网 络 安 全

——Web应用攻击

杭州师范大学信息科学与技术学院

刘雪娇　邮箱：liuxuejiao0406@163.com

本章节要求

- 关注国内外最新Web安全事件
- 了解Web应用攻击的相关概念和原理
- 了解Web应用的体系架构，理解Web安全威胁
- 学习典型的Web攻击方式

第七章 Web应用攻击

7.1 Web安全形势

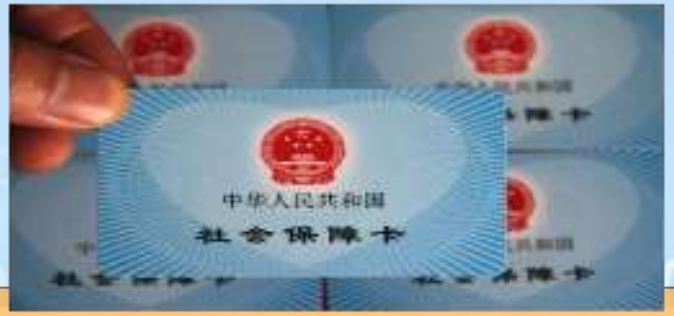
Web安全事件

2018年3 月10 名伊朗黑客在入侵美国等国家大学网站，盗取价值 34 亿美元的 31TB 资料。

2017年2月俄罗斯黑帽黑客Rasputin利用SQL注入漏洞，黑掉了60多所大学和美国政府机构的系统。

2016年10月软件Mirai控制的僵尸网络对美国域名服务器管理服务供应商Dyn发起DDoS攻击。

2015年11月Learning Lodge网站客户资料曾遭到未经授权者入侵，11月24日发现资料外泄。



Web安全形势

新浪微博出现了一次比较大的**XSS攻击**事件。

大量用户自动发送诸如：“郭美美事件的一些未注意到的细节”，“建党大业中穿帮的地方”等微博和私信，并自动关注一位名为hellosamy的用户。

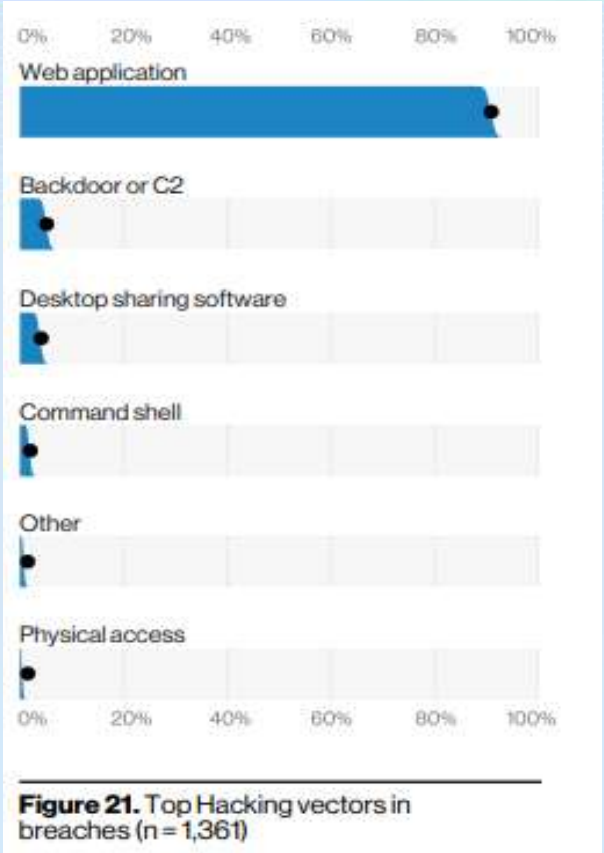
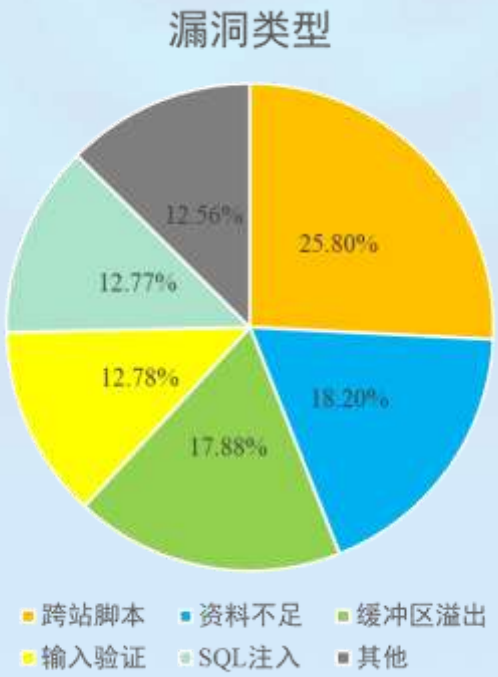
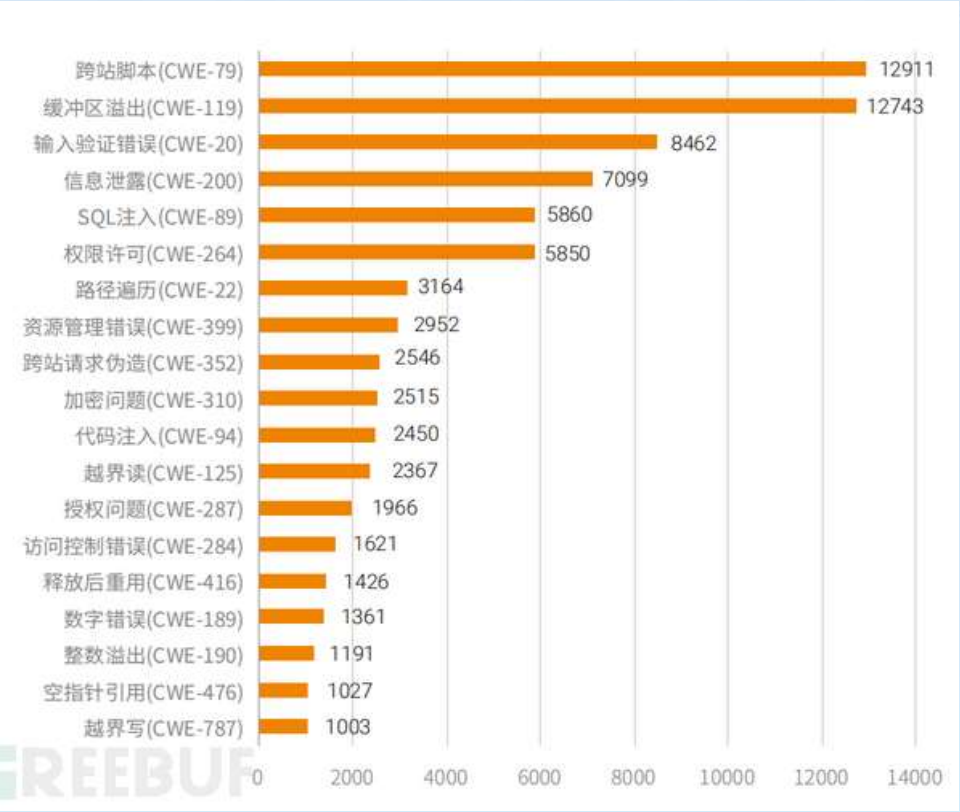


