# 初始化v3

#!/bin/bash  
#author kingofzhou  
#this script is only for CentOS 7.x  
#check the OS  
#url https://www.cnblogs.com/likecoke/p/13555145.html https://www.alongparty.cn/centos7-initialization.html https://www.modb.pro/db/43594 https://blog.whsir.com/post-2024.html  
  
platform=`uname -i`  
if [ $platform != "x86\_64" ];then  
echo "this script is only for 64bit Operating System !"  
exit 1  
fi  
echo "the platform is ok"  
  
echo -ne "\033[32m -->【是否要配置阿里yum源!y|n】: \033[0m"  
read aliyum  
  
  
cat << EOF  
+---------------------------------------+  
| your system is CentOS 7 x86\_64 |  
| start optimizing....... |  
+---------------------------------------  
EOF  
  
if [ $UID -ne 0 ]; then  
echo "Superuser privileges are required to run this script."  
exit 1  
fi  
  
sh\_path=$(cd `dirname $0`;pwd)  
  
blue() {  
echo -e "\033[34m $1 \033[0m" && sleep 2  
}  
  
red() {  
echo -e "\033[31m $1 \033[0m" && sleep 1  
}  
  
#############################【添加公网 DNS 地址】#############################################  
blue "添加公网 DNS 地址"  
cat >> /etc/resolv.conf << EOF  
nameserver 119.29.29.29  
EOF  
  
#############################【关闭ipv6】#############################################  
blue "关闭ipv6"  
echo "alias net-pf-10 off" >> /etc/modprobe.conf  
echo "alias ipv6 off" >> /etc/modprobe.conf  
chkconfig ip6tables off  
  
  
#############################【关闭selinux】#############################################  
blue "关闭selinux"  
sed -i '/SELINUX/s/enforcing/disabled/' /etc/selinux/config  
setenforce 0  
  
##############################【关闭iptables】############################################  
blue "关闭iptables"  
chkconfig iptables off  
service iptables stop  
systemctl stop firewalld  
systemctl disable firewalld  
  
############################【配置远程管理ssh服务】##############################################  
blue "配置远程管理ssh服务"  
sed -i 's/^GSSAPIAuthentication yes$/GSSAPIAuthentication no/' /etc/ssh/sshd\_config  
sed -i 's/#UseDNS yes/UseDNS no/' /etc/ssh/sshd\_config  
sed -i 's/#Port 22/Port 22/' /etc/ssh/sshd\_config  
systemctl restart sshd  
service sshd restart  
  
############################【安装必备软件包】##############################################  
  
if [ $aliyum == "y" ];then  
blue "开始配置阿里yum源！"  
cd /etc/yum.repos.d/  
mkdir -v backup  
mv -v \*.repo backup/  
#Yum 源更换为国内阿里源  
curl -s -o ./Centos-7.repo https://mirrors.aliyun.com/repo/Centos-7.repo  
sed -i -e '/mirrors.cloud.aliyuncs.com/d' -e '/mirrors.aliyuncs.com/d' Centos-7.repo  
#添加阿里的 epel 源  
curl -s -o ./epel.repo https://mirrors.aliyun.com/repo/epel-7.repo  
  
#yum 重新建立缓存  
yum clean all  
yum makecache  
fi  
  
############################【同步系统时间】##############################################  
blue "chrony时间同步设置"  
yum -y install chrony  
cat > /etc/chrony.conf << EOF  
# server 用于添加 ntp 服务器  
server s1a.time.edu.cn iburst minpoll 4 maxpoll 10  
server ntp.aliyun.com iburst minpoll 4 maxpoll 10  
# 记录系统时钟增益/损失时间的速率  
driftfile /var/lib/chrony/drift  
# 允许跳跃式校时 如果在前 3 次校时中时间差大于 1.0s  
makestep 1.0 3  
# 同步时间至 RTC  
rtcsync  
# 为日志文件指定目录  
logdir /var/log/chrony  
EOF  
  
systemctl enable chronyd  
systemctl restart chronyd  
  
chronyc -a makestep  
timedatectl set-local-rtc 0  
timedatectl set-ntp true  
timedatectl status  
  
