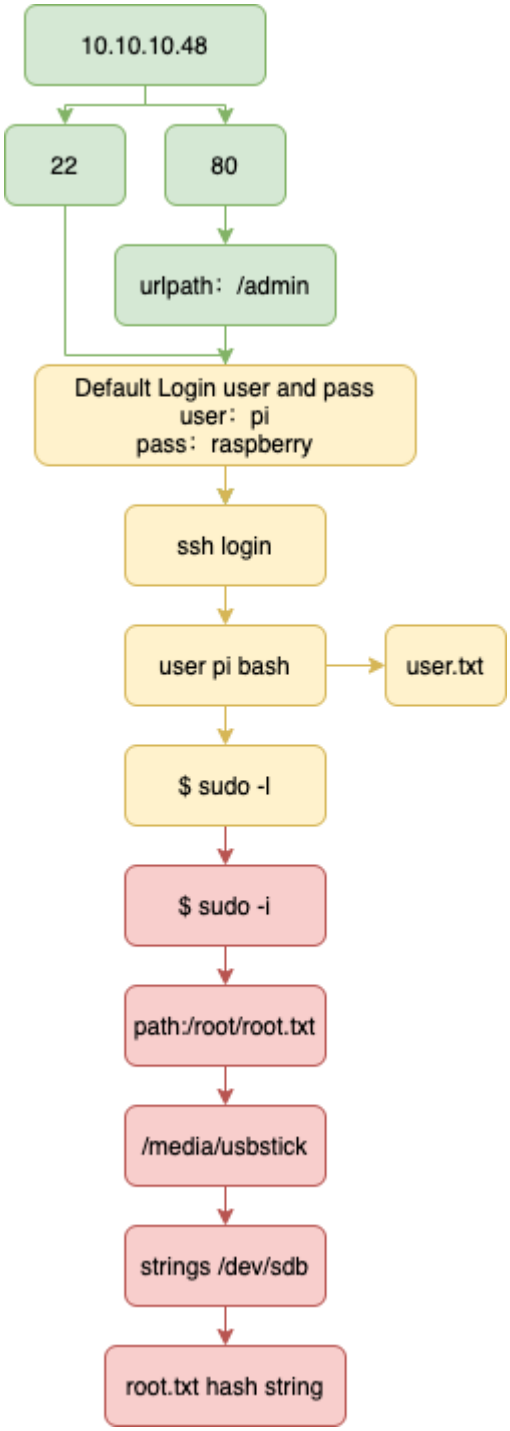
概述 （Overview）

攻击链 （Kiillchain）

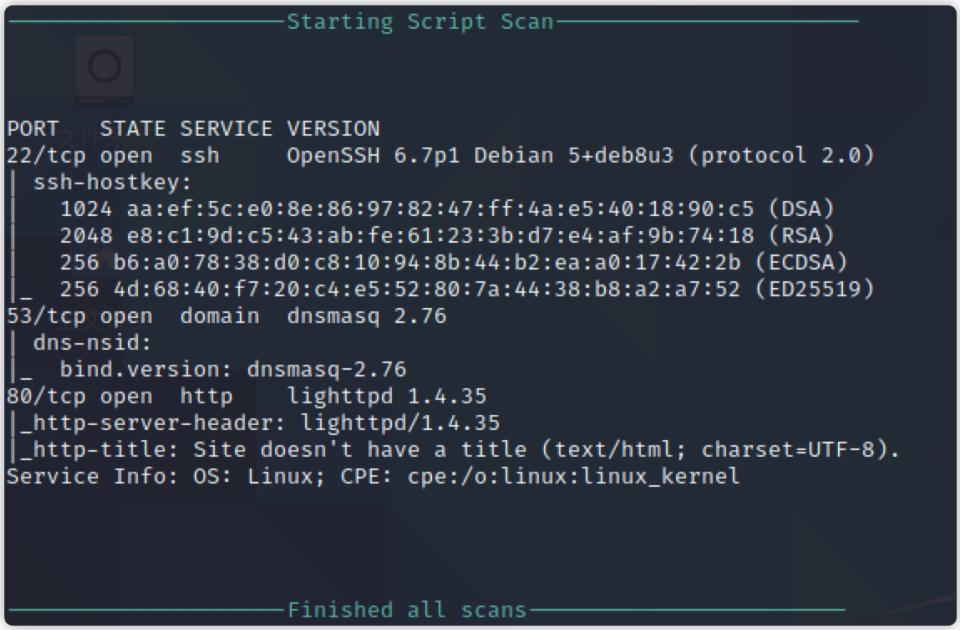


TTPs （Tactics, Techniques & Procedures）

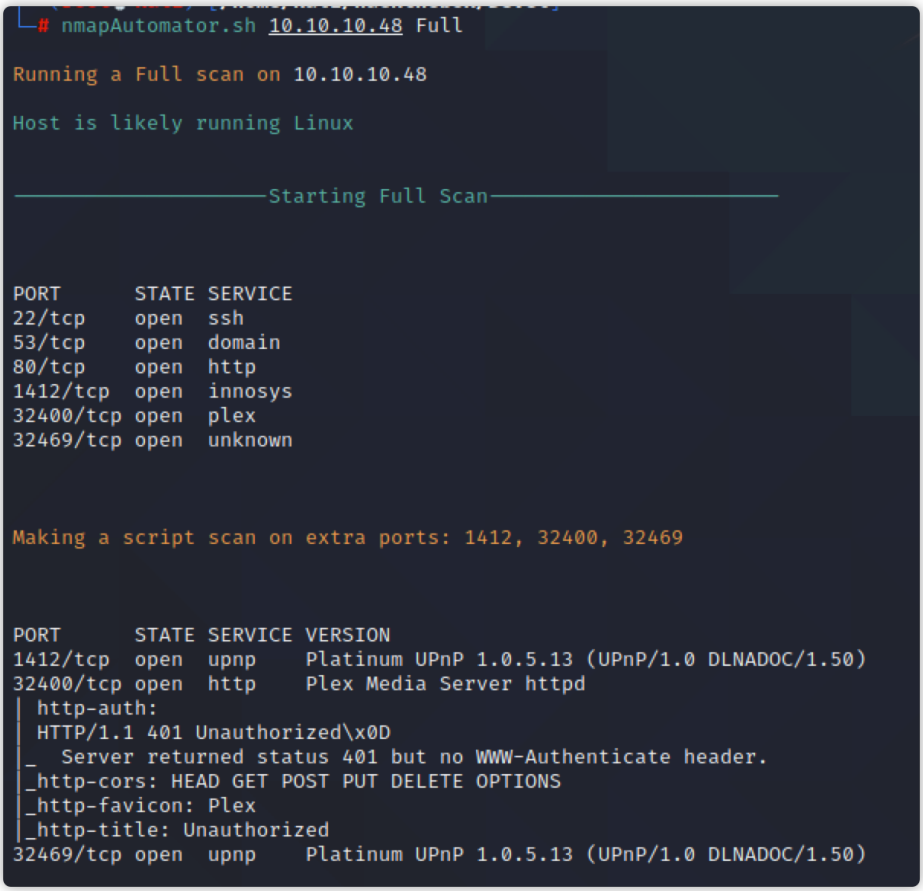
- nmapAutomator
- gobuster
- sudo
- strings
- dcfldd or dd
- testdisk
- photorec
- extundelete

阶段1：枚举

通过Nmap扫描识别到开放了 22、53、80 端口：



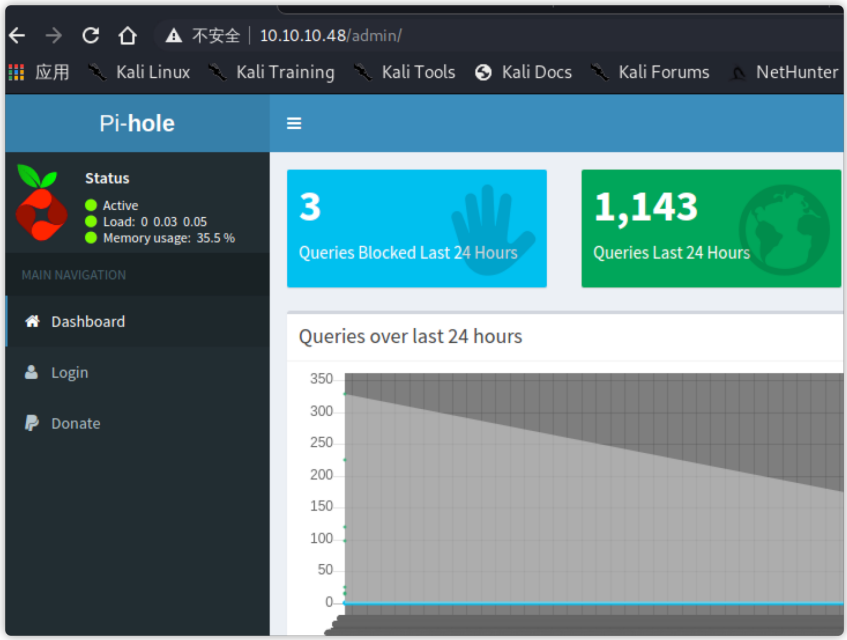
为保信息不存在遗漏，扫描了下全端口：



枚举下网站路径发现存在 `/admin`，访问后是树莓派Web管理界面。

```
┌──(root💀kali)-[/home/kali/hackthebox/Mirai]
└─# gobuster dir -u http://10.10.10.48 -w /usr/share/wordlists/dirb/common.txt
===============================================================
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.10.48
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.1.0
[+] Timeout:                 10s
===============================================================
2021/03/27 13:23:42 Starting gobuster in directory enumeration mode
===============================================================
/admin                (Status: 301) [Size: 0] [--> http://10.10.10.48/admin/]
Progress: 1624 / 4615 (35.19%)
```

根据扫出来的信息，都尝试了一下是否存在漏洞，折腾了一会发现都没法利用（WDNMD，全是兔子洞）....

