Web安全学习思维导图，[web入门菜鸡萌新必备]

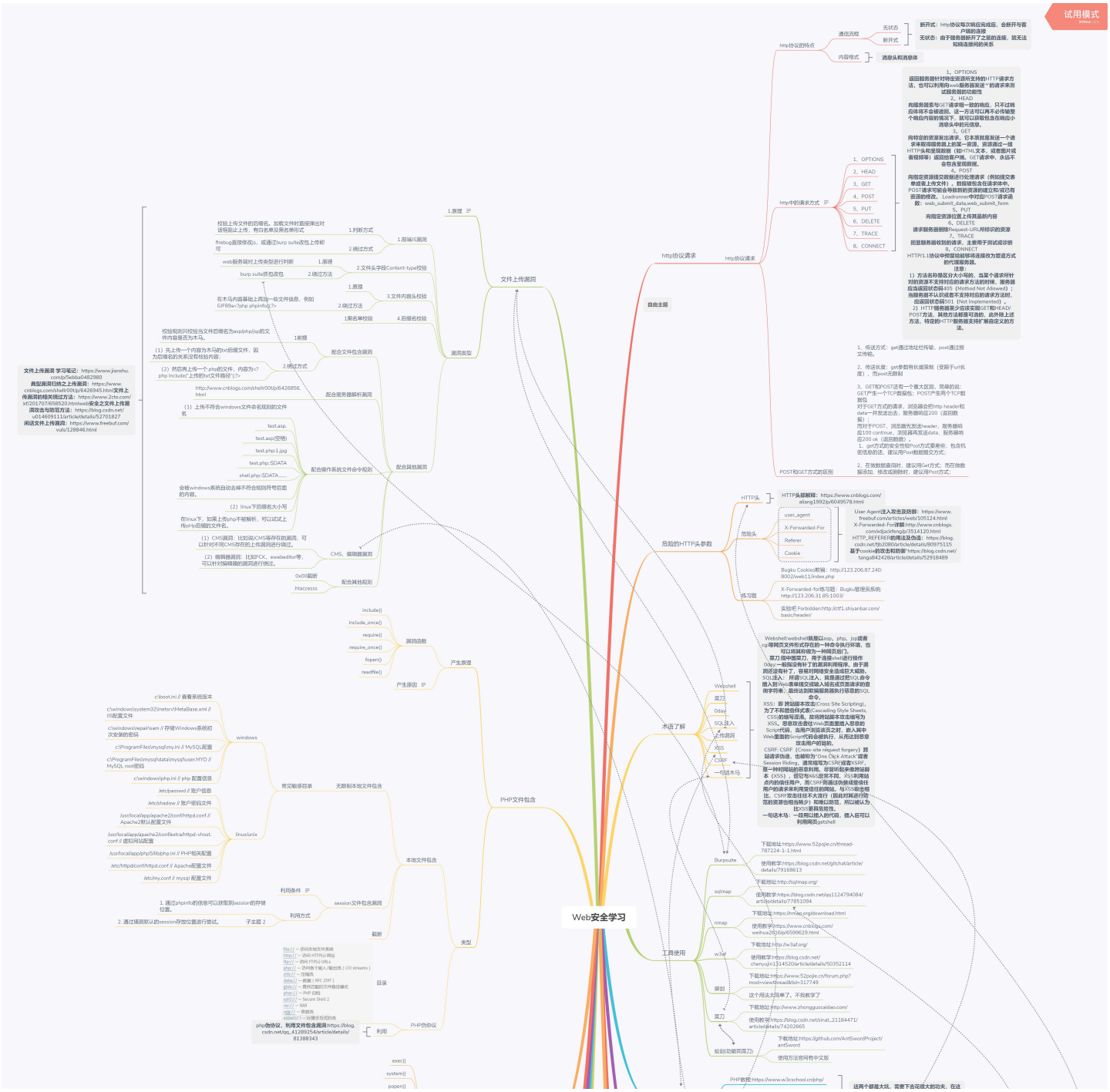
转载

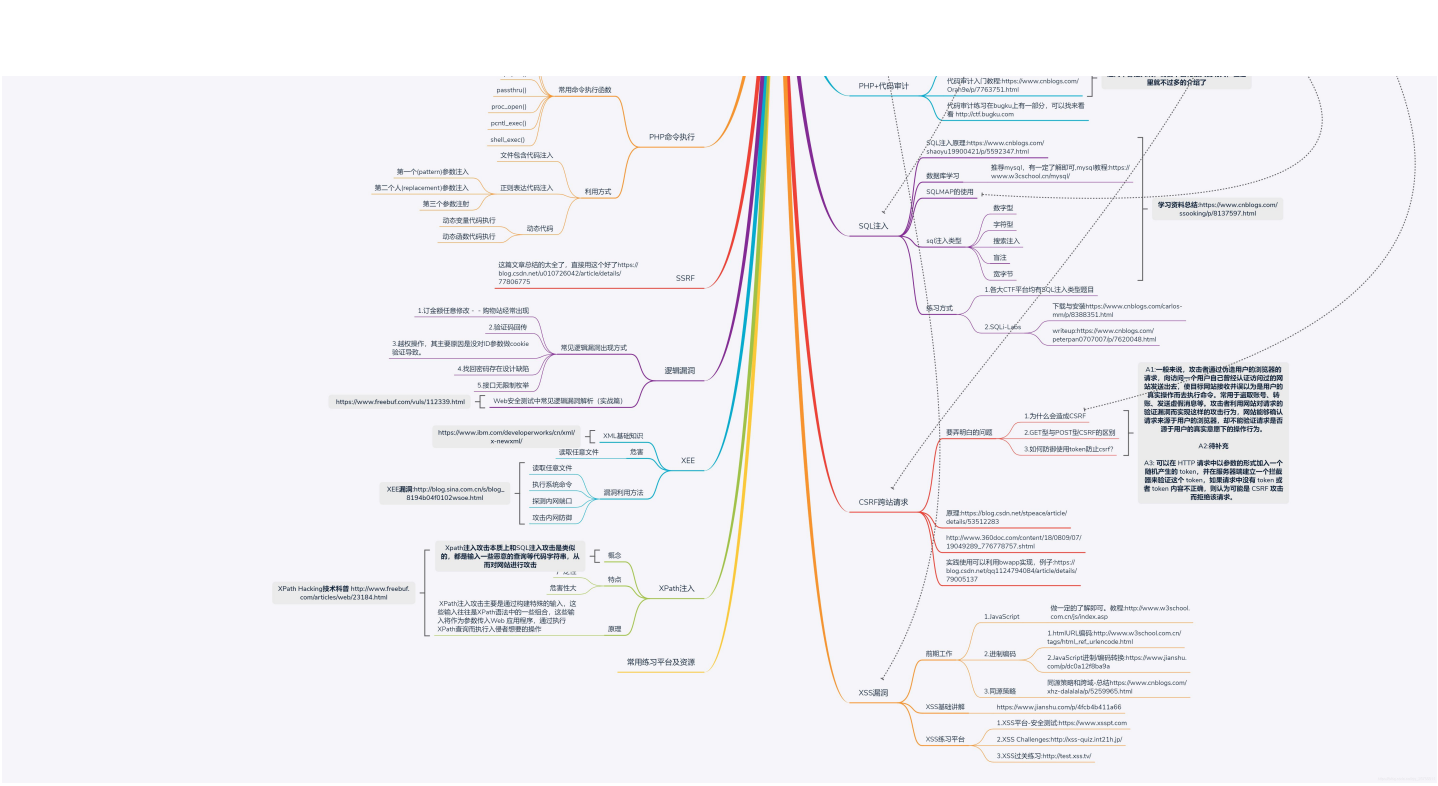
小 白 渣 于 2019-10-29 12:24:54 发布 629 收藏 8

文章标签： web安全

原文链接：https://blog.csdn.net/qq_25755011/article/details/102786732

写在前面，自制了一个web安全学习的思维导图，写的并不是很全面，自己本身其实也是个弟弟，总结了一些师傅们的资源，希望大家给予一定的建议和提升的空间，另外也给萌新一个思路吧~