Figure 21. Top Hacking vectors in breaches (n = 1,361)

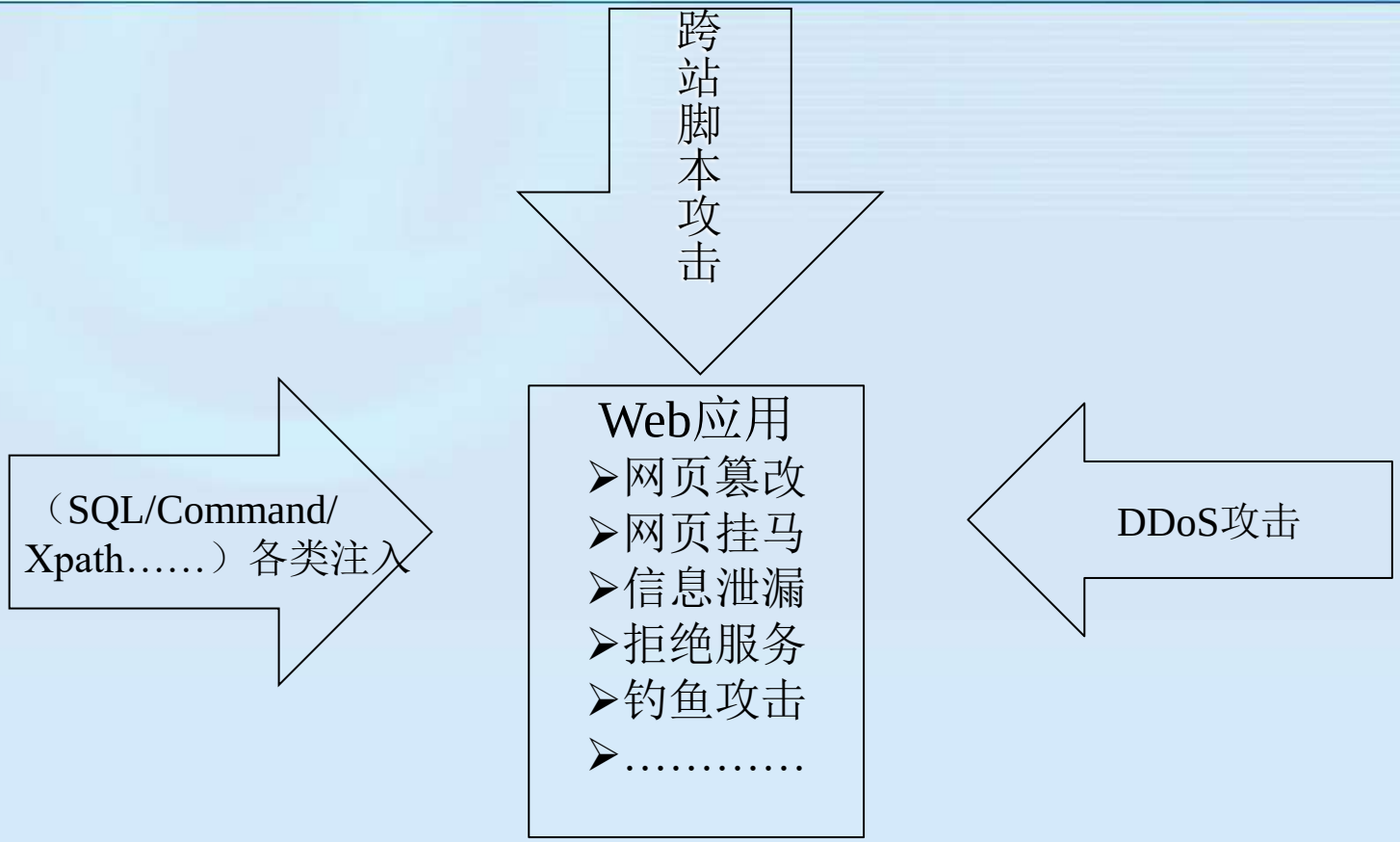
2020年5月，Verizon发布了《数据泄露调查报告》（简称DBIR）报告指出黑客攻击的目标中超过80%是Web应用