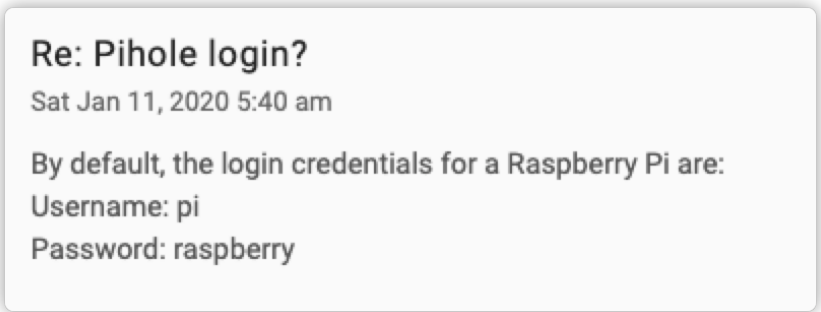


阶段2：工具和利用

阶段2.1：Pi-hole 默认口令

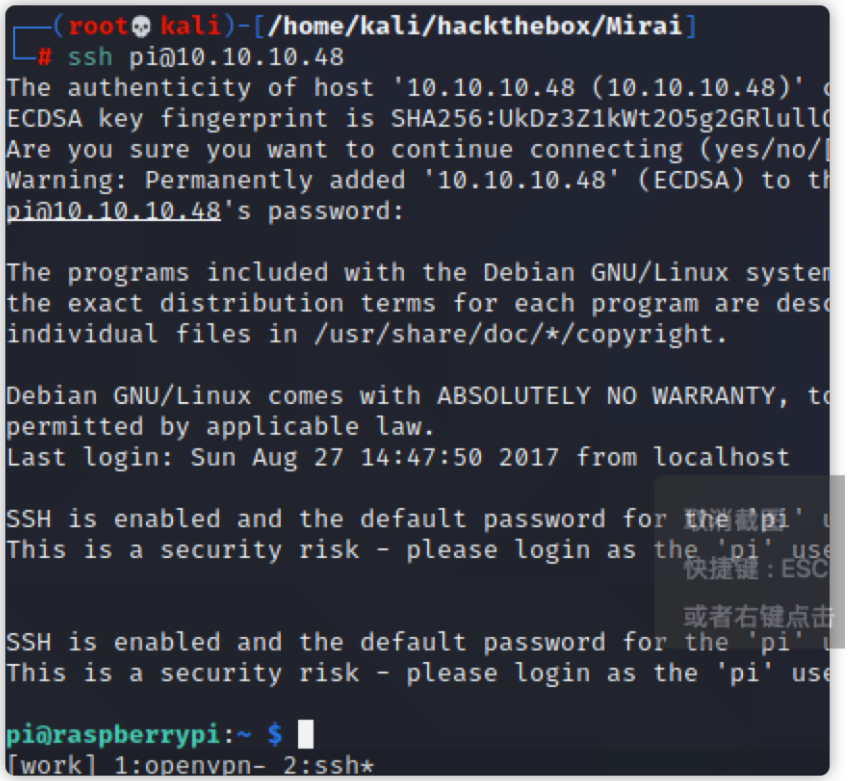
尝试搜索Web系统的默认口令，从而成功登录Web，但是界面功能很少就是一个纯粹的系统资源使用率的图标展示。

