登录

Google Bugbounty:点击劫持之DOM型XSS

落花四月 / 2019-09-01 10:20:00 / 浏览数 3418 渗透测试 渗透测试 顶(0) 踩(0)

Google Bugbounty:点击劫持之DOM型XSS

原文链接：https://appio.dev/vulns/clickjacking-xss-on-google-org/

一个鲜为人知的谷歌项目：谷歌危机地图，它的主要任务是帮助人们查找和使用关键的紧急信息。

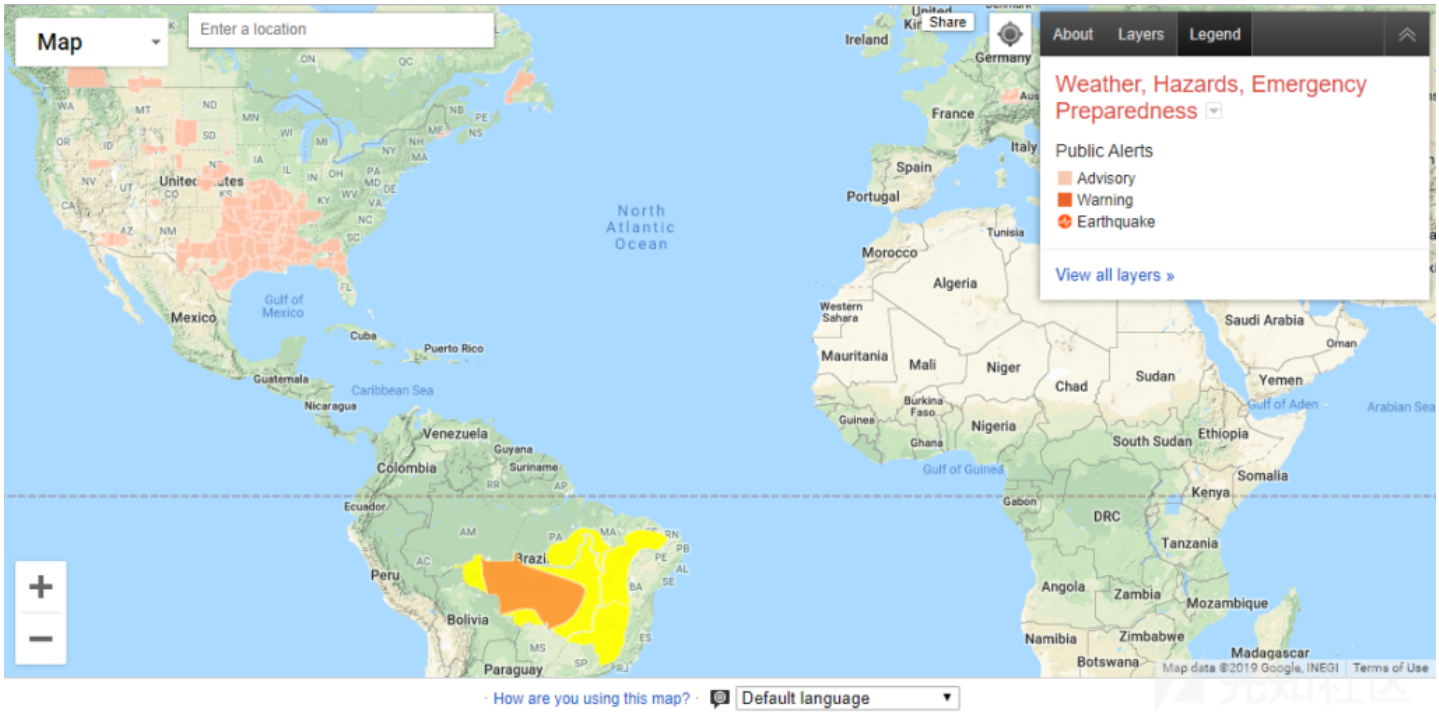
虽然它一直未被Google关闭，但是这个项目很少有人会使用它。

由于它是一个较旧的项目（创建于2012年）并且不经常更新，因此它是寻找漏洞的一个很好的目标。

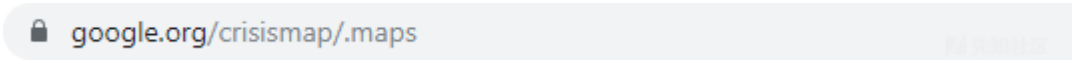
它托管在google.org域名，虽然，该域名的严重程度不如google.com（针对客户端漏洞），但它仍然是Google拥有的域名。

