补档--【THM】Content Discovery(网站内容发现)-学习

本文相关的TryHackMe实验房间链接： https://tryhackme.com/room/contentdiscovery

通过学习相关知识点：了解在网络服务器上发现可能导致漏洞的隐藏或私人内容的各种方法（网站内容发现也属于信息收集的范畴）。

H2 简介

首先，我们应该问，在 Web 应用安全的背景下，什么是网站内容？ 网站内容可以是很多东西：文件、视频、图片、备份、网站功能等。 当我们谈论网站内容发现时，我们不是在谈论我们可以在网站上看到的显而易见的东西，而是指那些网站并没有立即呈现给我们、并不总是供公众使用的网站内容。

例如，该网站内容可以是供员工使用的页面或门户、网站的旧版本、备份文件、配置文件、管理面板等。

我们将介绍在网站上发现内容的三种主要方式：手动发现、自动发现和 OSINT（开源情报）。

答题

H2 手动发现网站内容 - Robots.txt

我们可以在网站上手动检查多个位置以开始发现更多网站内容。

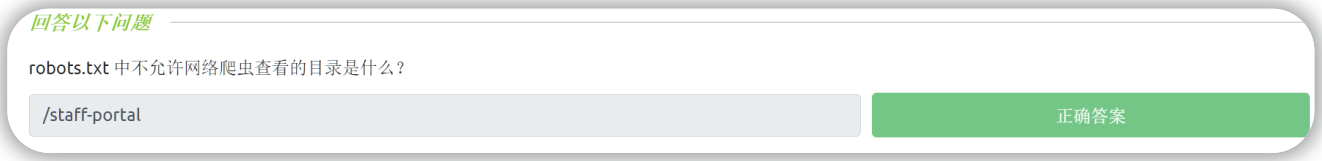
Robots.txt（在现实环境下该文件可能无法提供信息）

robots.txt 是一个文件，它将告诉搜索引擎哪些页面可以显示在搜索引擎结果中，哪些页面不允许显示在搜索引擎结果中，或者禁止特定搜索引擎完全抓取该网站所有资源。 通常的做法是使用robots.txt 限制该网站的某些区域，使其不会显示在搜索引擎结果中。

被限制搜索的页面可能是网站客户的管理门户或文件等区域。 robots.txt文件为我们提供了一个网站所有者不希望我们发现的关于网站资源位置的列表。

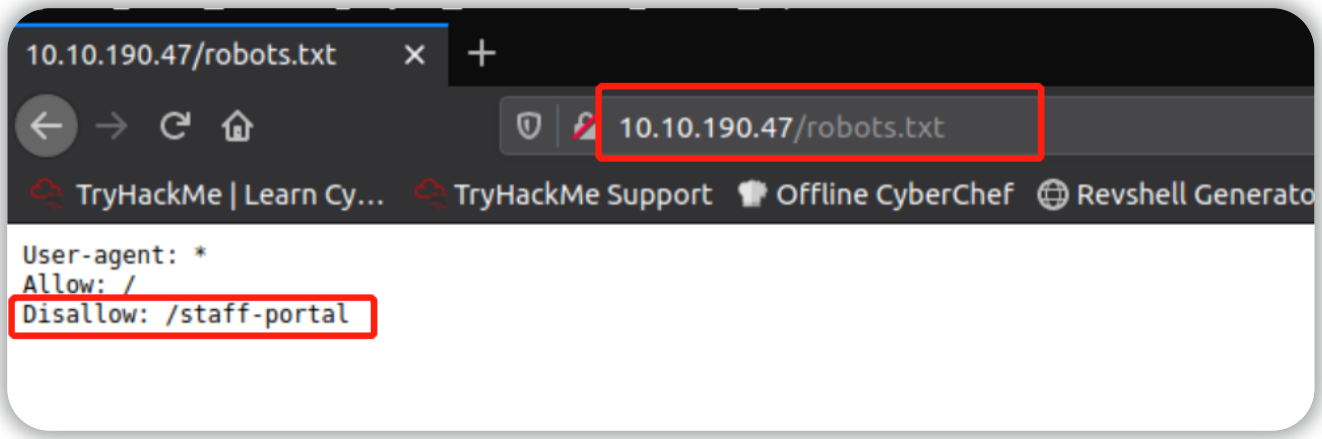
查看 Acme IT Support 网站（目标示例站点）上的 robots.txt 文件 - 为此，请在 AttackBox 上打开 Firefox，然后输入网址： http://MACHINE_IP/robots.txt （此 URL 将在你启动目标机器后 2 分钟更新）

