# **[Moonraker: 1](https://www.vulnhub.com/entry/moonraker-1,264/)**靶机练习实践报告

*下载地址:*

* [http://drive.google.com/open?id=13b2ewq5yqre2UbkLxZ58uHtLfk-SHvmA](https://drive.google.com/open?id=13b2ewq5yqre2UbkLxZ58uHtLfk-SHvmA)
* <https://download.vulnhub.com/moonraker/Moonraker.ova>
* <https://download.vulnhub.com/moonraker/Moonraker.ova.torrent>     ([Magnet](magnet:?xt=urn:btih:EC09FDDCE06DAEF6F11F0DE1BA8EEF91EE1B3448&dn=Moonraker.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))

Flag is /root/flag.txt

Learning Objectives: Client-side Attacks, NoSQL, RESTful, NodeJS, Linux Enumeration and Google-fu.

## 安装靶机

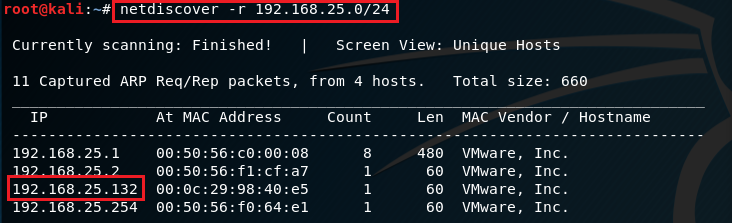
靶机是.ova文件，需要用VirtualBox打开，但我习惯于使用VMWare,因此修改靶机文件，使其适用于VMWare打开。

