信和不可信的数据，程序员更容易错误地信任未经验证的数据。

## CWE - 502 Deserialization of Untrusted Data

中文：**不受信任数据的反序列化**

### Description

* 应用程序对不受信任的数据进行反序列化，但未充分验证结果数据是否有效。

### Extended Description

* 序列化对象进行通信或保存以供以后使用通常很方便。但是，如果不使用加密技术来保护自身，则通常可以在不使用提供的访问器功能的情况下修改反序列化的数据或代码。此外，任何加密仍然是客户端安全 - 这是一个危险的安全假设。 不信任的数据不能被信任为格式良好。 当开发人员对“小工具链”或者在反序列化过程中可以自行执行的一系列实例和方法调用（即，在对象返回给调用者之前）没有限制时，攻击者有时可以利用它们来实现执行未经授权的操作，例如生成shell。

## CWE - 506 Embedded Malicious Code

中文：**嵌入式恶意代码**

### Description

* 该应用程序包含看似恶意的代码。

### Extended Description

* 恶意瑕疵已经获得了色彩缤纷的名字，包括特洛伊木马，活板门，定时炸弹和逻辑炸弹。开发人员可能会插入恶意代码，以便在将来的某个时间破坏应用程序或其主机系统的安全性。它通常是指执行有用服务但以用户不想要的方式利用程序用户权限的程序。

## CWE - 507 Trojan Horse

中文：**特洛伊木马**

### Description

* 该软件似乎包含良性或有用的功能，但它还包含违反正常操作的代码，这些代码违反了用户或系统管理员的预期安全策略。

### Extended Description

* 没有详细描述

## CWE - 508 Non-Replicating Malicious Code

中文：**非复制恶意代码**

### Description

* 非复制恶意代码仅驻留在受攻击的目标系统或软件上;它不会试图传播到其他系统。

### Extended Description

* 没有详细描述

## CWE - 509 Replicating Malicious Code (Virus or Worm)

中文：**复制恶意代码（病毒或蠕虫）**

### Description

* 复制恶意代码（包括病毒和蠕虫）会在成功破坏目标系统或软件后尝试攻击其他系统。

### Extended Description

* 没有详细描述

## CWE - 510 Trapdoor

中文：**暗门**

### Description

* 陷门是一段隐藏的代码，它响应特殊输入，允许用户访问资源而无需通过正常的安全执行机制。

### Extended Description

* 没有详细描述

## CWE - 511 Logic/Time Bomb

中文：**逻辑/时间炸弹**

### Description

* 该软件包含的代码旨在在特定时间过去或满足某个逻辑条件时破坏软件（或其环境）的合法操作。

### Extended Description

* 当引爆定时炸弹或逻辑炸弹时，它可能会执行拒绝服务，例如崩溃系统，删除关键数据或降低系统响应时间。这枚炸弹可能放在复制或非复制的特洛伊木马中。

## CWE - 512 Spyware

中文：**间谍软件**

### Description

* 该软件收集有关人类用户或用户活动的个人身份信息，但软件使用除自身之外的其他资源访问此信息，并且不需要用户明确批准或直接输入软件。

### Extended Description

* “间谍软件”是一个常用术语，有许多定义和解释。通常，它意味着收集信息或安装人类用户可能不允许的功能的软件，如果他们完全了解软件采取的操作。例如，用户可能希望税务软件收集社会安全号码并在提交纳税申报表时将其包括在内，但同一用户不希望游戏软件从该税务软件的数据中获取社会安全号码。

## CWE - 514 Covert Channel

中文：**隐蔽通道**

### Description

* 隐蔽通道是一种可用于以系统设计者不想要的方式传输信息的路径。

### Extended Description

* 通常，系统没有授权传输，也不知道它的发生。

## CWE - 515 Covert Storage Channel

中文：**隐蔽存储通道**

### Description

* 隐蔽存储信道通过一个程序的位设置和另一个程序读取这些位来传输信息。这种情况与普通操作的区别在于比特用于传送编码信息。

### Extended Description

* 当带外数据存储在消息中以用于存储器重用时，发生隐藏存储信道。隐蔽信道通常被分类为存储或定时信道。示例包括使用旨在仅保存审计信息以传达用户密码的文件 - 使用文件的名称或者与其相关联的状态位，所有用户都可以读取该信息以指示文件的内容。隐写信息以这样的方式隐藏信息，即除了预期接收者之外，没有人知道消息的存在，这是隐蔽存储信道的一个很好的例子。

## CWE - 516 DEPRECATED (Duplicate): Covert Timing Channel

中文：**已弃用（重复）：隐蔽时间通道**

### Description

* 这个弱点可以在CWE-385找到。

### Extended Description

* 没有详细描述

## CWE - 520 .NET Misconfiguration: Use of Impersonation

中文：**.NET配置错误：使用模拟**

### Description

* 允许.NET应用程序在潜在升级的底层操作和文件系统访问级别运行可能很危险，并导致各种形式的攻击。

### Extended Description

* .NET服务器应用程序可以选择使用对客户端进行身份验证的用户的身份来执行。此功能的目的是绕过.NET应用程序代码中的身份验证和访问控制检查。身份验证由底层Web服务器（Microsoft Internet信息服务IIS）完成，该服务器将经过身份验证的令牌或未经身份验证的匿名令牌传递给.NET应用程序。使用令牌模拟客户端，应用程序然后依赖于NTFS目录和文件中的设置来控制访问。模拟使应用程序能够在运行.NET应用程序的服务器上执行代码并在经过身份验证和授权的用户的上下文中访问资源。

## CWE - 521 Weak Password Requirements

中文：**密码要求低**

### Description

* 该产品不要求用户具有强密码，这使攻击者更容易破坏用户帐户。

### Extended Description

* 身份验证机制仅与其凭据一样强大。因此，要求用户拥有强密码非常重要。在尝试猜测用户密码时，缺乏密码复杂性会大大减少搜索空间，从而使暴力攻击更容易。

## CWE - 522 Insufficiently Protected Credentials

中文：**受保护的凭据不足**

### Description

* 当应用程序传输或存储认证凭证并使用易受未经授权的拦截和/或检索的不安全方法时，就会出现这种弱点。

### Extended Description

* 没有详细描述

## CWE - 523 Unprotected Transport of Credentials

中文：**不受保护的证书传输**

### Description

* 登录页面在从客户端到服务器的传输过程中没有使用足够的措施来保护用户名和密码。

### Extended Description

* 没有详细描述

## CWE - 524 Information Exposure Through Caching

中文：**通过缓存信息曝光**

### Description

* 应用程序使用缓存来维护对象，线程，连接，页面或密码池，以最大限度地减少访问它们所需的时间或它们连接的资源。如果实施不当，这些缓存可能允许访问未经授权的信息或导致拒绝服务漏洞。

### Extended Description

* 没有详细描述

## CWE - 525 Information Exposure Through Browser Caching

中文：**通过浏览器缓存进行信息曝光**

### Description

* 对于每个网页，应用程序应具有适当的缓存策略，指定应缓存页面及其表单字段的程度。

### Extended Description

* 没有详细描述

## CWE - 526 Information Exposure Through Environmental Variables

中文：**通过环境变量进行信息曝光**

### Description

* 环境变量可能包含有关远程服务器的敏感信息。

### Extended Description

* 没有详细描述

## CWE - 527 Exposure of CVS Repository to an Unauthorized Control Sphere

中文：**将CVS存储库暴露给未经授权的控制球体**

### Description

* 该产品将CVS存储库存储在目标或其他容器中，该目标或其他容器可供预期控制范围之外的actor访问。

### Extended Description

* 攻击者可以恢复Web服务器或其他服务器上的CVS子目录中包含的信息，并将其用于恶意目的。此信息可能包括用户名，文件名，路径根和IP地址。

## CWE - 528 Exposure of Core Dump File to an Unauthorized Control Sphere

中文：**核心转储文件暴露给未经授权的控制领域**

### Description

* 该产品在目录中生成核心转储文件，该目录可供预期控制范围之外的actor访问。

### Extended Description

* 没有详细描述

## CWE - 529 Exposure of Access Control List Files to an Unauthorized Control Sphere

中文：**将访问控制列表文件暴露给未经授权的控制领域**

### Description

* 该产品将访问控制列表文件存储在目标或其他容器中，该容器可供预期控制范围之外的actor访问。

### Extended Description

* 暴露这些访问控制列表文件可能会向攻击者提供有关站点或系统配置的信息。然后，该信息可用于绕过预期的安全策略或识别可从其发起攻击的可信系统。

## CWE - 530 Exposure of Backup File to an Unauthorized Control Sphere

中文：**将备份文件暴露给未经授权的控制领域**

### Description

* 备份文件存储在目标控制范围之外的actor可访问的目录中。

### Extended Description

* 通常，使用.~bk等扩展名重命名旧文件，以区别于生产文件。通常可以检索以这种方式重命名并留在webroot中的旧文件的源代码。此重命名可能已由Web服务器自动执行，或由管理员手动执行。

## CWE - 531 Information Exposure Through Test Code

中文：**通过测试代码的信息曝光**

### Description

* 可访问的测试应用程序可能带来各种安全风险。由于开发人员或管理员很少考虑除了他们自己以外的人甚至会知道这些应用程序的存在，因此他们通常会包含敏感信息或功能。

### Extended Description

* 没有详细描述

## CWE - 532 Information Exposure Through Log Files

中文：**通过日志文件的信息曝光**

### Description

* 写入日志文件的信息可能具有敏感性，可为攻击者提供有价值的指导或暴露敏感的用户信息。

### Extended Description

* 虽然在开发阶段记录所有信息可能会有所帮助，但在产品发布之前适当设置日志记录级别非常重要，这样敏感的用户数据和系统信息不会意外地暴露给潜在的攻击者。 可以生成和存储不同的日志文件：

服务器日志文件（例如server.log）。这可以提供有关文件的任何应用程序的信息。通常，这可以提供完整的路径名称和系统信息，有时也可以提供用户名和密码。 用于调试的日志文件

## CWE - 533 DEPRECATED: Information Exposure Through Server Log Files

中文：**已弃用：通过服务器日志文件进行信息泄露**

### Description

* 此条目已被弃用，因为它的抽象级别太低。见CWE-532。

### Extended Description

* 没有详细描述

## CWE - 534 DEPRECATED: Information Exposure Through Debug Log Files

中文：**已弃用：通过调试日志文件进行信息泄露**

### Description

* 此条目已被弃用，因为它的抽象级别太低。见CWE-532。

### Extended Description

* 没有详细描述

## CWE - 535 Information Exposure Through Shell Error Message

中文：**信息暴露通过Shell错误消息**

### Description

* 命令shell错误消息指示Web应用程序代码中存在未处理的异常。在许多情况下，攻击者可以利用导致这些错误的条件来获取对系统的未授权访问。

### Extended Description

* 没有详细描述

## CWE - 536 Information Exposure Through Servlet Runtime Error Message

中文：**通过Servlet运行时错误消息的信息泄露**

### Description

* servlet错误消息表明Web应用程序代码中存在未处理的异常，并可能为攻击者提供有用的信息。

### Extended Description

* 没有详细描述

## CWE - 537 Information Exposure Through Java Runtime Error Message

中文：**通过Java运行时错误消息的信息暴露**

### Description

* 在许多情况下，攻击者可以利用导致未处理的异常错误的条件来获得对系统的未授权访问。

### Extended Description

* 没有详细描述

## CWE - 538 File and Directory Information Exposure

中文：**文件和目录信息曝光**

### Description

* 该产品将敏感信息存储在目标控制范围之外的actor可访问的文件或目录中。

### Extended Description

* 没有详细描述

## CWE - 539 Information Exposure Through Persistent Cookies

中文：**通过持久性Cookie进行信息曝光**

### Description

* 持久性cookie是存储在浏览器硬盘上的cookie。这可能会导致安全性和隐私问题，具体取决于cookie中存储的信息以及访问方式。

### Extended Description

* Cookie是由Web应用程序发送但本地存储在浏览器中的一小部分数据。这使应用程序可以使用cookie在页面之间传递信息并存储变量信息。 Web应用程序控制cookie中存储的信息及其使用方式。存储在cookie中的典型信息类型是会话标识符，个性化和自定义信息，在极少数情况下甚至是用于启用自动登录的用户名。有两种不同类型的cookie：会话cookie和持久性cookie。会话cookie只存储在浏览器的内存中，不存储在任何地方，但持久性cookie存储在浏览器的硬盘上。

## CWE - 540 Information Exposure Through Source Code

中文：**信息通过源代码曝光**

### Description

* Web服务器上的源代码通常包含敏感信息，用户通常无法访问。

### Extended Description

* 在某些情况下，从区域或服务器中删除源代码至关重要。例如，在系统上获取Perl源代码允许攻击者理解脚本的逻辑并提取极其有用的信息，例如代码错误或登录和密码。

## CWE - 541 Information Exposure Through Include Source Code

中文：**信息曝光包括源代码**

### Description

* 如果可以访问包含文件源，则该文件可以包含用户名和密码，以及与应用程序和系统有关的敏感信息。

### Extended Description

* 没有详细描述

## CWE - 542 DEPRECATED: Information Exposure Through Cleanup Log Files

中文：**已弃用：通过清理日志文件进行信息泄露**

### Description

* 此条目已被弃用，因为它的抽象级别太低。见CWE-532。

### Extended Description

* 没有详细描述

## CWE - 543 Use of Singleton Pattern Without Synchronization in a Multithreaded Context

中文：**在多线程上下文中使用没有同步的单例模式**

### Description

* 在多线程环境中创建资源时，该软件使用单例模式。

### Extended Description

* 使用单例模式可能不是线程安全的。

## CWE - 544 Missing Standardized Error Handling Mechanism

中文：**缺少标准化错误处理机制**

### Description

* 该软件不使用标准化方法来处理整个代码中的错误，这可能会引入不一致的错误处理和由此产生的缺陷。

### Extended Description

* 如果应用程序单独处理错误消息，则逐个处理，这可能会导致错误处理不一致。错误的原因可能会丢失。此外，有关错误原因的详细信息可能会无意中返回给用户。

## CWE - 545 DEPRECATED: Use of Dynamic Class Loading

中文：**已弃用：使用动态类加载**

### Description

* 这个弱点已被弃用，因为它与CWE-470部分重叠，它描述了合法的程序员行为，其他部分需要集成到其他条目中。

### Extended Description

* 没有详细描述

## CWE - 546 Suspicious Comment

中文：**可疑的评论**

### Description

* 该代码包含表明存在错误，功能不完整或缺陷的注释。

### Extended Description

* 代码中的许多可疑注释（例如BUG，HACK，FIXME，LATER，LATER2，TODO）表示缺少安全功能和检查。其他人指出程序员应该修复的代码问题，例如硬编码变量，错误处理，不使用存储过程和性能问题。

## CWE - 547 Use of Hard-coded, Security-relevant Constants

中文：**使用硬编码，与安全相关的常量**

### Description

* 该程序使用硬编码常量而不是符号名称来表示安全关键值，这增加了代码维护或安全策略更改期间出错的可能性。

### Extended Description

* 如果开发人员未发现所有出现的硬编码常量，则如果其中一个常量未更改，则可能会做出错误的策略决策。更改这些值将需要在系统发布到现场后很难或不可能进行代码更改。此外，如果代码被公开，这些硬编码值可能会被攻击者使用。

## CWE - 548 Information Exposure Through Directory Listing

中文：**通过目录列表进行信息曝光**

### Description

* 目录列表被不恰当地暴露，从而向攻击者提供潜在的敏感信息。

### Extended Description

* 目录列表为攻击者提供了位于目录内的所有资源的完整索引。具体的风险和后果取决于列出和访问的文件。

## CWE - 549 Missing Password Field Masking

中文：**缺少密码字段屏蔽**

### Description

* 该软件在进入期间不会屏蔽密码，从而增加了攻击者观察和捕获密码的可能性。

### Extended Description

* 没有详细描述

## CWE - 550 Information Exposure Through Server Error Message

中文：**通过服务器错误消息暴露信息**

### Description

* 某些条件（如网络故障）将导致显示服务器错误消息。

### Extended Description

* 虽然错误信息本身并不危险，但攻击者可以从中收集错误信息，这可能会导致最终的问题。

## CWE - 551 Incorrect Behavior Order: Authorization Before Parsing and Canonicalization

中文：**行为顺序不正确：解析和规范化之前的授权**

### Description

* 如果Web服务器在检查请求的URL之前没有完全解析它们以进行授权，则攻击者可能会绕过授权保护。

### Extended Description

* 例如，字符串/./和/都表示当前目录。如果/ SomeDirectory是受保护的目录而攻击者请求/./SomeDirectory，则如果/./未在执行授权检查之前转换为/，则攻击者可能能够访问该资源。

## CWE - 552 Files or Directories Accessible to External Parties

中文：**可供外部各方访问的文件或目录**

### Description

* 文件或目录可以在不应该的环境中访问。

### Extended Description

* 没有详细描述

## CWE - 553 Command Shell in Externally Accessible Directory

中文：**外部可访问目录中的命令外壳**

### Description

* / cgi-bin /或其他可访问目录中存在可能的shell文件。这非常危险，攻击者可以使用它来执行Web服务器上的命令。

### Extended Description

* 没有详细描述

## CWE - 554 ASP.NET Misconfiguration: Not Using Input Validation Framework

中文：**ASP.NET配置错误：不使用输入验证框架**

### Description

* ASP.NET应用程序不使用输入验证框架。

### Extended Description

* 没有详细描述

## CWE - 555 J2EE Misconfiguration: Plaintext Password in Configuration File

中文：**J2EE配置错误：配置文件中的纯文本密码**

### Description

* J2EE应用程序将明文密码存储在配置文件中。

### Extended Description

* 在配置文件中存储明文密码允许任何能够读取该文件的人访问受密码保护的资源，使其成为攻击者的轻松目标。

## CWE - 556 ASP.NET Misconfiguration: Use of Identity Impersonation

中文：**ASP.NET配置错误：使用身份模拟**

### Description

* 配置ASP.NET应用程序以使用模拟凭据运行可能会为应用程序提供不必要的权限。

### Extended Description

* 使用模拟凭据允许ASP.NET应用程序以其正在执行的客户端的权限或在其配置中授予的任意权限运行。

## CWE - 558 Use of getlogin() in Multithreaded Application

中文：**在多线程应用程序中使用getlogin（）**

### Description

* 应用程序在多线程上下文中使用getlogin（）函数，可能导致它返回不正确的值。

### Extended Description

* getlogin（）函数返回一个指向字符串的指针，该字符串包含与调用进程关联的用户的名称。该函数不是可重入的，这意味着如果从另一个进程调用它，则内容不会被锁定，并且字符串的值可以被另一个进程更改。这使得使用起来非常危险，因为其他进程可以更改用户名，因此无法信任该函数的结果。

## CWE - 560 Use of umask() with chmod-style Argument

中文：**使用带有chmod样式参数的umask（）**

### Description

* 该产品使用不正确的参数调用umask（），该参数被指定为chmod（）的参数。

### Extended Description

* 没有详细描述

## CWE - 561 Dead Code

中文：**死代码**

### Description

* 该软件包含死代码，永远不能执行。

### Extended Description

* 死代码是永远不能在正在运行的程序中执行的源代码。周围的代码使得一段代码无法执行。

## CWE - 562 Return of Stack Variable Address

中文：**堆栈变量地址的返回**

### Description

* 函数返回堆栈变量的地址，这将导致意外的程序行为，通常以崩溃的形式。

### Extended Description

* 因为局部变量是在堆栈上分配的，所以当程序返回指向局部变量的指针时，它返回一个堆栈地址。后续函数调用可能会重复使用相同的堆栈地址，从而覆盖指针的值，该值不再对应于同一个变量，因为函数的堆栈帧在返回时无效。充其量这将导致指针的值意外更改。在许多情况下，它会导致程序在下次取消引用指针时崩溃。

## CWE - 563 Assignment to Variable without Use

中文：**不使用而赋值给变量**

### Description

* 变量的值已分配但从未使用过，使其成为死存储。

### Extended Description

* 赋值后，变量要么被赋予另一个值，要么超出范围。变量很可能只是残留，但未使用的变量也可能指出错误。

## CWE - 564 SQL Injection: Hibernate

中文：**SQL注入：Hibernate**

### Description

* 使用Hibernate执行使用用户控制的输入构建的动态SQL语句可以允许攻击者修改语句的含义或执行任意SQL命令。

### Extended Description

* 没有详细描述

## CWE - 565 Reliance on Cookies without Validation and Integrity Checking

中文：**依赖于Cookie而无需验证和完整性检查**

### Description

* 应用程序在执行安全关键操作时依赖于cookie的存在或值，但它无法正确确保该设置对关联用户有效。

### Extended Description

* 攻击者可以在浏览器中轻松修改cookie，也可以在浏览器外部实现客户端代码。在没有详细验证和完整性检查的情况下依赖cookie可以允许攻击者绕过身份验证，执行注入攻击（如SQL注入和跨站点脚本），或以其他方式修改输入。

## CWE - 566 Authorization Bypass Through User-Controlled SQL Primary Key

中文：**授权绕过用户控制的SQL主键**

### Description

* 该软件使用一个数据库表，该表包含一个演员不应该访问的记录，但它执行一个带有可由该actor控制的主键的SQL语句。

### Extended Description

* 当用户可以将主键设置为任何值时，用户可以修改该键以指向未授权的记录。 数据库访问控制错误发生在：

数据从不受信任的来源进入程序。 该数据用于指定SQL查询中主键的值。 不受信任的源没有权限可以访问关联表中的所有行。

## CWE - 567 Unsynchronized Access to Shared Data in a Multithreaded Context

中文：**多线程上下文中对共享数据的不同步访问**

### Description

* 该产品无法正确同步共享数据，例如跨线程的静态变量，这可能导致未定义的行为和不可预测的数据更改。

### Extended Description

* 在servlet中，共享静态变量不受并发访问的保护，但servlet是多线程的。这是J2EE应用程序中的典型编程错误，因为多线程由框架处理。当共享变量可能受到攻击者的影响时，一个线程最终可能会修改变量以包含对同样使用变量中的数据的其他线程无效的数据。 请注意，这个弱点并不是servlet独有的。

## CWE - 568 finalize() Method Without super.finalize()

中文：**finalize（）方法没有super.finalize（）**

### Description

* 该软件包含一个不调用super.finalize（）的finalize（）方法。

### Extended Description

* Java语言规范声明，finalize（）方法调用super.finalize（）是一种很好的做法。

## CWE - 570 Expression is Always False

中文：**表达始终是错误的**

### Description

* 该软件包含一个总是评估为false的表达式。

### Extended Description

* 没有详细描述

## CWE - 571 Expression is Always True

中文：**表达始终如一**

### Description

* 该软件包含一个总是评估为true的表达式。

### Extended Description

* 没有详细描述

## CWE - 572 Call to Thread run() instead of start()

中文：**调用Thread run（）而不是start（）**

### Description

* 程序调用线程的run（）方法而不是调用start（），这会导致代码在调用者的线程而不是被调用者中运行。

### Extended Description

* 在大多数情况下，直接调用Thread对象的run（）方法是一个错误。程序员打算开始一个新的控制线程，但是不小心调用了run（）而不是start（），因此run（）方法将在调用者的控制线程中执行。

