视频资源

基础入门（P1-P5）

P1. 概念名词

1.1 域名

什么是域名?

- 域名：是由一串用点分隔的名字组成的Internet上某一台计算机或计算机组的名称，用于在数据传输时对计算机的定位标识（有时也指地理位置）。

什么是二级域名多级域名?

- 二级域名：分两种
- 在国际顶级域名下的二级域名
- 国际顶级域名下二级域名， 二级域名一般是指域名注册人选择使用的网上名称，如“yahoo.com”；上网的商业组织通常使用自己的商标、商号或其他商业标志作为自己的网上名称，如“microsoft.com”。
- 国家顶级域名下二级域名
- 在国家顶级域名之下二级域名一般是指类似于国际顶级域名的表示注册人类别和功能的标志。例如，在“．com.cn”域名结构中，“.com”此时是置于国家顶级域名“.cn”下的二级域名，表示商业性组织，以此类推。

域名发现对于安全测试意义

- 可以给安全测试提供更多的测试点，方便获取更多相关信息

1.2 DNS

什么是 DNS?

- 域名系统（**D**omain **N**ame **S**ystem）。它是一个域名和IP地址相互映射的一个分布式数据库，能够使人更方便地访问互联网。DNS使用UDP端口53。对于每一级域名长度的限制是63个字符，域名总长度则不能超过253个字符。

本地 HOSTS 与 DNS 的关系?

- Hosts在本地将一些常用的网址域名与其对应的IP地址建立一个关联“数据库”，当我们访问域名时，系统会首先自动从Hosts文件中寻找对应的IP地址，一旦找到，系统会立即打开对应网页，如果没有找到，则系统会再将网址提交DNS域名解析服务器进行IP地址的解析。
- Hosts地址：C:\Windows\System32\drivers\etc\hosts

CDN 是什么? 与 DNS 的关系?

- CDN： 是构建在数据网络上的一种分布式的内容分发网。 可以提高系统的响应速度，也可以一定程度的拦截/防御攻击。

常见的 DNS 安全攻击有哪些?

- 缓存投毒：它是利用虚假Internet地址替换掉域名系统表中的地址，进而制造破坏。
- DNS劫持：是指在劫持的网络范围内拦截域名解析的请求，分析请求的域名，把审查范围以外的请求放行，否则返回假的IP地址或者什么都不做使请求失去响应，其效果就是对特定的网络不能访问或访问的是假网址。（针对面较广）

- 域名劫持：域名劫持就是在劫持的网络范围内拦截域名解析的请求，分析请求的域名，把审查范围以外的请求放行，否则直接返回假的IP地址或者什么也不做使得请求失去响应，其效果就是对特定的网址不能访问或访问的是假网址。（针对面窄一点）
- DNS DDOS攻击：通过控制大批僵尸网络利用真实DNS协议栈发起大量域名查询请求，利用工具软件伪造源IP发送海量DNS查询，发送海量DNS查询报文导致网络带宽耗尽而无法传送正常DNS查询请求。

1.3 脚本语言

常见的脚本语言类型有哪些?

- asp php aspx jsp javaweb pl py cgi 等

不同脚本类型与安全漏洞的关系?

- 不同脚本可能爆发漏洞的可能性有所不同
- 不同脚本漏洞的存在点可能不同，因为不同语言的适用范围不同

1.4 后门

什么是后门?

- 通常指那些绕过安全性控制而获取对程序或系统访问权的程序方法。
- 在软件的开发阶段，程序员常常会在软件内创建后门程序以便可以修改程序设计中的缺陷。

后门在安全测试中的实际意义?

- 可以更方便的链接到主机
- 在获取到玩主机权限的时候，后门可以充当命令控制台的角色

1.5 WEB

WEB 的组成架构模型?

- 网站源码：分脚本类型，分应用方向
- 操作系统：windows linux
- 中间件（搭建平台）：apache iis tomcat nginx 等
- 数据库：access mysql mssql oracle sybase db2 postsql 等

为什么要从 WEB 层面为主为首?