登录

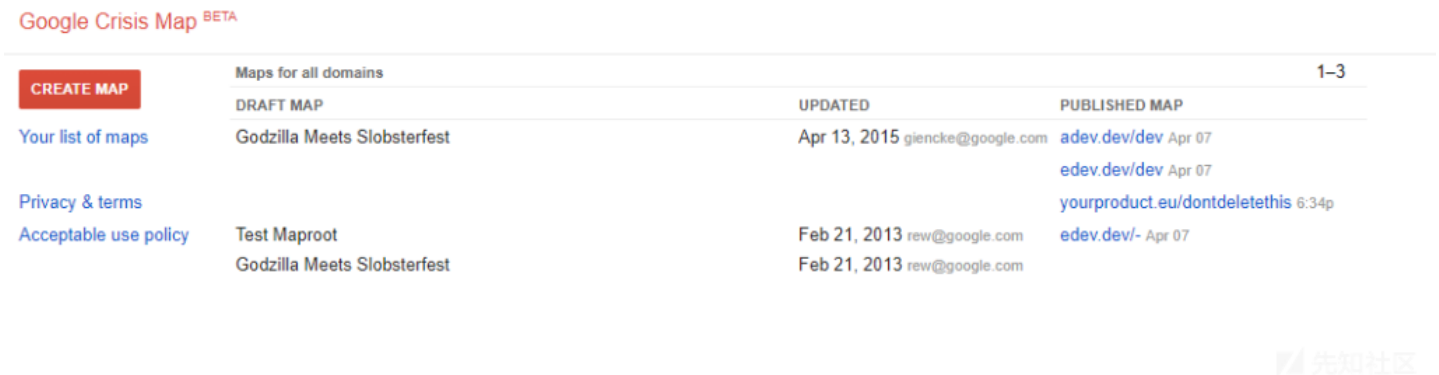
如果你去项目主页，你将会被重定向到地图的默认页面："天气和事件"，这个对我们并没有大多用处，因为我们唯一能做的就是查看地图。



有一种方法可以管理和创建新地图。如果我们在U□□RL的末尾添加.maps，则可以访问它：google.org/crisismap/.maps



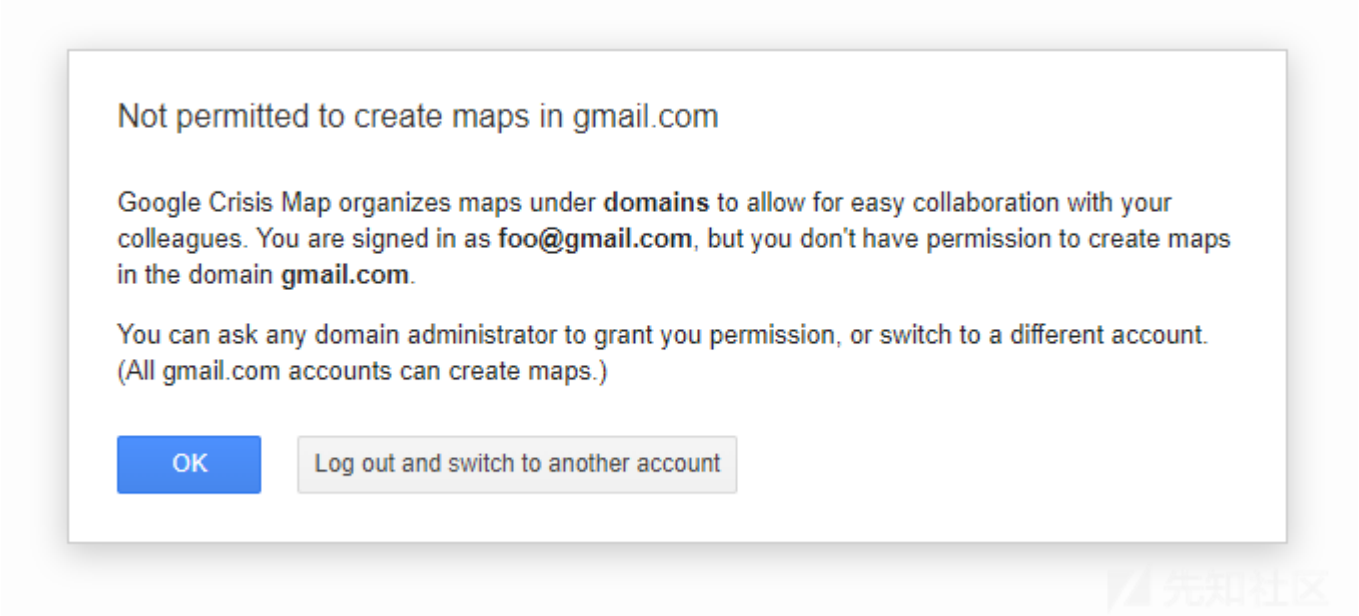
打开此页面后，你需要使用自己的Google帐户登录才能继续。现在你应该看到一个带有地图列表的仪表板，每个帐户都有三个默认地图。



