就业班第一阶段面试

自我介绍

领导您好，我叫张红伟，刚刚毕业于云南红河学院的信息安全专业。

我们学校当时是在和绿盟合作，组织过关于网络安全的攻防实训，也是掌握了一些安全基础，像一些常见的欺骗攻击，SQL注入，XSS，CSRF和一些常见的web安全漏洞。

代码学过python和java，对python理解更深一点，因为我的我的毕业设计就是用Python构建一个文本的情感分析模型。

毕业后呢也参与了一些行业内培训，目前掌握了winsows和linux的系统维护，安全检查，

还了解过阿帕奇，Nginx，Tomcat,weblogic这些中间件的配置维护和历史漏洞复现过程，还有像mysql,sqlserver,redis这样的常用数据库的安装使用，检查维护和历史漏洞复现

平时有时间的时候我也会去找一些小网站小程序，尝试挖一些简单的逻辑漏洞。

这些就是我目前的一个情况了

确定公司域名，用站长工具或者小蓝本，爱企查里找某个公司的域名

然后用扫子域名（Oneforall.py，fofa），找那种用CMS（找特征）（Joomla的企业门户，积分商城，论坛，Drupal，WordPress博客）搭的边缘站点

然后进行指纹识别，云悉，JSfinder，glass，ehole可以扫站点系统版本，中间件和应用框架

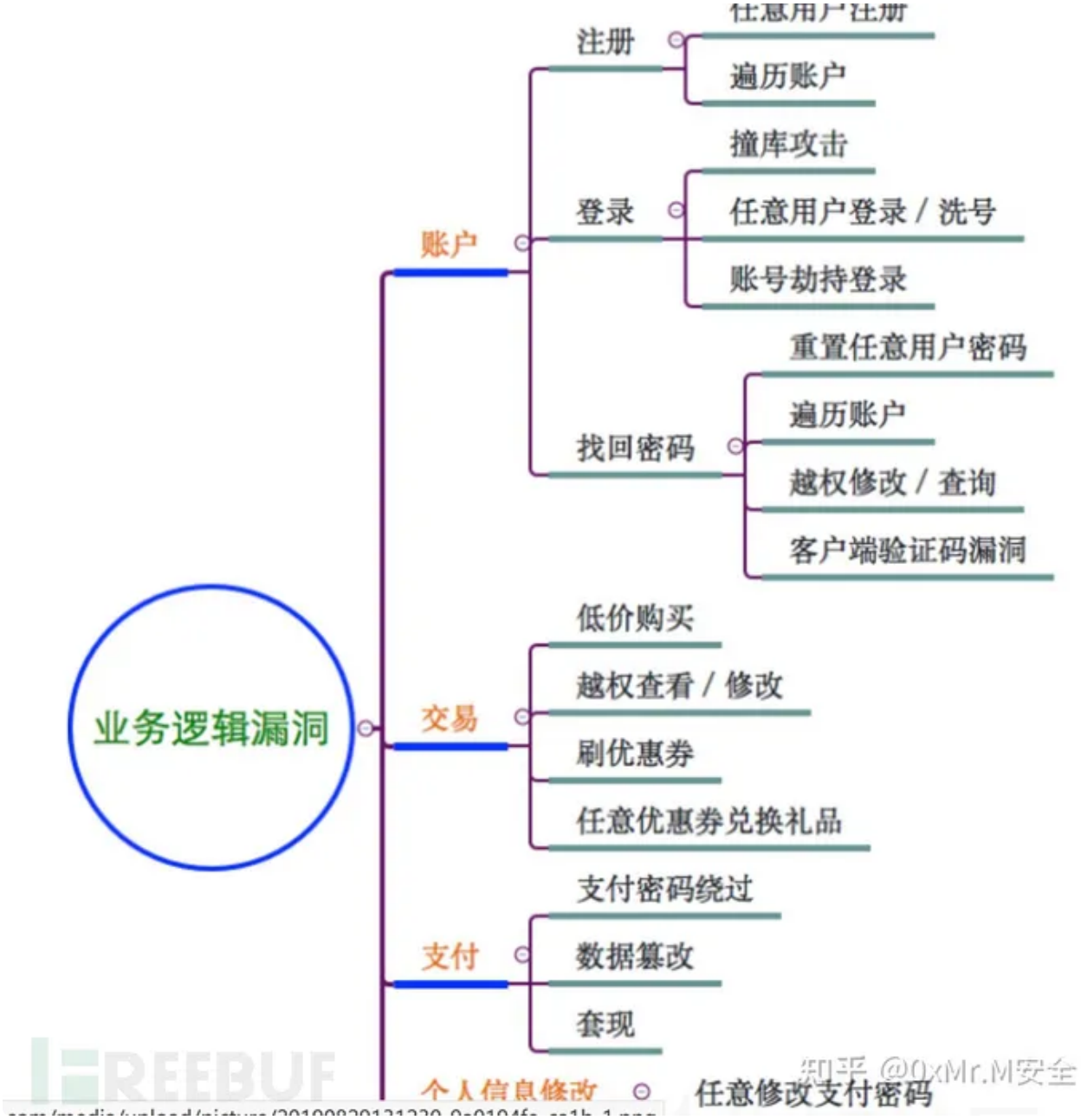
测试这些架构对应版本网上发布出来的Nday

还可以用Nmap，御剑扫端口扫目录

然后使用bp测试一些业务逻辑漏洞

另外小程序也是测试的一部分，可以用Proxfiler搭配burp来实现抓包改包发包的测试操作

测业务逻辑漏洞的时候先自己注册账号去试用户的功能点，在交易，越权方面可能会有发现，找了一圈之后再回到登录注册模块去尝试绕过



修改返回包的越权

id替换，遍历，或者修改cookie

- **如何手工快速判断目标站是 Windows 还是 Linux 服务器?**

  因为linux对大小写敏感，可以换一下url里面某个字符的大小写，改了访问不到就可能是Linux

  或者说ping一下看ttl值，linux是一般默认是64或255，Windows一般默认是32或128

- **常见的状态码有哪些?**

  它是有1到5做开头

  1开头的表示临时响应，比如100是等待客户端继续请求

  2开头表示请求成功，比如响应GET和POST请求的200

  3开头表示资源重定向，302是临时重定向，301是永久重定向，305是必须使用代理访问

  4开头表示客户端请求错误，找不到文件返回404，找到了无权查看返回403，请求超时返回408

  5开头表示服务器端错误，比如服务器内部错误的500，服务不可用的503

- **简述OSI七层模型，TCP/IP模型以及常见的协议**

  OSI七层模型是用来描述计算机网络系统的概念框架(应表会传网数物)

  应用层是网络服务和最终用户的接口，这一层的协议主要是应用程序和服务的协议

  有用于在网络上传输文件的FTP是21(实现控制连接),20(实现数据连接)