- web使用的比较广
- web网站了漏洞相对较多
- web 作为跳板深入到其他资源相对容易

1.6 WEB 相关安全漏洞

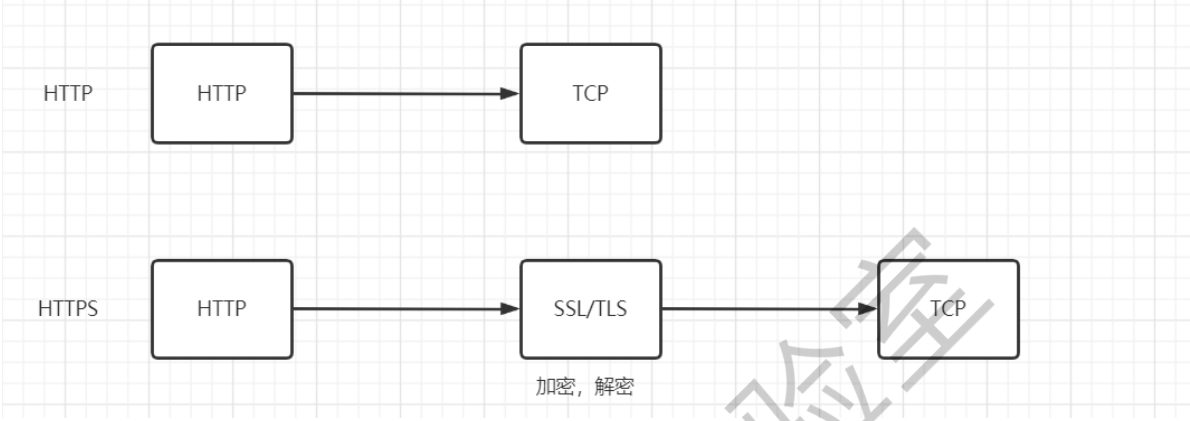
- WEB 源码类对应漏洞：SQL 注入，上传，XSS，代码执行，变量覆盖，逻辑漏洞，反序列化等
- WEB 中间件对应漏洞：未授权访问，变量覆盖...
- WEB 数据库对应漏洞：弱口令，权限提升...
- WEB 系统层对应漏洞：提权，远程代码执行
- 其他第三方对应漏洞
- APP 或 PC 应用结合类

资源

http://www.xyaz.cn
http://www.downcc.com/soft/11196.html
https://github.com/quasar/QuasarRAT/releases
https://pan.baidu.com/s/13_i1ExwEaA59GfMt1Rp0Hg提取码: 0b7b

P2. 数据包扩展

2.1 https&http



- HTTPS协议需要到CA申请证书，一般免费证书很少，需要交费。
- HTTP协议运行在TCP之上，所有传输的内容都是明文，HTTPS运行在SSL/TLS之上，SSL/TLS运行在TCP之上，所有传输的内容都经过加密的。
- HTTP和HTTPS使用的是完全不同的连接方式，用的端口也不一样，前者是**80**，后者是**443**。
- HTTPS可以有效的防止运营商劫持，解决了防劫持的一个大问题。

2.2 Request 请求数据包数据格式

Request Headers

```
POST /adduser HTTP/1.1
Host: localhost:8030
Connection: keep-alive
Content-Length: 16
Pragma: no-cache
Cache-Control: no-cache
Origin: chrome-extension://fdmmgilgnpjigdojojpjoooidkmcomcm
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko)
Chrome/66.0.3359.181 Safari/537.36
Content-Type: application/x-www-form-urlencoded
Accept: */*
Accept-Encoding: gzip, deflate, br
Accept-Language: zh-CN,zh;q=0.9
```

Form Data

- name=name&age=11

请求行

- 请求行由三个标记组成：请求方法、请求 URL 和 HTTP 版本，它们用空格分享。
- 例如：GET /index.html HTTP/1.1

HTTP 规划定义了 8 种可能的请求方法：
GET：检索 URL 中标识资源的一个简单请求
HEAD：与 GET 方法相同，服务器只返回状态行和头标，并不返回请求文档
POST：服务器接受被写入客户端输出流中的数据的请求
PUT：服务器保存请求数据作为指定 URL 新内容的请求
DELETE：服务器删除 URL 中命令的资源的请求
OPTIONS：关于服务器支持的请求方法信息的请求
TRACE：web 服务器反馈 Http 请求和其头标的请求
CONNECT ：已文档化，但当前未实现的一个方法，预留做隧道处理