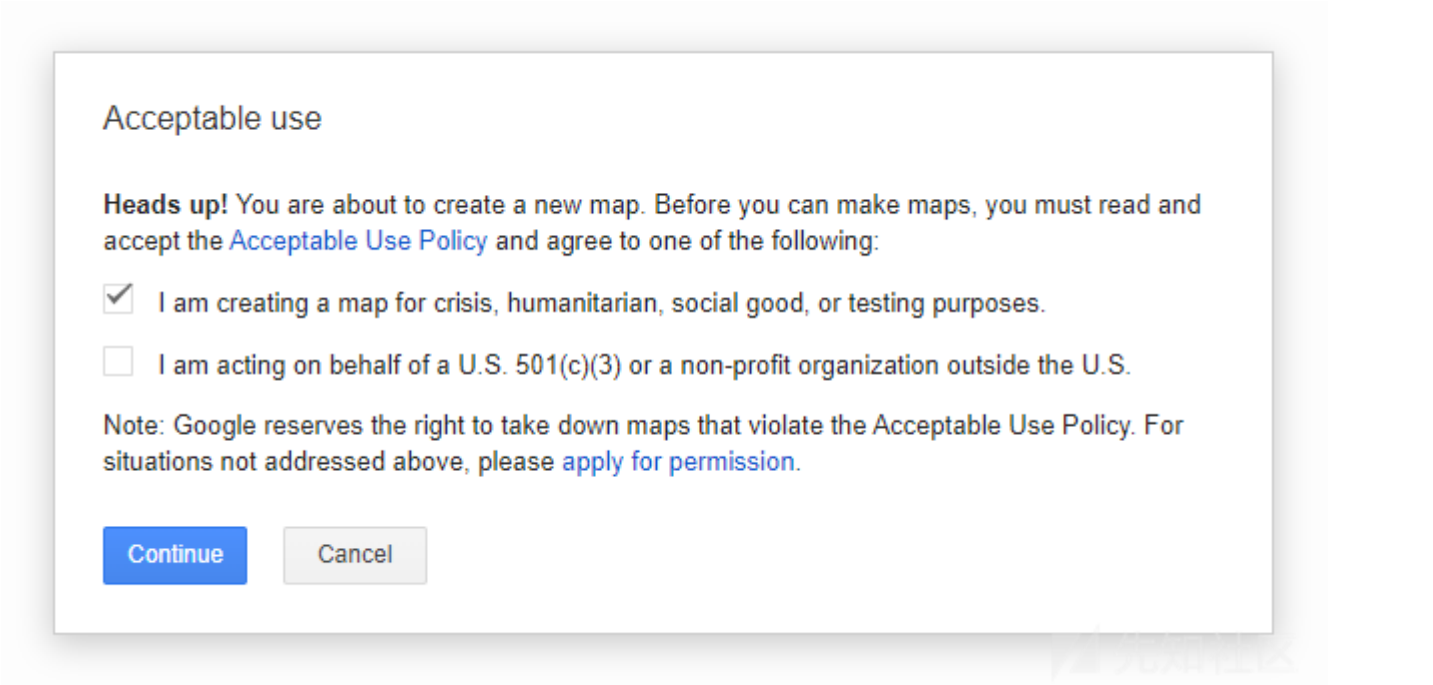
出于某种原因，如果你在自己的账户上发布一个地图，则每个人都会在"已发布地图"字段下的仪表板中看到该地图

创建地图

当你单击红色的"创建地图"按钮，你很可能会看到一条消息，指出gmail.com域不能用于创建新地图。



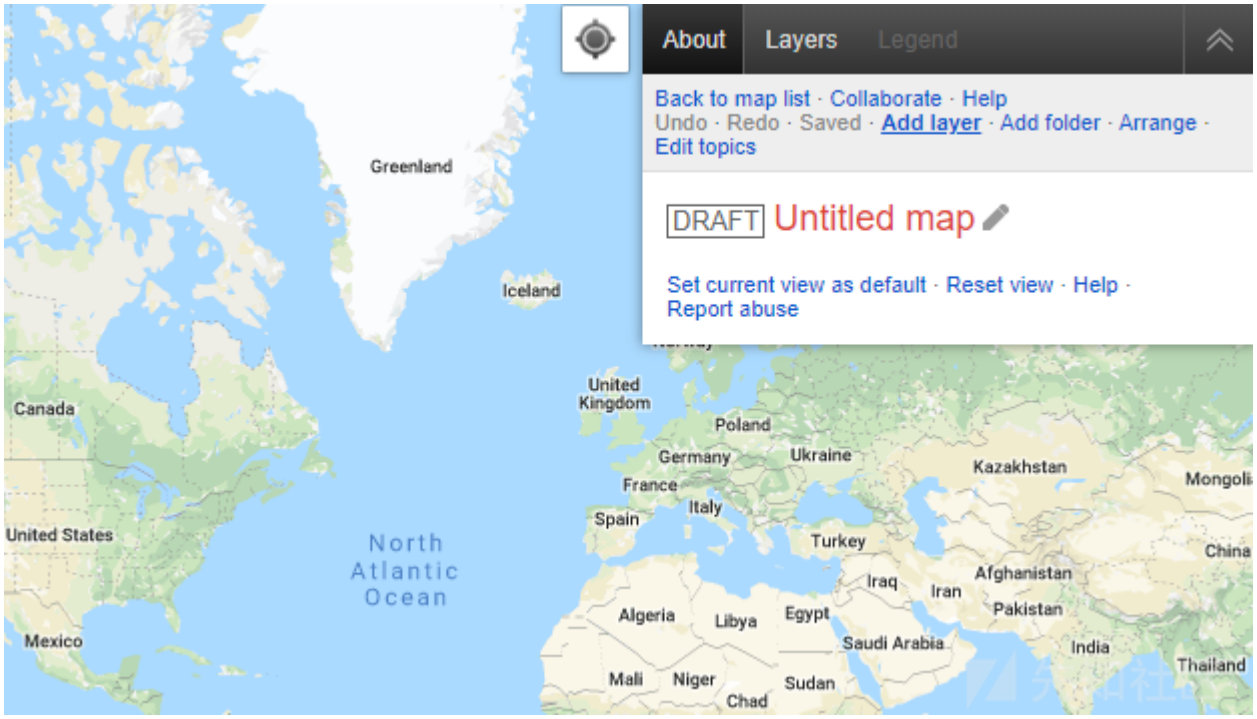
这意味着我们需要使用包含我们自定义的电子邮件登录。我们可以通过使用GSuite帐户或使用gmail.com以外的电子邮件登录来执行创建一个新地图的操作。