答题



查看目标站点的robots.txt文件：

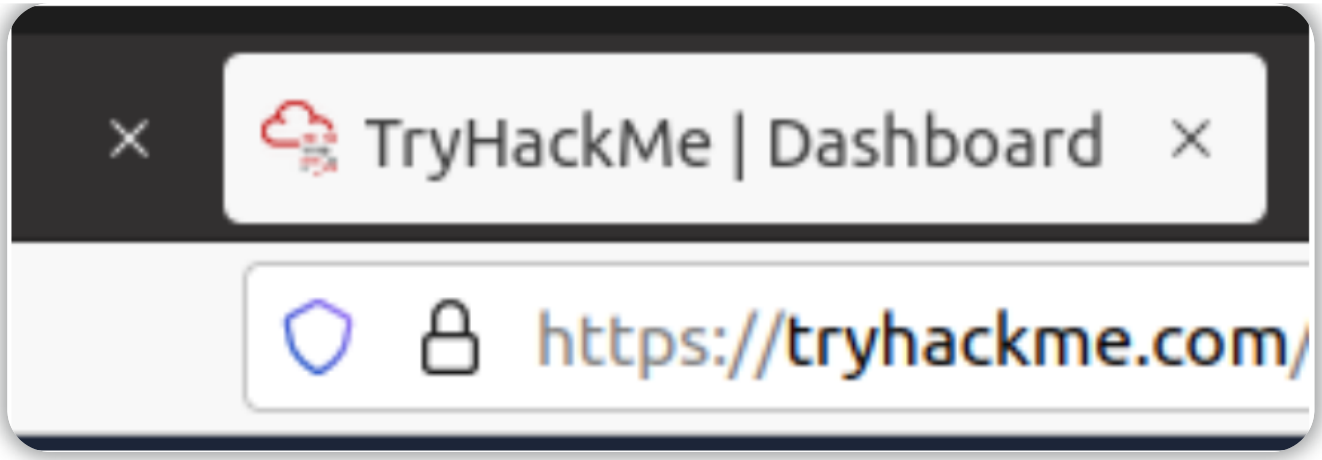
```
http://MACHINE_IP/robots.txt
```



H2 手动发现网站内容-Favicon

Favicon

favicon 是一个小图标，显示在浏览器的地址栏或选项卡中，用于品牌化网站。



有时，当使用网站框架构建网站时，框架自带的图标会作为安装的一部分而留下，如果网站开发人员没有用自定义图标替换它，这就可以让我们了解当前网站正在使用的框架。

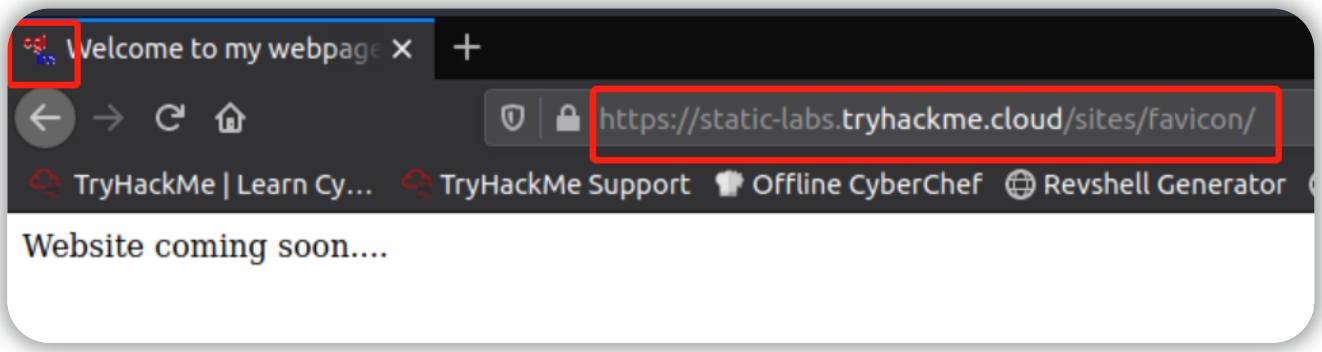
OWASP 托管了一个常见网站框架图标的数据库，你可以使用它来检查目标的favicon（图标）以查看该图标对应的网站框架是什么：

> https://wiki.owasp.org/index.php/OWASP_favicon_database

一旦我们知道了网站当前使用的框架堆栈，我们就可以使用外部资源来发现更多关于它的信息。

实践练习

在 AttackBox 上，打开 firefox 并输入URL https://static-labs.tryhackme.cloud/sites/favicon/ 在这里你会看到一个基本的网站，上面写着“网站即将推出.....”，在浏览器的地址选项卡上，你会注意到一个图标，你可以确认该站点正在使用一个网站图标。



查看该网页源代码，你会看到源代码第六行包含一个指向 `images/favicon.ico` 文件的链接。

```
    <title>Welcome to my webpage!</title>
    <link rel="shortcut icon" type="image/jpg" href="images/favicon.ico"/>
</head>
```

在 AttackBox 上运行以下命令，它将下载目标网页相关的 favicon 并获取其 md5 哈希值，然后你可以通过上面给出的 在线网站 查找该图标对应的网站框架：

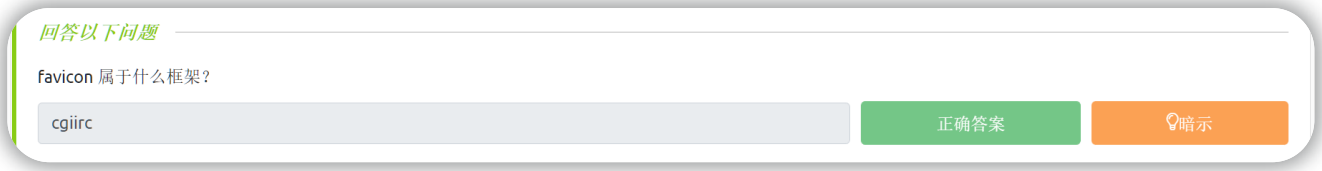
```
user@machine$ curl https://static-
labs.tryhackme.cloud/sites/favicon/images/favicon.ico | md5sum
```

注意：如果你得到的哈希值以 427e 结尾，那么代表你的 curl命令 失败了，你可能需要再试一次。

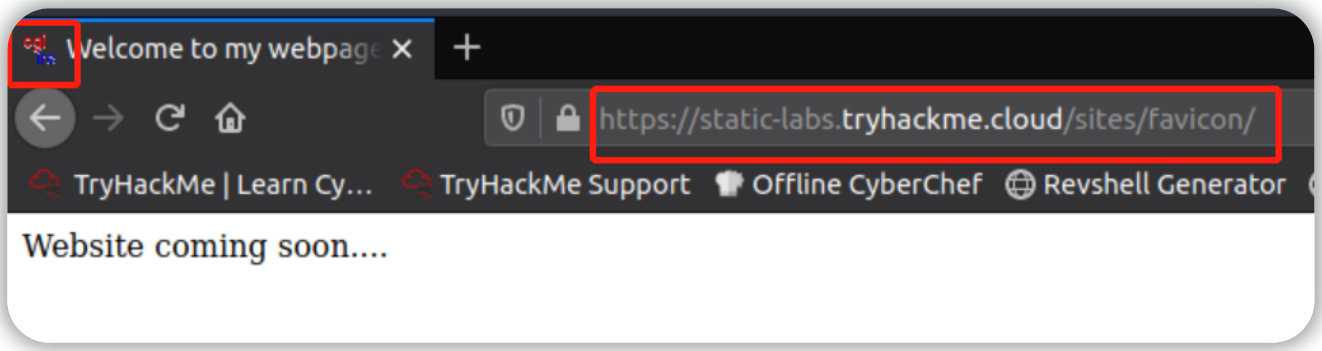
你也可以在 Windows 上的 Powershell 中运行命令，如下所示：

```
PS C:\> curl https://static-labs.tryhackme.cloud/sites/favicon/images/favicon.ico -
UseBasicParsing -o favicon.ico
PS C:\> Get-FileHash .\favicon.ico -Algorithm MD5
```

