# 遍历目录

## 描述

允许用户输入来控制文件系统操作中使用的路径可以使攻击者能够访问或修改通常最终用户无法访问的受保护系统资源。 在某些情况下，用户提供的输入可以直接传递给文件系统操作，也可以连接到一个或多个固定字符串以构建完全限定的路径。

当应用程序不正确地清理用户提供的文件名中的特殊字符序列时，可能会发生路径遍历（或目录遍历）漏洞。例如，攻击者可以指定一个文件名，如“../..”/etc/passwd”，它解析到目标目录之外的文件，攻击者通常不会被授权查看该文件。

此调用包含路径操作缺陷。 函数的参数是使用不受信任的输入构造的文件名。 如果允许攻击者指定全部或部分文件名，则可能会对服务器上的文件（包括webroot之外的文件）进行未经授权的访问，这些文件通常是最终用户无法访问的。暴露程度取决于输入验证程序的有效性（如果有的话）。

## 推荐方法

假设所有用户提供的输入都是恶意的。 验证所有用户提供的输入，以确保它符合预期的格式，尽可能使用集中数据验证例程。 使用黑名单时，请确保清理例程执行足够数量的迭代以删除所有不允许的字符实例，并确保最终结果不危险。

验证所有不受信任的输入，以确保它符合预期的格式，尽可能使用集中数据验证例程。 使用黑名单时，请确保清理例程执行足够数量的迭代以删除所有不允许的字符实例。

# Clickjacking

## 基本概念

点击劫持，也称为“用户界面纠正攻击”，是指攻击者使用多个透明或不透明图层来诱骗用户在打算点击顶层页面时点击其他页面上的按钮或链接。 因此，攻击者“劫持”针对其页面的点击并将其路由到另一个页面，很可能由另一个应用程序，域或两者所有。

使用类似的技术，键击也可能被劫持。 通过精心设计的样式表，iframe和文本框组合，用户可以相信他们输入的是他们的电子邮件或银行帐户的密码，而是键入由攻击者控制的隐形框架。

## 示例

例如，想象一个攻击者建立一个网站，上面有一个按钮，上面写着“点击这里免费获取iPod”。 但是，在该网页的顶部，攻击者已经使用您的邮件帐户加载了iframe，并在“免费iPod”按钮的顶部直接排列了“删除所有邮件”按钮。 受害者试图点击“免费iPod”按钮，但实际上点击了隐藏的“删除所有消息”按钮。 从本质上讲，攻击者“劫持”了用户的点击，因此名称为“Clickjacking”。

Clickjacking最臭名昭着的例子之一是对Adobe Flash插件设置页面的攻击。 通过将此页面加载到不可见的iframe中，攻击者可以欺骗用户更改Flash的安全设置，允许任何Flash动画使用计算机的麦克风和摄像头。

## 防御点击劫持

主要有两个方法来防止点击劫持：

1、发送适当的内容安全策略（CSP）frame-ancestors指令响应标头，指示浏览器不允许从其他域进行框架。（这将替换旧的X-Frame-Options HTTP标头。）

2、在UI中使用防御性代码以确保当前帧是最顶级的窗口。

# SQL注入

## 问题描述

当数据从不受信任的源进入应用程序并用于动态构造SQL查询时，会发生SQL注入漏洞。这允许攻击者操纵数据库查询以访问，修改或删除任意数据。 根据平台，数据库类型和配置，还可以对数据库执行管理操作，访问文件系统或执行任意系统命令。SQL注入攻击还可用于破坏身份验证和授权方案，这将使攻击者能够获得对应用程序受限部分的特权访问。

## 解决方法

1、使用参数化预准备语句而不是动态构造SQL查询。这将阻止数据库将绑定变量的内容解释为查询的一部分，并且是防止SQL注入的最有效方法。

2、使用正过滤器（白名单）验证用户提供的输入，以确保它符合预期的格式，尽可能使用集中数据验证程序。

3、在应用过滤器或正则表达式或将数据提交到数据库之前，规范化所有用户提供的数据。这意味着所有URL编码（％xx），HTML编码（＆＃xx;）或其他编码方案应该减少到应用程序期望的内部字符表示。这可以防止攻击者使用备用编码绕过过滤器的方案。

4、使用Hibernate等数据库抽象库时，不要假设API公开的所有方法都会自动防止SQL注入攻击。大多数库包含以不安全的方式将任意查询传递给数据库的方法。

优先使用PreparedSQLStatement，使用它提供的占位符来填充SQL中的参数。

重新定义从外部传入的SQL变量

参考mypage-cn这个项目中的EmailCtrl类中的listMessage方法中的fromRequestMessage(message)这个方法。

removeControlCharacter(String input)主要是这个方法。