> https://www.raspberrypi.org/forums/viewtopic.php?t=261662



阶段2.2：pi用户ssh登录

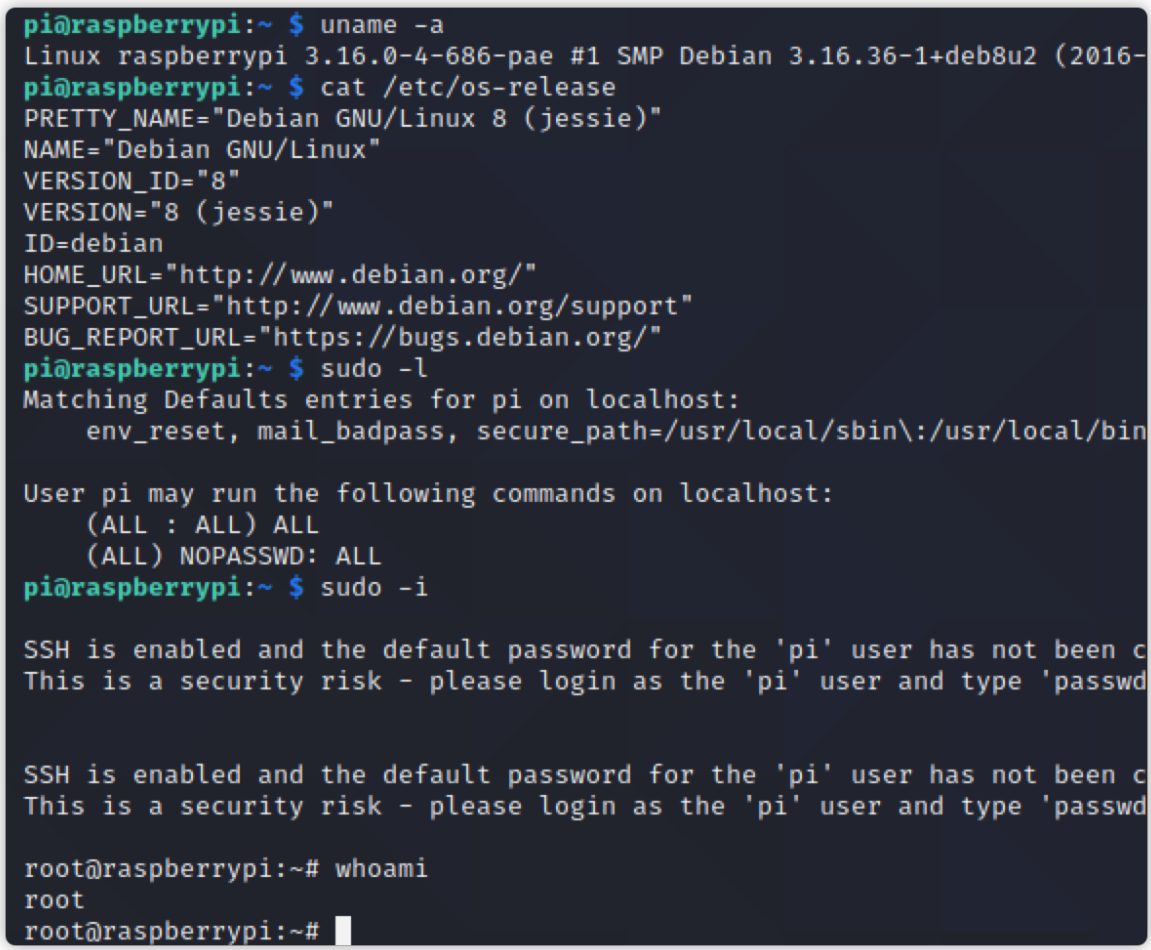
转而尝试ssh登录，发现能登录服务器。



阶段3：权限提升

阶段3.1：不安全的sudo配置

简单查看了下系统信息，尝试看看是否存在不安全的sudo配置，发现存在无需密码的命令权限提升，那就简单了：`$ sudo -i`



阶段3.2：root.txt文件恢复

本以为就这样结束了，简单靶机就这？然后 `root.txt` 提示不小心删除了这个文件的内容，但他备份到了USB里面。

```
4.0K -rw-r--r-- 1 root root 76 Aug 14  2017 root.txt
root@raspberrypi:~# cat root.txt
I lost my original root.txt! I think I may have a backup on my USB stick...
root@raspberrypi:~# df -h
Filesystem      Size  Used Avail Use% Mounted on
aufs            8.5G  2.8G  5.3G  34% /
tmpfs           100M  4.8M   96M   5% /run
/dev/sda1       1.3G  1.3G     0 100% /lib/live/mount/persistence/sda1
/dev/loop0      1.3G  1.3G     0 100% /lib/live/mount/rootfs/filesystem.squashfs
tmpfs           250M     0  250M   0% /lib/live/mount/overlay
/dev/sda2       8.5G  2.8G  5.3G  34% /lib/live/mount/persistence/sda2
devtmpfs         10M     0   10M   0% /dev
tmpfs           250M  8.0K  250M   1% /dev/shm
tmpfs           5.0M  4.0K  5.0M   1% /run/lock
tmpfs           250M     0  250M   0% /sys/fs/cgroup
tmpfs           250M  8.0K  250M   1% /tmp
/dev/sdb        8.7M   93K  7.9M   2% /media/usbstick
tmpfs            50M     0   50M   0% /run/user/999
tmpfs            50M     0   50M   0% /run/user/1000
root@raspberrypi:~# ls /media/usbstick
damnit.txt  lost+found
root@raspberrypi:~# cat /media/usbstick/damnit.txt
Damnit! Sorry man I accidentally deleted your files off the USB stick.
Do you know if there is any way to get them back?

-James
root@raspberrypi:~#
```

可以看到， `df -h` 列举出了服务器上的所有硬盘及文件挂载路径，/dev/sda 表示系统识别到的第一块磁盘，/dev/sdb 表示第二块也就是这里挂载的USB。 屁颠颠的在 `/media/usbstick` 里找 root.txt，又提示说不小心删除了这个文件。

那么这里就涉及到要做数据恢复了，然后去google一圈发现大部分都需要apt安装xxx软件，然后用软件还原。但想到也不能在服务器上直接安装这些东西，要是把删除的root.txt文件内的数据段掩盖了咋办?

回顾下系统实现删除文件的的具体逻辑：文件在写入U盘实际就是一段数据，而数据会存在U盘里的任意一个连续的区域，当删除文件时，数据并不是直接擦除它依然存在，就像指针的原理，只是把指向文件的连接给你剪短了，文件的实际数据内容还在。是不是有点绕？当写入其他数据时，有几率会覆盖之前删除的数据区间。

我这里的做法是直接通过 `strings` 查找 `/dev/sdb` 中可识别的字符串，找到里面符合root.txt内容的哈希（也可以通过 `xxd /dev/sdb，查看到所有的数据） 。

```
root@raspberrypi:/# strings /dev/
Display all 162 possibilities? (y or n)
root@raspberrypi:/# strings /dev/sd
sda   sda1  sda2  sdb
root@raspberrypi:/# strings /dev/sd
sda   sda1  sda2  sdb
root@raspberrypi:/# strings /dev/sdb
>r &
/media/usbstick
lost+found
root.txt
damnit.txt
>r &
>r &
/media/usbstick
lost+found
root.txt
damnit.txt
>r &
/media/usbstick
2]8^
lost+found
root.txt
damnit.txt
>r &
3d3e483143ff12ec505d026fa13e020b
Damnit! Sorry man I accidentally deleted your files off the USB stick.
Do you know if there is any way to get them back?
-James
root@raspberrypi:/#
```