  专为远程登录回话和其他网络服务提供安全性保障的SSH是22

  本地远程访问主机的Telnet:23

  客户端可以用来发送邮件的简单邮件传输协议SMTP:25

  将域名和ip相互映射的域名系统DNS:53

  集中管理，分配ip地址的动态主机配置协议DHCP:68(客户端请求到服务端的68号)67(服务器应答给客户端的67号)

  用于实现web请求响应的HTTP:80

  然后基于HTTP，加了SSL和TSL加密，身份验证以及数据完整性保护HTTPS:443

  适用于C/S结构脱机模型的电子邮件协议POP3:110

  还有远程桌面协议RDP是3389

  表示层是和加解密，翻译转换和解压缩相关

  LPP(轻量级表示协议)

  会话层是负责建立，管理和终止回话

  主要是像SSL,TSL这些的安全传输协议

  传输层是定义传输数据的协议和端口号，以及控流和差错校验，有TCP和UDP

  TCP是建立一对一的面向字节流的全双工可靠连接，可以实现流量控制和拥塞控制，像ftp,ssh,telnet,http,https,pop3,关系型数据库，默认端口3306的mysql,1521的Oracle,1433的SQLserver，还有默认端口是7001的java应用服务器Weblogic，这些相关服务协议用的都是TCP传输协议

  UDP是传输效率更高，但传输过程并不可靠，像DNS，DHCP，NTP(网络时间协议)用的就是UDP传输协议

网络层主要是进行逻辑寻址，实现不同网络之间的路径选择

主要协议有IP，ICMP，IGMP，以数据包为传输数据的基础单位，网络层以下的协议都不使用端口号，

数据链路层有建立逻辑连接，硬件地址寻址的功能，以太网和点对点协议（PPP）就是工作在数据链路层，以数据帧为传输数据的基础单位

物理层主要是定义数据传输过程中的电气，机械，功能的特性，以及传输介质，传输设备的一些标准，以bit为传输数据的基础单位

TCP/IP模型

把OSI模型的应用层，表示层，回话层整合为一个应用层，数据链路层和物理层整合为网络接口层，也可以叫数据链路层，就是TCP/IP模型

- **三次握手和四次挥手流程**

  假如我是客户端，您是请求端哈，第一次握手是我给您发过去一个值x，

  然后如果您成功收到了，为了表示您收到了，你需要把我给你发过去的值x加1发回来给我，同时你还得验证你到我的通路是正常的，所以你得同时给我另一个值y

  最后如果我收到了这个X+1和Y之后，我会把Y值加1，连同你经过你加1操作的X再发给你

  您成功收到包后三次握手都结束，证明双向通路正常，就可以开始传输数据了

  四次挥手就是我不想继续使用您作为服务器提供的服务了，

  我会给你发一个值x，您收到这个值会后，如果你同意关闭服务，会把x+1和另一个数字y发给我，然后再发一个包含x+1和数字z的包给我，意思是告诉我服务即将关闭，最后我把你第二次发给我的包里的z+1和x+1发给您，然后您关闭本次服务，四次挥手结束，一次TCP传输过程也就正常终止了