> 下面是大纲，可以参考内容

Web安全学习

http协议请求

http协议请求

http协议的特点

通信流程

- 无状态
- 断开式

内容格式

http中的请求方式

1、OPTIONS
返回服务器针对特定资源所支持的HTTP请求方法，也可以利用向web服务器发送‘*’的请求来测试服务器的功能性
2、HEAD
向服务器索与GET请求相一致的响应，只不过响应体将不会被返回。这一方法可以再不必传输整个响应内容的情况下，就可以获取包含在响应小消息头中的元信息。
3、GET
向特定的资源发出请求。它本质就是发送一个请求来取得服务器上的某一资源。资源通过一组HTTP头和呈现数据（如HTML文本，或者图片或者视频等）返回给客户端。GET请求中，永远不会包含呈现数据。
4、POST
向指定资源提交数据进行处理请求（例如提交表单或者上传文件）。数据被包含在请求体中。POST请求可能会导致新的资源的建立和/或已有资源的修改。 Loadrunner中对应POST请求函数：web_submit_data,web_submit_form
5、PUT
向指定资源位置上传其最新内容
6、DELETE
请求服务器删除Request-URL所标识的资源
7、TRACE
回显服务器收到的请求，主要用于测试或诊断
8、CONNECT
HTTP/1.1协议中预留给能够将连接改为管道方式的代理服务器。
注意：
1）方法名称是区分大小写的，当某个请求所针对的资源不支持对应的请求方法的时候，服务器应当返回状态码405（Mothod Not Allowed）；当服务器不认识或者不支持对应的请求方法时，应返回状态码501（Not Implemented）。
2）HTTP服务器至少应该实现GET和HEAD/POST方法，其他方法都是可选的，此外除上述方法，特定的HTTP服务器支持扩展自定义的方法。

- 1、OPTIONS
- 2、HEAD
- 3、GET
- 4、POST
- 5、PUT
- 6、DELETE
- 7、TRACE
- 8、CONNECT

POST和GET方式的区别

- 1、传送方式：get通过地址栏传输，post通过报文传输。

2、传送长度：get参数有长度限制（受限于url长度），而post无限制

3、GET和POST还有一个重大区别，简单的说：
GET产生一个TCP数据包；POST产生两个TCP数据包
对于GET方式的请求，浏览器会把http header和data一并发送出去，服务器响应200（返回数据）；
而对于POST，浏览器先发送header，服务器响应100 continue，浏览器再发送data，服务器响应200 ok（返回数据）。
1、get方式的安全性较Post方式要差些，包含机密信息的话，建议用Post数据提交方式；

2、在做数据查询时，建议用Get方式；而在做数据添加、修改或删除时，建议用Post方式；

危险的HTTP头参数

HTTP头

危险头

- user_agent
- X-Forwarded-For
- Referer
- Cookie

练习题

- Bugku Cookies欺骗：http://123.206.87.240:8002/web11/index.php
- X-Forwarded-for练习题：Bugku管理员系统:http://123.206.31.85:1003/
- 实验吧 Forbidden:http://ctf1.shiyanbar.com/basic/header/

术语了解

Webshell

菜刀

0day

SQL注入

上传漏洞

XSS

CSRF

一句话木马

工具使用

Burpsuite

- 下载地址:https://www.52pojie.cn/thread-787224-1-1.html
- 使用教学:https://blog.csdn.net/gitchat/article/details/79168613

sqlmap

- 下载地址:http://sqlmap.org/
- 使用教学:https://blog.csdn.net/qq1124794084/article/details/77851094

nmap

- 下载地址:https://nmap.org/download.html
- 使用教学:https://www.cnblogs.com/weihua2616/p/6599629.html

w3af

- 下载地址:http://w3af.org/
- 使用教学:https://blog.csdn.net/chenyujin1314520/article/details/50352114