学习数据恢复

复盘时看了下IPPSEC的解题思路，它用到了好几种工具值得学习一波。

首先用了磁盘备份工具dcfldd，是dd工具的增强版，这里使用它来将服务器上挂载的盘进行备份拉回到本地来分析。

```
kali>$ ssh pi@10.10.10.48 "sudo dcfldd if=/dev/sdb | gzip -1 -" | dcfldd of=pi.dd.gz
```

这样就得到了一个 `pi.dd.gz` 压缩包文件，在本机上解压后用 `binwalk（一个固件分析工具）` 进一步分析：

```
$ gunzip -d pi.dd.gz
```

```
$ binwalk -Me pi.dd
```

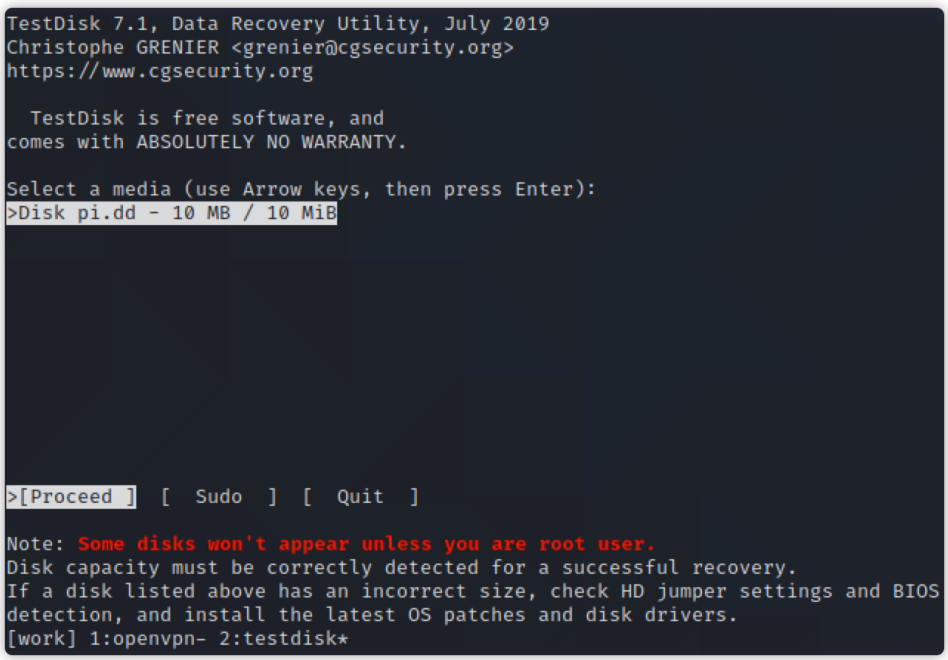
- -M，--matryoshka 递归扫描提取的文件
- -e，--extract 自动提取已知的文件类型

```
$ cat _pi.dd.extracted/0.ext
```

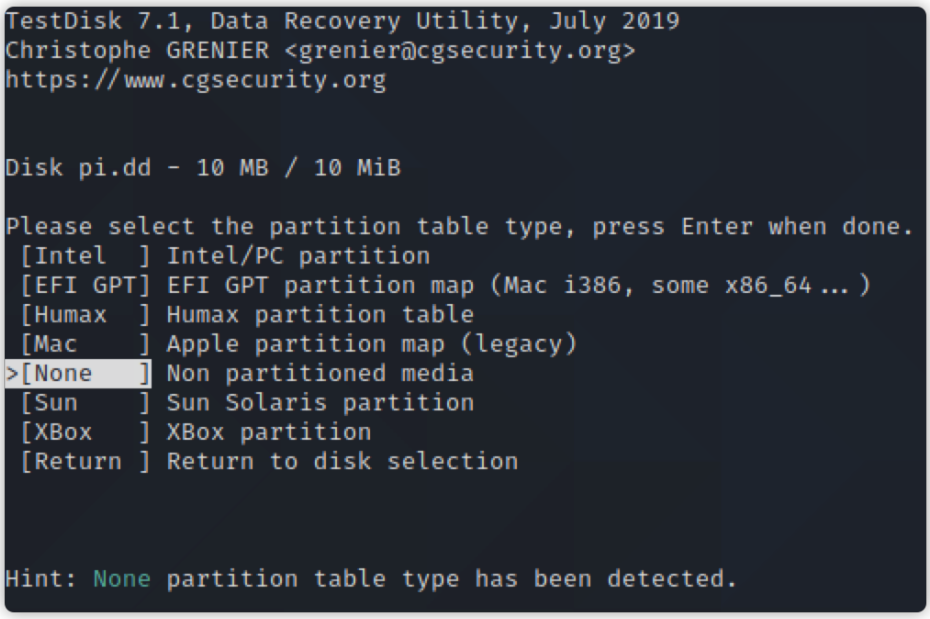
分析后发现并不含已删除文件内容，用 `testdisk` 尝试恢复已删除的文件：

```
$ testdisk pi.dd
```