## CWE - 573 Improper Following of Specification by Caller

中文：**调用者对规范的不正确遵循**

### Description

* 该软件不遵循或错误地遵循实现语言，环境，框架，协议或平台所要求的规范。

### Extended Description

* 当利用外部功能（例如API）时，调用者根据外部功能的要求这样做是很重要的，否则可能会导致意外行为，从而可能使系统容易受到任何数量的攻击。

## CWE - 574 EJB Bad Practices: Use of Synchronization Primitives

中文：**EJB不良做法：使用同步原语**

### Description

* 该程序通过使用线程同步原语违反Enterprise JavaBeans（EJB）规范。

### Extended Description

* Enterprise JavaBeans规范要求每个bean提供程序遵循一组编程指南，这些指南旨在确保bean在任何EJB容器中都是可移植的并且行为一致。在这种情况下，该程序违反了以下EJB准则：“企业bean不得使用线程同步原语来同步多个实例的执行。”规范以下列方式证明了这一要求：“这条规则是确保一致的运行时语义所必需的，因为虽然一些EJB容器可能使用单个JVM来执行所有企业bean的实例，但其他EJB容器可能会将实例分布在多个JVM上。”

## CWE - 575 EJB Bad Practices: Use of AWT Swing

中文：**EJB不良做法：使用AWT Swing**

### Description

* 该程序使用AWT / Swing违反了Enterprise JavaBeans（EJB）规范。

### Extended Description

* Enterprise JavaBeans规范要求每个bean提供程序遵循一组编程指南，这些指南旨在确保bean在任何EJB容器中都是可移植的并且行为一致。在这种情况下，程序违反了以下EJB准则：“企业bean不得使用AWT功能尝试将信息输出到显示器，或从键盘输入信息。”该规范通过以下方式证明了这一要求：“大多数服务器不允许应用程序与连接到服务器系统的键盘/显示器之间的直接交互。”

## CWE - 576 EJB Bad Practices: Use of Java I/O

中文：**EJB不良做法：使用Java I / O.**

### Description

* 该程序使用java.io包违反Enterprise JavaBeans（EJB）规范。

### Extended Description

* Enterprise JavaBeans规范要求每个bean提供程序遵循一组编程指南，这些指南旨在确保bean在任何EJB容器中都是可移植的并且行为一致。在这种情况下，该程序违反了以下EJB准则：“企业bean不得使用java.io包来尝试访问文件系统中的文件和目录。”该规范以下列方式证明了这一要求：“文件系统API不适合业务组件访问数据。业务组件应使用资源管理器API（如JDBC）来存储数据。”

## CWE - 577 EJB Bad Practices: Use of Sockets

中文：**EJB不良做法：使用套接字**

### Description

* 该程序使用套接字违反Enterprise JavaBeans（EJB）规范。

### Extended Description

* Enterprise JavaBeans规范要求每个bean提供程序遵循一组编程指南，这些指南旨在确保bean在任何EJB容器中都是可移植的并且行为一致。在这种情况下，该程序违反了以下EJB准则：“企业bean不得尝试侦听套接字，接受套接字上的连接，或使用套接字进行多播。”规范通过以下方式证明了这一要求：“EJB体系结构允许企业bean实例成为网络套接字客户端，但它不允许它成为网络服务器。允许实例成为网络服务器会与企业bean的基本功能 - 为EJB客户端提供服务。“

## CWE - 578 EJB Bad Practices: Use of Class Loader

中文：**EJB不良做法：使用类加载器**

### Description

* 该程序通过使用类加载器违反Enterprise JavaBeans（EJB）规范。

### Extended Description

* Enterprise JavaBeans规范要求每个bean提供程序遵循一组编程指南，这些指南旨在确保bean在任何EJB容器中都是可移植的并且行为一致。在这种情况下，程序违反了以下EJB准则：“企业bean不得尝试创建类加载器;获取当前类加载器;设置上下文类加载器;设置安全管理器;创建新的安全管理器;停止JVM ;或更改输入，输出和错误流。“规范通过以下方式证明了这一要求：“这些函数是为EJB容器保留的。允许企业bean使用这些函数可能会危及安全性并降低容器正确管理运行时环境的能力。”

## CWE - 579 J2EE Bad Practices: Non-serializable Object Stored in Session

中文：**J2EE不良做法：会话中存储的非序列化对象**

### Description

* 应用程序将非序列化对象存储为HttpSession属性，这会损害可靠性。

### Extended Description

* J2EE应用程序可以使用多个JVM，以提高应用程序的可靠性和性能。为了使多个JVM作为单个应用程序显示给最终用户，J2EE容器可以跨多个JVM复制HttpSession对象，这样，如果一个JVM变得不可用，另一个JVM就可以介入并取代它而不会中断应用程序的流程。这只有在所有会话数据都可序列化的情况下才有可能，允许在JVM之间复制会话。

## CWE - 580 clone() Method Without super.clone()

中文：**clone（）方法没有super.clone（）**

### Description

* 该软件包含一个clone（）方法，该方法不会调用super.clone（）来获取新对象。

### Extended Description

* clone（）的所有实现都应该通过调用super.clone（）来获取新对象。如果某个类不遵循此约定，则子类的clone（）方法将返回错误类型的对象。

## CWE - 581 Object Model Violation: Just One of Equals and Hashcode Defined

中文：**对象模型违规：只定义一个等于和Hashcode**

### Description

* 该软件不保持相等对象的相等哈希码。

### Extended Description

* 期望Java对象遵守与平等相关的许多不变量。其中一个不变量是等对象必须具有相同的哈希码。换句话说，如果a.equals（b）== true，则a.hashCode（）== b.hashCode（）。

## CWE - 582 Array Declared Public, Final, and Static

中文：**Array声明为Public，Final和Static**

### Description

* 程序声明一个数组public，final和static，这不足以防止修改数组的内容。

### Extended Description

* 因为数组是可变对象，所以最终约束要求数组对象本身只分配一次，但不保证数组元素的值。由于阵列是公共的，因此恶意程序可以更改存储在阵列中的值。因此，在大多数情况下，声明为public，final和static的数组是一个错误。

## CWE - 583 finalize() Method Declared Public

中文：**finalize（）方法声明为Public**

### Description

* 该程序通过声明finalize（）方法公开来违反移动代码的安全编码原则。

### Extended Description

* 除了在finalize（）的实现中调用super.finalize（）之外，程序绝不应该显式调用finalize。在移动代码情况下，如果攻击者可以恶意调用您的一个finalize（）方法，那么手动垃圾收集的其他容易出错的做法可能会成为安全威胁，因为它是通过公共访问声明的。

## CWE - 584 Return Inside Finally Block

中文：**返回内部最后阻止**

### Description

* 代码在finally块中有一个return语句，这将导致try块中的任何抛出异常被丢弃。

### Extended Description

* 没有详细描述

## CWE - 585 Empty Synchronized Block

中文：**空同步块**

### Description

* 该软件包含一个空的同步块。

### Extended Description

* 空的同步块实际上不会实现任何同步，并且可能指示代码的问题部分。可以发生空的同步块，因为在不移除同步块的情况下，注释掉同步块内不再需要的代码。

## CWE - 586 Explicit Call to Finalize()

中文：**明确调用Finalize（）**

### Description

* 该软件从终结器外部显式调用finalize（）方法。

### Extended Description

* 虽然Java语言规范允许从终结器外部调用对象的finalize（）方法，但这样做通常是个坏主意。例如，调用finalize（）显式意味着finalize（）将被多次调用：第一次是显式调用，最后一次是在对象被垃圾回收后进行的调用。

## CWE - 587 Assignment of a Fixed Address to a Pointer

中文：**将固定地址分配给指针**

### Description

* 该软件设置指向除NULL或0以外的特定地址的指针。

### Extended Description

* 使用固定地址是不可移植的，因为该地址可能在所有环境或平台中都无效。

## CWE - 588 Attempt to Access Child of a Non-structure Pointer

中文：**尝试访问非结构指针的子项**

### Description

* 将非结构类型转换为结构类型并访问字段可能导致内存访问错误或数据损坏。

### Extended Description

* 没有详细描述

## CWE - 589 Call to Non-ubiquitous API

中文：**调用非普遍存在的API**

### Description

* 该软件使用目标平台的所有版本上不存在的API函数。这可能会导致可移植性问题或不一致，从而导致拒绝服务或其他后果。

### Extended Description

* 某些提供OS支持的安全功能的功能在常用的所有OS版本上都不可用。同样，出于安全原因，函数经常被弃用或过时，不应使用。

## CWE - 590 Free of Memory not on the Heap

中文：**没有记忆而不是堆**

### Description

* 应用程序在指向未使用相关堆分配函数（如malloc（），calloc（）或realloc（）分配的内存的指针上调用free（）。

### Extended Description

* 当在无效指针上调用free（）时，程序的内存管理数据结构可能会损坏。这种损坏可能导致程序崩溃，或者在某些情况下，攻击者可能会导致free（）操作可控内存位置以修改关键程序变量或执行代码。

## CWE - 591 Sensitive Data Storage in Improperly Locked Memory

中文：**锁定内存不正确的敏感数据存储**

### Description

* 应用程序将敏感数据存储在未锁定或未正确锁定的内存中，这可能导致内存被虚拟内存管理器写入磁盘上的交换文件。这可以使外部参与者更容易访问数据。

### Extended Description

* 在Windows系统上，VirtualLock功能可以锁定一页内存，以确保它将保留在内存中而不会交换到磁盘。但是，在旧版本的Windows上，例如95,98或Me，VirtualLock（）函数只是一个存根，不提供保护。在POSIX系统上，mlock（）调用确保页面将驻留在内存中，但不保证页面不会出现在交换中。因此，它不适合用作敏感数据的保护机制。某些平台，特别是Linux，确实保证页面不会被交换，但这是非标准的并且不可移植。调用mlock（）也需要supervisor权限。必须检查这两个调用的返回值，以确保锁定操作实际上是成功的。

## CWE - 592 DEPRECATED: Authentication Bypass Issues

中文：**DEPRECATED：身份验证绕过问题**

### Description

* 这个弱点已被弃用，因为它涵盖了CWE-287中已经描述的冗余概念。

### Extended Description

* 没有详细描述

## CWE - 593 Authentication Bypass: OpenSSL CTX Object Modified after SSL Objects are Created

中文：**身份验证绕过：创建SSL对象后修改OpenSSL CTX对象**

### Description

* 连接创建开始后，软件会修改SSL上下文。

### Extended Description

* 如果程序在从其创建SSL对象后修改SSL\_CTX对象，则从原始上下文创建的旧SSL对象可能都会受到该更改的影响。

## CWE - 594 J2EE Framework: Saving Unserializable Objects to Disk

中文：**J2EE框架：将不可序列化的对象保存到磁盘**

### Description

* 当J2EE容器尝试将不可序列化的对象写入磁盘时，无法保证该过程将成功完成。

### Extended Description

* 在高负载条件下，大多数J2EE应用程序框架将对象刷新到磁盘以管理传入请求的内存要求。例如，会话范围对象，甚至是应用程序范围对象，在需要时写入磁盘。虽然这些应用程序框架完成了将对象写入磁盘的实际工作，但它们并未强制要求这些对象可序列化，从而使Web应用程序容易受到序列化失败引起的崩溃的影响。攻击者可以通过向服务器发送足够的请求来强制Web应用程序将对象保存到磁盘来安装拒绝服务攻击。

## CWE - 595 Comparison of Object References Instead of Object Contents

中文：**对象引用的比较而不是对象内容**

### Description

* 该程序比较对象引用而不是对象本身的内容，从而阻止它检测等效对象。

### Extended Description

* 例如，在Java中，使用比较对象通常会产生欺骗性结果，因为运算符会比较对象引用而不是值;通常，这意味着对字符串使用==实际上是比较字符串的引用，而不是它们的值。

## CWE - 596 DEPRECATED: Incorrect Semantic Object Comparison

中文：**弃用：语义对象比较不正确**

### Description

* 这个弱点已被弃用。它描述得很差，很难与其他条目区分开来。仅因特定领域的考虑而分配单独的ID也是不合适的。它最接近的等价物是CWE-1023。

### Extended Description

* 没有详细描述

## CWE - 597 Use of Wrong Operator in String Comparison

中文：**在字符串比较中使用错误的运算符**

### Description

* 比较字符串时，产品使用错误的运算符，例如，当使用equals（）方法时，使用“==”。

### Extended Description

* 在Java中，使用==或！=比较两个字符串的相等性实际上比较了两个对象的相等性，而不是它们的值。两个引用永远不会相等的机会很好。虽然这种弱点通常只会影响程序的正确性，但如果将相等性用于安全性决策，则可以利用它来影响程序的安全性。

## CWE - 598 Information Exposure Through Query Strings in GET Request

中文：**GET请求中通过查询字符串的信息暴露**

### Description

* Web应用程序使用GET方法处理包含敏感信息的请求，这些信息可以通过浏览器的历史记录，引用，Web日志和其他来源公开该信息。

### Extended Description

* 没有详细描述

## CWE - 599 Missing Validation of OpenSSL Certificate

中文：**缺少对OpenSSL证书的验证**

### Description

* 该软件使用OpenSSL并信任或使用证书，而不使用SSL*get*verify\_result（）函数来确保证书满足所有必要的安全要求。

### Extended Description

* 这可能允许攻击者使用无效证书声称自己是可信主机，使用过期证书或进行其他攻击，如果证书得到正确验证，则可以检测到这些攻击。

## CWE - 600 Uncaught Exception in Servlet

中文：**Servlet中未捕获的异常**

### Description

* Servlet不会捕获所有异常，这可能会泄露敏感的调试信息。

### Extended Description

* 当Servlet抛出异常时，Servlet容器发送回用户的默认错误响应通常包括调试信息。这些信息对攻击者来说非常有价值。例如，堆栈跟踪可能会向攻击者显示格式错误的SQL查询字符串，正在使用的数据库类型以及应用程序容器的版本。此信息使攻击者能够针对这些组件中的已知漏洞。

## CWE - 601 URL Redirection to Untrusted Site ('Open Redirect')

中文：**URL重定向到不受信任的站点（'打开重定向'）**

### Description

* Web应用程序接受用户控制的输入，该输入指定指向外部站点的链接，并在重定向中使用该链接。这简化了网络钓鱼攻击。

### Extended Description

* http参数可能包含URL值，并可能导致Web应用程序将请求重定向到指定的URL。通过将URL值修改为恶意站点，攻击者可能会成功启动网络钓鱼诈骗并窃取用户凭据。由于修改后的链接中的服务器名称与原始站点相同，因此网络钓鱼尝试具有更可靠的外观。

## CWE - 602 Client-Side Enforcement of Server-Side Security

中文：**客户端执行服务器端安全性**

### Description

* 该软件由依赖客户端实现旨在保护服务器的机制的服务器组成。

### Extended Description

* 当服务器依赖于客户端上的保护机制时，攻击者可以修改客户端行为以绕过保护机制，从而导致客户端和服务器之间可能发生意外交互。后果将根据机制试图保护的内容而有所不同。

## CWE - 603 Use of Client-Side Authentication

中文：**使用客户端身份验证**

### Description

* 客户端/服务器产品在客户端代码中执行身份验证，但不在服务器代码中执行身份验证，允许通过省略身份验证检查的已修改客户端绕过服务器端身份验证。

### Extended Description

* 客户端身份验证非常弱，可能很容易被破坏。任何攻击者都可以阅读源代码并对身份验证机制进行反向工程，以访问本应受到保护的应用程序部分。

## CWE - 605 Multiple Binds to the Same Port

中文：**多个绑定到同一端口**

### Description

* 当允许多个套接字绑定到同一端口时，该端口上的其他服务可能被盗或被欺骗。

### Extended Description

* 在大多数系统上，设置SO*REUSEADDR套接字选项和调用bind（）的组合允许任何进程绑定到先前进程与INADDR*ANY绑定的端口。这允许用户绑定到非特权端口上绑定到INADDR\_ANY的服务器的特定地址，并窃取其UDP数据包/ TCP连接。

## CWE - 606 Unchecked Input for Loop Condition

中文：**未经检查的循环条件输入**

### Description

* 产品未正确检查用于循环条件的输入，可能由于过度循环而导致拒绝服务。

### Extended Description

* 没有详细描述

## CWE - 607 Public Static Final Field References Mutable Object

中文：**公共静态最终字段引用可变对象**

### Description

* 公共或受保护的静态final字段引用可变对象，该对象允许通过恶意代码更改对象，或者意外地从另一个包更改对象。

### Extended Description

* 没有详细描述

## CWE - 608 Struts: Non-private Field in ActionForm Class

中文：**Struts：ActionForm类中的非私有字段**

### Description

* ActionForm类包含一个尚未声明为私有的字段，可以在不使用setter或getter的情况下访问该字段。

### Extended Description

* 没有详细描述

## CWE - 609 Double-Checked Locking

中文：**双重锁定**

### Description

* 该程序使用双重检查锁定来访问资源，而没有显式同步的开销，但锁定不充分。

### Extended Description

* 双重检查锁定指的是程序员检查资源是否已初始化，抓取锁定，再次检查以查看资源是否已初始化的情况，然后执行初始化（如果尚未发生）。不应该这样做，因为不能保证在所有语言和所有体系结构中都能工作。总之，其他线程可能无法在同步块内运行，并且无法保证看到操作的执行顺序与它们在同步块中出现的顺序相同。

## CWE - 610 Externally Controlled Reference to a Resource in Another Sphere

中文：**对另一个领域中的资源的外部控制引用**

### Description

* 该产品使用外部控制的名称或引用，该名称或引用解析为预期控制范围之外的资源。

### Extended Description

* 没有详细描述

## CWE - 611 Improper Restriction of XML External Entity Reference ('XXE')

中文：**XML外部实体引用的限制不当（'XXE'）**

### Description

* 该软件处理一个XML文档，该文档可以包含带有URI的XML实体，这些URI可以解析为预期控制范围之外的文档，从而导致产品将不正确的文档嵌入其输出中。

### Extended Description

* XML文档可选地包含文档类型定义（Document Type Definition，DTD），除了其他功能之外，它还支持XML实体的定义。可以通过以URI的形式提供替换字符串来定义实体。 XML解析器可以访问此URI的内容，并将这些内容嵌入到XML文档中以供进一步处理。 通过提交使用file：// URI定义外部实体的XML文件，攻击者可以使处理应用程序读取本地文件的内容。例如，诸如“file：/// c：/winnt/win.ini”之类的URI指定（在Windows中）文件C：\ Winnt \ win.ini，或者file：/// etc / passwd指定密码基于Unix的系统中的文件。将URI与其他方案（如http：//）一起使用，攻击者可以强制应用程序向攻击者无法直接访问的服务器发出传出请求，这可以用来绕过防火墙限制或隐藏端口扫描等攻击源。 读取URI的内容后，它将反馈到正在处理XML的应用程序中。该应用程序可以回送数据（例如，在错误消息中），从而暴露文件内容。

## CWE - 612 Information Exposure Through Indexing of Private Data

中文：**通过索引私有数据进行信息曝光**

### Description

* 该产品针对私有文档执行索引例程，但没有充分验证可以访问索引的actor还具有访问私有文档的权限。

### Extended Description

* 当针对一组私有文档应用索引例程，并且该索引的结果可供无法访问这些文档的外部人员使用时，外部人员可能能够通过进行有针对性的搜索来获取敏感信息。如果搜索结果包含不属于搜索查询的周围文本，则风险尤其危险。此问题可能出现在未配置（或实现）的搜索引擎中，以忽略应保持隐藏的关键文件;即使没有直接下载这些文件的权限，远程用户也可以阅读它们。

## CWE - 613 Insufficient Session Expiration

中文：**会话过期不足**

### Description

* 据WASC称，“会话过期不足是指网站允许攻击者重用旧会话凭证或会话ID进行授权。”

### Extended Description

* 没有详细描述

## CWE - 614 Sensitive Cookie in HTTPS Session Without 'Secure' Attribute

中文：**没有“安全”属性的HTTPS会话中的敏感Cookie**

### Description

* 未设置HTTPS会话中敏感cookie的安全属性，这可能导致用户代理通过HTTP会话以纯文本格式发送这些cookie。

### Extended Description

* 没有详细描述

## CWE - 615 Information Exposure Through Comments

中文：**信息曝光评论**

### Description

* 虽然添加一般注释非常有用，但是一些程序员倾向于保留重要数据，例如：与Web应用程序相关的文件名，旧链接或用户不想浏览的链接，旧代码片段等。

### Extended Description

* 发现这些注释的攻击者可以映射应用程序的结构和文件，公开站点的隐藏部分，并研究代码片段以对应用程序进行反向工程，这可能有助于对站点进行进一步的攻击。

## CWE - 616 Incomplete Identification of Uploaded File Variables (PHP)

中文：**上传文件变量的不完整标识（PHP）**

### Description

* PHP应用程序使用旧方法处理上传文件，方法是引用为每个文件设置的四个全局变量（例如 varname*size，$ varname*name，$ varname\_type）。这些变量可能被攻击者覆盖，导致应用程序处理未经授权的文件。

### Extended Description

* POST请求，cookie或其他填充或覆盖这些变量的方法可能会覆盖这些全局变量。这可以通过提供诸如“/ etc / passwd”之类的值来读取或处理任意文件。

## CWE - 617 Reachable Assertion

中文：**可达到的断言**

### Description

* 该产品包含一个可由攻击者触发的assert（）或类似语句，这会导致应用程序退出或其他比必要更严重的行为。

### Extended Description

* 虽然断言有助于捕获逻辑错误并降低达到更严重漏洞条件的可能性，但它仍然可能导致拒绝服务。 例如，如果服务器处理多个并发连接，并且在一个连接中发生assert（）导致所有其他连接被删除，则这是一个可达的断言，导致拒绝服务。

## CWE - 618 Exposed Unsafe ActiveX Method

中文：**暴露的不安全的ActiveX方法**

### Description

* ActiveX控件旨在用于Web浏览器，但它暴露了执行浏览器安全模型之外的操作的危险方法（例如区域或域）。

### Extended Description

* 与典型的Java或javascript相比，ActiveX控件可以对操作系统进行更大的控制。暴露的方法可能会受到各种漏洞的影响，具体取决于这些方法的实现行为，以及是否对提供的参数执行输入验证。如果没有完整性检查或原始验证，攻击者可以调用此方法。