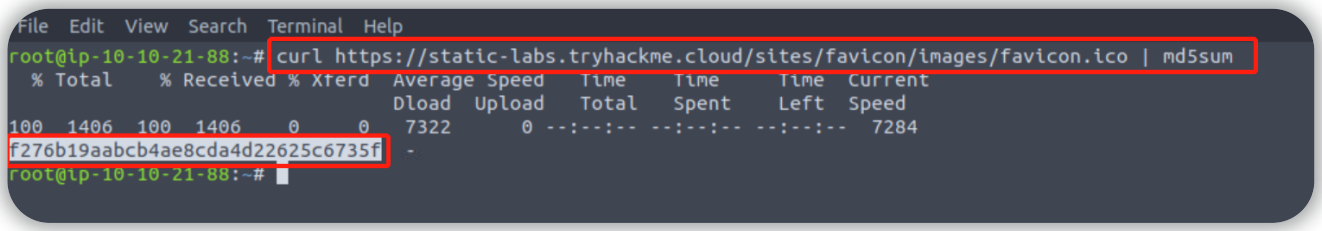
答题



查看目标站点图标：

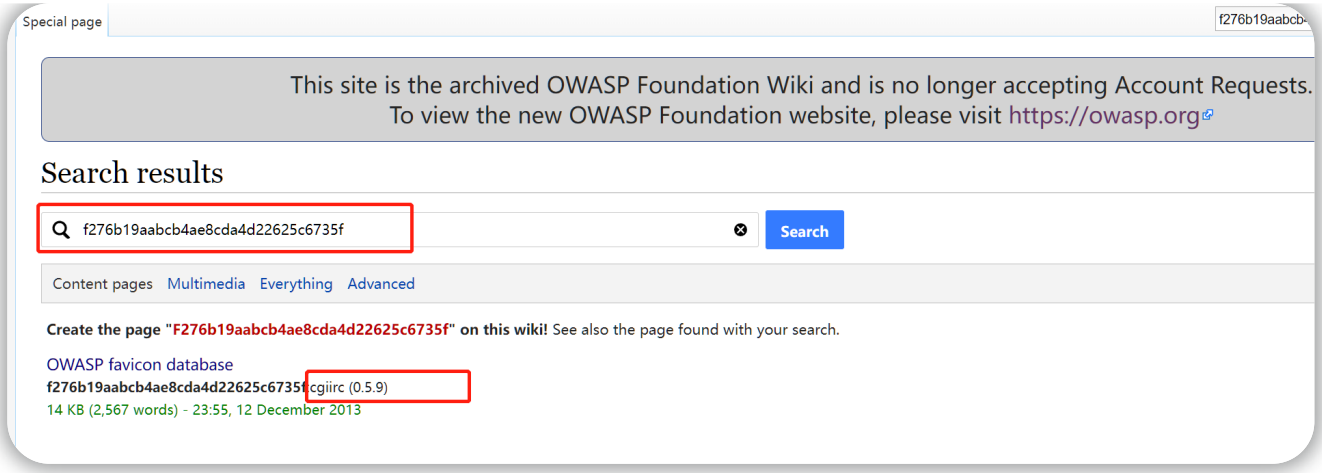


下载目标网页相关的 favicon 并获取其 md5 哈希值：



> f276b19aabcb4ae8cda4d22625c6735f

使用 https://wiki.owasp.org/index.php/OWASP_favicon_database 查找图标hash值对应的网站框架



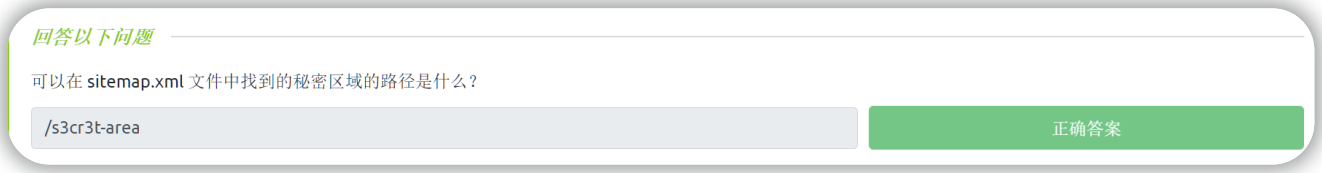
H2 手动发现网站内容-Sitemap.xml

Sitemap.xml

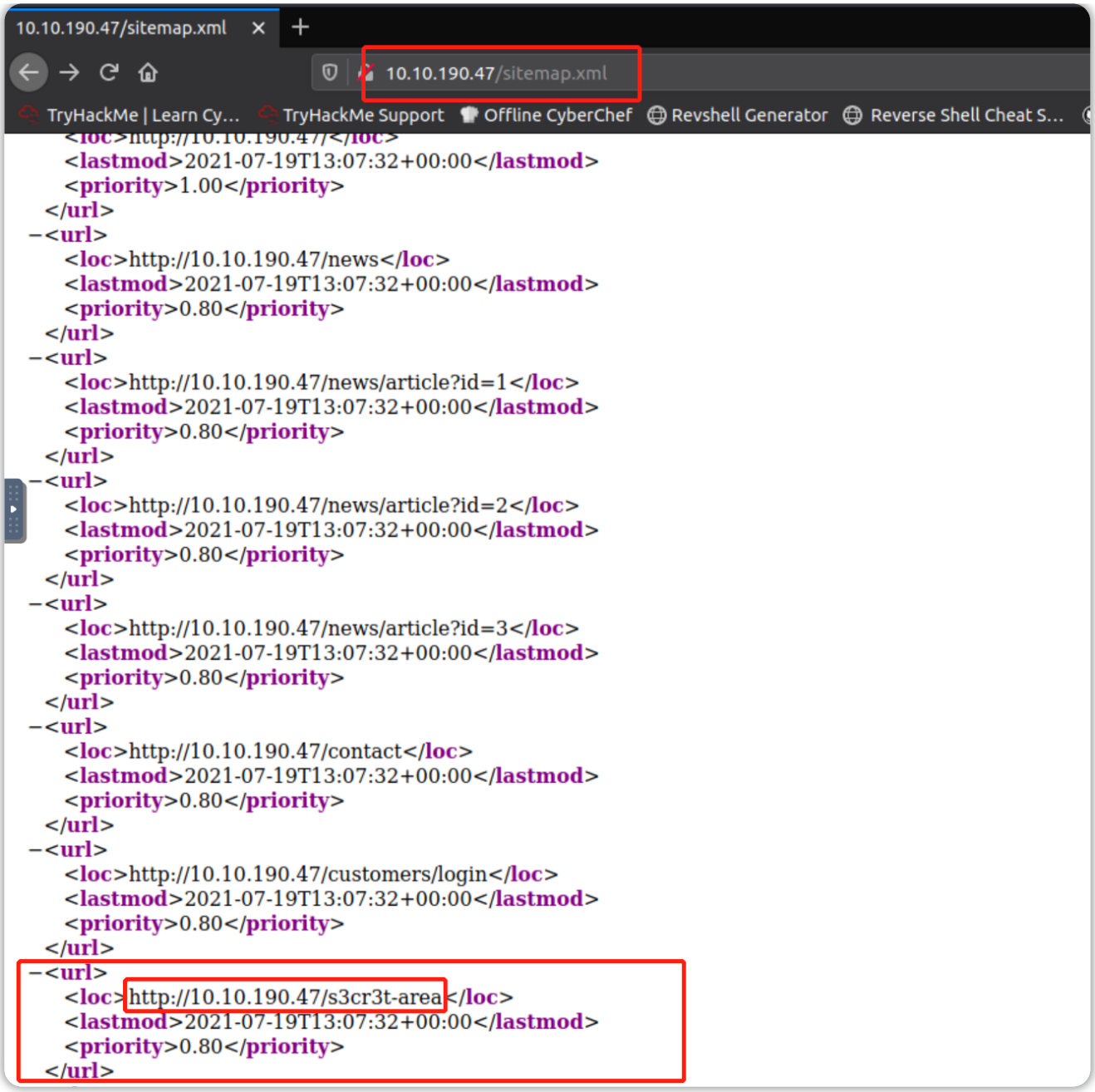
与限制搜索引擎爬虫查看范围的 robots.txt 文件不同，sitemap.xml 文件提供了网站所有者希望在搜索引擎上列出的每个文件的列表。 sitemap.xml 文件有时也可能会包含网站中难以导航的区域，甚至会列出当前网站不再使用但仍在幕后工作的一些旧网页。