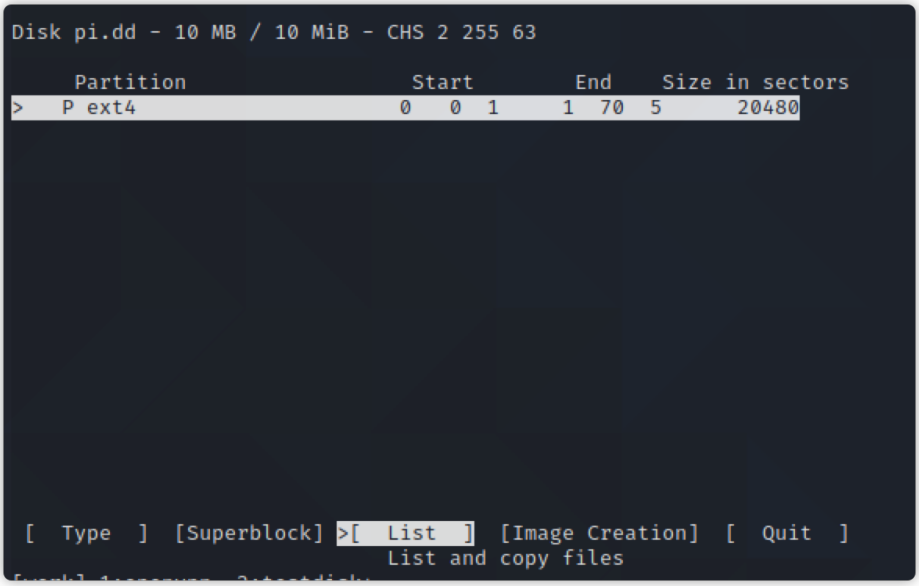
进入后选择要恢复的驱动器，这里只有一个：



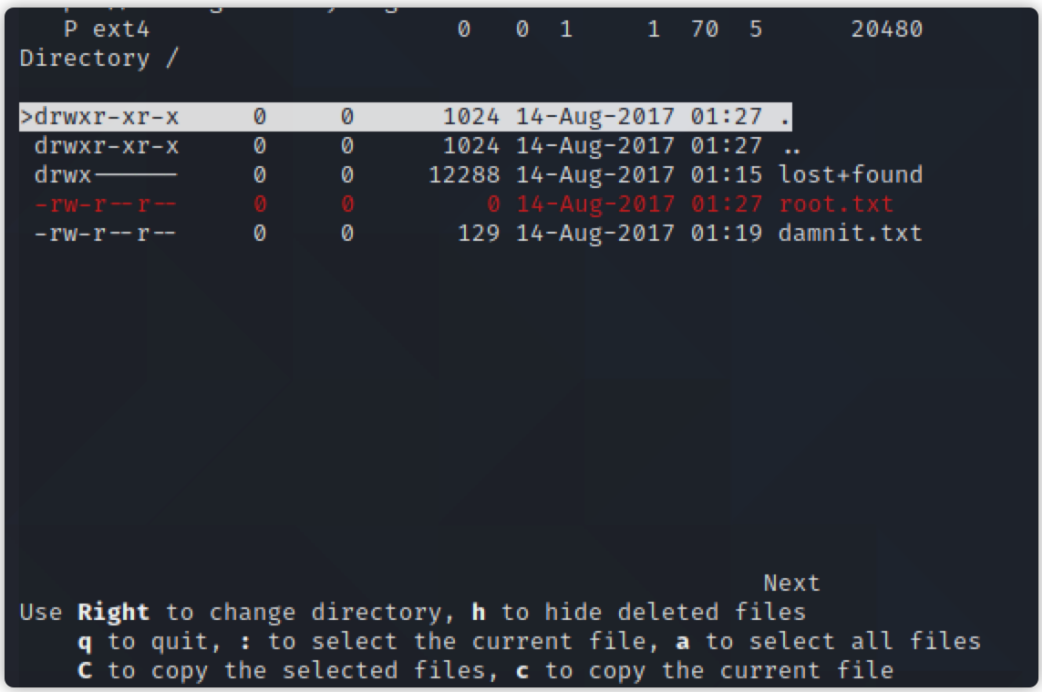
选择分区表类型，这里选择它默认显示的 `None` （系统将自动预测并突出显示最佳选择）。