跨站脚本 (CWE-79) 类型的漏洞数量以 12911 条占据第一。





Web应用攻击

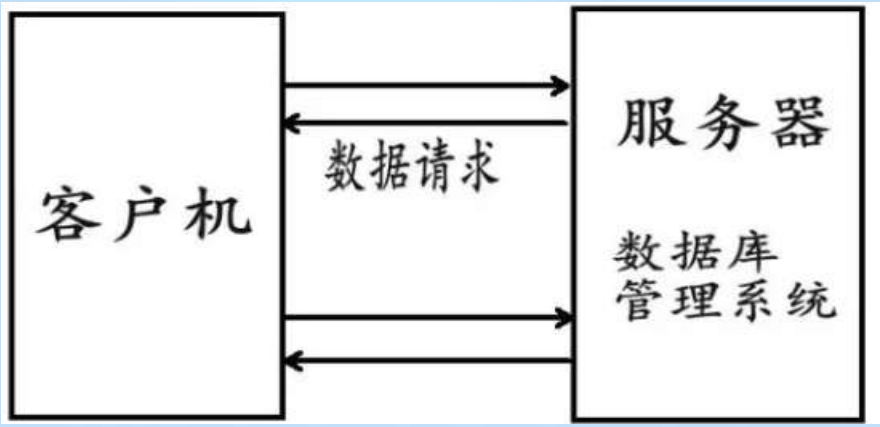

跨站脚本攻击
（SQL/Command/ Xpath……）各类注入
Web应用
网页篡改
网页挂马
信息泄漏
拒绝服务
钓鱼攻击
……
DDoS攻击

Web安全影响严重

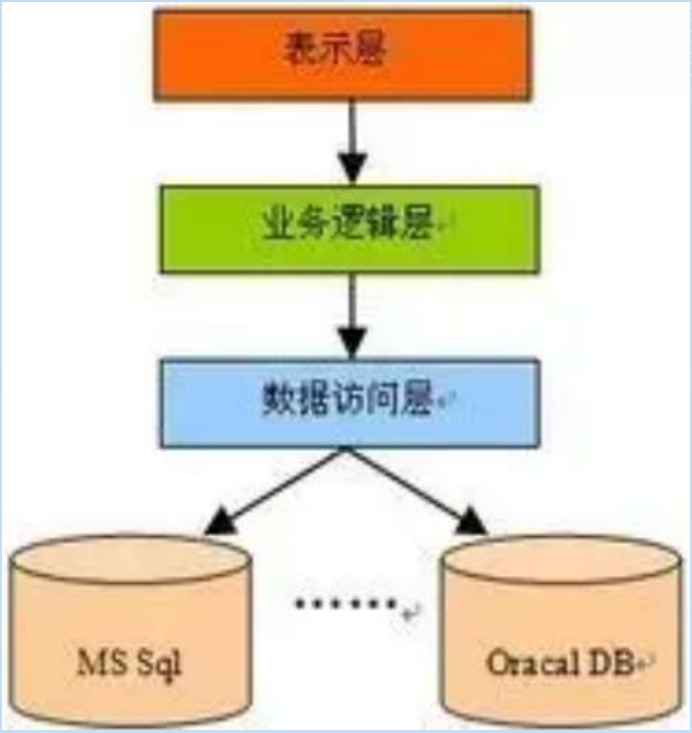
- 安全漏洞事件
  - 攻击得以奏效的重要原因之一
  - 恶意程序传播、感染的重要内因
- 网页挂马事件
  - 对广大网民危害较大
  - 恶意程序广泛传播的重要手段
- 恶意程序事件
  - 造成网站或用户敏感信息泄漏
- 网页篡改事件
  - 使网站形象和声誉受损
- 拒绝服务攻击事件
  - 直接导致网站业务损失
- 网页仿冒事件
  - 对用户的财产安全构成直接威胁

Web应用安全

Web应用程序是一种用浏览器支持的语言所编写或者能够在浏览器控制的环境中运行的软件形态，有C/S、B/S两种模式。

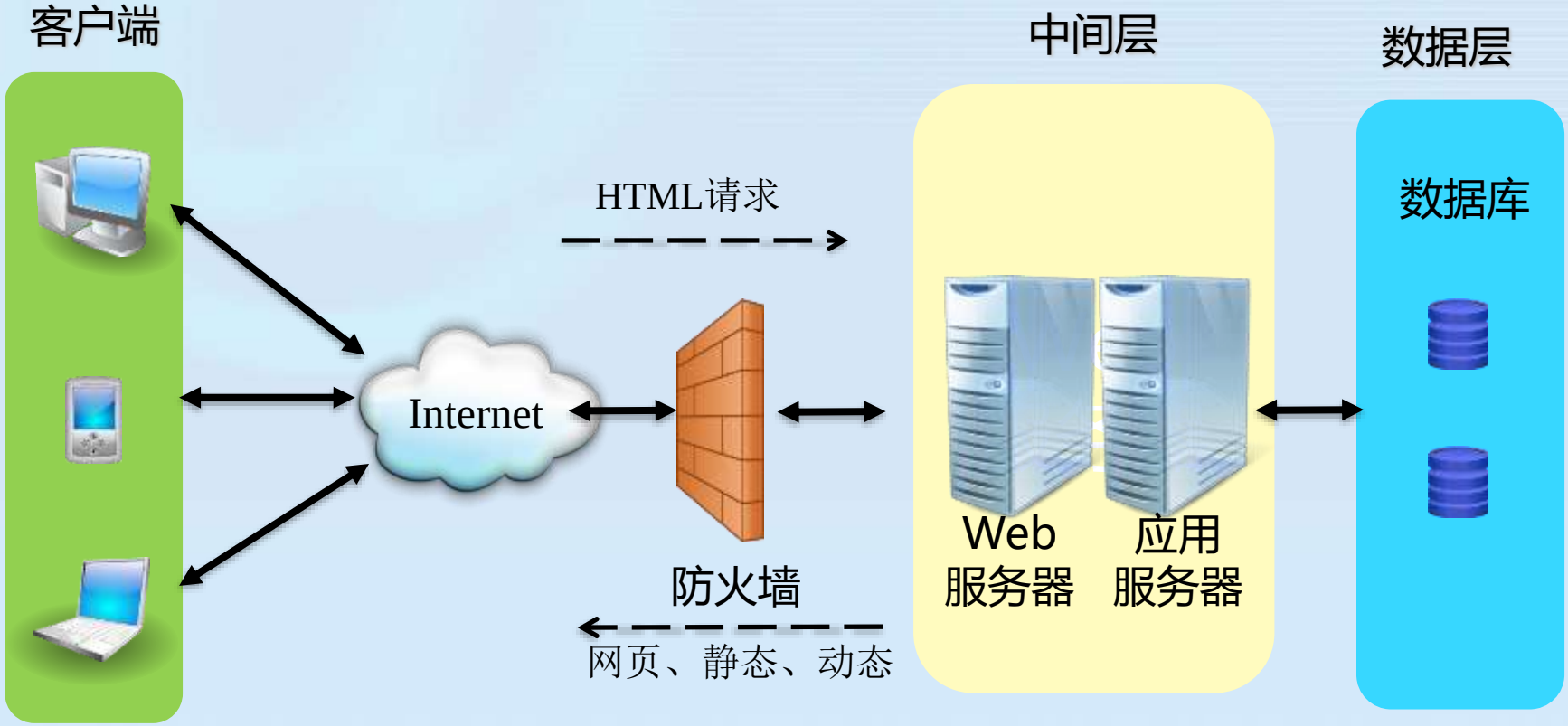


客户端（Client）&服务器（Server）



浏览器（Browser）&服务器（Server）

Web应用体系架构



Web相关技术

网站相关技术	举例
通信协议	HTTP、HTTPS
WEB语言	HTML、CSS、CGI、ASP、JSP、PHP……
WEB Server	IIS、Apache、Tomcat、WebLogic……
操作系统	Windows、Linux、FreeBSD、Solaris……
数据库	SQL Server、MySQL、DB2、Sybase……
发布和备份	FTP、CMS、SFTP

Web应用安全威胁

- **针对浏览器和终端用户：**

  以浏览器渗透攻击为核心的网页木马、phishing网站钓鱼等。

- **针对传输网络的网络协议：**

  如http明文传输的窃听，假冒身份攻击、拒绝服务攻击等。

- **系统层：**

  宿主的操作系统有着远程渗透攻击和本地渗透攻击的威胁。

- **Web服务器软件：**

  Web服务器也存在着安全漏洞，攻击者可以利用这点对服务器进行攻击。

- **Web应用程序：**

  程序员在实现Web应用程序的时候，会因为各种原因导致出现安全漏洞，被攻击者利用，包括SQL注入攻击、XSS跨站脚本攻击。

- **Web数据：**

  Web应用程序储存的数据、浏览器输入的数据都存在着被窃取、篡改的威胁。

Web攻击分类

web攻击分类	攻击方式
客户端	1.跨站脚本攻击（XSS）； 2.跨站点请求伪造（CSRF）； 3.点击劫持（Click Jacking）；
应用服务端	1.网站钓鱼； 2.邮件钓鱼； 3.XSS钓鱼；
服务器端	1.SQL注入攻击； 2.文件上传漏洞； 3.应用层拒绝服务攻击(DDOS)；

