WordPress插件挖洞系列--简单了解插件结构[一]

Poacher / 2018-12-15 10:15:00 / 浏览数 3302 技术文章 技术文章 顶(0) 踩(0)

[TOC]

1.关于此系列文章

WordPress插件漏洞挖掘系列文章流程大概为：简单了解插件结构-》尝试开发第一个插件-》插件漏洞的一次复现-》插件漏洞挖掘首尝试，在这系列文中对于有PHP基础

由于对WordPress以及插件运行流程所拥有知识点并没那么透彻，因此本文中可能会存在错误地方，还望各位师傅可以指出，感谢包容。

2.前言

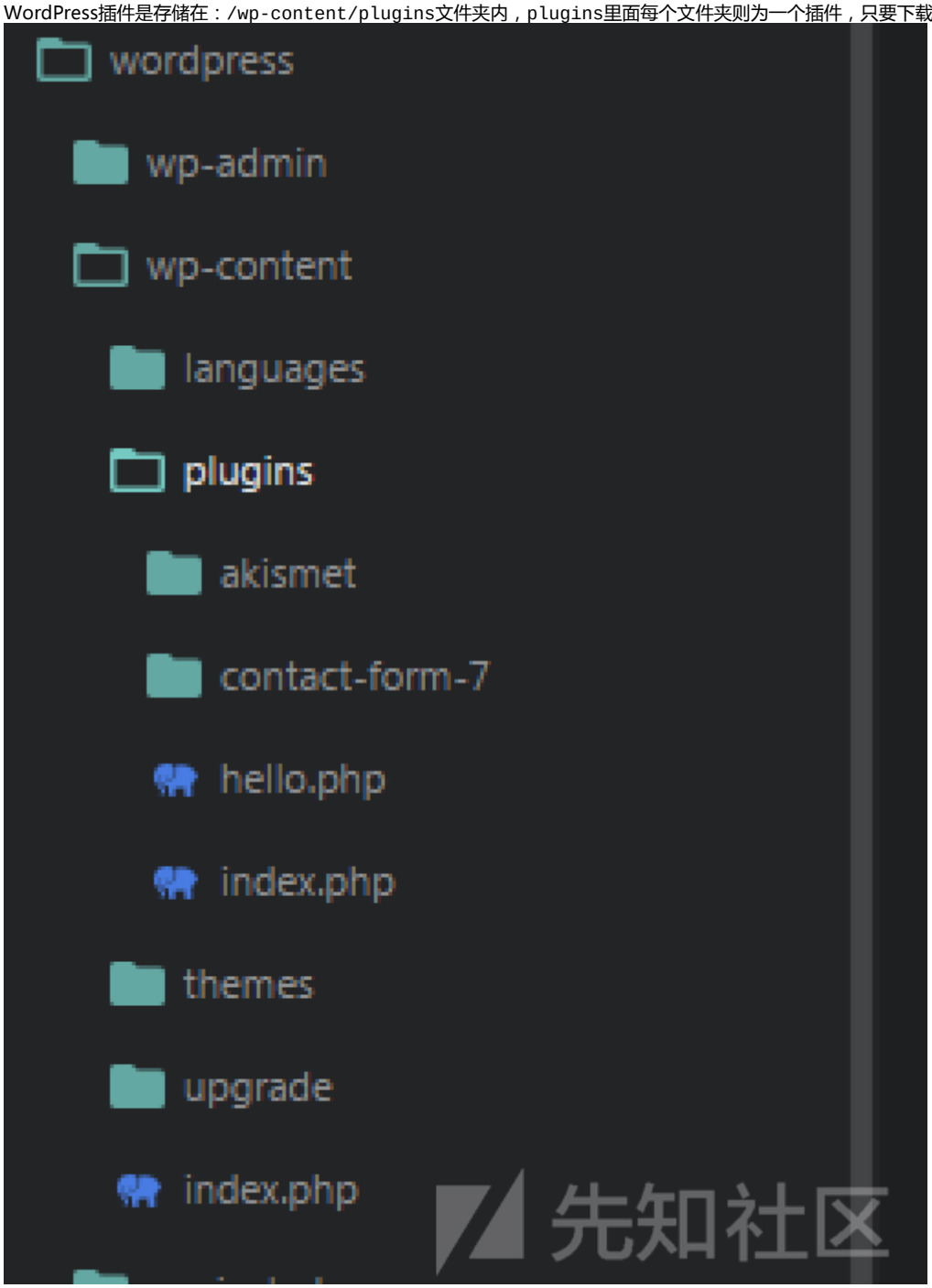
在挖掘一套程序的插件漏洞，首先得先了解下插件在这套程序中的一个流程是怎么样的，入口点、挂载点、系统函数等信息。

