

信号量简介

[beifeng@hadoop-senior datas]$ kill -l

1) SIGHUP 2) SIGINT 3) SIGQUIT 4) SIGILL 5) SIGTRAP

6) SIGABRT 7) SIGBUS 8) SIGFPE 9) SIGKILL 10) SIGUSR1

11) SIGSEGV 12) SIGUSR2 13) SIGPIPE 14) SIGALRM 15) SIGTERM

16) SIGSTKFLT 17) SIGCHLD 18) SIGCONT 19) SIGSTOP 20) SIGTSTP

21) SIGTTIN 22) SIGTTOU 23) SIGURG 24) SIGXCPU 25) SIGXFSZ

26) SIGVTALRM 27) SIGPROF 28) SIGWINCH 29) SIGIO 30) SIGPWR

31) SIGSYS 34) SIGRTMIN 35) SIGRTMIN+1 36) SIGRTMIN+2 37) SIGRTMIN+3

38) SIGRTMIN+4 39) SIGRTMIN+5 40) SIGRTMIN+6 41) SIGRTMIN+7 42) SIGRTMIN+8

43) SIGRTMIN+9 44) SIGRTMIN+10 45) SIGRTMIN+11 46) SIGRTMIN+12 47) SIGRTMIN+13

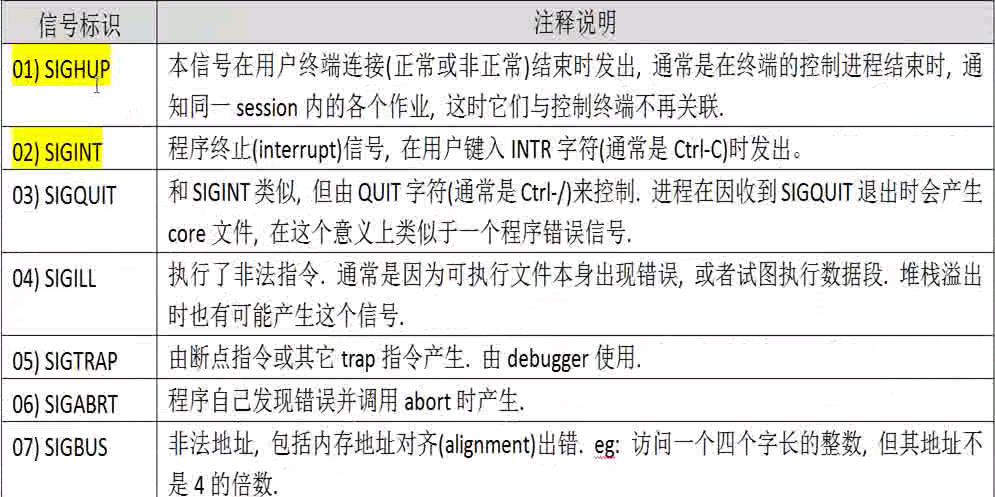
48) SIGRTMIN+14 49) SIGRTMIN+15 50) SIGRTMAX-14 51) SIGRTMAX-13 52) SIGRTMAX-12