XSS跨站脚本攻击

定义：跨站脚本（Cross-Site Scripting，简称为XSS或跨站脚本攻击）是一种针对网站应用程序的安全漏洞攻击技术，是代码注入的一种。

恶意攻击者往Web页面里注入恶意Script代码，当用户浏览这些网页时，就会执行其中的恶意代码。

攻击目标：浏览器端程序

攻击方式：身份窃取、盗取cookie信息、会话劫持、蠕虫、钓鱼等。

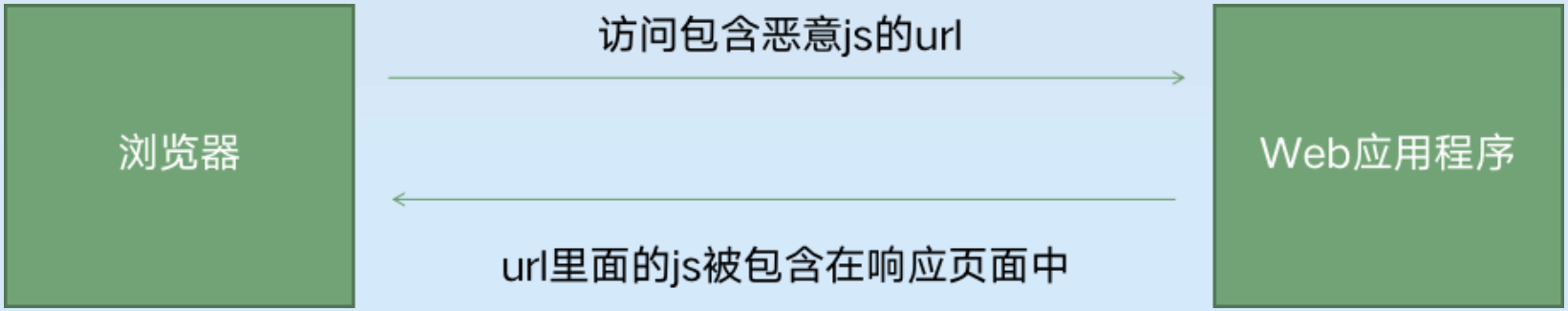
XSS的分类

XSS根据**效果不同**可以分为三类：

类型	说明
反射型XSS	当用户访问一个带有XSS代码的URL请求时，服务器端先接受处理后再发送代码到浏览器，浏览器解析数据后，最终造成XSS漏洞。也就是说，黑客往往需要诱使用户点击链接，才能成功，也叫“非持久型XSS”
存储型XSS	服务器端接收并存储攻击者的XSS代码，当用户再次访问某个页面时浏览器才会解析代码造成XSS攻击。也就是会把的输入数据“存储”在服务器端，具有很强的稳定性，也叫持久型XSS。
DOM XSS	DOM可以为JS提供对象的位置，进而改变某个对象或者页面。通过修改网页的DOM节点形成的XSS称之为DOM Based XSS

XSS的分类——反射型XSS

- 反射型XSS只是简单地将用户输入的数据直接或未经过完善的安全过滤就在浏览器中进行输出，导致输出的数据中存在可被浏览器执行的代码数据。由于此种类型的跨站代码存在于URL中，所以黑客通常需要通过诱骗或加密变形等方式，将存在恶意代码的链接发给用户，只有用户点击以后才能使得攻击成功实施。

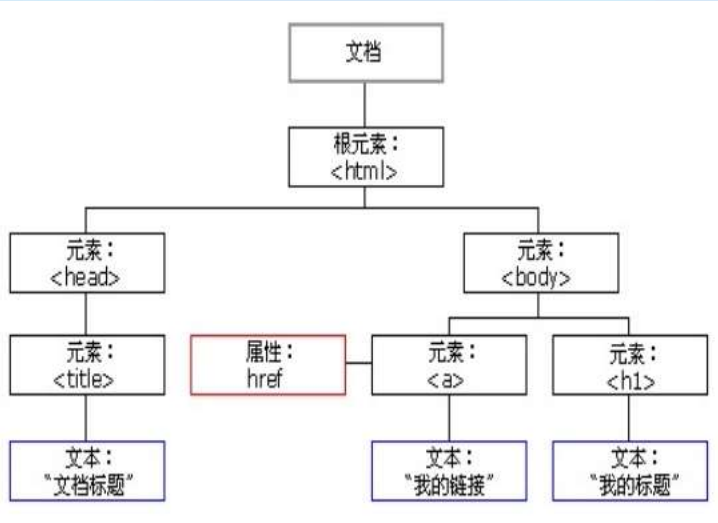


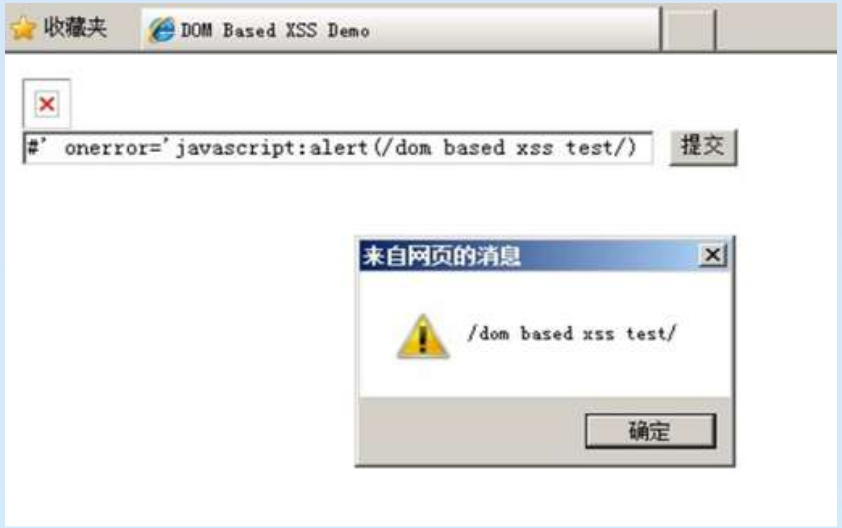
XSS的分类——存储型XSS

- 存储型XSS是指Web应用程序会将用户输入的数据信息保存在服务端的数据库或其他文件形式中，网页进行数据查询展示时，会从数据库中获取数据内容，并将数据内容在网页中进行输出展示，因此存储型XSS具有较强的稳定性。

XSS的分类——DOM型XSS

- DOM全称Document Object Model，使用DOM可以使程序和脚本能够动态访问和更新文档的内容、结构及样式。
- 基于DOM的XSS是通过修改页面的DOM节点数据信息而形成的XSS跨站脚本攻击。





