linux□□

- linux

linux

一、 [系统工具]

```
whoami |which |id |who |users |whereis |updatedb |locate |cat /etc/redhat-release | |free -h|
df -h |du -sh
env |printenv PATH |echo $SHELL |tty |uptime |getent hosts 127.0.0.1 |lsblk |iosta |wc -l
|uptime |more /etc/os-release
reboot |shutdown -h now |poweroff
more /etc/shells 支持那些shell
echo $PATH
echo $SHELL
```

```
|hostname |hostnamectl |hostnamectl set-hostname 147
uname {-r;-n;-a;-m}
df {-h;-i;-hT}
wc {-l;-w;-c;-m}
```

二、 [查看操作系统版本]

```
cat /etc/redhat-release
cat /etc/os-release
内核版本
uname {-r;-n;-a;-m}
```

三、 [查看 cpu信息]

```
lscpu
top
htop
uptime
ps aux --sort=-%cpu | head -20
```

四、 [查看内存信息]

```
free -h
vmstat 1
```

五、 [磁盘]

```
lsblk
fdisk -l
df -h
du -sh
find -name -size +100M file -delete
pgrep sshd
lsof {-p;-u;-i} {pid;root;:port}
dnf install -y sysstat iotop
iostat -x 1 5
iotop -o
```

六、 [通用]

```
mount {-a;-o;-l} | umoun
ln -s 绝对path1 绝对path2 软连接 path1(被保护)
```

七、 [引用 |变量 |函数]

```
export aaa='bbb'
alias aaa='bbb'
aaa() {bbb "$@"}
$aaa ;${aaa};$"aaa";``
$?;$#;$@;$*;$$
```

```
aaa=${aaa//旧内容/空}
aaa=${aaa/旧内容/新内容}
aaa=${aaa//旧内容/新内容}
aaa=${aaa%字符}
```

```
aaa=${aaa#字符}
```

八、 [数组]

```
()
aaa+=(d)
unset aaa-[1]
${aaa[@]} | ${!aaa[@]} | ${#aaa[@]}
```

九、 [系统基础命令]

```
ls {-a;-l;-r;-t;-F}
cd {-;..;../..}
mkdir {-p;-m 755;-v}
rmdir {-p}
rm {-r;-rf}
mv {-i;-v;!(排除项)} -->shopt -s extglob --> !()
touch {-t YYYYMMDDhhmm.ss}
bash {-n;-x}
```

十、 [复制]

```
cp {-r;-i;-v}
rsync {-av;--delete} {源文件 目标路径; }
scp -r 源文件 目标路径
```

七、 [打印]

```
echo {-n;-e}
printf {"%3d";"%03d";"%-3d";"%6s";"%-6s";"%.2f";"%-10s %5d\n";"%-10s %5s\n";"%.3f%%\n}
```

八、 [重定向]

```
tee {-a} file{1..9}
>
>>
```

□□ [□□]

```
find path {-name;-mtime;-ctime;-atime;-type;-size;-iname;-empty} {-or;-and;-delete;-exec cmd
{} \;;xargs cmd}
grep {-C;-A;-B;-i;-v;-n;-r;-c --include=;-P}
```

□□ [□□]

```
cat {-E;-n;-A;>;<<EOF}
tail {-f;-n 10;-fn10}
head {-n;-5;-c 100M /dev/zero} | dd if=/dev/urandom of=100M.file bs=100M count=1
more
less
```

□□□ [□□]

```
sed {-E;-e;-i;-n} {s;d;i;a;p;=;q} □□sed -i "$(($( wc -l < file ) - 2 )),\$d" file | sed -i "$d"
file
awk -F: '$NR>10 {print $1,$2,$NF}'
cut -d '.' -f 1 = awk '{print $1}'
vim {+10} {:%s/a/b/g;:set nonu} {vim ~/.vimrc}
```

□□□ [□□]

```
sort {-r;-n;-u;-f}
tac = sort -nr
uniq {-c;-d;-u;-i}
column {-t;-s ",";-R;-c} {xargs -n 5 < file | column -t}
xargs {-n 5;-I {};}
```