查看 Acme IT Support 网站（目标示例站点）上的 sitemap.xml 文件，查看是否有我们尚未发现的网站新内容： http://10.10.190.47/sitemap.xml （在 AttackBox 上的 FireFox 浏览器中打开它 ）。

答题



访问目标站点的sitemap文件： http://10.10.190.47/sitemap.xml



H2 手动发现网站内容-HTTP Headers（HTTP标头）

HTTP标头

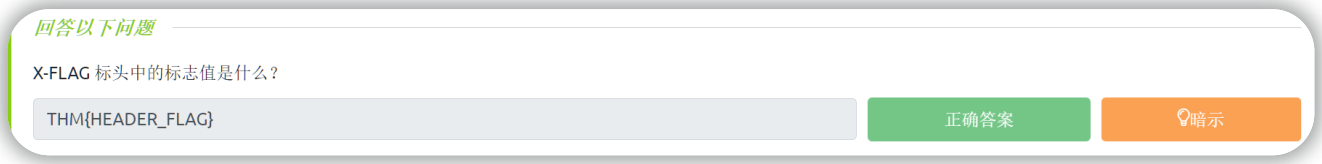
当我们向 Web 服务器发出请求时，服务器会返回各种 HTTP 标头。 这些标头有时可能会包含一些有用的信息，例如web服务器软件，可能还有网站 当前正在使用的编程/脚本语言。

在下面的示例中，我们可以看到web服务器是 **NGINX** 版本为 **1.18.0** ，运行的 **PHP** 版本是 **7.4.3**。 通过这些信息，我们可以找到正在使用的易受攻击的相关软件版本。 现在尝试对 Web 服务器运行 curl 命令，使用 `-v` 开关开启详细模式，这将输出HTTP标头的详细信息（可能会有一些有趣的东西！）。

具体命令和执行结果如下：

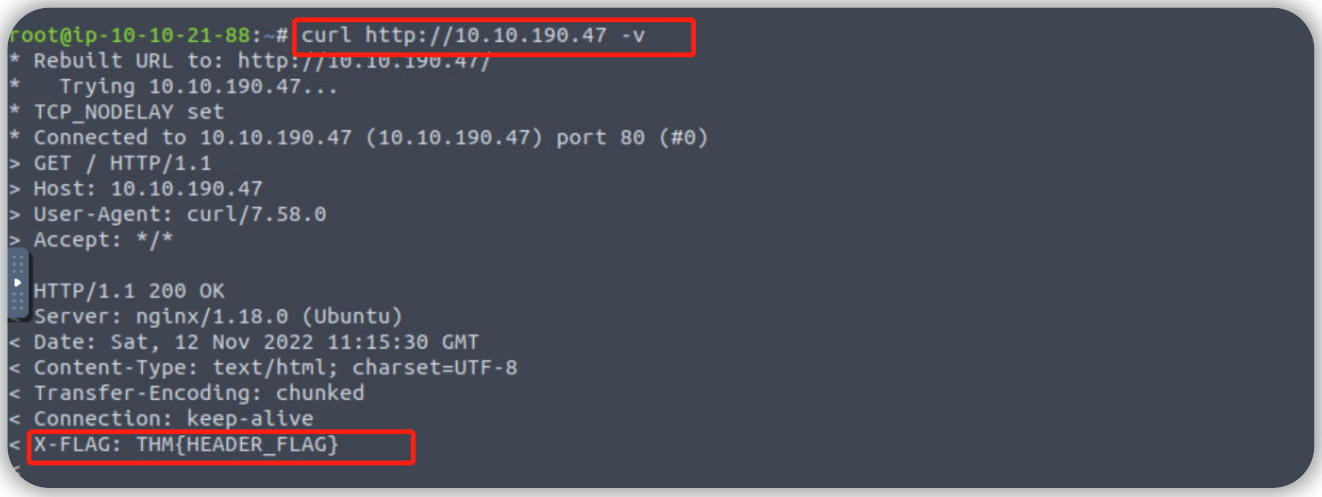
```
user@machine$ curl http://10.10.190.47 -v
* Trying 10.10.190.47:80 ...
* TCP_NODELAY set
* Connected to 10.10.190.47 (MACHINE_IP) port 80 (#0)
> GET / HTTP/1.1
> Host: MACHINE_IP
> User-Agent: curl/7.68.0
> Accept: */*
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 200 OK
< Server: nginx/1.18.0 (Ubuntu)
< X-Powered-By: PHP/7.4.3
< Date: Mon, 19 Jul 2021 14:39:09 GMT
< Content-Type: text/html; charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
```