53) SIGRTMAX-11 54) SIGRTMAX-10 55) SIGRTMAX-9 56) SIGRTMAX-8 57) SIGRTMAX-7

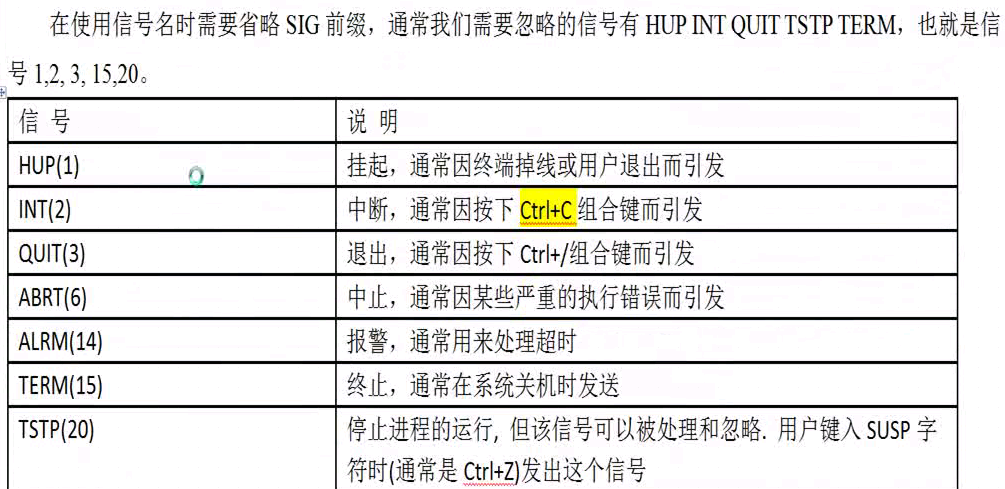
58) SIGRTMAX-6 59) SIGRTMAX-5 60) SIGRTMAX-4 61) SIGRTMAX-3 62) SIGRTMAX-2

63) SIGRTMAX-1 64) SIGRTMAX

[beifeng@hadoop-senior datas]$ trap –l

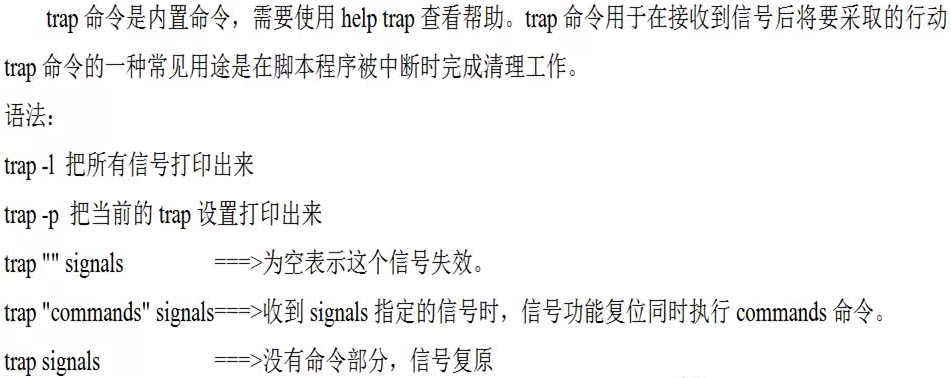


#### 常用信号



# Trap 信号处理

## trap 命令



## 信号与键盘对应

[beifeng@hadoop-senior datas]$ stty -a

speed 38400 baud; rows 36; columns 121; 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; flush = ^O; min = 1; time = 0;