单击"继续"按钮后，我们将被重定向到我们可以编辑新创建的地图的页面。

发现XSS

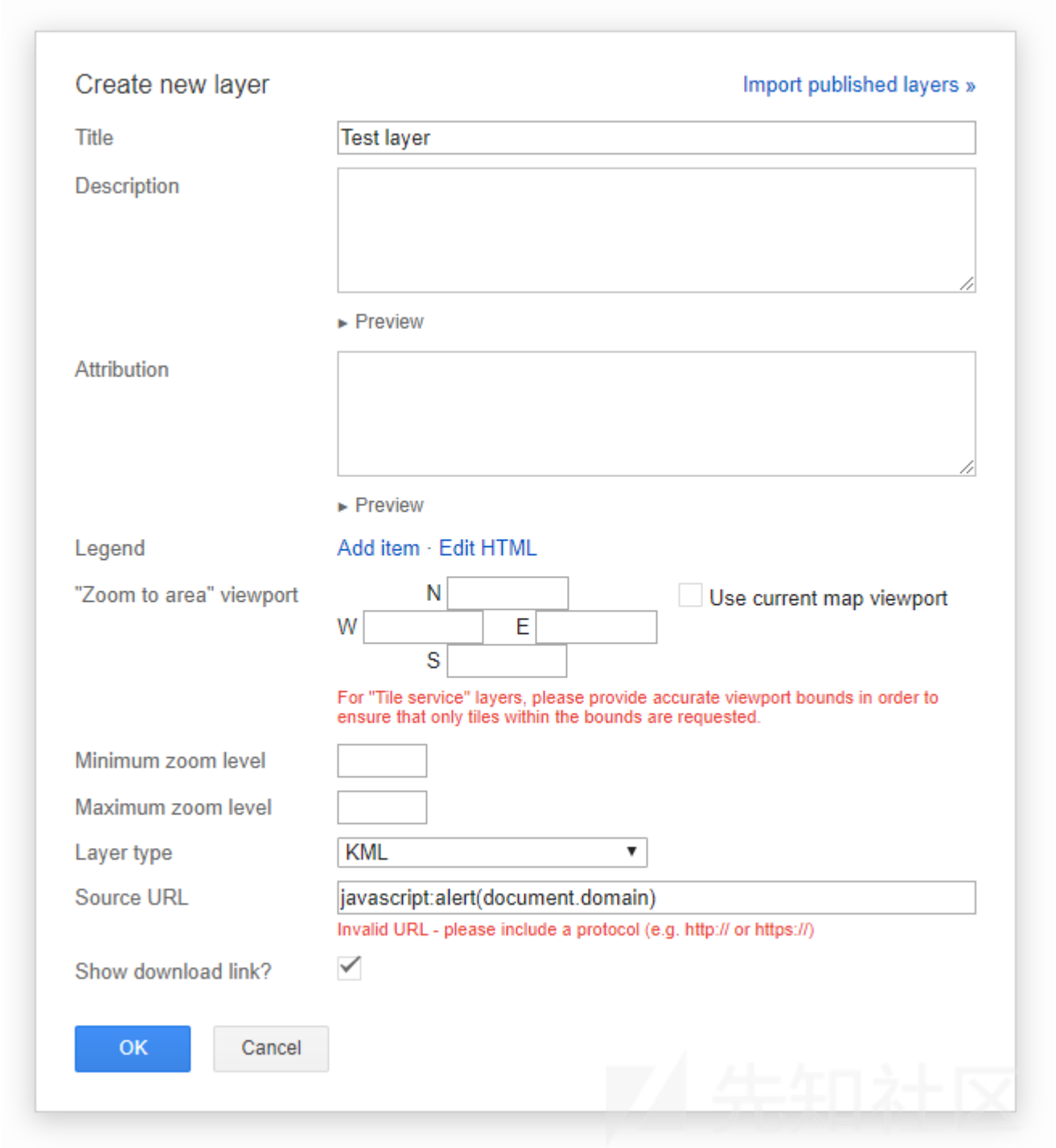
首先，我们将向地图添加一个新图层。



将弹出用于创建新图层的对话框。

我们可以在对话框中输入任何内容作为“标题”，现在，如果我们在“Source URL”字段中输入`javascript:alert(document.domain)`，将会出现错误。