请求头

由关键字/值对组成，每行一对，关键字和值用冒号分享。请求头标通知服务器腾于客户端的功能和标识。
HOST：主机或域名地址
Accept：指浏览器或其他客户可以接爱的 MIME 文件格式。Servlet 可以根据它判断并返回适当的文件格
式。
User-Agent：是客户浏览器名称
Host：对应网址 URL 中的 Web 名称和端口号。
Accept-Langeuage：指出浏览器可以接受的语言种类，如 en 或 en-us，指英语。
connection：用来告诉服务器是否可以维持固定的 HTTP 连接。http 是无连接的，HTTP/1.1 使用 Keep-Alive
为默认值，这样，当浏览器需要多个文件时(比如一个 HTML 文件和相关的图形文件)，不需要每次都建立连接
Cookie：浏览器用这个属性向服务器发送 Cookie。Cookie 是在浏览器中寄存的小型数据体，它可以记载
和服务器相关的用户信息，也可以用来实现会话功能。
Referer ： 表明产生请求的网页 URL 。如比从网页 /icconcept/index.jsp 中点击一个链接到网页
/icwork/search ， 在向服务器发送的 GET/icwork/search 中的请求中， Referer 是
http://hostname:8080/icconcept/index.jsp。这个属性可以用来跟踪 Web 请求是从什么网站来的。
Content-Type：用来表名 request 的内容类型。可以用 HttpServletRequest 的 getContentType()方法取得。
Accept-Charset：指出浏览器可以接受的字符编码。英文浏览器的默认值是 ISO-8859-1.
Accept-Encoding：指出浏览器可以接受的编码方式。编码方式不同于文件格式，它是为了压缩文件并加速文件传递速度。浏览器在接收到 Web 响应之后先解码，然后再检查文件格式。

空行

- 最后一个请求头标之后是空行，发送回车符和退行，通知服务器以下不再有头标。

请求数据

- 使用 POST 传送，最常使用的是 Content-Type 和 Content-Length 头标。

2.3 Response 返回数据包数据格式

Response 请求数据包数据格式

一个响应由四个部分组成；状态行、响应头标、空行、响应数据。
1.状态行：协议版本、数字形式的状态代码和状态描述，个元素之间以空格分隔
2.响应头标：包含服务器类型、日期、长度、内容类型等
3.空行：响应头与响应体之间用空行隔开
4.响应数据：浏览器会将实体内容中的数据取出来，生成相应的页面

HTTP 响应码：

```
1xx：信息，请求收到，继续处理
2xx：成功，行为被成功地接受、理解和采纳
3xx：重定向，为了完成请求，必须进一步执行的动作
4xx：客户端错误
5xx：服务器错
```

资源

https://www.mozhe.cn/
https://www.mozhe.cn/bug/N2dmMnp1b1VYd2EzLzBNR3MvRVlIZz09bW96aGUmozhe

