HP Fortify Audit Workbench

Developer Workbook

配电待建物资管理系统

Table of Contents

Executive Summary
Project Description
Issue Breakdown by Fortify Categories
Results Outline

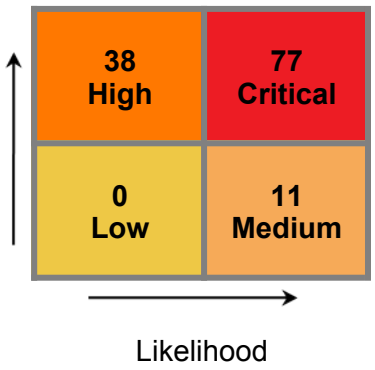
Executive Summary

This workbook is intended to provide all necessary details and information for a developer to understand and remediate the different issues discovered during the 配电待建物资管理系统 project audit. The information contained in this workbook is targeted at project managers and developers.

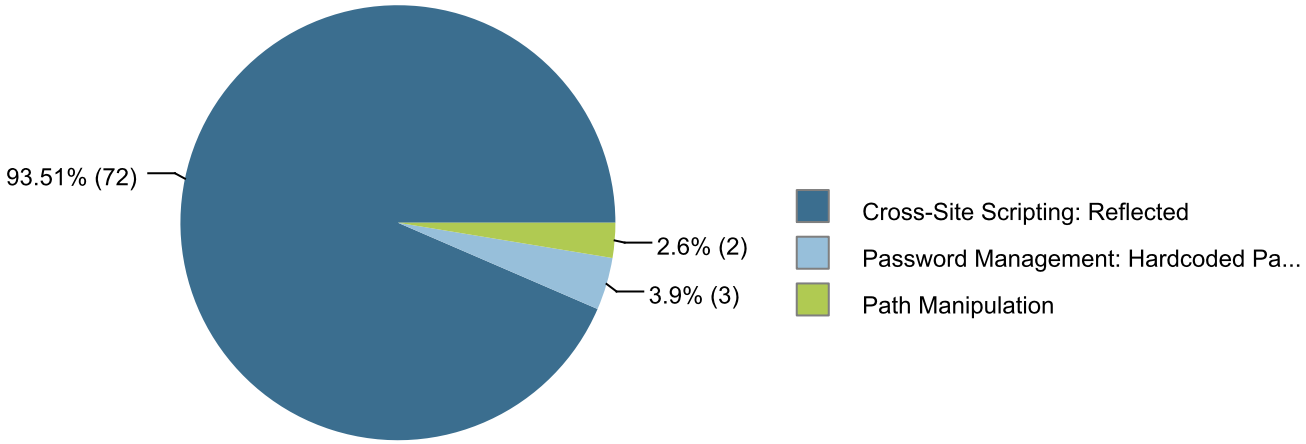
This section provides an overview of the issues uncovered during analysis.

Project Name:	配电待建物资管理系统
Project Version:	
SCA:	Results Present
WebInspect:	Results Not Present
SecurityScope:	Results Not Present
Other:	Results Not Present

Issues by Priority



Top Ten Critical Categories



Project Description

This section provides an overview of the HP Fortify scan engines used for this project, as well as the project meta-information.

SCA

Date of Last Analysis:	2016年9月26日 上午11:09	**Engine Version:**	6.40.0089
Host Name:	WIN-RPEHF1IVFSV	**Certification:**	VALID
Number of Files:	2,339	**Lines of Code:**	644,437

Issue Breakdown by Fortify Categories

The following table depicts a summary of all issues grouped vertically by Fortify Category. For each category, the total number of issues is shown by Fortify Priority Order, including information about the number of audited issues.

Category	Fortify Priority (audited/total)				Total Issues
	Critical	High	Medium	Low	
Cookie Security: HTTPOnly not Set	0	0 / 2	0	0	0 / 2
Cookie Security: HTTPOnly not Set on Session Cookie	0	0 / 2	0	0	0 / 2
Cookie Security: Persistent Session Cookie	0	0 / 4	0	0	0 / 4
Cross-Site Scripting: Poor Validation	0	0	0 / 1	0	0 / 1
Cross-Site Scripting: Reflected	0 / 72	0	0	0	0 / 72
Header Manipulation	0	0 / 8	0	0	0 / 8
Often Misused: File Upload	0	0	0 / 10	0	0 / 10
Open Redirect	0	0 / 4	0	0	0 / 4
Password Management: Hardcoded Password	0 / 3	0	0	0	0 / 3
Path Manipulation	0 / 2	0	0	0	0 / 2
Possible Variable Overwrite: Global Scope	0	0 / 13	0	0	0 / 13
Privacy Violation: Autocomplete	0	0 / 3	0	0	0 / 3
Weak Encryption	0	0 / 2	0	0	0 / 2

Results Outline

Cookie Security: HTTPOnly not Set (2 issues)

Abstract

程序创建了 cookie，但未能将 `HttpOnly` 标记设置为 `true`。

Explanation

所有主要浏览器均支持 `HttpOnly` cookie 属性，可阻止客户端脚本访问 cookie。Cross-site scripting 攻击通常会访问 cookie，以试图窃取会话标识符或 authentication 标记。如果未启用 `HttpOnly`，攻击者就能更容易地访问用户 cookie。

例 1：以下示例中的代码创建 cookie，但没有设置 `HttpOnly` 属性。

```
setcookie("emailCookie", $email, 0, "/", "www.example.com", TRUE);  //Missing 7th parameter to set HttpOnly
```

Recommendation

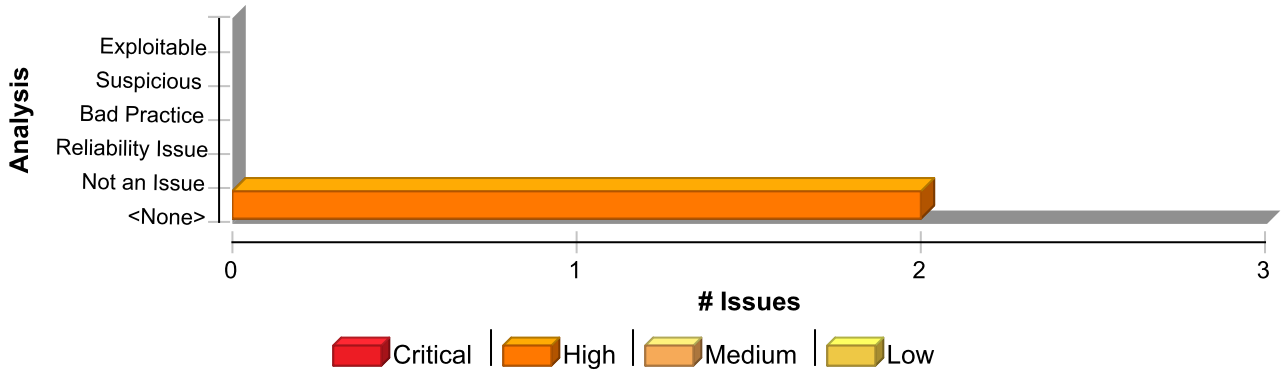
在创建 cookie 时启用 `HttpOnly` 属性。将 `setcookie()` 调用中的 `HttpOnly` 参数设置为 `true`，便可完成此配置。

例 2：以下示例中的代码创建的 cookie 与例 1 中的代码所创建的相同，但这次将 `HttpOnly` 参数设置为 `true`。

```
setcookie("emailCookie", $email, 0, "/", "www.example.com", TRUE, TRUE);
```

不要被 `HttpOnly` 欺骗进入虚假安全。由于已开发出了多种绕过它的机制，因此它并非完全有效。

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Cookie Security: HTTPOnly not Set	2	0	0	2
Total	**2**	**0**	**0**	**2**

Cookie Security: HTTPOnly not Set — High

Package: framework

framework/yiilite.php, line 2961 (Cookie Security: HTTPOnly not Set) — High

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: setcookie
Enclosing Method: addcookie()
File: framework/yiilite.php:2961
Taint Flags:

```
2958  if(version_compare(PHP_VERSION,'5.2.0','>='))
2959  setcookie($cookie->name,$value,$cookie->expire,$cookie->path,$cookie->domain,$cookie->secure,$cookie->httpOnly);
2960  else
2961  setcookie($cookie->name,$value,$cookie->expire,$cookie->path,$cookie->domain,$cookie->secure);
2962  }
2963  protected function removeCookie($cookie)
2964  {
```

framework/yiilite.php, line 2968 (Cookie Security: HTTPOnly not Set) — High

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: setcookie
Enclosing Method: removecookie()
File: framework/yiilite.php:2968
Taint Flags:

```
2965  if(version_compare(PHP_VERSION,'5.2.0','>='))
2966  setcookie($cookie->name,'',0,$cookie->path,$cookie->domain,$cookie->secure,$cookie->httpOnly);
2967  else
2968  setcookie($cookie->name,'',0,$cookie->path,$cookie->domain,$cookie->secure);
2969  }
2970 }
2971 class CUrlManager extends CApplicationComponent
```

Cookie Security: HTTPOnly not Set on Session Cookie (2 issues)

Abstract

程序创建了 cookie，但未能将 `HttpOnly` 标记设置为 `true`。

Explanation

所有主要浏览器均支持 `HttpOnly` cookie 属性，可阻止客户端脚本访问 cookie。Cross-site scripting 攻击通常会访问 cookie，以试图窃取会话标识符或 authentication 标记。如果未启用 `HttpOnly`，攻击者就能更容易地访问用户 cookie。

例 1：以下示例中的代码将创建会话 cookie，但未将 `HttpOnly` 参数设置为 `true`。

```
session_set_cookie_params(0, "/", "www.example.com", true, false);
```

Recommendation

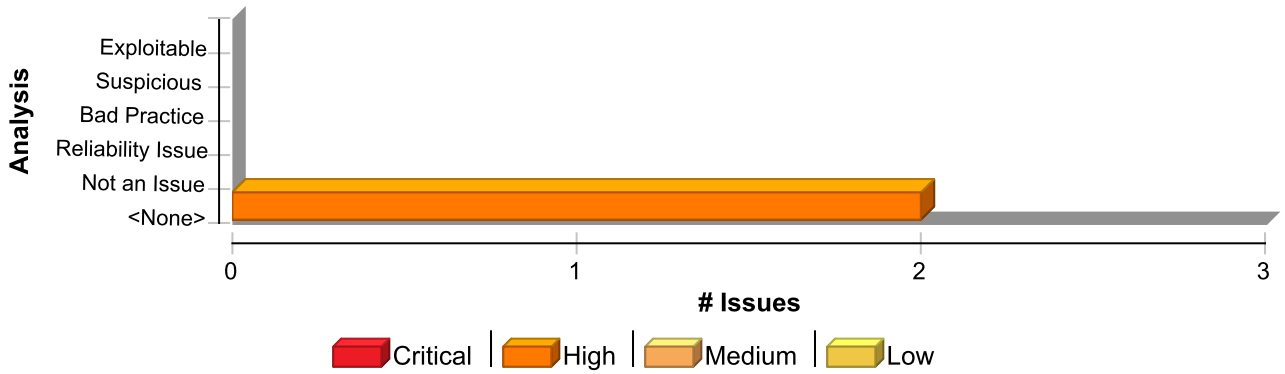
在创建会话 cookie 时启用 `HttpOnly` 属性。将 `session_set_cookie_params()` 调用中的 `HttpOnly` 参数设置为 `true`，便可完成此配置。

例 2：以下示例中的代码创建的 cookie 与例 1 中的代码所创建的相同，但这次将 `HttpOnly` 参数设置为 `true`。

```
session_set_cookie_params(0, "/", "www.example.com", true, true);
```

不要被 `HttpOnly` 欺骗进入虚假安全。由于已开发出了多种绕过它的机制，因此它并非完全有效。

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Cookie Security: HTTPOnly not Set on Session Cookie	2	0	0	2
Total	**2**	**0**	**0**	**2**

Cookie Security: HTTPOnly not Set on Session Cookie — High

Package: framework

framework/yiilite.php, line 4539 (Cookie Security: HTTPOnly not Set on Session Cookie) — High

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: session_set_cookie_params
Enclosing Method: setcookieparams()
File: framework/yiilite.php:4539
Taint Flags:

```
4536  if(isset($httponly))
4537  session_set_cookie_params($lifetime,$path,$domain,$secure,$httponly);
4538  else
4539  session_set_cookie_params($lifetime,$path,$domain,$secure);
4540  }
4541  public function getCookieMode()
4542  {
```

Package: framework.web

framework/web/CHttpSession.php, line 246 (Cookie Security: HTTPOnly not Set on Session Cookie) — High

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: session_set_cookie_params
Enclosing Method: setcookieparams()
File: framework/web/CHttpSession.php:246
Taint Flags:

```
243  if(isset($httponly))
244  session_set_cookie_params($lifetime,$path,$domain,$secure,$httponly);
245  else
246  session_set_cookie_params($lifetime,$path,$domain,$secure);
247  }
248
249  /**
```

Cookie Security: Persistent Session Cookie (4 issues)

Abstract

永久性会话 cookie 可导致危及帐户安全。

Explanation

永久性会话 cookie 在用户关闭了浏览器后仍然保持有效，并通常用作“记住我的信息”功能的一部分。因此，即使在用户关闭了浏览器后，永久性会话 cookie 也会使他们保持应用程序认证状态 — 假设他们没有明确注销。这就意味着如果第二个用户打开浏览器，他将以上个用户的身份自动登录。除非在受控环境中部署了应用程序，而在该环境中，不允许用户从共享计算机登录，否则即使用户关闭了浏览器，攻击者也可能危及用户帐户的安全。

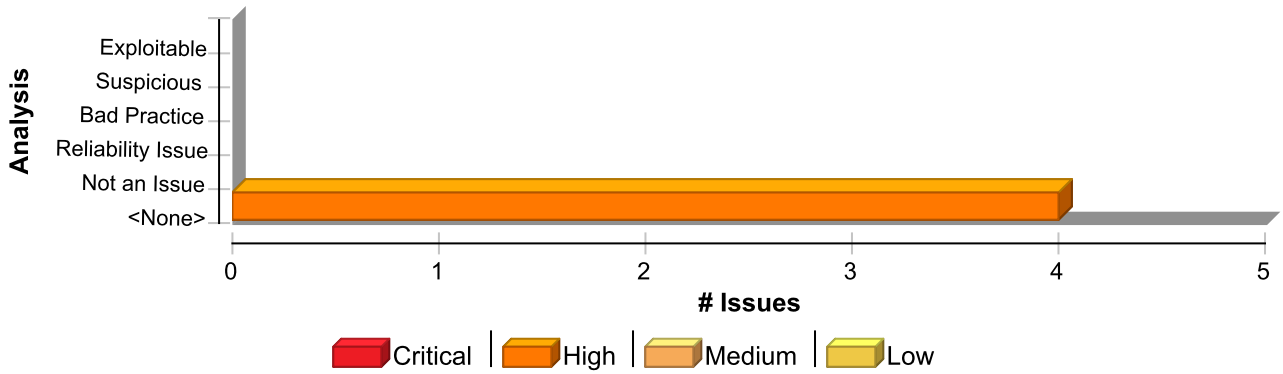
示例：以下代码将会话 cookie 设置为在 10 年后过期。

```
session_set_cookie_params(time()+60*60*24*365*10, "/", "www.example.com",
false, true);
```

Recommendation

不要使用永久性会话 cookie。在 PHP 配置文件中将 `session.cookie_lifetime` 设置为 `0`，便可完成此配置。

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Cookie Security: Persistent Session Cookie	4	0	0	4
Total	**4**	**0**	**0**	**4**

Cookie Security: Persistent Session Cookie — High

Package: framework

framework/yiilite.php, line 4539 (Cookie Security: Persistent Session Cookie) — High

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Semantic)

Cookie Security: Persistent Session Cookie — High

Package: framework

framework/yiilite.php, line 4539 (Cookie Security: Persistent Session Cookie) — High

Sink Details

Sink: session_set_cookie_params()
Enclosing Method: setcookieparams()
File: framework/yiilite.php:4539
Taint Flags:

```
4536  if(isset($httponly))
4537  session_set_cookie_params($lifetime,$path,$domain,$secure,$httponly);
4538  else
4539  session_set_cookie_params($lifetime,$path,$domain,$secure);
4540  }
4541  public function getCookieMode()
4542  {
```

framework/yiilite.php, line 4537 (Cookie Security: Persistent Session Cookie) — High

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Semantic)

Sink Details

Sink: session_set_cookie_params()
Enclosing Method: setcookieparams()
File: framework/yiilite.php:4537
Taint Flags:

```
4534  extract($data);
4535  extract($value);
4536  if(isset($httponly))
4537  session_set_cookie_params($lifetime,$path,$domain,$secure,$httponly);
4538  else
4539  session_set_cookie_params($lifetime,$path,$domain,$secure);
4540  }
```

Package: framework.web

framework/web/CHttpSession.php, line 246 (Cookie Security: Persistent Session Cookie) — High

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Semantic)

Sink Details

Sink: session_set_cookie_params()
Enclosing Method: setcookieparams()
File: framework/web/CHttpSession.php:246

Cookie Security: Persistent Session Cookie — High

Package: framework.web

framework/web/CHttpSession.php, line 246 (Cookie Security: Persistent Session Cookie) — High

Taint Flags:

```
243  if(isset($httponly))
244  session_set_cookie_params($lifetime,$path,$domain,$secure,$httponly);
245  else
246  session_set_cookie_params($lifetime,$path,$domain,$secure);
247  }
248
249  /**
```

framework/web/CHttpSession.php, line 244 (Cookie Security: Persistent Session Cookie) — High

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Semantic)

Sink Details

Sink: session_set_cookie_params()
Enclosing Method: setcookieparams()
File: framework/web/CHttpSession.php:244
Taint Flags:

```
241  extract($data);
242  extract($value);
243  if(isset($httponly))
244  session_set_cookie_params($lifetime,$path,$domain,$secure,$httponly);
245  else
246  session_set_cookie_params($lifetime,$path,$domain,$secure);
247  }
```

Cross-Site Scripting: Poor Validation (1 issue)

Abstract

依靠 HTML、XML 或其他类型编码验证用户输入可能会导致浏览器执行恶意代码。

Explanation

使用特定的编码函数（例如 `htmlspecialchars()` 或 `htmlentities()`）能避免一部分 cross-site scripting 攻击，但不能完全避免。根据数据出现的上下文，除 HTML 编码的基本字符 <、>、& 和 " 以及 XML 编码的字符 <、>、&、" 和 ' 之外（仅当已设置 `ENT_QUOTES` 时），其他字符可能具有元意。依靠此类编码函数等同于用一个安全性较差的黑名单来防止 cross-site scripting 攻击，并且可能允许攻击者注入恶意代码，并在浏览器中加以执行。由于不可能始终准确地确定静态显示数据的上下文，所以即便进行了编码，HPE Security Fortify 安全编码规则包仍会报告跨站脚本攻击结果，并将其显示为 Cross-Site Scripting: Poor Validation 问题。

Cross-Site Scripting (XSS) 漏洞在以下情况下发生：

1. 数据通过一个不可信赖的数据源进入 Web 应用程序。对于 Reflected XSS，不可信赖的源大多数情况下为 Web 请求；而对于 Persistent（也称为 Stored）XSS，该源通常为数据库查询的结果。

2. 未检验包含在动态内容中的数据，便将其传送给了 Web 用户。

传送到 Web 浏览器的恶意内容通常采用 JavaScript 代码片段的形式，但也可能会包含一些 HTML、Flash 或者其他任意一种可以被浏览器执行的代码。基于 XSS 的攻击手段花样百出，几乎是无穷无尽的，但通常它们都会包含传输给攻击者的私人数据（如 Cookie 或者其他会话信息）。在攻击者的控制下，指引受害者进入恶意的网络内容；或者利用易受攻击的站点，对用户的机器进行其他恶意操作。

示例 1：下面的代码片段会从 HTTP 请求中读取 `text` 参数，使用 HTML 加以编码，并将该参数显示在脚本标签之间的警报框中。

如果 `text` 只包含标准的字母或数字文本，这个例子中的代码就能正确运行。如果 `text` 具有单引号、圆括号和分号，则会结束 `alert` 文本框，之后将执行代码。

起初，这个例子似乎是不会轻易遭受攻击的。毕竟，有谁会输入导致恶意代码的 URL，并且还在自己的电脑上运行呢？真正的危险在于攻击者会创建恶意的 URL，然后采用电子邮件或者社会工程的欺骗手段诱使受害者访问此 URL 的链接。当受害者单击这个链接时，他们不知不觉地通过易受攻击的网络应用程序，将恶意内容带到了自己的电脑中。这种对易受攻击的 Web 应用程序进行盗取的机制通常被称为反射式 XSS。

正如例子中所显示的，XSS 漏洞是由于 HTTP 响应中包含了未经验证的数据代码而引起的。受害者遭受 XSS 攻击的途径有三种：

– 如例 1 所述，系统从 HTTP 请求中直接读取数据，并在 HTTP 响应中返回数据。当攻击者诱使用户为易受攻击的 Web 应用程序提供危险内容，而这些危险内容随后会反馈给用户并在 Web 浏览器中执行，就会发生反射式 XSS 盗取。发送恶意内容最常用的方法是，把恶意内容作为一个参数包含在公开发表的 URL 中，或者通过电子邮件直接发送给受害者。以这种手段构造的 URL 构成了多种“网络钓鱼”(phishing) 阴谋的核心，攻击者借此诱骗受害者访问指向易受攻击站点的 URL。站点将攻击者的内容反馈给受害者以后，便会执行这些内容，接下来会把用户计算机中的各种私密信息（比如包含会话信息的 cookie）传送给攻击者，或者执行其他恶意活动。

— 应用程序将危险数据存储在数据库或其他可信赖的数据存储器中。这些危险数据随后会被回写到应用程序中，并包含在动态内容中。Persistent XSS 盗取发生在如下情况：攻击者将危险内容注入到数据存储器中，且该存储器之后会被读取并包含在动态内容中。从攻击者的角度看，注入恶意内容的最佳位置莫过于一个面向许多用户，尤其是相关用户显示的区域。相关用户通常在应用程序中具备较高的特权，或相互之间交换敏感数据，这些数据对攻击者来说有利用价值。如果某一个用户执行了恶意内容，攻击者就有可能以该用户的名义执行某些需要特权的操作，或者获得该用户个人所有的敏感数据的访问权限。

— 应用程序之外的数据源将危险数据储存在一个数据库或其他数据存储器中，随后这些危险数据被当作可信赖的数据回写到应用程序中，并储存在动态内容中。

Recommendation

针对 XSS 的解决方法是，确保在适当位置进行验证，并检验其属性是否正确。

由于 XSS 漏洞出现在应用程序的输出中包含恶意数据时，因此，合乎逻辑的做法是在数据流出应用程序的前一刻对其进行验证。然而，由于 Web 应用程序常常会包含复杂而难以理解的代码，用以生成动态内容，因此，这一方法容易产生遗漏错误（遗漏验证）。降低这一风险的有效途径是对 XSS 也执行输入验证。