无效的网址 - 请包含协议（例如：http://或者https://）



这意味着，它要求你在保存新图层之前检查URL是否有效。验证URL经过反混淆的JavaScript代码如下所示：

```
1. if (url && !url.toLowerCase().match("^\\s*(http://|https://|docs://|$)")) {
1.   showError("Invalid URL - please include a protocol (e.g. http:// or https://)");
1. }
```

验证是在保存请求发送到后端之前在客户端进行的。

修改请求

我们可以使用像Fiddler或Burp Suite这样的Web代理来修改请求并发送修改后的内容：

首先，我们需要将“Source URL”更改为有效的URL(例如：https://example.com)

我们将单击“确定”按钮并单击“保存”以发送保存请求，然后我们将修改请求，

下面的请求：

```
POST https://google.org/crisismap/.api/maps/1234

{
```

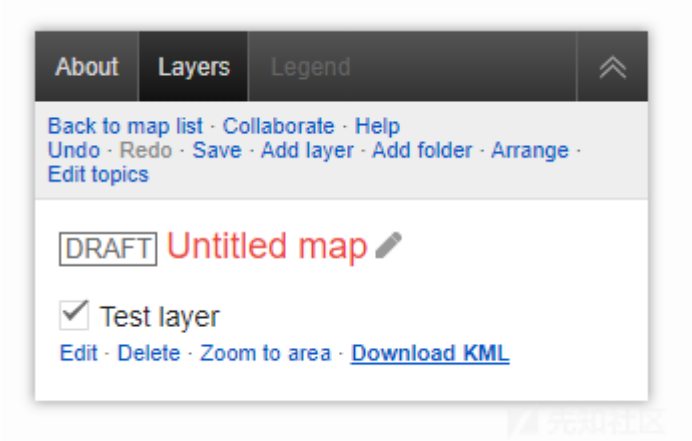
```
 "id": "1234",
 "title": "Untitled map",
 "base_map_type": "GOOGLE_ROADMAP",
 "layers": [{
"id": "1",
"title": "Test layer",
"visibility": "DEFAULT_ON",
"type": "KML",
"source": {
 "kml": {
"url": "https://example.com"
 }
}
 }]
}
```

我们可以使用`javascript:alert(document.domain)`代替`https://example.com`并发送修改后的请求。

测试XSS

现在请求已发送并保存，因此我们将重新加载页面。

打开“图层”，然后单击“下载KML”。



点击下载链接后，XSS将被触发，并弹出带有域名的警告框！

如何修复

为什么会这样？URL验证仅发生在前端而不是后端，这意味着可以通过验证后端的URL来解决这个问题。

XSS显示在DOM中之前，验证URL，而不是将其保存在后端时检查URL，这很显然不是Google开发人员的初衷。

因此，如果URL无效，则不会将其用作链接。它将使用无意义的值，例如：`about:invalid`。

```
<a href="about:invalid#zClosurez">Download KML</a>
```

影响

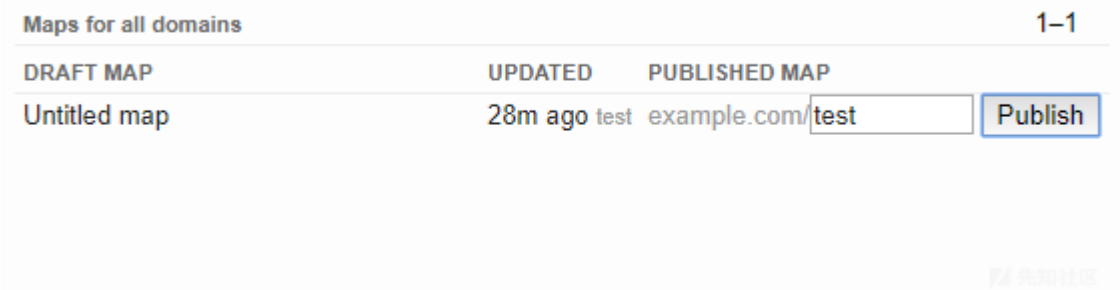
我们有一个指向javascript的链接：URI中包含payload，该链接位于用于管理地图的页面上，你必须登录并拥有访问该页面的权限。

因为只有我们能够执行此XSS，很显然这是反射型XSS。

现在我们如何能够让self-XSS变成有用的XSS呢？

扩大影响

我们创建的每张地图都可以发布，供公众查看，如果你通过example.com的电子邮件登录，则可以将地图发布到URL：http://google.org/crisismap/example.com