## CWE - 619 Dangling Database Cursor ('Cursor Injection')

中文：**悬空数据库光标（'光标注入'）**

### Description

* 如果未正确关闭数据库游标，则其他用户可以访问它，同时保留最初分配的相同权限，将游标保留为“悬空”。

### Extended Description

* 例如，未处理的异常可能会产生不正确的悬空游标。问题的影响取决于游标的角色，但SQL注入攻击通常是可能的。

## CWE - 620 Unverified Password Change

中文：**未验证的密码更改**

### Description

* 为用户设置新密码时，产品不需要了解原始密码或使用其他形式的身份验证。

### Extended Description

* 攻击者可以使用此方法更改其他用户的密码，从而获得与该用户关联的权限。

## CWE - 621 Variable Extraction Error

中文：**变量提取错误**

### Description

* 该产品使用外部输入来确定提取信息的变量的名称，而不验证指定变量的名称是否有效。这可能导致程序覆盖非预期的变量。

### Extended Description

* 例如，在PHP中，提取可用于提供类似于register*globals的功能，这是一种在生产系统中经常禁用的危险功能。在没有正确参数的情况下调用extract（）或import*request\_variables（）可能会覆盖任意全局变量，包括超全局变量。 其他解释语言也可以使用类似的功能，包括自定义语言。

## CWE - 622 Improper Validation of Function Hook Arguments

中文：**函数钩子参数的不正确验证**

### Description

* 产品会向用户可访问的API函数添加挂钩，但不会正确验证参数。这可能会导致漏洞。

### Extended Description

* 此类挂钩可用于以特权运行的防御软件，例如反病毒或防火墙，它可以挂钩内核调用。当参数未经验证时，它们可用于绕过保护方案或攻击产品本身。

## CWE - 623 Unsafe ActiveX Control Marked Safe For Scripting

中文：**不安全的ActiveX控件标记为脚本安全**

### Description

* ActiveX控件旨在限制使用，但已将其标记为可安全执行脚本。

### Extended Description

* 这可能允许攻击者通过访问控件的网页使用危险功能，这可能导致不同的结果漏洞，具体取决于控件的行为。

## CWE - 624 Executable Regular Expression Error

中文：**可执行的正则表达式错误**

### Description

* 该产品使用正则表达式，或者（1）包含具有用户控制输入的可执行组件，或者（2）允许用户通过插入模式修饰符来启用执行。

### Extended Description

* 可以在PHP preg\_replace（）函数中使用Case（2），并且可能在其他语言中将用户控制的输入插入到稍后被解析为正则表达式的字符串中。

## CWE - 625 Permissive Regular Expression

中文：**允许正则表达式**

### Description

* 该产品使用的正则表达式不足以限制允许值的集合。

### Extended Description

* 这有效地使正则表达式接受与模式匹配的子串，从而产生与目标的部分比较。在某些情况下，这可能会导致其他弱点。常见错误包括：

不识别目标字符串的开头和结尾 使用通配符而不是可接受的字符范围 其他

## CWE - 626 Null Byte Interaction Error (Poison Null Byte)

中文：**空字节交互错误（Poison Null Byte）**

### Description

* 在不同表示或组件之间传递数据时，产品无法正确处理空字节或NUL字符。

### Extended Description

* 空字节（NUL字符）可以在表示或语言之间具有不同的含义。例如，它是标准C库中的字符串终止符，但Perl和PHP字符串不会将其视为终止符。当两个表示相交时 - 例如当Perl或PHP调用底层C功能时 - 这会产生具有意外结果的交互错误。 ASP已经报道了类似的问题。用C编写的其他口译员也可能受到影响。 通过终止添加到文件名的硬编码扩展，毒性空字节在路径遍历攻击中经常有用。它可以在PHP中的正则表达式处理中发挥作用。

## CWE - 627 Dynamic Variable Evaluation

中文：**动态变量评估**

### Description

* 在用户可以在运行时影响变量名称的语言中，如果不控制变量名，攻击者可以读取或写入任意变量，或访问任意函数。

### Extended Description

* 由此产生的漏洞取决于应用程序的行为，无论是在交叉点还是在相关变量或函数可访问的任何控制/数据流中。

## CWE - 628 Function Call with Incorrectly Specified Arguments

中文：**函数调用具有错误指定的参数**

### Description

* 该产品使用未正确指定的参数调用函数，过程或例程，从而导致始终不正确的行为和由此产生的弱点。

### Extended Description

* 有多种方法可以引入这种弱点，包括：

错误的变量或参考; 参数数量不正确; 参数的顺序不正确; 错误的论点类型;要么 错误的价值。

## CWE - 636 Not Failing Securely ('Failing Open')

中文：**没有安全失败（'失败打开'）**

### Description

* 当产品遇到错误情况或故障时，其设计要求它回退到比其他可用选项安全性低的状态，例如选择最弱的加密算法或使用最宽松的访问控制限制。

### Extended Description

* 通过进入不太安全的状态，产品继承了与该状态相关的弱点，使其更容易妥协。至少，它会导致管理员产生错误的安全感。这种弱点通常是由于希望“功能失效”以最小化管理和支持成本而不是“失败安全”。

## CWE - 637 Unnecessary Complexity in Protection Mechanism (Not Using 'Economy of Mechanism')

中文：**保护机制中不必要的复杂性（不使用'机制经济'）**

### Description

* 该软件使用了比必要的更复杂的机制，当机制未被正确理解，建模，配置，实现或使用时，这可能导致最终的弱点。

### Extended Description

* 安全机制应尽可能简单。复杂的安全机制可能会导致部分实现和兼容性问题，从而导致假设和实现的安全性不匹配。这个原则的必然结果是数据规范应该尽可能简单，因为复杂的数据规范会导致复杂的验证代码。复杂的任务和系统也可能需要通过复杂的安全检查来保护，因此应该首选简单的系统。

## CWE - 638 Not Using Complete Mediation

中文：**不使用完整调解**

### Description

* 每次实体访问资源时，软件都不会对资源执行访问检查，如果该实体的权限或特权随时间发生变化，则可能会产生缺陷。

### Extended Description

* 没有详细描述

## CWE - 639 Authorization Bypass Through User-Controlled Key

中文：**授权绕过用户控制的密钥**

### Description

* 系统的授权功能不会阻止一个用户通过修改标识数据的密钥值来访问另一个用户的数据或记录。

### Extended Description

* 基于用户控制下的某个键值在系统中进行用户记录的检索。密钥通常将识别存储在系统中的用户相关记录，并且将用于查找该记录以呈现给用户。攻击者可能必须是系统中经过身份验证的用户。但是，授权过程不会正确检查数据访问操作，以确保执行操作的经过身份验证的用户具有足够的权限来执行所请求的数据访问，从而绕过系统中存在的任何其他授权检查。 例如，攻击者可以查看检索用户特定数据的地方（例如搜索屏幕）并确定正在查找的项目的密钥是否可从外部控制。键可以是HTML表单字段中的隐藏字段，可以作为URL参数或未加密的cookie变量传递，然后在每种情况下都可以篡改键值。 这种弱点的一个表现是系统使用顺序或其他易于猜测的会话ID，这将允许一个用户轻松切换到另一个用户的会话并读取/修改他们的数据。

## CWE - 640 Weak Password Recovery Mechanism for Forgotten Password

中文：**忘记密码的弱密码恢复机制**

### Description

* 该软件包含一种机制，用户可以在不知道原始密码的情况下恢复或更改密码，但机制很弱。

### Extended Description

* 应用程序通常具有一种机制，该机制为用户在忘记密码时提供访问其帐户的权限。通常，密码恢复机制很弱，其结果是使得合法系统用户以外的人更有可能获得对该用户帐户的访问权。弱密码恢复方案完全破坏了强密码认证方案。 这个弱点可能是安全问题太容易猜测或找不到答案（例如，因为问题太常见，或者可以使用社交媒体找到答案）。或者密码恢复机制代码中可能存在实现弱点，例如可能欺骗系统将新密码通过电子邮件发送到除用户之外的电子邮件帐户。密码重置率可能没有限制，因此如果攻击者试图快速连续恢复密码，则合法用户可能会被攻击者拒绝服务。系统可以将原始密码发送给用户，而不是生成新的临时密码。总之，密码恢复功能，如果不经过精心设计和实施，通常会成为系统中最薄弱的环节，可能会以允许攻击者未经授权访问系统的方式被滥用。

## CWE - 641 Improper Restriction of Names for Files and Other Resources

中文：**文件和其他资源的名称限制不当**

### Description

* 应用程序使用来自上游组件的输入构造文件或其他资源的名称，但它不限制或不正确地限制结果名称。

### Extended Description

* 这可能会产生缺陷。例如，如果这些资源的名称包含脚本字符，则如果应用程序曾在动态生成的网页上显示资源的名称，则可能会在客户端的浏览器中执行脚本。或者，如果某些应用程序解析器使用了资源，则特制名称可能会利用解析器内部的某些漏洞，从而可能导致在服务器计算机上执行任意代码。问题将根据这些格式错误的资源名称的使用情况以及是否存在漏洞或由目标技术做出的假设而有所不同，这些技术将使代码执行成为可能。

## CWE - 642 External Control of Critical State Data

中文：**关键状态数据的外部控制**

### Description

* 该软件将有关其用户或软件本身的安全关键状态信息存储在未经授权的参与者可访问的位置。

### Extended Description

* 如果攻击者可以在未经检测的情况下修改状态信息，则可以使用它来执行未经授权的操作或访问意外资源，因为应用程序员不期望可以更改状态。 状态信息可以存储在各种位置，例如cookie，隐藏的Web表单字段，输入参数或参数，环境变量，数据库记录，设置文件中等。所有这些位置都有可能被修改由攻击者当此状态信息用于控制安全性或确定资源使用情况时，它可能会创建漏洞。例如，应用程序可以执行身份验证，然后将状态保存在“authenticated = true”cookie中。攻击者可能只是创建此cookie以绕过身份验证。

## CWE - 643 Improper Neutralization of Data within XPath Expressions ('XPath Injection')

中文：**XPath表达式中的数据中和不正确（'XPath注入'）**

### Description

* 该软件使用外部输入动态构造用于从XML数据库检索数据的XPath表达式，但它不会中和或错误地中和该输入。这允许攻击者控制查询的结构。

### Extended Description

* 实际效果是攻击者可以控制从XML数据库中选择的信息，并可以使用该功能来控制应用程序流，修改逻辑，检索未经授权的数据或绕过重要检查（例如身份验证）。

## CWE - 644 Improper Neutralization of HTTP Headers for Scripting Syntax

中文：**脚本语法的HTTP标头中和不正确**

### Description

* 该应用程序不会中和或错误地中和HTTP标头中的Web脚本语法，可以由可以处理原始标头的Web浏览器组件（如Flash）使用。

### Extended Description

* 攻击者可能能够对启用了这些组件的用户进行跨站点脚本攻击和其他攻击。 如果应用程序不中和放置在来自服务器的HTTP响应的标头中的用户控制的数据，则标头可能包含将在客户端的浏览器上下文中执行的脚本，可能导致跨站点脚本漏洞或可能导致HTTP响应分裂攻击。仔细控制HTTP响应头和HTTP响应体中放置的数据非常重要，以确保不存在脚本语法，并考虑各种编码。

## CWE - 645 Overly Restrictive Account Lockout Mechanism

中文：**过度限制性账户锁定机制**

### Description

* 该软件包含一个帐户锁定保护机制，但该机制限制太多，可以轻松触发，这允许攻击者通过导致其帐户被锁定来拒绝为合法用户提供服务。

### Extended Description

* 帐户锁定是应用程序中经常出现的安全功能，作为对基于密码的系统身份验证机制的暴力攻击的对策。在一定次数的登录尝试失败后，用户的帐户可能会被禁用一段时间或者直到管理员解锁。其他安全事件也可能触发帐户锁定。但是，攻击者可能会使用此安全功能拒绝向合法系统用户提供服务。因此，确保帐户锁定安全机制不会过于严格，这一点很重要。

## CWE - 646 Reliance on File Name or Extension of Externally-Supplied File

中文：**依赖文件名或外部提供文件的扩展**

### Description

* 该软件允许上载文件，但它依赖于文件名或文件扩展名来确定适当的行为。攻击者可以使用此方法使文件被错误分类并以危险的方式处理。

### Extended Description

* 应用程序可能会使用用户提供的文件的文件名或扩展名来确定正确的操作过程，例如选择应该传递控制权的正确进程，决定应该提供哪些数据，或者应该使用哪些资源分配。如果攻击者可能导致代码错误地分类提供的文件，则可能会发生错误的操作。例如，攻击者可以提供以“.php.gif”扩展名结尾的文件，该扩展名似乎是GIF图像，但将作为PHP代码处理。在极端情况下，代码执行是可能的，但攻击者也可能导致资源耗尽，拒绝服务，调试或系统数据（包括应用程序源代码）的暴露，或绑定到特定的服务器端进程。这种弱点可能是由于Web和应用程序服务器使用的任何技术中的漏洞，由于配置错误或应用程序本身的另一个漏洞造成的。

## CWE - 647 Use of Non-Canonical URL Paths for Authorization Decisions

中文：**使用非规范URL路径进行授权决策**

### Description

* 该软件定义策略命名空间，并根据URL是规范的假设做出授权决策。这可以允许非规范URL绕过授权。

### Extended Description

* 如果应用程序定义策略命名空间并根据URL进行授权决策，但在做出授权决策之前不需要或转换为规范URL，则会打开应用程序进行攻击。例如，如果应用程序仅允许访问http://www.example.com/mypage，则攻击者可能可以使用等效的URL绕过此限制，例如：

http://WWW.EXAMPLE.COM/mypage http://www.example.com/%6Dypage（备用编码） http://192.168.1.1/mypage（IP地址） http://www.example.com/mypage/（尾随/） http://www.example.com:80/mypage

因此，重要的是指定基于路径信息的访问控制策略，其中所有备用编码都被拒绝（这可以通过默认拒绝规则来完成）。

## CWE - 648 Incorrect Use of Privileged APIs

中文：**不正确使用特权API**

### Description

* 该应用程序不符合需要额外权限的函数调用的API要求。这可能允许攻击者通过错误地调用函数来获得权限。

### Extended Description

* 当应用程序包含执行需要提升级别权限的操作的某些功能时，特权API的调用者必须小心：

确保API所做的假设是有效的，例如参数的有效性 说明API设计/实施中的已知缺陷 从安全的上下文中调用API

如果API的调用者不遵循这些要求，则可能允许恶意用户或进程提升其权限，劫持该进程或窃取敏感数据。 例如，重要的是要知道特权API在返回调用者之前是否没有脱离其特权，或者特权函数是否可以对调用者传递给它的数据，上下文或状态信息做出某些假设。始终知道何时以及如何调用特权API以确保无法利用其提升的特权级别非常重要。

## CWE - 649 Reliance on Obfuscation or Encryption of Security-Relevant Inputs without Integrity Checking

中文：**在没有完整性检查的情况下依赖于对安全相关输入进行模糊处理或加密**

### Description

* 该软件使用外部参与者不应该可变的输入的模糊处理或加密，但软件不使用完整性检查来检测这些输入是否已被修改。

### Extended Description

* 当应用程序依赖于混淆或错误应用/弱加密来保护客户端可控制的令牌或参数时，这可能会对用户状态，系统状态或服务器上的某些决策产生影响。在不保护令牌/参数的完整性的情况下，应用程序易受攻击的攻击，其中对手盲目地遍历所述令牌/参数的可能值的空间以试图获得优势。攻击者的目标是找到另一个允许的值，以某种方式提升他们在系统中的权限，披露信息或以某种有益于攻击者的方式改变系统的行为。如果应用程序不保护这些关键令牌/参数的完整性，则无法确定这些值是否已被篡改。不应依赖用于保护数据以保密的措施来提供完整性服务。

## CWE - 650 Trusting HTTP Permission Methods on the Server Side

中文：**信任服务器端的HTTP权限方法**

### Description

* 服务器包含一种保护机制，该机制假定使用HTTP GET访问的任何URI都不会导致对关联资源的状态更改。这可能允许攻击者绕过预期的访问限制并进行资源修改和删除攻击，因为某些应用程序允许GET修改状态。

### Extended Description

* HTTP GET方法和一些其他方法旨在检索资源，而不是改变服务器端的应用程序或资源的状态。此外，HTTP规范要求GET请求（和其他请求）不应有副作用。相信它足以防止意外的资源更改，应用程序可能会禁止HTTP请求对资源表示执行DELETE，PUT和POST操作。但是，HTTP协议本身没有任何内容实际上阻止HTTP GET方法执行的不仅仅是查询数据。开发人员可以轻松编写接受HTTP GET请求的程序，这些程序实际上可以在服务器上创建，更新或删除数据。例如，基于REST的Web服务的常见做法是让HTTP GET请求修改服务器端的资源。但是，只要发生这种情况，就需要在应用程序中正确实施访问控制。不应假设只有HTTP DELETE，PUT，POST和其他方法才有权改变请求中正在访问的资源的表示。

## CWE - 651 Information Exposure Through WSDL File

中文：**信息暴露通过WSDL文件**

### Description

* Web服务体系结构可能需要公开Web服务定义语言（WSDL）文件，该文件包含有关可公开访问的服务的信息以及这些服务的调用者应如何与它们交互（例如，他们期望的参数和返回的类型）。

### Extended Description

* 如果符合以下任何条件，则可能会发生信息泄露：

WSDL文件可供更广泛的受众访问。 WSDL文件包含有关不应公开访问的方法/服务的信息或有关已弃用方法的信息。由于WSDL通常是从代码自动生成的，因此更有可能出现此问题。 WSDL文件中的信息有助于猜测不应公开访问的方法/资源的名称/位置。

## CWE - 652 Improper Neutralization of Data within XQuery Expressions ('XQuery Injection')

中文：**XQuery表达式中的数据中和不正确（'XQuery Injection'）**

### Description

* 该软件使用外部输入动态构造用于从XML数据库检索数据的XQuery表达式，但它不会中和或错误地中和该输入。这允许攻击者控制查询的结构。

### Extended Description

* 实际效果是攻击者可以控制从XML数据库中选择的信息，并可以使用该功能来控制应用程序流，修改逻辑，检索未经授权的数据或绕过重要检查（例如身份验证）。

## CWE - 653 Insufficient Compartmentalization

中文：**分区不足**

### Description

* 该产品没有充分划分需要不同权限级别，权限或权限的功能或进程。

### Extended Description

* 当较低权限用户可访问的功能出现弱点时，如果没有强大的边界，攻击可能会将损害范围扩大到更高权限的用户。

## CWE - 654 Reliance on a Single Factor in a Security Decision

中文：**依赖于安全决策中的单一因素**

### Description

* 保护机制完全或在很大程度上依赖于单个条件或单个对象或实体的完整性的评估，以便做出关于授予对受限资源或功能的访问的决定。

### Extended Description

* 没有详细描述

## CWE - 655 Insufficient Psychological Acceptability

中文：**心理可接受性不足**

### Description

* 该软件具有保护机制，使用起来太困难或不方便，鼓励非恶意用户禁用或绕过该机制，无论是偶然还是故意。

### Extended Description

* 没有详细描述

## CWE - 656 Reliance on Security Through Obscurity

中文：**通过晦涩来依赖安全**

### Description

* 该软件使用一种保护机制，其强度在很大程度上取决于其隐蔽性，因此其算法或关键数据的知识足以击败该机制。

### Extended Description

* 如果攻击者能够对机制的内部工作进行逆向工程，那么对“通过默默无闻的安全”的依赖会产生由此产生的弱点。请注意，默默无闻可以是防御的一小部分，因为它可以为攻击者创造更多的工作;但是，如果将其作为主要保护手段，则存在重大风险。

## CWE - 657 Violation of Secure Design Principles

中文：**违反安全设计原则**

### Description

* 该产品违反了公认的安全设计原则。

### Extended Description

* 这可能会导致缺陷或使开发人员更容易在实现过程中引入相关的弱点。因为代码以设计为中心，所以修复设计问题可能是资源密集型的。

## CWE - 662 Improper Synchronization

中文：**同步不当**

### Description

* 该软件尝试以独占方式使用共享资源，但不会阻止或错误地阻止另一个线程或进程使用该资源。

### Extended Description

* 没有详细描述

## CWE - 663 Use of a Non-reentrant Function in a Concurrent Context

中文：**在并发上下文中使用非重入函数**

### Description

* 该软件在并发上下文中调用非重入函数，其中竞争代码序列（例如，线程或信号处理程序）可能有机会调用相同的函数或以其他方式影响其状态。

### Extended Description

* 没有详细描述

## CWE - 664 Improper Control of a Resource Through its Lifetime

中文：**资源的生命周期控制不当**

### Description

* 在整个创建，使用和发布的生命周期内，该软件不会维护或错误地保持对资源的控制。

### Extended Description

* 资源通常有关于如何创建，使用和销毁的明确说明。当软件不遵循这些说明时，可能会导致意外行为和潜在的可利用状态。 即使没有明确的指示，也应该遵守各种原则，例如“在创建完成之前不要使用对象”，或“在对象被破坏后不使用对象”。

## CWE - 665 Improper Initialization

中文：**初始化不正确**

### Description

* 该软件不会初始化或错误地初始化资源，这可能会在访问或使用资源时使资源处于意外状态。

### Extended Description

* 当关联资源预期具有某些属性或值（例如确定用户是否已经过身份验证的变量）时，这会产生安全隐患。

## CWE - 666 Operation on Resource in Wrong Phase of Lifetime

中文：**终身错误阶段的资源运作**

### Description

* 该软件在资源生命周期的错误阶段对资源执行操作，这可能导致意外行为。

### Extended Description

* 当开发人员想要初始化，使用或释放资源时，遵循有关如何操作该资源的规范并确保资源处于预期状态非常重要。在这种情况下，软件希望在资源处于其生命周期的错误阶段时对资源执行通常有效的操作，初始化，使用或释放。

## CWE - 667 Improper Locking

中文：**锁定不当**

### Description

* 该软件无法正确获取资源锁定，或者未正确释放资源上的锁定，从而导致意外的资源状态更改和行为。

### Extended Description

* 没有详细描述

## CWE - 668 Exposure of Resource to Wrong Sphere

中文：**资源暴露于错误的领域**

### Description

* 该产品将资源暴露给错误的控制范围，从而为非预期的参与者提供对资源的不适当访问。

### Extended Description

* 诸如文件和目录之类的资源可能会通过诸如不安全的权限之类的机制无意中暴露，或者当程序意外地对错误的对象进行操作时。例如，程序可能希望私有文件只能提供给特定用户。这有效地定义了一个控制范围，旨在防止攻击者访问这些私有文件。如果文件权限不安全，则除用户之外的其他方将能够访问这些文件。 单独的控制范围可能有效地要求用户只能访问私有文件，而不能访问系统上的任何其他文件。如果程序不确保用户仅请求私有文件，则用户可能能够访问系统上的其他文件。 在任何一种情况下，最终结果都是资源暴露给了错误的一方。