由于 Web 应用程序必须验证输入信息以避免其他漏洞（如 SQL Injection），因此，一种相对简单的解决方法是，加强一个应用程序现有的输入验证机制，将 XSS 检测包括其中。尽管有一定的价值，但 XSS 输入验证并不能取代严格的输出验证。应用程序可能通过共享的数据存储或其他可信赖的数据源接受输入，而该数据存储所接受的输入源可能并未执行适当的输入验证。因此，应用程序不能间接地依赖于该数据或其他任意数据的安全性。这就意味着，避免 XSS 漏洞的最佳方法是验证所有进入应用程序以及由应用程序传送至用户端的数据。

针对 XSS 漏洞进行验证最安全的方式是，创建一份安全字符白名单，允许其中的字符出现在 HTTP 内容中，并且只接受完全由这些经认可的字符组成的输入。例如，有效的用户名可能仅包含字母数字字符，电话号码可能仅包含 0-9 的数字。然而，这种解决方法在 Web 应用程序中通常是行不通的，因为许多字符对浏览器来说都具有特殊的含义，在写入代码时，这些字符仍应被视为合法的输入，比如一个 Web 设计版就必须接受带有 HTML 代码片段的输入。

更灵活的解决方法称为黑名单方法，但其安全性较差，这种方法在进行输入之前就有选择地拒绝或避免了潜在的危险字符。为了创建这样一个列表，首先需要了解对于 Web 浏览器具有特殊含义的字符集。虽然 HTML 标准定义了哪些字符具有特殊含义，但是许多 Web 浏览器会设法更正 HTML 中的常见错误，并可能在特定的上下文中认为其他字符具有特殊含义，这就是我们不鼓励使用黑名单作为阻止 XSS 的方法的原因。卡耐基梅隆大学 (Carnegie Mellon University) 软件工程学院 (Software Engineering Institute) 下属的 CERT(R) (CERT(R) Coordination Center) 合作中心提供了有关各种上下文中认定的特殊字符的具体信息 [1]：

在有关块级别元素的内容中（位于一段文本的中间）：

— "<" 是一个特殊字符，因为它可以引入一个标签。

— "&" 是一个特殊字符，因为它可以引入一个字符实体。

— ">" 是一个特殊字符，之所以某些浏览器将其认定为特殊字符，是基于一种假设，即该页的作者本想在前面添加一个 "<"，却错误地将其遗漏了。

下面的这些原则适用于属性值：

— 对于外加双引号的属性值，双引号是特殊字符，因为它们标记了该属性值的结束。

— 对于外加单引号的属性值，单引号是特殊字符，因为它们标记了该属性值的结束。

— 对于不带任何引号的属性值，空格字符（如空格符和制表符）是特殊字符。

- "&" 与某些特定变量一起使用时是特殊字符，因为它引入了一个字符实体。

例如，在 URL 中，搜索引擎可能会在结果页面内提供一个链接，用户可以点击该链接来重新运行搜索。可以将这一方法运用于编写 URL 中的搜索查询语句，这将引入更多特殊字符：

— 空格符、制表符和换行符是特殊字符，因为它们标记了 URL 的结束。

- "&" 是特殊字符，因为它可引入一个字符实体或分隔 CGI 参数。

— 非 ASCII 字符（即 ISO-8859-1 编码表中所有高于 128 的字符）不允许出现在 URL 中，因此在此上下文中也被视为特殊字符。

— 在服务器端对在 HTTP 转义序列中编码的参数进行解码时，必须过滤掉输入中的 "%" 符号。例如，当输入中出现 "%68%65%6C%6C%6F" 时，只有从输入的内容中过滤掉 "%"，上述字符串才能在网页上显示为 "hello"。

在 的正文内：

- 如果可以将文本直接插入到已有的脚本标签中，应该过滤掉分号、省略号、中括号和换行符。

服务器端脚本：

— 如果服务器端脚本会将输入中的感叹号 (!) 转换成输出中的双引号 (")，则可能需要对此进行更多过滤。

其他可能出现的情况：

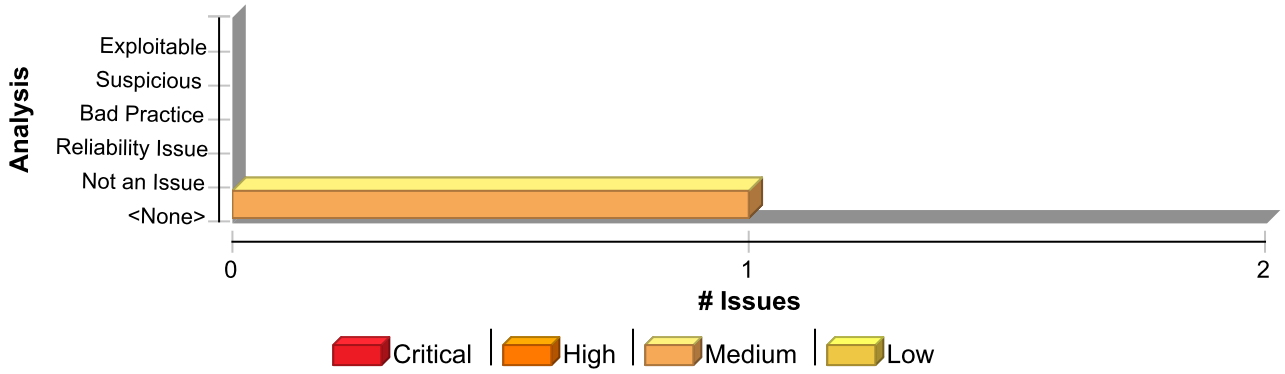
— 如果攻击者在 UTF-7 中提交了一个请求，那么特殊字符 "<" 可能会显示为 "+ADw-"，并可能会绕过过滤。如果输出包含在没有确切定义编码格式的网页中，有些浏览器就会设法根据内容自动识别编码（此处采用 UTF-7 格式）。

一旦在应用程序中确定了针对 XSS 攻击执行验证的正确要点，以及验证过程中要考虑的特殊字符，下一个难点就是定义验证过程中处理各种特殊字符的方式。如果应用程序认定某些特殊字符为无效输入，那么您可以拒绝任何带有这些无效特殊字符的输入。第二种选择就是采用过滤手段来删除这些特殊字符。然而，过滤的负面作用在于，过滤内容的显示将发生改变。在需要完整显示输入内容的情况下，过滤的这种负面作用可能是无法接受的。

如果必须接受带有特殊字符的输入，并将其准确地显示出来，验证机制一定要对所有特殊字符进行编码，以便删除其具有的含义。官方的 HTML 规范 [2] 提供了特殊字符对应的 ISO 8859-1 编码值的完整列表。

许多应用程序服务器都试图避免应用程序出现 Cross-Site Scripting 漏洞，具体做法是为负责设置特定 HTTP 响应内容的函数提供各种实现方式，以检验是否存在进行 Cross-Site Scripting 攻击必需的字符。不要依赖运行应用程序的服务器，以此确保该应用程序的安全。开发了某个应用程序后，并不能保证在其生命周期中它会在哪些应用程序服务器中运行。由于标准和已知盗取方式的演变，我们不能保证应用程序服务器也会保持同步。

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Cross-Site Scripting: Poor Validation	1	0	0	1
Total	**1**	**0**	**0**	**1**

Cross-Site Scripting: Poor Validation — Medium

Package: web.protected.vendor.phpqrcode

web/protected/vendor/phpqrcode/index.php, line 81 (Cross-Site Scripting: Poor Validation) — Medium

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_REQUEST['data']
File: web/protected/vendor/phpqrcode/index.php:76

```
73
74  //config form
75  echo '<form action="index.php" method="post">
76  Data: <input name="data" value="'.(isset($_REQUEST['data'])?
htmlspecialchars($_REQUEST['data']):'PHP QR Code :)').'" /> 
77  ECC: <select name="level">
78  <option value="L"'.(($errorCorrectionLevel=='L')?' selected':'').'>L -
smallest</option>
79  <option value="M"'.(($errorCorrectionLevel=='M')?' selected':'').'>M</
option>
```

Sink Details

Sink: builtin_echo()
File: web/protected/vendor/phpqrcode/index.php:81
Taint Flags: HTML_ENCODE, POORVALIDATION, VALIDATED_CROSS_SITE_SCRIPTING_DOM, VALIDATED_CROSS_SITE_SCRIPTING_PERSISTENT, VALIDATED_CROSS_SITE_SCRIPTING_REFLECTED, WEB, XSS

```
78  <option value="L"'.(($errorCorrectionLevel=='L')?' selected':'').'>L - smallest</option>
```

Cross-Site Scripting: Poor Validation — Medium

Package: web.protected.vendor.phpqrcode

web/protected/vendor/phpqrcode/index.php, line 81 (Cross-Site Scripting: Poor Validation) — Medium

```
79  <option value="M"'.(($errorCorrectionLevel=='M')?' selected':'').'>M</option>
80  <option value="Q"'.(($errorCorrectionLevel=='Q')?' selected':'').'>Q</option>
81  <option value="H"'.(($errorCorrectionLevel=='H')?' selected':'').'>H - best</option>
82  </select> 
83  Size: <select name="size">';
84
```

Cross-Site Scripting: Reflected (72 issues)

Abstract

向一个 Web 浏览器发送未经验证的数据会导致该浏览器执行恶意代码。

Explanation

Cross-Site Scripting (XSS) 漏洞在以下情况下发生：

1. 数据通过一个不可信赖的数据源进入 Web 应用程序。对于 Reflected XSS，不可信赖的源通常为 Web 请求，而对于 Persisted（也称为 Stored）XSS，该源通常为数据库或其他后端数据存储。

2. 未检验包含在动态内容中的数据，便将其传送给了 Web 用户。

传送到 Web 浏览器的恶意内容通常采用 JavaScript 代码片段的形式，但也可能会包含一些 HTML、Flash 或者其他任意一种可以被浏览器执行的代码。基于 XSS 的攻击手段花样百出，几乎是无穷无尽的，但通常它们都会包含传输给攻击者的私人数据（如 Cookie 或者其他会话信息）。在攻击者的控制下，指引受害者进入恶意的网络内容；或者利用易受攻击的站点，对用户的机器进行其他恶意操作。

例 1：下面的 PHP 代码片段可从一个 HTTP 请求中读取雇员 ID `eid`，并将其显示给用户。

```
...
```

如果 `eid` 只包含标准的字母或数字文本，这个例子中的代码就能正确运行。如果 `eid` 里有包含元字符或源代码中的值，那么 Web 浏览器就会像显示 HTTP 响应那样执行代码。

起初，这个例子似乎是不会轻易遭受攻击的。毕竟，有谁会输入导致恶意代码的 URL，并且还在自己的电脑上运行呢？真正的危险在于攻击者会创建恶意的 URL，然后采用电子邮件或者社会工程的欺骗手段诱使受害者访问此 URL 的链接。当受害者单击这个链接时，他们不知不觉地通过易受攻击的网络应用程序，将恶意内容带到了自己的电脑中。这种对易受攻击的 Web 应用程序进行盗取的机制通常被称为反射式 XSS。

例 2：下面的 PHP 代码片段可根据一个给定的雇员 ID 查询数据库，并显式出相应的雇员姓名。

如同例 1，如果对 `name` 的值处理得当，该代码就能正常地执行各种功能；如若处理不当，就会对代码的盗取行为无能为力。同样，这段代码暴露出的危险较小，因为 `name` 的值是从数据库中读取的，而且显然这些内容是由应用程序管理的。然而，如果 `name` 的值是由用户提供的数据产生，数据库就会成为恶意内容沟通的通道。如果不对数据库中存储的所有数据进行恰当的输入验证，那么攻击者便能在用户的 Web 浏览器中执行恶意命令。这种类型的 Persistent XSS（也称为 Stored XSS）盗取极其阴险狡猾，因为数据存储导致的间接性使得辨别威胁的难度增大，而且还提高了一个攻击影响多个用户的可能性。XSS 盗取会从访问提供留言簿 (guestbook) 的网站开始。攻击者会在这些留言簿的条目中嵌入 JavaScript，接下来所有访问该留言簿的用户都会执行这些恶意代码。

正如例子中所显示的，XSS 漏洞是由于 HTTP 响应中包含了未经验证的数据代码而引起的。受害者遭受 XSS 攻击的途径有三种：

– 如例 1 所述，系统从 HTTP 请求中直接读取数据，并在 HTTP 响应中返回数据。当攻击者诱使用户为易受攻击的 Web 应用程序提供危险内容，而这些危险内容随后会反馈给用户并在 Web 浏览器中执行，就会发生反射式 XSS 盗取。发送恶意内容最常用的方法是，把恶意内容作为一个参数包含在公开发表的 URL 中，或者通过电子邮件直接发送给受害者。以这种手段构造的 URL 构成了多种“网络钓鱼”(phishing) 阴谋的核心，攻击者借此诱骗受害者访问指向易受攻击站点的 URL。站点将攻击者的内容反馈给受害者以后，便会执行这些内容，接下来会把用户计算机中的各种私密信息（比如包含会话信息的 cookie）传送给攻击者，或者执行其他恶意活动。

– 如例 2 所述，应用程序将危险数据储存在一个数据库或其他可信赖的数据存储器中。这些危险数据随后会被回写到应用程序中，并包含在动态内容中。Persistent XSS 盗取发生在如下情况：攻击者将危险内容注入到数据存储器中，且该存储器之后会被读取并包含在动态内容中。从攻击者的角度看，注入恶意内容的最佳位置莫过于一个面向许多用户，尤其是相关用户显示的区域。相关用户通常在应用程序中具备较高的特权，或相互之间交换敏感数据，这些数据对攻击者来说有利用价值。如果某一个用户执行了恶意内容，攻击者就有可能以该用户的名义执行某些需要特权的操作，或者获得该用户个人所有的敏感数据的访问权限。

— 应用程序之外的数据源将危险数据储存在一个数据库或其他数据存储器中，随后这些危险数据被当作可信赖的数据回写到应用程序中，并储存在动态内容中。

Recommendation

针对 XSS 的解决方法是，确保在适当位置进行验证，并检验其属性是否正确。

由于 XSS 漏洞出现在应用程序的输出中包含恶意数据时，因此，合乎逻辑的做法是在数据流出应用程序的前一刻对其进行验证。然而，由于 Web 应用程序常常会包含复杂而难以理解的代码，用以生成动态内容，因此，这一方法容易产生遗漏错误（遗漏验证）。降低这一风险的有效途径是对 XSS 也执行输入验证。

由于 Web 应用程序必须验证输入信息以避免其他漏洞（如 SQL Injection），因此，一种相对简单的解决方法是，加强一个应用程序现有的输入验证机制，将 XSS 检测包括其中。尽管有一定的价值，但 XSS 输入验证并不能取代严格的输出验证。应用程序可能通过共享的数据存储或其他可信赖的数据源接受输入，而该数据存储所接受的输入源可能并未执行适当的输入验证。因此，应用程序不能间接地依赖于该数据或其他任意数据的安全性。这就意味着，避免 XSS 漏洞的最佳方法是验证所有进入应用程序以及由应用程序传送至用户端的数据。

针对 XSS 漏洞进行验证最安全的方式是，创建一份安全字符白名单，允许其中的字符出现在 HTTP 内容中，并且只接受完全由这些经认可的字符组成的输入。例如，有效的用户名可能仅包含字母数字字符，电话号码可能仅包含 0-9 的数字。然而，这种解决方法在 Web 应用程序中通常是行不通的，因为许多字符对浏览器来说都具有特殊的含义，在写入代码时，这些字符仍应被视为合法的输入，比如一个 Web 设计版就必须接受带有 HTML 代码片段的输入。

更灵活的解决方法称为黑名单方法，但其安全性较差，这种方法在进行输入之前就有选择地拒绝或避免了潜在的危险字符。为了创建这样一个列表，首先需要了解对于 Web 浏览器具有特殊含义的字符集。虽然 HTML 标准定义了哪些字符具有特殊含义，但是许多 Web 浏览器会设法更正 HTML 中的常见错误，并可能在特定的上下文中认为其他字符具有特殊含义，这就是我们不鼓励使用黑名单作为阻止 XSS 的方法的原因。卡耐基梅隆大学 (Carnegie Mellon University) 软件工程学院 (Software Engineering Institute) 下属的 CERT(R) (CERT(R) Coordination Center) 合作中心提供了有关各种上下文中认定的特殊字符的具体信息 [1]：

在有关块级别元素的内容中（位于一段文本的中间）：

– "<" 是一个特殊字符，因为它可以引入一个标签。

– "&" 是一个特殊字符，因为它可以引入一个字符实体。

– ">" 是一个特殊字符，之所以某些浏览器将其认定为特殊字符，是基于一种假设，即该页的作者本想在前面添加一个 "<"，却错误地将其遗漏了。

下面的这些原则适用于属性值：

— 对于外加双引号的属性值，双引号是特殊字符，因为它们标记了该属性值的结束。

— 对于外加单引号的属性值，单引号是特殊字符，因为它们标记了该属性值的结束。

— 对于不带任何引号的属性值，空格字符（如空格符和制表符）是特殊字符。

- "&" 与某些特定变量一起使用时是特殊字符，因为它引入了一个字符实体。

例如，在 URL 中，搜索引擎可能会在结果页面内提供一个链接，用户可以点击该链接来重新运行搜索。可以将这一方法运用于编写 URL 中的搜索查询语句，这将引入更多特殊字符：

— 空格符、制表符和换行符是特殊字符，因为它们标记了 URL 的结束。

- "&" 是特殊字符，因为它可引入一个字符实体或分隔 CGI 参数。

— 非 ASCII 字符（即 ISO-8859-1 编码表中所有高于 128 的字符）不允许出现在 URL 中，因此在此上下文中也被视为特殊字符。

— 在服务器端对在 HTTP 转义序列中编码的参数进行解码时，必须过滤掉输入中的 "%" 符号。例如，当输入中出现 "%68%65%6C%6C%6F" 时，只有从输入的内容中过滤掉 "%"，上述字符串才能在网页上显示为 "hello"。

在 的正文内：

- 如果可以将文本直接插入到已有的脚本标签中，应该过滤掉分号、省略号、中括号和换行符。

服务器端脚本：

— 如果服务器端脚本会将输入中的感叹号 (!) 转换成输出中的双引号 (")，则可能需要对此进行更多过滤。

其他可能出现的情况：

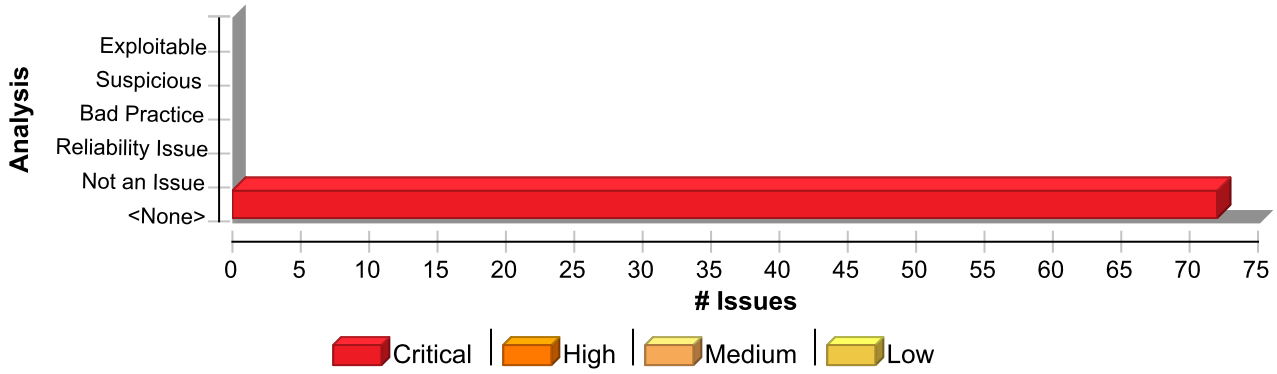
— 如果攻击者在 UTF-7 中提交了一个请求，那么特殊字符 "<" 可能会显示为 "+ADw-"，并可能会绕过过滤。如果输出包含在没有确切定义编码格式的网页中，有些浏览器就会设法根据内容自动识别编码（此处采用 UTF-7 格式）。

一旦在应用程序中确定了针对 XSS 攻击执行验证的正确要点，以及验证过程中要考虑的特殊字符，下一个难点就是定义验证过程中处理各种特殊字符的方式。如果应用程序认定某些特殊字符为无效输入，那么您可以拒绝任何带有这些无效特殊字符的输入。第二种选择就是采用过滤手段来删除这些特殊字符。然而，过滤的负面作用在于，过滤内容的显示将发生改变。在需要完整显示输入内容的情况下，过滤的这种负面作用可能是无法接受的。

如果必须接受带有特殊字符的输入，并将其准确地显示出来，验证机制一定要对所有特殊字符进行编码，以便删除其具有的含义。官方的 HTML 规范 [2] 提供了特殊字符对应的 ISO 8859-1 编码值的完整列表。

许多应用程序服务器都试图避免应用程序出现 Cross-Site Scripting 漏洞，具体做法是为负责设置特定 HTTP 响应内容的函数提供各种实现方式，以检验是否存在进行 Cross-Site Scripting 攻击必需的字符。不要依赖运行应用程序的服务器，以此确保该应用程序的安全。开发了某个应用程序后，并不能保证在其生命周期中它会在哪些应用程序服务器中运行。由于标准和已知盗取方式的演变，我们不能保证应用程序服务器也会保持同步。

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Cross-Site Scripting: Reflected	72	0	0	72
Total	**72**	**0**	**0**	**72**

Cross-Site Scripting: Reflected — Critical

Package: web.protected.vendor.message

web/protected/vendor/message/WMessage.php, line 24 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_POST['materialIDs']
From: inventorycontroller.actionadd
File: web/protected/controllers/InventoryController.php:41

```
38  WMessage::ajaxInfo("      ", 0);
39  }
40  } else {
41  $materialIDs = $_POST['materialIDs'];
42  if ($materialIDs == "") {
43  WMessage::ajaxInfo("         ", 0);
44  }
```

Sink Details

Sink: builtin_echo()
Enclosing Method: ajaxinfo()
File: web/protected/vendor/message/WMessage.php:24
Taint Flags: WEB, XSS

```
21  */
22  public static function ajaxInfo($info="      ",$status=1,array $data=array()){
23  header("content-type:application/json");
24  echo json_encode(array(
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.vendor.message

web/protected/vendor/message/WMessage.php, line 24 (Cross-Site Scripting: Reflected) — Critical

```
25  "info"=>$info,
26  "status"=>$status,
27  "data"=>$data
```

Package: web.protected.views.UseMaterial

web/protected/views/UseMaterial/select_list2.php, line 22 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['goodsName']
File: web/protected/views/UseMaterial/select_list2.php:22