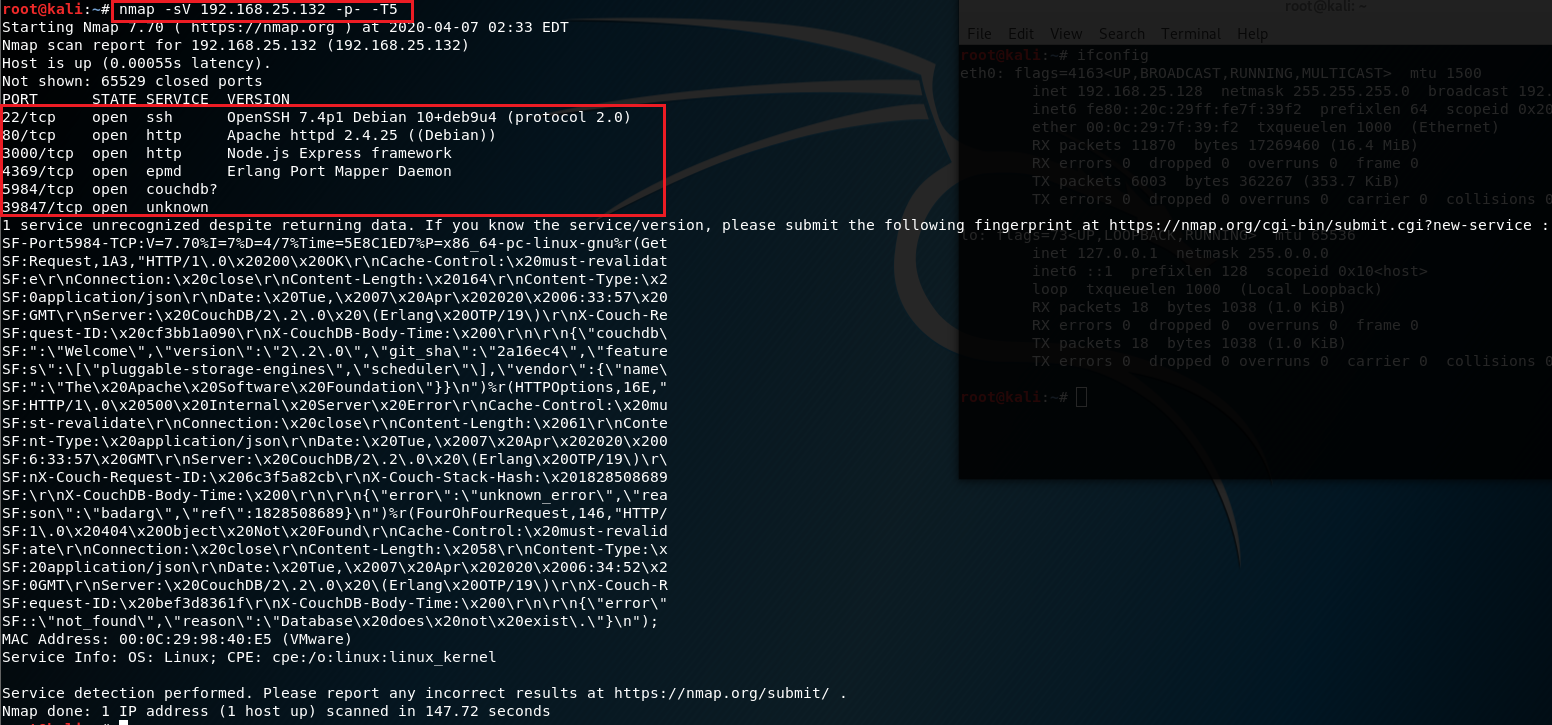
解压ova文件，得到.ovf文件和.vmdk文件。

直接用VMWare打开.ovf文件即可。

## 夺旗步骤

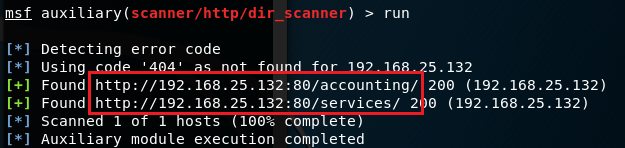
第一步：IP扫描，端口扫描



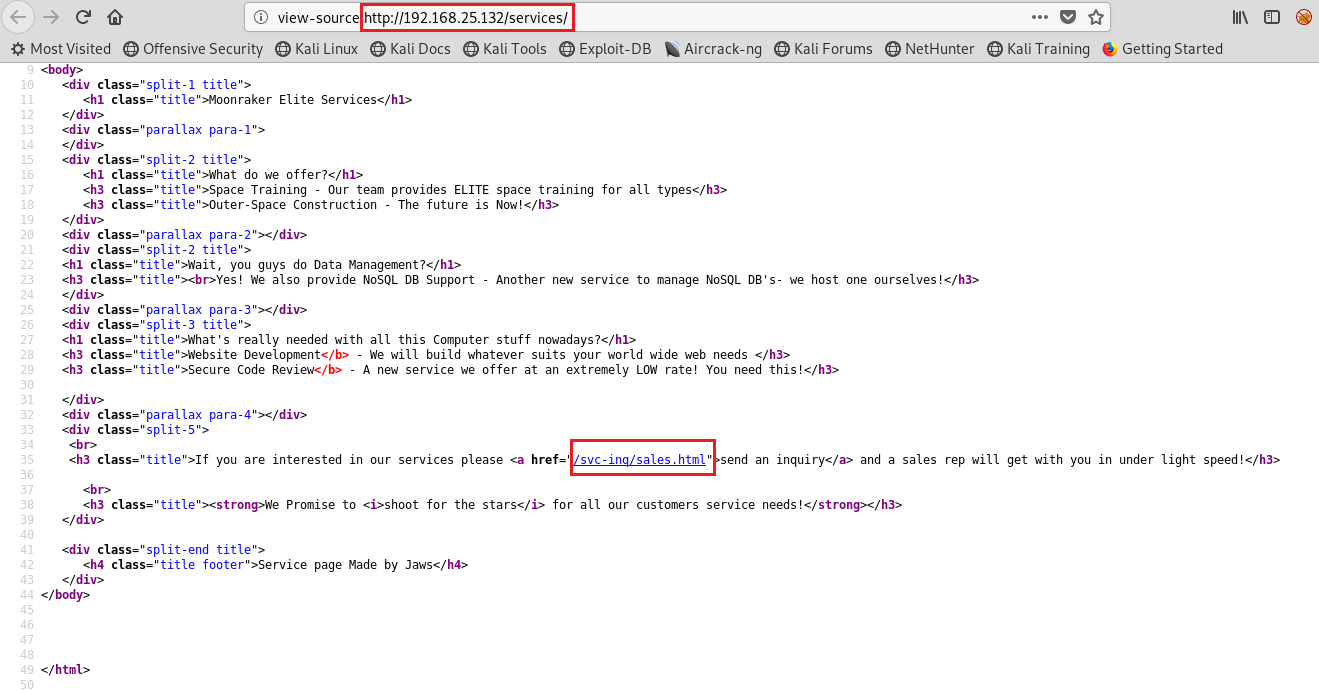


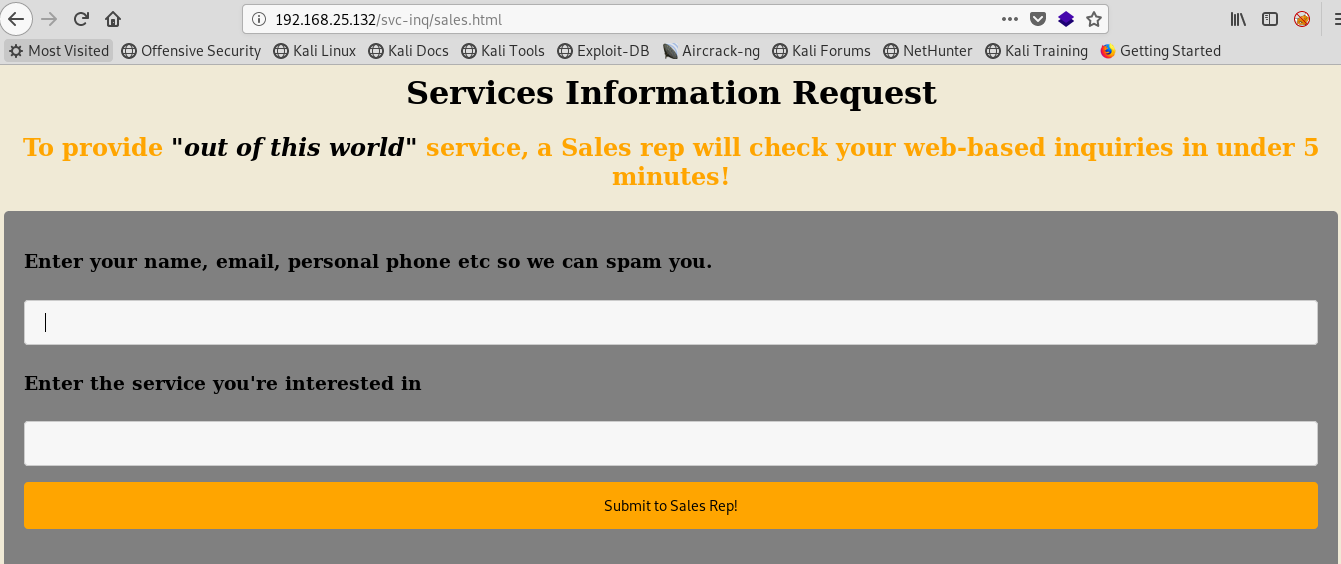
靶机IP为192.168.25.132，开放端口22(ssh),80(apache),3000(nodejs),5984(couchdb)和39847。首先爆破ssh,失败。

第二步：访问80端口、目录扫描和网站渗透测试



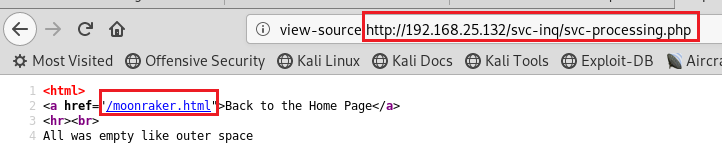
逐个访问上述的地址，http://192.168.25.132:80/services发现一个提示。