十七、 [随机数]

```
shuf -i 1-35 -n5
echo $RANDOM
```

十八、 [循环]

```
seq 1 10 | xargs -I {}
for i in {1..10}; ; done
for ((i=1;i<=10;i++));do ;done
i=1;while ((i<=10));do ;((i++));done
while read {-s;-p} ;do ;done
```

十九、 [判断]

```
if cmd; then ;fi
if cmd; then ;elif cmd; then ;else cmd ;fi
[] {-z;-n;-f;-d;-e;-eq;-ne;-gt;-ge;-lt;-le;==;!=;}
[[]] 正则 && || ;;
(()) {>;<;>=;<=;==;++}
{!;-o;-a}
case $ in; $);;$);;esac
```

二十、 [下载]

```
wget {-O}
curl {-o;-s;-X;-H;-d}
```

```
rpm {-Uvh;-ivh--nodeps;-e;-q;-qa;-ql;-V;--import}
yum install
```

十七、 [历史命令小工具]

```
echo -e 'export PROMPT_COMMAND="history -a"\nexport HISTFILESIZE=10000\nexport HISTSIZE=10000'
>> ~/.bashrc && source ~/.bashrc
tail -f ~/.bash_history
```

十八、 [用户]

```
adduser/useradd {-r}
userdel {-r}
passwd {-l;-u;-e}
chage {-d 0;-d 2026-04-25;-l;-E 2026-04-25;-M;-m;-W 14}
su - user001
sudo {echo 'devops ALL=(ALL) NOPASSWD:ALL' >> /etc/sudoers}
{/etc/passwd;/etc/shadow;/etc/group;/home/name;/var/spool/mail/name}
```

十九、 [组]

```
groupadd
groupdel {-f}
usermod {-aG;-G} group user
gpasswd {-a;-d} user group
```

二十、 [权限]

```
chmod {u/g/o+-r/w/x;-R 755;-x;}
chown {-R;user;:user;root.root}
umask
```

二十一、 [运行进程]

```
ps {-ef;-aux}
UID          PID    PPID  C STIME TTY          TIME CMD
top {-d 1;-p PID;-u user;-n 2;-b;} {P;M;N;T;1;E;e;H;i;k;q; ;}
pgrep {-l;-a}
kill {-2;-9}
pkill = pgrep + kill
```

```
ss {-ant;-nutlp;tln}
netstat
nohup 命令 &|&|ctrl+z-->jobs-->bg|fg
```

二十六、 [日志]

```
journalctl {-u;-f;--vacuum-time=2weeks;--disk-usage}
logrotate {-d;-f}
vim /etc/logrotate.d/nginx
/var/log/nginx/*.log {
    daily
    missingok
    rotate 15          # 保留15天的日志，超过自动删除
    compress           # 旧日志gzip压缩节省磁盘
    delaycompress
    postrotate
        /bin/systemctl reload nginx > /dev/null
    endscript
}
```

```
find /var/log/nginx -type f -mtime +15 -name "*.log-*" -delete
```

二十七、 [时间]

```
date {-s;+"%Y-%m-%d %H:%M:%S";+%F;+%T;-d "-1 day" +%F}
ntp|chrony
ntpq -p
chronyc {tracking;sources -v}
```

二十八、 [刷新]

```
crontab {-e;-l;-r}
watch {-n;-d;-t}
at
```

二十九、 [selinux]

```
getenforce {sestatus}
setenforce 0 {1}
vim /etc/selinux/config {enforcing;permissive;disabled}
sed -i 's/^SELINUX=.*/SELINUX=disabled/' /etc/selinux/config
```

□□□ [□□□]

```
firewall-cmd {--permanent;--add-service=http;--zone=public;--reload;--state;--list-services;--add-port=8080/tcp;--list-all}
[--add-service= ; --remove-service= ; --add-port= ; --remove-port= ; --reload]
iptables -L
iptables -A INPUT -p tcp --dport 22 -j ACCEPT
```