任何人都可以打开此URL并查看我们创建的地图，要使XSS正常工作，用户将打开或导航到此页面，打开“图层”，然后单击“下载KML”链接。

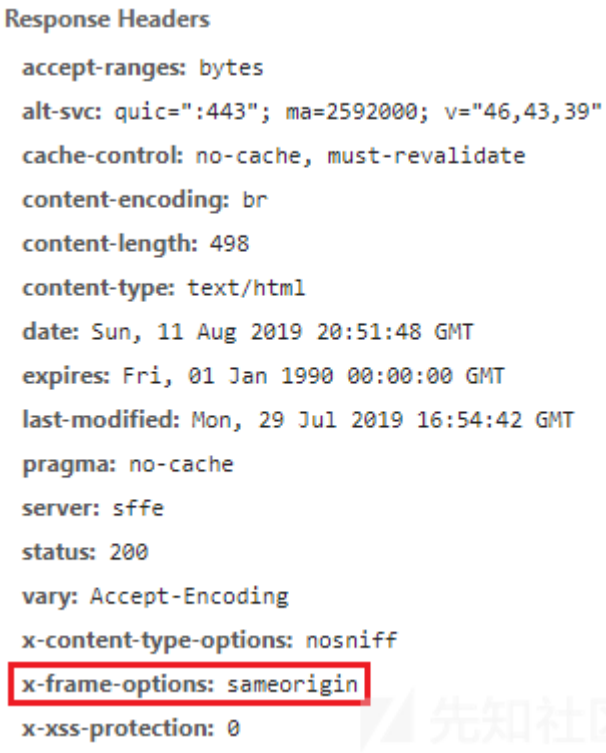
这意味着它不再是反射型XSS，但是用户必须做太多步骤才能让这个XSS变得有用。

点击劫持

如果我们查看响应HTTP头信息，我们可以看到google.org不发送X-Frame-Options,如下图所示：

Response Headers

```
accept-ranges: bytes
alt-svc: quic=":443"; ma=2592000; v="46,43,39"
cache-control: private, max-age=0
content-encoding: gzip
content-length: 13993
content-type: text/html
date: Sun, 11 Aug 2019 20:45:31 GMT
expires: Sun, 11 Aug 2019 20:45:31 GMT
last-modified: Mon, 03 Jun 2019 06:30:00 GMT
server: sffe
status: 200
vary: Accept-Encoding
x-content-type-options: nosniff
x-xss-protection: 0
```

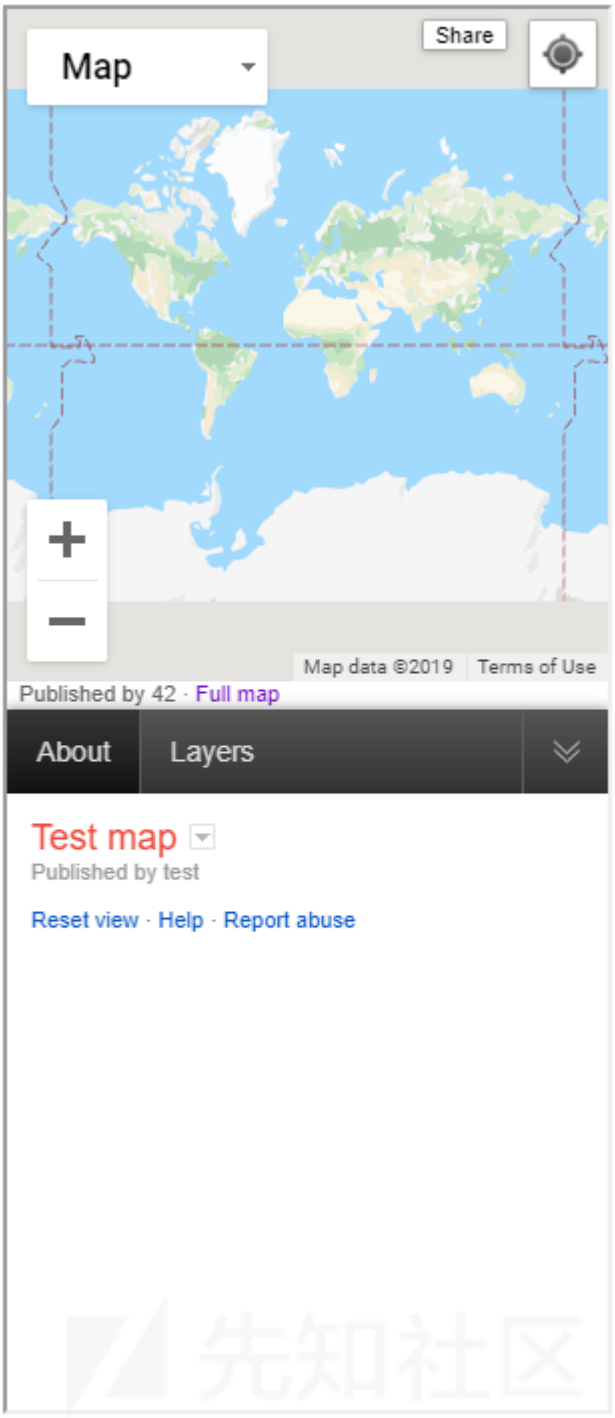


X-Frame-Options
HTTP响应头可用于指示是否应允许浏览器在<frame>，<iframe>，<embed>或<object>中呈现页面，网站可以通过确保其内容未嵌入到其他网站中来避免点击劫持攻击

google.org上缺少HTTP header意味着我们可以将已发布的地图嵌入到我们自己网站上的iframe中。

```
<iframe src="https://google.org/crisismap/example.com/test"></iframe>
```