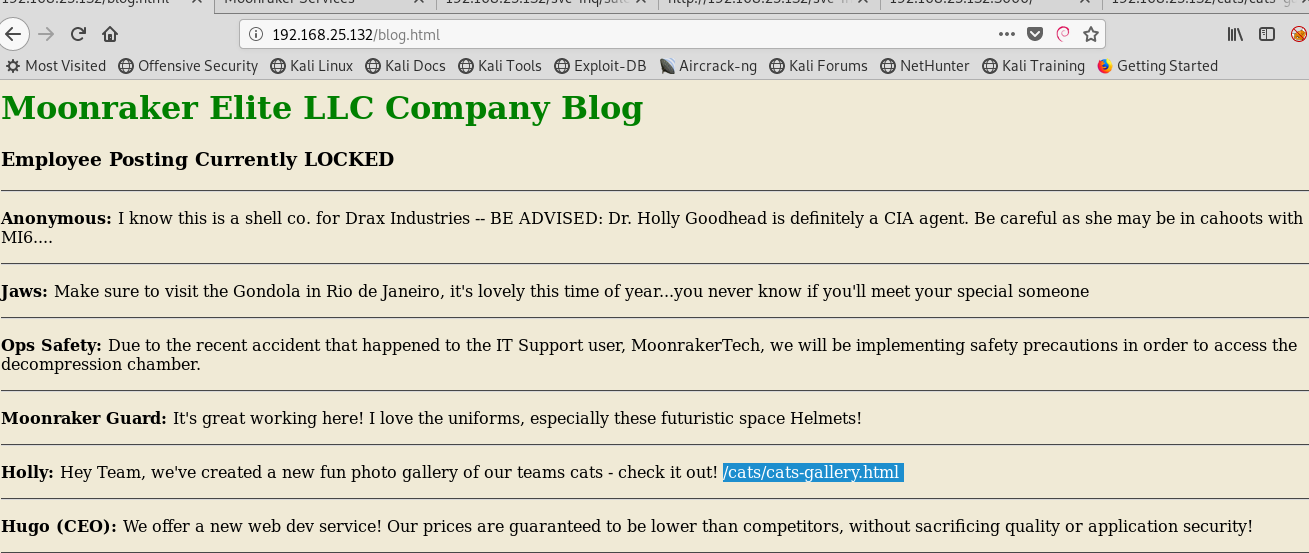


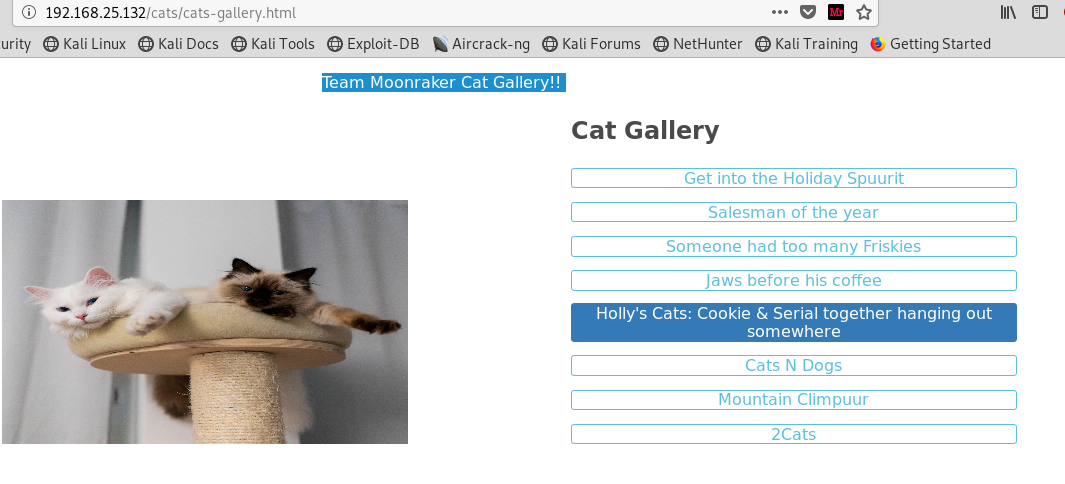
提示销售代表会查看输入的信息，可以伪造网站地址，欺骗用户点击，稍后分析。









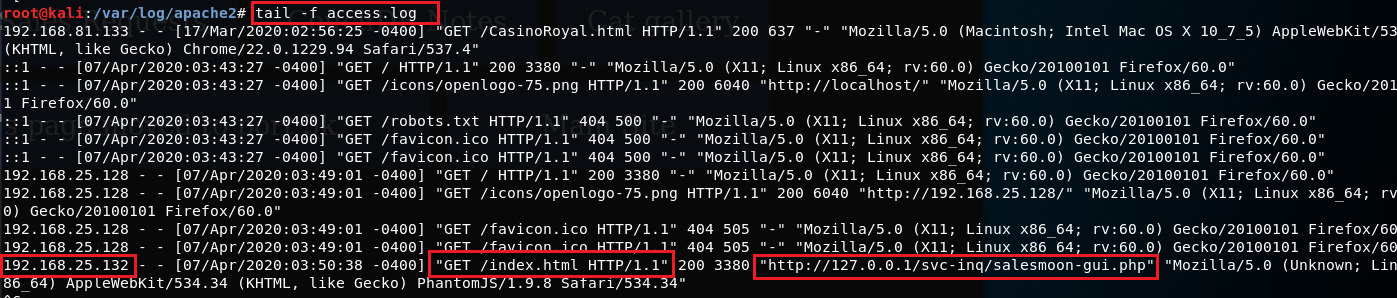


有一个提示：某个地方有cookie和序列化。

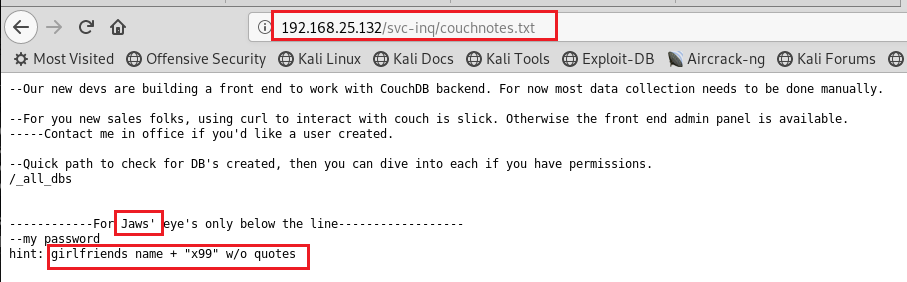
然后，nikto扫描没有发现有价值信息。

第三步:整理收集到的结果，进一步分析。

启动服务器，在之前找到的http://192.168.25.132/svc-inq/sales.html地址处输入伪造的<img src=”http://192.168.25.128:80/index.html”/>,查看访问记录。