□□□ [ufw]

```
ufw status numbered
ufw status verbose
ufw show added
```

```
ufw {allow;deny;reset;enable;disable;reload}
```

□□□ [yum|dnf|□□]

```
yum -y {install;remove;list installed;list installed;list available;update;search;info;repolist all;deplist;provides} {--enablerepo=;--disablerepo=}
yum clean all && yum makecache
dnf clean packages
dnf module {list;remove;reset;enable;install;switch-to;distro}
dnf {check-update;list updates;upgrade}
```

□□□ [□]

```
dnf repolist enabled
dnf repolist all
dnf config-manager --set-enabled □
```

```
dnf config-manager --set-disabled 库
dnf install --enablerepo
dnf list --available mysql-community-server --showduplicates
```

三十三、 [模块]

```
dnf module enable <模块名>:<流>
dnf module disable <模块名>
dnf module reset <模块名>
dnf module list <模块名>
dnf module install <模块名>:<流>
dnf module remove php -y
dnf module switch-to php:8.2 -y
dnf distro-sync php -y
```

三十四、 [createrepo]

```
yum -y install createrepo
mkdir -p /mnt/repo && createrepo /mnt/repo
cat > /etc/yum.repos.d/local.repo <<EOF
[local]\nname=Local Repo\nbaseurl=file:///mnt/repo\ngpgcheck=0\nenabled=1
EOF
```

三十五、 [服务]

```
systemctl {start;stop;restart;status;enable;disable;is-active;reload;is-enable;kill;daemon-reload;list-units --type=service}
```

```
vim /etc/systemd/system/name.service
[Unit]
Description=自定义服务名称与功能描述
After=network.target network-online.target□□
Wants=network-online.target
[Service]
ExecStart=程序启动路径  (/usr/bin/python3 /opt/file.sh)
ExecStop=程序关闭路径
```

```
ExecRestart=程序重启路径
ExecReload=程序热加载路径
Type=simple  (forking)
Restart=on-failure
[Install]
WantedBy=multi-user.target
```

三十六、 [压缩 |打包]

```
zip {-r}    □□unzip {-d;-l}
gzip {-c;-d}□□gunzip
bzip2 {-d} □□□bunzip2
tar {-zcvf;-zxvf;-jcvf;-zxvf;cvf;xvf;ztvf} -C path
```

三十七、 [密钥]

```
ssh-keygen -t rsa {~/.ssh/}
ssh-keygen -t rsa -b 4096 -C "libai"
ssh-copy-id -i /root/.ssh/id_rsa.pub root@192.168.20.146
ssh root@192.168.20.146
```

三十八、 [限制 root用户登录数量]

```
echo 'root hard maxlogins 2' >> /etc/security/limits.conf
/root/.ssh/
```

三十九、 [全局终端闲置超时]

```
echo 'export TMOUT=3600' >> /etc/profile && source /etc/profile
```

四十、 [root远程登录]

```
vim /etc/ssh/sshd_config
ClientAliveInterval 0
```

```
PermitRootLogin yes
```

四十一、 [root远程登录]

```
vim /etc/ssh/sshd_config {PermitRootLogin no}
```

四十二、 [重置 root密码]

```
开机页面,开机时快速按 ESC 或 方向键 中断启动流程
e
linux16 ro --> rw (获得读写权限)
quiet-->rd.break 进入紧急模式
ctrl + x 进入紧急维护模式里面(switch_root:/#)
mount -o remount,rw /sysroot 将根目录挂载为可读写
chroot /sysroot 切换到根目录的一个环境,进入到系统当中(sh-4.2)
passwd root  (注意小键盘失灵的问题)
touch /.autorelabel 重建 SELinux 标签标记 (fixfiles relabel)
exit (switch_root:/#) 退出chroot环境
exit (系统会自动重启)
```

四十三、 [ubuntu]

```
开机页面,开机时快速按 ESC 或 方向键 中断启动流程
e
linux16 ro --> rw (获得读写权限)
init=/bin/bash
passwd root  (注意小键盘失灵的问题)
exit(会显示内核崩坏)
```

```
Advanced options for Ubuntu
Ubuntu, with Linux 5.15.0-181-generic (recovery mode)
passwd root  (注意小键盘失灵的问题)
exit
```