答题



在攻击机上输入以下命令：

```
curl http://10.10.190.47 -v
```



> THM{HEADER_FLAG}

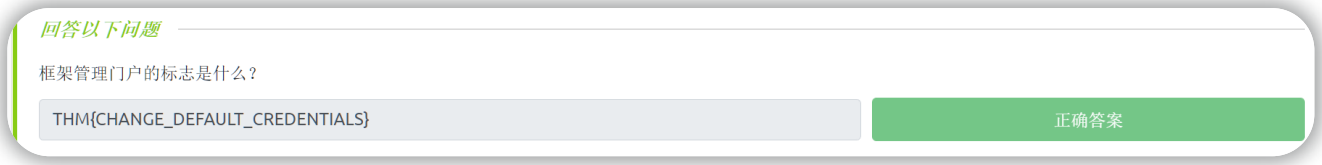
H2 手动发现网站内容-Framework Stack（框架堆栈）

框架堆栈

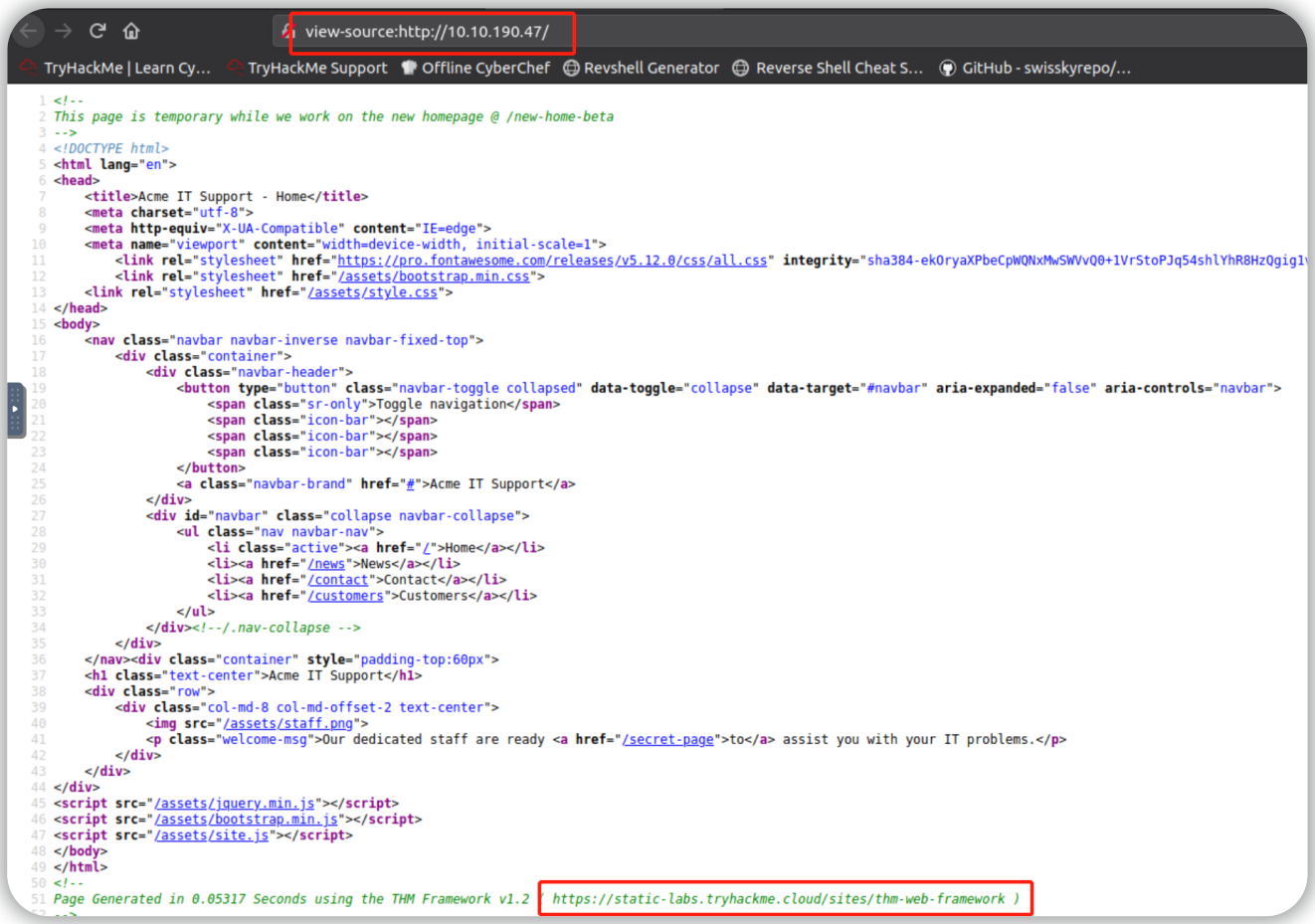
一旦你确认了目标网站正在使用的网站框架，无论是从上面的 favicon 示例中，还是通过在页面源码中寻找诸如comments、版权声明或credits等线索，你可以尝试找到关于网站框架的来源网站，然后我们就可以了解有关该框架的更多信息，以便我们发现更多关于目标网站的内容。

查看我们的 Acme IT Support 网站 (示例目标站点) 的页面源码，你会在末尾看到一条评论，其中包含了该页面加载时间以及一个指向框架来源网站的链接。 让我们查看框架的来源网站。 框架来源网站的文档页面为我们提供了关于框架管理门户的路径，我们可以在Acme IT Support 网站上访问该路径，从而帮助我们在本次实验环境下获取一个flag。