```
19      ¢ →
20  <input class="grid_text" name="goodsCode" value="<?php echo
$_GET['goodsCode'];?>" />
21      |
22  <input class="grid_text" name="goodsName" value="<?php echo
$_GET['goodsName'];?>" />
23   ○
24  <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro'];?
>" />
25  <input type="submit" value=" ← " class="grid_button grid_button_s">
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/UseMaterial/select_list2.php:22
Taint Flags: WEB, XSS

```
19      ¢ →
20  <input class="grid_text" name="goodsCode" value="<?php echo $_GET['goodsCode'];?>" />
21      |
22  <input class="grid_text" name="goodsName" value="<?php echo $_GET['goodsName'];?>" />
23   ○
24  <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro'];?>" />
25  <input type="submit" value=" ← " class="grid_button grid_button_s">
```

web/protected/views/UseMaterial/return_form_list.php, line 55 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.UseMaterial

web/protected/views/UseMaterial/return_form_list.php, line 55 (Cross-Site Scripting: Reflected) — Critical

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['glPro']
File: web/protected/views/UseMaterial/return_form_list.php:55

```
52
53  <input class="grid_text" name="formCode" value="<?php echo
$_GET['formCode']; ?>" />
54   ○
55  <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro']; ?
>" />
56  <input type="submit" value=" ← " class="grid_button grid_button_s" />
57  </td></tr>
58  </table>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/UseMaterial/return_form_list.php:55
Taint Flags: WEB, XSS

```
52
53  <input class="grid_text" name="formCode" value="<?php echo $_GET['formCode']; ?>" />
54   ○
55  <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro']; ?>" />
56  <input type="submit" value=" ← " class="grid_button grid_button_s" />
57  </td></tr>
58  </table>
```

web/protected/views/UseMaterial/receive_form_list.php, line 52 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['formCode']
File: web/protected/views/UseMaterial/receive_form_list.php:52

```
49  <option value="qx" <?php if($_GET['nature']=="qx"){echo "selected";}?>>
</option>
50  </select>
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.UseMaterial

web/protected/views/UseMaterial/receive_form_list.php, line 52 (Cross-Site Scripting: Reflected) — Critical

```
51   ↓
52   <input class="grid_text" name="formCode" value="<?php echo
$_GET['formCode']; ?>" />
53   ○
54   <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro']; ?
>" />
55   <input type="hidden" name="type" value="<?php echo $_GET['type']; ?>" />
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/UseMaterial/receive_form_list.php:52
Taint Flags: WEB, XSS

```
49  <option value="qx" <?php if($_GET['nature']=="qx"){echo "selected";}?>>   </option>
50  </select>
51   ↓
52  <input class="grid_text" name="formCode" value="<?php echo $_GET['formCode']; ?>" />
53   ○
54  <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro']; ?>" />
55  <input type="hidden" name="type" value="<?php echo $_GET['type']; ?>" />
```

web/protected/views/UseMaterial/select_list2.php, line 24 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['glPro']
File: web/protected/views/UseMaterial/select_list2.php:24

```
21      |
22   <input class="grid_text" name="goodsName" value="<?php echo
$_GET['goodsName'];?>" />
23   ○
24   <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro'];?
>" />
25   <input type="submit" value=" ← " class="grid_button grid_button_s">
26 <!-- <div style="text-align: left">-->
27 <!--      ¢ →  -->
```

Sink Details

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.UseMaterial

web/protected/views/UseMaterial/select_list2.php, line 24 (Cross-Site Scripting: Reflected) — Critical

Sink: builtin_echo()
File: web/protected/views/UseMaterial/select_list2.php:24
Taint Flags: WEB, XSS

```
21      |
22   <input class="grid_text" name="goodsName" value="<?php echo $_GET['goodsName'];?>" />
23    ○
24   <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro'];?>" />
25   <input type="submit" value=" ← " class="grid_button grid_button_s">
26 <!-- <div style="text-align: left">-->
27 <!--      ¢ → -->
```

web/protected/views/UseMaterial/return_form_list.php, line 45 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['type']
File: web/protected/views/UseMaterial/return_form_list.php:45

```
42   <form method="get" action="<?= Yii::app()->createUrl("UseMaterial/
ReturnMFList") ?>">
43   <table>
44   <tr><td>
45   <input type="hidden" name="type" value="<?php echo $_GET['type'];?>">
46
47   <select class="grid_text" name="nature" style="width:100px;height: 24px;">
48   <option></option>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/UseMaterial/return_form_list.php:45
Taint Flags: WEB, XSS

```
42   <form method="get" action="<?= Yii::app()->createUrl("UseMaterial/ReturnMFList") ?>">
43   <table>
44   <tr><td>
45   <input type="hidden" name="type" value="<?php echo $_GET['type'];?>">
46
47   <select class="grid_text" name="nature" style="width:100px;height: 24px;">
48   <option></option>
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.UseMaterial

web/protected/views/UseMaterial/receive_form_list.php, line 55 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['type']
File: web/protected/views/UseMaterial/receive_form_list.php:55

```
52  <input class="grid_text" name="formCode" value="<?php echo
$_GET['formCode']; ?>" />
53   ○
54  <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro']; ?
>" />
55  <input type="hidden" name="type" value="<?php echo $_GET['type']; ?>" />
56  <input type="submit" value=" ← " class="grid_button grid_button_s" />
57  </td></tr>
58  </table>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/UseMaterial/receive_form_list.php:55
Taint Flags: WEB, XSS

```
52  <input class="grid_text" name="formCode" value="<?php echo $_GET['formCode']; ?>" />
53   ○
54  <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro']; ?>" />
55  <input type="hidden" name="type" value="<?php echo $_GET['type']; ?>" />
56  <input type="submit" value=" ← " class="grid_button grid_button_s" />
57  </td></tr>
58  </table>
```

web/protected/views/UseMaterial/use_material_list.php, line 35 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['goodsName']
File: web/protected/views/UseMaterial/use_material_list.php:35

```
32        ¢
33  <input class="grid_text" name="goodsCode" value="<?php echo
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.UseMaterial

web/protected/views/UseMaterial/use_material_list.php, line 35 (Cross-Site Scripting: Reflected) — Critical

```
$_GET['goodsCode']; ?>" />
34
35  <input class="grid_text" name="goodsName" value="<?php echo
$_GET['goodsName']; ?>" />
36
37  <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro']; ?
>" style="width:100px;" />
38  <?=$typeName;?>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/UseMaterial/use_material_list.php:35
Taint Flags: WEB, XSS

```
32
33  <input class="grid_text" name="goodsCode" value="<?php echo $_GET['goodsCode']; ?>" />
34
35  <input class="grid_text" name="goodsName" value="<?php echo $_GET['goodsName']; ?>" />
36
37  <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro']; ?>" style="width:
100px;" />
38  <?=$typeName;?>
```

web/protected/views/UseMaterial/select_list2.php, line 20 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['goodsCode']
File: web/protected/views/UseMaterial/select_list2.php:20

```
17
18  <input class="grid_text" name="batchCode" value="<?php echo
$_GET['batchCode'];?>" />
19
20  <input class="grid_text" name="goodsCode" value="<?php echo
$_GET['goodsCode'];?>" />
21
22  <input class="grid_text" name="goodsName" value="<?php echo
$_GET['goodsName'];?>" />
23
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.UseMaterial

web/protected/views/UseMaterial/select_list2.php, line 20 (Cross-Site Scripting: Reflected) — Critical

Sink Details

Sink: builtin_echo()
File: web/protected/views/UseMaterial/select_list2.php:20
Taint Flags: WEB, XSS

```
17
18  <input class="grid_text" name="batchCode" value="<?php echo $_GET['batchCode'];?>" />
19      ¢ →
20  <input class="grid_text" name="goodsCode" value="<?php echo $_GET['goodsCode'];?>" />
21      |
22  <input class="grid_text" name="goodsName" value="<?php echo $_GET['goodsName'];?>" />
23   ○
```

web/protected/views/UseMaterial/select_list.php, line 18 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['batchCode']
File: web/protected/views/UseMaterial/select_list.php:18

```
15  <td align="right"style="width: 520px;">
16  <form method="get" action="<?=Yii::app()->createUrl("UseMaterial/
SelectList")?>">
17
18  <input class="grid_text" name="batchCode" value="<?php echo
$_GET['batchCode'];?>" />
19      ¢ →
20  <input class="grid_text" name="goodsCode" value="<?php echo
$_GET['goodsCode'];?>" />
21      |
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/UseMaterial/select_list.php:18
Taint Flags: WEB, XSS

```
15  <td align="right"style="width: 520px;">
16  <form method="get" action="<?=Yii::app()->createUrl("UseMaterial/SelectList")?>">
17
18  <input class="grid_text" name="batchCode" value="<?php echo $_GET['batchCode'];?>" />
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.UseMaterial

web/protected/views/UseMaterial/select_list.php, line 18 (Cross-Site Scripting: Reflected) — Critical

```
19      ¢ →
20  <input class="grid_text" name="goodsCode" value="<?php echo $_GET['goodsCode'];?>" />
21      |
```

web/protected/views/UseMaterial/receive_form_list.php, line 54 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['glPro']
File: web/protected/views/UseMaterial/receive_form_list.php:54

```
51   ↓
52   <input class="grid_text" name="formCode" value="<?php echo
$_GET['formCode']; ?>" />
53   ○
54   <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro']; ?
>" />
55   <input type="hidden" name="type" value="<?php echo $_GET['type']; ?>" />
56   <input type="submit" value=" ← " class="grid_button grid_button_s" />
57   </td></tr>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/UseMaterial/receive_form_list.php:54
Taint Flags: WEB, XSS

```
51   ↓
52   <input class="grid_text" name="formCode" value="<?php echo $_GET['formCode']; ?>" />
53   ○
54   <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro']; ?>" />
55   <input type="hidden" name="type" value="<?php echo $_GET['type']; ?>" />
56   <input type="submit" value=" ← " class="grid_button grid_button_s" />
57   </td></tr>
```

web/protected/views/UseMaterial/select_list.php, line 22 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.UseMaterial

web/protected/views/UseMaterial/select_list.php, line 22 (Cross-Site Scripting: Reflected) — Critical

Source Details

Source: Read $_GET['goodsName']
File: web/protected/views/UseMaterial/select_list.php:22

```
19      ¢ →
20  <input class="grid_text" name="goodsCode" value="<?php echo
$_GET['goodsCode'];?>" />
21      |
22  <input class="grid_text" name="goodsName" value="<?php echo
$_GET['goodsName'];?>" />
23  <input type="submit" value=" ← " class="grid_button grid_button_s">
24 <!-- <div style="text-align: left">-->
25 <!--     ¢ →  -->
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/UseMaterial/select_list.php:22
Taint Flags: WEB, XSS

```
19      ¢ →
20  <input class="grid_text" name="goodsCode" value="<?php echo $_GET['goodsCode'];?>" />
21      |
22  <input class="grid_text" name="goodsName" value="<?php echo $_GET['goodsName'];?>" />
23  <input type="submit" value=" ← " class="grid_button grid_button_s">
24 <!-- <div style="text-align: left">-->
25 <!--     ¢ →  -->
```

web/protected/views/UseMaterial/select_list2.php, line 18 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['batchCode']
File: web/protected/views/UseMaterial/select_list2.php:18

```
15  <td align="right"style="width: 520px;">
16  <form method="get" action="<?=Yii::app()->createUrl("UseMaterial/
SelectList2")?>">
17
18  <input class="grid_text" name="batchCode" value="<?php echo
$_GET['batchCode'];?>" />
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.UseMaterial

web/protected/views/UseMaterial/select_list2.php, line 18 (Cross-Site Scripting: Reflected) — Critical

```
19      ¢ →
20   <input class="grid_text" name="goodsCode" value="<?php echo
$_GET['goodsCode'];?>" />
21      |
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/UseMaterial/select_list2.php:18
Taint Flags: WEB, XSS

```
15  <td align="right"style="width: 520px;">
16  <form method="get" action="<?=Yii::app()->createUrl("UseMaterial/SelectList2")?>">
17
18  <input class="grid_text" name="batchCode" value="<?php echo $_GET['batchCode'];?>" />
19      ¢ →
20  <input class="grid_text" name="goodsCode" value="<?php echo $_GET['goodsCode'];?>" />
21      |
```

web/protected/views/UseMaterial/edit_photo.php, line 2 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['id']
File: web/protected/views/UseMaterial/edit_photo.php:2

```
1 <form action="<?=Yii::app()->request->url?>" id="form" name="form"
enctype="multipart/form-data" style="padding: 10px;" method="post">
2   <input name="id" type="hidden" value="<?=$_GET['id']?>">
3   <table border="0" cellspacing="0" cellpadding="0" class="github_tb"
width="100%">
4   <tr>
5   <td>     </td>
6   <td>
7   <input type="file" name="pic"></td>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/UseMaterial/edit_photo.php:2
Taint Flags: WEB, XSS

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.UseMaterial

web/protected/views/UseMaterial/edit_photo.php, line 2 (Cross-Site Scripting: Reflected) — Critical

```
1 <form action="<?=Yii::app()->request->url?>" id="form" name="form" enctype="multipart/form-
data" style="padding: 10px;" method="post">
2  <input name="id" type="hidden" value="<?=$_GET['id']?>">
3  <table border="0" cellspacing="0" cellpadding="0" class="github_tb" width="100%">
4  <tr>
5  <td>    </td>
6  <td>
7  <input type="file" name="pic"></td>
```

web/protected/views/UseMaterial/use_material_list.php, line 28 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['type']
File: web/protected/views/UseMaterial/use_material_list.php:28

```
25  </td>
26  <td align="right">
27  <form method="get" action="<?= Yii::app()->createUrl("UseMaterial/
UseMaterialList") ?>">
28  <input type="hidden" name="type" value="<?php echo $_GET['type']; ?>" />
29  <?php $typeName = $_GET['type']=="receive"?" ↓ ":"   ";?>
30  <table>
31  <tr><td>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/UseMaterial/use_material_list.php:28
Taint Flags: WEB, XSS

```
25  </td>
26  <td align="right">
27  <form method="get" action="<?= Yii::app()->createUrl("UseMaterial/UseMaterialList") ?>">
28  <input type="hidden" name="type" value="<?php echo $_GET['type']; ?>" />
29  <?php $typeName = $_GET['type']=="receive"?" ↓ ":"   ";?>
30  <table>
31  <tr><td>
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.UseMaterial

web/protected/views/UseMaterial/use_material_list.php, line 61 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['type']
File: web/protected/views/UseMaterial/use_material_list.php:61

```
58  }
59  ?>
60  </select>
61  <input type="hidden" name="type" value="<?php echo $_GET['type']; ?>" />
62  <input type="submit" value=" ← " class="grid_button grid_button_s" />
63  </td></tr>
64  </table>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/UseMaterial/use_material_list.php:61
Taint Flags: WEB, XSS

```
58  }
59  ?>
60  </select>
61  <input type="hidden" name="type" value="<?php echo $_GET['type']; ?>" />
62  <input type="submit" value=" ← " class="grid_button grid_button_s" />
63  </td></tr>
64  </table>
```

web/protected/views/UseMaterial/scrap_form_list.php, line 31 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['formCode']
File: web/protected/views/UseMaterial/scrap_form_list.php:31

```
28  <table>
29  <tr><td>
30
31  <input class="grid_text" name="formCode" value="<?php echo
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.UseMaterial

web/protected/views/UseMaterial/scrap_form_list.php, line 31 (Cross-Site Scripting: Reflected) — Critical

```
$_GET['formCode']; ?>" />
32    ○
33   <input class="grid_text" name="projectName" value="<?php echo
$_GET['projectName']; ?>" />
34   <input type="submit" value=" ← " class="grid_button grid_button_s" />
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/UseMaterial/scrap_form_list.php:31
Taint Flags: WEB, XSS

```
28  <table>
29  <tr><td>
30
31  <input class="grid_text" name="formCode" value="<?php echo $_GET['formCode']; ?>" />
32   ○
33  <input class="grid_text" name="projectName" value="<?php echo $_GET['projectName']; ?>" />
34  <input type="submit" value=" ← " class="grid_button grid_button_s" />
```

web/protected/views/UseMaterial/use_material_list.php, line 50 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['formCode']
File: web/protected/views/UseMaterial/use_material_list.php:50

```
47   <?=$typeName;?>
48   <input class="grid_text" name="batchCode" value="<?php echo
$_GET['batchCode']; ?>" />
49   <?=$typeName;?>
50   <input class="grid_text" name="formCode" value="<?php echo
$_GET['formCode']; ?>" />
51    ¬ ₩
52   <select class="grid_text" name="storeID" id="storeID" style="width:
130px;height: 24px">
53   <option></option>
```

Sink Details

Sink: builtin_echo()

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.UseMaterial

web/protected/views/UseMaterial/use_material_list.php, line 50 (Cross-Site Scripting: Reflected) — Critical

File: web/protected/views/UseMaterial/use_material_list.php:50
Taint Flags: WEB, XSS

```
47  <?=$typeName;?>
48  <input class="grid_text" name="batchCode" value="<?php echo $_GET['batchCode']; ?>" />
49  <?=$typeName;?>
50  <input class="grid_text" name="formCode" value="<?php echo $_GET['formCode']; ?>" />
51   ¬ ₩
52  <select class="grid_text" name="storeID" id="storeID" style="width:130px;height: 24px">
53  <option></option>
```

web/protected/views/UseMaterial/scrap_form_list.php, line 33 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['projectName']
File: web/protected/views/UseMaterial/scrap_form_list.php:33

```
30
31  <input class="grid_text" name="formCode" value="<?php echo
$_GET['formCode']; ?>" />
32   ○
33  <input class="grid_text" name="projectName" value="<?php echo
$_GET['projectName']; ?>" />
34  <input type="submit" value="←  " class="grid_button grid_button_s" />
35  </td></tr>
36  </table>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/UseMaterial/scrap_form_list.php:33
Taint Flags: WEB, XSS

```
30
31  <input class="grid_text" name="formCode" value="<?php echo $_GET['formCode']; ?>" />
32   ○
33  <input class="grid_text" name="projectName" value="<?php echo $_GET['projectName']; ?>" />
34  <input type="submit" value="←  " class="grid_button grid_button_s" />
35  </td></tr>
36  </table>
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.UseMaterial

web/protected/views/UseMaterial/use_material_list.php, line 48 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['batchCode']
File: web/protected/views/UseMaterial/use_material_list.php:48

```
45  </td></tr>
46  <tr><td>
47  <?=$typeName;?>
48  <input class="grid_text" name="batchCode" value="<?php echo
$_GET['batchCode']; ?>" />
49  <?=$typeName;?>
50  <input class="grid_text" name="formCode" value="<?php echo
$_GET['formCode']; ?>" />
51   ¬ ₩
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/UseMaterial/use_material_list.php:48
Taint Flags: WEB, XSS

```
45  </td></tr>
46  <tr><td>
47  <?=$typeName;?>
48  <input class="grid_text" name="batchCode" value="<?php echo $_GET['batchCode']; ?>" />
49  <?=$typeName;?>
50  <input class="grid_text" name="formCode" value="<?php echo $_GET['formCode']; ?>" />
51   ¬ ₩
```

web/protected/views/UseMaterial/return_form_list.php, line 53 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['formCode']
File: web/protected/views/UseMaterial/return_form_list.php:53

```
50  <option value="qx" <?php if($_GET['nature']=="qx"){echo "selected";}?>>
</option>
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.UseMaterial

web/protected/views/UseMaterial/return_form_list.php, line 53 (Cross-Site Scripting: Reflected) — Critical

```
51  </select>
52
53  <input class="grid_text" name="formCode" value="<?php echo
$_GET['formCode']; ?>" />
54   ○
55  <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro']; ?
>" />
56  <input type="submit" value=" ← " class="grid_button grid_button_s" />
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/UseMaterial/return_form_list.php:53
Taint Flags: WEB, XSS

```
50  <option value="qx" <?php if($_GET['nature']=="qx"){echo "selected";}?>>   </option>
51  </select>
52
53  <input class="grid_text" name="formCode" value="<?php echo $_GET['formCode']; ?>" />
54   ○
55  <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro']; ?>" />
56  <input type="submit" value=" ← " class="grid_button grid_button_s" />
```

web/protected/views/UseMaterial/use_material_list.php, line 33 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['goodsCode']
File: web/protected/views/UseMaterial/use_material_list.php:33

```
30  <table>
31  <tr><td>
32      ¢
33  <input class="grid_text" name="goodsCode" value="<?php echo
$_GET['goodsCode']; ?>" />
34      |
35  <input class="grid_text" name="goodsName" value="<?php echo
$_GET['goodsName']; ?>" />
36   ○
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.UseMaterial

web/protected/views/UseMaterial/use_material_list.php, line 33 (Cross-Site Scripting: Reflected) — Critical

Sink Details

Sink: builtin_echo()
File: web/protected/views/UseMaterial/use_material_list.php:33
Taint Flags: WEB, XSS

```
30  <table>
31  <tr><td>
32      ¢
33  <input class="grid_text" name="goodsCode" value="<?php echo $_GET['goodsCode']; ?>" />
34      |
35  <input class="grid_text" name="goodsName" value="<?php echo $_GET['goodsName']; ?>" />
36   ○
```

web/protected/views/UseMaterial/use_material_list.php, line 37 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['glPro']
File: web/protected/views/UseMaterial/use_material_list.php:37

```
34      |
35  <input class="grid_text" name="goodsName" value="<?php echo
$_GET['goodsName']; ?>" />
36   ○
37  <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro']; ?
>" style="width:100px;" />
38  <?=$typeName;?>
39  <select class="grid_text" name="nature" style="height: 24px;">
40  <option></option>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/UseMaterial/use_material_list.php:37
Taint Flags: WEB, XSS

```
34      |
35  <input class="grid_text" name="goodsName" value="<?php echo $_GET['goodsName']; ?>" />
36   ○
37  <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro']; ?>" style="width:
100px;" />
38  <?=$typeName;?>
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.UseMaterial

web/protected/views/UseMaterial/use_material_list.php, line 37 (Cross-Site Scripting: Reflected) — Critical

```
39  <select class="grid_text" name="nature" style="height: 24px;">
40  <option></option>
```

web/protected/views/UseMaterial/select_list.php, line 20 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['goodsCode']
File: web/protected/views/UseMaterial/select_list.php:20

```
17
18  <input class="grid_text" name="batchCode" value="<?php echo
$_GET['batchCode'];?>" />
19      ¢ →
20  <input class="grid_text" name="goodsCode" value="<?php echo
$_GET['goodsCode'];?>" />
21      |
22  <input class="grid_text" name="goodsName" value="<?php echo
$_GET['goodsName'];?>" />
23  <input type="submit" value=" ← " class="grid_button grid_button_s">
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/UseMaterial/select_list.php:20
Taint Flags: WEB, XSS

```
17
18  <input class="grid_text" name="batchCode" value="<?php echo $_GET['batchCode'];?>" />
19      ¢ →
20  <input class="grid_text" name="goodsCode" value="<?php echo $_GET['goodsCode'];?>" />
21      |
22  <input class="grid_text" name="goodsName" value="<?php echo $_GET['goodsName'];?>" />
23  <input type="submit" value=" ← " class="grid_button grid_button_s">
```