- **简单描述常见的协议以及默认端口号**

  常见的协议一般都是工作在应用层服务终端用户的，或者工作在传输层定义数据传输

  比如应用层，有用于在网络上传输文件的FTP是21(实现控制连接),20(实现数据连接)

  专为远程登录回话和其他网络服务提供安全性保障的SSH是22

  本地远程访问主机的Telnet:23

  客户端可以用来发送邮件的简单邮件传输协议SMTP:25

  将域名和ip相互映射的域名系统DNS:53

  集中管理，分配ip地址的动态主机配置协议DHCP:68(客户端请求到服务端的68号)67(服务器应答给客户端的67号)

  用于实现web请求响应的HTTP:80

  然后基于HTTP，加了SSL和TSL加密，身份验证以及数据完整性保护HTTPS:443

  适用于C/S结构脱机模型的电子邮件协议POP3:110

  还有远程桌面协议RDP是3389

  这些都是比较常见的应用层协议

  然后在传输层，还有比较重要也是比较常见的数据传输协议就是传输控制协议TCP和用户数据报协议UDP

  TCP是建立一对一的面向字节流的全双工可靠连接，可以实现流量控制和拥塞控制，像ftp,ssh,telnet,http,https,pop3,一些关系型数据库，默认端口3306的mysql,1521的Oracle,1433的SQLserver，还有默认端口是7001的java应用服务器Weblogic，这些相关服务协议用的都是TCP传输协议

  UDP是传输效率更高，但传输过程并不可靠，像DNS，DHCP，NTP(网络时间协议)用的则是UDP传输协议

- **Windows的基线检查**

  知道一些常规项目

  比如检查是否设置了合理的密码使用期限策略和强制密码历史，是否启用密码复杂性要求和密码错误多次账户锁定策略

  检查是否设置匿名账户访问控制，是否禁止未登录关机，是否合理的设置了可关闭或远程关闭电脑的账户和组

  检查是否设置了会话超时自动断开，注册表有没有异常启动项，锁定会话时显示信息的级别是否合理，以及是否对所有驱动器关闭Windows自动播放

- **在对系统进行维护过程中，你如何评估系统上的NTFS权限设置是否安全?**

  我会通过分析和检查NTFS的权限配置，找潜在的安全漏洞和风险，比如权限配置是否遵循最小特权原则，有没有权限过度授予或权限继承断裂这些问题，根据这些来确定系统的安全性水平

- **与共享权限相比，NTFS权限有什么特点**

  共享文件是对共享资源进行控制的，NTFS权限是对每个文件或文件夹单独设置权限，实现一个更精细化的权限控制

- **怎么理解NTFS权限的最小特权原则?**

  按最低限度配置用户权限，让他刚好可以完成所需工作，这种严格限制使用者权限的办法这样可以最大程度降低系统被攻击破坏的风险

- **你对Windows批处理了解吗？它们通常用于做什么?**

  批处理就是通过组合多条cmd命令的.bat文本文件，它的特点对一些较复杂的操作本来需要一条一条地执行很多命令，但是你把这些命令写成批处理脚本后，只需要点击运行这个批处理脚本文件就行了，就是用来更快捷得执行一组文件操作，程序运行，信息获取的的一些相关命令的方法

- **在渗透测试中，如果想让你写一个批处理脚本用于收集系统信息，你会怎么写?**

  可以通过systeminfo获取系统信息,ipconfig获取ip配置信息,netstat获取网络配置信息，tasklist获取当前正在运行的进程列表，reg query 查询Windos下的CurrentVersion来获取系统版本，组件，设置等相关信息，然后把这些信息先后保存到txt文件里