为了出现XSS，用户现在甚至不必离开我们的网站。但是他们仍然需要点击iframe中的两个位置（“图层”“下载KML”），如下图所示：



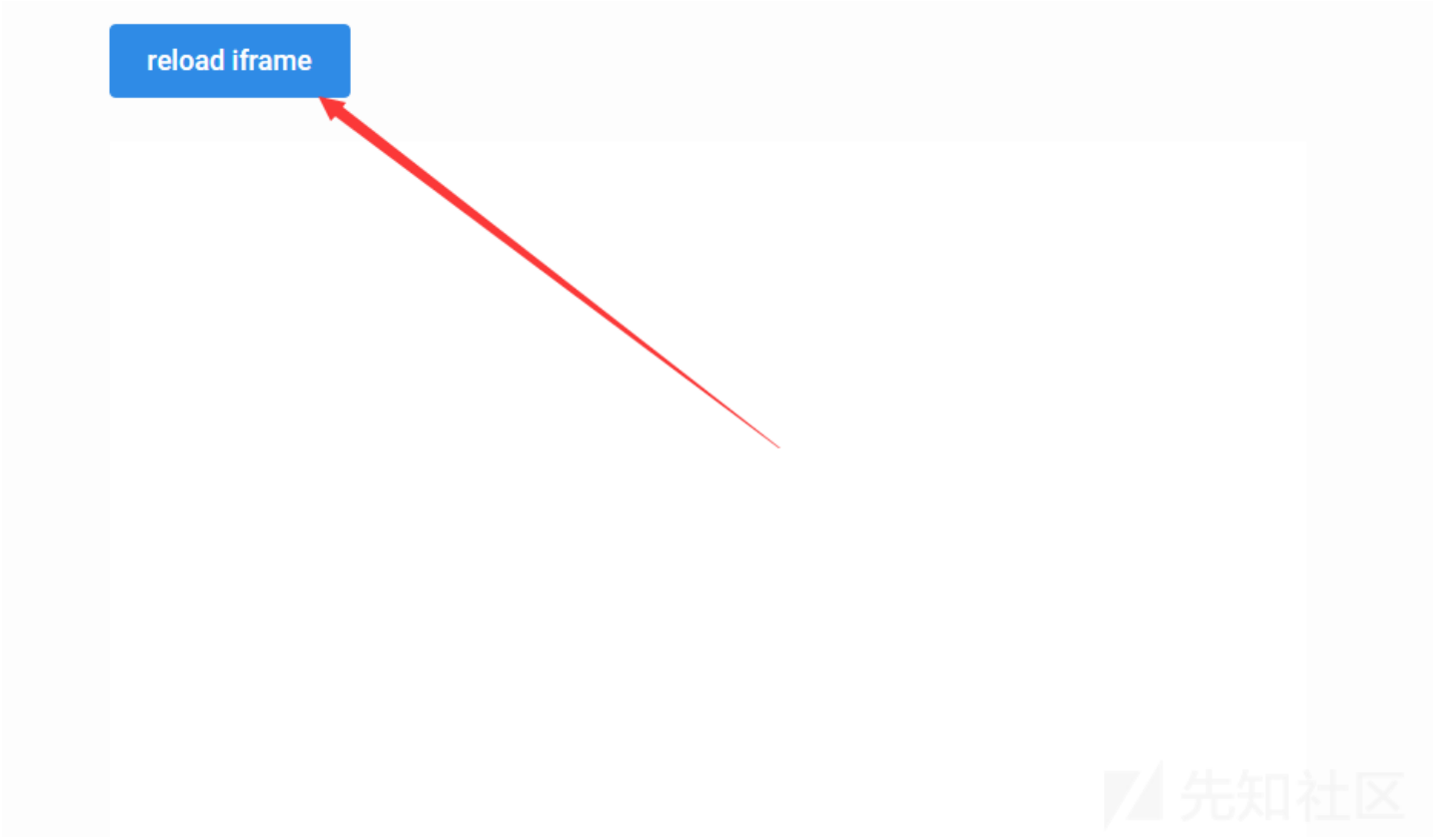
iframe加载在我们的网站上，这意味着我们可以使用CSS和JavaScript来操作它.