-parenb -parodd cs8 -hupcl -cstopb cread -clocal -crtscts -cdtrdsr

-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 echok

### Ctrl + C 失效

[beifeng@hadoop-senior datas]$ trap -p

[beifeng@hadoop-senior datas]$ trap "" 2

[beifeng@hadoop-senior datas]$ trap -p

trap -- '' SIGINT

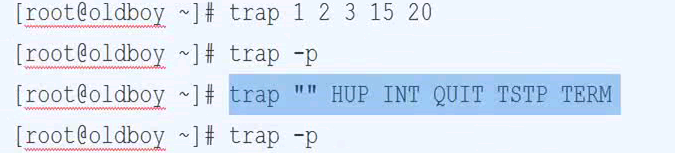
复原：

[beifeng@hadoop-senior datas]$ trap 2

[beifeng@hadoop-senior datas]$ trap "echo -n 'You pressed ctrl+c'" 2

[beifeng@hadoop-senior datas]$ ^CYou pressed ctrl+c

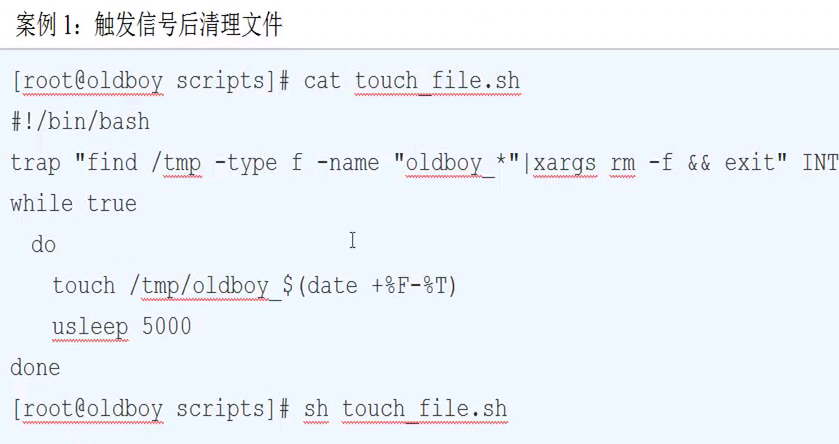








## 脚本示例



#!/bin/sh

trap "find /tmp -type f -name "oldboy\*"|xargs rm -f && exit" INT

while true

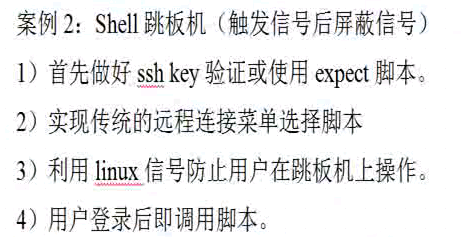
do

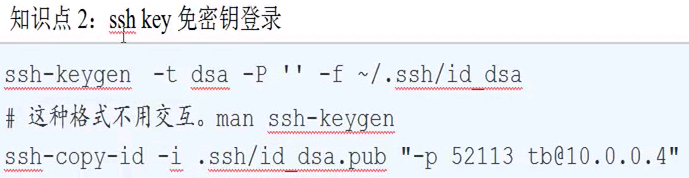
touch /tmp/oldboy\_$(date +%F-%T)

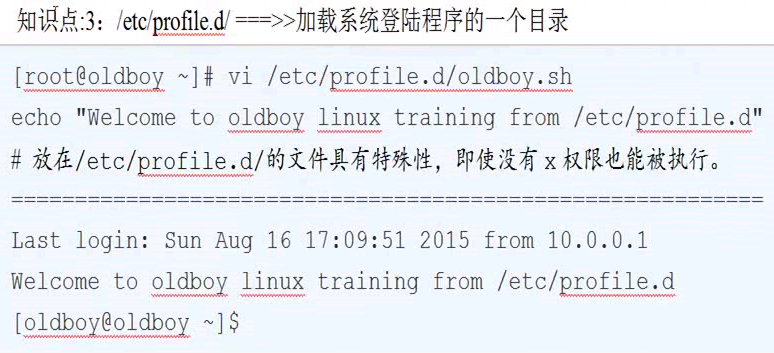
usleep 5000

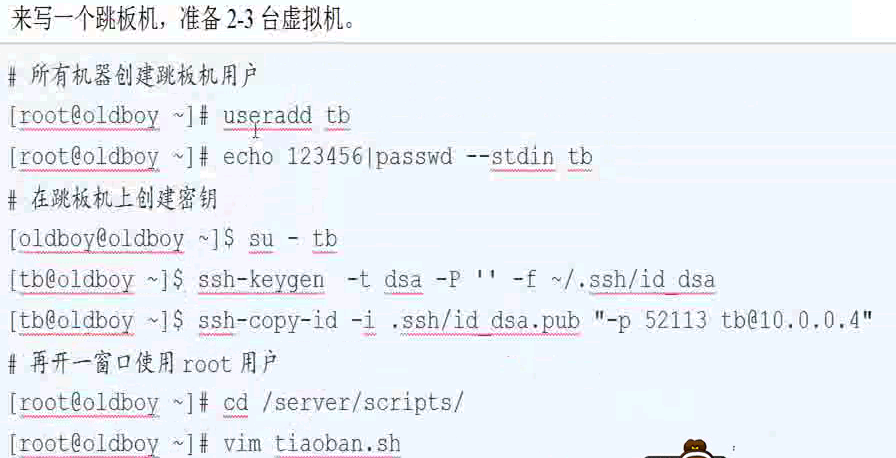
done

## shell 跳板机









[root@hadoop-senior ~]# useradd agent

[root@hadoop-senior ~]# echo 123456|passwd --stdin agent

[root@hadoop-senior ~]# su - agent

Hi Welcome to hadoop cluster !!!