##################################################################################  
blue "安装必备软件包"  
  
yum install -y epel-release unzip zip gcc gcc-c++ make autoconf libtool-ltdl-devel gd freetype-devel libxml2-devel libjpeg-devel libpng-devel \  
openssl-devel curl-devel bison patch unzip ncurses-devel sudo bzip2 mlocate flex lrzsz lsof setuptool system-config-firewall-tui ntp libaio-devel \  
wget nmap finger elinks tcl sysstat vim wget net-tools tree telnet libcurl openssl cpp binutils glibc-kernheaders glibc-common glibc-devel automake \  
libtool zlib-devel dos2unix bash-completion bash-completion-extras nc ntpdate open-sshclients screen nload yum-utils device-mapper-persistent-data \  
lvm2 cronolog  
  
yum-config-manager --add-repo http://mirrors.aliyun.com/docker-ce/linux/centos/docker-ce.repo  
yum list docker-ce --show-duplicates  
  
chkconfig bluetooth off  
systemctl disable postfix.service  
############################【关闭防火墙】##############################################  
blue "关闭防火墙"  
systemctl disable firewalld.service  
systemctl stop firewalld.service  
  
############################【关闭NetworkManager】##############################################  
blue "关闭NetworkManage"  
  
systemctl stop NetworkManager  
systemctl disable NetworkManager  
  
  
############################【设置软件硬件资源与内核同时打开文件最大值】##############################################  
blue "设置软件硬件资源与内核同时打开文件最大值"  
#并发连接数  
sed -i 's#4096#102400#g' /etc/security/limits.d/20-nproc.conf   
  
echo "ulimit -SHn 102400" >> /etc/rc.local  
cat >> /etc/security/limits.conf << EOF  
\* soft nofile 655350  
\* hard nofile 655350  
\* soft nproc 655350  
\* hard nproc 655350  
EOF  
chmod +x /etc/rc.d/rc.local  
  
############################【设置 ssh】##############################################  
blue "设置 ssh"   
sed -i 's/^GSSAPIAuthentication yes$/GSSAPIAuthentication no/' /etc/ssh/sshd\_config  
sed -i 's/#UseDNS yes/UseDNS no/' /etc/ssh/sshd\_config  
systemctl restart sshd.service  
  
############################【关闭control-alt-delete3键重启】##############################################  
blue "关闭control-alt-delete3键重启"  
sed -i 's#exec /sbin/shutdown -r now#\#exec /sbin/shutdown -r now#' /etc/init/control-alt-delete.conf  
  
############################【内核参数优化】##############################################  
blue "内核参数优化"  
true > /etc/sysctl.conf  
cat >> /etc/sysctl.conf << EOF  
#CTCDN 系统优化参数  
#关闭 ipv6  
net.ipv6.conf.all.disable\_ipv6 = 1  
net.ipv6.conf.default.disable\_ipv6 = 1  
#决定检查过期多久邻居条目  
net.ipv4.neigh.default.gc\_stale\_time=120  
#使用 arp\_announce / arp\_ignore 解决 ARP 映射问题  
net.ipv4.conf.default.arp\_announce = 2  
net.ipv4.conf.all.arp\_announce=2  
net.ipv4.conf.lo.arp\_announce=2  
# 避免放大攻击  
net.ipv4.icmp\_echo\_ignore\_broadcasts = 1  
# 开启恶意 icmp 错误消息保护  
net.ipv4.icmp\_ignore\_bogus\_error\_responses = 1  
#关闭路由转发  
net.ipv4.ip\_forward = 0  
net.ipv4.conf.all.send\_redirects = 0  
net.ipv4.conf.default.send\_redirects = 0  
#开启反向路径过滤  
net.ipv4.conf.all.rp\_filter = 1  
net.ipv4.conf.default.rp\_filter = 1  
#处理无源路由的包  
net.ipv4.conf.all.accept\_source\_route = 0  
net.ipv4.conf.default.accept\_source\_route = 0  
#关闭 sysrq 功能  
kernel.sysrq = 0  
#core 文件名中添加 pid 作为扩展名  
kernel.core\_uses\_pid = 1  
# 开启 SYN 洪水攻击保护  
net.ipv4.tcp\_syncookies = 1  
#修改消息队列长度  
kernel.msgmnb = 65536  
kernel.msgmax = 65536  
#设置最大内存共享段大小 bytes  
kernel.shmmax = 68719476736  
kernel.shmall = 4294967296  
#timewait 的数量，默认 180000  
net.ipv4.tcp\_max\_tw\_buckets = 6000  
net.ipv4.tcp\_sack = 1  
net.ipv4.tcp\_window\_scaling = 1  
net.ipv4.tcp\_rmem = 4096 87380 4194304  
net.ipv4.tcp\_wmem = 4096 16384 4194304  
net.core.wmem\_default = 8388608  
net.core.rmem\_default = 8388608  
net.core.rmem\_max = 16777216  
net.core.wmem\_max = 16777216  
#每个网络接口接收数据包的速率比内核处理这些包的速率快时，允许送到队列的数据包的最大数目  
net.core.netdev\_max\_backlog = 262144  
#限制仅仅是为了防止简单的 DoS 攻击  
net.ipv4.tcp\_max\_orphans = 3276800  
#未收到客户端确认信息的连接请求的最大值  
net.ipv4.tcp\_max\_syn\_backlog = 262144  
net.ipv4.tcp\_timestamps = 0  
#内核放弃建立连接之前发送 SYNACK 包的数量  
net.ipv4.tcp\_synack\_retries = 1  
#内核放弃建立连接之前发送 SYN 包的数量  
net.ipv4.tcp\_syn\_retries = 1  
#启用 timewait 快速回收  
net.ipv4.tcp\_tw\_recycle = 1  
#开启重用。允许将 TIME-WAIT sockets 重新用于新的 TCP 连接  
net.ipv4.tcp\_tw\_reuse = 1  
net.ipv4.tcp\_mem = 94500000 915000000 927000000  
net.ipv4.tcp\_fin\_timeout = 1  
#当 keepalive 起用的时候，TCP 发送 keepalive 消息的频度。缺省是 2 小时  
net.ipv4.tcp\_keepalive\_time = 1800  
net.ipv4.tcp\_keepalive\_probes = 3  
net.ipv4.tcp\_keepalive\_intvl = 15  
#允许系统打开的端口范围  
net.ipv4.ip\_local\_port\_range = 1024 65000  
#修改防火墙表大小，默认 65536  
net.netfilter.nf\_conntrack\_max=655350  
net.netfilter.nf\_conntrack\_tcp\_timeout\_established=1200  
# 确保无人能修改路由表  
net.ipv4.conf.all.accept\_redirects = 0  
net.ipv4.conf.default.accept\_redirects = 0  
net.ipv4.conf.all.secure\_redirects = 0  
net.ipv4.conf.default.secure\_redirects = 0  
EOF  
/sbin/sysctl -p  
  