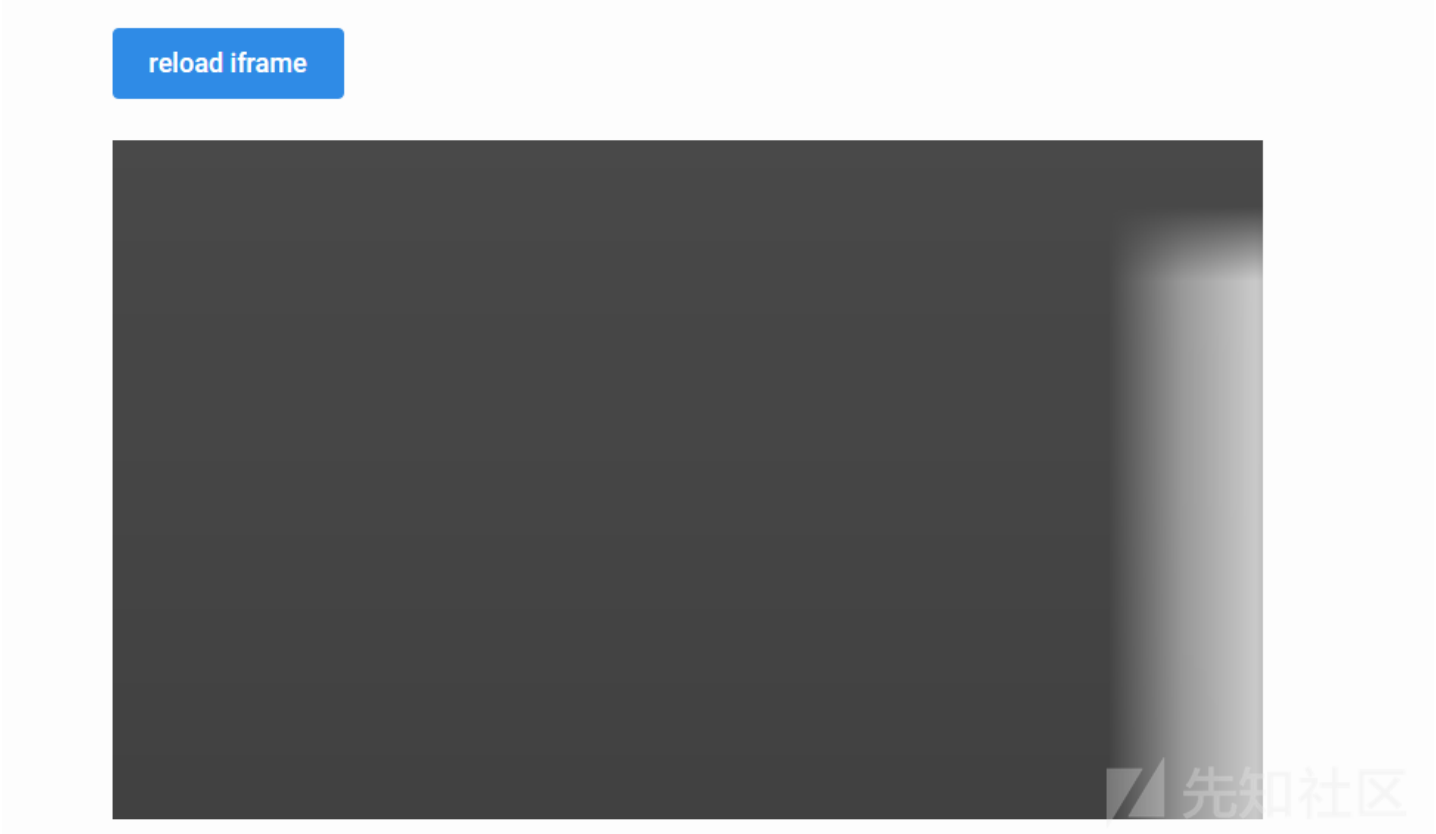
我想到的第一件事就是将黑色DIV放在我们希望用户点击的位置,然后检测单击并将DIV移动到第二个点。

这个方法很好，但仍然需要用户点击两个不同的位置。

但更有效的解决方案是绝对定位iframe，以便用户根本不必移动光标。

以下是现场演示。它将iframe缩放50倍并将其移动到我们希望用户单击的位置。首先到“图层”选项卡，点击后，它会在带payload的链接上移动：





为了证明此漏洞已修复的事实，链接转到`https://`而不是`javascript:`

结论

这里有几件事要做。

不要相信用户输入，在使用之前始终验证/转义它，甚至在保存之前更好地检查它是否有效。

通过正确设置X-Frame-Options header，不允许在你的网站中嵌入其它站点的iframe。

在查找漏洞时，请尝试查找漏洞的最高严重性。例如，如果你找到XSS，请尝试通过查找错误配置的Cookie或端点将漏洞危害提升到帐户接管。

寻找仍然适合Bug Bounty程序范围的旧项目。我在Google Crisis Map中发现了另外两个漏洞，我也会发布有关它们的文章。

时间线

1. 2018.09.12 漏洞报告
3. 2018.10.12 等级改变为P1
5. 2018.10.12 检测漏洞
7. 2018.10.12 漏洞存在
9. 2018.11.12 奖金发出

上一篇：Towards the Detec... 下一篇：SSRF in the Wild:...
1. 1 条回复

alit**** 2019-09-01 16:02:55

后面翻译的好乱，没检查吗？和原文差了好多

0 回复Ta

登录 后跟帖

先知社区

现在登录

热门节点

技术文章

社区小黑板

目录

RSS 关于社区 友情链接 社区小黑板