## CWE - 669 Incorrect Resource Transfer Between Spheres

中文：**球体之间的资源转移不正确**

### Description

* 产品无法正确地将资源/行为转移到另一个领域，或者以不当方式控制该资源的方式从另一个领域不正确地导入资源/行为。

### Extended Description

* 没有详细描述

## CWE - 670 Always-Incorrect Control Flow Implementation

中文：**始终不正确的控制流实施**

### Description

* 该代码包含一个控制流路径，该路径不反映路径要实现的算法，导致在导航此路径时出现错误行为。

### Extended Description

* 这个弱点捕获了特定代码段相对于它正在实现的算法总是不正确的情况。例如，如果C程序员打算在单个块中包含多个语句但不包括括号（CWE-483），那么逻辑总是不正确的。此问题与代码通常正常运行的大多数弱点形成对比，除非它以恶意方式进行外部操作。

## CWE - 671 Lack of Administrator Control over Security

中文：**缺乏管理员对安全性的控制**

### Description

* 该产品使用安全功能，以防止产品管理员定制安全设置以反映产品使用的环境。这会导致产生的弱点或阻止其在管理员所需的安全级别上运行。

### Extended Description

* 如果产品的管理员无法始终管理与安全相关的决策，则可能无法保护产品免受外部威胁（包括产品开发人员）的影响。例如，管理员无法更改硬编码的帐户名和密码，从而使该产品暴露给管理员无法阻止的攻击。

## CWE - 672 Operation on a Resource after Expiration or Release

中文：**到期或释放后对资源的操作**

### Description

* 在该资源过期，释放或撤销之后，该软件使用，访问或以其他方式操作资源。

### Extended Description

* 没有详细描述

## CWE - 673 External Influence of Sphere Definition

中文：**球体定义的外部影响**

### Description

* 该产品不会阻止外部参与者对控制领域的定义。

### Extended Description

* 通常，产品在代码本身内定义其控制范围，或通过产品管理员的配置定义。在某些情况下，外部方可以更改控制范围的定义。这通常是由此产生的弱点。

## CWE - 674 Uncontrolled Recursion

中文：**不受控制的递归**

### Description

* 产品无法正确控制发生的递归量，这会消耗过多的资源，例如分配的内存或程序堆栈。

### Extended Description

* 没有详细描述

## CWE - 675 Duplicate Operations on Resource

中文：**资源重复操作**

### Description

* 当操作仅应用一次时，产品对资源执行相同的操作两次或更多次。

### Extended Description

* 没有详细描述

## CWE - 676 Use of Potentially Dangerous Function

中文：**使用潜在危险函数**

### Description

* 该程序调用一个潜在的危险函数，如果使用不当可能会引入漏洞，但该函数也可以安全使用。

### Extended Description

* 没有详细描述

## CWE - 680 Integer Overflow to Buffer Overflow

中文：**整数溢出到缓冲区溢出**

### Description

* 产品执行计算以确定要分配多少内存，但可能发生整数溢出，导致分配的内存少于预期，从而导致缓冲区溢出。

### Extended Description

* 没有详细描述

## CWE - 681 Incorrect Conversion between Numeric Types

中文：**数字类型之间的转换不正确**

### Description

* 从一种数据类型转换为另一种数据类型（例如long到整数）时，可以以产生意外值的方式省略或转换数据。如果在敏感上下文中使用结果值，则可能会发生危险行为。

### Extended Description

* 没有详细描述

## CWE - 682 Incorrect Calculation

中文：**计算不正确**

### Description

* 该软件执行的计算会生成不正确或意外的结果，这些结果稍后将用于安全关键决策或资源管理。

### Extended Description

* 当软件错误地执行安全关键计算时，可能会导致不正确的资源分配，不正确的权限分配或其他事项之间的比较失败。计算错误的许多直接结果可能导致更大的问题，例如保护机制失败甚至任意代码执行。

## CWE - 683 Function Call With Incorrect Order of Arguments

中文：**函数调用具有不正确的参数顺序**

### Description

* 软件调用函数，过程或例程，但调用者以不正确的顺序指定参数，从而导致产生的弱点。

### Extended Description

* 虽然编译器可能会在某些语言中捕获此弱点，但在被调用函数接受可变数量或类型的参数（例如C中的格式字符串）的情况下，它可能会更频繁地发生。它也可能出现在语言或环境中不强制强打字。

## CWE - 684 Incorrect Provision of Specified Functionality

中文：**提供的指定功能不正确**

### Description

* 代码根据其发布的规范不起作用，可能导致错误使用。

### Extended Description

* 向外部方提供功能时，软件的行为必须符合指定的详细信息。如果没有记录细微差别的要求，则该功能可能会对调用者产生意外行为，从而可能导致可利用状态。

## CWE - 685 Function Call With Incorrect Number of Arguments

中文：**函数调用参数数量不正确**

### Description

* 软件调用函数，过程或例程，但调用者指定的参数太多或参数太少，这可能导致未定义的行为和由此产生的弱点。

### Extended Description

* 没有详细描述

## CWE - 686 Function Call With Incorrect Argument Type

中文：**函数调用具有不正确的参数类型**

### Description

* 软件调用函数，过程或例程，但调用者指定的参数是错误的数据类型，这可能导致产生的弱点。

### Extended Description

* 这种弱点最有可能发生在松散类型的语言中，或者在强类型语言中，其中在编译时不能强制执行变量参数的类型，或者存在隐式转换。

## CWE - 687 Function Call With Incorrectly Specified Argument Value

中文：**函数调用具有错误指定的参数值**

### Description

* 软件调用函数，过程或例程，但调用者指定包含错误值的参数，这可能导致产生的弱点。

### Extended Description

* 没有详细描述

## CWE - 688 Function Call With Incorrect Variable or Reference as Argument

中文：**函数调用具有不正确的变量或引用作为参数**

### Description

* 软件调用函数，过程或例程，但调用者将错误的变量或引用指定为参数之一，这可能导致未定义的行为和由此产生的弱点。

### Extended Description

* 没有详细描述

## CWE - 689 Permission Race Condition During Resource Copy

中文：**资源复制期间的权限竞争条件**

### Description

* 在复制或克隆资源时，产品在复制完成之前不会设置资源的权限或访问控制，从而在复制过程中将资源暴露给其他领域。

### Extended Description

* 没有详细描述

## CWE - 690 Unchecked Return Value to NULL Pointer Dereference

中文：**未选中的返回值为NULL指针取消引用**

### Description

* 如果函数失败，则在调用可以使用NULL指针返回的函数后，产品不会检查错误，这会导致生成的NULL指针取消引用。

### Extended Description

* 虽然未经检查的返回值弱点不仅限于NULL指针的返回（参见CWE-252中的示例），但函数通常返回NULL以指示错误状态。如果未选中此错误条件，则可能会发生NULL指针取消引用。

## CWE - 691 Insufficient Control Flow Management

中文：**控制流量管理不足**

### Description

* 代码在执行期间不能充分管理其控制流，从而创建可以以意外方式修改控制流的条件。

### Extended Description

* 没有详细描述

## CWE - 692 Incomplete Blacklist to Cross-Site Scripting

中文：**不完整的黑名单到跨站点脚本**

### Description

* 该产品使用基于黑名单的保护机制来抵御XSS攻击，但黑名单不完整，允许XSS变种成功。

### Extended Description

* 虽然XSS可能看起来很容易预防，但Web浏览器解析网页的方式差异很大，黑名单无法跟踪所有变化。 “XSS备忘单”[REF-564]包含大量旨在绕过不完整黑名单的攻击。

## CWE - 693 Protection Mechanism Failure

中文：**保护机制失败**

### Description

* 该产品不使用或错误地使用保护机制，以提供足够的防御以防止对产品的定向攻击。

### Extended Description

* 这种弱点包括三种不同的情况。当应用程序没有定义针对某类攻击的任何机制时，会发生“丢失”保护机制。 “不充分”的保护机制可能会提供一些防御 - 例如，针对最常见的攻击 - 但它并不能防止所有预期的攻击。最后，当一个机制可用并在产品中主动使用时，会发生“忽略”机制，但开发人员尚未在某些代码路径中应用它。

## CWE - 694 Use of Multiple Resources with Duplicate Identifier

中文：**使用具有重复标识符的多个资源**

### Description

* 该软件在需要唯一标识符的上下文中使用可具有相同标识符的多个资源。

### Extended Description

* 如果软件假定每个资源都具有唯一标识符，则如果攻击者可以使多个资源与同一标识符相关联，则软件可以在错误的资源上运行。

## CWE - 695 Use of Low-Level Functionality

中文：**使用低级功能**

### Description

* 该软件使用软件应该运行的框架或规范明确禁止的低级功能。

### Extended Description

* 使用低级功能可能会以意外的方式违反规范，从而有效地禁用内置保护机制，引入可利用的不一致性或以其他方式暴露攻击功能。

## CWE - 696 Incorrect Behavior Order

中文：**行为顺序不正确**

### Description

* 该软件执行多个相关行为，但行为以错误的顺序执行，可能产生结果的弱点。

### Extended Description

* 没有详细描述

## CWE - 697 Incorrect Comparison

中文：**比较不正确**

### Description

* 该软件在安全相关的上下文中比较两个实体，但比较不正确，这可能导致产生的弱点。

### Extended Description

* 这个弱点课程涵盖了几种可能性：

比较检查一个因素不正确; 比较应考虑多种因素，但根本不检查其中的一些因素; 比较检查错误的因素。

## CWE - 698 Execution After Redirect (EAR)

中文：**重定向后执行（EAR）**

### Description

* Web应用程序将重定向发送到另一个位置，但它不是退出，而是执行其他代码。

### Extended Description

* 没有详细描述

## CWE - 703 Improper Check or Handling of Exceptional Conditions

中文：**不正确的检查或处理特殊情况**

### Description

* 该软件无法正确预测或处理在软件正常运行期间很少发生的异常情况。

### Extended Description

* 没有详细描述

## CWE - 704 Incorrect Type Conversion or Cast

中文：**类型转换或转换不正确**

### Description

* 该软件无法将对象，资源或结构从一种类型正确转换为另一种类型。

### Extended Description

* 没有详细描述

## CWE - 705 Incorrect Control Flow Scoping

中文：**控制流程范围不正确**

### Description

* 在完成任务或检测到异常情况后，软件无法将控制流正确地返回到正确的位置。

### Extended Description

* 没有详细描述

## CWE - 706 Use of Incorrectly-Resolved Name or Reference

中文：**使用错误解析的名称或参考**

### Description

* 该软件使用名称或引用来访问资源，但名称/引用解析为预期控制范围之外的资源。

### Extended Description

* 没有详细描述

## CWE - 707 Improper Enforcement of Message or Data Structure

中文：**消息或数据结构的不正确执行**

### Description

* 在从上游组件读取或发送到下游组件之前，软件不强制执行或错误地强制执行结构化消息或数据格式良好。

### Extended Description

* 如果消息格式错误，则可能导致错误地解释消息。 这种弱点通常适用于产品准备另一个进程必须采取行动的控制消息（例如命令或查询）以及旨在作为数据的恶意输入可以进入控制平面的情况。然而，这种弱点也适用于并非总是存在控制影响的更一般情况。

## CWE - 708 Incorrect Ownership Assignment

中文：**所有权分配不正确**

### Description

* 软件将所有者分配给资源，但所有者不在预期的控制范围内。

### Extended Description

* 这可以允许资源由预期控制范围之外的参与者操纵。

## CWE - 710 Improper Adherence to Coding Standards

中文：**对编码标准的不当遵守**

### Description

* 该软件不遵循某些开发编码规则，这可能导致由此产生的弱点或增加相关漏洞的严重性。

### Extended Description

* 没有详细描述

## CWE - 732 Incorrect Permission Assignment for Critical Resource

中文：**关键资源的权限分配不正确**

### Description

* 该软件以允许非预期的actor读取或修改该资源的方式指定安全关键资源的权限。

### Extended Description

* 当为资源提供权限设置以提供对超出要求的更广泛的参与者的访问时，它可能导致敏感信息的暴露，或者由非预期方修改该资源。当资源与程序配置，执行或敏感用户数据相关时，这尤其危险。

## CWE - 733 Compiler Optimization Removal or Modification of Security-critical Code

中文：**编译器优化删除或修改安全关键代码**

### Description

* 开发人员在软件中构建了一个安全关键的保护机制，但编译器优化了程序，以便删除或修改该机制。

### Extended Description

* 没有详细描述

## CWE - 749 Exposed Dangerous Method or Function

中文：**暴露的危险方法或功能**

### Description

* 该软件提供了一个应用程序编程接口（API）或类似的接口，用于与外部参与者进行交互，但该接口包含一个未被适当限制的危险方法或功能。

### Extended Description

* 根据暴露方法的行为，这种弱点可能导致各种各样的弱点。它可以应用于任何数量的技术和方法，例如ActiveX控件，Java函数，IOCTL等。 暴露可以通过几种不同的方式发生：

1）功能/方法从未打算暴露给外部参与者。 2）功能/方法仅供有限的一组参与者访问，例如来自单个网站的基于因特网的访问。

## CWE - 754 Improper Check for Unusual or Exceptional Conditions

中文：**不正确检查异常或特殊情况**

### Description

* 软件不会检查或不正确地检查在软件的日常操作期间不会经常发生的异常或异常情况。

### Extended Description

* 程序员可以假设某些事件或条件永远不会发生或者不需要担心，例如内存条件低，由于限制性权限而无法访问资源，或者行为不当的客户端或组件。但是，攻击者可能会故意触发这些异常情况，从而违反程序员的假设，可能会引入不稳定，不正确的行为或漏洞。 请注意，此条目并非仅限于使用异常和异常处理，这些异常和异常处理是检查和处理异常或意外情况的机制。

## CWE - 755 Improper Handling of Exceptional Conditions

中文：**对特殊情况的处理不当**

### Description

* 该软件无法处理或错误处理异常情况。

### Extended Description

* 没有详细描述

## CWE - 756 Missing Custom Error Page

中文：**缺少自定义错误页面**

### Description

* 该软件不会向用户返回自定义错误页面，可能会泄露敏感信息。

### Extended Description

* 没有详细描述

## CWE - 757 Selection of Less-Secure Algorithm During Negotiation ('Algorithm Downgrade')

中文：**在协商过程中选择安全性较低的算法（'算法降级'）**

### Description

* 协议或其实现支持多个参与者之间的交互，并允许这些参与者协商哪个算法应该用作保护机制，例如加密或认证，但是它不选择双方都可用的最强算法。

### Extended Description

* 当可以强制安全机制降级以使用安全性较低的算法时，这可以使攻击者更容易通过利用较弱的算法来破坏软件。受害者可能不知道正在使用安全性较低的算法。例如，如果攻击者可以强制通信通道使用明文而不是强加密数据，则攻击者可以通过嗅探来读取通道，而不是通过使用强力技术尝试解密数据的额外努力。

## CWE - 758 Reliance on Undefined, Unspecified, or Implementation-Defined Behavior

中文：**依赖未定义，未指定或实现定义的行为**

### Description

* 该软件以某种方式使用API函数，数据结构或其他实体，这种方式依赖于并不总是保证为该实体保留的属性。

### Extended Description

* 当所需的属性发生变化时，例如当软件移植到不同的平台或发生交互错误（CWE-435）时，这可能导致产生的弱点。

## CWE - 759 Use of a One-Way Hash without a Salt

中文：**使用没有盐的单向哈希**

### Description

* 该软件对不应该是可逆的输入使用单向加密哈希，例如密码，但软件也不使用salt作为输入的一部分。

### Extended Description

* 这使得攻击者可以更容易地使用字典攻击技术（如彩虹表）预先计算哈希值。 应该注意的是，尽管普遍认为，使用带有哈希的好盐并不足以增加针对个人密码的攻击者的努力，或者具有大量可用计算资源的攻击者，例如云基于服务或专业，廉价的硬件。如果散列函数的计算成本不高，则脱机密码破解仍然有效;许多加密函数被设计为高效且易受使用大量计算资源的攻击，即使哈希密码强大也是如此。与其他策略（如自适应散列函数）相比，使用salt只会略微增加攻击者的计算要求。有关详细信息，请参阅CWE-916。

## CWE - 760 Use of a One-Way Hash with a Predictable Salt

中文：**使用具有可预测盐的单向哈希**

### Description

* 该软件对不应该是可逆的输入使用单向加密哈希，例如密码，但软件使用可预测的盐作为输入的一部分。

### Extended Description

* 这使得攻击者可以更容易地使用诸如彩虹表之类的字典攻击技术预先计算哈希值，从而有效地禁用不可预测的盐将提供的保护。 应该注意的是，尽管普遍认为，使用带有哈希的好盐并不足以增加针对个人密码的攻击者的努力，或者具有大量可用计算资源的攻击者，例如云基于服务或专业，廉价的硬件。如果散列函数的计算成本不高，则脱机密码破解仍然有效;许多加密函数被设计为高效且易受使用大量计算资源的攻击，即使哈希密码强大也是如此。与其他策略（如自适应散列函数）相比，使用salt只会略微增加攻击者的计算要求。有关详细信息，请参阅CWE-916。

## CWE - 761 Free of Pointer not at Start of Buffer

中文：**没有指针不在缓冲区的开始**

### Description

* 应用程序在指向堆上分配的内存资源的指针上调用free（），但指针不在缓冲区的开头。

### Extended Description

* 这可能导致应用程序崩溃，或在某些情况下，修改关键程序变量或执行代码。 当使用malloc（）系列函数之一显式地在堆上分配内存并且调用free（）时，通常会发生此弱点，但指针算法导致指针位于缓冲区的内部或末尾。

## CWE - 762 Mismatched Memory Management Routines

中文：**不匹配的内存管理例程**

### Description

* 应用程序尝试将内存资源返回给系统，但它调用的释放函数与最初用于分配该资源的函数不兼容。

### Extended Description

* 这种弱点通常可以描述为不匹配的内存管理例程，例如：

内存是在堆栈上自动分配的，但它是使用内存管理例程free（）（CWE-590）解除分配的，该程序用于显式分配的堆内存。 使用一组内存管理函数显式分配内存，并使用不同的集解除分配。例如，可以使用C ++中的malloc（）而不是new运算符分配内存，然后使用delete运算符取消分配。

当内存管理功能不匹配时，后果可能与代码执行，内存损坏或程序崩溃一样严重。后果和易用性将根据例程的实现和被管理的对象而有所不同。

## CWE - 763 Release of Invalid Pointer or Reference

中文：**发布无效指针或引用**

### Description

* 应用程序尝试将内存资源返回给系统，但调用错误的释放函数或错误地调用适当的释放函数。

### Extended Description

* 这种弱点可以采取多种形式，例如：

通过一种存储器管理方法明确地或隐含地分配存储器，并使用不同的非兼容功能（CWE-762）解除分配。 选择的函数调用或内存管理例程是合适的，但它们使用不正确，例如在CWE-761中。

## CWE - 764 Multiple Locks of a Critical Resource

中文：**关键资源的多重锁定**

### Description

* 该软件锁定关键资源的次数超过预期，导致系统出现意外状态。

### Extended Description

* 当软件在并发环境中运行并重复锁定关键资源时，后果将根据锁的类型，锁的实现和受保护的资源而有所不同。在某些情况下，例如使用信号量，资源会被合并，额外的锁定调用会减少总可用池的大小，从而可能导致性能下降或拒绝服务。如果这可以由攻击者触发，则它类似于无限制锁（CWE-412）。在二进制锁的上下文中，任何重复的锁定尝试都可能永远不会成功，因为锁已经被保持并且可能无法进行。

## CWE - 765 Multiple Unlocks of a Critical Resource

中文：**多次解锁关键资源**

### Description

* 该软件解锁关键资源的次数超过预期，导致系统出现意外状态。

### Extended Description

* 当软件在并发环境中运行并重复解锁关键资源时，后果将根据锁的类型，锁的实现和受保护的资源而有所不同。在某些情况下，例如信号量，资源被合并，额外的解锁调用将增加可用资源数量的计数，当系统接近容量时可能导致崩溃或不可预测的行为。

## CWE - 766 Critical Data Element Declared Public

中文：**关键数据元素已公开声明**

### Description

* 当预期的安全策略要求其为私有时，软件会声明关键变量，字段或成员是公开的。

### Extended Description

* 此问题使得维护软件变得更加困难，因为软件通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 767 Access to Critical Private Variable via Public Method

中文：**通过公共方法访问关键私有变量**

### Description

* 该软件定义了一个读取或修改私有变量的公共方法。

### Extended Description

* 如果攻击者修改变量以包含意外值，则可能违反代码其他部分的假设。此外，如果攻击者可以读取私有变量，则可能会泄露敏感信息或使其更容易发起进一步的攻击。

## CWE - 768 Incorrect Short Circuit Evaluation

中文：**短路评估不正确**

### Description

* 该软件包含一个带有多个逻辑表达式的条件语句，其中一个非前导表达式可能产生副作用。这可能导致在执行条件之后程序中的意外状态，因为短路逻辑可以防止发生副作用。

### Extended Description

* 使用短路评估虽然在C标准中有明确定义，但可能会以引入难以检测的逻辑错误的方式改变控制流，可能在软件执行期间导致错误。如果攻击者可以发现这种不一致，那么获取对系统的任意控制可能是可利用的。 如果在正常情况下假定“或”语句的第一个条件为真，或者假定“和”语句的第一个条件为假，则任何后续条件都可能包含其自身未检测到的逻辑错误在代码审查或测试期间。 最后，短路评估的使用可能降低代码的可维护性。

## CWE - 769 DEPRECATED: Uncontrolled File Descriptor Consumption

中文：**DEPRECATED：不受控制的文件描述符消耗**

### Description

* 此条目已被弃用，因为它与CWE-774重复。所有内容均已转移至CWE-774。

### Extended Description

* 没有详细描述

## CWE - 770 Allocation of Resources Without Limits or Throttling

中文：**无限制或限制的资源分配**

### Description

* 软件代表参与者分配可重用资源或资源组，而不会对可分配的资源的大小或数量施加任何限制，这违反了该参与者的预期安全策略。

### Extended Description

* 没有详细描述

## CWE - 771 Missing Reference to Active Allocated Resource

中文：**缺少对活动分配资源的引用**

### Description

* 该软件未正确维护对已分配资源的引用，这会阻止资源被回收。

### Extended Description

* 这不一定适用于自动执行垃圾收集的语言或框架，因为删除所有引用可能会充当资源可以回收的信号。

## CWE - 772 Missing Release of Resource after Effective Lifetime

中文：**有效寿命后缺少资源释放**

### Description

* 软件在其有效生命周期结束后，即在不再需要资源之后不释放资源。

### Extended Description

* 如果资源在使用后未释放，则可能允许攻击者导致拒绝服务。

## CWE - 773 Missing Reference to Active File Descriptor or Handle

中文：**缺少对活动文件描述符或句柄的引用**

### Description

* 该软件未正确维护对文件描述符或句柄的引用，这会阻止回收文件描述符/句柄。

### Extended Description

* 这可能导致软件使用所有可用的文件描述符或句柄，这可能会阻止其他进程执行关键文件处理操作。

## CWE - 774 Allocation of File Descriptors or Handles Without Limits or Throttling

中文：**无限制或限制的文件描述符或句柄的分配**

### Description

* 该软件代表演员分配文件描述符或句柄，而不对可分配的描述符数量施加任何限制，这违反了该演员的预期安全策略。

### Extended Description

* 这可能导致软件使用所有可用的文件描述符或句柄，这可能会阻止其他进程执行关键文件处理操作。

## CWE - 775 Missing Release of File Descriptor or Handle after Effective Lifetime

中文：**缺少文件描述符的释放或在有效生命周期后处理**

### Description

* 软件在其有效生命周期结束后，即在不再需要文件描述符/句柄之后，不释放文件描述符或句柄。

### Extended Description

* 如果在使用后未释放文件描述符或句柄（通常通过显式关闭它），攻击者可以通过使用所有可用文件描述符/句柄或以其他方式阻止其他系统进程获取自己的文件描述符/句柄来导致拒绝服务。