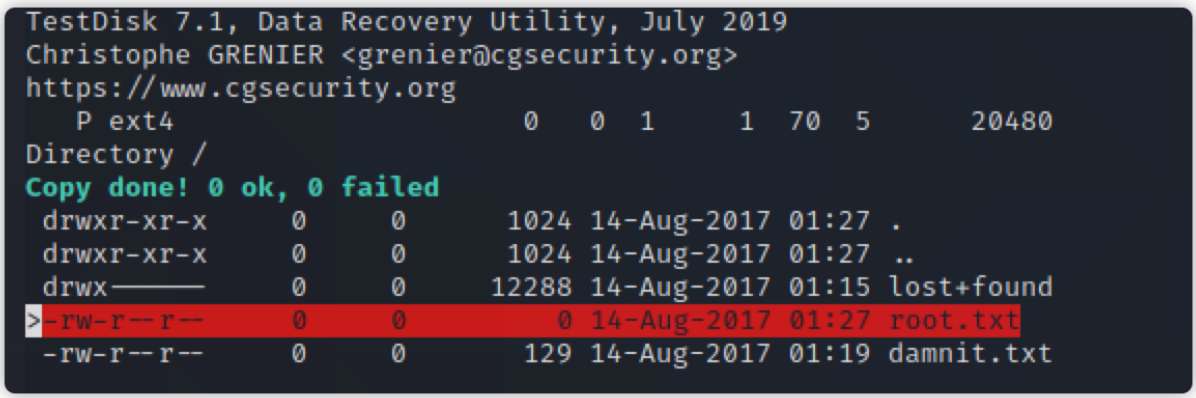
接下来，单击“ ENTER ”继续。选择已删除的文件源驱动器分区。

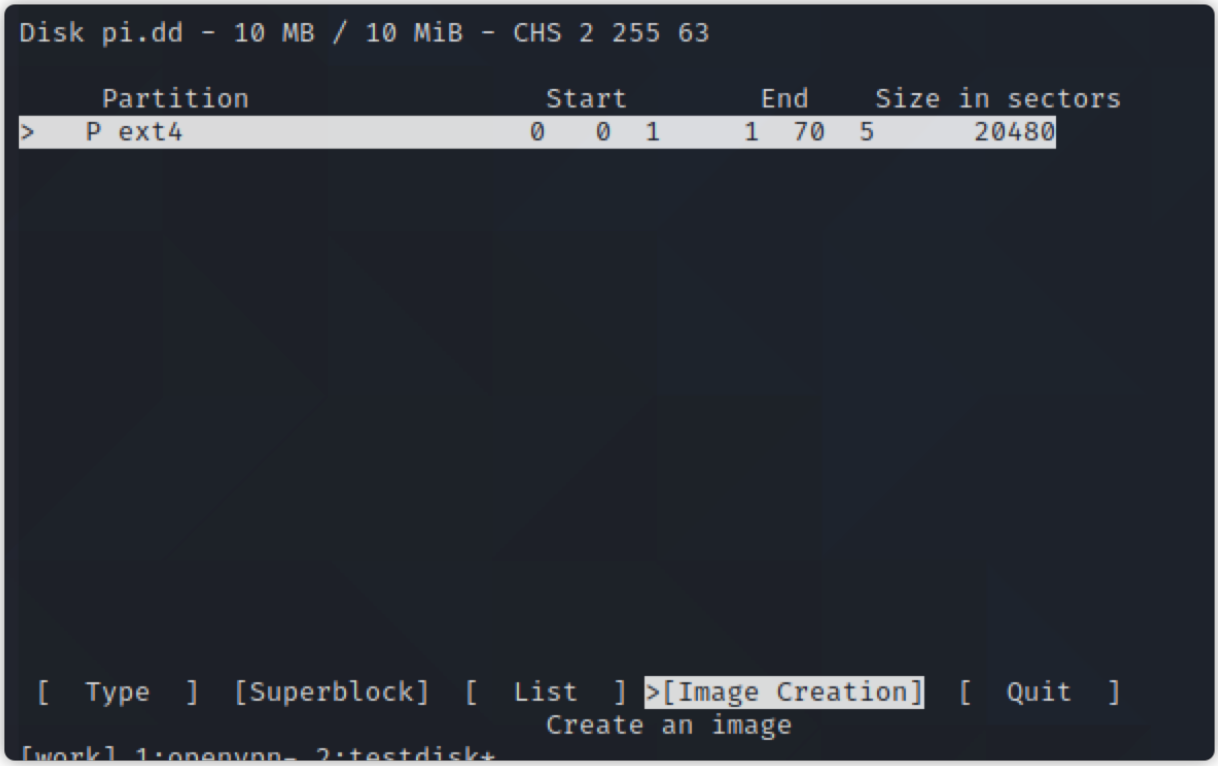


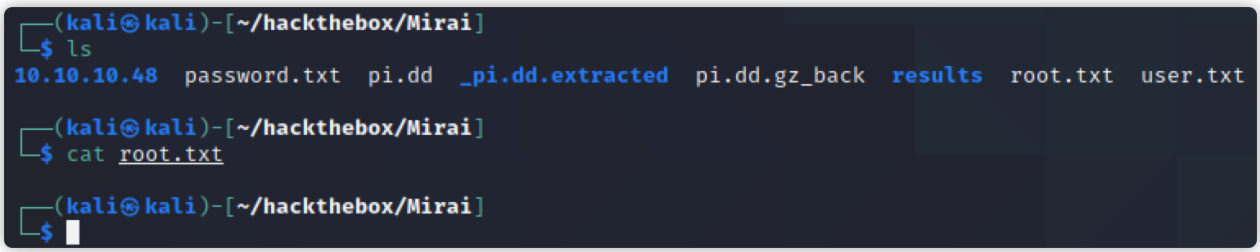
检查已删除的文件源目录，这里可以看到 `root.txt` 已被标识为删除。



键盘上的字母 `c` 复制要还原的文件。如果顺利则会提示 `Copy done!` 的通知。







但是在恢复的步骤中可以明显看到红色的 `root.txt` 文件，文件内容长度为0。所以恢复出来也是没有任何内容的...