Package: web.protected.views.auth

web/protected/views/auth/auth_item_list.php, line 32 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.auth

web/protected/views/auth/auth_item_list.php, line 32 (Cross-Site Scripting: Reflected) — Critical

Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['name']
File: web/protected/views/auth/auth_item_list.php:32

```
29  <td align="right"><form id="form" method="get" action="<?=Yii::app()->createUrl("auth/itemlist")?>">
30  <input type="hidden" name="type" id="type" />
31
32  <input class="grid_text" name="name" value="<?php echo $_GET['name'];?>">
33  <input type="submit" value="  " class="grid_button grid_button_s">
34  <input type="button" value="  " class="grid_button grid_button_s" onclick="Auth.setFormType('2')">
35  <input type="button" value="  " class="grid_button grid_button_s" onclick="Auth.setFormType('1')">
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/auth/auth_item_list.php:32
Taint Flags: WEB, XSS

```
29  <td align="right"><form id="form" method="get" action="<?=Yii::app()->createUrl("auth/itemlist")?>">
30  <input type="hidden" name="type" id="type" />
31
32  <input class="grid_text" name="name" value="<?php echo $_GET['name'];?>">
33  <input type="submit" value="  " class="grid_button grid_button_s">
34  <input type="button" value="  " class="grid_button grid_button_s" onclick="Auth.setFormType('2')">
35  <input type="button" value="  " class="grid_button grid_button_s" onclick="Auth.setFormType('1')">
```

Package: web.protected.views.inout

web/protected/views/inout/io_list.php, line 57 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['borrowerName']
File: web/protected/views/inout/io_list.php:57

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.inout

web/protected/views/inout/io_list.php, line 57 (Cross-Site Scripting: Reflected) — Critical

```
54  <input class="grid_text" name="property" value="<?php echo
$_GET['property']; ?>">
55  <?php else: ?>
56   ↓
57  <input class="grid_text" name="borrowerName" value="<?php echo
$_GET['borrowerName']; ?>" size="8">
58  <?php endif; ?>
59
60  <?php if ($_GET['inOut'] == 'out'): ?>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/inout/io_list.php:57
Taint Flags: WEB, XSS

```
54  <input class="grid_text" name="property" value="<?php echo $_GET['property']; ?>">
55  <?php else: ?>
56   ↓
57  <input class="grid_text" name="borrowerName" value="<?php echo $_GET['borrowerName']; ?>"
size="8">
58  <?php endif; ?>
59
60  <?php if ($_GET['inOut'] == 'out'): ?>
```

web/protected/views/inout/io_list.php, line 54 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['property']
File: web/protected/views/inout/io_list.php:54

```
51      ¢
52  <input class="grid_text" name="bh" value="<?php echo $_GET['bh']; ?>">
53
54  <input class="grid_text" name="property" value="<?php echo
$_GET['property']; ?>">
55  <?php else: ?>
56   ↓
57  <input class="grid_text" name="borrowerName" value="<?php echo
$_GET['borrowerName']; ?>" size="8">
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.inout

web/protected/views/inout/io_list.php, line 54 (Cross-Site Scripting: Reflected) — Critical

Sink Details

Sink: builtin_echo()
File: web/protected/views/inout/io_list.php:54
Taint Flags: WEB, XSS

```
51      ¢
52  <input class="grid_text" name="bh" value="<?php echo $_GET['bh']; ?>">
53
54  <input class="grid_text" name="property" value="<?php echo $_GET['property']; ?>">
55  <?php else: ?>
56   ↓
57  <input class="grid_text" name="borrowerName" value="<?php echo $_GET['borrowerName']; ?>"
size="8">
```

web/protected/views/inout/io_list.php, line 105 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['dend']
File: web/protected/views/inout/io_list.php:105

```
102
103  <input class="grid_text" name="dstart" style="width:80px" id="dstart"
value="<?php echo $_GET['dstart']; ?>">
104  -
105  <input class="grid_text" name="dend" style="width:80px" id="dend"
value="<?php echo $_GET['dend']; ?>">
106  <input type="submit" value=" ← " class="grid_button grid_button_s">
107  <?php if ($_GET['inOut'] == 'out'): ?>
108  <?php if ($_GET['h'] == '1'): ?>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/inout/io_list.php:105
Taint Flags: WEB, XSS

```
102
103  <input class="grid_text" name="dstart" style="width:80px" id="dstart" value="<?php echo
$_GET['dstart']; ?>">
104  -
105  <input class="grid_text" name="dend" style="width:80px" id="dend" value="<?php echo
$_GET['dend']; ?>">
106  <input type="submit" value=" ← " class="grid_button grid_button_s">
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.inout

web/protected/views/inout/io_list.php, line 105 (Cross-Site Scripting: Reflected) — Critical

```
107  <?php if ($_GET['inOut'] == 'out'): ?>
108  <?php if ($_GET['h'] == '1'): ?>
```

web/protected/views/inout/io_list.php, line 111 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['page']
File: web/protected/views/inout/io_list.php:111

```
108  <?php if ($_GET['h'] == '1'): ?>
109  <input type="button" value="  →" class="grid_button grid_button_s" onclick="location.href = '/inout/import.html'">
110  <?php else: ?>
111  <input type="button" value="   " class="grid_button grid_button_s" onclick="location.href = '/inout/export.html?sql=<?php echo $sql; ?>&page=<?php echo $_GET['page']; ?>'">
112  <?php endif; ?>
113  <?php endif; ?>
114  </td>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/inout/io_list.php:111
Taint Flags: WEB, XSS

```
108  <?php if ($_GET['h'] == '1'): ?>
109  <input type="button" value="  →" class="grid_button grid_button_s" onclick="location.href = '/inout/import.html'">
110  <?php else: ?>
111  <input type="button" value="   " class="grid_button grid_button_s" onclick="location.href = '/inout/export.html?sql=<?php echo $sql; ?>&page=<?php echo $_GET['page']; ?>'">
112  <?php endif; ?>
113  <?php endif; ?>
114  </td>
```

web/protected/views/inout/io_list.php, line 81 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.inout

web/protected/views/inout/io_list.php, line 81 (Cross-Site Scripting: Reflected) — Critical

Source: Read $_GET['whereGo']
File: web/protected/views/inout/io_list.php:81

```
78  <input class="grid_text" name="goodsName" value="<?php echo
$_GET['goodsName']; ?>" size="8">
79
80
81  <!-- <input class="grid_text" name="whereGo" value="<?php echo
$_GET['whereGo']; ?>" size="8">-->
82  <!--
83     → ¢
84  <select name="ioType" id="ioType">
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/inout/io_list.php:81
Taint Flags: WEB, XSS

```
78  <input class="grid_text" name="goodsName" value="<?php echo $_GET['goodsName']; ?>"
size="8">
79
80
81  <!-- <input class="grid_text" name="whereGo" value="<?php echo $_GET['whereGo']; ?>"
size="8">-->
82  <!--
83     → ¢
84  <select name="ioType" id="ioType">
```

web/protected/views/inout/io_list.php, line 52 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['bh']
File: web/protected/views/inout/io_list.php:52

```
49  <td><input type="hidden" name='inOut' value="<?php echo $_GET['inOut']; ?
>"/>
50  <?php if ($_GET['inOut'] == 'in'): ?>
51      ¢
52  <input class="grid_text" name="bh" value="<?php echo $_GET['bh']; ?>">
53
54  <input class="grid_text" name="property" value="<?php echo
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.inout

web/protected/views/inout/io_list.php, line 52 (Cross-Site Scripting: Reflected) — Critical

```
$_GET['property']; ?>">
55  <?php else: ?>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/inout/io_list.php:52
Taint Flags: WEB, XSS

```
49  <td><input type="hidden" name='inOut' value="<?php echo $_GET['inOut']; ?>"/>
50  <?php if ($_GET['inOut'] == 'in'): ?>
51      ¢
52  <input class="grid_text" name="bh" value="<?php echo $_GET['bh']; ?>">
53
54  <input class="grid_text" name="property" value="<?php echo $_GET['property']; ?>">
55  <?php else: ?>
```

web/protected/views/inout/io_list.php, line 71 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['bh']
File: web/protected/views/inout/io_list.php:71

```
68  ">
69  <?php if ($_GET['h'] == '1'): ?>
70      ¢
71  <input class="grid_text" name="bh" value="<?php echo $_GET['bh']; ?>">
72  <input type="hidden" name='h' value="1"/>
73  <?php endif; ?>
74  <?php endif; ?></td>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/inout/io_list.php:71
Taint Flags: WEB, XSS

```
68  ">
69  <?php if ($_GET['h'] == '1'): ?>
70      ¢
71  <input class="grid_text" name="bh" value="<?php echo $_GET['bh']; ?>">
72  <input type="hidden" name='h' value="1"/>
73  <?php endif; ?>
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.inout

web/protected/views/inout/io_list.php, line 71 (Cross-Site Scripting: Reflected) — Critical

```
74  <?php endif; ?></td>
```

web/protected/views/inout/edit_photo.php, line 2 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['id']
File: web/protected/views/inout/edit_photo.php:2

```
1 <form action="<?=Yii::app()->request->url?>" id="form" name="form"
enctype="multipart/form-data" style="padding: 10px;" method="post">
2  <input name="id" type="hidden" value="<?=$_GET['id']?>">
3  <table border="0" cellspacing="0" cellpadding="0" class="github_tb"
width="100%">
4  <tr>
5  <td>  ¬      </td>
6  <td>
7  <input type="file" name="in_photo"></td>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/inout/edit_photo.php:2
Taint Flags: WEB, XSS

```
1 <form action="<?=Yii::app()->request->url?>" id="form" name="form" enctype="multipart/form-
data" style="padding: 10px;" method="post">
2  <input name="id" type="hidden" value="<?=$_GET['id']?>">
3  <table border="0" cellspacing="0" cellpadding="0" class="github_tb" width="100%">
4  <tr>
5  <td>  ¬      </td>
6  <td>
7  <input type="file" name="in_photo"></td>
```

web/protected/views/inout/io_list.php, line 103 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['dstart']
File: web/protected/views/inout/io_list.php:103

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.inout

web/protected/views/inout/io_list.php, line 103 (Cross-Site Scripting: Reflected) — Critical

```
100  </select>
101  -->
102
103  <input class="grid_text" name="dstart" style="width:80px" id="dstart"
value="<?php echo $_GET['dstart']; ?>">
104  -
105  <input class="grid_text" name="dend" style="width:80px" id="dend"
value="<?php echo $_GET['dend']; ?>">
106  <input type="submit" value=" ← " class="grid_button grid_button_s">
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/inout/io_list.php:103
Taint Flags: WEB, XSS

```
100  </select>
101  -->
102
103  <input class="grid_text" name="dstart" style="width:80px" id="dstart" value="<?php echo
$_GET['dstart']; ?>">
104  -
105  <input class="grid_text" name="dend" style="width:80px" id="dend" value="<?php echo
$_GET['dend']; ?>">
106  <input type="submit" value=" ← " class="grid_button grid_button_s">
```

web/protected/views/inout/io_list.php, line 49 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['inOut']
File: web/protected/views/inout/io_list.php:49

```
46  <td align="right"><form method="get" action="<?= Yii::app()->
>createUrl("inout/list") ?>">
47  <table>
48  <tr>
49  <td><input type="hidden" name='inOut' value="<?php echo $_GET['inOut']; ?
>"/>
50  <?php if ($_GET['inOut'] == 'in'): ?>
51      ¢
52  <input class="grid_text" name="bh" value="<?php echo $_GET['bh']; ?>">
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.inout

web/protected/views/inout/io_list.php, line 49 (Cross-Site Scripting: Reflected) — Critical

Sink Details

Sink: builtin_echo()
File: web/protected/views/inout/io_list.php:49
Taint Flags: WEB, XSS

```
46  <td align="right"><form method="get" action="<?= Yii::app()->createUrl("inout/list") ?>">
47  <table>
48  <tr>
49  <td><input type="hidden" name='inOut' value="<?php echo $_GET['inOut']; ?>"/>
50  <?php if ($_GET['inOut'] == 'in'): ?>
51      ¢
52  <input class="grid_text" name="bh" value="<?php echo $_GET['bh']; ?>">
```

web/protected/views/inout/io_list.php, line 78 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['goodsName']
File: web/protected/views/inout/io_list.php:78

```
75  </tr>
76  <tr>
77  <td>
78  <input class="grid_text" name="goodsName" value="<?php echo
$_GET['goodsName']; ?>" size="8">
79
80
81  <!-- <input class="grid_text" name="whereGo" value="<?php echo
$_GET['whereGo']; ?>" size="8">-->
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/inout/io_list.php:78
Taint Flags: WEB, XSS

```
75  </tr>
76  <tr>
77  <td>
78  <input class="grid_text" name="goodsName" value="<?php echo $_GET['goodsName']; ?>"
size="8">
79
80
81  <!-- <input class="grid_text" name="whereGo" value="<?php echo $_GET['whereGo']; ?>"
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.inout

web/protected/views/inout/io_list.php, line 78 (Cross-Site Scripting: Reflected) — Critical

```
size="8">-->
```

Package: web.protected.views.inventory

web/protected/views/inventory/inventory_list.php, line 12 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['sdate']
File: web/protected/views/inventory/inventory_list.php:12

```
9  ?></td>
10  <td align="right"><form method="get" action="<?= Yii::app()->createUrl("inventory/list") ?>">
11     ¦
12  <input class="grid_text" type="date" name="sdate" value="<?php echo $_GET['sdate']; ?>">
13
14  <input class="grid_text" type="date" name="edate" value="<?php echo $_GET['edate']; ?>">
15  <input type="submit" value=" ← " class="grid_button grid_button_s">
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/inventory/inventory_list.php:12
Taint Flags: WEB, XSS

```
9  ?></td>
10  <td align="right"><form method="get" action="<?= Yii::app()->createUrl("inventory/list") ?>">
11     ¦
12  <input class="grid_text" type="date" name="sdate" value="<?php echo $_GET['sdate']; ?>">
13
14  <input class="grid_text" type="date" name="edate" value="<?php echo $_GET['edate']; ?>">
15  <input type="submit" value=" ← " class="grid_button grid_button_s">
```

web/protected/views/inventory/inventory_list.php, line 14 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.inventory

web/protected/views/inventory/inventory_list.php, line 14 (Cross-Site Scripting: Reflected) — Critical

Source Details

Source: Read $_GET['edate']
File: web/protected/views/inventory/inventory_list.php:14

```
11      ¦
12   <input class="grid_text" type="date" name="sdate" value="<?php echo
$_GET['sdate']; ?>">
13
14   <input class="grid_text" type="date" name="edate" value="<?php echo
$_GET['edate']; ?>">
15   <input type="submit" value=" ← " class="grid_button grid_button_s">
16   </form></td>
17   </tr>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/inventory/inventory_list.php:14
Taint Flags: WEB, XSS

```
11      ¦
12   <input class="grid_text" type="date" name="sdate" value="<?php echo $_GET['sdate']; ?>">
13
14   <input class="grid_text" type="date" name="edate" value="<?php echo $_GET['edate']; ?>">
15   <input type="submit" value=" ← " class="grid_button grid_button_s">
16   </form></td>
17   </tr>
```

Package: web.protected.views.layouts

web/protected/views/layouts/layout_header.php, line 25 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['name']
File: web/protected/views/layouts/layout_header.php:25

```
22   <form action="" method="get" style="display:none;">
23   <div class="so_panel">
24   <div class="so_panel_div">
25   <input type="text" name="name" value="<?php echo $_GET['name'];?>"
class="so_input_text" />
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.layouts

web/protected/views/layouts/layout_header.php, line 25 (Cross-Site Scripting: Reflected) — Critical

```
26  </div>
27  <button type="submit" class="so_input_btn">   </button>
28  </div>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/layouts/layout_header.php:25
Taint Flags: WEB, XSS

```
22  <form action="" method="get" style="display:none;">
23  <div class="so_panel">
24  <div class="so_panel_div">
25  <input type="text" name="name" value="<?php echo $_GET['name'];?>" class="so_input_text" />
26  </div>
27  <button type="submit" class="so_input_btn">   </button>
28  </div>
```

Package: web.protected.views.material

web/protected/views/material/material_list.php, line 24 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['goodsCode']
File: web/protected/views/material/material_list.php:24

```
21  </td>
22  <td align="right"><form method="get" action="<?= Yii::app()->createUrl("material/list") ?>">
23      ¢ →
24  <input class="grid_text" name="goodsCode" value="<?php echo $_GET['goodsCode']; ?>">
25      |
26  <input class="grid_text" name="goodsName" value="<?php echo $_GET['goodsName']; ?>">
27  <?php if (Auth::has(AI::R_Materialer)): ?>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/material/material_list.php:24

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.material

web/protected/views/material/material_list.php, line 24 (Cross-Site Scripting: Reflected) — Critical

Taint Flags: WEB, XSS

```
21  </td>
22  <td align="right"><form method="get" action="<?= Yii::app()->createUrl("material/list") ?>">
23      ¢ →
24  <input class="grid_text" name="goodsCode" value="<?php echo $_GET['goodsCode']; ?>">
25      |
26  <input class="grid_text" name="goodsName" value="<?php echo $_GET['goodsName']; ?>">
27  <?php if (Auth::has(AI::R_Materialer)): ?>
```

web/protected/views/material/warningl_list.php, line 69 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['style']
File: web/protected/views/material/warningl_list.php:69

```
66  <!--<select name="type" id="type">
67  <option value="">   </option>
68  </select> -->
69  <input type="text" name="style" id="style" value="<?=$_GET['style']?>" class="easyui-combobox" data-options="valueField:'value',textField:'label',url:'/goods/stylecombobox',panelHeight:100" />
70  <input type="submit" value=" ← " class="grid_button grid_button_s">
71  </form></td>
72  </tr>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/material/warningl_list.php:69
Taint Flags: WEB, XSS

```
66  <!--<select name="type" id="type">
67  <option value="">   </option>
68  </select> -->
69  <input type="text" name="style" id="style" value="<?=$_GET['style']?>" class="easyui-combobox" data-options="valueField:'value',textField:'label',url:'/goods/stylecombobox',panelHeight:100" />
70  <input type="submit" value=" ← " class="grid_button grid_button_s">
71  </form></td>
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.material

web/protected/views/material/warningl_list.php, line 69 (Cross-Site Scripting: Reflected) — Critical

```
72  </tr>
```

web/protected/views/material/in_form_list.php, line 30 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['informCode']
File: web/protected/views/material/in_form_list.php:30

```
27      ○
28   <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro']; ?>" />
29    → ¬
30   <input class="grid_text" name="informCode" value="<?php echo $_GET['informCode']; ?>" />
31   <input type="submit" value=" ← " class="grid_button grid_button_s" />
32   </td></tr>
33   </table>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/material/in_form_list.php:30
Taint Flags: WEB, XSS

```
27      ○
28   <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro']; ?>" />
29    → ¬
30   <input class="grid_text" name="informCode" value="<?php echo $_GET['informCode']; ?>" />
31   <input type="submit" value=" ← " class="grid_button grid_button_s" />
32   </td></tr>
33   </table>
```

web/protected/views/material/detail_list.php, line 13 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.material

web/protected/views/material/detail_list.php, line 13 (Cross-Site Scripting: Reflected) — Critical

Source: Read $_GET['goodsName']
File: web/protected/views/material/detail_list.php:13

```
10  ?></td>
11  <td align="right"><form method="get" action="<?=Yii::app()->createUrl("material/detaillist")?>">
12
13  <input class="grid_text" name="goodsName" value="<?php echo $_GET['goodsName'];?>">
14  <input type="submit" value=" ← " class="grid_button grid_button_s">
15  </form></td>
16  </tr>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/material/detail_list.php:13
Taint Flags: WEB, XSS

```
10  ?></td>
11  <td align="right"><form method="get" action="<?=Yii::app()->createUrl("material/detaillist")?>">
12
13  <input class="grid_text" name="goodsName" value="<?php echo $_GET['goodsName'];?>">
14  <input type="submit" value=" ← " class="grid_button grid_button_s">
15  </form></td>
16  </tr>
```

web/protected/views/material/select_list.php, line 45 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['glProCode']
File: web/protected/views/material/select_list.php:45

```
42          ○
43  <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro'];?>" />
44          ○  ¢
45  <input class="grid_text" name="glProCode" value="<?php echo $_GET['glProCode'];?>" />
46  <input type="submit" value=" ← " class="grid_button grid_button_s" />
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.material

web/protected/views/material/select_list.php, line 45 (Cross-Site Scripting: Reflected) — Critical

```
47  </td>
48  </tr>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/material/select_list.php:45
Taint Flags: WEB, XSS

```
42          ○
43  <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro'];?>" />
44          ○  ¢
45  <input class="grid_text" name="glProCode" value="<?php echo $_GET['glProCode'];?>" />
46  <input type="submit" value=" ← " class="grid_button grid_button_s" />
47  </td>
48  </tr>
```

web/protected/views/material/move_form_list.php, line 46 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['batchCode']
File: web/protected/views/material/move_form_list.php:46

```
43      ¬
44  <input class="grid_text" name="moveFormCode" value="<?php echo
$_GET['moveFormCode']; ?>" />
45
46  <input class="grid_text" name="batchCode" value="<?php echo
$_GET['batchCode']; ?>" />
47  <input type="submit" value=" ← " class="grid_button grid_button_s" />
48  </td></tr>
49  </table>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/material/move_form_list.php:46
Taint Flags: WEB, XSS

```
43      ¬
44  <input class="grid_text" name="moveFormCode" value="<?php echo $_GET['moveFormCode']; ?>" /
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.material

web/protected/views/material/move_form_list.php, line 46 (Cross-Site Scripting: Reflected) — Critical

```
>
45
46  <input class="grid_text" name="batchCode" value="<?php echo $_GET['batchCode']; ?>" />
47  <input type="submit" value="  " class="grid_button grid_button_s" />
48  </td></tr>
49  </table>
```

web/protected/views/material/select_list.php, line 35 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['goodsName']
File: web/protected/views/material/select_list.php:35

```
32  ?>
33  </select>
34
35  <input class="grid_text" name="goodsName" value="<?php echo
$_GET['goodsName'];?>" />
36
37  <input class="grid_text" name="goodsCode" value="<?php echo
$_GET['goodsCode'];?>" />
38  </td>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/material/select_list.php:35
Taint Flags: WEB, XSS

```
32  ?>
33  </select>
34
35  <input class="grid_text" name="goodsName" value="<?php echo $_GET['goodsName'];?>" />
36
37  <input class="grid_text" name="goodsCode" value="<?php echo $_GET['goodsCode'];?>" />
38  </td>
```

web/protected/views/material/warningl_list.php, line 57 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.material

web/protected/views/material/warningl_list.php, line 57 (Cross-Site Scripting: Reflected) — Critical

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['goodsName']
File: web/protected/views/material/warningl_list.php:57

```
54  ?></td>
55  <td align="right"><form method="get" action="<?=Yii::app()->createUrl("material/warning")?>">
56
57  <input class="grid_text" name="goodsName" value="<?php echo $_GET['goodsName'];?>">
58  <!--  ¢
59  <select name="category" id="category" onchange="Goods.callGetAjaxTypeList(this.value,true);">
60  <option value="">  </option>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/material/warningl_list.php:57
Taint Flags: WEB, XSS