  具体写的话，就是先@echo off 关闭回显

  依次执行

  systeminfo

  ipconfig /all

  netstat –ano

  tasklist

  reg query 注册表中Windows下的CurrentVersion获取操作系统设置，版本，组件相关信息

  执行的同时要分别把信息写入到指定的txt文件中

  最后把所有信息整合到一个文件里，就行了

系统信息收集批处理脚本 Plain Text

```
@echo off
echo 正在收集系统信息，请稍候...

rem 收集系统信息
systeminfo > system_info.txt
ipconfig /all > network_info.txt
netstat -ano > netstat_info.txt
tasklist > tasklist_info.txt
reg query "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion" > registry_info.txt

echo 系统信息收集完成。
rem 生成报告
echo 生成报告中...
echo. > report.txt
echo系统信息报告  >> report.txt
type system_info.txt >> report.txt
echo. >> report.txt
echo网络信息报告  >> report.txt
type network_info.txt >> report.txt
echo. >> report.txt
echo  网络连接报告>> report.txt
type netstat_info.txt >> report.txt
echo. >> report.txt
echo  进程列表报告 >> report.txt
type tasklist_info.txt >> report.txt
echo. >> report.txt
echo  注册表信息报告>> report.txt
type registry_info.txt >> report.txt

echo 报告已生成，存储在 report.txt 文件中。
Pause

```

- **你如何保证在渗透测试中使用Windows批处理不会对系统造成损坏或不可逆的影响?**

  可以虚拟机这样的安全环境中测试脚本，确保不会损坏系统，然后在执行脚本之前备份系统防止意外

- **Linux如何保护SSH?**

  禁止root远程登录

  禁止空口令登录

  设置超时自动退出

  如果安全等级高甚至可以禁止口令，使用私钥文件登录

- **什么是Linux的权限？如何修改它们?**

  Linux的权限决定了用户或用户组对文件或目录的访问权限

  常规的有rwx（421），也就是读写执行三种权限，有当前用户，当前用户所属组，其他用户三组权限配置，可以通过chmod更改权限，chown更改所属用户或所属组

- **SUID和SGID如何影响文件权限?**

  SUID和SGID都是特殊类型的文件权限

  SUID只针对可执行文件权限复制，作用于普通文件没有任何意义，用户执行添加了SUID权限的文件后，发起的进程将会以文件所有者的权限运行(o+s)

  SGID针对可执行文件赋权的时候，发起的进程会以用户所属组的权限运行

  SGID针对目录赋权时，该目录创建文件属组集成父目录属组(g+s)

  粘滞位（stickbit）

  将粘滞位赋权给目录后，在该目录下创建的目录或文件只有创建者和管理员可以删除

- **Linux如何查看系统内核版本?**

  linux的/proc/version文件里的第一行内容就是系统内核版本

  在bash终端里uname –a查看所有系统信息，包括系统内核版本

- **Linux的基线检查**

  检查是否存在空账号，root权限账号，然后禁用和删除无用账号，检查密码使用期限，密码复杂度，连续错误多次锁定的相关策略。设置umask（决定用户创建文件的默认权限）值防止默认权限过高

  远程登录取消telnet采用SSH，检查SSH服务安全，修改SSH协议版本为2，关闭不必要的服务，禁止root直接登录和远程登录，设置连接超时自动退出

  检查是否启用syslogd日志，配置日志目录权限或者日志服务器，记录所有用户的登录和使用过程

- **拿到一个Webshell发现有.htaccess文件，能干什么?**

  .htaccess是apache配置特定目录特定规则的文件，他可以改变网页的行为，比如重定向，改变默认页面，指定密码保护等，可以查看这个文件获取重定向规则，密码保护目录，错误保护路径，在有权修改的情况下还可以将某个页面重定向到其他页面，或者插入FilesMatch标签使jpg文件被解析成php文件

- **你知道那些中间件的解析漏洞?**

  apache的2.4.0到2.4.29存在解析换行漏洞，他在解析文件时会匹配文件名后面的换行符，攻击者可以在上传木马文件时在文件名后面加上换行的十六进制转义符\x0A来绕过文件上传限制，获取webshell

  apache还有一个多后缀解析漏洞，apache是允许一个文件可以拥有多个以点分割的后缀名识别时从左往右依次识别，如果管理员为.php后缀的文件添加处理程序是配置错误，就会导致只要后缀中存在.php，文件就会被当做php文件执行，也成为了攻击者获得webshell的一种方式

  Nginx也存在文件名逻辑漏洞，访问路径中如果有空格或者空字符，会导致Nginx错误解析URL地址，攻击者可以通过在路径中加入空格\0.php来把其他类型的文件解析成php文件，实现绕过服务器限制解析php文件