[agent@hadoop-senior ~]$ ssh-keygen -t dsa -P '' -f ~/.ssh/id\_dsa

[agent@hadoop-senior ~]$ ssh-copy-id -i .ssh/id\_dsa.pub [agent@192.168.23.131](mailto:agent@192.168.23.131)

[agent@hadoop-senior ~]$ ssh-copy-id -i .ssh/id\_dsa.pub [agent@192.168.23.132](mailto:agent@192.168.23.132)

vi jump\_sever.sh

#! /bin/sh

function trapper() {

trap '' INT QUIT TSTP TERM HUP

}

function menu(){

#It's a - before EOF

cat<<-EOF

================Server List=================

1)192.168.23.131

2)192.168.23.132

3)exit

============================================

EOF

#It's a tab before EOF

}

function host() {

case "$1" in

1)

ssh $USER@192.168.23.131

;;

2)

ssh $USER@192.168.23.132

;;

3|\*)

exit

esac

}

function main() {

trapper

while true

do

clear

menu

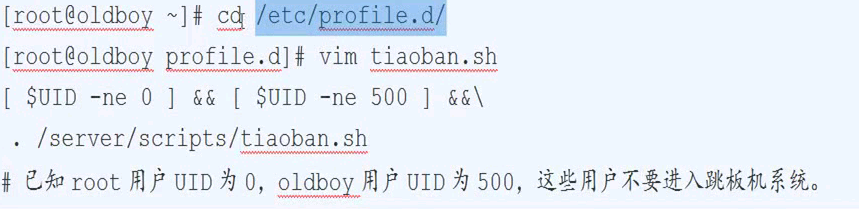
read -p "Pls select :" option

host $option

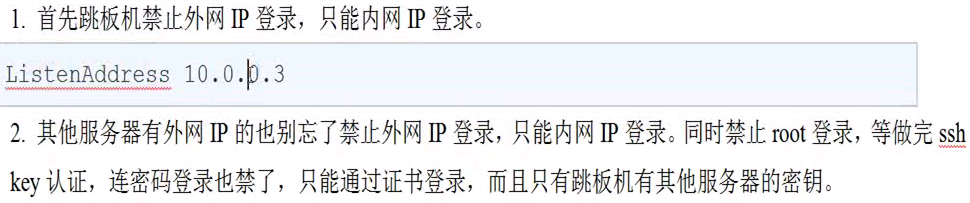
done

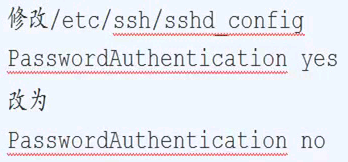
}

main



## 跳板机安全应用







# 其它

**用VMware Workstation 11的LAN区段功能构造内网环境**

2015-03-09 17:24:32

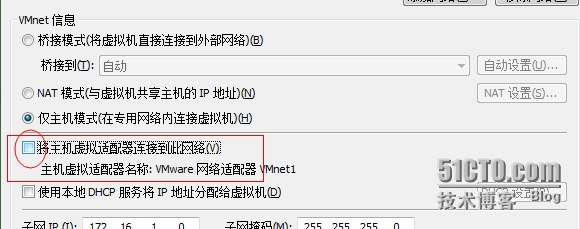
标签：[VMware](http://blog.51cto.com/tag-VMware.html) [LAN区段](http://blog.51cto.com/tag-LAN%E5%8C%BA%E6%AE%B5.html) [内网环境](http://blog.51cto.com/tag-%E5%86%85%E7%BD%91%E7%8E%AF%E5%A2%83.html)

原创作品，允许转载，转载时请务必以超链接形式标明文章 [原始出处](http://life2death.blog.51cto.com/7550586/1618744) 、作者信息和本声明。否则将追究法律责任。<http://life2death.blog.51cto.com/7550586/1618744>

