# web安全

私密性：个人资料不会被非法获取

可靠性：个人资料不会被篡改

# XSS攻击

**跨站脚本攻击（Cross Site Scripting）**

本网站运行了其他网站的脚本

现代游览器会自动筛选xss攻击，另外一些框架也会自动转义”<””>”

页面中所有能填写信息的地方都可以填写<script></script>

可以获取页面数据，获取cookie，劫持页面逻辑，发送请求

反射型：通过url等执行

存储型：保存到数据库中

可能产生xss攻击的地方：所有用户可输入的位置

直接通过js添加进页面中，信息是不会产生xss攻击的，必须通过服务器返回，页面数据重新加载才能实现xss攻击

**攻击方式**

**URL**

http://127.0.0.1?a=<script>alert(1)</script>

**属性：**

<img src=”{{src}}” />

<img src=”1” onload=”alert(1)” /> # 蓝色用户精心编制的代码

**JavaScript：**

<script>

let data=”{{data}}”

let data=”hello”;alert(1);””;

</script>

**富文本：**

直接填写**脚本代码**

**<a href=\”javascript.alert(1)\”>你好</a>**

### 应对

1\转义单引号、双引号、空格、小于号、大于号

2\富文本过滤，使用富文本标签

## csp

Content Security Policy(内容安全策略）

用于指定哪些内容可执行

Content-Security-Policy:default-src 'none'

X-XSS-Protection: 1; mode=block

# csrf攻击

跨站请求伪造 （Cross Site Request Forgy）

用户打开一个网站，却在另一个网站做了一个动作

例如，一个不验证用户信息的接口。

禁止第三方

same-site,

添加token

# cookis

前端数据存储，前端可修改cookie，遵守同源策略，只有协议，端口，域名全部一致时，才算同源

cookie特性，域名，有效期，路径（根据路径），http-only,secure

cookie应用存储个性化的标识，

做登录用户的凭证，

通过修改cookie修改，可以登录

通过签名等验证-jwt模式

**cookie与xss的关系**

XSS可偷取cookie，http-only添加，防止cookie被盗

**cookie与CSRF的关系**

CSRF利用用户cookie，通过携带用户cookie发起请求，添加验证码等

## cookie安全案例

cookie使用

# 前端点击劫持

钓鱼网站

点击劫持，将目标网站嵌入到当前网站上，并隐藏目标网站，诱导用户点击

设置验证码

通过设置请求头

X-Frame-Options头，禁止内嵌Deny禁止内嵌，same-origin同一个网站，allow-from允许指定网站

# 五、HTTP传输窃听

原因是http协议是明文传输，被接听后会直接破译，

代理、局域网、wifi、中任何可篡改你的数据

使用https协议

traceroute [www.baidu.com](http://www.baidu.com) linux

tracert [www.baidu.com](http://www.baidu.com) windows

\* 表示服务器未按预期返回数据，但把数据进行了转发

## 部署https网站

证书申请

curl <https://get.acme.sh> | sh

安装位置是在/root/.acme.sh

执行命令acme.sh

server{

listen 443 ssh http2;

ssl\_certificate 目录

ssl\_certificate\_key 目录

}

# 六、密码安全

严禁密码明文存储

单向变换

加盐

哈希算法-摘要算法

雪崩效应，无法反推，md5 sha1，sha256

密码传输：https，频率限制（防止密码猜解），前端加密传输-防止用户原始密码泄露

# 七、SQL注入

sqlite 嵌入型设备的数据库

**SQL注入**

sql注入

select \* from table where id=1 or 1=1

查询的关键字上配置一些特殊的语句如 1 or 1=1查询整个库的内容

关闭错误输出

使用orm，使用参数化查询

**NoSql注入防御**

$gt:0

# 上传漏洞

上传脚本后，通过url执行脚本

1. 限制上传文件后缀，
2. 文件类型检查
3. 文件内容检查
4. 程序输出，不执行文件，使用函数读取文件内容
5. 可写与可执行权限互斥
6. 网站权限不使用root用户

# DOS拒绝服务攻击

模拟正常用户，大量占用服务资源--无法正常

TCP半连接，产生等等

HTTP连接，大量

DNS，攻击域名解析服务器

**DDOS-大规模分布式拒绝服务器攻击**

流量大，分布式，大量肉鸡，代理