- **Tomcat外部访问默认端口**

  默认请求端口8080，8443

  服务关闭端口8005

  默认AJP服务端口8009

- **Tomcat常见漏洞类型以及如何利用**

  Tomcat漏洞主要涉及远程代码执行，后台弱口令，文件包含，控制台暴露

  远程代码执行分操作系统，如果在Windows系统上运行，启用了HTTP的PUT请求，攻击者可以通过构造攻击请求向服务器上传包含任意代码的jsp文件，实现任意代码代码执行。如果是Unomi（java服务器）服务器，可以通过构造MVEL或ONGl表达式来发送恶意请求,使Unomi服务器执行任意代码

  弱口令漏洞可以通过BP爆破进入后台上传war包获取shell

  文件包含漏洞是因为AJP协议存在缺陷，默认情况下AJP连接是打开的，可以通过构造特定参数读取或包含（加载或运行）服务器webapps下的任意文件，如果目标服务器同时可以上传文件，就可以进一步实现远程代码执行

  控制台暴露漏洞是当Tomcat以系统管理员身份或以系统服务运行时，java取得了系统用户或系统管理员的全部权限，而且Tomcat的默认账密都是tomcat，攻击者可以登录后台上传war包，进而控制整个服务器

- **Tomcat重启后，webapps下删除的后台会不会又回来?**

  不会，webapps目录是tomcat查找要部署的web应用程序位置，对该目录做的修改也会在重启时反映出来，但是如果用war文件部署应用程序，只要war文件还在webapps下面，重启时就会重新部署

- **如何获取并确定Weblogic资产**

  fofa，钟馗之眼，谷歌hacking，然后访问网站有Error 404，应该就是Weblogic

- **Weblogic常见漏洞**

  主要有弱口令，SSRF服务器端请求伪造，任意文件上传还有反序列化漏洞

  弱口令是通过爆破账密登录后台

  SSRF是攻击者可以发送任意http请求攻击内网中的Redis，fastcgi等脆弱组件

  任意文件上传漏洞是启用web服务测试页后，通过访问ws_utc/config.do上传木马获得shell

  反序列化漏洞是基于Weblogic T3 协议和wls-wast组件引起远程代码执行的反序列化漏洞，T3协议可以在前台无需登录的情况下进行RMI反序列化漏洞攻击，WLS组件对外提供webservice服务，使用XML解码器解析用户传入的XML数据，解析过程中出现反序列化漏洞，导致任意代码执行

- **weblogic弱口令渗透思路**

  爆破控制台登录密码，进入控制台后部署木马war包 ，获得shell

- **Weblogic任意文件上传渗透思路，如何防御?**

  访问ws_utd/config.do，更改静态文件路径，上传jsp木马war包，获取shell

  防御思路就是关闭web服务测试页

  升级版本，设置config.do登录授权后访问，在ips等防御产品中加入相应特征

- **Weblogic XML解码器反序列化渗透思路以及如何防御?**

  使用漏扫工具发现漏洞，BP抓包，改写poc包，成功写入文件的话尝试写入反弹shell

  防御思路是根据实际环境删除wls-wsat组件

  下载官方安全补丁

- **得到mysqlde权限之后写webshell的条件**

  在知道网站绝对路径，然后Mysql配置中secure_file_priv为空的前提下，可以select into outfile命令写入木马，也可以通过备份上传一句话木马获取webshell