原来用host only模式构造内网环境，只是虚拟机访问不了外网，但其实主机还是能连上这个网段，不能很好的模拟真实环境。

[wKiom1T9ZX3ShmuOAAAkMN7EWXw481.jpg](http://s3.51cto.com/wyfs02/M02/5A/6E/wKiom1T9ZX3ShmuOAAAkMN7EWXw481.jpg)

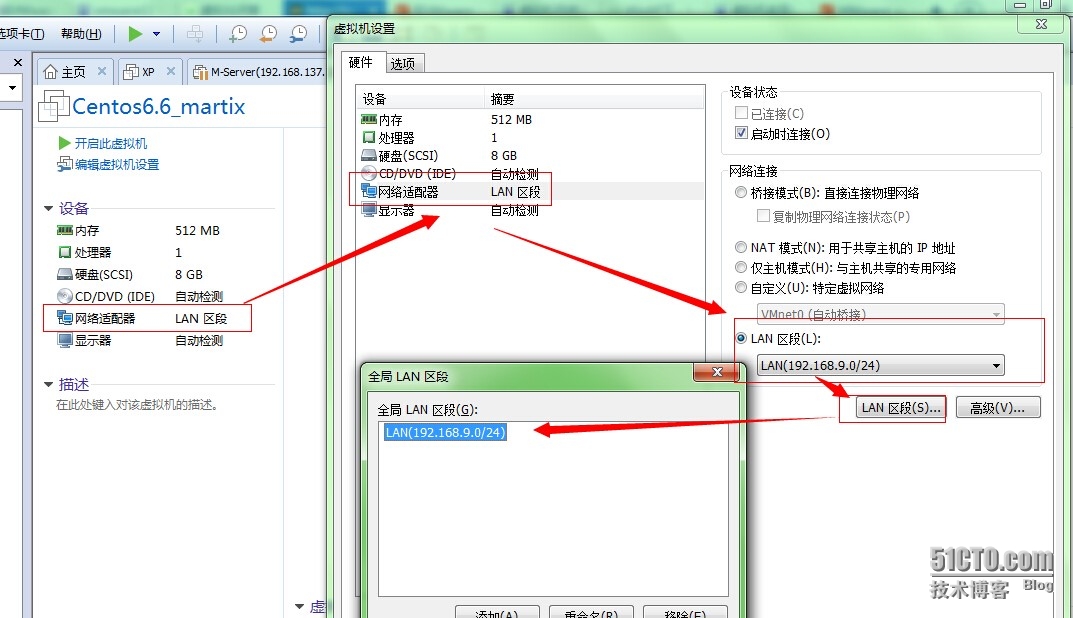
后来有人和我说了一种方法，把这个复选框钩去掉，就可以让主机访问不了这个网段

[](http://s3.51cto.com/wyfs02/M00/5A/6A/wKioL1T9Zpmh6uHwAAFdk2A-Qso373.jpg)

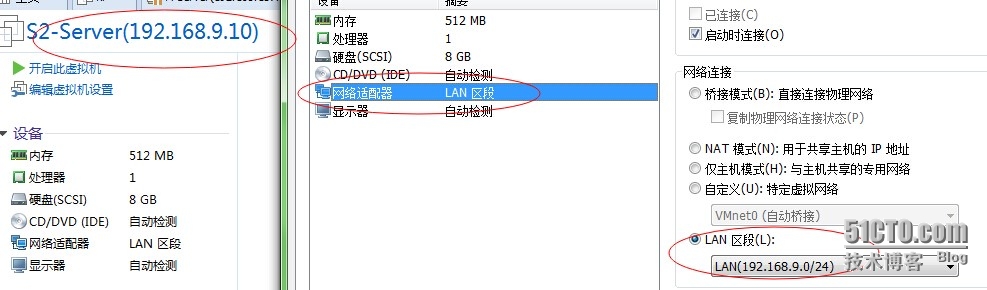
但是这个方法有个缺点，当需要内网划分多个子网时，就要创建多个host only的虚拟网卡，就要在系统里生成多个虚拟网卡，老版本的VMware总共支持10个虚拟网卡，新版本VMware11支持20个，扣除VMnet0、VMnet1、VMnet8，就分别只能构造7个或17个自定义网络