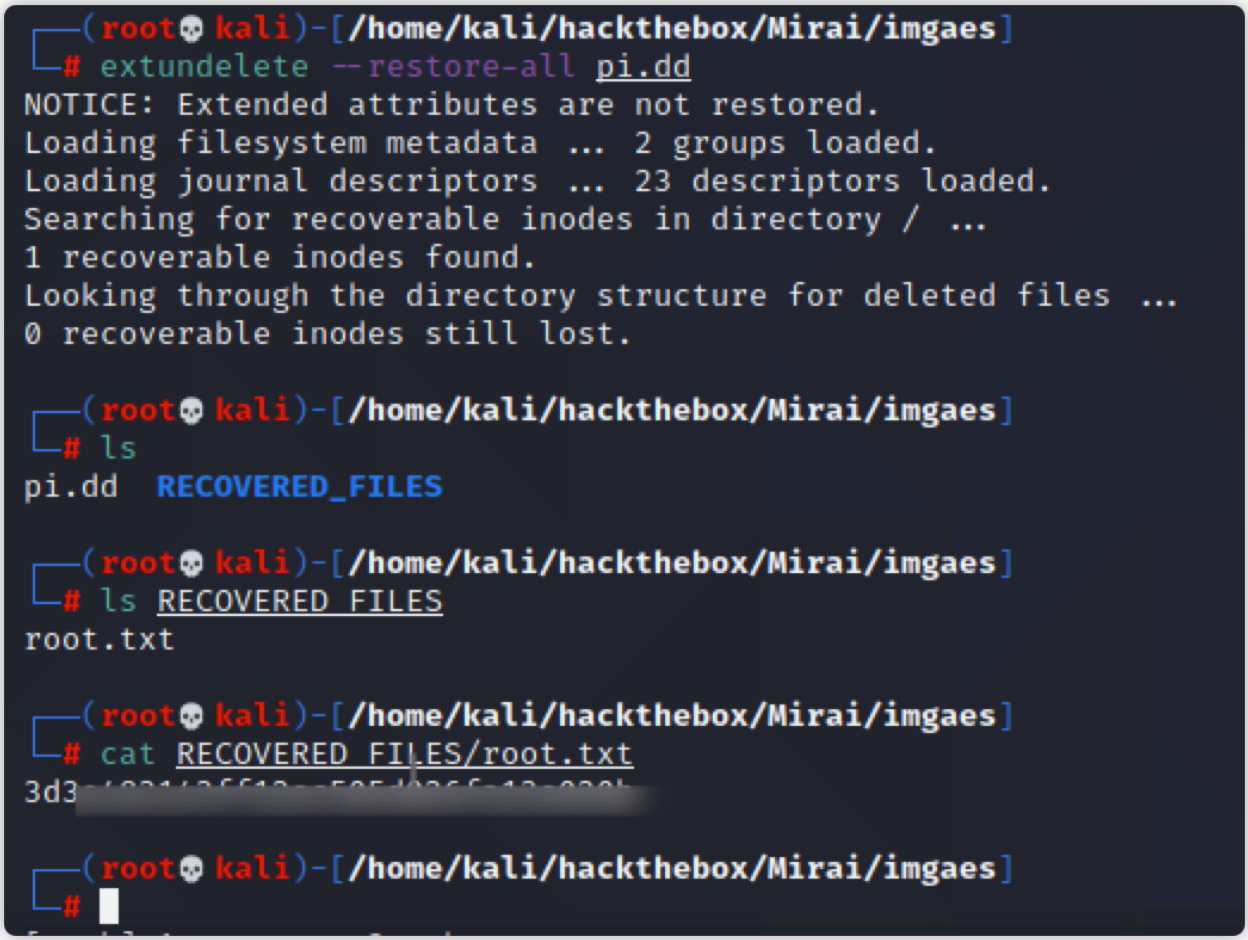
随后又尝试了 `photorec`：

```
$ photorec pi.dd
```

一顿操作后最终只是将磁盘内容导出了，并没有还原已删除的root.txt和内容。看来只有直接strings这种方式了...

接着复盘，又找到了 `extundelete` 这款工具，自动帮我恢复了 root.txt 且内容也在。

> https://mp.weixin.qq.com/s/YTARAIWkoZDm6jo-l2_EPA



参考

- https://mp.weixin.qq.com/s/uqpjC3Dy78RpsV4NGtkCow
- https://hackfun.org/2017/09/07/Kali-Linux%E5%8F%96%E8%AF%81%E5%88%86%E6%9E%90%E4%B9%8BBinwalk/
- https://www.howtoing.com/recover-deleted-files-using-testdisk-in-linux
- https://mp.weixin.qq.com/s/YTARAIWkoZDm6jo-l2_EPA