XSS的危害

- 窃取用户cookie
- DDoS攻击
- 网页挂马
- 提升用户权限
- 获取客户端信息
- 传播跨站脚本蠕虫
- 劫持用户（浏览器）会话
- 强制弹出广告页面、刷流量
- 任意篡改页面信息、删除文章

DVWA平台XSS

Reflected:
<script>alert('security')</script>　　(low)
<s<script>cript>alert('security')</script>　　(medium)
<img src=1 onerror=alert(1)>　　(high)

Storage:
<script>alert('security')</script>　　(low)
<s<script>cript>alert(‘security’)</script> 改maxlength　(medium)
<img src=1 onerror=alert(1)>改maxlength　　(high)

DOM:
<script>alert(document.cookie)</script>　　(low)
</option></select><img src=1 onerror=alert(document.cookie)>　　(medium)
方法一：#</option></select><img src=1 onerror=alert(document.cookie)>　　(high)
方法二：#</option></select><body onload=alert(document.cookie)>

XSS的防御

1. 检测XSS

- 手工检测XSS：对敏感字符（< > ” ’ ）、函数、onClick等进行转义
- 全自动检测：专业XSS扫描工具

2.修复XSS漏洞

- 检查所有用户可控输入与输出点，针对性地过滤、转义
- 对重要Cookie字段使用HttpOnly参数，解决Cookie劫持攻击

XSS钓鱼

攻击者可构造html/css/javascript代码,制作一个与原网站风格相似的登录界面,或把一些html元素进行移动/覆盖/遮挡, 使得原来的功能被篡改。

XSS钓鱼：为了窃取用户密码，攻击者将XXS与“钓鱼”相结合，利用JS在当前网页“画出”一个伪造登录框，当用户输入用户名和密码后，其密码就会被发送到攻击者的服务器上。

网站钓鱼：钓鱼网站一般都会使用具有欺骗性的域名，通过各种文字欺骗用户。

邮件钓鱼：钓鱼邮件是垃圾邮件的一种，它比广告邮件更加具有针对性。

SQL注入攻击

定义：攻击者利用设计上的漏洞，将SQL命令插入到Web表单提交或输入域名或页面请求的查询字符串，最终欺骗服务器执行恶意的SQL命令。

攻击对象：服务器端的数据库

产生原因：

- SQL语句被拼接
- 转义字符处理不当
- 后台查询语句处理不当
- 动态生成SQL语句时没有对用户输入的数据进行验证

SQL注入攻击典例

假设一个用户登录程序中从user表中匹配用户名密码，原sql语句为select * from user where name='' and pwd=''

➢输入用户名为：aaa;drop table user

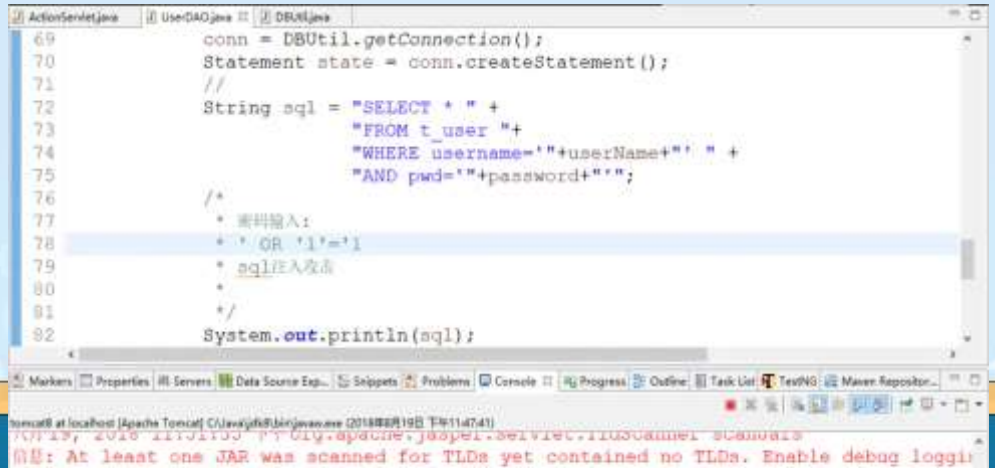
sql语句：select * from user where name='aaa';drop table user

结果：数据库表被删除。

➢输入密码为：'or '1'='1 （**SQL万能密码**）

sql语句：select * from user where name = ‘asdad’ AND pwd= ‘’ or ‘1’ = ‘1’ 因为 ‘1’ = ‘1’ 永远成立。

结果：返回user表里的全部账户名和密码。



SQL注入漏洞分类

- **注入漏洞分类：**

  数字型注入、字符型注入、Cookie注入、POST注入、延时注入等

- **常见数据库注入：**

  Oracle 11g、MySQL 5.1、SQL Server

- **注入工具：**

  SQLMap、Pangolin

典型攻击流程


判断网站是否是动态网站
判断是否有注入点
手工或者通过工具查看数据库信息
获取敏感信息，如管理员账号密码
破解管理员密码
登录网站后台
如果权限比较高，可以写入WebShell

防范SQL注入攻击

- **严格的数据类型：**防御数字型注入只需在程序中严格判断数据类型，使用is_numeric()、ctype_digit()等函数判断数据类型即可。
- **特殊字符转义：**只需要建一个相应的数据库编码器，然后调用ESAPI.encoder().encodeForSQL()方法即可对字符串编码。
- **使用预编译语句：**在Java中，提供了三个接口与数据库交互，分别是Statement、CallableStatement和PreparedStatement。
- **框架技术：**Java、C#、PHP等语言都有自己的框架，也发展的越来越成熟，而且有较高的安全性。
- **存储过程：**是在大型数据库系统中，一组为了完成特定功能或经常使用的SQL语句集，经编译后存储在数据库中。

7.5 文件上传漏洞攻击

定义：如果Web应用没有严格限制用户上传的文件后缀以及文件类型，导致允许攻击者向某个可通过 Web访问的目录上传任意PHP文件，并能够将这些文件传递给PHP解释器，就可以在远程服务器上执行任意PHP脚本，获得执行服务器端命令的能力。这也被称为Webshell问题。

攻击目标：服务器端程序

产生原因：

- 对于上传文件的MIMETYPE没有做检查
- 对于上传文件的后缀名（扩展名）没有做较为严格的限制
- 权限上没有对于上传的文件目录设置不可执行权限
- 对于web server上传文件或者指定目录的行为没有做限制