四十四、 [mail邮件]

```
dnf install -y mailx
cat >>/etc/mail.rc <<'EOA'
set from=15090023916@139.com
set smtp=smtps://smtp.139.com:465
set smtp-auth-user=15090023916@139.com
set smtp-auth-password=a8efec1183f4815ffe00
set smtp-auth=login
set ssl-verify=ignore
set nss-config-dir=/etc/pki/nssdb/
EOA
echo "正文" | mail -v -s "标题" -a file 15090023916@139.com
ctrl + D
```

```
dnf install -y mutt
echo "正文" | mutt -s "标题" -a 10wan_mysql_insert -- 15090023916@139.com
```

四十五、 [部署共享文件夹]

```
nfs服务部署
服务端
yum -y install nfs-utils
rpm -q nfs-utils
mkdir /opt/share && cd /opt/share && ll
echo '/opt/share 192.168.20.145/24(rw,sync,no_root_squash)' > /etc/exports
systemctl start nfs-server &&systemctl enable nfs-server && systemctl status nfs-server
客户端
mount -t 文件系统类型 源设备 挂载点 {临时挂载}
mkdir /opt/nfs && cd /opt/nfs
mount -t nfs 192.168.20.146:/opt/share /opt/nfs
```

```
mkdir /mnt/nfs   #永久挂载
echo '192.168.20.146:/opt/share  /mnt/nfs  nfs  defaults  0  0' >> /etc/fstab
systemctl daemon-reload && mount -a && df -h
```

四十六、 [LVM]

```
硬盘
关机 -->工具集 { lsblk;df -h;df -hT;find /dev/ -name nvme0n*/lv01}
yum -y install lvm2
pvcreate nvme0n*
vgcreate vg01 nvme0n2 □□□□&& vgextend vg01 /dev/nvme0n3 /dev/nvme0n4
lvcreate -L 3G -n lv01 vg01 □    && lvextend -L +3G /dev/vg01/lv01
mkfs.xfs /dev/vg01/lv01□□□□mkfs.ext4 /dev/vg01/lv01
xfs_growfs /dev/vg01/lv01□□□ext4 {resize2fs /dev/vg01/lv01}
mkdir /mnt/lvxfs && mount /dev/vg01/lv01 /mnt/lvxfs □□□(xfs□ext4) = mount -t xfs /dev/vg01/lv01
/mnt/lvxfs
缩 {ext4}
e2fsck -f /dev/vg01/lv01
resize2fs /dev/vg01/lv01 8G
lvreduce -L -4G/8G /dev/vg01/lv01
```

```
pvs {pvdisplay};vgs {vgdisplay};lvs {lvdisplay}
lvremove vgremove pvremove
lvcreate -L 1G -s -n lv02 /dev/vg01/lv01
```

```
强制移除损坏的物理卷□vgreduce --removemissing --force vg_data {pvscan}
fuser {-m;-mk}
```

```
fdisk {-l}
echo "- - -" > /sys/class/scsi_host/host0/scan
echo 1 > /sys/block/sdb/device/delete
```

四十七、 [源码编译安装 nginx]

```
yum -y install gcc pcre-devel zlib-devel openssl-devel make #{C语言编译器; 正则库; 压缩库; 加密库}
```

```
wget https://nginx.org/download/nginx-1.30.2.tar.gz && tar -zxvf nginx-1.30.2.tar.gz && cd
nginx-1.30.2
./congigure --prefix=/opt/nginx --with-http_ssl_module && make && make install
echo 'export PATH=$PATH:/opt/nginx/sbin' >> ~/.bashrc  && source ~/.bashrc
{nginx;nginx -s stop;nginx -s quit;nginx -s reload;-t}
```

```
vim /usr/lib/systemd/system/nginx.service
[Unit]
```

```
Description=Nginx Source Service
After=network.target
[Service]
Type=forking
PIDFile=/opt/nginx/logs/nginx.pid
ExecStart=/opt/nginx/sbin/nginx
ExecReload=/opt/nginx/sbin/nginx -s reload
ExecStop=/opt/nginx/sbin/nginx -s quit
PrivateTmp=true
[Install]
WantedBy=multi-user.target
```

```
cd && rm -rf nginx-1.30.2 nginx-1.30.2.tar.gz /opt/nginx/
/usr/lib/systemd/system/nginx.service && sed -i '$d' ~/.bashrc
```