- **mysql的网站注入，5.0版本和5.0以上有什么区别?**

  5.0以下是没有information_schema这个数据库的，不能列表名，只能跑表名，5.0以下是多用户单操作，5.0以上是多用户多操作

- **如何判定靶机数据库是Mysql?**

  扫端口，看默认3306是否开启，开了的话应该是Mysql

  注入过程中有报错limit，就可能是Mysql或者postgresql

  还有可以用select version()，看版本信息，5或者8开头的就是mysql

- **如何判断靶机的数据库是sql-server**

  扫描端口1433是否开启

  注入过程如果报错里面有top说明是sqlserver

  注入过程中，可以拼接select @@version查看版本信息，如果查询结果包含Microsoft或sql server就是sqlserver

- **mysql用户密码存放位置，加密方式?**

  mysql数据库的users表下面有一个authentication_string字段，使用的是sha1加密生成一个160位的散列值

- **Redis的未授权访问漏洞可以采用什么样的渗透手段?**

  获取包含敏感信息的键值对

  通过持久化上传一句话木马获得websehll

  利用持久化把攻击机SSH公钥写入服务器，然后私钥登录

  利用持久化写计划任务获得反弹shell

  利用主从复制搭配redis-rogue-server工具获得反弹shell

- **mysql基线检查**

  检查是否禁用local_infile选项

  是否修改默认端口

  是否使用非root权限用户启动数据库服务

  是否配置log-error选项

  以及是否禁用了log-raw选项

  确保没有用户配置了通配符主机名（%）

  是否允许匿名登录，如果有删除匿名账户

  是否配置skip-smymbolic-links选项为yes，忽略所有符号链接

- **Mysql配置文件，mysql数据库端口**

  linux是/etc/my.cnf

  windows是安装目录下的my.ini

  端口3306

- **Sqlserver基线检查**

  删除无关账号，检查空密码账号，密码更新时间和密码强度

  检查是否按用户分配账号，避免共享账号

  检查用户权限配置是否符合最小权限原则

  更改服务端口，只允许可信ip建立远程连接

  配置日志功能，记录数据库安全事件、

  停用不必要的存储过程，更新系统补丁

- **redis基线检查**

  使用非root用户启动

  是否仅绑定本地或内网ip

  启用保护模式

  设置redis的文件权限为600

  更改服务端口

  禁用或重命名危险命令

  启用高复杂度的密码认证(config set requirepass 密码，登录时auth 密码)

  在服务允许的情况下，更新漏洞风险比较低的Redis版本

- **查询某个数据库中某个表的所有列**

  select column_name from information_schama.columns where table_name='xx' and table_schema='xx'

- **菜刀webshell流量特征**

  菜刀一般是Post请求,只有一个参数，用base64编码，数据解码后都系统命令

  请求来自固定ip

  请求的路径一般是php，jsp或者asp

  请求体里面有assert,eval,base64

- **蚁剑流量分析**

  每个请求体都存在@ini_set(“display_errors”, “0”);@set_time_limit(0)开头。并且后面存在base64等字符
  响应包的结果返回格式为：
  随机数
  响应内容
  随机数

- **冰蝎流量分析**

  冰蝎2.0流量特征：

  第一阶段请求中返回包状态码为200，返回内容必定是16位的密钥

  请求包存在：Accept: text/html, image/gif, image/jpeg, ; q=.2, /; q=.2

  建立连接后所有请求 Cookie的格式都为: Cookie: PHPSESSID=; path=/；

  3.0是每个请求头里都有一段固定的Pragma和accept行

- **哥斯拉流量分析**

  请求包有固定的accept行

  返回包有固定的Cache-Control行

- **mysql读取文件方法和读取条件**

  load_file()

  load data infile()

  条件是有这个文件的权限和secure_file_priv配置项不为空

  还可以使用system cat查看文件内容

- **如何利用sqlserver上传一句话木马**

  主要是通过备份上传一句话木马，在某个数据库下建一个只有一列文本类型字段的表，然后备份数据库，然后把一句话木马写入表里，再备份到远程可以利用木马的文件路径下，就上传成功了，也可以把一句话木马转成16进制，通过上传image格式的图片马绕过Windows安全检测

- **拿到一个sql注入点后，后续操作?**

  查看数据信息，爆表爆库

  利用数据库备份上传一句话木马获取系统权限

  找到数据库账密后登录后台利用文件上传功能上传一句话木马

  使用UDF和MOF提权方式执行系统命令创建管理员用户

- **sql注入原理**

  数据库的用户输入和程序命令之间没有做到泾渭分明，接受用户输入的相关参数未经处理直接带入数据库查询，让攻击者可以将其他拼接在url里提交给web程序然后执行

- **sql注入成功的基础**

  sql命令的正确拼接

  应用程序相信用户输入的数据

- **如何检测sql注入点?**

  找asp,php,jsp,aspx结尾的url网页链接，还有登录，注册，留言板，修改密码等post提交数据的地方

  在id或者page参数后面加上单引号，双引号，括号，斜杠看报错，一般报错会显示数据库类型，然后根据报错猜测它的查询命令，尝试闭合，闭合成功就可以根据回显的列数来实现注入