```
54  ?></td>
55  <td align="right"><form method="get" action="<?=Yii::app()->createUrl("material/warning")?>">
56
57  <input class="grid_text" name="goodsName" value="<?php echo $_GET['goodsName'];?>">
58  <!--  ¢
59  <select name="category" id="category" onchange="Goods.callGetAjaxTypeList(this.value,true);">
60  <option value="">  </option>
```

web/protected/views/material/material_list.php, line 26 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['goodsName']
File: web/protected/views/material/material_list.php:26

```
23      ¢ →
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.material

web/protected/views/material/material_list.php, line 26 (Cross-Site Scripting: Reflected) — Critical

```
24  <input class="grid_text" name="goodsCode" value="<?php echo
$_GET['goodsCode']; ?>">
25      |
26  <input class="grid_text" name="goodsName" value="<?php echo
$_GET['goodsName']; ?>">
27  <?php if (Auth::has(AI::R_Materialer)): ?>
28    ¬
29  <select class="grid_text" name="storeID" id="storeID" style="height:
24px">
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/material/material_list.php:26
Taint Flags: WEB, XSS

```
23      ¢ →
24  <input class="grid_text" name="goodsCode" value="<?php echo $_GET['goodsCode']; ?>">
25      |
26  <input class="grid_text" name="goodsName" value="<?php echo $_GET['goodsName']; ?>">
27  <?php if (Auth::has(AI::R_Materialer)): ?>
28    ¬
29  <select class="grid_text" name="storeID" id="storeID" style="height: 24px">
```

web/protected/views/material/move_form_list.php, line 44 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['moveFormCode']
File: web/protected/views/material/move_form_list.php:44

```
41  ?>
42  </select>
43    ¬
44  <input class="grid_text" name="moveFormCode" value="<?php echo
$_GET['moveFormCode']; ?>" />
45
46  <input class="grid_text" name="batchCode" value="<?php echo
$_GET['batchCode']; ?>" />
47  <input type="submit" value=" ← " class="grid_button grid_button_s" />
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.material

web/protected/views/material/move_form_list.php, line 44 (Cross-Site Scripting: Reflected) — Critical

Sink Details

Sink: builtin_echo()
File: web/protected/views/material/move_form_list.php:44
Taint Flags: WEB, XSS

```
41  ?>
42  </select>
43    ¬
44  <input class="grid_text" name="moveFormCode" value="<?php echo $_GET['moveFormCode']; ?>" /
>
45
46  <input class="grid_text" name="batchCode" value="<?php echo $_GET['batchCode']; ?>" />
47  <input type="submit" value=" ← " class="grid_button grid_button_s" />
```

web/protected/views/material/in_form_list.php, line 28 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['glPro']
File: web/protected/views/material/in_form_list.php:28

```
25      ○   ₵
26  <input class="grid_text" name="glProCode" value="<?php echo
$_GET['glProCode']; ?>" />
27      ○
28  <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro']; ?
>" />
29    → ¬
30  <input class="grid_text" name="informCode" value="<?php echo
$_GET['informCode']; ?>" />
31  <input type="submit" value=" ← " class="grid_button grid_button_s" />
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/material/in_form_list.php:28
Taint Flags: WEB, XSS

```
25      ○   ₵
26  <input class="grid_text" name="glProCode" value="<?php echo $_GET['glProCode']; ?>" />
27      ○
28  <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro']; ?>" />
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.material

web/protected/views/material/in_form_list.php, line 28 (Cross-Site Scripting: Reflected) — Critical

```
29   → ¬
30  <input class="grid_text" name="informCode" value="<?php echo $_GET['informCode']; ?>" />
31  <input type="submit" value=" ← " class="grid_button grid_button_s" />
```

web/protected/views/material/in_form_list.php, line 26 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['glProCode']
File: web/protected/views/material/in_form_list.php:26

```
23  <table>
24  <tr><td>
25      ○  ₵
26  <input class="grid_text" name="glProCode" value="<?php echo $_GET['glProCode']; ?>" />
27      ○
28  <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro']; ?>" />
29   → ¬
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/material/in_form_list.php:26
Taint Flags: WEB, XSS

```
23  <table>
24  <tr><td>
25      ○  ₵
26  <input class="grid_text" name="glProCode" value="<?php echo $_GET['glProCode']; ?>" />
27      ○
28  <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro']; ?>" />
29   → ¬
```

web/protected/views/material/select_list.php, line 43 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.material

web/protected/views/material/select_list.php, line 43 (Cross-Site Scripting: Reflected) — Critical

Source Details

Source: Read $_GET['glPro']
File: web/protected/views/material/select_list.php:43

```
40  <tr height="30">
41  <td>
42          ○
43  <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro'];?>" />
44          ○  ¢
45  <input class="grid_text" name="glProCode" value="<?php echo $_GET['glProCode'];?>" />
46  <input type="submit" value="← " class="grid_button grid_button_s" />
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/material/select_list.php:43
Taint Flags: WEB, XSS

```
40  <tr height="30">
41  <td>
42          ○
43  <input class="grid_text" name="glPro" value="<?php echo $_GET['glPro'];?>" />
44          ○  ¢
45  <input class="grid_text" name="glProCode" value="<?php echo $_GET['glProCode'];?>" />
46  <input type="submit" value="← " class="grid_button grid_button_s" />
```

web/protected/views/material/select_list.php, line 37 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['goodsCode']
File: web/protected/views/material/select_list.php:37

```
34      |
35  <input class="grid_text" name="goodsName" value="<?php echo $_GET['goodsName'];?>" />
36      ¢ →
37  <input class="grid_text" name="goodsCode" value="<?php echo $_GET['goodsCode'];?>" />
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.material

web/protected/views/material/select_list.php, line 37 (Cross-Site Scripting: Reflected) — Critical

```
38  </td>
39  </tr>
40  <tr height="30">
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/material/select_list.php:37
Taint Flags: WEB, XSS

```
34      |
35  <input class="grid_text" name="goodsName" value="<?php echo $_GET['goodsName'];?>" />
36      ¢ →
37  <input class="grid_text" name="goodsCode" value="<?php echo $_GET['goodsCode'];?>" />
38  </td>
39  </tr>
40  <tr height="30">
```

Package: web.protected.views.prefloodmaterial

web/protected/views/prefloodmaterial/material_info.php, line 64 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['name']
File: web/protected/views/prefloodmaterial/material_info.php:64

```
61  <option value="    ¦    "<?php if($_GET['className']==="    ¦    ")echo
"selected";?>>    ¦    </option>
62  </select>
63
64  <input class="grid_text" name="name" value="<?php echo $_GET['name']; ?>">
65  <input type="submit" value=" ← " class="grid_button grid_button_s">
66  <button type="button" rel="add" class="grid_button">   </button>
67  <input type="button" class="grid_button grid_button_s"
id="controlDisplay" value="     ₩" />
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/prefloodmaterial/material_info.php:64
Taint Flags: WEB, XSS

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.prefloodmaterial

web/protected/views/prefloodmaterial/material_info.php, line 64 (Cross-Site Scripting: Reflected) — Critical

```
61  <option value="    ¦     "<?php if($_GET['className']==="    ¦     ")echo "selected";?>>
¦     </option>
62  </select>
63
64  <input class="grid_text" name="name" value="<?php echo $_GET['name']; ?>">
65  <input type="submit" value=" ← " class="grid_button grid_button_s">
66  <button type="button" rel="add" class="grid_button">   </button>
67  <input type="button" class="grid_button grid_button_s" id="controlDisplay" value="     ₩
" />
```

web/protected/views/prefloodmaterial/material_list.php, line 64 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['name']
File: web/protected/views/prefloodmaterial/material_list.php:64

```
61  <option value="    ¦     "<?php if($_GET['className']==="    ¦     ")echo
"selected";?>>    ¦     </option>
62  </select>
63
64  <input class="grid_text" name="name" value="<?php echo $_GET['name']; ?>">
65  <input type="submit" value=" ← " class="grid_button grid_button_s">
66  <button type="button" rel="add" class="grid_button">   </button>
67  <input type="button" class="grid_button grid_button_s"
id="controlDisplay" value="     ₩" />
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/prefloodmaterial/material_list.php:64
Taint Flags: WEB, XSS

```
61  <option value="    ¦     "<?php if($_GET['className']==="    ¦     ")echo "selected";?>>
¦     </option>
62  </select>
63
64  <input class="grid_text" name="name" value="<?php echo $_GET['name']; ?>">
65  <input type="submit" value=" ← " class="grid_button grid_button_s">
66  <button type="button" rel="add" class="grid_button">   </button>
67  <input type="button" class="grid_button grid_button_s" id="controlDisplay" value="     ₩
" />
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.scrap

web/protected/views/scrap/scrap_form_list.php, line 30 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['type']
File: web/protected/views/scrap/scrap_form_list.php:30

```
27  <form method="get" action="<?= Yii::app()->createUrl("scrap/
ScrapFormList") ?>">
28  <table>
29  <tr><td>
30  <input type="hidden" value="<?php echo $_GET['type'];?>" name="type" />
31
32  <input class="grid_text" name="formCode" value="<?php echo
$_GET['formCode']; ?>" />
33   o
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/scrap/scrap_form_list.php:30
Taint Flags: WEB, XSS

```
27  <form method="get" action="<?= Yii::app()->createUrl("scrap/ScrapFormList") ?>">
28  <table>
29  <tr><td>
30  <input type="hidden" value="<?php echo $_GET['type'];?>" name="type" />
31
32  <input class="grid_text" name="formCode" value="<?php echo $_GET['formCode']; ?>" />
33   o
```

web/protected/views/scrap/scrap_form_list.php, line 32 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['formCode']
File: web/protected/views/scrap/scrap_form_list.php:32

```
29  <tr><td>
30  <input type="hidden" value="<?php echo $_GET['type'];?>" name="type" />
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.scrap

web/protected/views/scrap/scrap_form_list.php, line 32 (Cross-Site Scripting: Reflected) — Critical

```
31
32  <input class="grid_text" name="formCode" value="<?php echo
$_GET['formCode']; ?>" />
33   ○
34  <input class="grid_text" name="projectName" value="<?php echo
$_GET['projectName']; ?>" />
35  <input type="submit" value=" ← " class="grid_button grid_button_s" />
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/scrap/scrap_form_list.php:32
Taint Flags: WEB, XSS

```
29  <tr><td>
30  <input type="hidden" value="<?php echo $_GET['type'];?>" name="type" />
31
32  <input class="grid_text" name="formCode" value="<?php echo $_GET['formCode']; ?>" />
33   ○
34  <input class="grid_text" name="projectName" value="<?php echo $_GET['projectName']; ?>" />
35  <input type="submit" value=" ← " class="grid_button grid_button_s" />
```

web/protected/views/scrap/scrap_form_list.php, line 34 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['projectName']
File: web/protected/views/scrap/scrap_form_list.php:34

```
31
32  <input class="grid_text" name="formCode" value="<?php echo
$_GET['formCode']; ?>" />
33   ○
34  <input class="grid_text" name="projectName" value="<?php echo
$_GET['projectName']; ?>" />
35  <input type="submit" value=" ← " class="grid_button grid_button_s" />
36  </td></tr>
37  </table>
```

Sink Details

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.scrap

web/protected/views/scrap/scrap_form_list.php, line 34 (Cross-Site Scripting: Reflected) — Critical

Sink: builtin_echo()
File: web/protected/views/scrap/scrap_form_list.php:34
Taint Flags: WEB, XSS

```
31
32  <input class="grid_text" name="formCode" value="<?php echo $_GET['formCode']; ?>" />
33   ○
34  <input class="grid_text" name="projectName" value="<?php echo $_GET['projectName']; ?>" />
35  <input type="submit" value="← " class="grid_button grid_button_s" />
36  </td></tr>
37  </table>
```

Package: web.protected.views.store

web/protected/views/store/user_store_list.php, line 15 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['userName']
File: web/protected/views/store/user_store_list.php:15

```
12  <td align="right">
13  <form method="get" action="<?= Yii::app()->createUrl("userstore/list") ?>">
14
15  <input class="grid_text" name="userName" value="<?php echo $_GET['userName']; ?>">
16
17  <input type="submit" value="← " class="grid_button grid_button_s">
18  <input type="button" value="        ¬" class="grid_button" onclick="Store.bindUser()" />
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/store/user_store_list.php:15
Taint Flags: WEB, XSS

```
12  <td align="right">
13  <form method="get" action="<?= Yii::app()->createUrl("userstore/list") ?>">
14
15  <input class="grid_text" name="userName" value="<?php echo $_GET['userName']; ?>">
16
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.store

web/protected/views/store/user_store_list.php, line 15 (Cross-Site Scripting: Reflected) — Critical

```
17  <input type="submit" value="←  " class="grid_button grid_button_s">
18  <input type="button" value="        ¬" class="grid_button" onclick="Store.bindUser()" />
```

Package: web.protected.views.task

web/protected/views/task/task_book_list.php, line 37 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['type']
File: web/protected/views/task/task_book_list.php:37

```
34  <form method="get" action="<?= Yii::app()->createUrl("Task/
TaskBookList") ?>">
35  <table>
36  <tr><td>
37  <input type="hidden" value="<?php echo $_GET['type'];?>" name="type" />
38
39  <input class="grid_text date" type="date" name="sdate" value="<?php echo
$_GET['sdate']; ?>" />
40
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/task/task_book_list.php:37
Taint Flags: WEB, XSS

```
34  <form method="get" action="<?= Yii::app()->createUrl("Task/TaskBookList") ?>">
35  <table>
36  <tr><td>
37  <input type="hidden" value="<?php echo $_GET['type'];?>" name="type" />
38
39  <input class="grid_text date" type="date" name="sdate" value="<?php echo $_GET['sdate']; ?
>" />
40
```

web/protected/views/task/select_list.php, line 18 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.task

web/protected/views/task/select_list.php, line 18 (Cross-Site Scripting: Reflected) — Critical

Source Details

Source: Read $_GET['batchCode']
File: web/protected/views/task/select_list.php:18

```
15  <td align="right"style="width: 520px;">
16  <form method="get" action="<?=Yii::app()->createUrl("Task/SelectList")?>">
17
18  <input class="grid_text" name="batchCode" value="<?php echo
$_GET['batchCode'];?>" />
19      ¢ →
20  <input class="grid_text" name="goodsCode" value="<?php echo
$_GET['goodsCode'];?>" />
21      |
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/task/select_list.php:18
Taint Flags: WEB, XSS

```
15  <td align="right"style="width: 520px;">
16  <form method="get" action="<?=Yii::app()->createUrl("Task/SelectList")?>">
17
18  <input class="grid_text" name="batchCode" value="<?php echo $_GET['batchCode'];?>" />
19      ¢ →
20  <input class="grid_text" name="goodsCode" value="<?php echo $_GET['goodsCode'];?>" />
21      |
```

web/protected/views/task/select_list.php, line 22 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['goodsName']
File: web/protected/views/task/select_list.php:22

```
19      ¢ →
20  <input class="grid_text" name="goodsCode" value="<?php echo
$_GET['goodsCode'];?>" />
21      |
22  <input class="grid_text" name="goodsName" value="<?php echo
$_GET['goodsName'];?>" />
23  <input type="submit" value=" ← " class="grid_button grid_button_s">
24  </form>
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.task

web/protected/views/task/select_list.php, line 22 (Cross-Site Scripting: Reflected) — Critical

```
25   </td>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/task/select_list.php:22
Taint Flags: WEB, XSS

```
19       ¢ →
20   <input class="grid_text" name="goodsCode" value="<?php echo $_GET['goodsCode'];?>" />
21       |
22   <input class="grid_text" name="goodsName" value="<?php echo $_GET['goodsName'];?>" />
23   <input type="submit" value=" ← " class="grid_button grid_button_s">
24   </form>
25   </td>
```

web/protected/views/task/task_book_list.php, line 43 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['zrbz']
File: web/protected/views/task/task_book_list.php:43

```
40
41   <input class="grid_text date" type="date" name="edate" value="<?php echo
$_GET['edate']; ?>" />
42       ¢ ←
43   <input class="grid_text" name="zrbz" value="<?php echo $_GET['zrbz']; ?
>" />
44   <input type="submit" value=" ← " class="grid_button grid_button_s" />
45   </td></tr>
46   </table>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/task/task_book_list.php:43
Taint Flags: WEB, XSS

```
40
41   <input class="grid_text date" type="date" name="edate" value="<?php echo $_GET['edate']; ?
>" />
42       ¢ ←
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.task

web/protected/views/task/task_book_list.php, line 43 (Cross-Site Scripting: Reflected) — Critical

```
43  <input class="grid_text" name="zrbz" value="<?php echo $_GET['zrbz']; ?>" />
44  <input type="submit" value="↩ " class="grid_button grid_button_s" />
45  </td></tr>
46  </table>
```

web/protected/views/task/task_book_list.php, line 41 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['edate']
File: web/protected/views/task/task_book_list.php:41

```
38
39  <input class="grid_text date" type="date" name="sdate" value="<?php echo
$_GET['sdate']; ?>" />
40
41  <input class="grid_text date" type="date" name="edate" value="<?php echo
$_GET['edate']; ?>" />
42      ₵ ↩
43  <input class="grid_text" name="zrbz" value="<?php echo $_GET['zrbz']; ?
>" />
44  <input type="submit" value="↩ " class="grid_button grid_button_s" />
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/task/task_book_list.php:41
Taint Flags: WEB, XSS

```
38
39  <input class="grid_text date" type="date" name="sdate" value="<?php echo $_GET['sdate']; ?
>" />
40
41  <input class="grid_text date" type="date" name="edate" value="<?php echo $_GET['edate']; ?
>" />
42      ₵ ↩
43  <input class="grid_text" name="zrbz" value="<?php echo $_GET['zrbz']; ?>" />
44  <input type="submit" value="↩ " class="grid_button grid_button_s" />
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.task

web/protected/views/task/task_book_list.php, line 39 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['sdate']
File: web/protected/views/task/task_book_list.php:39

```
36  <tr><td>
37  <input type="hidden" value="<?php echo $_GET['type'];?>" name="type" />
38
39  <input class="grid_text date" type="date" name="sdate" value="<?php echo $_GET['sdate']; ?>" />
40
41  <input class="grid_text date" type="date" name="edate" value="<?php echo $_GET['edate']; ?>" />
42      ¢ ←
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/task/task_book_list.php:39
Taint Flags: WEB, XSS

```
36  <tr><td>
37  <input type="hidden" value="<?php echo $_GET['type'];?>" name="type" />
38
39  <input class="grid_text date" type="date" name="sdate" value="<?php echo $_GET['sdate']; ?>" />
40
41  <input class="grid_text date" type="date" name="edate" value="<?php echo $_GET['edate']; ?>" />
42      ¢ ←
```

web/protected/views/task/select_list.php, line 20 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['goodsCode']
File: web/protected/views/task/select_list.php:20

```
17
18  <input class="grid_text" name="batchCode" value="<?php echo
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.task

web/protected/views/task/select_list.php, line 20 (Cross-Site Scripting: Reflected) — Critical

```
$_GET['batchCode'];?>" />
19      ¢ →
20  <input class="grid_text" name="goodsCode" value="<?php echo
$_GET['goodsCode'];?>" />
21      |
22  <input class="grid_text" name="goodsName" value="<?php echo
$_GET['goodsName'];?>" />
23  <input type="submit" value=" ← " class="grid_button grid_button_s">
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/task/select_list.php:20
Taint Flags: WEB, XSS

```
17
18  <input class="grid_text" name="batchCode" value="<?php echo $_GET['batchCode'];?>" />
19      ¢ →
20  <input class="grid_text" name="goodsCode" value="<?php echo $_GET['goodsCode'];?>" />
21      |
22  <input class="grid_text" name="goodsName" value="<?php echo $_GET['goodsName'];?>" />
23  <input type="submit" value=" ← " class="grid_button grid_button_s">
```

Package: web.protected.views.task2

web/protected/views/task2/task_book_list.php, line 37 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['type']
File: web/protected/views/task2/task_book_list.php:37

```
34  <form method="get" action="<?= Yii::app()->createUrl("Task/
TaskBookList") ?>">
35  <table>
36  <tr><td>
37  <input type="hidden" value="<?php echo $_GET['type'];?>" name="type" />
38
39  <input class="grid_text date" type="date" name="sdate" value="<?php echo
$_GET['sdate']; ?>" />
40
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.task2

web/protected/views/task2/task_book_list.php, line 37 (Cross-Site Scripting: Reflected) — Critical

Sink Details

Sink: builtin_echo()
File: web/protected/views/task2/task_book_list.php:37
Taint Flags: WEB, XSS

```
34  <form method="get" action="<?= Yii::app()->createUrl("Task/TaskBookList") ?>">
35  <table>
36  <tr><td>
37  <input type="hidden" value="<?php echo $_GET['type'];?>" name="type" />
38
39  <input class="grid_text date" type="date" name="sdate" value="<?php echo $_GET['sdate']; ?>" />
40
```

web/protected/views/task2/task_book_list.php, line 43 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['zrbz']
File: web/protected/views/task2/task_book_list.php:43

```
40
41  <input class="grid_text date" type="date" name="edate" value="<?php echo $_GET['edate']; ?>" />
42      ¢ ←
43  <input class="grid_text" name="zrbz" value="<?php echo $_GET['zrbz']; ?>" />
44  <input type="submit" value=" ← " class="grid_button grid_button_s" />
45  </td></tr>
46  </table>
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/task2/task_book_list.php:43
Taint Flags: WEB, XSS

```
40
41  <input class="grid_text date" type="date" name="edate" value="<?php echo $_GET['edate']; ?>" />
42      ¢ ←
43  <input class="grid_text" name="zrbz" value="<?php echo $_GET['zrbz']; ?>" />
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.task2

web/protected/views/task2/task_book_list.php, line 43 (Cross-Site Scripting: Reflected) — Critical

```
44  <input type="submit" value="↩ " class="grid_button grid_button_s" />
45  </td></tr>
46  </table>
```

web/protected/views/task2/task_book_list.php, line 41 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['edate']
File: web/protected/views/task2/task_book_list.php:41

```
38
39  <input class="grid_text date" type="date" name="sdate" value="<?php echo $_GET['sdate']; ?>" />
40
41  <input class="grid_text date" type="date" name="edate" value="<?php echo $_GET['edate']; ?>" />
42      ¢ ↩
43  <input class="grid_text" name="zrbz" value="<?php echo $_GET['zrbz']; ?>" />
44  <input type="submit" value="↩ " class="grid_button grid_button_s" />
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/task2/task_book_list.php:41
Taint Flags: WEB, XSS