文件上传带来的安全问题

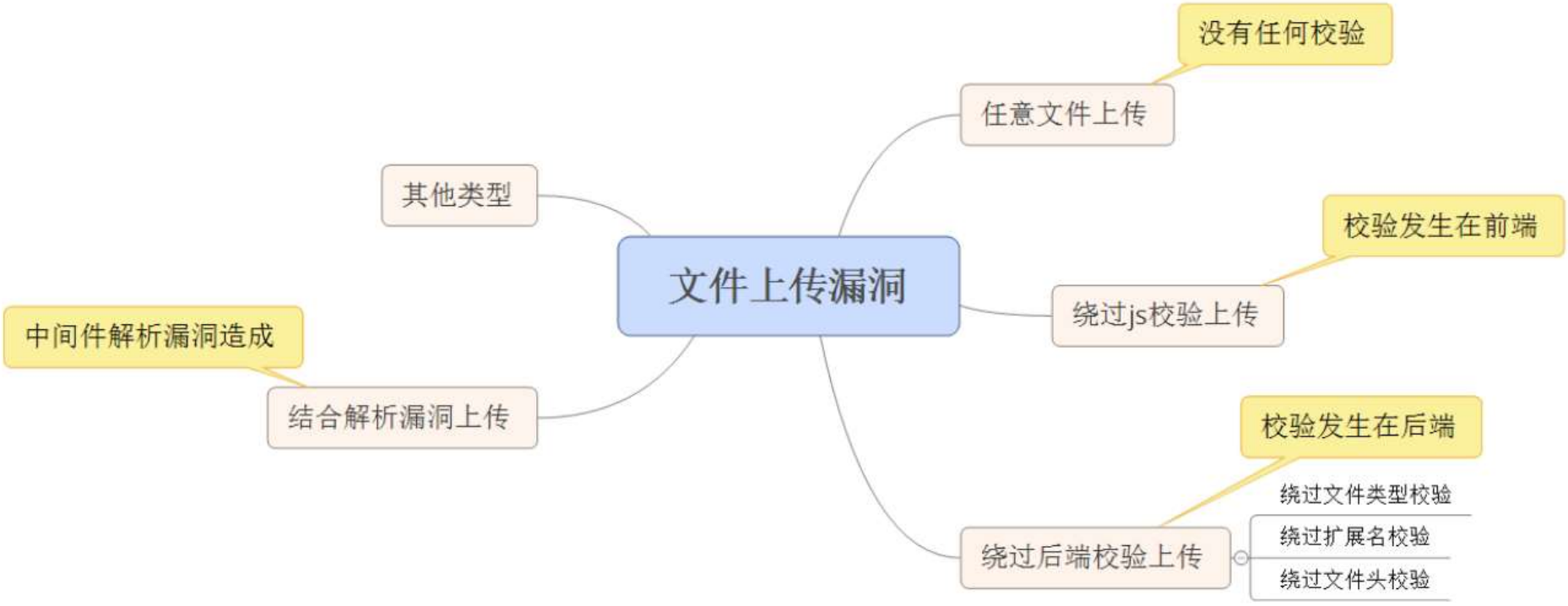
- 上传文件是Web脚本语言，服务器的Web容器解释并执行了用户上传的脚本，导致代码执行。
- 上传文件是Flash的策略文件crossdomain.xml,黑客用以控制Flash在该域下的行为(其他通过类似方式控制策略文件的情况类似)。
- 上传文件是病毒、木马文件，黑客用以诱骗用户或者管理员下载执行。
- 上传文件是钓鱼图片或为包含了脚本的图片，在某些版本的浏览器中会被作为脚本执行，被用于钓鱼和欺诈。

文件上传漏洞

- 文件上传漏洞可能存在于整个文件上传流程的每一环。

- 产生原因
- 文件上传时检查不严
- 文件上传后修改文件名时处理不当
- 使用第三方插件时引入

文件上传漏洞分类


文件上传漏洞
任意文件上传
没有任何校验
绕过js校验上传
校验发生在前端
绕过后端校验上传
校验发生在后端
绕过文件类型校验
绕过扩展名校验
绕过文件头校验
结合解析漏洞上传
中间件解析漏洞造成
其他类型

防范文件上传漏洞攻击

➢客户端检测：

客户端使用JS检测，在文件未上传时，就对文件进行验证。

➢服务器端检测：

检测文件的MIME类型，文件扩展名是否合法，或检测文件中是否嵌入恶意代码。

➢过滤方法：

白名单与黑名单扩展名过滤、文件类型检测（不仅仅通过后缀名判断，还要通过判断文件起始的几个字节来判断）、文件重命名等操作。

跨站请求伪造攻击

CSRF(Cross-site request forgery)，跨站请求伪造。

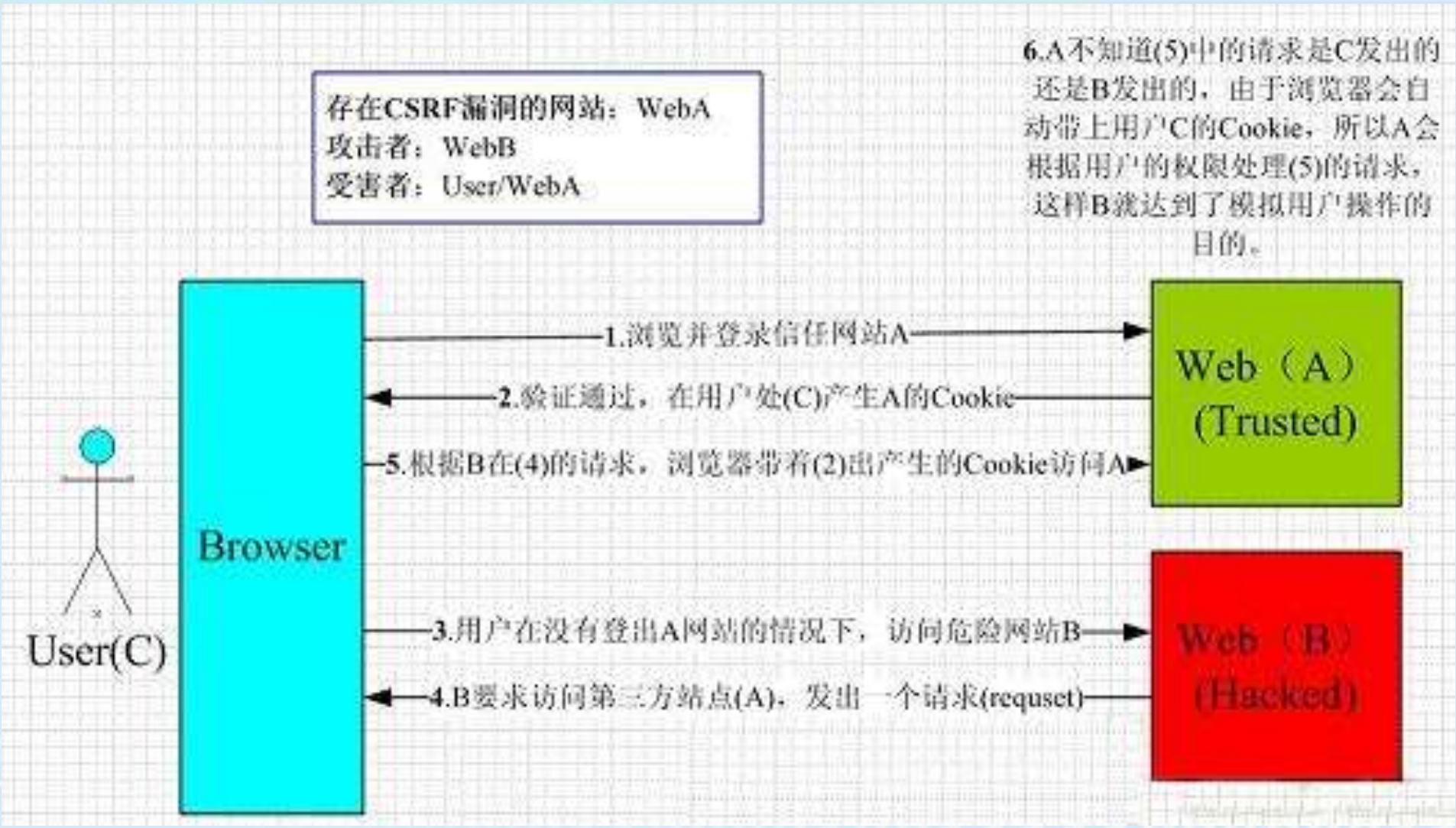
攻击者利用会话机制的漏洞，引导用户单击恶意网页，其中的恶意代码欺骗用户的浏览器去访问一个曾经认证过的网站并运行一些操作从而引发攻击。