四十八、 [php]

```
yum -y install php-fpm php-mysqlnd php-mbstring php-gd php-xml \
php-json php-curl php-zip php-fileinfo php-opcache php-bcmath
```

四十九、 [web网页]

```
<meta charset="UTF-8">
<h1>我是最大的标题</h1>
<a href="/aaa.html">点击跳转到 aaa.html</a>
<h2>我是第二大的标题</h2>
<p>我是普通文字，正文内容</p>
<br>
<strong>我是加粗的字</strong>
https://nginx.org/en/download.html
```

```
sudo ./configure --prefix=/usr/local/nginx --with-http_ssl_module --with-http_v2_module --
with-pcre
sudo vim /etc/init.d/nginx
case "$1" in
  start)
    /usr/local/nginx/sbin/nginx
```

```
    ;;
  stop)
    /usr/local/nginx/sbin/nginx -s stop
    ;;
  restart)
    /usr/local/nginx/sbin/nginx -s reload
    ;;
  *)
    echo "Usage: $0 {start|stop|restart}"
    exit 1
    ;;
esac
```

五十、 [源仓库]

```
[BaseOS]
name=手动创建的阿里云源仓库-baseOs
 #baseurl=https://mirrors.aliyun.com/rockylinux/8/BaseOS/x86_64/os/
mirrorlist=https://mirrors.rockylinux.org/mirrorlist?arch=$basearch&repo=BaseOS-$releasever
gpgcheck=1
 #gpgkey=https://mirrors.aliyun.com/rockylinux/8/BaseOS/x86_64/os/RPM-GPG-KEY-rockyofficial
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-rockyofficial
enabled=1
```

```
[AppStream]
name=手动创建的阿里云源仓库-appstream
 #baseurl=https://mirrors.aliyun.com/rockylinux/8/AppStream/x86_64/os/
mirrorlist=https://mirrors.rockylinux.org/mirrorlist?arch=$basearch&repo=AppStream-$releasever
gpgcheck=1
 #gpgkey=https://mirrors.aliyun.com/rockylinux/8/AppStream/x86_64/os/RPM-GPG-KEY-rockyofficial
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-rockyofficial
enabled=1
```

```
[epel]
name=手动创建的阿里云源仓库-epel
 #baseurl=https://mirrors.aliyun.com/epel/8/Everything/x86_64
metalink=https://mirrors.fedoraproject.org/metalink?repo=epel-
8&arch=$basearch&infra=$infra&content=$contentdir
gpgcheck=1
```

```
gpgkey=https://mirrors.aliyun.com/epel/RPM-GPG-KEY-EPEL-8
enabled=1
$basearch {x86_64} | Packages/ {包} | repodata/repomd.xml {索引}
```

```
wget https://mirrors.aliyun.com/epel/epel-release-latest-8.noarch.rpm
rpm -ivh epel-release-latest-8.noarch.rpm
```

```
dnf install -y epel-release
```

```
dnf install -y https://rpms.remirepo.net/enterprise/remi-release-8.rpm
```

```
--enablerepo=epel --disablerepo=* | --disablerepo=epel
yum-config-manager {--add-repo;--disable;--enable}
```

五十一、 [本地仓库部署]

```
mkdir -p /local_repo && yum install --downloadonly --downloaddir=/local_repo nginx php-fpm
mariadb
yum install createrepo -y && createrepo /local_repo
vim /etc/yum.repos.d/local.repo
[local-repo]
name=本地离线软件仓库
baseurl=file:///local_repo
enabled=1
gpgcheck=0
```