## CWE - 776 Improper Restriction of Recursive Entity References in DTDs ('XML Entity Expansion')

中文：**DTD中递归实体引用的不正确限制（'XML实体扩展'）**

### Description

* 该软件使用XML文档并允许使用文档类型定义（DTD）定义其结构，但它不能正确控制实体的递归定义的数量。

### Extended Description

* 如果DTD包含大量嵌套或递归实体，则在解析时会导致数据爆炸式增长，从而导致拒绝服务。

## CWE - 777 Regular Expression without Anchors

中文：**没有锚点的正则表达式**

### Description

* 该软件使用正则表达式执行中和，但正则表达式未锚定，可能允许恶意或格式错误的数据泄漏。

### Extended Description

* 执行白名单验证等任务时，会检查数据并对其进行修改，以确保数据格式正确并符合安全值列表。如果未锚定正则表达式，则可能在与正则表达式匹配的任何字符串之前或之后包含恶意或格式错误的数据。允许的恶意数据类型取决于应用程序的上下文以及正则表达式中省略的锚点。

## CWE - 778 Insufficient Logging

中文：**记录不足**

### Description

* 发生安全关键事件时，软件要么不记录事件，要么在记录事件时忽略有关事件的重要详细信息。

### Extended Description

* 如果未正确记录安全关键事件（例如登录尝试失败），则可能使恶意行为更难以检测，并可能在攻击成功后阻碍取证分析。

## CWE - 779 Logging of Excessive Data

中文：**记录过多的数据**

### Description

* 该软件记录太多信息，使得日志文件难以处理，并可能阻碍恢复工作或攻击后的取证分析。

### Extended Description

* 虽然日志记录是一种很好的做法，并且非常高级别的日志记录适用于调试开发阶段，但生产环境中过多的日志记录可能会妨碍系统管理员检测异常情况的能力。这可以在尝试穿透系统时为攻击者提供掩护，使审计跟踪混乱以进行取证分析，或者使调试生产环境中的问题变得更加困难。

## CWE - 780 Use of RSA Algorithm without OAEP

中文：**使用没有OAEP的RSA算法**

### Description

* 该软件使用RSA算法，但未包含最佳非对称加密填充（OAEP），这可能会削弱加密。

### Extended Description

* 填充方案通常与加密算法一起使用，以使明文不易预测并使攻击工作复杂化。 OAEP方案通常与RSA一起使用，以消除可预测的通用文本的影响。

## CWE - 781 Improper Address Validation in IOCTL with METHOD\_NEITHER I/O Control Code

中文：**使用METHOD\_NEITHER I / O控制代码在IOCTL中进行不正确的地址验证**

### Description

* 该软件定义了一个IOCTL，它使用METHOD\_NEITHER进行I / O，但它不验证或错误地验证提供的地址。

### Extended Description

* 当IOCTL使用METHOD\_NEITHER选项进行I / O控制时，IOCTL负责验证已提供给它的地址。如果验证缺失或不正确，攻击者可以提供任意内存地址，从而导致代码执行或拒绝服务。

## CWE - 782 Exposed IOCTL with Insufficient Access Control

中文：**暴露的IOCTL访问控制不足**

### Description

* 该软件实现了一个IOCTL，其功能应该受到限制，但它没有正确实施IOCTL的访问控制。

### Extended Description

* 当IOCTL包含特权功能并且不必要地暴露时，攻击者可以通过调用IOCTL来访问此功能。即使功能是良性的，如果程序员假设IOCTL只能由受信任的进程访问，则可能很少或根本没有验证传入的数据，暴露了如果攻击者无法调用IOCTL则永远无法访问的弱点直。 IOCTL的实现在操作系统类型和版本之间会有所不同，因此攻击和预防方法可能会有很大差异。

## CWE - 783 Operator Precedence Logic Error

中文：**运算符优先逻辑错误**

### Description

* 该程序使用一个表达式，其中运算符优先级导致使用不正确的逻辑。

### Extended Description

* 虽然操作符优先级逻辑错误通常只是一个错误，但如果它们用于安全关键代码（例如进行身份验证决策），则会产生严重后果。

## CWE - 784 Reliance on Cookies without Validation and Integrity Checking in a Security Decision

中文：**依赖于Cookie而无需验证和完整性检查安全决策**

### Description

* 应用程序使用保护机制，该机制依赖于cookie的存在或值，但它不能正确确保cookie对关联用户有效。

### Extended Description

* 攻击者可以在浏览器中轻松修改cookie，也可以在浏览器外部实现客户端代码。攻击者可以通过修改cookie以包含预期值来绕过授权和身份验证等保护机制。

## CWE - 785 Use of Path Manipulation Function without Maximum-sized Buffer

中文：**使用没有最大大小缓冲区的路径操作函数**

### Description

* 该软件调用一个函数来规范化路径或文件名，但它提供的输出缓冲区小于最大可能的大小，例如PATH\_MAX。

### Extended Description

* 将不适当大小的输出缓冲区传递给路径操作函数可能会导致缓冲区溢出。这些函数包括realpath（），readlink（），PathAppend（）等。

## CWE - 786 Access of Memory Location Before Start of Buffer

中文：**缓冲区启动前访问内存位置**

### Description

* 软件使用索引或指针读取或写入缓冲区，索引或指针在缓冲区开始之前引用存储器位置。

### Extended Description

* 这通常发生在指针或其索引递减到缓冲区之前的位置，指针算术导致有效内存位置开始之前的位置或使用负索引时。

## CWE - 787 Out-of-bounds Write

中文：**越界写作**

### Description

* 软件将数据写入预期缓冲区的结束或开始之前。

### Extended Description

* 通常，这可能导致数据损坏，崩溃或代码执行。软件可以修改索引或执行引用超出缓冲区边界的存储器位置的指针算术。随后的写操作会产生未定义或意外的结果。

## CWE - 788 Access of Memory Location After End of Buffer

中文：**缓冲区结束后访问内存位置**

### Description

* 软件使用索引或指针读取或写入缓冲区，该索引或指针在缓冲区结束后引用内存位置。

### Extended Description

* 这通常发生在指针或其索引递减到缓冲区之前的位置时;当指针算术结果在缓冲区之前的位置时;或者当使用负索引时，它会在缓冲区之前生成一个位置。

## CWE - 789 Uncontrolled Memory Allocation

中文：**不受控制的内存分配**

### Description

* 产品根据不受信任的大小值分配内存，但它不验证或错误地验证大小，允许分配任意数量的内存。

### Extended Description

* 没有详细描述

## CWE - 790 Improper Filtering of Special Elements

中文：**特殊元素的过滤不当**

### Description

* 软件从上游组件接收数据，但在将特殊元素发送到下游组件之前不会过滤或错误过滤特殊元素。

### Extended Description

* 没有详细描述

## CWE - 791 Incomplete Filtering of Special Elements

中文：**特殊元素的不完全过滤**

### Description

* 该软件从上游组件接收数据，但在将特殊元素发送到下游组件之前不会完全过滤特殊元素。

### Extended Description

* 没有详细描述

## CWE - 792 Incomplete Filtering of One or More Instances of Special Elements

中文：**一个或多个特殊元素实例的不完全过滤**

### Description

* 软件从上游组件接收数据，但在将特定元素的一个或多个实例发送到下游组件之前不会完全过滤它们。

### Extended Description

* 这种性质的不完全过滤包括：

仅当存在更多时，才过滤特殊元素的单个实例，或者 不过滤所有实例或存在多个特殊元素的所有元素。

## CWE - 793 Only Filtering One Instance of a Special Element

中文：**仅过滤特殊元素的一个实例**

### Description

* 该软件从上游组件接收数据，但仅在将特定元素的单个实例发送到下游组件之前对其进行过滤。

### Extended Description

* 这种性质的不完全过滤可能是位置相关的，因为仅过滤了第一个或最后一个元素。

## CWE - 794 Incomplete Filtering of Multiple Instances of Special Elements

中文：**多特殊元素实例的不完全过滤**

### Description

* 软件从上游组件接收数据，但在将特定元素发送到下游组件之前不会过滤所有特定元素的实例。

### Extended Description

* 这种性质的不完全过滤可能适用于：

顺序元素（彼此相邻的特殊元素）或 非顺序元素（在不同位置多次出现的特殊元素）。

## CWE - 795 Only Filtering Special Elements at a Specified Location

中文：**仅在指定位置过滤特殊元素**

### Description

* 该软件从上游组件接收数据，但仅考虑指定位置的特殊元素，从而丢失在将其发送到下游组件之前可能存在的剩余特殊元素。

### Extended Description

* 过滤器可能仅在特殊元素出现时考虑它们的实例：

相对于标记（例如“在字符串的开头/结尾;第二个参数”），或 在绝对位置（例如“字节数10”）。

这可能会在数据中留下与过滤器位置不匹配的特殊元素，但仍可能存在危险。

## CWE - 796 Only Filtering Special Elements Relative to a Marker

中文：**仅过滤相对于标记的特殊元素**

### Description

* 软件从上游组件接收数据，但仅考虑相对于标记定位的特殊元素（例如“在字符串的开头/结尾;第二个参数”），从而丢失在将其发送到之前可能存在的剩余特殊元素下游组件。

### Extended Description

* 没有详细描述

## CWE - 797 Only Filtering Special Elements at an Absolute Position

中文：**仅在绝对位置过滤特殊元素**

### Description

* 软件从上游组件接收数据，但仅考虑绝对位置处的特殊元素（例如“字节数10”），从而丢失在将其发送到下游组件之前可能存在的剩余特殊元素。

### Extended Description

* 没有详细描述

## CWE - 798 Use of Hard-coded Credentials

中文：**使用硬编码凭证**

### Description

* 该软件包含硬编码凭证，例如密码或加密密钥，用于其自身的入站身份验证，与外部组件的出站通信或内部数据的加密。

### Extended Description

* 硬编码凭证通常会创建一个重要漏洞，允许攻击者绕过软件管理员配置的身份验证。系统管理员可能难以检测到此漏洞。即使被检测到，也很难修复，因此管理员可能被迫完全禁用该产品。主要有两种变化：

入站：该软件包含一种身份验证机制，可根据硬编码凭据集检查输入凭据。 出站：软件连接到另一个系统或组件，它包含用于连接到该组件的硬编码凭据。

在入站变体中，将创建默认管理帐户，并将简单密码硬编码到产品中并与该帐户关联。此硬编码密码对于产品的每次安装都是相同的，并且系统管理员通常无法在不手动修改程序或修补软件的情况下更改或禁用密码。如果密码被发现或发布（在Internet上很常见），那么任何知道此密码的人都可以访问该产品。最后，由于软件的所有安装都具有相同的密码，即使在不同的组织中，也可以实现诸如蠕虫之类的大规模攻击。 Outbound变体适用于使用后端服务进行身份验证的前端系统。后端服务可能需要一个可以轻松发现的固定密码。程序员可以简单地将这些后端凭证硬编码到前端软件中。该程序的任何用户都可以提取密码。具有硬编码密码的客户端系统构成了更大的威胁，因为从二进制文件中提取密码通常非常简单。

## CWE - 799 Improper Control of Interaction Frequency

中文：**交互频率控制不当**

### Description

* 该软件没有正确限制它与演员的交互次数或频率，例如传入请求的数量。

### Extended Description

* 这可以允许演员比预期更频繁地执行动作。演员可以是人或自动过程，例如病毒或机器人。这可能会导致拒绝服务，破坏程序逻辑（例如限制人们进行单一投票）或其他后果。例如，身份验证例程可能不会限制攻击者猜测密码的次数。或者，一个网站可能会进行民意调查，但只希望人们每天最多投票一次。

## CWE - 804 Guessable CAPTCHA

中文：**猜猜CAPTCHA**

### Description

* 该软件使用CAPTCHA挑战，但挑战可以被非人类演员猜测或自动识别。

### Extended Description

* 自动攻击者可以绕过对CAPTCHA挑战的预期保护，并以比人为可能更高的频率执行操作，例如发起垃圾邮件攻击。 可猜测的CAPTCHA可能有几种不同的原因：

未经模糊处理的源图像没有足够失真的音频或视频图像。 生成的问题是可以自动识别的格式，例如数学问题。 可能答案数量有限的问题，例如出生年份或喜爱的运动队。 可以使用数据库访问答案的一般知识或琐事问题，例如国家首都或受欢迎的演员。 与CAPTCHA相关联的其他数据可以提供关于其内容的提示，例如其文件名包含在CAPTCHA中使用的单词的图像。

## CWE - 805 Buffer Access with Incorrect Length Value

中文：**缓冲区访问长度值不正确**

### Description

* 该软件使用顺序操作来读取或写入缓冲区，但它使用不正确的长度值，使其访问超出缓冲区范围的内存。

### Extended Description

* 当长度值超过目标大小时，可能会发生缓冲区溢出。

## CWE - 806 Buffer Access Using Size of Source Buffer

中文：**使用源缓冲区大小的缓冲区访问**

### Description

* 在读取或写入目标缓冲区时，软件使用源缓冲区的大小，这可能导致它访问超出缓冲区范围的内存。

### Extended Description

* 当目标的大小小于源的大小时，可能会发生缓冲区溢出。

## CWE - 807 Reliance on Untrusted Inputs in a Security Decision

中文：**依赖于安全决策中的不受信任的输入**

### Description

* 应用程序使用依赖于输入的存在或值的保护机制，但输入可以由不可信的actor以绕过保护机制的方式修改。

### Extended Description

* 开发人员可能会认为不能修改cookie，环境变量和隐藏表单字段等输入。但是，攻击者可以使用自定义客户端或其他攻击来更改这些输入。可能无法检测到此更改。当根据这些输入的值进行身份验证和授权等安全决策时，攻击者可以绕过软件的安全性。 如果没有足够的加密，完整性检查或其他机制，则无法信任源自局外人的任何输入。

## CWE - 820 Missing Synchronization

中文：**缺少同步**

### Description

* 该软件以并发方式使用共享资源，但不尝试同步对资源的访问。

### Extended Description

* 如果未同步对共享资源的访问，则资源可能不处于软件期望的状态。这可能会导致意外或不安全的行为，尤其是在攻击者可以影响共享资源的情况下。

## CWE - 821 Incorrect Synchronization

中文：**同步不正确**

### Description

* 该软件以并发方式使用共享资源，但它无法正确同步对资源的访问。

### Extended Description

* 如果未正确同步对共享资源的访问，则资源可能不处于软件期望的状态。这可能会导致意外或不安全的行为，尤其是在攻击者可以影响共享资源的情况下。

## CWE - 822 Untrusted Pointer Dereference

中文：**不受信任的指针解除引用**

### Description

* 程序从不受信任的源获取值，将此值转换为指针，并取消引用结果指针。

### Extended Description

* 攻击者可以为程序不期望的内存位置提供指针。如果指针被解除引用以进行写操作，则攻击可能允许修改关键程序状态变量，导致崩溃或执行代码。如果取消引用操作用于读取，则攻击可能允许读取敏感数据，导致崩溃或将程序变量设置为意外值（因为该值将从意外的内存位置读取）。 这种弱点有几种变体，包括但不一定限于：

不受信任的值直接作为函数调用调用。 在操作系统内核或驱动程序中，“userland”和特权内存空间之间存在边界，不受信任的指针可能通过API或系统调用进入（有关此类示例，请参阅CWE-781）。 当不必接受输入时，无意中接受来自不受控制的控制范围的值。当代码最初开发为由非网络环境中的单个用户运行时，可能会发生这种情况，然后将代码移植到网络环境或以其他方式暴露给网络环境。

## CWE - 823 Use of Out-of-range Pointer Offset

中文：**使用超出范围的指针偏移**

### Description

* 程序对有效指针执行指针运算，但它使用的偏移量可以指向结果指针的有效存储器位置的预期范围之外。

### Extended Description

* 虽然指针可以包含对任意存储器位置的引用，但是程序通常仅打算使用指针来访问存储器的有限部分，例如用于访问单个阵列的连续存储器。 程序可以使用偏移量来访问存储在结构化数据中的字段或子元素。如果偏移来自不受信任的来源，是计算错误的结果，或者由于另一个错误而发生，则偏移可能超出范围。 如果攻击者可以控制或影响偏移量，使其指向结构的预期边界之外，则攻击者可能能够读取或写入程序中其他地方使用的内存位置。因此，攻击可能会改变通过程序变量访问的软件状态，导致崩溃或不稳定的行为，并可能导致代码执行。

## CWE - 824 Access of Uninitialized Pointer

中文：**访问未初始化的指针**

### Description

* 程序访问或使用尚未初始化的指针。

### Extended Description

* 如果指针包含未初始化的值，则该值可能不指向有效的内存位置。这可能导致程序读取或写入意外的内存位置，从而导致拒绝服务。如果未初始化的指针用作函数调用，则可以调用任意函数。如果攻击者可以影响指针中包含的未初始化内存部分，则可以利用此弱点执行代码或执行其他攻击。 根据内存布局，相关的内存管理行为和程序操作，攻击者可能能够影响未初始化指针的内容，从而获得对要访问的内存位置的更精细控制。

## CWE - 825 Expired Pointer Dereference

中文：**已过期指针解除引用**

### Description

* 该程序取消引用一个指针，该指针包含以前有效但不再有效的内存位置。

### Extended Description

* 当程序释放内存但它保持指向该内存的指针时，可能会在以后重新分配内存。如果访问原始指针以读取或写入数据，则可能导致程序读取或修改由不同功能或进程使用的数据。根据新分配的内存的使用方式，这可能会导致拒绝服务，信息泄露或代码执行。

## CWE - 826 Premature Release of Resource During Expected Lifetime

中文：**在预期的生命周期内过早释放资源**

### Description

* 该程序释放一个仍然打算由程序本身或另一个actor使用的资源。

### Extended Description

* 这个弱点集中在程序不应该释放资源但是无论如何都要执行释放的错误。这与程序在适当的时间释放资源的弱点不同，但它维护对资源的引用，稍后它将访问该资源。对于这个弱点，资源在后续访问时仍应有效。 当程序释放仍在使用的资源时，可能仍将对此资源进行操作，这可能在此期间被重新利用，导致类似于CWE-825的问题。后果可能包括拒绝服务，信息泄露或代码执行。

## CWE - 827 Improper Control of Document Type Definition

中文：**文件类型定义控制不当**

### Description

* 该软件不会将对文档类型定义（DTD）的引用限制为预期的控制范围。这可能允许攻击者引用任意DTD，可能导致软件暴露文件，消耗过多的系统资源，或代表攻击者执行任意http请求。

### Extended Description

* 在处理DTD时，他们可能会尝试在执行解析的计算机上读取或包含文件。如果攻击者能够控制DTD，则攻击者可能能够指定敏感资源或请求或提供恶意内容。 例如，SOAP规范禁止SOAP消息包含DTD。

## CWE - 828 Signal Handler with Functionality that is not Asynchronous-Safe

中文：**具有非异步安全功能的信号处理程序**

### Description

* 该软件定义了一个信号处理程序，它包含非异步安全的代码序列，即功能不可重入，或者可以被中断。

### Extended Description

* 这可能导致意外的系统状态，具有各种潜在后果，具体取决于上下文，包括拒绝服务和代码执行。 信号处理器通常用于中断程序的正常功能，或甚至其他信号，以便通知事件的过程。当信号处理程序使用全局或静态变量，或调用最终依赖于此类状态或其关联元数据的函数时，它可能会破坏正常功能正在使用的系统状态。这可能使程序受到竞争条件或允许攻击者导致程序状态被破坏的其他弱点。虽然拒绝服务通常是后果，但在某些情况下，可以利用这种弱点来执行代码。 有几种不同的方案可以引入此问题：

从处理程序中调用非重入函数。一个例子是malloc（），它在管理内存时修改内部全局变量。很少有函数实际上是可重入的。 代码序列（不一定是函数调用）包含全局变量或相关元数据或结构的非原子使用，可由程序的其他功能（包括其他信号处理程序）访问。通常，注册相同的功能来处理多个信号。 信号处理函数最多只能运行一次，但可以多次调用。这可能通过重复传递相同的信号，或通过传递具有相同处理函数的不同信号（CWE-831）来实现。

请注意，在某些环境或上下文中，信号处理程序可能会自行中断。 如果信号处理程序和软件的正常行为都必须对同一组状态变量进行操作，并且在正常执行对这些变量的修改过程中接收到信号，则变量可能处于不正确或损坏状态在信号处理程序执行期间，返回时可能仍然不正确或损坏。

## CWE - 829 Inclusion of Functionality from Untrusted Control Sphere

中文：**从不受信任的控制领域中包含功能**

### Description

* 软件从预期控制范围之外的源导入，要求或包含可执行功能（例如库）。

### Extended Description

* 当包括第三方功能（例如Web小部件，库或其他功能源）时，软件必须有效地信任该功能。如果没有足够的保护机制，该功能本质上可能是恶意的（无论是来自不受信任的来源，被欺骗，还是来自受信任来源的传输中的修改）。该功能还可能包含其自身的弱点，或授予对应保持对基本系统私有的其他功能和状态信息的访问权限，例如系统状态信息，敏感应用程序数据或Web应用程序的DOM。 这可能会导致许多不同的后果，具体取决于所包含的功能，但一些示例包括注入恶意软件，通过向不受信任的功能授予过多权限或权限的信息暴露，基于DOM的XSS漏洞，窃取用户的cookie或打开重定向到恶意软件（ CWE-601）。

## CWE - 830 Inclusion of Web Functionality from an Untrusted Source

中文：**从不受信任的来源中包含Web功能**

### Description

* 该软件包括来自另一个域的Web功能（例如Web小部件），这使得它在软件的域内操作，可能授予对不可信源的软件的完全访问和控制。

### Extended Description

* 在基于Web的环境中包含第三方功能是有风险的，尤其是在功能源不受信任的情况下。 即使第三方是受信任的来源，如果受信任的来源受到损害，或者如果代码在从第三方传输到软件时被修改，则软件仍可能受到攻击和恶意行为。 这种弱点在Web上的“mashup”开发中很常见，其中可能包括来自其他域的源功能。例如，可以使用“”标记插入基于Javascript的Web小部件，这会导致代码在软件的域中运行，而不是远程站点从中加载小部件。因此，包含的代码可以访问本地DOM，包括开发人员可能不希望远程站点能够访问的cookie和其他数据。 这种依赖性可能是期望的，甚至是需要的，但有时程序员并不知道存在依赖性。