```
38
39  <input class="grid_text date" type="date" name="sdate" value="<?php echo $_GET['sdate']; ?>" />
40
41  <input class="grid_text date" type="date" name="edate" value="<?php echo $_GET['edate']; ?>" />
42      ¢ ↩
43  <input class="grid_text" name="zrbz" value="<?php echo $_GET['zrbz']; ?>" />
44  <input type="submit" value="↩ " class="grid_button grid_button_s" />
```

web/protected/views/task2/task_book_list.php, line 39 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.task2

web/protected/views/task2/task_book_list.php, line 39 (Cross-Site Scripting: Reflected) — Critical

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['sdate']
File: web/protected/views/task2/task_book_list.php:39

```
36   <tr><td>
37   <input type="hidden" value="<?php echo $_GET['type'];?>" name="type" />
38
39   <input class="grid_text date" type="date" name="sdate" value="<?php echo $_GET['sdate']; ?>" />
40
41   <input class="grid_text date" type="date" name="edate" value="<?php echo $_GET['edate']; ?>" />
42       ¢ ←
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/task2/task_book_list.php:39
Taint Flags: WEB, XSS

```
36  <tr><td>
37  <input type="hidden" value="<?php echo $_GET['type'];?>" name="type" />
38
39  <input class="grid_text date" type="date" name="sdate" value="<?php echo $_GET['sdate']; ?>" />
40
41  <input class="grid_text date" type="date" name="edate" value="<?php echo $_GET['edate']; ?>" />
42      ¢ ←
```

Package: web.protected.views.user

web/protected/views/user/user_list.php, line 21 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['userName']
File: web/protected/views/user/user_list.php:21

```
18   <td align="right">
```

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.user

web/protected/views/user/user_list.php, line 21 (Cross-Site Scripting: Reflected) — Critical

```
19  <form method="get" action="<?=Yii::app()->createUrl("user/list")?>">
20
21  <input class="grid_text" name="userName" value="<?php echo
$_GET['userName'];?>">
22
23  <input class="grid_text" name="loginName" value="<?php echo
$_GET['loginName'];?>">
24      ¬
```

Sink Details

Sink: builtin_echo()
File: web/protected/views/user/user_list.php:21
Taint Flags: WEB, XSS

```
18  <td align="right">
19  <form method="get" action="<?=Yii::app()->createUrl("user/list")?>">
20
21  <input class="grid_text" name="userName" value="<?php echo $_GET['userName'];?>">
22
23  <input class="grid_text" name="loginName" value="<?php echo $_GET['loginName'];?>">
24    ¬
```

web/protected/views/user/user_list.php, line 23 (Cross-Site Scripting: Reflected) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_GET['loginName']
File: web/protected/views/user/user_list.php:23

```
20
21  <input class="grid_text" name="userName" value="<?php echo
$_GET['userName'];?>">
22
23  <input class="grid_text" name="loginName" value="<?php echo
$_GET['loginName'];?>">
24      ¬
25  <select name="storeID" id="storeID" >
26  <option></option>
```

Sink Details

Sink: builtin_echo()

Cross-Site Scripting: Reflected — Critical

Package: web.protected.views.user

web/protected/views/user/user_list.php, line 23 (Cross-Site Scripting: Reflected) — Critical

File: web/protected/views/user/user_list.php:23
Taint Flags: WEB, XSS

```
20
21  <input class="grid_text" name="userName" value="<?php echo $_GET['userName'];?>">
22
23  <input class="grid_text" name="loginName" value="<?php echo $_GET['loginName'];?>">
24    ¬
25  <select name="storeID" id="storeID" >
26  <option></option>
```

Header Manipulation (8 issues)

Abstract

HTTP 响应头文件中包含未验证的数据会引发 cache-poisoning、cross-site scripting、cross-user defacement、page hijacking、cookie manipulation 或 open redirect。

Explanation

以下情况中会出现 Header Manipulation 漏洞：

1. 数据通过一个不可信赖的数据源进入 Web 应用程序，最常见的是 HTTP 请求。

2. 数据包含在一个 HTTP 响应头文件里，未经验证就发送给了 Web 用户。

如同许多软件安全漏洞一样，Header Manipulation 只是通向终端的一个途径，它本身并不是终端。从本质上看，这些漏洞是显而易见的：一个攻击者将恶意数据传送到易受攻击的应用程序，且该应用程序将数据包含在 HTTP 响应头文件中。

其中最常见的一种 Header Manipulation 攻击是 HTTP Response Splitting。为了成功地实施 HTTP Response Splitting 盗取，应用程序必须允许将那些包含 CR（回车，由 %0d 或 \r 指定）和 LF（换行，由 %0a 或 \n 指定）的字符输入到头文件中。攻击者利用这些字符不仅可以控制应用程序要发送的响应剩余头文件和正文，还可以创建完全受其控制的其他响应。

如今的许多现代应用程序服务器可以防止 HTTP 头文件感染恶意字符。例如，当新行传递到 `header()` 函数时，最新版本的 PHP 将生成一个警告并停止创建头文件。如果您的 PHP 版本能够阻止设置带有换行符的头文件，则其具备对 HTTP Response Splitting 的防御能力。然而，单纯地过滤换行符可能无法保证应用程序不受 Cookie Manipulation 或 Open Redirects 的攻击，因此必须在设置带有用户输入的 HTTP 头文件时采取措施。

示例： 下段代码会从 HTTP 请求读取位置，并在 HTTP 响应的位置字段的头文件中对其进行设置。

假设在请求中提交了一个由标准的字母和数字字符组成的字符串，如 "index.html"，那么包含此 cookie 的 HTTP 响应可能表现为以下形式：

```
HTTP/1.1 200 OK
...
location: index.html
...
```

然而，因为该位置的值由未经验证的用户输入组成，所以仅当提交给 `some_location` 的值不包含任何 CR 和 LF 字符时，响应才会保留这种形式。如果攻击者提交的是一个恶意字符串，比如“index.html\r\nHTTP/1.1 200 OK\r\n...”，那么 HTTP 响应就会被分割成以下形式的两个响应：

```
HTTP/1.1 200 OK
...
location: index.html
```

```
HTTP/1.1 200 OK
...
```

显然，第二个响应已完全由攻击者控制，攻击者可以用所需的头文件和正文内容构建该响应。攻击者可以构建任意 HTTP 响应，从而发起多种形式的攻击，包括：cross-user defacement、网络和浏览器 cache poisoning、cross-site scripting 和 page hijacking。

Cross-User Defacement：攻击者可以向一个易受攻击的服务器发出一个请求，导致服务器创建两个响应，其中第二个响应可能会被曲解为对其他请求的响应，而这一请求很可能是与服务器共享相同 TCP 连接的另一用户发出的。这种攻击可以通过以下方式实现：攻击者诱骗用户，让他们自己提交恶意请求；或在远程情况下，攻击者与用户共享同一个连接到服务器（如共享代理服务器）的 TCP 连接。最理想的情况是，攻击者只能通过这种做法让用户相信自己的应用程序已经遭受了黑客攻击，进而对应用程序的安全性失去信心。最糟糕的情况是，攻击者可能提供经特殊技术处理的内容，这些内容旨在模仿应用程序的执行方式，但会重定向用户的私人信息（如帐号和密码），将这些信息发送给攻击者。

Cache Poisoning： 如果多用户 Web 缓存或者单用户浏览器缓存将恶意构建的响应缓存起来，该响应的破坏力会更大。如果响应缓存在共享的 Web 缓存（如在代理服务器中常见的缓存）中，那么使用该缓存的所有用户都会不断收到恶意内容，直到清除该缓存项为止。同样，如果响应缓存在单个用户的浏览器中，那么在清除该缓存项以前，该用户会不断收到恶意内容。然而，影响仅局限于本地浏览器的用户。

Cross-Site Scripting：一旦攻击者控制了应用程序传送的响应，就可以选择多种恶意内容来传播给用户。Cross-Site Scripting 是最常见的攻击形式，这种攻击在响应中包含了恶意的 JavaScript 或其他代码，并在用户的浏览器中执行。基于 XSS 的攻击手段花样百出，几乎是无穷无尽的，但通常它们都会包含传输给攻击者的私人数据（如 Cookie 或者其他会话信息）。在攻击者的控制下，指引受害者进入恶意的网络内容；或者利用易受攻击的站点，对用户的机器进行其他恶意操作。对于易受攻击的应用程序用户，最常见且最危险的攻击就是使用 JavaScript 将会话和 authentication 信息返回给攻击者，而后攻击者就可以完全控制受害者的帐号了。

Page Hijacking：除了利用一个易受攻击的应用程序向用户传输恶意内容，还可以利用相同的根漏洞，将服务器生成的供用户使用的敏感内容重定向，转而供攻击者使用。攻击者通过提交一个会导致两个响应的请求，即服务器做出的预期响应和攻击者创建的响应，致使某个中间节点（如共享的代理服务器）误导服务器所生成的响应，将本来应传送给用户的响应错误地传给攻击者。因为攻击者创建的请求产生了两个响应，第一个被解析为针对攻击者请求做出的响应，第二个则被忽略。当用户通过同一 TCP 连接发出合法请求时，攻击者的请求已经在此处等候，并被解析为针对受害者这一请求的响应。这时，攻击者将第二个请求发送给服务器，代理服务器利用针对受害者（用户）的、由该服务器产生的这一请求对服务器做出响应，因此，针对受害者的这一响应中会包含所有头文件或正文中的敏感信息。

Cookie Manipulation：当与类似 Cross-Site Request Forgery 的攻击相结合时，攻击者就可以篡改、添加、甚至覆盖合法用户的 cookie。

Open Redirect：如果允许未验证的输入来控制重定向机制所使用的 URL，可能会有利于攻击者发动钓鱼攻击。

Recommendation

针对 Header Manipulation 的解决方法是，确保在适当位置进行输入验证并检验其属性是否正确。

由于 Header Manipulation 漏洞出现在应用程序的输出中包含恶意数据时，因此，合乎逻辑的做法是在应用程序输出数据前一刻对其进行验证。然而，由于 Web 应用程序常常会包含复杂而难以理解的代码，用以生成动态响应，因此，这一方法容易产生遗漏错误（遗漏验证）。降低这一风险的有效途径是对 Header Manipulation 也执行输入验证。

由于 Web 应用程序必须验证输入信息以避免出现其他漏洞（如 SQL Injection），因此，一种相对简单的解决方法是扩充应用程序现有的输入验证机制，增加针对 Header Manipulation 的检查。尽管具有一定的价值，但 Header Manipulation 输入验证并不能取代严格的输出验证。应用程序可能通过共享的数据存储或其他可信

赖的数据源接受输入，而该数据存储所接受的输入源可能并未执行适当的输入验证。因此，应用程序不能间接地依赖于该数据或其他任意数据的安全性。这就意味着，避免 Header Manipulation 漏洞的最佳方法是验证所有应用程序输入数据或向用户输出的数据。

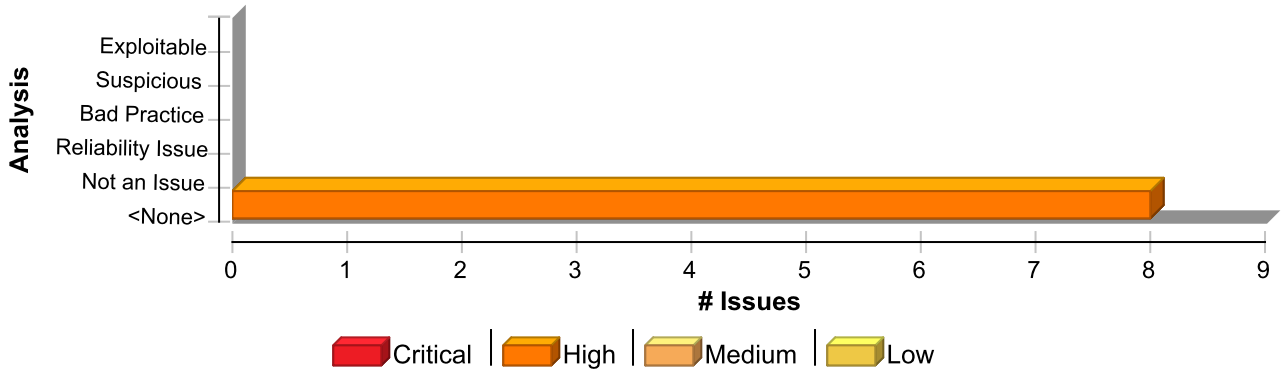
针对 Header Manipulation 漏洞进行验证最安全的方式是创建一份安全字符白名单，其中的字符允许出现在 HTTP 响应头文件中，并且只接受完全由这些受认可的字符组成的输入。例如，有效的用户名可能仅包含字母数字字符，帐号可能仅包含 0-9 的数字。

更灵活的解决方法称为黑名单方法，但其安全性较差，这种方法在进行输入之前就有选择地拒绝或避免了潜在的危险字符。为了创建这样的列表，首先需要了解在 HTTP 响应头文件中具有特殊含义的一组字符。尽管 CR 和 LF 字符是 HTTP Response Splitting 攻击的核心，但其他字符，如 ":"（冒号）和 "="（等号），在响应头文件中同样具有特殊的含义。

一旦在应用程序中确定了针对 Header Manipulation 攻击执行验证的正确点，以及验证过程中要考虑的特殊字符，下一个难题就是确定在验证过程中该如何处理各种特殊字符。应用程序应拒绝任何要添加到 HTTP 响应头文件中的包含特殊字符的输入，这些特殊字符（特别是 CR 和 LF）是无效字符。

许多应用程序服务器都试图避免应用程序出现 HTTP Response Splitting 漏洞，其做法是为负责设置 HTTP 头文件和 cookie 的函数提供各种执行方式，以检验是否存在进行 HTTP Response Splitting 攻击必需的字符。不要依赖运行应用程序的服务器，以此确保该应用程序的安全。开发了某个应用程序后，并不能保证在其生命周期中它会在哪些应用程序服务器中运行。由于标准和已知盗取方式的演变，我们不能保证应用程序服务器也会保持同步。

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Header Manipulation	8	0	0	8
Total	**8**	**0**	**0**	**8**

Header Manipulation	**High**
Package: framework	
framework/yiilite.php, line 2622 (Header Manipulation)	**High**

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Header Manipulation — High

Package: framework

framework/yiilite.php, line 2622 (Header Manipulation) — High

Source: Read $_SERVER['HTTP_HOST']
From: chttprequest.gethostinfo
File: framework/web/CHttpRequest.php:314

```
311  else
312  $http='http';
313  if(isset($_SERVER['HTTP_HOST']))
314  $this->_hostInfo=$http.'://'.$_SERVER['HTTP_HOST'];
315  else
316  {
317  $this->_hostInfo=$http.'://'.$_SERVER['SERVER_NAME'];
```

Sink Details

Sink: header()
Enclosing Method: redirect()
File: framework/yiilite.php:2622
Taint Flags: WEB, XSS

```
2619  {
2620  if(strpos($url,'/')===0 && strpos($url,'//')!==0)
2621  $url=$this->getHostInfo().$url;
2622  header('Location: '.$url, true, $statusCode);
2623  if($terminate)
2624  Yii::app()->end();
2625  }
```

framework/yiilite.php, line 2622 (Header Manipulation) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_SERVER['HTTP_HOST']
From: chttprequest.gethostinfo
File: framework/yiilite.php:2369

```
2366  else
2367  $http='http';
2368  if(isset($_SERVER['HTTP_HOST']))
2369  $this->_hostInfo=$http.'://'.$_SERVER['HTTP_HOST'];
2370  else
2371  {
2372  $this->_hostInfo=$http.'://'.$_SERVER['SERVER_NAME'];
```

Header Manipulation — High

Package: framework

framework/yiilite.php, line 2622 (Header Manipulation) — High

Sink Details

Sink: header()
Enclosing Method: redirect()
File: framework/yiilite.php:2622
Taint Flags: WEB, XSS

```
2619  {
2620  if(strpos($url,'/')===0 && strpos($url,'//')!==0)
2621  $url=$this->getHostInfo().$url;
2622  header('Location: '.$url, true, $statusCode);
2623  if($terminate)
2624  Yii::app()->end();
2625  }
```

framework/yiilite.php, line 2784 (Header Manipulation) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_SERVER['HTTP_RANGE']
From: chttprequest.sendfile
File: framework/yiilite.php:2763

```
2760  header("Content-Range: bytes $contentStart-$contentEnd/$fileSize");
2761  throw new CHttpException(416,'Requested Range Not Satisfiable');
2762  }
2763  $range=str_replace('bytes=','',$_SERVER['HTTP_RANGE']);
2764  //range requests starts from "-", so it means that data must be dumped
the end point.
2765  if($range[0]==='-')
2766  $contentStart=$fileSize-substr($range,1);
```

Sink Details

Sink: header()
Enclosing Method: sendfile()
File: framework/yiilite.php:2784
Taint Flags: WEB, XSS

```
2781  $wrongContentStart=($contentStart>$contentEnd || $contentStart>$fileSize-1 ||
$contentStart<0);
2782  if($wrongContentStart)
2783  {
2784  header("Content-Range: bytes $contentStart-$contentEnd/$fileSize");
2785  throw new CHttpException(416,'Requested Range Not Satisfiable');
```

Header Manipulation — High

Package: framework

framework/yiilite.php, line 2784 (Header Manipulation) — High

```
2786  }
2787  header('HTTP/1.1 206 Partial Content');
```

framework/yiilite.php, line 2788 (Header Manipulation) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_SERVER['HTTP_RANGE']
From: chttprequest.sendfile
File: framework/yiilite.php:2763

```
2760  header("Content-Range: bytes $contentStart-$contentEnd/$fileSize");
2761  throw new CHttpException(416,'Requested Range Not Satisfiable');
2762  }
2763  $range=str_replace('bytes=','',$_SERVER['HTTP_RANGE']);
2764  //range requests starts from "-", so it means that data must be dumped
the end point.
2765  if($range[0]==='-')
2766  $contentStart=$fileSize-substr($range,1);
```

Sink Details

Sink: header()
Enclosing Method: sendfile()
File: framework/yiilite.php:2788
Taint Flags: WEB, XSS

```
2785  throw new CHttpException(416,'Requested Range Not Satisfiable');
2786  }
2787  header('HTTP/1.1 206 Partial Content');
2788  header("Content-Range: bytes $contentStart-$contentEnd/$fileSize");
2789  }
2790  else
2791  header('HTTP/1.1 200 OK');
```

Package: framework.web

framework/web/CHttpRequest.php, line 803 (Header Manipulation) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Header Manipulation — High

Package: framework.web

framework/web/CHttpRequest.php, line 803 (Header Manipulation) — High

Source: Read $_SERVER['HTTP_HOST']
From: chttprequest.gethostinfo
File: framework/web/CHttpRequest.php:314

```
311  else
312  $http='http';
313  if(isset($_SERVER['HTTP_HOST']))
314  $this->_hostInfo=$http.'://'.$_SERVER['HTTP_HOST'];
315  else
316  {
317  $this->_hostInfo=$http.'://'.$_SERVER['SERVER_NAME'];
```

Sink Details

Sink: header()
Enclosing Method: redirect()
File: framework/web/CHttpRequest.php:803
Taint Flags: WEB, XSS

```
800  {
801  if(strpos($url,'/')===0 && strpos($url,'//')!==0)
802  $url=$this->getHostInfo().$url;
803  header('Location: '.$url, true, $statusCode);
804  if($terminate)
805  Yii::app()->end();
806  }
```

framework/web/CHttpRequest.php, line 803 (Header Manipulation) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_SERVER['HTTP_HOST']
From: chttprequest.gethostinfo
File: framework/yiilite.php:2369

```
2366  else
2367  $http='http';
2368  if(isset($_SERVER['HTTP_HOST']))
2369  $this->_hostInfo=$http.'://'.$_SERVER['HTTP_HOST'];
2370  else
2371  {
2372  $this->_hostInfo=$http.'://'.$_SERVER['SERVER_NAME'];
```

Header Manipulation — High

Package: framework.web

framework/web/CHttpRequest.php, line 803 (Header Manipulation) — High

Sink Details

Sink: header()
Enclosing Method: redirect()
File: framework/web/CHttpRequest.php:803
Taint Flags: WEB, XSS

```
800  {
801  if(strpos($url,'/')===0 && strpos($url,'//')!==0)
802  $url=$this->getHostInfo().$url;
803  header('Location: '.$url, true, $statusCode);
804  if($terminate)
805  Yii::app()->end();
806  }
```

framework/web/CHttpRequest.php, line 1044 (Header Manipulation) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_SERVER['HTTP_RANGE']
From: chttprequest.sendfile
File: framework/web/CHttpRequest.php:1018

```
1015  throw new CHttpException(416,'Requested Range Not Satisfiable');
1016  }
1017
1018  $range=str_replace('bytes=','',$_SERVER['HTTP_RANGE']);
1019
1020  //range requests starts from "-", so it means that data must be dumped
the end point.
1021  if($range[0]==='-')
```

Sink Details

Sink: header()
Enclosing Method: sendfile()
File: framework/web/CHttpRequest.php:1044
Taint Flags: WEB, XSS

```
1041
1042  if($wrongContentStart)
1043  {
1044  header("Content-Range: bytes $contentStart-$contentEnd/$fileSize");
1045  throw new CHttpException(416,'Requested Range Not Satisfiable');
1046  }
```

Header Manipulation — High

Package: framework.web

framework/web/CHttpRequest.php, line 1044 (Header Manipulation) — High

```
1047
```

framework/web/CHttpRequest.php, line 1049 (Header Manipulation) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_SERVER['HTTP_RANGE']
From: chttprequest.sendfile
File: framework/web/CHttpRequest.php:1018

```
1015  throw new CHttpException(416,'Requested Range Not Satisfiable');
1016  }
1017
1018  $range=str_replace('bytes=','',$_SERVER['HTTP_RANGE']);
1019
1020  //range requests starts from "-", so it means that data must be dumped
the end point.
1021  if($range[0]==='-')
```

Sink Details

Sink: header()
Enclosing Method: sendfile()
File: framework/web/CHttpRequest.php:1049
Taint Flags: WEB, XSS

```
1046  }
1047
1048  header('HTTP/1.1 206 Partial Content');
1049  header("Content-Range: bytes $contentStart-$contentEnd/$fileSize");
1050  }
1051  else
1052  header('HTTP/1.1 200 OK');
```

Often Misused: File Upload (10 issues)