  通过and 1=1和and 1=2，返回不同页面说明存在注入漏洞

- **数据库注入类型**

  按注入网页类型分登录注入和CMS注入

  按注入点值的属性分为数字型和字符型

  还可以分有回显，无回显，一阶注入，二阶注入

  其他业务场景下还有update,delete,insert,like,order by这些相关命令的注入

  还有宽字节注入，http分割注入，HTTP参数污染等等，每种注入方式都有自己的流量特征，可以根据流量特征辨别是否是sql注入

- **mysql注入爆表暴库会利用什么信息作为辅助?**

  第一个mysql5.0以上会有一个默认的information-schema数据库，里面存储里整个数据库的相关信息

  tables表里记录了所有的数据表和数据表所属的数据库

  columns表里记录了所有字段和字段所属表以及字段所属数据库

  schemata表里记录了所有的数据库名，以及数据库相关信息

  可以通过这个默认表查询获取数据库的表明和字段名，进而查找字段里的数据

  然后是一些mysql常用函数

  user();查看当前登录mysql的用户名

  database();查看当前使用的数据库

  version();查看当前数据库版本

  length():字符串长度

  substr()和mid();截取指定长度的字符串

  ascii();和ord():是返回ASCII码值

  concat();连接字符串

  group_concat();将多个查询结果连接起来，放同一行

  limit m,n 从m开始，到m+n行

- **sqlserver爆表暴库会利用哪些信息?**

  sqlserver有一个master数据库，可以通股票这个数据库下的Sysdatabases查看所有数据库的名字

  然后所有表名可以通过Sysobjects数据库查看，所有的字段可以通过Syscolumns数据库查看

  @@version可以查看数据库版本

  @@servername可以查看主机版本

  hostname是当前数据库名

  sb_name然后中括号里填填数字就是查询对应的数据库名，不填就是第一个数据库名，括号里写1就是第二个数据库名

- **sql注入过程中的注释有什么用?**

  注释掉后面的sql语句，起到闭合sql命令的作用

  另外还可以通过内联注释绕过web应用防火墙

- **为什么一个mysql数据库服务器只有80端口开放?**

  可能更改里数据库端口，或者说做了站库分离（就是网站程序和数据库分别部署在了两台服务器上）

- **如何突破注入时字符被转义?**

  宽字符注入，或者16进制编码绕过（hex编码绕过）

- **sql注入的绕过方式**

  大小写绕过，双写绕过，url编码绕过，内联注释绕过

- **注入时如果and和or，或者if被过滤了，如何绕过**

  mysql对大小写不敏感，所以改大小写搭配双写尝试绕过

  通过16进制或者url编码绕过

  用符合替换关键字，比如and可以用两个逻辑与符号&&，or可以用两个逻辑或符号||替换

  还可以使用内联注释/*!*/和多行注释/**/绕过

- **注入时空格被过滤了，怎么办**

  换url编码，看%0a,0b,0c,0d,09,还有%a0能不能行

- **SQL注入中，可以利用的报错函数有哪些?**

  updatexml(),extractvalue(),floor(),如果在监控流量时发现了这些函数，应该适当配合其他信息判定是否是注入

  updatexml()本来是通过三个参数来更改指定文档，指定标签下的内容的，但是可以在xml标签路径下注入sql语句引发路径错误，然后SQL命令会被执行然后返回到报错中

  ' and updatexml(1,concat(0x7e,(payload),0x7e),1) or '1'='1

  extractvalue函数是用来提取指定位置xml数据的，但和updatexml函数一样，在标签路径的地方插入sql查询语句，可以导致路径报错的同时执行sql语句并回显到报错语句中

  ' and extractvalue('1',concat(0x7e,(payload),0x7e)) or '1'='1

  通过向下取整函数floor和随机数生成函数rand搭配制造制造主键重复错误，把sql查询语句和floor组合，可以在返回逐渐重复报错的时候执行sql语句并回显

  ?id=1' union select 1,2,3 from (select count(*),concat((select concat(version(),database(),user()) limit 0,1),floor(rand(0)*2))x from information_schema.tables group by x)a --+

- **盲注分类**

  盲注都不回显信息，需要利用if这样的函数来做一个字符一个字符的判断，

  时间盲注就是页面没有任何回显，只能通过构造if条件，如果条件为真则延时一段时间来试出数据库信息

  布尔盲注就是能通过网页反馈得知当前语句条件是否为真来获取数据库信息

- **如果网站get和post都做了防护，如何绕过?**

  http头注入，

  比如上市UA头，Referer，cookie字段的注入是否可行

- **注入漏洞只可以查账号密码吗?**

  账号密码是最低权限就可以实现的

  sqlserver的sa账号可以获取系统权限，dbowner可以获取webshell，public账号可以拖库

  mysql的话可以在知道网站路径，而且mysql的secure_file_priv的值为空时，可以通过上传木马获得websehll

  当然如果拿到了后台管理员账号，可以找后台漏洞，比如提权，文件上传等等

- **发现demo.jsp?uid=110注入点，过去websehl的思路有哪些?**

  如果有写入权限就select into file，把查询数据保存到文件中上传木马获取webshell

  还有sqlmap有一个–os–shell的选项，和上传木马的方式原理类似，但是更快

  但最容易实现的，也最优选的方法是构造联合查询语句获得管理员的账密，扫出后台登录后该包上传木马获取webshell

- **sqlmap使用方法**

  get型的话就是sqlmap –u 引号包裹的注入点的网址，后面跟––batch和要查询的内容

  如果是post或者http头注入，就需要在用bp抓post包，把注入点换成星号，然后保存到文本文件中，sqlmap –r 加文件，后面跟––batch,––purge和要查询的内容