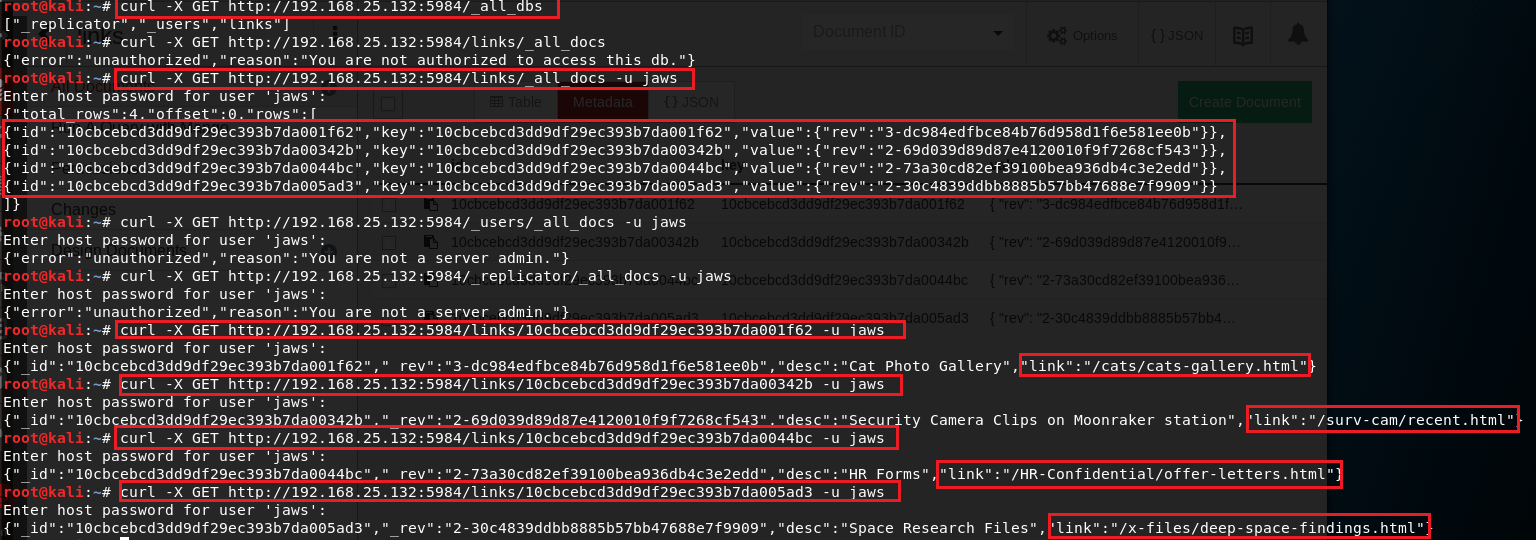


意外发现一个新的地址，打开地址，全站查找信息。发现一个提示用户口令的文件，在ssh和couchdb中尝试该口令，将全站出现的女性名字全部试了一遍，都不是，考点有一个提示“Google-fu.”，难道是google搜索？

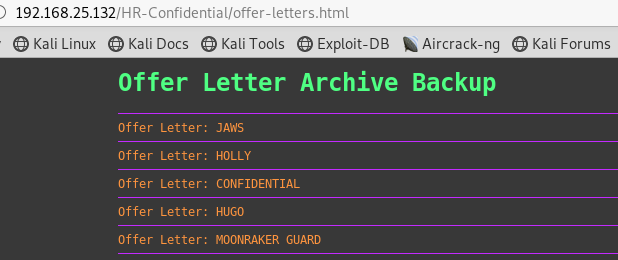




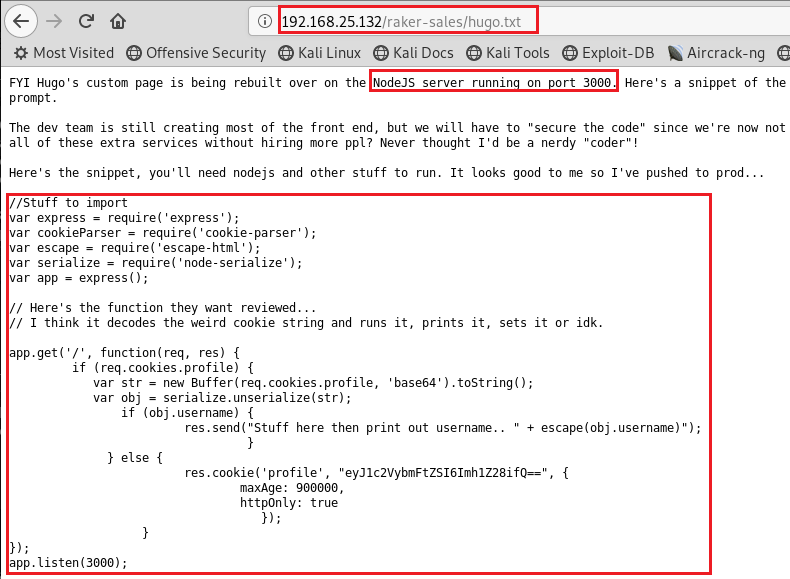
获得用户口令“jaws:dollyx99”,在couchdb中成功登陆。进一步分析couchdb中的数据，发现4个新的地址，逐个访问。