答题

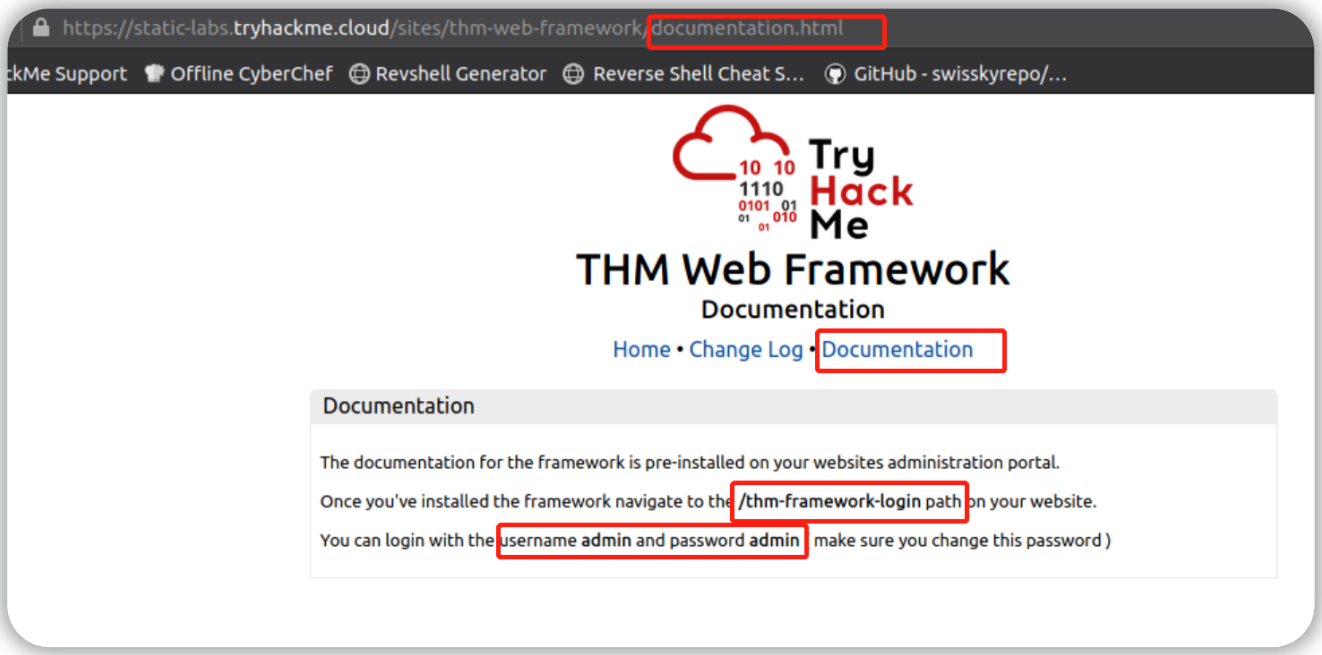


查看目标站点的页面源代码，获取指向框架来源网站的链接：

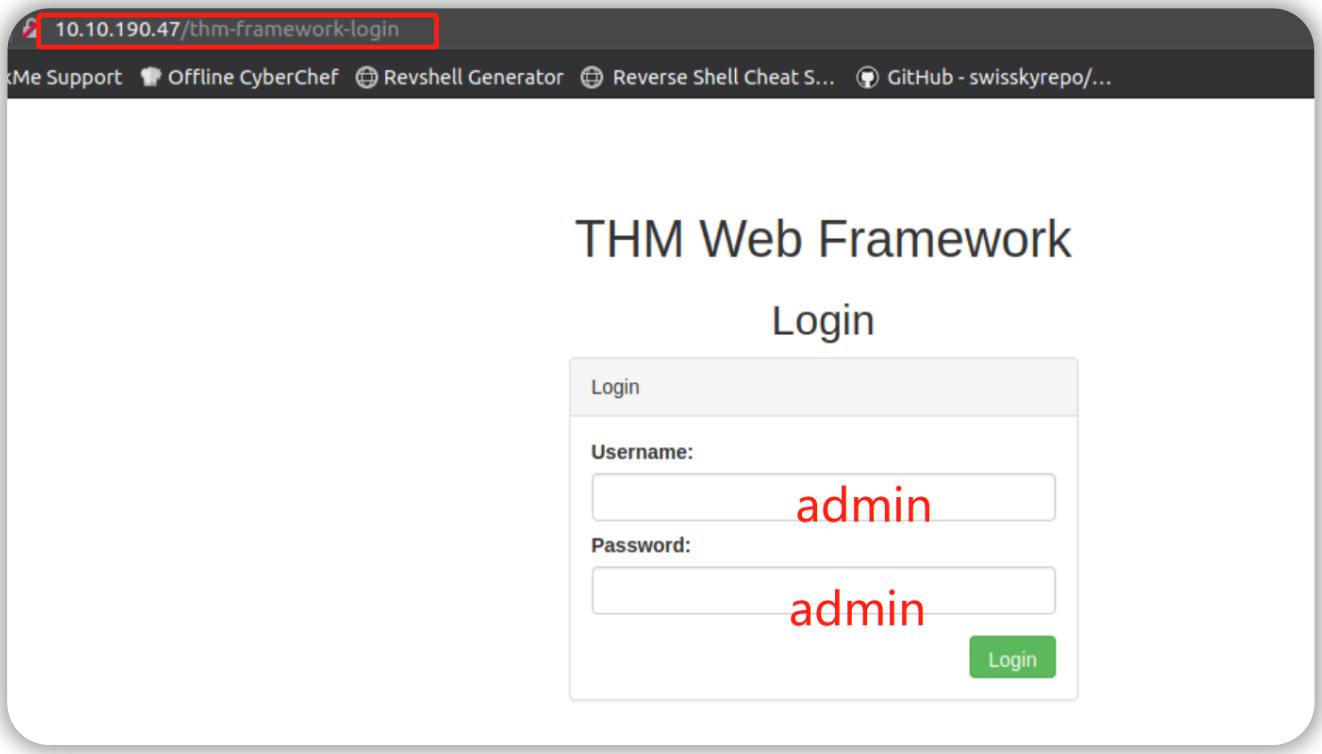


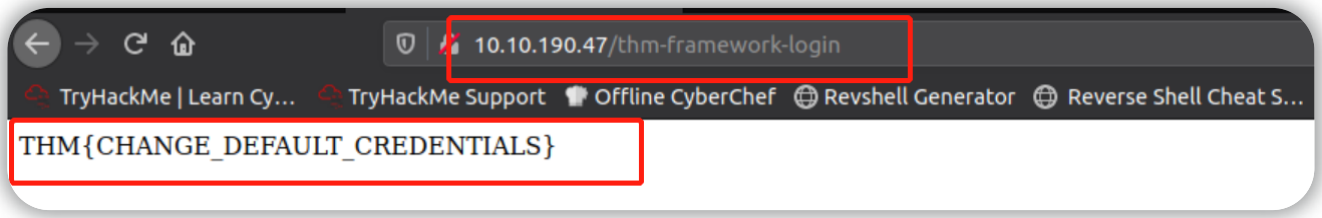
> https://static-labs.tryhackme.cloud/sites/thm-web-framework