3.插件简单介绍

3.1.插件目录结构

正常来说无论是开发程序还是开发程序的插件，有开发经验的开发者都是会遵循开发所需的原则，比如：命名规范、以及目录结构格式等，这样一来也是更加可以清晰的看懂

WordPress插件是存储在：/wp-content/plugins文件夹内，plugins里面每个文件夹则为一个插件，只要下载下来的插件默认都会存储在这个位置：



比如创建一个插件名字为test-plugin，那么插件主文件就可以起为test-plugin.php，插件中可能会引用到一些：js、css、image资源文件或者其它PHP文件，那么
即可，正常来说开发者都会把插件文件夹名字起的与插件主文件名字相同，当然这个是因人而异，问题不大。

无论开发程序还是插件的时候都应该架构一个良好清晰的目录结构，
比如上述所说开发插件所需要的资源文件等，都可以分开存储，这样一来目录结构则更加简洁可观，如：

```
test-plugin
 |----include
 |      |----js
 |      |----css
 |      |----xxxx.php
 |----image
 |----wp-test-plugin.php
 |----uninstall.php
 |----settings.php
 |----readme.txt
```

在wordpress中绝大部分插件的目录结构基本上会有以上述所呈现的样子。只有可能资源文件存放位置不同，但是插件主文件(wp-test-plugin.php)、插件卸载文件(u

---

3.2.插件命名

在开发插件编写函数的时候在函数名或者插件名前加一个不容易重复的前缀是很有必要的，一套程序中插件有无数，因此起名也成为了头痛的事情，因为只要起的通用一点的

4.WordPress插件钩子(挂载点) * 重要

4.1.插件钩子介绍

钩子(hooks)在WordPress插件中起到重要的作用，主要做些什么呢？其实就是说在你插件流程运行到某个特定的地方的时候就会调用当前与钩子有关联的函数，也就是这
举个例子，比如我在某个帖子内提交一条评论留言，在原有评论功能里是没有评论有回复则邮件发送给评论者进行提示，那么这个时候就可以开发一个插件，并且运用到钩子

4.2.插件钩子类型

4.2.1.动作钩子

动作钩子就是我们上面介绍所说的类似，运行到某个指定地方，就会被执行的一种插件钩子。
动作钩子对应的使用方法为：add_action ('■■','■■■')该函数共有4个参数，最后两个都有默认值，正常情况不必填写，默认即可。

- 事件：可以为WordPress内置调用do_action()定义的一些动作事件，可自行定义，官方上有发布对应版本可用的钩子列表，这里也可以是主题或者插件内的一个动作
- 函数名：当钩子生效时调用的函数，可以是自己在插件内定义的函数，也可以是PHP或者WoredPress内置函数。

用法如下：

```
function lock_article($post_id)
{
   //■■■■■■■■■■■■■■■■■■■■■■■■
}
add_action('publish_post','lock_article');
```

这个时候，只要流程运行到了publish_post事件动作，那么就会触发lock_article函数

4.2.2.过滤器

这种类型只要用于修改发送出去或者保存的数据，可以通过过滤器在运行到某个地方的时候对传输进来的数据进行操作。
过滤器对应的使用方法为：add_filter ('■■','■■■')该函数共有4个参数，最后两个都有默认值，正常情况不必填写，默认即可。