五十二、 [服务]

```
/etc/systemd/system/
```

```
[Unit]
Description=  *.service
After=network.target
Before=  *.service.target
```

```
[Service]
Environment="JAVA_HOME=/xxx"
```

```
ExecStart=path/*.sh
ExecStop=path/*.sh
Type=forking
Restart=on-failure
RestartSec=aaa
```

```
[Install]
WantedBy=multi-user.target
```

```
systemd-analyze verify   *.service
systemctl daemon-reload
systemctl start   *.service
systemctl stop   *.service
systemctl enable   *.service
systemctl disable   *.service
systemctl status   *.service
```

五十三、 [网卡]

```
nmtui #图像化设置网卡
uuidgen
/etc/sysconfig/network-scripts/ifcfg-ens33
```

```
TYPE=Ethernet□□# en
BOOTPROTO=static□# dhcp none
NAME=ens33
DEVICE=ens33
ONBOOT=yes
IPADDR=192.168.20.144
NETMASK=255.255.255.0□□# PREFIX=24
GATEWAY=192.168.20.1
DNS1=223.5.5.5
DNS2=114.114.114.114
```

```
PROXY_METHOD=none
BROWSER_ONLY=no
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
```

```
UUID=****
uuidgen
```

```
device=dev=d
connection=con=c
nmcli device status
nmcli connection show
nmcli connection reload
nmcli device reapply ens33
nmcli connection down ens33
nmcli connection up ens33
systemctl restart network
systemctl restart NetworkManager
nmcli con reload
nmcli con up "我的网卡" # 启用新名称配置
```

```
nmcli con mod "ens160" ipv4.addresses "192.168.20.145/24"
nmcli con mod "ens160" ipv4.gateway "192.168.20.10
nmcli con mod "ens160" ipv4.dns "144.144.144.144
nmcli con mod "ens160" ipv4.method manual
```

```
ip a
ip route
cat /etc/resolv.conf
```

五十四、 [远程连接]

```
/etc/ssh/sshd_config
```

```
ssh root@ip
sshd -t
systemctl restart sshd
想要不卡
UseDNS no
GSSAPIAuthentication no
PermitRootLogin yes
PasswordAuthentication yes
```

五十五、脚本通用

```
aaa=${1:-default}
command -v name
```

五十六、 [大模型]

```
https://ollama.com
```

五十六、 [lnmp架构]

```
[n]
yum install -y nginx
systemctl enable --now nginx && systemctl status nginx
```

```
server {
    listen 80 default_server;
    server_name localhost;
```

```
    root /web;
    index index.html index.php index.htm;
```

```
    location / {
        try_files $uri $uri/ =404;
    }
```

```
    location ~ \.php$ {
        fastcgi_pass 127.0.0.1:9000;
        fastcgi_param SCRIPT_FILENAME /web$fastcgi_script_name;
        include fastcgi_params;
    }
}
mkdir -p /web
chown -R nginx:nginx /web
```

```
grep -v "^;" /etc/php-fpm.conf | grep -v "^$"
```

```
dnf update nginx* -y
nginx -t
[m]
dnf install -y https://dev.mysql.com/get/mysql84-community-release-el9-4.noarch.rpm #(9.5)
dnf install -y mysql-community-server
grep password /var/log/mysqld.log
alter user "root"@"localhost" identified by '123456';
```

```
dnf install -y mysql-server
systemctl enable --now mysqld && systemctl status mysqld
mysql_secure_installation
```

```
[php-fpm]
yum install -y php php-fpm php-mysqlnd
vim /etc/php-fpm.d/www.conf
user = nginx
group = nginx
sed -i 's/^user.*/user = nginx/' /etc/php-fpm.d/www.conf
sed -i 's/^group.*/group = nginx/' /etc/php-fpm.d/www.conf
```

```
grep -v "^#" /etc/nginx/nginx.conf | grep -v "^$"
grep -v "^;" /etc/php-fpm.d/www.conf | grep -v "^$"
grep -E '^(user|group)' /etc/php-fpm.d/www.conf
```

```
systemctl enable --now php-fpm && systemctl status php-fpm
```

```
rm -f /etc/php-fpm.d/www.conf
dnf reinstall -y php-fpm
```

```
sed -i.bak 's/^group.*/group = nginx/' /etc/php-fpm.d/www.conf
diff file1 file2
```

```
echo "<?php phpinfo(); ?>" > /web/info.php
systemctl status nginx && systemctl status mysqld && systemctl status php-fpm
```