  有些情况下默认注入不一定会成功，可以指定level和risk参数来调整注入测试等级和注入风险参数尝试更多攻击方法来绕过一些安全措施

  –r是指定标记注入点的数据包

  –u是指定注入的url
  –batche–smart 智能判断测试

  ––mobile 模拟测试手机环境站点

  –m 批量注入

  ––current–user 获取当前数据库用户

  ––current–db 获取当前连接数据库名

  ––is–dba 判断当前用户是否为管理

  ––users 列出数据库所有用户

  ––dbs 列出当前服务下所有数据库

  ––tables 数据库名

  ––columns –T tablename –D dbname 查看某个数据库下某个表的字段名

  –T tablename –C username,passwo rd ––dump 获取当前连接数据库下指定表的字段值

  ––file–read 读文件

  ––os–shell 系统交互shell

  ––tamper ""使用脚本，kali的脚本是在**usr/share/sqlmap/tamper/**

- **有的sql注入漏洞会出现类似base64编码，如何利用?**

  通过对参数也base64加密来尝试注入

- **SQL注入的防护方法**

  用函数过滤掉注入需要用到的一些字符，比如空格，if，and，or，union，select，注释符等等，指定id必须为数字

  使用腾讯，阿里云，360提供的防注入脚本，直接用include包含放在网站配置目录里就行

  采用php数据对象（pdo）预处理和waf拦截

- **如果逗号被过滤了怎么办?**

  可以在注入语句中使用join来尝试绕过

  比如说select 1,2,3用select * from （select 1)a join (select 2)b join (select 3)c

- **什么是二次注入？和普通注入有什么区别**

  二次注入就是通过向数据库存储恶意数据后再利用的注入方法，之前了解过的就是注册恶意账号名，然后登录后可以重置指定用户的密码，二次注入更难被发现，但是条件是恶意数据成功上传，而且数据库必须完全相信用户存储的数据，然后在直接将恶意数据交给用户

  可以通过这种方式绕过注入关键字被转义的情况

  普通注入很容易被扫描工具扫到，二次注入需要先把恶意数据存入数据库，然后利用恶意数据实现注入

- **sql注入无回显，盲注又被封了，这时利用dnslog来获取回显信息，解释一下利用原理，使用条件，以及怎么利用?**

  把注入语句组合在子域名里然后发送域名解析请求给dnslog提供的域名服务器上，然后所有查询的结果都会以拼接域名的样式泄漏到dnslog平台上

  需要数据库中有可以直接或间接引发dns解析的子程序，比如UNC路径(一种命名惯例，实现网络硬盘共享，格式是\服务器名\共享文件夹， 常用来在局域网中共享文件服务器或者打印机)，但是linux是没有unc的，所以linux服务器只能通过ping或者curl这些来实现利用

  mysql是通过select load_fIle()，把查询语句和dnslog平台指定的子域名组合的形式来实现利用是，但是因为用到了load_file，所以mysql配置项中的secure_file_priv必须为空，也就是账户必须要有写的权限

  sqlserver就是利用存储过程，先创建一个变量，像mysql一样拼接数据库查询语句和dnslog平台提供的子域名，然后使用

  master..xp_diretree

  master..xp_fileexist

  master..xp_subdirs

  这三个存储过程任选其一来实现利用

  在利用的时候还有一个条件，就是sql查询语句长度是受unc路径最大128字符的长度限制的，所以sql语句不能过长

- **说一个mysql的漏洞**

  mysql有一个身份验证绕过漏洞，他在验证密码是否正确时，存在错误处理返回一个判定密码正确的非0值这样一种情况，就是在知道用户名的前提下，只要一直尝试，就可以强行登录mysql，利用方法我知道两个

  一个是metasploit框架的mysql_authbypass_hashdump模块攻击，也可以写一个bash小脚本

  for i in `seq 1 1000`; do mysql –u root –P3306 –password=bad –h 192.168.137.211 2>/dev/null; done

  通过随便写一个密码去试指定服务器指定用户的mysql服务，忽略所有错误信息，重复试1000次

- **站库分离的判断方式**

  1、网络连接状态：查看端口连接状态，如果连接着别人的数据库端口，则本机可能就是web网站程序服务器

  2、数据库配置文件：数据库ip不是127.0.0.1，localhost，而是其他ip，也可能是站库分离

  3、通过内置库，函数，表

  mysql可以select @@hostname查询服务端主机名

  在默认数据库information_schema的processlist可以获取数据库登录用户名，主机和端口号

  sqlserver的话，select HOST_NAME查看服务端主机名称

  select @@SERVERNAME 查看客户端主机名和端口