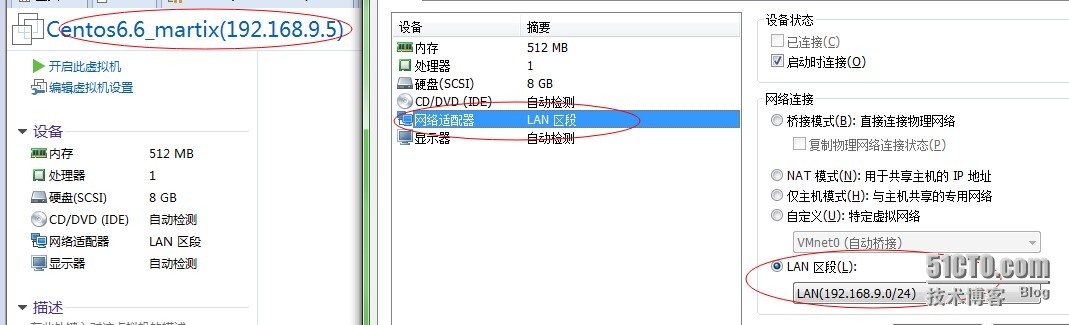
而采用LAN区段方式则没有限制，挺方便的。

**1、配置LAN区段，就是命个名，不用配置网络相关的**

[](http://s3.51cto.com/wyfs02/M00/5A/6A/wKioL1T9ZpmT1Rl7AAQcqPMBrwU446.jpg)

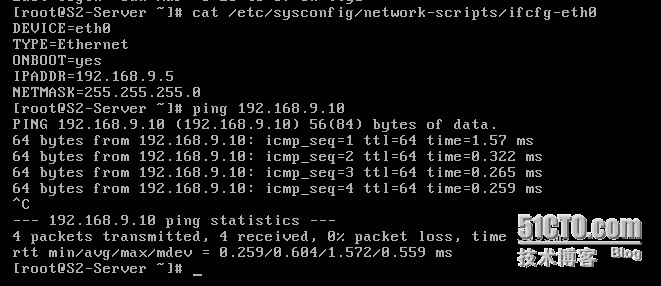
**2、想要在同一内网的机器选择同一个LAN**

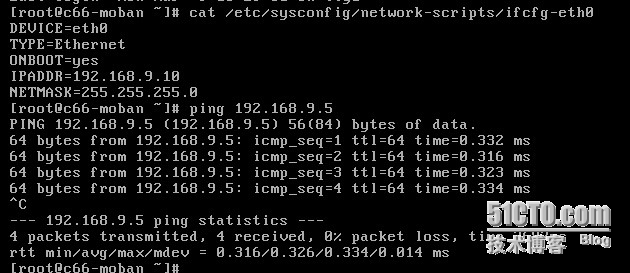
[](http://s3.51cto.com/wyfs02/M00/5A/6E/wKiom1T9ZX6A7Ze_AAJLZLFjCDQ373.jpg)

[](http://s3.51cto.com/wyfs02/M00/5A/6E/wKiom1T9ZX6iftPKAAKEIylzjFw987.jpg)

**3、最后仅需要在网卡配置一下就可以了，无关的网卡配置我清除了，保留5行好对比**

3.1 互相能够ping通

[](http://s3.51cto.com/wyfs02/M01/5A/6A/wKioL1T9Zpqz9I_0AAIFGDfW8FU630.jpg)

[](http://s3.51cto.com/wyfs02/M01/5A/6E/wKiom1T9ZX7xjBI-AAHwNelP8Bo429.jpg)

3.2 但主机上ping这2台虚拟机不通

[](http://s3.51cto.com/wyfs02/M02/5A/6A/wKioL1T9ZprQThOEAAHXeYDt8gM024.jpg)

3.3 想在主机上连上内网机器，就需要通过网关了，或者2台虚拟机加块网卡