攻击结果：冒充用户执行特定操作。

攻击原理：利用Web中用户身份验证的漏洞：简单的身份验证只能保证请求发自某个用户的浏览器，却不能保证请求本身是用户自愿发出的。

可以这么理解CSRF攻击：攻击者盗用了你的身份，以你的名义进行某些非法操作。CSRF能够使用你的账户发送邮件，获取你的敏感信息，甚至盗走你的财产。

CSRF示意图


存在CSRF漏洞的网站：WebA
攻击者：WebB
受害者：User/WebA
6.A不知道(5)中的请求是C发出的还是B发出的，由于浏览器会自动带上用户C的Cookie，所以A会根据用户的权限处理(5)的请求，这样B就达到了模拟用户操作的目的。
Browser
User(C)
1.浏览并登录信任网站A
2.验证通过，在用户处(C)产生A的Cookie
5.根据B在(4)的请求，浏览器带着(2)出产生的Cookie访问A
Web（A）
(Trusted)
3.用户在没有登出A网站的情况下，访问危险网站B
4.B要求访问第三方站点(A)，发出一个请求(requset)
Web（B）
(Hacked)

防范CSRF攻击

- **验证 Referer**

  HTTP头中的Referer字用以标明请求来源于哪个地址，判断该请求是否为第三方网站发起的。

- **验证码**

  CSRF的攻击过程往往是用户所不知情的，而验证码强制用户与应用交互才能最终完成请求。能够很好的遏制CSRF的攻击。

- **Token**

  服务器下发一个随机Token，每次发起请求时将Token携带在URL上，服务器验证 Token 是否有效。

- **Cookie保护**

  对Cookie设置SameSite\HttpOnly属性。SameSite属性表示Cookie不随着跨域请求发送，HttpOnly让恶意网站无法通过脚本获取到Cookie。

1.手工审查web应用程序结构和源代码：

➢静态和动态生成的页面：

一般情况下静态页面不会遭受注入攻击，但是其中很可能有一些有价值的信息。比如表单隐藏字段和注释，都可能会包含着注入用户口令的一些敏感信息。动态页面的价值则更大，攻击者会探查所使用的脚本编程语言、页面命名规则、参数名称、类型与含义等。

➢目录结构：

攻击者会对管理员目录、备份目录、数据目录等进行查看，来确定这些目录是否存在，并且是否存在可以利用的不安全的配置，这些不安全的配置是否可以用来搜索可利用的文件。

- **辅助性文件：**

  Web应用程序有着一系列辅助性文件，如css级联样式表、xml样式表等，手动查看这些文件可能会得到有用的信息。

- **输入表单：**

  表单时Web应用程序用来接受用户输入的主要途径，手动检查源代码可以发现一些关键表单，进而获取一些关键数据，然后这些数据可以帮助攻击者尝试绕过表单的正常处理。

- **查询参数字符串**

  Web应用程序很容易收集到一些动态页面文件的查询参数字符串，这些字符串可以被利用。

2.Web应用程序安全评估与漏洞审查：

对web应用程序的攻击主要集中在身份验证、会话管理、数据库操作、输入合法性检查。纯手工的评测自然也是效率很低。辅助工具主要有：浏览器插件；免费工具集，如Fiddler，WebScarab等；商业web应用安全评估系统和扫描器。

本讲小结

Web安全形势严峻：

- Web安全事件频发
- Web漏洞占比上升
- Web威胁类型

Web应用结构与潜在威胁

常见的Web应用攻击利用方式与防范方法：

XSS、SQL注入、文件上传、CSRF...

Web应用攻击实验简介

实验任务一：XSS跨站脚本攻击实验

实验原理： XSS跨站脚本攻击的目标是浏览器端程序，其利用Web应用对用户输入内容过滤不足的漏洞，在Web页面里插入恶意代码，当用户浏览该页面时，嵌入其中的恶意代码就会被执行，从而带来危害。

实验任务二：SQL注入攻击实验

实验原理： SQL注入攻击一般针对服务器端的数据库，其利用Web应用程序对输入代码过滤不足的漏洞，使用户输入影响SQL查询语句的语法，从而带来危害。

实验主要原理

实验任务三：文件上传攻击实验

实验原理：大部分的网站和应用系统都有上传功能，而程序员在开发任意文件上传功能时，并未考虑文件格式后缀的合法性校验或者是否只在前端通过js进行后缀检验。这时攻击者可以上传一个与网站脚本语言相对应的恶意代码动态脚本，例如(jsp、asp、php、aspx文件后缀)到服务器上，从而访问这些恶意脚本中包含的恶意代码，进行动态解析最终达到执行恶意代码的效果，进一步影响服务器安全。

实验相关工具

DVWA(Damn Vulnerable Web Application)

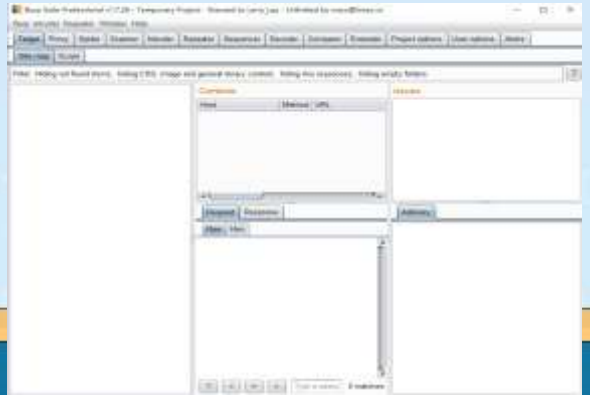
这是一个用来进行安全脆弱性鉴定的PHP/MySQL/Web应用，旨在为安全专业人员测试自己的专业技能和工具提供合法的环境，帮助Web开发者更好的理解Web应用安全防范的过程。