## CWE - 831 Signal Handler Function Associated with Multiple Signals

中文：**与多个信号相关的信号处理器功能**

### Description

* 该软件定义了一个函数，用作多个信号的处理程序。

### Extended Description

* 虽然有时是有意和安全的，但是当使用相同的函数来处理多个信号时，如果函数使用其本地声明之外的任何状态（例如全局变量或非重入函数）或者具有任何副作用，则可能发生竞争条件。 攻击者可以发送一个调用处理函数的信号;在许多操作系统中，这通常会阻止相同的信号再次调用处理程序，至少在处理程序函数完成执行之前。但是，攻击者可以发送与同一处理函数关联的不同信号。这可能会在原始处理函数仍在执行时中断它。如果存在共享状态，则状态可能已损坏。这可能会导致各种潜在后果，具体取决于上下文，包括拒绝服务和代码执行。 当信号处理程序仅被设计为执行一次（如果有的话）时，会出现另一种很少探索的可能性。通过发送多个信号，攻击者可以多次调用该函数。这可能会产生额外的，无意的副作用。甚至可能不需要竞争条件;攻击者可以发送一个信号，等待它被处理，然后发送另一个信号。

## CWE - 832 Unlock of a Resource that is not Locked

中文：**解锁未锁定的资源**

### Description

* 该软件尝试解锁未锁定的资源。

### Extended Description

* 根据锁定功能，解锁非锁定资源可能会导致内存损坏或对资源（或其关联的元数据用于跟踪锁定）的其他修改。

## CWE - 833 Deadlock

中文：**僵局**

### Description

* 该软件包含多个线程或可执行段，它们等待彼此释放必要的锁定，从而导致死锁。

### Extended Description

* 没有详细描述

## CWE - 834 Excessive Iteration

中文：**过度迭代**

### Description

* 该软件执行迭代或循环而不充分限制循环执行的次数。

### Extended Description

* 如果迭代可能受到攻击者的影响，则此弱点可能允许攻击者消耗过多的资源，如CPU或内存。在许多情况下，循环不需要是无限的，以便导致足够的资源消耗，从而对软件或其主机系统产生不利影响;它取决于每次迭代消耗的资源量。

## CWE - 835 Loop with Unreachable Exit Condition ('Infinite Loop')

中文：**循环带有无法到达的退出条件（'无限循环'）**

### Description

* 该程序包含具有无法到达的退出条件的迭代或循环，即无限循环。

### Extended Description

* 如果循环可能受到攻击者的影响，则此弱点可能允许攻击者消耗过多的资源，如CPU或内存。

## CWE - 836 Use of Password Hash Instead of Password for Authentication

中文：**使用密码哈希而不是密码进行身份验证**

### Description

* 该软件在数据存储中记录密码哈希，从客户端接收密码哈希，并将提供的哈希与从数据存储获得的哈希进行比较。

### Extended Description

* 某些身份验证机制依赖于客户端生成密码的哈希值，可能会减少服务器上的负载或避免通过网络发送密码。但是，当客户端用于生成散列时，攻击者可以通过获取散列的副本来绕过身份验证，例如，通过使用SQL注入来破坏身份验证凭据的数据库，或利用信息泄露。然后，攻击者可以使用修改后的客户端重放被盗的哈希，而无需了解原始密码。 因此，与客户端散列的服务器端比较不提供比使用没有散列的密码更多的安全性。

## CWE - 837 Improper Enforcement of a Single, Unique Action

中文：**单一，独特行动的不当执行**

### Description

* 该软件要求参与者应该只能执行一次操作，或者只能执行一个唯一操作，但软件不会强制执行或不正确地执行此限制。

### Extended Description

* 在各种应用中，仅期望用户执行一次特定动作，例如投票，请求退款或进行购买。如果未强制执行此限制，有时可能会产生安全隐患。例如，在投票应用程序中，攻击者可以通过多次投票来尝试“填充投票箱”。如果这些投票分开计算，则攻击者可以直接影响谁赢得投票。根据软件的用途，这可能会产生重大的业务影响。

## CWE - 838 Inappropriate Encoding for Output Context

中文：**输出上下文的不适当编码**

### Description

* 在向下游组件生成输出时，软件使用或指定编码，但指定的编码与下游组件预期的编码不同。

### Extended Description

* 这种弱点可能导致下游组件使用解码方法，该方法产生的数据与软件要发送的数据不同。当使用错误的编码时 - 即使密切相关 - 下游组件可能会错误地解码数据。当控制和数据之间提供的边界被无意中断时，这会产生安全性后果，因为结果数据可能会引入控制字符或软件未发送的特殊元素。然后，结果数据可用于绕过保护机制，例如输入验证，并启用注入攻击。 虽然使用输出编码对于确保组件之间的通信是准确的至关重要，但使用错误的编码 - 即使密切相关 - 可能会导致下游组件误解输出。 例如，HTML实体编码用于网页的HTML主体中的元素。但是，程序员可能在生成输出时使用实体编码，该输出用于HTML标记的属性，该标记可能包含不受HTML编码影响的功能Javascript。 虽然Web应用程序最受关注此问题，但这种弱点可能适用于使用可支持多种编码的通信流的任何类型的软件。

## CWE - 839 Numeric Range Comparison Without Minimum Check

中文：**没有最小检查的数字范围比较**

### Description

* 程序检查一个值以确保它小于或等于最大值，但它也不验证该值是否大于或等于最小值。

### Extended Description

* 某些程序使用有符号整数或浮点数，即使它们的值仅为正数或0时。输入验证检查可能会假定该值为正，并且仅检查最大值。如果值为负，但代码假定值为正，则可能产生错误。如果负值用于内存分配，阵列访问，缓冲区访问等，则该错误可能具有安全性后果。最终，该错误可能导致缓冲区溢出或其他类型的内存损坏。 在仅有正面的上下文中使用负数可能会对其他类型的资源产生安全影响。例如，购物车可能会检查用户没有请求超过10个项目，但是对-3项目的请求可能会导致应用程序计算负价格并记入攻击者的帐户。

## CWE - 841 Improper Enforcement of Behavioral Workflow

中文：**行为工作流程的不当执行**

### Description

* 该软件支持一个会话，其中一个actor必须执行多个行为，但它不能正确确保actor按所需顺序执行行为。

### Extended Description

* 通过以意外顺序执行操作或通过省略步骤，攻击者可以操纵软件的业务逻辑或使其进入无效状态。在某些情况下，这也可能暴露出由此产生的弱点。 例如，文件共享协议可能要求actor在能够传输文件之前执行单独的步骤以提供用户名，然后提供密码。如果文件共享服务器接受密码命令后跟传输命令，但未提供任何用户名，则软件可能仍会执行传输。 请注意，这与CWE-696不同，后者侧重于软件以错误顺序执行操作的时间;这个条目密切相关，但它的重点是确保参与者以正确的顺序执行操作。 与工作流程相关的行为包括：

步骤按预期顺序执行。 不会省略所需的步骤。 步骤不会中断。 步骤及时进行。

## CWE - 842 Placement of User into Incorrect Group

中文：**将用户放置到不正确的组中**

### Description

* 软件或管理员将用户置于不正确的组中。

### Extended Description

* 如果不正确的组具有比预期组更多的访问权限或特权，则用户可能能够绕过预期的安全策略来访问意外资源或执行意外操作。访问控制系统可能无法检测到此组成员身份的恶意使用情况。

## CWE - 843 Access of Resource Using Incompatible Type ('Type Confusion')

中文：**使用不兼容类型访问资源（'类型混淆'）**

### Description

* 程序使用一种类型分配或初始化诸如指针，对象或变量之类的资源，但稍后它使用与原始类型不兼容的类型访问该资源。

### Extended Description

* 当程序使用不兼容类型访问资源时，这可能会触发逻辑错误，因为资源没有预期的属性。在没有内存安全性的语言中，例如C和C ++，类型混淆可能导致越界内存访问。 虽然在C中解析具有许多不同嵌入对象类型的数据时，这种弱点经常与联合相关联，但是它可以存在于可以以多种方式解释相同变量或存储器位置的任何应用程序中。 这个弱点并不是C和C ++独有的。例如，可以通过在期望标量时提供数组参数来触发PHP应用程序中的错误，反之亦然。像Perl这样的语言，当它被访问时，它执行一种类型的变量的自动转换，就像它是另一种类型一样，也可以包含这些问题。

## CWE - 862 Missing Authorization

中文：**缺少授权**

### Description

* 当actor尝试访问资源或执行操作时，该软件不执行授权检查。

### Extended Description

* 假设具有给定身份的用户，授权是基于用户的权限和适用于资源的任何权限或其他访问控制规范来确定该用户是否可以访问给定资源的过程。 如果未应用访问控制检查，则用户可以访问数据或执行不应允许执行的操作。这可能导致各种各样的问题，包括信息泄露，拒绝服务和任意代码执行。

## CWE - 863 Incorrect Authorization

中文：**授权不正确**

### Description

* 当actor尝试访问资源或执行操作时，软件会执行授权检查，但它不能正确执行检查。这允许攻击者绕过预期的访问限制。

### Extended Description

* 假设具有给定身份的用户，授权是基于用户的权限和适用于资源的任何权限或其他访问控制规范来确定该用户是否可以访问给定资源的过程。 如果未正确应用访问控制检查，则用户可以访问数据或执行不应允许执行的操作。这可能导致各种各样的问题，包括信息泄露，拒绝服务和任意代码执行。

## CWE - 908 Use of Uninitialized Resource

中文：**使用未初始化的资源**

### Description

* 该软件使用尚未正确初始化的资源。

### Extended Description

* 当关联资源预期具有某些属性或值时，这可能会产生安全隐患。

## CWE - 909 Missing Initialization of Resource

中文：**缺少资源的初始化**

### Description

* 该软件不初始化关键资源。

### Extended Description

* 许多资源需要初始化才能正确使用。如果资源未初始化，则可能包含不可预测或过期的数据，或者可能将其初始化为无效的默认值。当资源预期具有某些属性或值时，这会产生安全隐患。

## CWE - 910 Use of Expired File Descriptor

中文：**使用过期文件描述符**

### Description

* 该软件在关闭后使用或访问文件描述符。

### Extended Description

* 在释放特定文件或设备的文件描述符之后，可以重复使用它。代码可能无法写入原始文件，因为重用的文件描述符可能引用不同的文件或设备。

## CWE - 911 Improper Update of Reference Count

中文：**引用计数更新不当**

### Description

* 该软件使用引用计数来管理资源，但它不会更新或错误地更新引用计数。

### Extended Description

* 在跟踪有多少对象包含对特定资源的引用时，例如在内存管理或垃圾回收中，可以使用引用计数。当引用计数达到零时，可以取消分配或重用资源，因为没有更多的对象使用它。如果引用计数意外地达到零，则资源可能会过早释放，即使它仍在使用中。如果所有对象不再使用该资源，但引用计数不为零，则可能永远不会释放该资源。

## CWE - 912 Hidden Functionality

中文：**隐藏的功能**

### Description

* 该软件包含未记录的功能，不是规范的一部分，并且无法通过对软件用户或管理员显而易见的界面或命令序列进行访问。

### Extended Description

* 隐藏的功能可以采取多种形式，例如故意恶意代码，包含游戏等无关功能的“复活节彩蛋”，减少维护或支持成本的开发人员友好型快捷方式，如硬编码帐户等。从安全角度来看，甚至当功能不是故意恶意或破坏时，它可能会增加软件的攻击面并暴露超出预期功能已经暴露的其他弱点。即使不容易访问，隐藏的功能也可能对修改应用程序控制流的攻击很有用。

## CWE - 913 Improper Control of Dynamically-Managed Code Resources

中文：**动态管理代码资源的不正确控制**

### Description

* 该软件不能正确地限制对动态管理的代码资源（如变量，对象，类，属性，函数或可执行指令或语句）的读取或写入。

### Extended Description

* 许多语言提供了强大的功能，允许程序员动态创建或修改现有代码或代码使用的资源，如变量和对象。虽然这些功能可以提供显着的灵活性并缩短开发时间，但如果攻击者能够以意想不到的方式直接影响这些代码资源，则会非常危险。

## CWE - 914 Improper Control of Dynamically-Identified Variables

中文：**动态识别变量的不正确控制**

### Description

* 该软件不能正确地限制对动态识别变量的读取或写入。

### Extended Description

* 许多语言都提供了强大的功能，允许程序员访问由输入字符串指定的任意变量。虽然这些功能可以提供显着的灵活性并缩短开发时间，但如果攻击者可以修改具有安全隐患的非预期变量，则这些功能非常危险。

## CWE - 915 Improperly Controlled Modification of Dynamically-Determined Object Attributes

中文：**动态确定的对象属性的不正确控制修改**

### Description

* 该软件接收来自上游组件的输入，该组件指定要在对象中初始化或更新的多个属性，属性或字段，但是它不能正确地控制可以修改哪些属性。

### Extended Description

* 如果对象包含仅供内部使用的属性，则其意外修改可能会导致漏洞。 这种弱点有时可以通过特定于语言的机制来实现，例如质量分配，自动绑定或对象注入。

## CWE - 916 Use of Password Hash With Insufficient Computational Effort

中文：**使用密码哈希计算努力不足**

### Description

* 该软件为密码生成哈希，但它使用的方案不能提供足够的计算量，使密码破解攻击不可行或昂贵。

### Extended Description

* 许多密码存储机制计算散列并存储散列，而不是将原始密码存储在明文中。在此设计中，身份验证涉及接受传入密码，计算其哈希值，并将其与存储的哈希值进行比较。 许多哈希算法被设计为以最小的开销快速执行，甚至是加密哈希。但是，这种效率是密码存储的一个问题，因为它可以减少攻击者的暴力密码破解工作量。如果攻击者可以通过其他方法（例如存储哈希值的数据库上的SQL注入）获取哈希值，则攻击者可以离线存储哈希值，并使用各种技术通过有效计算哈希值来破解密码。如果没有内置工作负载，现代攻击可以在很短的时间内使用大规模并行计算（如云计算）和GPU，ASIC计算大量哈希值，甚至耗尽所有可能密码的整个空间或FPGA硬件。在这种情况下，有效的哈希算法可以帮助攻击者。 散列方案的若干属性与其针对离线，大规模并行攻击的强度相关：

计算哈希值所需的CPU时间量（“拉伸”） 计算哈希所需的内存量（“内存难”操作） 包括随机值和密码，作为哈希计算的输入（“salting”） 给定散列，除了通过猜测可能的输入（“单向”散列）之外，没有已知的方法来确定产生该散列值的输入（例如，密码）。 相对于该方案可以生成的所有可能哈希的数量，为多个不同输入产生相同哈希的可能性很小（“抗冲突性”）

请注意，软件的安全要求可能因环境和密码值而异。不同的方案可能无法提供所有这些属性，但仍可为环境提供足够的安全性。相反，解决方案在保留一个属性方面可能非常强大，对于另一个属性的攻击仍然非常弱，或者它可能无法显着降低大规模并行攻击的效率。

## CWE - 917 Improper Neutralization of Special Elements used in an Expression Language Statement ('Expression Language Injection')

中文：**表达语言语句中使用的特殊元素的不正当中和（'表达式语言注入'）**

### Description

* 该软件使用来自上游组件的外部影响输入在Java Server Page（JSP）中构造表达式语言（EL）语句的全部或部分，但它不会中和或不正确地中和可能之前修改预期EL语句的特殊元素它被执行了。

### Extended Description

* 没有详细描述

## CWE - 918 Server-Side Request Forgery (SSRF)

中文：**服务器端请求伪造（SSRF）**

### Description

* Web服务器从上游组件接收URL或类似请求并检索此URL的内容，但它不足以确保将请求发送到预期目标。

### Extended Description

* 通过向意外的主机或端口提供URL，攻击者可以看到服务器正在发送请求，可能绕过阻止攻击者直接访问URL的防火墙等访问控制。服务器可以用作代理来对内部网络中的主机进行端口扫描，使用其他URL，例如可以访问系统上的文档（使用file：//），或使用其他协议，如gopher：//或tftp ：//，可以提供对请求内容的更大控制。

## CWE - 920 Improper Restriction of Power Consumption

中文：**不正确的功耗限制**

### Description

* 该软件在电源是有限资源的环境中运行，该资源不能自动补充，但软件不能正确地限制其操作消耗的电量。

### Extended Description

* 在诸如嵌入式或移动设备之类的环境中，电源可以是有限的资源，例如电池，其不能由软件本身自动补充，并且设备可能不总是直接连接到可靠的电源。如果软件过快地使用太多电力，则可能导致设备（以及随后的软件）在电源恢复之前停止工作，或者由于电力成本增加而增加设备所有者的财务负担。 应用程序的正常运行会消耗功率。但是，在某些情况下，攻击者可能会使应用程序消耗的功率超出预期，使用以下组件：

显示 中央处理器 磁盘I / O. 全球定位系统 声音 麦克风 USB接口

## CWE - 921 Storage of Sensitive Data in a Mechanism without Access Control

中文：**在没有访问控制的机制中存储敏感数据**

### Description

* 该软件将敏感信息存储在没有内置访问控制的文件系统或设备中。

### Extended Description

* 虽然许多现代文件系统或设备使用某种形式的访问控制来限制对数据的访问，但并非所有存储机制都具有此功能。例如，存储卡，软盘，CD和USB设备通常可供系统内的任何用户访问。当敏感数据存储在多用户环境中的这些机制中时，这可能成为问题，因为系统上的任何人都可以读取或写入此数据。 在Android设备上，外部存储通常可由设备上的其他应用程序全局读取和写入。外部存储也可以通过移动设备的USB连接轻松访问，也可以通过设备的存储卡端口进行物理访问。

## CWE - 922 Insecure Storage of Sensitive Information

中文：**敏感信息的不安全存储**

### Description

* 该软件存储敏感信息，而不适当地限制未经授权的演员的读或写访问。

### Extended Description

* 如果未正确限制读取访问权限，则攻击者可以窃取敏感信息。如果写访问权限未得到适当限制，则攻击者可以修改并可能删除数据，从而导致错误的结果和可能的拒绝服务。

## CWE - 923 Improper Restriction of Communication Channel to Intended Endpoints

中文：**通信信道对预期端点的不正确限制**

### Description

* 该软件为特权或受保护的操作建立到（或来自）端点的通信通道，但是它没有正确地确保它与正确的端点通信。

### Extended Description

* 攻击者可能能够欺骗来自不同系统或进程的预期端点，从而获得与预期端点相同的访问级别。 虽然此问题经常涉及基于网络的客户端和服务器之间的身份验证，但其他类型的通信通道和端点可能存在此缺陷。

## CWE - 924 Improper Enforcement of Message Integrity During Transmission in a Communication Channel

中文：**通信信道传输过程中消息完整性的不正确执行**

### Description

* 该软件与端点建立通信信道并从该端点接收消息，但是它不能充分确保在传输期间不修改消息。

### Extended Description

* 中间人（MITM）攻击者可能能够修改消息并欺骗端点。

## CWE - 925 Improper Verification of Intent by Broadcast Receiver

中文：**广播接收机对意图的不正确验证**

### Description

* Android应用程序使用广播接收器接收Intent但未正确验证Intent来自授权来源。

### Extended Description

* 由操作字符串标识的某些类型的Intent只能由操作系统本身广播，而不能由第三方应用程序广播。但是，当应用程序注册接收这些隐式系统意图时，它也会注册以接收任何显式意图。虽然恶意应用程序无法发送隐式系统意图，但它可以向目标应用程序发送显式意图，这可能假设任何接收到的意图是有效的隐式系统意图而不是来自其他应用程序的明确意图。这可能会导致意外行为。

## CWE - 926 Improper Export of Android Application Components

中文：**Android应用程序组件的不正确导出**

### Description

* Android应用程序导出一个供其他应用程序使用的组件，但不能正确限制哪些应用程序可以启动组件或访问它包含的数据。

### Extended Description

* 不正确地导出组件的攻击和后果可能取决于导出的组件：

如果不限制对导出的活动的访问，则任何应用程序都可以启动该活动。这可能允许恶意应用程序获得对敏感信息的访问，修改应用程序的内部状态，或欺骗用户与受害者应用程序交互，同时相信他们仍在与恶意应用程序交互。 如果不限制对导出的服务的访问，则任何应用程序都可以启动并绑定到服务。根据公开的功能，这可能允许恶意应用程序执行未经授权的操作，获取对敏感信息的访问权限或破坏应用程序的内部状态。 如果对Content Provider的访问不仅限于预期的应用程序，则恶意应用程序可能能够访问敏感数据。请注意，在4.2之前的Android中，内容提供程序会自动导出，除非它已显式声明为NOT导出。

## CWE - 927 Use of Implicit Intent for Sensitive Communication

中文：**使用隐含意图进行敏感交流**

### Description

* Android应用程序使用隐式意图将敏感数据传输到其他应用程序。

### Extended Description

* 由于隐式intent没有指定接收数据的特定应用程序，因此任何应用程序都可以通过对该意图使用Intent Filter来处理intent。这可以允许不受信任的应用程序获取敏感数据。标准广播意图有两种变体，有序和粘性。 有序广播意图按接收者声明的优先级顺序发送给一系列注册接收者。恶意接收器可以通过阻止广播在链中进一步传播而使自己具有高优先级并导致拒绝服务。还存在恶意数据修改的可能性，因为接收器也可能在将其传递给下一个接收器之前改变Intent内的数据。下游组件无法断言数据在链中的早期未被更改。 在最初的广播之后仍然可以访问粘性广播意图。旧的粘性意图将再次播放给将来注册它的任何新接收器，这大大增加了信息暴露的可能性。此外，粘性广播不能受可能适用于其他类型意图的权限的保护。 此外，任何广播意图可以包括引用接收组件通常不具有访问权限的数据的URI。意图的发送者可以包括特权，授予接收者对意图中包含的特定URI的读或写访问权。拦截此意图的恶意接收器也将获得这些权限，并能够在指定的URI处读取或写入资源。

## CWE - 939 Improper Authorization in Handler for Custom URL Scheme

中文：**自定义URL方案处理程序中的不正确授权**

### Description

* 该软件使用自定义URL方案的处理程序，但它没有正确地限制哪些actor可以使用该方案调用处理程序。

### Extended Description

* 移动平台和其他体系结构允许使用自定义URL方案来促进应用程序之间的通信。在iOS的情况下，这是进行应用程序间通信的唯一方法。该实现由开发人员自行决定，可能会在应用程序中打开安全漏洞。一个示例可能是潜在的危险功能，例如通过自定义URL方案修改文件。