P3. 搭建安全拓展

> 域名扫描只能扫描出来域名文件，而域名文件只是占服务器资源的一小部分；
> IP扫描可以直接扫描出来服务器的根目录，得到的信息更多；

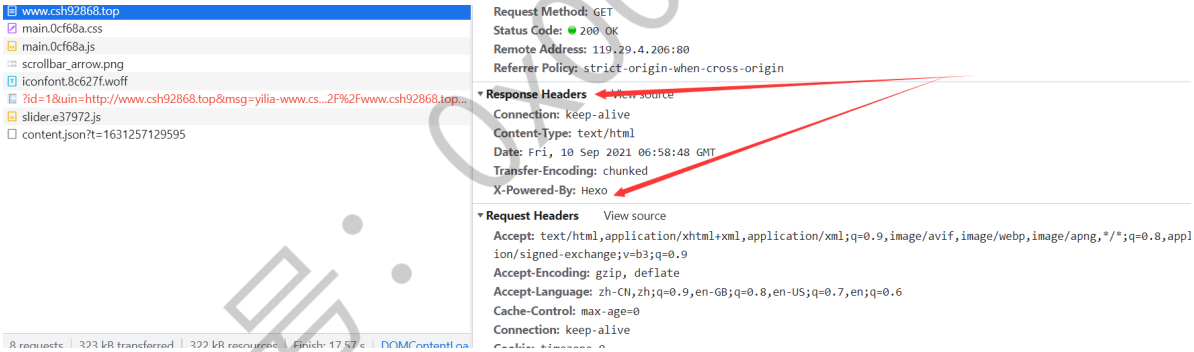
3.1 网站搭建环境

- ASP,PHP,ASPX,JSP,PY,JAVAWEB 等环境

WEB 源码中敏感文件

- 后台路径，数据库配置文件，备份文件等

3.2 基于中间件的简要识别



3.3 后门注意事项

- 后门是否给予执行权限
- 后门是否给予操作目录或文件权限
- 后门是否给予其他用户权限

3.4 Apache HTTPD 多后缀解析漏洞复现记录

Apache HTTPD 多后缀解析漏洞资源
Apache HTTPD 多后缀解析漏洞复现

P4. WEB 源码拓展

> WEB 源码在安全测试中是非常重要的信息来源，可以用来**代码审计**漏洞也可以用来做信息突破口，其中 WEB 源码有很多技术需要简明分析。

4.1 知识点

- 关于 WEB 源码目录结构
- 关于 WEB 源码脚本类型
- 关于 WEB 源码应用分类
- 关于 WEB 源码其他说明

4.2 信息敏感点

- 敏感目录结构：数据库配置文件，后台目录，模版目录，数据库目录等
- web脚本类型：ASP,PHP,ASPX,JSP,JAVAWEB 等脚本类型源码
- 应用分类：社交，论坛，门户，第三方，博客等不同的代码机制对应漏洞
- 开源，未开源问题，框架非框架问题，关于 CMS 识别问题及后续等
- 关于源码获取的相关途径：搜索，咸鱼淘宝，第三方源码站

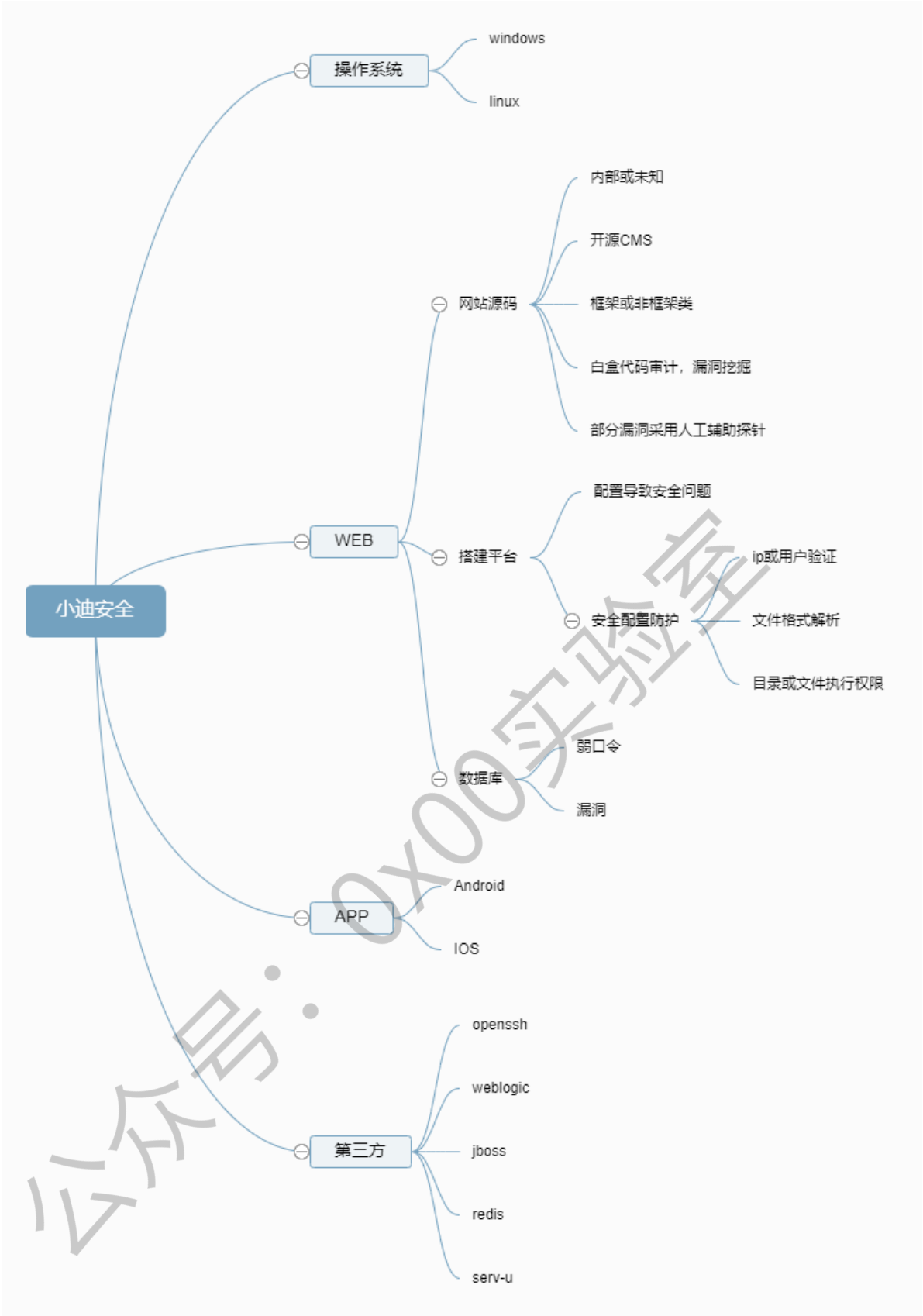
关注应用分类及脚本类型估摸出可能存在的漏洞（其中框架类例外），在获取源码后可进行本地安全测试或代码审计，也可以分析其目录工作原理（数据库备份，bak 文件等），未获取到的源码采用各种方法想办法获取。