############################【vim 定义退格键可删除最后一个字符类型】##############################################  
blue "vim 定义退格键可删除最后一个字符类型"  
echo 'alias vi=vim' >> /etc/profile  
echo 'stty erase ^H' >> /etc/profile  
echo "alias grep='grep --color=auto'" >> /etc/profile  
echo "alias tf='tail -f'" >> /etc/profile  
echo "alias ll='ls -lh'" >> /etc/profile  
echo "alias md5='md5sum'" >> /etc/profile  
  
cat >> /root/.vimrc << EOF  
set tabstop=4  
set shiftwidth=4  
set expandtab  
syntax on  
"set number  
EOF  
  
############################【设置系统默认启动级别】##############################################  
blue "设置系统默认启动级别"  
sed -i '/^id/s/5/3/' /etc/inittab  
  
  
############################【配置PS环境】##############################################  
blue "配置PS环境"  
sed -i '/^PS1=.\*/d' /etc/profile  
echo "PS1='[\[\e[1;35m\]\u\[\e[1;33m\]@\[\e[1;32m\]\h \[\e[1;0m\]\e[4m`pwd`\e[m\e[1;37m \[\e[1;0m\]\[\e[1;34m\]\t\[\e[1;0m\]]\n\[\e[1;31m\]\$\[\e[0m\] '" >> /etc/profile  
  
  
############################【设置命令历史记录参数】##############################################   
blue "设置命令历史记录参数"  
sed -i 's/HISTSIZE=1000/HISTSIZE=20/' /etc/profile  
  
grep 'HISTFILESIZE' /etc/bashrc &>/dev/null  
if [ $? -ne 0 ]  
then  
cat << EOF >> /etc/bashrc  
HISTFILESIZE=4000  
HISTSIZE=4000  
HISTTIMEFORMAT='%F/%T'  
EOF  
fi  
source /etc/bashrc  
  
  
############################【升级openssl】##############################################   
blue "升级openssl"  
echo -e "\033[46m ================start install openssl-1.1.1================ ... \033[0m"  
mkdir /usr/local/src/apr && cd /usr/local/src/apr  
#wget https://www.openssl.org/source/old/1.1.1/openssl-1.1.1.tar.gz  
wget --no-check-certificate https://www.openssl.org/source/openssl-1.1.1m.tar.gz  
tar xzvf openssl-1.1.1m.tar.gz && cd openssl-1.1.1m  
./config --prefix=/usr/local/openssl && make -j $(grep processor /proc/cpuinfo | wc -l) && make install  
mv /usr/bin/openssl{,.ori}  
ln -sv /usr/local/openssl/bin/openssl /usr/bin/openssl  
cd /usr/local/src/apr/openssl-1.1.1m && /bin/cp -av libssl.so.1.1 /usr/lib64/libssl.so.1.1 && /bin/cp -av libcrypto.so.1.1 /usr/lib64/libcrypto.so.1.1  
  