## CWE - 940 Improper Verification of Source of a Communication Channel

中文：**通信渠道来源的不正确验证**

### Description

* 该软件建立一个通信通道来处理由一个actor发起的传入请求，但是它没有正确地验证该请求来自预期的来源。

### Extended Description

* 当攻击者可以成功建立来自不受信任来源的通信通道时，攻击者可能能够获得权限并访问意外功能。

## CWE - 941 Incorrectly Specified Destination in a Communication Channel

中文：**通信通道中错误指定的目的地**

### Description

* 该软件创建一个通信通道来启动对actor的传出请求，但它没有正确指定该actor的预期目标。

### Extended Description

* 目的地的攻击者可能会欺骗受信任的服务器来窃取数据或导致拒绝服务。 至少有两个明显的弱点可能导致软件与非预期目的地进行通信：

如果软件允许攻击者控制指定的目标，则攻击者可以使其连接到不受信任或恶意的目标。例如，由于UDP是无连接协议，因此可以通过在数据包中指定错误的源地址来欺骗UDP数据包;当服务器接收到数据包并发送回复时，它将通过使用传入数据包的源 - 即错误源来指定目的地。然后可以欺骗服务器将流量发送到错误的主机，这对于隐藏真实的攻击源和进行分布式拒绝服务（DDoS）是有效的。作为另一个示例，服务器端请求伪造（SSRF）和XML外部实体（XXE）可用于欺骗服务器向由于防火墙限制而无法由攻击者直接访问的主机发出传出请求。 如果软件错误地指定了目标，则可以控制此目标的攻击者可能会欺骗受信任的服务器。虽然最常见的情况可能是由于管理员的配置错误，但这可能是其他弱点造成的。例如，软件可能会错误地解析电子邮件或IP地址，并将敏感数据发送到非预期的目的地。作为另一个示例，Android应用程序可以使用“粘性广播”来与特定应用程序的接收器通信，但是由于粘性广播可以由*任何*接收器处理，这可以允许恶意应用程序访问仅受限制的数据。用于不同的应用程序。

## CWE - 942 Overly Permissive Cross-domain Whitelist

中文：**过度宽容的跨域白名单**

### Description

* 该软件使用跨域策略文件，该文件包含不应信任的域。

### Extended Description

* 跨域策略文件（Flash中的“crossdomain.xml”和Silverlight中的“clientaccesspolicy.xml”）定义域的白名单，允许服务器从中生成跨域请求。在发出跨域请求时，Flash或Silverlight客户端将首先在目标服务器上查找策略文件。如果找到，并且明确允许托管应用程序的域发出请求，则发出请求。 因此，如果跨域策略文件包含不应受信任的域（例如使用通配符时），则应用程序可能会受到这些不受信任的域的攻击。 过度宽松的策略文件允许在跨站点脚本（CWE-79）中看到许多相同的攻击。一旦用户执行了恶意Flash或Silverlight应用程序，他们就容易受到各种攻击。攻击者可以将受害者机器上的私人信息（例如可能包含会话信息的cookie）传输给攻击者。攻击者可以代表受害者向网站发送恶意请求，如果受害者具有管理该网站的管理员权限，则可能对该网站特别危险。 在许多情况下，攻击可以在受害者甚至没有意识到的情况下发起。

## CWE - 943 Improper Neutralization of Special Elements in Data Query Logic

中文：**数据查询逻辑中特殊元素的中和不正确**

### Description

* 应用程序生成一个查询，用于访问或操作数据库（如数据库）中的数据，但它不会中和或错误地中和可以修改查询的预期逻辑的特殊元素。

### Extended Description

* 根据查询语言的功能，攻击者可以在查询中注入额外的逻辑，以便：

修改预期的选择标准，从而改变返回，修改或以其他方式操纵哪些数据实体（例如，记录） 在查询中附加其他命令 返回比预期更多的实体 返回的实体少于预期 导致实体以意外方式排序

执行其他命令或更改返回哪些实体的能力具有明显的风险。但是，当应用程序逻辑依赖于实体的顺序或数量时，这也可能导致漏洞。例如，如果应用程序查询期望仅返回指定管理用户的一个实体，但攻击者可以更改返回的实体，则可能导致逻辑返回常规用户的信息并错误地认为用户具有管理权限。 虽然这种弱点最常与SQL注入相关联，但还有许多其他查询语言也受到注入攻击，包括HTSQL，LDAP，DQL，XQuery，Xpath和“NoSQL”语言。

## CWE - 1004 Sensitive Cookie Without 'HttpOnly' Flag

中文：**没有'HttpOnly'标志的敏感Cookie**

### Description

* 该软件使用cookie来存储敏感信息，但cookie没有标记HttpOnly标志。

### Extended Description

* HttpOnly标志指示兼容的浏览器阻止客户端脚本访问cookie。在Set-Cookie HTTP响应头中包含HttpOnly标志有助于降低与跨站点脚本（XSS）相关的风险，其中攻击者的脚本代码可能会尝试读取cookie的内容并泄露获得的信息。设置后，支持该标志的浏览器不会通过XSS执行的客户端脚本向第三方显示cookie的内容。

## CWE - 1007 Insufficient Visual Distinction of Homoglyphs Presented to User

中文：**提供给用户的同形视觉视觉区别不足**

### Description

* 该软件向用户显示信息或标识符，但是显示机制不容易使用户区分视觉上相似或相同的字形（同形字），这可能导致用户误解字形并执行意外的，不安全的动作。 。

### Extended Description

* 某些字形，图片或图标在语义上可以与程序不同，而与人类用户看起来非常相似或相同。这些被称为同形体。例如，小写的“l”（椭圆形）和大写的“I”（眼睛）具有不同的字符代码，但是这些字符可以以完全相同的方式显示给用户，具体取决于字体。这也可以在不同的字符集之间发生。例如，拉丁大写字母“A”和希腊大写字母“Α”（Alpha）被程序视为不同，但可以以完全相同的方式显示给用户。重音标记也可能导致字母看起来非常相似，例如拉丁语大写字母标记“À”和等效的“À”具有尖锐的重音。 攻击者可以利用这种视觉相似性来攻击诸如网络钓鱼等攻击。通过提供链接到攻击者控制的主机名，该主机名看起来像受害者信任的主机名。在对同形体的不同使用中，对手可以创建后门用户名，其在视觉上类似于常规用户的用户名，这使得系统管理员在查看日志时更难以检测恶意用户名。

## CWE - 1021 Improper Restriction of Rendered UI Layers or Frames

中文：**渲染的UI图层或框架的不正确限制**

### Description

* Web应用程序不限制或不正确地限制属于另一个应用程序或域的框架对象或UI层，这可能导致用户混淆用户正在与哪个界面进行交互。

### Extended Description

* Web应用程序应该限制是否允许在框架，iframe，对象，嵌入或applet元素中呈现它。如果没有这些限制，用户可能会被欺骗与应用程序进行交互。

## CWE - 1022 Use of Web Link to Untrusted Target with window.opener Access

中文：**使用window.opener Access将Web链接用于不受信任的目标**

### Description

* Web应用程序生成指向其控制范围之外的不受信任的外部站点的链接，但它不能正确地阻止外部站点修改window.opener对象的安全关键属性，例如location属性。

### Extended Description

* 当用户单击指向外部站点（“目标”）的链接时，target =“\_ blank”属性会导致目标站点的内容在新窗口或选项卡中打开，该窗口或选项卡在与原始页面相同的进程中运行。 window.opener对象记录有关提供链接的原始页面的信息。如果攻击者可以在目标页面上运行脚本，那么他们可以读取或修改window.opener对象的某些属性，包括location属性 - 即使原始站点和目标站点不是同一个源。攻击者可以修改位置属性以自动将用户重定向到恶意站点，例如作为网络钓鱼攻击的一部分。由于此重定向发生在原始窗口/选项卡中 - 这不一定是可见的，因为浏览器将显示重点放在新目标页面上 - 用户可能不会注意到任何可疑的重定向。

## CWE - 1023 Incomplete Comparison with Missing Factors

中文：**与缺失因素的不完全比较**

### Description

* 该软件在必须考虑每个实体的多个因素或特征的实体之间进行比较，但该比较不包括这些因素中的一个或多个。这可能导致产生的弱点，例如，通过操作错误的对象。

### Extended Description

* 没有详细描述

## CWE - 1024 Comparison of Incompatible Types

中文：**不相容类型的比较**

### Description

* 该软件执行两个实体之间的比较，但实体具有不同的，不兼容的类型，在直接比较时无法保证提供正确的结果。

### Extended Description

* 在严格类型但支持转换/转换的语言中，例如C或C ++，程序员可能会假设将一个实体转换为与另一个实体相同的类型将确保比较将正确执行，但这不能保证。在没有严格类型的语言中，例如PHP或JavaScript，可能会隐式转换/转换为程序员不知道的类型，从而导致意外结果;例如，字符串“123”可能会转换为数字类型。见例子。

## CWE - 1025 Comparison Using Wrong Factors

中文：**使用错误因素的比较**

### Description

* 该软件执行两个实体之间的比较，但比较检查实体的错误因素或特征，这可能导致不正确的结果和由此产生的弱点。

### Extended Description

* 当代码无意中提取对对象的引用而不是其相关内容时，会出现这种弱点的常见示例。

## CWE - 1037 Processor Optimization Removal or Modification of Security-critical Code

中文：**处理器优化删除或修改安全关键代码**

### Description

* 开发人员在软件中构建安全关键保护机制，但处理器优化程序的执行，以便删除或修改机制。

### Extended Description

* 没有详细描述

## CWE - 1038 Insecure Automated Optimizations

中文：**不安全的自动优化**

### Description

* 该产品使用一种自动优化代码的机制，例如改善性能等特性，但优化可能会产生可能违反预期安全假设的意外副作用。

### Extended Description

* 没有详细描述

## CWE - 1039 Automated Recognition Mechanism with Inadequate Detection or Handling of Adversarial Input Perturbations

中文：**具有不充分检测或处理对抗性输入扰动的自动识别机制**

### Description

* 该产品使用机器学习等自动机制将复杂的数据输入（例如图像或音频）识别为特定的概念或类别，但它没有正确检测或处理以导致机制的方式修改或构造的输入检测一个不同的，不正确的概念。

### Extended Description

* 当使用机器学习等技术自动对输入流进行分类，并将这些分类用于安全关键决策时，分类中的任何错误都可能引入一个漏洞，允许攻击者使产品做出错误的安全决策。如果自动机制没有开发或“训练”有足够的输入数据，那么攻击者可能能够制造故意触发错误分类的恶意输入。 目标技术包括但不一定限于：

自动语音识别 自动图像识别

例如，攻击者可能会修改道路标志或路面标记，以欺骗自动驾驶车辆误读标志/标记并执行危险行为。

## CWE - 1041 Use of Redundant Code

中文：**使用冗余代码**

### Description

* 该软件具有包含相同代码的多个功能，方法，过程，宏等。

### Extended Description

* 此问题使得维护软件变得更加困难，因为软件通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。例如，如果有两个相同代码的副本，程序员可能会修复一个副本中的弱点，而忘记在另一个副本中修复相同的弱点。

## CWE - 1042 Static Member Data Element outside of a Singleton Class Element

中文：**Singleton类元素之外的静态成员数据元素**

### Description

* 代码包含一个声明为static（但不是final）的成员元素，其中父类元素不是单例类 - 也就是说，一个只能在Create的'to'关联中使用一次的类元素行动。

### Extended Description

* 此问题可能会使软件执行速度变慢。如果攻击者可以访问相关代码，则此性能问题可能会引入漏洞。

## CWE - 1043 Data Element Aggregating an Excessively Large Number of Non-Primitive Elements

中文：**数据元素聚合过多的非原始元素**

### Description

* 该软件使用具有过多数量的子元素的数据元素，其具有非原始数据类型，例如结构或聚合对象。

### Extended Description

* 此问题可能会使软件执行速度变慢。如果攻击者可以访问相关代码，则此性能问题可能会引入漏洞。 虽然“过大”的解释可能因每个产品或开发人员而异，但CISQ建议默认使用5个子元素。

## CWE - 1044 Architecture with Number of Horizontal Layers Outside of Expected Range

中文：**具有预期范围之外的水平层数的体系结构**

### Description

* 该软件的架构包含太多 - 或太少 - 水平层。

### Extended Description

* 此问题使得维护软件变得更加困难，因为软件通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。 虽然每个产品或开发人员对“预期范围”的解释可能有所不同，但CISQ建议默认最少4层，最多8层。

## CWE - 1045 Parent Class with a Virtual Destructor and a Child Class without a Virtual Destructor

中文：**具有虚拟析构函数和没有虚拟析构函数的子类的父类**

### Description

* 父类具有虚拟析构函数方法，但父类具有没有虚析构函数的子类。

### Extended Description

* 此问题可能会阻止软件可靠地运行，因为孩子可能无法执行必要的销毁操作。如果攻击者可以访问相关代码，则此可靠性问题可能会引入漏洞，例如内存泄漏（CWE-401）。

## CWE - 1046 Creation of Immutable Text Using String Concatenation

中文：**使用字符串连接创建不可变文本**

### Description

* 该软件使用字符串连接操作创建不可变的文本字符串。

### Extended Description

* 与使用文本缓冲数据元素相比，该编程模式可能是低效的。 此问题可能会使软件执行速度变慢。如果攻击者可以访问相关代码，则此性能问题可能会引入漏洞。

## CWE - 1047 Modules with Circular Dependencies

中文：**具有循环依赖关系的模块**

### Description

* 该软件包含模块，其中一个模块具有循环回自身的引用，即存在循环依赖性。

### Extended Description

* 作为一个例子，使用Java，这个弱点可能表明包之间的循环。 由于模块化不足，这个问题使得维护软件变得更加困难，这通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。 此问题可能会阻止软件可靠地运行。如果攻击者可以访问相关代码，则此可靠性问题可能会引入漏洞。

## CWE - 1048 Invokable Control Element with Large Number of Outward Calls

中文：**具有大量外拨呼叫的可调用控制元素**

### Description

* 该代码包含可调用的控制元素，这些控制元素包含对可调用的上下文外部的其他应用程序对象的过多数量的引用，即过大的扇出值。

### Extended Description

* 虽然每个产品或开发人员对“过大的扇出值”的解释可能有所不同，但CISQ建议默认使用5个引用对象。 此问题使得维护软件变得更加困难，因为软件通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1049 Excessive Data Query Operations in a Large Data Table

中文：**大数据表中过多的数据查询操作**

### Description

* 该软件在大型数据表上执行大量连接和子查询的数据查询。

### Extended Description

* 此问题可能会使软件执行速度变慢。如果攻击者可以访问相关代码，则此性能问题可能会引入漏洞。 虽然每个产品或开发人员对“大数据表”和“大量连接或子查询”的解释可能有所不同，但CISQ建议对“大”数据表默认为100万行，默认最小为5个连接，以及默认的最少3个子查询。

## CWE - 1050 Excessive Platform Resource Consumption within a Loop

中文：**循环中过多的平台资源消耗**

### Description

* 该软件具有循环体或循环条件，其包含直接或间接消耗平台资源的控制元素，例如，消息传递，会话，锁或文件描述符。

### Extended Description

* 此问题可能会使软件执行速度变慢。如果攻击者可以影响循环中的迭代次数，那么此性能问题可能会通过消耗比预期更多的平台资源来允许拒绝服务。

## CWE - 1051 Initialization with Hard-Coded Network Resource Configuration Data

中文：**使用硬编码网络资源配置数据进行初始化**

### Description

* 该软件使用充当网络资源标识符的硬编码值初始化数据。

### Extended Description

* 此问题可能会阻止软件可靠地运行，例如如果它在环境中运行，则不使用硬编码的网络资源标识符。如果攻击者可以访问相关代码，则此可靠性问题可能会引入漏洞。

## CWE - 1052 Excessive Use of Hard-Coded Literals in Initialization

中文：**在初始化中过度使用硬编码文字**

### Description

* 该软件使用硬编码文字初始化数据元素，该文字不是简单的整数或静态常量元素。

### Extended Description

* 此问题使得修改或维护软件变得更加困难，这会通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1053 Missing Documentation for Design

中文：**缺少设计文档**

### Description

* 该产品没有代表其设计方式的文档。

### Extended Description

* 此问题可能使理解和维护代码变得更加困难。它可以使检测和/或修复漏洞变得更加困难和耗时。

## CWE - 1054 Invocation of a Control Element at an Unnecessarily Deep Horizontal Layer

中文：**在不必要的深水平层调用控制元素**

### Description

* 一个架构层的代码调用驻留在比相邻层更深的层的代码，即，调用跳过至少一个层，并且调用的代码不是可以从任何水平层引用的垂直实用层的一部分。

### Extended Description

* 此问题使得理解和维护软件变得更加困难，这会通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1055 Multiple Inheritance from Concrete Classes

中文：**具体类的多重继承**

### Description

* 该软件包含一个具有多个具体类继承的类。

### Extended Description

* 此问题使得维护软件变得更加困难，因为软件通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1056 Invokable Control Element with Variadic Parameters

中文：**具有可变参数的可调控制元素**

### Description

* 命名可调用或方法控制元素具有支持变量（可变参数）数量的参数或参数的签名。

### Extended Description

* 此问题可能会阻止软件可靠地运行。如果攻击者可以访问相关代码，则此可靠性问题可能会引入漏洞。 对于可变参数，手动分析确定调用哪个函数/方法可能是困难的或低效的。

## CWE - 1057 Data Access Operations Outside of Expected Data Manager Component

中文：**预期数据管理器组件之外的数据访问操作**

### Description

* 该软件根据设计要求使用专用的中央数据管理器组件，但它包含执行不使用此数据管理器的数据访问操作的代码。

### Extended Description

* 此问题可能使软件执行速度比预期慢，因为预期的中央数据管理器可能已针对性能或其他质量特性进行了明确优化。如果攻击者可以访问相关代码，则此性能问题可能会引入漏洞。

## CWE - 1058 Invokable Control Element in Multi-Thread Context with non-Final Static Storable or Member Element

中文：**具有非最终静态可存储或成员元素的多线程上下文中的可调用控制元素**

### Description

* 该代码包含在多线程环境中运行但具有不安全的非最终静态可存储或成员数据元素的函数或方法。

### Extended Description

* 此问题可能会阻止软件可靠地运行。如果攻击者可以访问相关代码，则此可靠性问题可能会引入漏洞。

## CWE - 1059 Incomplete Documentation

中文：**不完整的文档**

### Description

* 无论是纸质还是电子形式的文档都不包含产品所有相关元素的描述，例如其用法，结构，接口，设计，实现，配置，操作等。

### Extended Description

* 此问题使得维护软件变得更加困难，因为软件通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1060 Excessive Number of Inefficient Server-Side Data Accesses

中文：**过多的低效服务器端数据访问次数**

### Description

* 该软件执行太多数据查询而不使用诸如存储过程之类的有效数据处理功能。

### Extended Description

* 由于计算开销，此问题可能使软件执行速度更慢。如果攻击者可以访问相关代码，则此性能问题可能会引入漏洞。 虽然“太多数据查询”的交互可能因每个产品或开发人员而异，但CISQ建议默认最多5个数据查询用于低效的功能/过程。

## CWE - 1061 Insufficient Encapsulation

中文：**封装不足**

### Description

* 该软件没有充分隐藏数据或方法的内部表示和实现细节，这可能允许外部组件或模块意外修改数据，调用意外功能或引入程序员不想要的依赖项。

### Extended Description

* 此问题使得维护软件变得更加困难，因为软件通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1062 Parent Class with References to Child Class

中文：**引用子类的父类**

### Description

* 代码有一个父类，它包含对子类，方法或其成员的引用。

### Extended Description

* 此问题可能会阻止软件可靠地运行。如果攻击者可以访问相关代码，则此可靠性问题可能会引入漏洞。

## CWE - 1063 Creation of Class Instance within a Static Code Block

中文：**在静态代码块中创建类实例**

### Description

* 静态代码块创建类的实例。

### Extended Description

* 此模式标识了使用声明为静态的代码块中的值初始化可存储数据元素或成员数据元素的情况。 此问题可以通过在需要之前执行初始化来使软件执行得更慢。如果攻击者可以访问相关代码，则此性能问题可能会引入漏洞。

## CWE - 1064 Invokable Control Element with Signature Containing an Excessive Number of Parameters

中文：**具有包含过多参数的签名的可调用控制元素**

### Description

* 该软件包含一个函数，子例程或方法，其签名具有不必要的大量参数/参数。

### Extended Description

* 此问题使得理解和/或维护软件变得更加困难，这会通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。 同时插入“大量参数”。可能因每个产品或开发人员而异，CISQ建议默认最多7个参数/参数。

## CWE - 1065 Runtime Resource Management Control Element in a Component Built to Run on Application Servers

中文：**构建为在应用程序服务器上运行的组件中的运行时资源管理控制元素**

### Description

* 应用程序使用来自应用程序服务器的已部署组件，但它也使用低级功能/方法来管理资源，而不是应用程序服务器提供的API。

### Extended Description

* 此问题可能会阻止软件可靠地运行。如果攻击者可以访问相关代码，则此可靠性问题可能会引入漏洞。

## CWE - 1066 Missing Serialization Control Element

中文：**缺少序列化控制元素**

### Description

* 该软件包含一个可序列化的数据元素，该元素没有相关的序列化方法。

### Extended Description

* 此问题可能会阻止软件可靠地运行，例如通过触发一个例子。如果攻击者可以访问相关代码，则此可靠性问题可能会引入漏洞。 例如，数据元素的可序列化特性来自.NET中的可序列化SerializableAttribute属性以及Java中java.io.Serializable接口的继承。

## CWE - 1067 Excessive Execution of Sequential Searches of Data Resource

中文：**过度执行数据资源的序列搜索**

### Description

* 该软件包含针对SQL表或视图的数据查询，该查询以不使用索引的方式配置，并可能导致执行顺序搜索。

### Extended Description

* 此问题可能会使软件执行速度变慢。如果攻击者可以访问相关代码，则此性能问题可能会引入漏洞。

## CWE - 1068 Inconsistency Between Implementation and Documented Design

中文：**实施与文件设计之间的不一致**

### Description

* 产品的实施与相关文档中描述的设计不一致。

### Extended Description

* 此问题使得由于不一致而更难以维护软件，这通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1069 Empty Exception Block

中文：**空例外块**

### Description

* 可调用代码块包含不包含任何代码的异常处理块，即为空。

### Extended Description

* 当使用异常处理块（例如Catch和Finally块）但该块为空时，这可能会阻止软件可靠地运行。如果攻击者可以访问相关代码，则此可靠性问题可能会引入漏洞。

## CWE - 1070 Serializable Data Element Containing non-Serializable Item Elements

中文：**包含非可序列化项元素的可序列化数据元素**

### Description

* 该软件包含可序列化的可存储数据元素，如字段或成员，但数据元素包含不可序列化的成员元素。

### Extended Description

* 此问题可能会阻止软件可靠地运行。如果攻击者可以访问相关代码，则此可靠性问题可能会引入漏洞。 例如，数据元素的可序列化特性来自.NET中的可序列化SerializableAttribute属性以及Java中java.io.Serializable接口的继承。