访问框架来源网站，得知关于框架管理门户的路径：



访问框架管理门户路径，登录并获取flag：





THM{CHANGE_DEFAULT_CREDENTIALS}

H2 OSINT(开源情报)-Google Hacking / Dorking

还有一些外部资源可以帮助你发现有关目标网站的信息，这些资源通常被称为 OSINT 或开源情报，因为它们是收集信息的免费工具。

Google Hacking / Dorking（谷歌Hack语法）

Google hacking / Dorking (谷歌Hack语法)能够利用 Google 的高级搜索引擎功能，从而允许你搜索一些自定义内容。

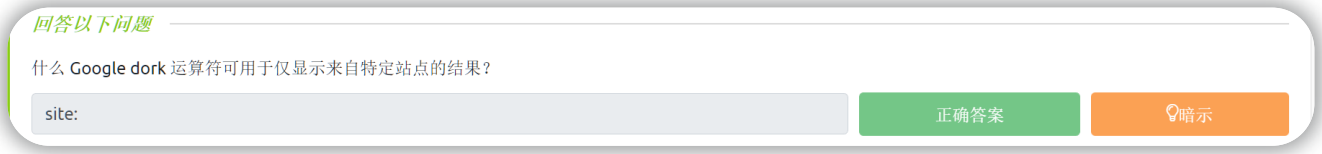
你可以通过使用谷歌Hack语法 `site:` 从某个域名中筛选搜索结果，例如 (`site:tryhackme.com`)；你也可以将该语法与某些搜索词进行匹配，例如 `admin` (`site:tryhackme.com admin`)，这会返回来自 tryhackme.com 网站且内容中包含单词 `admin` 的搜索结果。

你也可以组合多个过滤器（filters）语法，以下是关于更多过滤器（filters ）使用的示例：

Filter	Example	Description
site	site:tryhackme.com	returns results only from the specified website address
inurl	inurl:admin	returns results that have the specified word in the URL
filetype	filetype:pdf	returns results which are a particular file extension
intitle	intitle:admin	returns results that contain the specified word in the title

更多关于谷歌Hacking的信息可以在这里找到： https://en.wikipedia.org/wiki/Google_hacking

答题

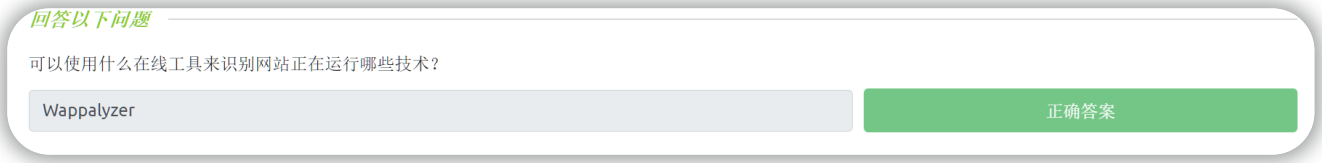


H2 OSINT(开源情报)-Wappalyzer

Wappalyzer

Wappalyzer (https://www.wappalyzer.com/) 是一个在线工具和浏览器扩展程序，可以帮助识别当前网站正在使用的IT技术，例如网站框架、内容管理系统 (CMS)、支付处理器等等，它也可以发现网站当前运行的应用程序的版本号。

答题

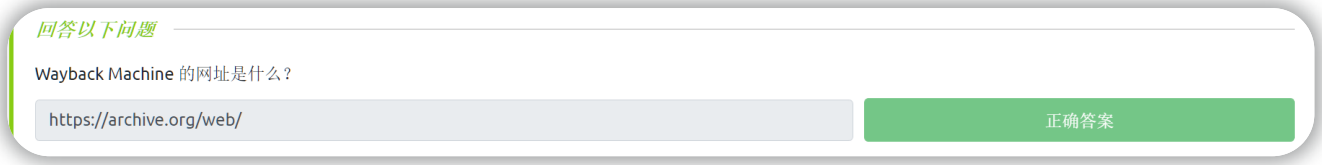


H2 OSINT(开源情报)- Wayback Machine

Wayback Machine

Wayback Machine (https://archive.org/web/)是一个可追溯到 90 年代后期的网站历史档案，你可以在该网站上搜索一个域名，它会显示该网站服务抓取网页并保存内容的所有历史时间。该网站服务可以帮助我们发现在当前网站上可能仍处于活动状态的一些旧页面。

答题



H2 OSINT(开源情报)- GitHub

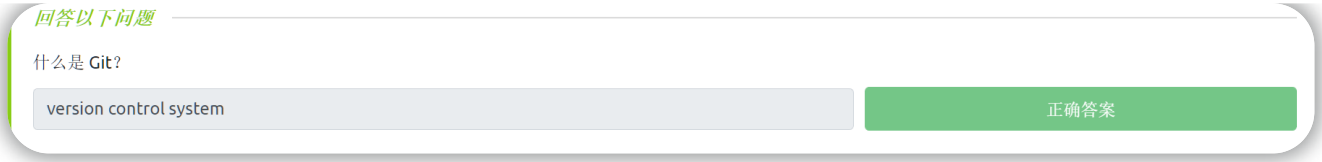
GitHub

要了解 GitHub，首先需要了解 Git。

Git 是一个版本控制系统（**version control system**），用于跟踪项目中文件的更改情况。 这使得团队工作变得更容易，因为你可以看到每个团队成员正在编辑的内容以及他们对文件所做的一些更改。 当 Git 用户完成文件更改后，他们会通过一条消息提交它们，然后将它们推送回项目中心位置（存储库），以供其他用户将这些更改拉到他们的本地计算机上。

GitHub 是托管在互联网上的 Git 版本。 GitHub 上的存储库可以设置为公共或私有，并具有各种访问控制功能。 你可以使用 GitHub 的搜索功能来查找公司名称或网站名称，以尝试找到属于目标网站的存储库。一旦有所发现，你就可以访问你尚未找到的目标网站或者目标框架的源代码、目标可能使用的默认密码或其他内容。

答题



H2 OSINT(开源情报)- S3 Buckets（S3存储桶）

S3 Buckets（S3存储桶）

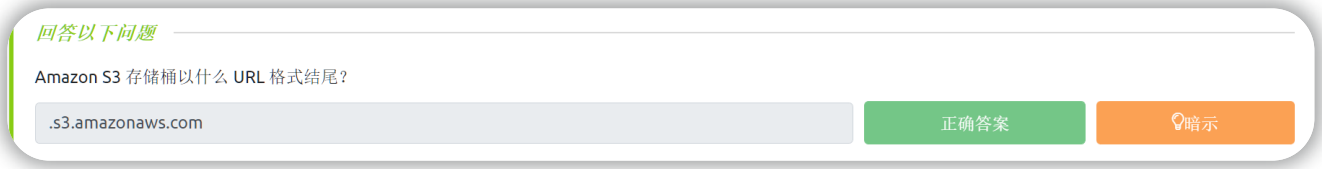
S3 Buckets 是 Amazon AWS 提供的一种存储服务，它允许人们将文件或者静态网站内容保存在云中，然后可以通过 HTTP 和 HTTPS 进行访问操作。 文件的所有者可以设置访问权限以使文件公开、私有甚至可写。 有时这些访问权限设置并不正确，文件所有者可能在无意中设置 允许访客访问不应向公众提供的文件。