御剑

- 下载地址:https://www.52pojie.cn/forum.php?mod=viewthread&tid=317749
- 这个用法太简单了。不找教学了

菜刀

- 下载地址:http://www.zhongguocaidao.com/
- 使用教学:https://blog.csdn.net/sinat_21184471/article/details/74202665

蚁剑(功能同菜刀)

- 下载地址:https://github.com/AntSwordProject/antSword
- 使用方法官网有中文版

PHP+代码审计

PHP教程:https://www.w3cschool.cn/php/

代码审计入门教程:https://www.cnblogs.com/Oran9e/p/7763751.html

代码审计练习在bugku上有一部分，可以找来看看 http://ctf.bugku.com

SQL注入

SQL注入原理:https://www.cnblogs.com/shaoyu19900421/p/5592347.html

数据库学习

- 推荐mysql，有一定了解即可,mysql教程:https://www.w3cschool.cn/mysql/

SQLMAP的使用

sql注入类型

- 数字型
- 字符型
- 搜索注入
- 盲注
- 宽字节

练习方式

1.各大CTF平台均有SQL注入类型题目

2.SQLi-Labs

- 下载与安装https://www.cnblogs.com/carlos-mm/p/8388351.html
- writeup:https://www.cnblogs.com/peterpan0707007/p/7620048.html

CSRF跨站请求

要弄明白的问题

- 1.为什么会造成CSRF
- 2.GET型与POST型CSRF的区别
- 3.如何防御使用token防止csrf?

原理:https://blog.csdn.net/stpeace/article/details/53512283

http://www.360doc.com/content/18/0809/07/19049289_776778757.shtml

实践使用可以利用bwapp实现，例子:https://blog.csdn.net/qq1124794084/article/details/79005137

XSS漏洞

前期工作

1.JavaScript

- 做一定的了解即可。教程:http://www.w3school.com.cn/js/index.asp

2.进制编码

- 1.htmlURL编码:http://www.w3school.com.cn/tags/html_ref_urlencode.html
- 2.JavaScript进制/编码转换:https://www.jianshu.com/p/dc0a12f8ba9a

3.同源策略

- 同源策略和跨域-总结https://www.cnblogs.com/xhz-dalalala/p/5259965.html

XSS基础讲解

- https://www.jianshu.com/p/4fcb4b411a66

XSS练习平台

- 1.XSS平台-安全测试:https://www.xsspt.com
- 2.XSS Challenges:http://xss-quiz.int21h.jp/
- 3.XSS过关练习:http://test.xss.tv/

常用练习平台及资源

XPath注入

概念

特点

- 广泛性
- 危害性大

原理

- XPath注入攻击主要是通过构建特殊的输入，这些输入往往是XPath语法中的一些组合，这些输入将作为参数传入Web 应用程序，通过执行XPath查询而执行入侵者想要的操作

XEE

XML基础知识

危害

- 读取任意文件

漏洞利用方法

- 读取任意文件
- 执行系统命令
- 探测内网端口
- 攻击内网防御

逻辑漏洞

常见逻辑漏洞出现方式

- 1.订金额任意修改——购物站经常出现
- 2.验证码回传
- 3.越权操作，其主要原因是没对ID参数做cookie验证导致。
- 4.找回密码存在设计缺陷
- 5.接口无限制枚举

Web安全测试中常见逻辑漏洞解析（实战篇）

SSRF

这篇文章总结的太全了，直接用这个好了
https://blog.csdn.net/u010726042/article/details/77806775

PHP命令执行

常用命令执行函数

- exec()
- system()
- popen()
- passthru()
- proc_open()
- pcntl_exec()
- shell_exec()