Abstract

允许用户上传文件可能会让攻击者注入危险内容或恶意代码，并在服务器上运行。

Explanation

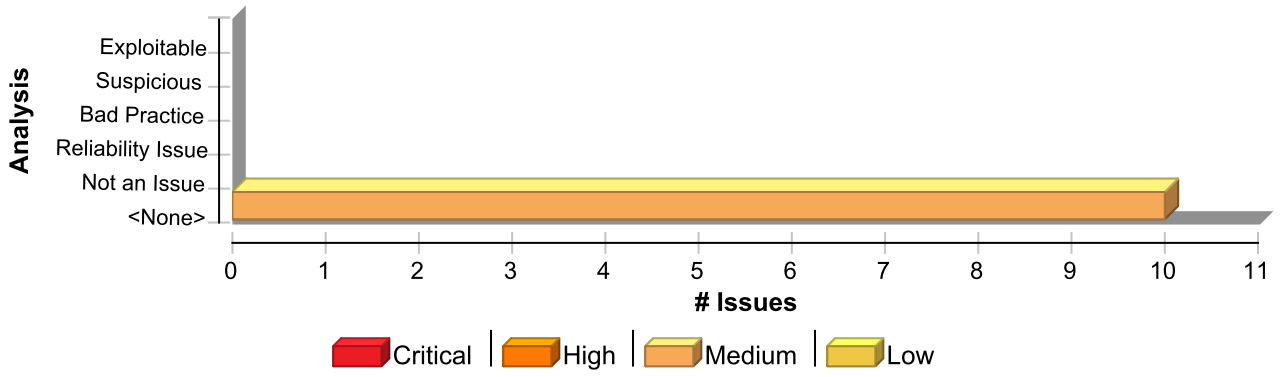
无论编写程序所用的语言是什么，最具破坏性的攻击通常都会涉及执行远程代码，攻击者借此可在程序上下文中成功执行恶意代码。如果允许攻击者向某个可通过 Web 访问的目录上传文件，并能够将这些文件传递给代码解释器（如 JSP/ASPX/PHP），他们就能促使这些文件中包含的恶意代码在服务器上执行。
即使程序将上传的文件存储在一个无法通过 Web 访问的目录中，攻击者仍然有可能通过向服务器环境引入恶意内容来发动其他攻击。如果程序容易出现 path manipulation、command injection 或危险的 file inclusion 漏洞，那么攻击者就可能上传带恶意内容的文件，并利用另一种漏洞促使程序读取或执行该文件。

类型为 `file` 的 标签表示程序接受文件上传。
示例：

Recommendation

如果可以避免上传文件，请不要允许此操作。如果程序必须允许文件上传，则应当只接受程序需要的特定类型的内容，从而阻止攻击者提供恶意内容。依赖于上传内容的攻击通常要求攻击者能够提供他们自行选择的内容。限制程序能够接受的内容，可以在最大程度上限制可能被攻击的范围。检查文件名、扩展名和文件内容，确保它们都是应用程序所需的，并可供应用程序使用。

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Often Misused: File Upload	10	0	0	10
Total	**10**	**0**	**0**	**10**

Often Misused: File Upload	**Medium**
Package: framework.web	
framework/web/CUploadedFile.php, line 198 (Often Misused: File Upload)	**Medium**
Issue Details	

Often Misused: File Upload — Medium

Package: framework.web

framework/web/CUploadedFile.php, line 198 (Often Misused: File Upload) — Medium

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: move_uploaded_file()
Enclosing Method: saveas()
File: framework/web/CUploadedFile.php:198
Taint Flags:

```
195  if($this->_error==UPLOAD_ERR_OK)
196  {
197  if($deleteTempFile)
198  return move_uploaded_file($this->_tempName,$file);
199  elseif(is_uploaded_file($this->_tempName))
200  return copy($this->_tempName, $file);
201  else
```

Package: web.protected.vendor.http

web/protected/vendor/http/WUpload.php, line 253 (Often Misused: File Upload) — Medium

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: move_uploaded_file()
Enclosing Method: save()
File: web/protected/vendor/http/WUpload.php:253
Taint Flags:

```
250  $this->fileName=$this->fileShortName.".".$this->fileExtName;
251  $this->fileSaveFullName=$this->savePath.$this->fileName;
252  //  ₩
253  $result=move_uploaded_file($this->sourceData['tmp_name'], $this-
>getFileSaveFullName("gbk"));
254  if (!$result) {
255  throw new Exception(self::$infoList['save_fail']);
256  }
```

Package: web.protected.views.UseMaterial

web/protected/views/UseMaterial/upload_file.php, line 46 (Often Misused: File Upload) — Medium

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Content)

Often Misused: File Upload — Medium

Package: web.protected.views.UseMaterial

web/protected/views/UseMaterial/upload_file.php, line 46 (Often Misused: File Upload) — Medium

Sink Details

File: web/protected/views/UseMaterial/upload_file.php:46
Taint Flags:

```
43  endforeach;?>
44  <tr>
45  <form id="form" name="form" class="" action="/UseMaterial/uploadFile" method="post"
enctype="multipart/form-data">
46  <td><label>       </label><input type="file" name="file" />
47  <input type="hidden" name="formID" value="<?=$formID?>" /></td>
48  <input type="hidden" name="type" value="<?=$type?>" /></td>
49  <td><button type="submit" class="grid_button"> £  </button></td>
```

web/protected/views/UseMaterial/ReturnMF.php, line 54 (Often Misused: File Upload) — Medium

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Content)

Sink Details

File: web/protected/views/UseMaterial/ReturnMF.php:54
Taint Flags:

```
51  </tr>
52 <!-- <tr style="height: 20px;">-->
53 <!-- <td width="70" align="right"><strong>     </strong></td>-->
54 <!-- <td width="100"><input id="pic" name="pic" type="file" /></td>-->
55 <!-- <td colspan="3" style="color: #8F8F8F">   ¯ ¬ ₩  ↓       </td>-->
56 <!-- </tr>-->
57  <tr style="height: 50px;">
```

web/protected/views/UseMaterial/edit_photo.php, line 7 (Often Misused: File Upload) — Medium

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Content)

Sink Details

File: web/protected/views/UseMaterial/edit_photo.php:7
Taint Flags:

```
4  <tr>
5  <td>     </td>
6  <td>
```

Often Misused: File Upload — Medium

Package: web.protected.views.UseMaterial

web/protected/views/UseMaterial/edit_photo.php, line 7 (Often Misused: File Upload) — Medium

```
7  <input type="file" name="pic"></td>
8  </tr>
9  <tr>
10  <td> </td>
```

Package: web.protected.views.inout

web/protected/views/inout/import.php, line 48 (Often Misused: File Upload) — Medium

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Content)

Sink Details

File: web/protected/views/inout/import.php:48
Taint Flags:

```
45  <tr>
46  <td align="right">Excel  </td>
47  <td>
48  <input type="file" name="file" id="file" />
49  <br />
50  <a href="/res/mod_in_out.xls">      ¥</a></td>
51  </tr>
```

web/protected/views/inout/edit_photo.php, line 7 (Often Misused: File Upload) — Medium

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Content)

Sink Details

File: web/protected/views/inout/edit_photo.php:7
Taint Flags:

```
4  <tr>
5  <td>  ¬      </td>
6  <td>
7  <input type="file" name="in_photo"></td>
8  </tr>
9  <tr>
10  <td>  ¬      </td>
```

web/protected/views/inout/edit_photo.php, line 12 (Often Misused: File Upload) — Medium

Issue Details

Often Misused: File Upload — Medium

Package: web.protected.views.inout

web/protected/views/inout/edit_photo.php, line 12 (Often Misused: File Upload) — Medium

Kingdom: API Abuse
Scan Engine: SCA (Content)

Sink Details

File: web/protected/views/inout/edit_photo.php:12
Taint Flags:

```
9  <tr>
10  <td>  ¬      </td>
11  <td>
12  <input type="file" name="out_photo"></td>
13  </tr>
14  <tr>
15  <td> </td>
```

Package: web.protected.views.material

web/protected/views/material/upload_file.php, line 38 (Often Misused: File Upload) — Medium

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Content)

Sink Details

File: web/protected/views/material/upload_file.php:38
Taint Flags:

```
35  endforeach;?>
36  <tr>
37  <form id="form" name="form" class="" action="/material/uploadFile" method="post"
enctype="multipart/form-data">
38  <td><label>       </label><input type="file" name="file" />
39  <input type="hidden" name="formID" value="<?=$formID?>" /></td>
40  <td><button type="submit" class="grid_button"> £  </button></td>
41  </form>
```

web/protected/views/material/import.php, line 45 (Often Misused: File Upload) — Medium

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Content)

Sink Details

File: web/protected/views/material/import.php:45
Taint Flags:

Often Misused: File Upload — Medium

Package: web.protected.views.material

web/protected/views/material/import.php, line 45 (Often Misused: File Upload) — Medium

```
42  <tr>
43  <td align="right">    → </td>
44  <td>
45  <input type="file" name="file" id="file" />
46  <br />
47  <a href="/res/material_import.xls">      ¥</a></td>
48  </tr>
```

Open Redirect (4 issues)

Abstract

如果允许未验证的输入控制重定向机制所使用的 URL，可能会有利于攻击者发动钓鱼攻击。

Explanation

通过重定向，Web 应用程序能够引导用户访问同一应用程序内的不同网页或访问外部站点。应用程序利用重定向来帮助进行站点导航，有时还跟踪用户退出站点的方式。当 Web 应用程序将客户端重定向到攻击者可以控制的任意 URL 时，就会发生 Open redirect 漏洞：

攻击者可能利用 Open redirect 漏洞诱骗用户访问某个可信赖站点的 URL，并将他们重定向到恶意站点。攻击者通过对 URL 进行编码，使最终用户很难注意到重定向的恶意目标，即使将这一目标作为 URL 参数传递给可信赖的站点时也会发生这种情况。因此，Open redirect 常被作为钓鱼手段的一种而滥用，攻击者通过这种方式来获取最终用户的敏感数据。

例 1：以下 PHP 代码会在用户单击链接时，指示用户浏览器打开从 `dest` 请求参数中解析的 URL。

如果受害者收到一封电子邮件，指示该用户打开 "http://trusted.example.com/ecommerce/redirect.php?dest=www.wilyhacker.com" 链接，用户有可能会打开该链接，因为他会认为这个链接将转到可信赖的站点。然而，一旦用户打开该链接，上面的代码会将浏览器重定向至 "http://www.wilyhacker.com"。

很多用户都被告知，要始终监视通过电子邮件收到的 URL，以确保链接指向一个他们所熟知的可信赖站点。尽管如此，如果攻击者对目标 URL 进行 16 进制编码：
"http://trusted.example.com/ecommerce/redirect.php?dest=%77%69%6C%79%68%61%63%6B%65%72%2E%63%6F%6D"

那么，即使再聪明的最终用户也可能会被欺骗，打开该链接。

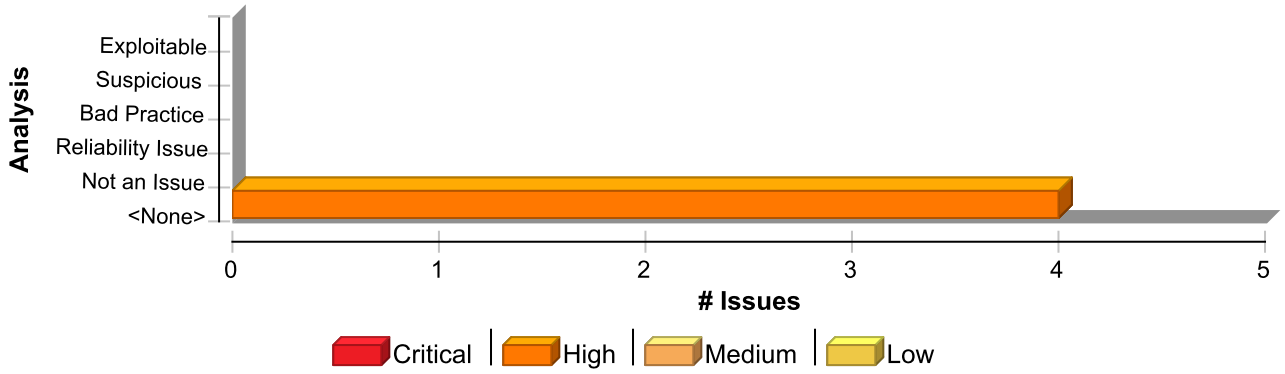
Recommendation

不应当允许未验证的用户输入控制重定向机制中的目标 URL。而应采用间接方法：创建一份合法 URL 列表，用户可以指定其中的内容并且只能从中进行选择。利用这种方法，就绝不会直接使用用户提供的输入来指定要重定向到的 URL。

例 2：以下代码引用了一个通过有效 URL 传播的数组。用户单击的链接将通过与所需 URL 对应的数组索引来传递。

但在某些情况下，这种方法并不可行，因为这样一份合法 URL 列表过于庞大、难以跟踪。这种情况下，有一种类似的方法也能限制用于重定向用户的域，这种方法至少可以防止攻击者向用户发送恶意的外部站点。

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Open Redirect	4	0	0	4
Total	**4**	**0**	**0**	**4**

Open Redirect — High

Package: framework

framework/yiilite.php, line 2622 (Open Redirect) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_SERVER['HTTP_HOST']
From: chttprequest.gethostinfo
File: framework/web/CHttpRequest.php:314

```
311  else
312  $http='http';
313  if(isset($_SERVER['HTTP_HOST']))
314  $this->_hostInfo=$http.'://'.$_SERVER['HTTP_HOST'];
315  else
316  {
317  $this->_hostInfo=$http.'://'.$_SERVER['SERVER_NAME'];
```

Sink Details

Sink: header()
Enclosing Method: redirect()
File: framework/yiilite.php:2622
Taint Flags: WEB, XSS

```
2619  {
2620  if(strpos($url,'/')===0 && strpos($url,'//')!==0)
2621  $url=$this->getHostInfo().$url;
2622  header('Location: '.$url, true, $statusCode);
2623  if($terminate)
```

Open Redirect — High

Package: framework

framework/yiilite.php, line 2622 (Open Redirect) — High

```
2624  Yii::app()->end();
2625  }
```

framework/yiilite.php, line 2622 (Open Redirect) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_SERVER['HTTP_HOST']
From: chttprequest.gethostinfo
File: framework/yiilite.php:2369

```
2366  else
2367  $http='http';
2368  if(isset($_SERVER['HTTP_HOST']))
2369  $this->_hostInfo=$http.'://'.$_SERVER['HTTP_HOST'];
2370  else
2371  {
2372  $this->_hostInfo=$http.'://'.$_SERVER['SERVER_NAME'];
```

Sink Details

Sink: header()
Enclosing Method: redirect()
File: framework/yiilite.php:2622
Taint Flags: WEB, XSS

```
2619  {
2620  if(strpos($url,'/')===0 && strpos($url,'//')!==0)
2621  $url=$this->getHostInfo().$url;
2622  header('Location: '.$url, true, $statusCode);
2623  if($terminate)
2624  Yii::app()->end();
2625  }
```

Package: framework.web

framework/web/CHttpRequest.php, line 803 (Open Redirect) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Open Redirect — High

Package: framework.web

framework/web/CHttpRequest.php, line 803 (Open Redirect) — High

Source: Read $_SERVER['HTTP_HOST']
From: chttprequest.gethostinfo
File: framework/web/CHttpRequest.php:314

```
311  else
312  $http='http';
313  if(isset($_SERVER['HTTP_HOST']))
314  $this->_hostInfo=$http.'://'.$_SERVER['HTTP_HOST'];
315  else
316  {
317  $this->_hostInfo=$http.'://'.$_SERVER['SERVER_NAME'];
```

Sink Details

Sink: header()
Enclosing Method: redirect()
File: framework/web/CHttpRequest.php:803
Taint Flags: WEB, XSS

```
800  {
801  if(strpos($url,'/')===0 && strpos($url,'//')!==0)
802  $url=$this->getHostInfo().$url;
803  header('Location: '.$url, true, $statusCode);
804  if($terminate)
805  Yii::app()->end();
806  }
```

framework/web/CHttpRequest.php, line 803 (Open Redirect) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_SERVER['HTTP_HOST']
From: chttprequest.gethostinfo
File: framework/yiilite.php:2369

```
2366  else
2367  $http='http';
2368  if(isset($_SERVER['HTTP_HOST']))
2369  $this->_hostInfo=$http.'://'.$_SERVER['HTTP_HOST'];
2370  else
2371  {
2372  $this->_hostInfo=$http.'://'.$_SERVER['SERVER_NAME'];
```

Open Redirect — High

Package: framework.web

framework/web/CHttpRequest.php, line 803 (Open Redirect) — High

Sink Details

Sink: header()
Enclosing Method: redirect()
File: framework/web/CHttpRequest.php:803
Taint Flags: WEB, XSS

```
800  {
801  if(strpos($url,'/')===0 && strpos($url,'//')!==0)
802  $url=$this->getHostInfo().$url;
803  header('Location: '.$url, true, $statusCode);
804  if($terminate)
805  Yii::app()->end();
806  }
```

Password Management: Hardcoded Password (3 issues)

Abstract

Hardcoded password 可能会削弱系统安全性，并且无法轻易修正出现的安全问题。

Explanation

使用硬编码方式处理密码绝非好方法。这不仅是因为所有项目开发人员都可以使用通过硬编码方式处理的密码，而且还会使解决这一问题变得极其困难。一旦代码投入使用，除非对软件进行修补，否则您再也不能改变密码了。如果帐户中的密码保护减弱，系统所有者将被迫在安全性和可行性之间做出选择。

示例：以下代码用 hardcoded password 来连接数据库：

```
...
$link = mysql_connect($url, 'scott', 'tiger');
if (!$link) {
    die('Could not connect: ' . mysql_error());
}
...
```

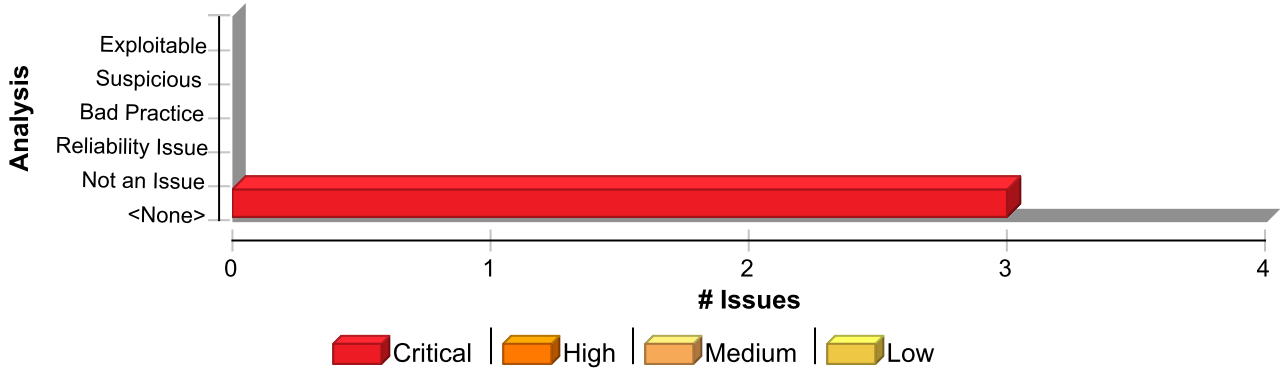
该代码可以正常运行，但是任何有该代码权限的人都能得到这个密码。一旦程序发布，将无法变更数据库用户 "scott" 和密码 "tiger"，除非是要修补该程序。心怀不轨的雇员可以利用手中掌握的信息访问权限入侵系统。

Recommendation

绝不能对密码进行硬编码。通常情况下，应对密码加以模糊化，并在外部资源文件中进行管理。在系统中采用明文的形式存储密码，会造成任何有充分权限的人读取和无意中误用密码。

有些第三方产品宣称可以采用更加安全的方式管理密码。较为安全的解决方法来是采用由用户创建的所有者机制，而这似乎也是如今唯一可行的方法。

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Password Management: Hardcoded Password	3	0	0	3
Total	**3**	**0**	**0**	**3**

Password Management: Hardcoded Password — Critical

Package: web.protected.config

web/protected/config/main.php, line 39 (Password Management: Hardcoded Password) — Critical

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: ArrayAccess
File: web/protected/config/main.php:39
Taint Flags:

```
36  //      ₩           Gii    l
37  'gii'=>array(
38  'class'=>'system.gii.GiiModule',
39  'password'=>'123',
40  //      ◆  ₩    ○ Gii
41  'ipFilters'=>array('127.0.0.1','::1'),
42  ),
```

web/protected/config/main.php, line 78 (Password Management: Hardcoded Password) — Critical

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: ArrayAccess
File: web/protected/config/main.php:78
Taint Flags:

```
75  //
76  'username' => 'root',
77  //  →
78  'password' => 'abc112233',
79  // ¢ →
80  'charset' => 'utf8',
81  //                    SQL   ¦       false                ¯¯       |  ¬       true
```

Password Management: Hardcoded Password — Critical

Package: web.protected.vendor.phpexcel.PHPExcel.Shared

web/protected/vendor/phpexcel/PHPExcel/Shared/PasswordHasher.php, line 49 (Password Management: Hardcoded Password) — Critical

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: $password
Enclosing Method: hashpassword()
File: web/protected/vendor/phpexcel/PHPExcel/Shared/PasswordHasher.php:49
Taint Flags:

```
46  * @return string Hashed password
47  */
48  public static function hashPassword($pPassword = '') {
49  $password = 0x0000;
50  $charPos = 1; // char position
51
52  // split the plain text password in its component characters
```

Path Manipulation (2 issues)

Abstract

通过用户输入控制 file system 操作所用的路径，借此攻击者可以访问或修改其他受保护的系统资源。

Explanation

当满足以下两个条件时，就会产生 path manipulation 错误：

1. 攻击者能够指定某一 file system 操作中所使用的路径。

2. 攻击者可以通过指定特定资源来获取某种权限，而这种权限在一般情况下是不可能获得的。

例如，在某一程序中，攻击者可以获得特定的权限，以重写指定的文件或是在其控制的配置环境下运行程序。

例 1： 下面的代码使用来自于 HTTP 请求的输入来创建一个文件名。程序员没有考虑到攻击者可能使用像 "`../../tomcat/conf/server.xml`" 一样的文件名，从而导致应用程序删除它自己的配置文件。

```
$rName = $_GET['reportName'];
$rFile = fopen("/usr/local/apfr/reports/" . rName,"a+");
...
unlink($rFile);
```

例 2： 下面的代码使用来自于配置文件的输入来决定打开哪个文件，并返回给用户。如果程序在一定的权限下运行，且恶意用户能够篡改配置文件，那么他们可以通过程序读取系统中以 `.txt` 扩展名结尾的所有文件。

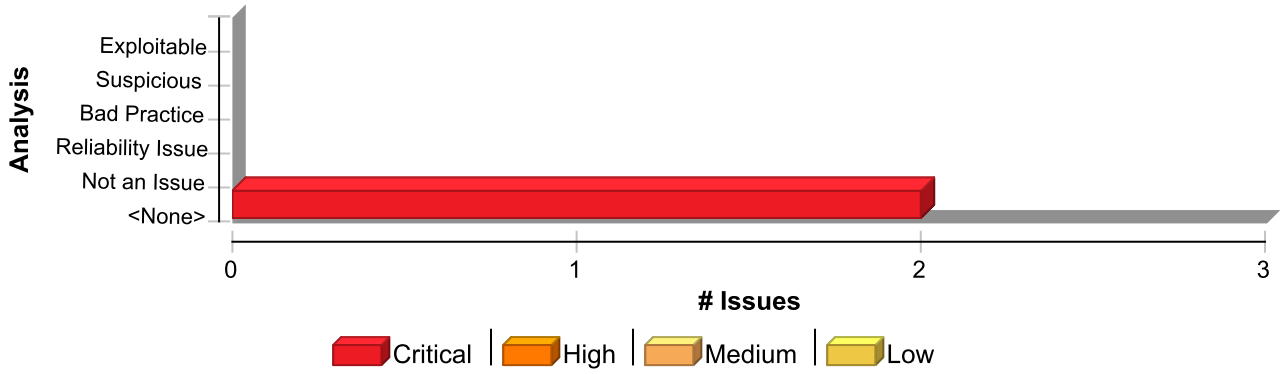
```
...
$filename = $CONFIG_TXT['sub'] . ".txt";
$handle = fopen($filename,"r");
$amt = fread($handle, filesize($filename));
echo $amt;
...
```