## CWE - 1071 Empty Code Block

中文：**空代码块**

### Description

* 源代码包含一个不包含任何代码的块，即块是空的。

### Extended Description

* 空代码块可以出现在条件，函数或方法定义，异常处理程序等的主体中。虽然空代码块可能是有意的，但它也可能表示不完整的实现，意外的代码删除，意外的宏扩展等。对于某些编程语言和构造，语法可能允许空块，但块中缺少任何行为可能违反约定或API，使其成为错误。

## CWE - 1072 Data Resource Access without Use of Connection Pooling

中文：**不使用连接池的数据资源访问**

### Description

* 该软件通过数据库访问数据资源，而不使用连接池功能。

### Extended Description

* 此问题可能会使软件执行速度变慢，因为连接池允许重用连接，而无需打开和关闭新连接的开销和时间消耗。如果攻击者可以访问相关代码，则此性能问题可能会引入漏洞。

## CWE - 1073 Non-SQL Invokable Control Element with Excessive Number of Data Resource Accesses

中文：**具有过多数据资源访问的非SQL可调度控制元素**

### Description

* 该软件包含具有功能或方法的客户机，该功能或方法包含通过数据管理器发送的大量数据访问/查询，即，不使用有效的数据库功能。

### Extended Description

* 此问题可能会使软件执行速度变慢。如果攻击者可以访问相关代码，则此性能问题可能会引入漏洞。 虽然“大量数据访问/查询”的交互可能因每个产品或开发人员而异，但CISQ建议每个功能/方法默认最多2个数据访问。

## CWE - 1074 Class with Excessively Deep Inheritance

中文：**具有过度深度继承的类**

### Description

* 类具有过高的继承级别，即它具有大量父类。

### Extended Description

* 此问题使得理解和维护软件变得更加困难，这会通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。 虽然“大量父类”的交互可能因每个产品或开发人员而异，但CISQ建议默认最多7个父类。

## CWE - 1075 Unconditional Control Flow Transfer outside of Switch Block

中文：**交换机块外的无条件控制流转移**

### Description

* 该软件在诸如开关块之类的分支结构之外的代码中执行无条件控制传输（例如“goto”）。

### Extended Description

* 此问题使得维护软件变得更加困难，因为软件通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1076 Insufficient Adherence to Expected Conventions

中文：**对遵守预期公约的遵守不足**

### Description

* 产品的体系结构，源代码，设计，文档或其他工件不遵循必需的约定。

### Extended Description

* 此问题使得维护软件变得更加困难，因为软件通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1077 Floating Point Comparison with Incorrect Operator

中文：**与不正确的算子的浮点比较**

### Description

* 代码执行比较，例如两个浮点（浮点）值之间的相等性测试，但它使用不考虑精度损失可能性的比较运算符。

### Extended Description

* 使用浮点值进行数值计算 由于舍入错误，可能会生成不精确的结果。 结果，可能产生两种不同的计算 在数学上相等，但略有不同的数字 不转换为的不同位表示 相同的数学上相等的值。结果，平等 测试或其他比较可能会产生意外 结果。 此问题可能会阻止软件可靠地运行。如果攻击者可以访问相关代码，则此可靠性问题可能会引入漏洞。

## CWE - 1078 Inappropriate Source Code Style or Formatting

中文：**不恰当的源代码样式或格式**

### Description

* 源代码不符合缩进，空格，注释等所需的样式或格式。

### Extended Description

* 没有详细描述

## CWE - 1079 Parent Class without Virtual Destructor Method

中文：**没有虚拟析构函数方法的父类**

### Description

* 父类包含一个或多个子类，但父类没有虚拟析构函数方法。

### Extended Description

* 此问题可能会阻止软件由于未定义或意外行为而可靠地运行。如果攻击者可以访问相关代码，则此可靠性问题可能会引入漏洞。

## CWE - 1080 Source Code File with Excessive Number of Lines of Code

中文：**具有过多代码行的源代码文件**

### Description

* 源代码文件包含太多代码行。

### Extended Description

* 此问题使得理解和/或维护软件变得更加困难，这会通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。 对于每个产品或开发人员而言，“太多代码行”的交叉可能会有所不同，CISQ建议默认阈值为1000。

## CWE - 1082 Class Instance Self Destruction Control Element

中文：**类实例自毁控制元素**

### Description

* 代码包含一个类实例，它调用方法或函数来删除或销毁自身。

### Extended Description

* 例如，在C ++中，“删除此”将导致对象自行删除。 此问题可能会阻止软件可靠地运行。如果攻击者可以访问相关代码，则此可靠性问题可能会引入漏洞。

## CWE - 1083 Data Access from Outside Expected Data Manager Component

中文：**来自外部预期数据管理器组件的数据访问**

### Description

* 该软件旨在通过特定数据管理器组件（如关系数据库或非SQL数据库）管理数据访问，但它包含执行数据访问操作而不使用该组件的代码。

### Extended Description

* 当软件具有数据访问组件时，该设计可旨在处理通过该组件的所有数据访问操作。如果在该组件之外执行数据访问操作，则这可能表示违反了预期的设计。 此问题可能会阻止软件可靠地运行。如果攻击者可以访问相关代码，则此可靠性问题可能会引入漏洞。

## CWE - 1084 Invokable Control Element with Excessive File or Data Access Operations

中文：**具有过多文件或数据访问操作的可调用控制元素**

### Description

* 函数或方法包含太多使用数据管理器或文件资源的操作。

### Extended Description

* 此问题使得维护软件变得更加困难，因为软件通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。 虽然每个产品或开发人员对“太多操作”的交互可能会有所不同，但CISQ建议对同一数据管理器或文件默认最多7个操作。

## CWE - 1085 Invokable Control Element with Excessive Volume of Commented-out Code

中文：**具有过量注释代码的可调控制元素**

### Description

* 函数，方法，过程等包含在其体内注释掉的过多代码。

### Extended Description

* 此问题使得维护软件变得更加困难，因为软件通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。 虽然每个产品或开发人员的“过量”的交集可能会有所不同，但CISQ建议默认阈值为注释代码的2％。

## CWE - 1086 Class with Excessive Number of Child Classes

中文：**具有过多子类的类**

### Description

* 一个类包含不必要的大量子节点。

### Extended Description

* 此问题使得理解和维护软件变得更加困难，这会通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。 虽然每个产品或开发人员对“大量儿童”的交互可能有所不同，但CISQ建议默认最多10个子类。

## CWE - 1087 Class with Virtual Method without a Virtual Destructor

中文：**具有虚拟方法的类没有虚拟析构函数**

### Description

* 类包含虚方法，但该方法没有关联的虚析构函数。

### Extended Description

* 此问题可能会阻止软件可靠地运行，例如由于未定义的行为。如果攻击者可以访问相关代码，则此可靠性问题可能会引入漏洞。

## CWE - 1088 Synchronous Access of Remote Resource without Timeout

中文：**远程资源的同步访问没有超时**

### Description

* 代码具有对远程资源的同步调用，但调用没有超时，或者超时设置为无限。

### Extended Description

* 此问题可能会阻止软件可靠地运行，因为远程资源的中断可能导致软件挂起。如果攻击者可以访问相关代码，则此可靠性问题可能会引入漏洞。

## CWE - 1089 Large Data Table with Excessive Number of Indices

中文：**具有过多指数的大数据表**

### Description

* 该软件使用包含过多索引的大型数据表。

### Extended Description

* 此问题可能会使软件执行速度变慢。如果攻击者可以访问相关代码，则此性能问题可能会引入漏洞。 虽然“大数据表”和“过多索引”的交互可能因每个产品或开发人员而异，但CISQ建议“大”表的默认阈值为1000000行，默认阈值为3个索引。

## CWE - 1090 Method Containing Access of a Member Element from Another Class

中文：**包含从另一个类访问成员元素的方法**

### Description

* 类的方法执行直接访问来自另一个类的成员元素的操作。

### Extended Description

* 这个问题表明封装不良并且使得理解和维护软件变得更加困难，这通过使查找和/或修复漏洞更加困难或耗时而间接地影响安全性。它也可能使引入漏洞更容易。

## CWE - 1091 Use of Object without Invoking Destructor Method

中文：**在不调用析构函数方法的情况下使用Object**

### Description

* 该软件包含一个访问对象的方法，但以后不会调用该元素的关联的finalize /析构函数方法。

### Extended Description

* 通过保留超过必要的内存和/或其他资源，此问题可以使软件执行速度更慢。如果攻击者可以访问相关代码，则此性能问题可能会引入漏洞。

## CWE - 1092 Use of Same Invokable Control Element in Multiple Architectural Layers

中文：**在多个建筑图层中使用相同的Invokable控件元素**

### Description

* 该软件在多个体系结构层中使用相同的控制元素。

### Extended Description

* 此问题使得理解和维护软件变得更加困难，这会通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1093 Excessively Complex Data Representation

中文：**过于复杂的数据表示**

### Description

* 该软件对其数据结构或这些结构之间的相互关系使用不必要的复杂内部表示。

### Extended Description

* 此问题使得理解或维护软件变得更加困难，通过使查找和/或修复漏洞更加困难或耗时，间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1094 Excessive Index Range Scan for a Data Resource

中文：**数据资源的索引范围扫描过多**

### Description

* 该软件包含大型数据表的索引范围扫描，但扫描可以覆盖大量行。

### Extended Description

* 此问题可能会使软件执行速度变慢。如果攻击者可以访问相关代码，则此性能问题可能会引入漏洞。 虽然“大数据表”和“过多索引范围”的交互可能因每个产品或开发人员而异，但CISQ建议阈值为1000000表行，并且索引范围的阈值为10。

## CWE - 1095 Loop Condition Value Update within the Loop

中文：**循环中的循环条件值更新**

### Description

* 该软件使用具有控制流条件的循环，该循环条件基于在循环体内更新的值。

### Extended Description

* 此问题使得理解和/或维护软件变得更加困难，这会通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1096 Singleton Class Instance Creation without Proper Locking or Synchronization

中文：**没有正确锁定或同步的单例类实例创建**

### Description

* 该软件实现了Singleton设计模式，但没有使用适当的锁定或其他同步机制来确保单例类仅实例化一次。

### Extended Description

* 此问题可能会阻止软件可靠地运行，例如通过使instantion进程非线程安全并引入死锁（CWE-833）或活锁条件。如果攻击者可以访问相关代码，则此可靠性问题可能会引入漏洞。

## CWE - 1097 Persistent Storable Data Element without Associated Comparison Control Element

中文：**具有相关比较控制元素的持久可存储数据元素**

### Description

* 该软件使用可存储的数据元素，该数据元素不具有支持比较所需的所有相关功能或方法。

### Extended Description

* 例如，对于Java，持久化的类需要定义hashCode（）和equals（）方法。 由于不正确或意外的比较结果，此问题可能会阻止软件可靠地运行。如果攻击者可以访问相关代码，则此可靠性问题可能会引入漏洞。

## CWE - 1098 Data Element containing Pointer Item without Proper Copy Control Element

中文：**包含没有正确复制控制元素的指针项的数据元素**

### Description

* 代码包含一个数据元素，其指针没有关联的副本或构造函数方法。

### Extended Description

* 此问题可能会阻止软件可靠地运行。如果攻击者可以访问相关代码，则此可靠性问题可能会引入漏洞。

## CWE - 1099 Inconsistent Naming Conventions for Identifiers

中文：**标识符的命名约定不一致**

### Description

* 产品的代码，文档或其他工件不会对变量，可调用项，相关可调用项组，I / O功能，数据类型，文件名或类似类型的元素使用相同的命名约定。

### Extended Description

* 该问题使得由于不一致而更难以理解和/或维护软件，这通过使查找和/或修复漏洞更加困难或耗时而间接地影响安全性。它也可能使引入漏洞更容易。

## CWE - 1100 Insufficient Isolation of System-Dependent Functions

中文：**系统相关函数的隔离不足**

### Description

* 产品或代码不会将与系统相关的功能隔离到单独的独立模块中。

### Extended Description

* 此问题使得维护和/或移植软件变得更加困难，这会通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1101 Reliance on Runtime Component in Generated Code

中文：**依赖于生成代码中的运行时组件**

### Description

* 该产品使用自动生成的代码，如果没有特定的运行时支持组件，则无法执行。

### Extended Description

* 此问题使得维护软件变得更加困难，因为软件通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1102 Reliance on Machine-Dependent Data Representation

中文：**依赖于机器相关的数据表示**

### Description

* 该代码使用依赖于低级数据表示的数据表示或可能在不同处理器，物理机，操作系统或其他物理组件之间变化的构造。

### Extended Description

* 此问题使得维护和/或移植软件变得更加困难，这会通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1103 Use of Platform-Dependent Third Party Components

中文：**使用与平台相关的第三方组件**

### Description

* 该产品依赖于第三方软件组件，这些组件不能在所有需要的平台上提供相同的功能。

### Extended Description

* 此问题使得维护软件变得更加困难，因为软件通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1104 Use of Unmaintained Third Party Components

中文：**使用未维护的第三方组件**

### Description

* 该产品依赖于原始开发人员不支持或维护的第三方组件或原始开发人员的可信代理。

### Extended Description

* 依赖不再维护的组件可能很难或无法修复重大错误，漏洞或质量问题。实际上，未维护的代码可能会过时。 此问题使得维护软件变得更加困难，因为软件通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1105 Insufficient Encapsulation of Machine-Dependent Functionality

中文：**机器相关功能的不完全封装**

### Description

* 产品或代码使用与机器相关的功能，但它没有充分封装或隔离此功能与其余代码。

### Extended Description

* 此问题使得移植或维护软件变得更加困难，这会通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1106 Insufficient Use of Symbolic Constants

中文：**符号常量的不完全使用**

### Description

* 源代码使用可能需要随时间变化或演变的文字常量，而不是使用符号常量。

### Extended Description

* 此问题使得维护软件变得更加困难，因为软件通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1107 Insufficient Isolation of Symbolic Constant Definitions

中文：**符号常数定义不完全隔离**

### Description

* 源代码使用符号常量，但它没有充分将这些常量的定义放入更集中或隔离的位置。

### Extended Description

* 此问题使得维护软件变得更加困难，因为软件通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1108 Excessive Reliance on Global Variables

中文：**过度依赖全局变量**

### Description

* 代码的结构过于依赖于在代码中的各个点上使用或设置全局变量，而不是在更窄，更本地的上下文中保留相关信息。

### Extended Description

* 此问题使得维护软件变得更加困难，因为软件通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1109 Use of Same Variable for Multiple Purposes

中文：**为多重目的使用相同的变量**

### Description

* 该代码包含可调用的，块或其他代码元素，其中相同的变量用于控制多个唯一任务或存储多个数据实例。

### Extended Description

* 将相同变量用于多种目的可能使人更难以阅读或理解代码，从而可能隐藏其他质量问题。 此问题使得维护软件变得更加困难，因为软件通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1110 Incomplete Design Documentation

中文：**不完整的设计文档**

### Description

* 产品的设计文档没有充分描述控制流程，数据流，系统初始化，任务之间的关系，组件，基本原理或设计的其他重要方面。

### Extended Description

* 没有详细描述

## CWE - 1111 Incomplete I/O Documentation

中文：**不完整的I / O文档**

### Description

* 产品的文档没有充分定义输入，输出或系统/软件接口。

### Extended Description

* 没有详细描述

## CWE - 1112 Incomplete Documentation of Program Execution

中文：**不完整的程序执行文档**

### Description

* 该文档未完全定义用于控制或影响产品特定程序执行方式的所有机制。

### Extended Description

* 这包括环境变量，配置文件，注册表项，命令行开关或选项或系统设置。

## CWE - 1113 Inappropriate Comment Style

中文：**不恰当的评论风格**

### Description

* 源代码使用不一致或不符合产品预期标准的注释样式或格式。

### Extended Description

* 由于可读性不足而导致维护软件更加困难，因此通过使查找和/或修复漏洞更加困难或耗时，间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1114 Inappropriate Whitespace Style

中文：**不恰当的空白风格**

### Description

* 源代码包含在代码中不一致的空格，或者不符合产品的预期标准。

### Extended Description

* 此问题使得理解和维护软件变得更加困难，这会通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1115 Source Code Element without Standard Prologue

中文：**没有标准序言的源代码元素**

### Description

* 源代码包含诸如源文件之类的元素，这些元素不能始终如一地提供已为项目标准化的序言或标题。

### Extended Description

* 缺乏序言可能使准确和快速理解相关代码变得更加困难。标准序言或标题可能包含模块名称，版本号，作者，日期，目的，功能，假设，限制，准确性考虑等信息。 由于可分析性不足，此问题使得维护软件变得更加困难，这通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1116 Inaccurate Comments

中文：**不准确的评论**

### Description

* 源代码包含的注释不能准确描述或解释与注释关联的代码部分的各个方面。

### Extended Description

* 当评论没有准确反映相关的代码元素时，这可能会给审阅者带来混淆（由于不一致），或者使得验证代码正确实现预期行为变得更加困难和效率更低。 此问题使得维护软件变得更加困难，因为软件通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1117 Callable with Insufficient Behavioral Summary

中文：**可用行为概述不足来调用**

### Description

* 该代码包含一个函数或方法，其签名和/或相关的内联文档无法充分描述可调用的输入，输出，副作用，假设或返回代码。

### Extended Description

* 此问题使得维护软件变得更加困难，因为软件通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1118 Insufficient Documentation of Error Handling Techniques

中文：**错误处理技术的文档不足**

### Description

* 该文档没有充分描述用于错误处理，异常处理或类似机制的技术。

### Extended Description

* 文档可能需要涵盖多个层的错误处理技术，例如模块，可执行文件，可编译代码单元或可调用。

## CWE - 1119 Excessive Use of Unconditional Branching

中文：**过度使用无条件分支**

### Description

* 代码使用了太多无条件分支（例如“goto”）。

### Extended Description

* 此问题使得理解和/或维护软件变得更加困难，这会通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1120 Excessive Code Complexity

中文：**代码过度复杂**

### Description

* 代码太复杂，使用定义明确的定量测量计算。

### Extended Description

* 此问题使得理解和/或维护软件变得更加困难，这会通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。 此问题可能会使软件执行速度变慢。如果攻击者可以访问相关代码，则此性能问题可能会引入漏洞。

## CWE - 1121 Excessive McCabe Cyclomatic Complexity

中文：**过度的McCabe Cyclomatic复杂性**

### Description

* 该代码包含超过理想最大值的McCabe圈复杂度。

### Extended Description

* 此问题使得理解和/或维护软件变得更加困难，这会通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1122 Excessive Halstead Complexity

中文：**过度的Halstead复杂性**

### Description

* 该代码的结构使得Halstead复杂性度量超过期望的最大值。

### Extended Description

* 存在各种Halstead复杂性度量，例如程序词汇量大小或量。 此问题使得理解和/或维护软件变得更加困难，这会通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1123 Excessive Use of Self-Modifying Code

中文：**过度使用自修改代码**

### Description

* 该产品使用过多的自修改代码。

### Extended Description

* 此问题使得理解或维护软件变得更加困难，通过使查找和/或修复漏洞更加困难或耗时，间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1124 Excessively Deep Nesting

中文：**过度深度嵌套**

### Description

* 代码包含可调用或其他代码分组，其中嵌套/分支太深。

### Extended Description

* 此问题使得维护软件变得更加困难，因为软件通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1125 Excessive Attack Surface

中文：**过度攻击面**

### Description

* 该产品具有攻击面，其定量测量超过所需的最大值。

### Extended Description

* 源于软件安全性，“攻击面”度量通常反映不可信方（即潜在攻击者）可以使用的输入点和输出点的数量。较大的攻击面提供了更多的攻击场所，并为开发人员提供了更多引入漏洞的机会。在某些情况下，这项措施可能反映出安全以外的其他质量方面;例如，具有许多输入和输出的产品可能需要大量测试以改善代码覆盖。

## CWE - 1126 Declaration of Variable with Unnecessarily Wide Scope

中文：**具有不必要的宽范围的变量声明**

### Description

* 源代码在一个范围内声明一个变量，但该变量仅在较窄的范围内使用。

### Extended Description

* 此问题使得理解和/或维护软件变得更加困难，这会通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1127 Compilation with Insufficient Warnings or Errors

中文：**编译时警告或错误不足**

### Description

* 编译代码时没有启用足够的警告，这可能会阻止检测到细微的错误或质量问题。

### Extended Description

* 此问题使得维护软件变得更加困难，因为软件通过使查找和/或修复漏洞更加困难或耗时而间接影响安全性。它也可能使引入漏洞更容易。

## CWE - 1164 Irrelevant Code

中文：**不相关的代码**

### Description

* 该程序包含对执行不重要的代码，即不进行状态更改并且没有改变数据或控制流的副作用，因此删除代码对功能或正确性没有影响。

### Extended Description

* 不相关的代码可能包括死代码， 未使用的初始化，空块，可能完全的代码 因优化等原因删除

## CWE - 1173 Improper Use of Validation Framework

中文：**错误使用验证框架**

### Description

* 应用程序不使用或错误地使用源语言或独立库提供的输入验证框架。

### Extended Description

* 许多现代编码语言为开发人员提供了输入验证框架，使输入验证任务更容易，更不容易出错。这些框架将根据指定的条件自动检查所有输入，并在收到无效输入时直接执行错误处理程序。这些框架的不正确使用（即，不正确的实现或完全丢失）不能直接利用，但如果在应用程序中稍后不执行适当的输入验证，则可能导致可利用的情况。不使用提供的输入验证框架也会损害代码的可维护性，因为未来的开发人员可能无法识别用于代替验证框架的下游输入验证。

## CWE - 1174 ASP.NET Misconfiguration: Improper Model Validation

中文：**ASP.NET配置错误：模型验证不正确**

### Description

* ASP.NET应用程序不使用或错误地使用模型验证框架。

### Extended Description

* 没有详细描述

## CWE - 1176 Inefficient CPU Computation

中文：**CPU计算效率低下**

### Description

* 该程序使用的算法执行CPU计算，这些算法不如开发人员的需要那样有效，即，可以进一步优化计算。

### Extended Description

* 此问题可能使软件执行速度变慢，可能以用户可察觉的方式执行。如果攻击者可以影响必须执行的计算量，例如通过触发最坏情况的复杂性，这个性能问题可能会引入漏洞。

## CWE - 1177 Use of Prohibited Code

中文：**使用禁止代码**

### Description

* 该软件使用已明确禁止的功能，库或第三方组件，无论是开发人员还是客户。

### Extended Description

* 开发人员或客户可能出于各种原因（包括真实或可疑的漏洞）限制或消除功能，库或第三方组件的使用;难以安全使用;出口管制或许可证要求;代码过时或维护不善;内部代码被安排弃用;等等 为了降低漏洞风险，开发人员可能会维护一个程序员必须避免使用的“禁止”功能列表，因为这些功能很难或不可能安全使用。此问题还可能使软件成本更高且难以维护。