S3 存储桶的格式为 `http(s)://{name}.s3.amazonaws.com` ，其中 `{name}` 由所有者自己决定，例如 `tryhackme-assets.s3.amazonaws.com` 。

我们可以通过多种方式发现 S3 存储桶，例如在网站的页面源码中、在GitHub 存储库中查找目标URL，我们甚至能够自动化进行发现S3 存储桶的过程。

一种常见的自动化发现S3 Buckets的方法是：使用公司名称后跟常用术语自行拼接url，例如 `{name}-assets` 、 `{name}-www` 、 `{name}-public` 、 `{name}-private` 等，后面跟上 `.s3.amazonaws.com` 进行url拼接即可。

答题



H2 自动发现网站内容

什么是自动发现?

自动发现是一个使用工具发现网站内容而不是手动发现网站内容的过程。

此过程是自动化的，因为它通常会发送针对目标 Web 服务器的数百、数千甚至数百万个请求，这些请求会检查目标网站上是否存在某些文件或某些目录，以便让我们能够访问我们之前未知其存在的相关资源。这个过程需要通过使用一种叫做wordlists（字典）的资源来实现。

什么是wordlists（字典）?

字典是包含一长串常用词的文本文件，它们可以涵盖许多不同的用例，例如，密码字典将包括最常用的密码列表；而在本例中我们需要一个包含最常用目录和文件名的字典。

预装在 THM AttackBox 上的字典资源是 https://github.com/danielmiessler/SecLists 该字典项目由 Daniel Miessler维护。

自动化工具

有许多不同的网站内容发现工具可用，它们都有各自的功能和缺陷，我们将介绍三种预装在我们的 AttackBox 上的工具：ffuf、dirb 和 gobuster。

在 AttackBox 上执行以下三个命令，针对的目标为 Acme IT Support 网站，看看你会得到什么结果。

使用ffuf进行网站内容发现:

```
user@machine$ ffuf -w /usr/share/wordlists/SecLists/Discovery/Web-Content/common.txt -u http://MACHINE_IP/FUZZ
```

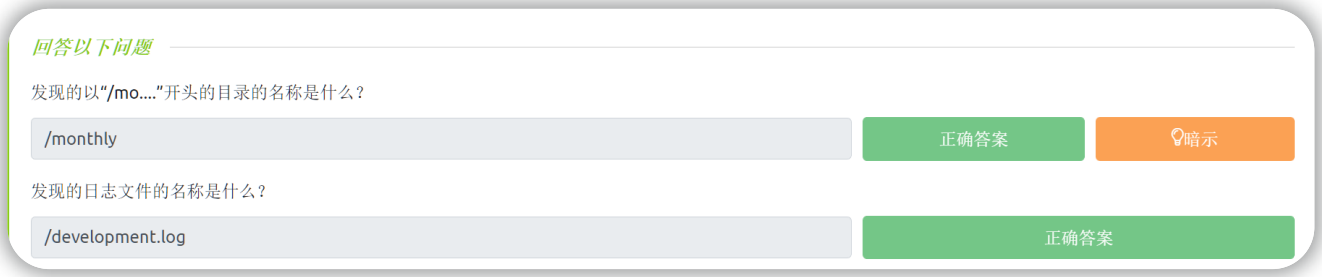
使用dirb进行网站内容发现：

```
user@machine$ dirb http://MACHINE_IP/ /usr/share/wordlists/SecLists/Discovery/Web-Content/common.txt
```

使用Gobuster进行网站内容发现：

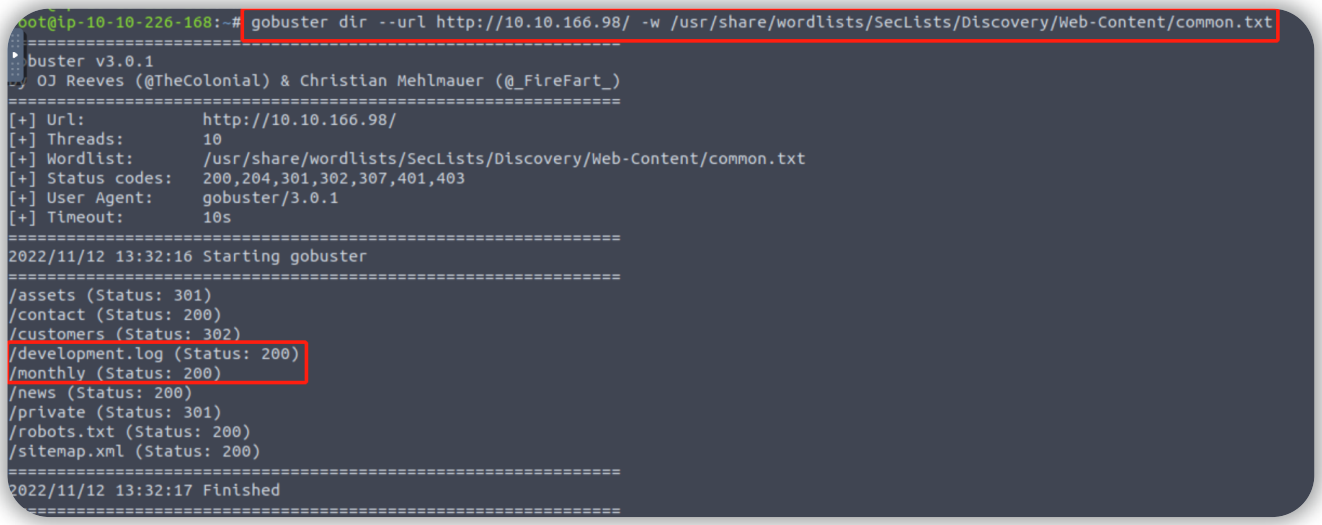
```
user@machine$ gobuster dir --url http://MACHINE_IP/ -w /usr/share/wordlists/SecLists/Discovery/Web-Content/common.txt
```

答题



使用上面介绍的三个工具之一，回答问题：

```
gobuster dir --url http://10.10.166.98/ -w
/usr/share/wordlists/SecLists/Discovery/Web-Content/common.txt
```



/monthly

/development.log