□□□ [root□□□□□]

```
vi /etc/pam.d/system-auth
root密码强度要求
```

```
bash {-n;-x;-e}
```

五十八、 [克隆注意事项]

```
mac地址
sed -i "s/^UUID=.*/UUID=\"$(uuidgen)\"/" /etc/sysconfig/network-scripts/ifcfg-ens160
hostnamectl set-hostname 154
sed -i "s/IPADDR.*/IPADDR=192.168.20.154/" /etc/sysconfig/network-scripts/ifcfg-ens160
nmcli con reload
```

五十九、 [内核调优]

```
"该参数控制系统在内存不足时,内核将页面交换到磁盘的程度(交换分区)。默认值为60,建议值为10-30
vm.swappiness = 30
该参数控制系统是否允许超额分配内存。默认值为0,迄建议值为1
vm.overcommit_memory = 1
该参数控制系统内核缓存的大小和清理频率。默认值为100,建议值为50-100
vm.vfs_cache_pressure = 70
该参数控制系统保留的最小空闲内存。默认值为4096,建议值为65536,一般建议为系统总内存的1%左右,但最小不低于65536(64MB),最大不超过655360(640MB
vm.min_free_kbytes = 65535
该参数控制系统TCP连接的最大排队数量,默认值为128,建议直为1024
net.core.somaxconn = 1024
该参数控制系统TCP连接的保持时间,默认值为7200秒,建议值为600-1200
net.ipv4.tcp_keepalive_time = 1200
增加系统使用的端口范围,建议值为1024-65535
net.ipv4.ip_local_port_range = 1024 65535
该参数控制系统可以打开的文件句柄数量。默认值为65536,建议值为1048576
fs.file-max = 1048576
该参数控制系统可以创建的最大进程数,默认最大是65535
kernel.pid_max = 65535
调整用户进程最大数量,默认最大是65535
kernel.threads-max = 65535
sysctl --system
sysctl -a
```

```
vim /etc/security/limits.conf
* soft nofile 65535
* hard nofile 65535
ulimit -n
ulimit -Hn
```

六十、 [升级内核]

```
cp -r /boot /boot.bak
cp -r /etc /etc.bak
yum update -y
uname -r
提前准被高版本rpm包
rpm -Uvh
grub2-set-default 0
reboot
```

```
ps -eo pid,user,%mem,rss,args | awk '$4>=10000'
ps -ef | awk '/nginx/ && !/awk/ {count++} END {print count}'
dnf install -y sysstat
iostat -x -d 1 1 | awk 'NR>3 && $1~/(sd|hd|nvme)/ && $10>0 {print $1,$10}' | column -t
```

六十一、 [三剑客]

```
sed -e 's/^/ <h1>/' -e 's#$# </h1>#' file
sed 's#\(.*\)#<h1>\1</h1>#' file
while read line; do echo "<h1>$line</h1>";done < file
```

```
awk '{aaa+=$2} END {print aaa}' file
awk '{print aaa+=$1}' file
awk '{print aaa+=$2} END {print aaa}' file
awk '{ aaa+=$3} END {print aaa}' file
awk '{for(i=1;i<=NF;i++) aaa[i]+=$i} END {print aaa[3]}' file
awk '{for(i=1;i<=NF;i++) aaa[i]+=$i} END{for(j=1;j<=NF;j++) print j, aaa[j]}' file
awk 'BEGIN {print "开始"} {print $1} END {print "结束"}' file
awk 'NR%2==1' file
awk 'OFS="-" {print $1,$2,$3}' file
awk '{if($2>20) print $1,$2}' file
```

```
systemctl daemon-reexec
systemctl daemon-reload
```