style.css：这个文件可以用来生成md5值来查寻cms框架。
可以通过网页刷新时加载的文件来搜索cms。

资源

https://www.yunsee.cn
https://w.ddosi.workers.dev
http://whatweb.bugscaner.com
https://github.com/Lucifer1993/cmsprint
https://github.com/M4tir/Github-Monitor
https://websec.readthedocs.io/zh/latest/language/index.html

P5. 系统&数据库



5.1 操作系统层面

识别操作系统常见方法

- 看字母大小写，windows对大小写不敏感，Linux敏感
- 看ping值
  --ttl在64左右，linux
  --ttl在128左右windows
- nmap -O ip

简要两者区别及识别意义情况

- 可以帮助我们明确思路
- 可以筛选掉不符合系统的情况

操作系统层面漏洞类型对应意义

- 覆盖面广
- 获取的权限高
- 危害性大

5.2 数据库层面

识别数据库类型常见方法

- nmap -O ip
- nmap ip -p 端口，通过端口开放反推数据库

数据库类型区别及识别意义

- 数据库的漏洞和类型相性很强
- 不同数据库漏洞爆发点不太一样
- 能确定数据库类型、版本，会对渗透有很大帮助

数据库常见漏洞类型及攻击

- 弱口令
- sql注入

简要数据库层面漏洞影响范围

- 要参考数据库的重要程来判定影响范围

常见的数据库结构

- ASP+Access
- php+mysql
- axpx+mssql
- jsp+mssql，oracle
- python+mongodb

服务器端口

- 关系型数据库
  --MySQL:3306
  --SqlServer:1433
  --Oracle:1521
- NOSQL数据库
  --MongoDB:27017
  --Redis:6379
  --memcached:11211

第三方

- 如何判断那些有第三方平台或软件
  -- 端口扫描
  -- 特征匹配
- 简要为什么要识别第三方平台或软件
  -- 可以提供额外的攻击面

- 常见第三方平台或软件漏洞类型及攻击
  -- 弱口令
- 简要第三方平台或软件安全测试的范围

> 除去常规WEB安全及APP安全测试外，类似服务器单一或复杂的其他服务(邮件，游戏，负载均衡等)，也可以作为安全测试目标，此类目标测试原则只是少了WEB应用或其他安全问题。所以明确安全测试思路是很重要的!

资源

https://nmap.org/
https://www.kali.org/downloads/
https://github.com/hellogoldsnakeman/masnmapscan-V1.0
https://pypi.tuna.tsinghua.edu.cn/simple-------清华大学python镜像
https://pypi.tuna.tsinghua.edu.cn-------清华大学python镜像