Recommendation

防止 path manipulation 的最佳方法是采用一些间接手段：例如创建一份合法资源名的列表，并且规定用户只能选择其中的文件名。通过这种方法，用户就不能直接由自己来指定资源的名称了。

但在某些情况下，这种方法并不可行，因为这样一份合法资源名的列表过于庞大、难以跟踪。因此，程序员通常在这种情况下采用黑名单的办法。在输入之前，黑名单会有选择地拒绝或避免潜在的危险字符。但是，任何这样一份黑名单都不可能是完整的，而且将随着时间的推移而过时。更好的方法是创建一份白名单，允许其中的字符出现在资源名称中，且只接受完全由这些被认可的字符组成的输入。

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Path Manipulation	2	0	0	2
Total	**2**	**0**	**0**	**2**

Path Manipulation — Critical

Package: web.protected.vendor.http

web/protected/vendor/http/WUpload.php, line 253 (Path Manipulation) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_POST['formID']
From: usematerialcontroller.actionuploadfile
File: web/protected/controllers/UseMaterialController.php:896

```
893  if ($_FILES['file']['name'] == "") {
894  exit("→          ");
895  }
896  $formID = $_POST['formID'];
897  if($_POST['type']=='receive'){
898  $rs = ReceiveForm::model()->findByPk($formID);
899  }else{
```

Sink Details

Sink: move_uploaded_file()
Enclosing Method: save()
File: web/protected/vendor/http/WUpload.php:253
Taint Flags: WEB, XSS

```
250  $this->fileName=$this->fileShortName.".".$this->fileExtName;
251  $this->fileSaveFullName=$this->savePath.$this->fileName;
252  //  ₩
253  $result=move_uploaded_file($this->sourceData['tmp_name'], $this-
>getFileSaveFullName("gbk"));
```

Path Manipulation — Critical

Package: web.protected.vendor.http

web/protected/vendor/http/WUpload.php, line 253 (Path Manipulation) — Critical

```
254  if (!$result) {
255  throw new Exception(self::$infoList['save_fail']);
256  }
```

web/protected/vendor/http/WUpload.php, line 253 (Path Manipulation) — Critical

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_POST['formID']
From: materialcontroller.actionuploadfile
File: web/protected/controllers/MaterialController.php:762

```
759  if ($_FILES['file']['name'] == "") {
760  exit(" →          ");
761  }
762  $formID = $_POST['formID'];
763  $rs = MoveForm::model()->findByPk($formID);
764  $path = "upload/move_from_pic/";
765  if (!file_exists($path)) {
```

Sink Details

Sink: move_uploaded_file()
Enclosing Method: save()
File: web/protected/vendor/http/WUpload.php:253
Taint Flags: WEB, XSS

```
250  $this->fileName=$this->fileShortName.".".$this->fileExtName;
251  $this->fileSaveFullName=$this->savePath.$this->fileName;
252  //  ₩
253  $result=move_uploaded_file($this->sourceData['tmp_name'], $this-
>getFileSaveFullName("gbk"));
254  if (!$result) {
255  throw new Exception(self::$infoList['save_fail']);
256  }
```

Possible Variable Overwrite: Global Scope (13 issues)

Abstract

此程序会调用可覆盖全局变量的函数，这可能会为攻击者打开方便之门。

Explanation

对于可覆盖已初始化的全局变量的函数，攻击者能够通过它影响依赖被覆盖变量的代码的执行情况。
可覆盖全局变量。

例 1：如果攻击者为下面这段 PHP 代码中的 `str` 提供恶意值，则对 `mb_parse_str()` 的调用可能会覆盖所有任意值，包括 `first`。在这种情况下，如果包含 JavaScript 的恶意值覆盖 `first`，则该程序会很容易受到 cross-site scripting 攻击。

Recommendation

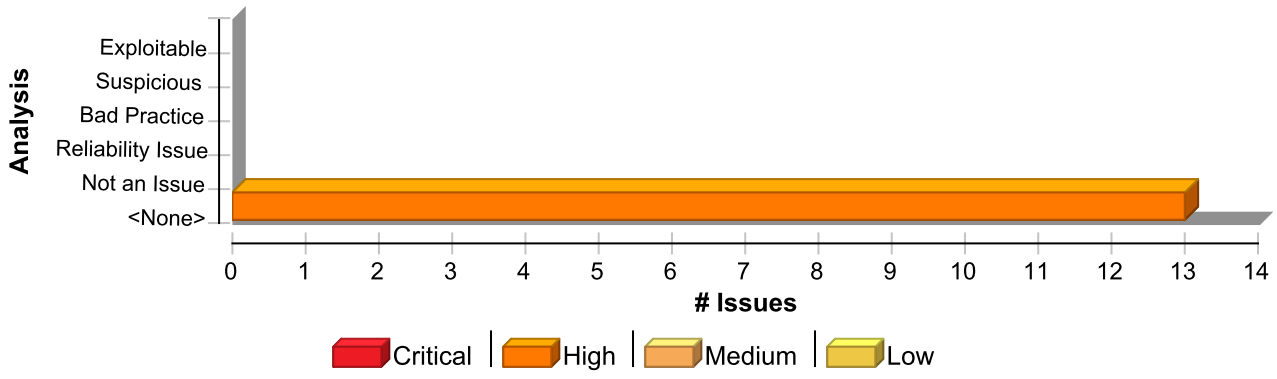
对于能够覆盖全局变量的函数，可通过下列方式避免它们覆盖全局变量：

- 调用 `mb_parse_str(string $encoded_string [, array &$result ])`（具有第二个参数），这样可以捕获操作结果以及防止函数覆盖全局变量。

- 调用 `extract(array $var_array [, int $extract_type [, string $prefix]])`（其中第二个参数设为 EXTR_SKIP），这样可以防止函数覆盖已定义的全局变量。

例 2：下列代码将第二个参数用于 `mb_parse_str()`，以杜绝例 1 中的漏洞。

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Possible Variable Overwrite: Global Scope	13	0	0	13
Total	**13**	**0**	**0**	**13**

Possible Variable Overwrite: Global Scope — High

Package: framework

framework/yiilite.php, line 4534 (Possible Variable Overwrite: Global Scope) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: extract
Enclosing Method: setcookieparams()
File: framework/yiilite.php:4534
Taint Flags:

```
4531  public function setCookieParams($value)
4532  {
4533  $data=session_get_cookie_params();
4534  extract($data);
4535  extract($value);
4536  if(isset($httponly))
4537  session_set_cookie_params($lifetime,$path,$domain,$secure,$httponly);
```

framework/yiilite.php, line 4535 (Possible Variable Overwrite: Global Scope) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: extract
Enclosing Method: setcookieparams()
File: framework/yiilite.php:4535
Taint Flags:

```
4532  {
4533  $data=session_get_cookie_params();
4534  extract($data);
4535  extract($value);
4536  if(isset($httponly))
4537  session_set_cookie_params($lifetime,$path,$domain,$secure,$httponly);
4538  else
```

framework/yiilite.php, line 3412 (Possible Variable Overwrite: Global Scope) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_POST['?']

Possible Variable Overwrite: Global Scope — High

Package: framework

framework/yiilite.php, line 3412 (Possible Variable Overwrite: Global Scope) — High

From: ccodegenerator.prepare
File: framework/gii/CCodeGenerator.php:157

```
154  $model->loadStickyAttributes();
155  if(isset($_POST[$modelClass]))
156  {
157  $model->attributes=$_POST[$modelClass];
158  $model->status=CCodeModel::STATUS_PREVIEW;
159  if($model->validate())
160  {
```

Sink Details

Sink: extract()
Enclosing Method: renderinternal()
File: framework/yiilite.php:3412
Taint Flags: WEB, XSS

```
3409  {
3410  // we use special variable names here to avoid conflict when extracting data
3411  if(is_array($_data_))
3412  extract($_data_,EXTR_PREFIX_SAME,'data');
3413  else
3414  $data=$_data_;
3415  if($_return_)
```

framework/yiilite.php, line 856 (Possible Variable Overwrite: Global Scope) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: extract
Enclosing Method: evaluateexpression()
File: framework/yiilite.php:856
Taint Flags:

```
853  {
854  if(is_string($_expression_))
855  {
856  extract($_data_);
857  return eval('return '.$_expression_.';');
858  }
859  else
```

Possible Variable Overwrite: Global Scope — High

Package: framework

framework/yiilite.php, line 3412 (Possible Variable Overwrite: Global Scope) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_POST['answers']
From: ccodegenerator.actionindex
File: framework/gii/CCodeGenerator.php:66

```
63  $model=$this->prepare();
64  if($model->files!=array() && isset($_POST['generate'], $_POST['answers']))
65  {
66  $model->answers=$_POST['answers'];
67  $model->status=$model->save() ? CCodeModel::STATUS_SUCCESS :
CCodeModel::STATUS_ERROR;
68  }
69
```

Sink Details

Sink: extract()
Enclosing Method: renderinternal()
File: framework/yiilite.php:3412
Taint Flags: WEB, XSS

```
3409  {
3410  // we use special variable names here to avoid conflict when extracting data
3411  if(is_array($_data_))
3412  extract($_data_,EXTR_PREFIX_SAME,'data');
3413  else
3414  $data=$_data_;
3415  if($_return_)
```

Package: framework.base

framework/base/CComponent.php, line 611 (Possible Variable Overwrite: Global Scope) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: extract
Enclosing Method: evaluateexpression()
File: framework/base/CComponent.php:611
Taint Flags:

Possible Variable Overwrite: Global Scope — High

Package: framework.base

framework/base/CComponent.php, line 611 (Possible Variable Overwrite: Global Scope) — High

```
608  {
609  if(is_string($_expression_))
610  {
611  extract($_data_);
612  return eval('return '.$_expression_.';');
613  }
614  else
```

Package: framework.cli.commands

framework/cli/commands/MessageCommand.php, line 81 (Possible Variable Overwrite: Global Scope) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: extract
Enclosing Method: run()
File: framework/cli/commands/MessageCommand.php:81
Taint Flags:

```
78
79  $config=require($args[0]);
80  $translator='Yii::t';
81  extract($config);
82
83  if(!isset($sourcePath,$messagePath,$languages))
84  $this->usageError('The configuration file must specify "sourcePath", "messagePath" and
"languages".');
```

Package: framework.utils

framework/utils/CFileHelper.php, line 59 (Possible Variable Overwrite: Global Scope) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: extract
Enclosing Method: copydirectory()
File: framework/utils/CFileHelper.php:59
Taint Flags:

Possible Variable Overwrite: Global Scope — High

Package: framework.utils

framework/utils/CFileHelper.php, line 59 (Possible Variable Overwrite: Global Scope) — High

```
56  $fileTypes=array();
57  $exclude=array();
58  $level=-1;
59  extract($options);
60  if(!is_dir($dst))
61  self::mkdir($dst,$options,true);
62
```

framework/utils/CFileHelper.php, line 111 (Possible Variable Overwrite: Global Scope) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: extract
Enclosing Method: findfiles()
File: framework/utils/CFileHelper.php:111
Taint Flags:

```
108  $fileTypes=array();
109  $exclude=array();
110  $level=-1;
111  extract($options);
112  $list=self::findFilesRecursive($dir,'',$fileTypes,$exclude,$level);
113  sort($list);
114  return $list;
```

Package: framework.web

framework/web/CHttpSession.php, line 241 (Possible Variable Overwrite: Global Scope) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: extract
Enclosing Method: setcookieparams()
File: framework/web/CHttpSession.php:241
Taint Flags:

```
238  public function setCookieParams($value)
239  {
240  $data=session_get_cookie_params();
```

Possible Variable Overwrite: Global Scope — High

Package: framework.web

framework/web/CHttpSession.php, line 241 (Possible Variable Overwrite: Global Scope) — High

```
241  extract($data);
242  extract($value);
243  if(isset($httponly))
244  session_set_cookie_params($lifetime,$path,$domain,$secure,$httponly);
```

framework/web/CHttpSession.php, line 242 (Possible Variable Overwrite: Global Scope) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: extract
Enclosing Method: setcookieparams()
File: framework/web/CHttpSession.php:242
Taint Flags:

```
239  {
240  $data=session_get_cookie_params();
241  extract($data);
242  extract($value);
243  if(isset($httponly))
244  session_set_cookie_params($lifetime,$path,$domain,$secure,$httponly);
245  else
```

framework/web/CBaseController.php, line 119 (Possible Variable Overwrite: Global Scope) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_POST['?']
From: ccodegenerator.prepare
File: framework/gii/CCodeGenerator.php:157

```
154  $model->loadStickyAttributes();
155  if(isset($_POST[$modelClass]))
156  {
157  $model->attributes=$_POST[$modelClass];
158  $model->status=CCodeModel::STATUS_PREVIEW;
159  if($model->validate())
160  {
```

Possible Variable Overwrite: Global Scope — High

Package: framework.web

framework/web/CBaseController.php, line 119 (Possible Variable Overwrite: Global Scope) — High

Sink Details

Sink: extract()
Enclosing Method: renderinternal()
File: framework/web/CBaseController.php:119
Taint Flags: WEB, XSS

```
116  {
117  // we use special variable names here to avoid conflict when extracting data
118  if(is_array($_data_))
119  extract($_data_,EXTR_PREFIX_SAME,'data');
120  else
121  $data=$_data_;
122  if($_return_)
```

framework/web/CBaseController.php, line 119 (Possible Variable Overwrite: Global Scope) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read $_POST['answers']
From: ccodegenerator.actionindex
File: framework/gii/CCodeGenerator.php:66

```
63  $model=$this->prepare();
64  if($model->files!=array() && isset($_POST['generate'], $_POST['answers']))
65  {
66  $model->answers=$_POST['answers'];
67  $model->status=$model->save() ? CCodeModel::STATUS_SUCCESS :
CCodeModel::STATUS_ERROR;
68  }
69
```

Sink Details

Sink: extract()
Enclosing Method: renderinternal()
File: framework/web/CBaseController.php:119
Taint Flags: WEB, XSS

```
116  {
117  // we use special variable names here to avoid conflict when extracting data
118  if(is_array($_data_))
```

Possible Variable Overwrite: Global Scope	High
Package: framework.web	
framework/web/CBaseController.php, line 119 (Possible Variable Overwrite: Global Scope)	**High**

```
119  extract($_data_,EXTR_PREFIX_SAME,'data');
120  else
121  $data=$_data_;
122  if($_return_)
```

Privacy Violation: Autocomplete (3 issues)

Abstract

借助表单的自动完成功能，某些浏览器可以在历史记录中保留敏感信息。

Explanation

启用自动完成功能后，某些浏览器会保留会话中的用户输入，以便随后使用该计算机的用户查看之前提交的信息。

Recommendation

对于表单或敏感输入，显式禁用自动完成功能。通过禁用自动完成功能，之前输入的信息不会在用户输入时以明文形式显示。这也会禁用大多数主要浏览器的“记住密码”功能。

例 1：在 HTML 表单中，通过在 `form` 标签上将 `autocomplete` 属性的值显式设置为 `off`，禁用所有输入字段的自动完成功能。

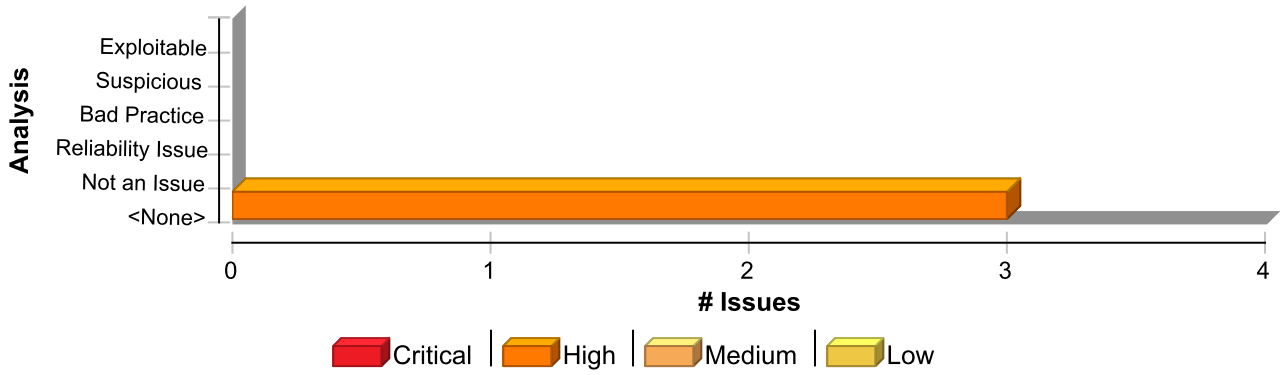
Address:
Password:

例 2：或者，通过在相应的标签上将 `autocomplete` 属性的值显式设置为 `off`，禁用特定输入字段的自动完成功能。

Address:
Password:

请注意，`autocomplete` 属性的默认值为 `on`。因此，处理敏感输入时请不要忽略该属性。

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Privacy Violation: Autocomplete	3	0	0	3
Total	**3**	**0**	**0**	**3**

Privacy Violation: Autocomplete — High

Package: web.protected.views.user

web/protected/views/user/user_password.php, line 33 (Privacy Violation: Autocomplete) — High

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Content)

Sink Details

File: web/protected/views/user/user_password.php:33
Taint Flags:

```
30  <tr>
31  <td align="right"><strong>    → </strong></td>
32  <td><label for="srcPwd"></label>
33  <input type="password" name="srcPwd" id="srcPwd"></td>
34  </tr>
35  <tr>
36  <td align="right"><strong>    → </strong></td>
```

web/protected/views/user/user_password.php, line 38 (Privacy Violation: Autocomplete) — High

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Content)

Sink Details

File: web/protected/views/user/user_password.php:38
Taint Flags:

```
35  <tr>
36  <td align="right"><strong>    → </strong></td>
37  <td><label for="newPwd1"></label>
38  <input type="password" name="newPwd1" id="newPwd1"></td>
39  </tr>
40  <tr>
41  <td align="right"><strong>      → </strong></td>
```

web/protected/views/user/user_password.php, line 43 (Privacy Violation: Autocomplete) — High

Issue Details

Privacy Violation: Autocomplete	High
Package: web.protected.views.user	
web/protected/views/user/user_password.php, line 43 (Privacy Violation: Autocomplete)	High

Kingdom: Security Features
Scan Engine: SCA (Content)

Sink Details

File: web/protected/views/user/user_password.php:43
Taint Flags:

```
40  <tr>
41  <td align="right"><strong>       → </strong></td>
42  <td><label for="newPwd2"></label>
43  <input type="password" name="newPwd2" id="newPwd2"></td>
44  </tr>
45  <tr>
46  <td> </td>
```

Weak Encryption (2 issues)

Abstract

识别调用会使用无法保证敏感数据的保密性的弱加密算法。

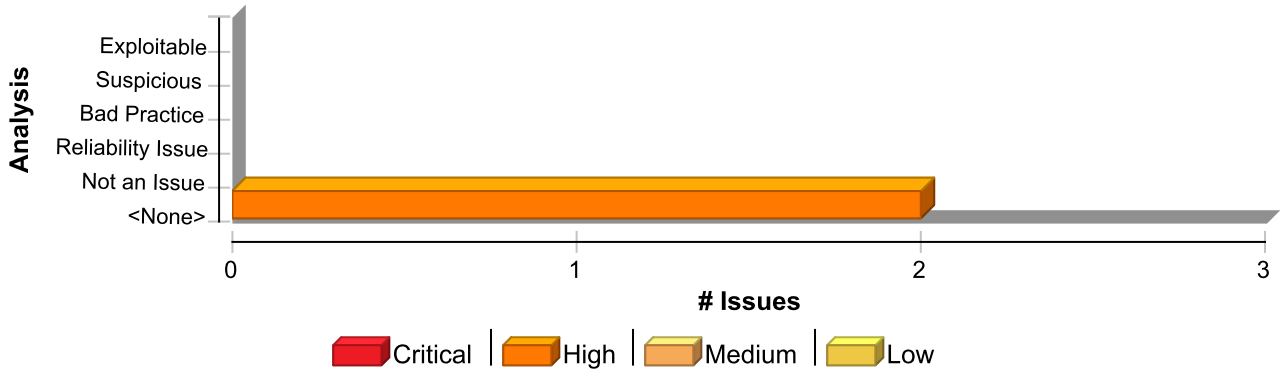
Explanation

陈旧的加密算法（如 DES）再也不能为敏感数据提供足够的保护了。加密算法依赖于密钥大小，这是确保加密强度的主要方法之一。加密强度通常通过生成有效密钥所需的时间和计算能力来衡量。计算能力的提高使得能够在合理的时间内获得较小的加密密钥。例如，在二十世纪七十年代首次开发出该算法时，在 DES 中使用的 56 位密钥造成了巨大的计算障碍，但今天，使用常用设备能在不到一天的时间内破解 DES。

Recommendation

使用密钥较大的强加密算法来保护敏感数据。作为 DES 的备选强加密算法的示例包括 Rijndael（高级加密标准，简称 AES）和 Triple DES (3DES)。在选择一种算法之前，应首先确定您的组织是否对某个特定算法和实施实现了标准化。

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Weak Encryption	2	0	0	2
Total	**2**	**0**	**0**	**2**

Weak Encryption — High

Package: framework.utils

framework/utils/CPasswordHelper.php, line 94 (Weak Encryption) — High

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Semantic)

Sink Details

Sink: crypt()
Enclosing Method: hashpassword()
File: framework/utils/CPasswordHelper.php:94

Weak Encryption — High

Package: framework.utils

framework/utils/CPasswordHelper.php, line 94 (Weak Encryption) — High

Taint Flags:

```
91  {
92  self::checkBlowfish();
93  $salt=self::generateSalt($cost);
94  $hash=crypt($password,$salt);
95
96  if(!is_string($hash) || (function_exists('mb_strlen') ? mb_strlen($hash, '8bit') :
strlen($hash))<32)
97  throw new CException(Yii::t('yii','Internal error while generating hash.'));
```

framework/utils/CPasswordHelper.php, line 120 (Weak Encryption) — High

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Semantic)

Sink Details

Sink: crypt()
Enclosing Method: verifypassword()
File: framework/utils/CPasswordHelper.php:120
Taint Flags:

```
117  $matches[1]<4 || $matches[1]>31)
118  return false;
119
120  $test=crypt($password,$hash);
121  if(!is_string($test) || strlen($test)<32)
122  return false;
123
```