############################【安装jemalloc】##############################################  
blue "安装jemalloc"  
if test -d /usr/local/include/jemalloc;then  
blue "jemalloc已部署"  
red "设置jemalloc软链接"  
ln -sfv /usr/local/lib/libjemalloc.so.2 /usr/lib64/libjemalloc.so.1  
else  
blue "安装 autogen autoconf"  
yum -y install autogen autoconf  
  
install () {  
tar -xf jemalloc-5.2.1.tar.bz2 && cd jemalloc-5.2.1  
./autogen.sh  
make -j4  
make install  
ln -s /usr/local/lib/libjemalloc.so.2 /usr/lib64/libjemalloc.so.1  
echo '/usr/local/lib' > /etc/ld.so.conf.d/local.conf  
ldconfig  
cd ../  
}  
  
blue "下载and解压and部署"  
if test -f jemalloc-5.2.1.tar.bz2;then   
install   
else   
while true; do  
#wget --no-check-certificate https://github.com/jemalloc/jemalloc/releases/download/5.2.1/jemalloc-5.2.1.tar.bz2  
wget https://github.com/jemalloc/jemalloc/releases/download/5.2.1/jemalloc-5.2.1.tar.bz2  
if [ "$?" != 0 ];then   
red "未下载成功，将继续尝试下载，请耐心等待！"   
else   
blue "下载成功,开始部署！"  
install  
break  
fi  
done  
fi  
fi  
  
  
  
  
  
############################【删除不必要的用户】##############################################   
blue "删除不必要的用户"  
for USERS in adm lp sync shutdown halt mail news uucp operator games gopher  
do  
grep $USERS /etc/passwd &>/dev/null  
if [ $? -eq 0 ]  
then  
userdel $USERS &> /dev/null  
fi  
done  
  
  
##############################【删除不必要的用户组】#############################################  
blue "删除不必要的用户组"  
for GRP in adm lp mail news uucp games gopher mailnull floppy dip pppusers popusers slipusers daemon  
do  
grep $GRP /etc/group &> /dev/null  
if [ $? -eq 0 ]  
then  
groupdel $GRP &> /dev/null  
fi  
done  
  
############################【修改10分钟无活动自动退出】##############################################  
blue "修改10分钟无活动，自动退出"  
echo "TMOUT=600" >>/etc/profile  
  
############################【创建行为审计日志记录】##############################################  
blue "创建行为审计日志记录"  
FILENAME="/var/log/usermonitor.log"  
PATHNAME="/etc/profile"  
FINDNAME="HISTORY\_FILE"  
if [[ ! -f ${FILENAME} ]]  
then  
#创建行为审计日志文件  
touch ${FILENAME}   
#将日志文件的所有者改为权限低的用户NOBODY  
chown nobody:nobody ${FILENAME}   
#赋予所有用户对日志文件写的权限  
chmod 002 ${FILENAME}  
#使所有用户对日志文件只有追加权限  
chattr +a ${FILENAME}  
fi   
  
if [[ `cat ${PATHNAME} | grep ${FINDNAME} | wc -l` < 1 ]]; then  
cat >> ${PATHNAME} <<"EOF"  
export HISTORY\_FILE=/var/log/usermonitor.log  
export PROMPT\_COMMAND='{ date "+%y-%m-%d %T ##### $(who am i |awk "{print \$1\" \"\$2\" \"\$5}") #### $(id|awk "{print \$1}") #### $(history 1 | { read x cmd; echo "$cmd"; })"; } >> ${HISTORY\_FILE}'  
EOF  
else  
exit 0   
fi  
  
source $PATHNAME  
  
############################【开机启动优化】##############################################  
blue "开机启动优化"  
for i in acpid gpm lvm2-monitor anacron haldaemon mcstrans oddjobd setroubleshoot atd halt mdmonitor pand single auditd hidd mdmpd pcscd smartd hplip messagebus avahi-daemon ip6tables microcode\_ctl snmptrapd avahi-dnsconfd ipmi multipathd psacct bluetooth rawdevices svnserve conman irda netconsole rdisc cpuspeed irqbalance netfs readahead\_early tcsd iscsi netplugd readahead\_later winbind cups iscsid restorecond wpa\_supplicant cups-config-daemon kdump NetworkManager rpcgssd dnsmasq killall nfs rpcidmapd ypbind dund krb524 nfslock rpcsvcgssd yum-updatesd firstboot lm\_sensors nscd saslauthd  
do  
chkconfig $i off > /dev/null 2>&1  
done  
cat << EOF  
+-------------------------------------------------+  
| 优 化 已 完 成 |  
| 5s 后 重启 这台服务器 ! |  
+-------------------------------------------------+  
EOF  
sleep 5  
reboot ##重启加载内核修改