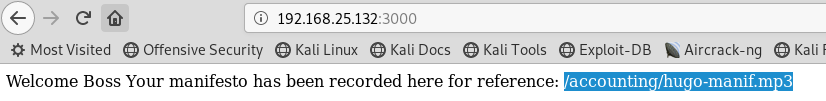
发现一个/HR-Confidential/offer-letters.html页面中有四个用户口令,“hugo:TempleLasersL2K “,”jaws:dollyx99”,“holly:ArchivesPistolsL2K “,“guard:FacProxsL2K”。



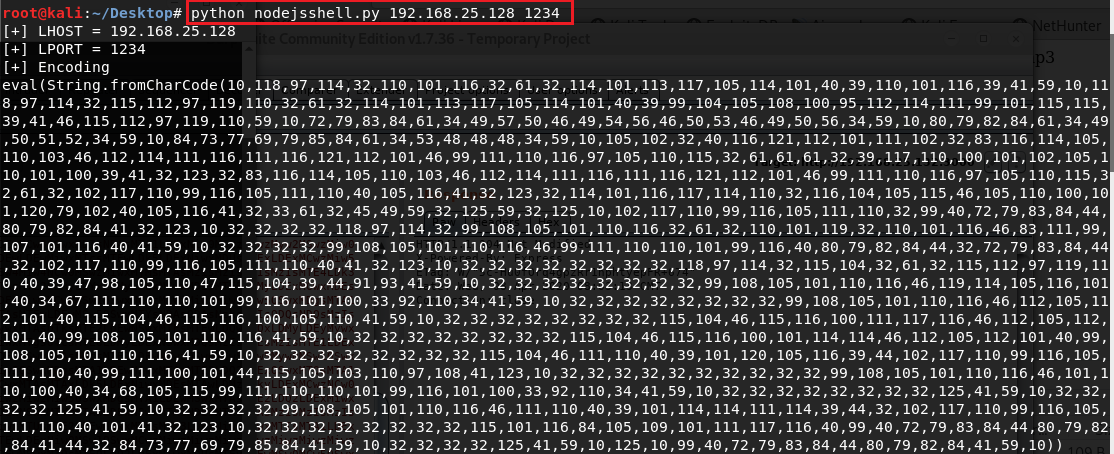
用这四个口令分别登陆ssh，失败，登陆couchdb,没有权限访问的两个数据库还是无法访问。继续分析/svc-inq/salesmoon-gui.php,找到另一个提示文件，提到了端口3000，提到了cookie,我们找到过一个cookie和序列化的提示，难道存在序列化漏洞？分析代码，obj.username可能存在序列化漏洞。



访问3000端口，需要用户口令，用之前获得的4个口令逐个尝试，“hugo:TempleLasersL2K “成功登陆。根据提示访问/accounting/hugo-manif.mp3，什么也没发现。