事件：可以为官方提供的过滤器钩子，也就是经过apply_filters()进行定义的钩子，可自行定义，决定了何时执行过滤器。

函数名：当过滤器钩子生效的时候调用的函数，可以是一个PHP内置函数，一WordPress核心函数，或者是自己在插件文件中定义的函数。

用法如下：

```
add_filter( 'wp_title', 'test_filter_title');

function test_filter_title($title)
{
   //■■■■■■■■■■■■■■■
}
```

那么只要在流程中执行了传递以及处理数据，并且走到了wp_title这个过滤钩子的话，就会直接test_filter_title函数，对原有的数据进行处理。

6.2.3.钩子类型简单总结

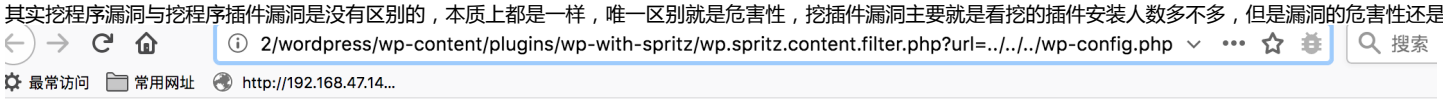
其实两种类型道理都是一样，运行到指定地方，做指定的事情，也就是实现了不更改程序核心代码从而修改功能流程。
注意：自己定义的函数以及钩子调用必须保存在同一个文件中。
add_action与add_filter的最后两个参数分别是：优先级(默认10)，以及钩子接收的参数数量(默认1)
如果你发现你的钩子与其它插件或者其它代码起冲突，这个时候就可以使用remove_action或者remove_filter进行移除钩子，也是差不多的用法，在特定的时候移除掉

5.总结

开发插件涉及到的内容并没有以上那么少，但是必须知道的都在上面已经讲到，有些东西得配合插件开发一起讲才能更加清晰的理解，所以在下一篇中，所分享的内容可能会

当你看到这里的时候，如果上文所描述的内容你觉得没压力，能消化的话，那么很高兴的告诉你，你已经可以成功开发一款简单的WordPress插件啦，那么在下一篇文中
当然，如果你看完之后感觉有点理不清也没关系，相信你看完下一篇插件开发实例之后在回来看这一篇文，相信你会看的更加透彻。

其实挖程序漏洞与挖程序插件漏洞是没有区别的，本质上都是一样，唯一区别就是危害性，挖插件漏洞主要就是看挖的插件安装人数多不多，但是漏洞的危害性还是一样的，




```php
<?php
/**
 * WordPress基础配置文件。
 *
 * 这个文件被安装程序用于自动生成wp-config.php配置文件，
 * 您可以不使用网站，您需要手动复制这个文件，
 * 并重命名为“wp-config.php”，然后填入相关信息。
 *
 * 本文件包含以下配置选项：
 *
 * * MySQL设置
 * * 密钥
 * * 数据库表名前缀
 * * ABSPATH
 *
 * @link https://codex.wordpress.org/zh-cn:%E7%BC%96%E8%BE%91_wp-config.php
 *
 * @package WordPress
 */

// ** MySQL 设置 - 具体信息来自您正在使用的主机 ** //
/** WordPress数据库的名称 */
define('DB_NAME', 'wordpress');

/** MySQL数据库用户名 */
define('DB_USER', 'root');

/** MySQL数据库密码 */
define('DB_PASSWORD', 'root');

/** MySQL主机 */
define('DB_HOST', 'localhost');

/** 创建数据表时默认的文字编码 */
define('DB_CHARSET', 'utf8mb4');
```


上一篇：PCB_final_shotsho... 下一篇：分析Pwn2Own上的一个Adob...

1. 2 条回复

流沙 2019-05-07 09:16:35

期待下一章节, 好久了, 没看到继续写!!

0 回复Ta

Poacher 2019-06-16 15:14:19

@流沙 工作过忙，已经写到一半了，还在一点点挤出时间在写。

0 回复Ta

登录 后跟帖

先知社区

现在登录

热门节点

技术文章

社区小黑板

目录

RSS 关于社区 友情链接 社区小黑板