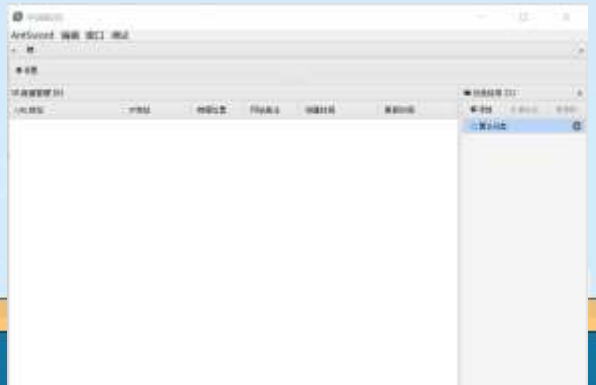
Burp Suite

这是一个用于攻击Web应用程序的集成平台。它包含了许多工具，并为这些工具设计了许多接口，以促进加快攻击应用程序的过程。

中国蚁剑（ant Sword）

中国蚁剑是一款开源的跨平台网站管理工具，它主要面向于合法授权的渗透测试安全人员以及进行常规操作的网站管理员。

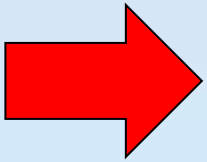


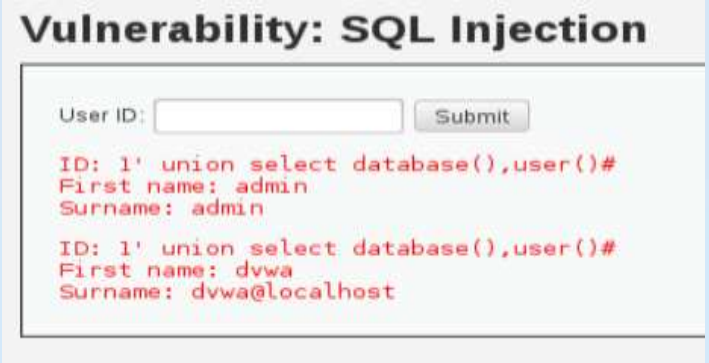


实验相关命令

猜测字段数，输入“1’ union select 1,2#”

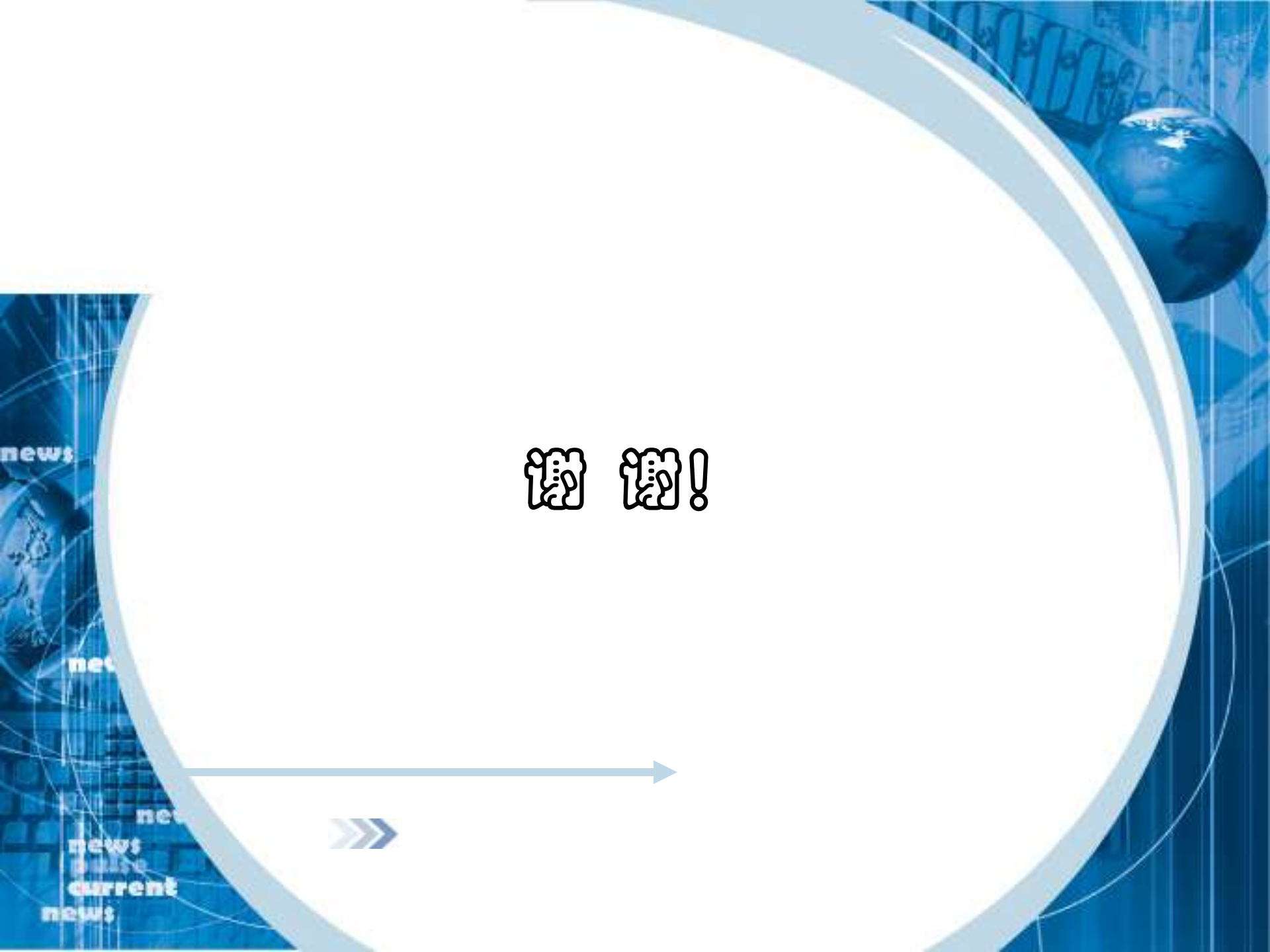
在Union注入时，例如一个网站的参数传递执行的查询有3个字段，很可能其中的1或2个字段的查询结果是会返回到前端，这时候我们利用一个简单的select 1,2,3，根据显示在页面的数字，就知道哪个是会返回到前端的，然后在把对应的数字改成我们想要查询的内容即可。





第一步：select 1,2#返回First name和Surname，说明1，2位有效

第二步：把1，2位改成database()和user(),完成查询目标


谢 谢!