利用方式

文件包含代码注入

正则表达代码注入

- 第一个(pattern)参数注入
- 第二个人(replacement)参数注入
- 第三个参数注射

动态代码

- 动态变量代码执行
- 动态函数代码执行

PHP文件包含

产生原理

漏洞函数

- include()
- include_once()
- require()
- require_once()
- fopen()
- readfile()

产生原因

文件包含函数加载的参数没有经过过滤或者严格的定义，可以被用户控制，包含其他恶意文件，导致了执行了非预期的代码。

类型

本地文件包含

无限制本地文件包含

常见敏感目录

windows

- c:\boot.ini // 查看系统版本
- c:\windows\system32\inetsrv\MetaBase.xml // IIS配置文件
- c:\windows\repair\sam // 存储Windows系统初次安装的密码
- c:\ProgramFiles\mysql\my.ini // MySQL配置
- c:\ProgramFiles\mysql\data\mysql\user.MYD // MySQL root密码
- c:\windows\php.ini // php 配置信息

linux/unix

- /etc/passwd // 账户信息
- /etc/shadow // 账户密码文件
- /usr/local/app/apache2/conf/httpd.conf // Apache2默认配置文件
- /usr/local/app/apache2/conf/extra/httpd-vhost.conf // 虚拟网站配置
- /usr/local/app/php5/lib/php.ini // PHP相关配置
- /etc/httpd/conf/httpd.conf // Apache配置文件
- /etc/my.conf // mysql 配置文件

session文件包含漏洞

利用条件

session的存储位置可以获取。

利用方式

1. 通过phpinfo的信息可以获取到session的存储位置。

子主题 2

2. 通过猜测默认的session存放位置进行尝试。

截断

PHP伪协议

- 目录
- 利用

文件上传漏洞

1.原理

文件上传漏洞是指由于程序员在对用户文件上传部分的控制不足或者处理缺陷，而导致的用户可以越过其本身权限向服务器上上传可执行的动态脚本文件。这里上传的文件可以是木马，病毒，恶意脚本或者WebShell等。这种攻击方式是最为直接和有效的，“文件上传”本身没有问题，有问题的是文件上传后，服务器怎么处理、解释文件。如果服务器的处理逻辑做的不够安全，则会导致严重的后果。

漏洞类型

1.前端JS漏洞

1.判断方式

- 校验上传文件的后缀名。加载文件时直接弹出对话框阻止上传，有白名单及黑名单形式

2.绕过方式

- firebug直接修改js，或通过burp suite改包上传即可

2.文件头字段Content-type校验

1.原理

- web服务端对上传类型进行判断

2.绕过方法

- burp suite抓包改包

3.文件内容头校验

1.原理

2.绕过方法

- 在木马内容基础上再加一些文件信息，例如GIF89a<?php phpinfo();?>

4.后缀名校验

- 1黑名单校验

配合其他漏洞

配合文件包含漏洞

1前提

- 校验规则只校验当文件后缀名为asp/php/jsp的文件内容是否为木马。

2.绕过方式

- （1）先上传一个内容为木马的txt后缀文件，因为后缀名的关系没有检验内容；
- （2）然后再上传一个.php的文件，内容为<?php Include(“上传的txt文件路径”);?>

.配合服务器解析漏洞

- http://www.cnblogs.com/shellr00t/p/6426856.html

配合操作系统文件命名规则

配合操作系统文件命令规则

- （1）上传不符合windows文件命名规则的文件名
- test.asp.
- test.asp(空格)
- test.php:1.jpg
- test.php::$DATA
- shell.php::$DATA…….
- 会被windows系统自动去掉不符合规则符号后面的内容。
- （2）linux下后缀名大小写
- 在linux下，如果上传php不被解析，可以试试上传pHp后缀的文件名。

CMS、编辑器漏洞

- （1）CMS漏洞：比如说JCMS等存在的漏洞，可以针对不同CMS存在的上传漏洞进行绕过。
- （2）编辑器漏洞：比如FCK，ewebeditor等，可以针对编辑器的漏洞进行绕过。

配合其他规则

- 0x00截断
- .htaccesss