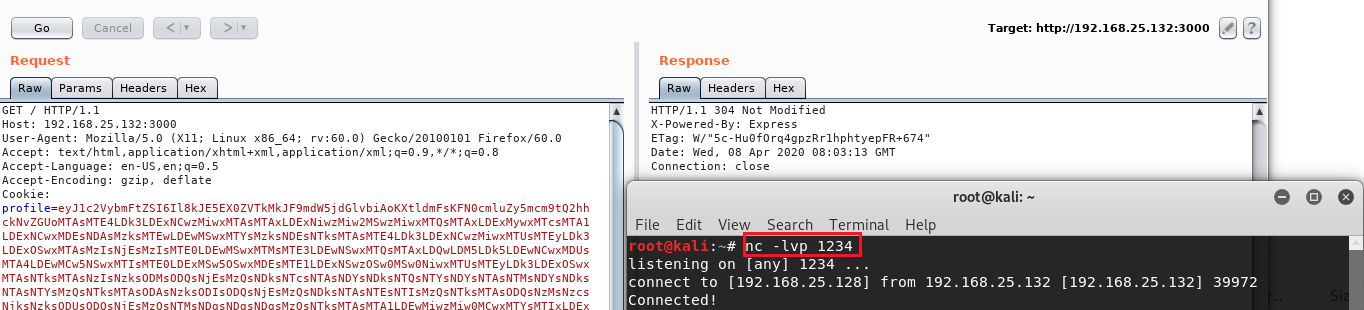


上网搜索nodejs序列化漏洞相关信息，找到一个利用脚本，直接拿来主义。

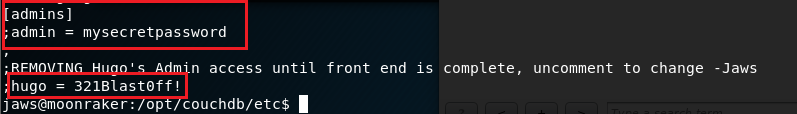




将脚本进行base64是因为nodejs后台代码会对cookie进行base64解码，将cookie替换成shellcode,本地监听设置的端口1234.

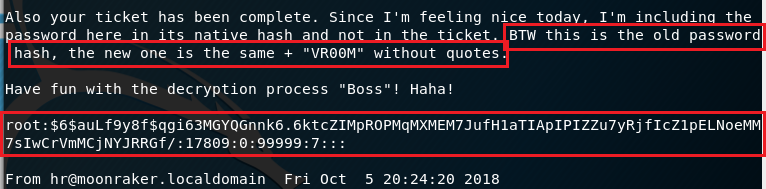


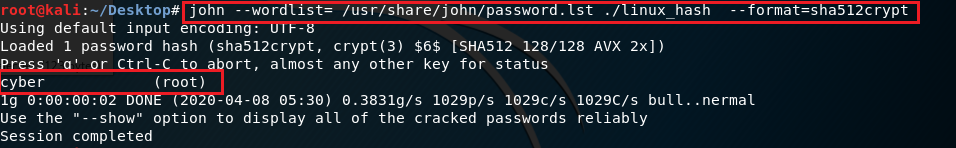
获得一个shell。全目录遍历查看，发现/opt/couchdb/etc/local.ini文件中又发现两个口令。同样在ssh和couchdb中尝试，还在http://102.168.25.132:3000/尝试，发现“hugo:321Blastff!”可以通过ssh登陆，获取hugo权限的shell。



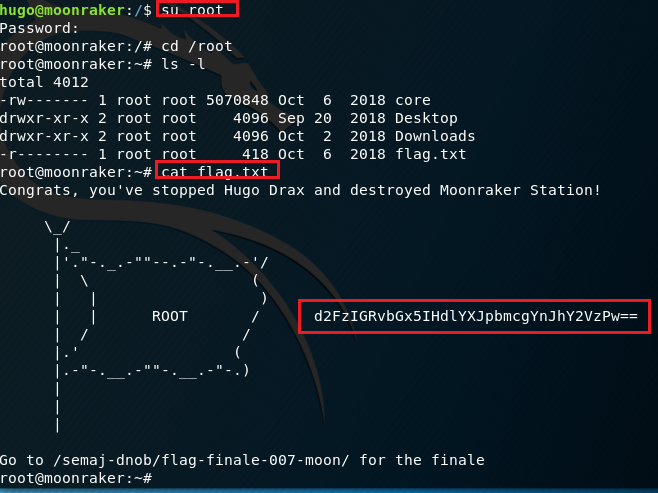


还是无法查看/root目录下的文件，继续搜索，在/var/mail/hugo下查看到一个信息。root用户的密码，条件反射的爆破。





得到root用户新口令”root:cyberVR00M”,登陆root用户，读取flag。



## 总结

1. 灵活运用之前靶机练习中学到的搭建攻击者服务器的办法
2. 常见漏洞的脚本要备齐，此处新收集nodejs序列化漏洞利用脚本