nginx

ngsky

2018.9.9

V1.0

一、nginx安装



1.安装

tar -zxvf tengine-2.2.2.tar.gz

cd tengine-2.2.2

yum install -y gcc

./configure

make && make install

2.启动

.  
./nginx

3.配置自启动

创建脚本 nginx

|  |
| --- |
| #!/bin/sh  #  # nginx - this script starts and stops the nginx daemon  #  # chkconfig: - 85 15  # description: NGINX is an HTTP(S) server, HTTP(S) reverse \  # proxy and IMAP/POP3 proxy server  # processname: nginx  # config: /etc/nginx/nginx.conf  # config: /etc/sysconfig/nginx  # pidfile: /var/run/nginx.pid    # Source function library.  . /etc/rc.d/init.d/functions    # Source networking configuration.  . /etc/sysconfig/network    # Check that networking is up.  [ "$NETWORKING" = "no" ] && exit 0    nginx="/usr/sbin/nginx"  prog=$(basename $nginx)    NGINX\_CONF\_FILE="/etc/nginx/nginx.conf"    [ -f /etc/sysconfig/nginx ] && . /etc/sysconfig/nginx    lockfile=/var/lock/subsys/nginx    make\_dirs() {  # make required directories  user=`$nginx -V 2>&1 | grep "configure arguments:" | sed 's/[^\*]\*--user=\([^ ]\*\).\*/\1/g' -`  if [ -z "`grep $user /etc/passwd`" ]; then  useradd -M -s /bin/nologin $user  fi  options=`$nginx -V 2>&1 | grep 'configure arguments:'`  for opt in $options; do  if [ `echo $opt | grep '.\*-temp-path'` ]; then  value=`echo $opt | cut -d "=" -f 2`  if [ ! -d "$value" ]; then  # echo "creating" $value  mkdir -p $value && chown -R $user $value  fi  fi  done  }    start() {  [ -x $nginx ] || exit 5  [ -f $NGINX\_CONF\_FILE ] || exit 6  make\_dirs  echo -n $"Starting $prog: "  daemon $nginx -c $NGINX\_CONF\_FILE  retval=$?  echo  [ $retval -eq 0 ] && touch $lockfile  return $retval  }    stop() {  echo -n $"Stopping $prog: "  killproc $prog -QUIT  retval=$?  echo  [ $retval -eq 0 ] && rm -f $lockfile  return $retval  }    restart() {  configtest || return $?  stop  sleep 1  start  }    reload() {  configtest || return $?  echo -n $"Reloading $prog: "  killproc $nginx -HUP  RETVAL=$?  echo  }    force\_reload() {  restart  }    configtest() {  $nginx -t -c $NGINX\_CONF\_FILE  }    rh\_status() {  status $prog  }    rh\_status\_q() {  rh\_status >/dev/null 2>&1  }    case "$1" in  start)  rh\_status\_q && exit 0  $1  ;;  stop)  rh\_status\_q || exit 0  $1  ;;  restart|configtest)  $1  ;;  reload)  rh\_status\_q || exit 7  $1  ;;  force-reload)  force\_reload  ;;  status)  rh\_status  ;;  condrestart|try-restart)  rh\_status\_q || exit 0  ;;  \*)  echo $"Usage: $0 {start|stop|status|restart|condrestart|try-restart|reload|force-reload|configtest}"  exit 2  esac |

将脚本放入 /etc/init.d/目录下，其中修改如下配置,改成自己安装的nginx位置

|  |
| --- |
| nginx="/usr/sbin/nginx"  NGINX\_CONF\_FILE="/etc/nginx/nginx.conf" |

新增执行权限

chomd +x nginx

配置开机启动

chkconfig nginx on

启动方式

service nginx start

service nginx status / stop

防火墙开启80端口

firewall-cmd --zone=public –add-port=80/tcp –permanent

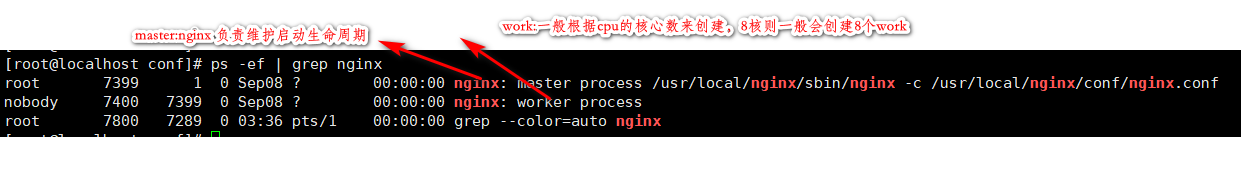
firewall-cmd –reload

firewall-cmd –query-port=80/tcp

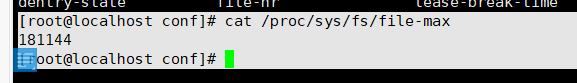
三、nginx配置

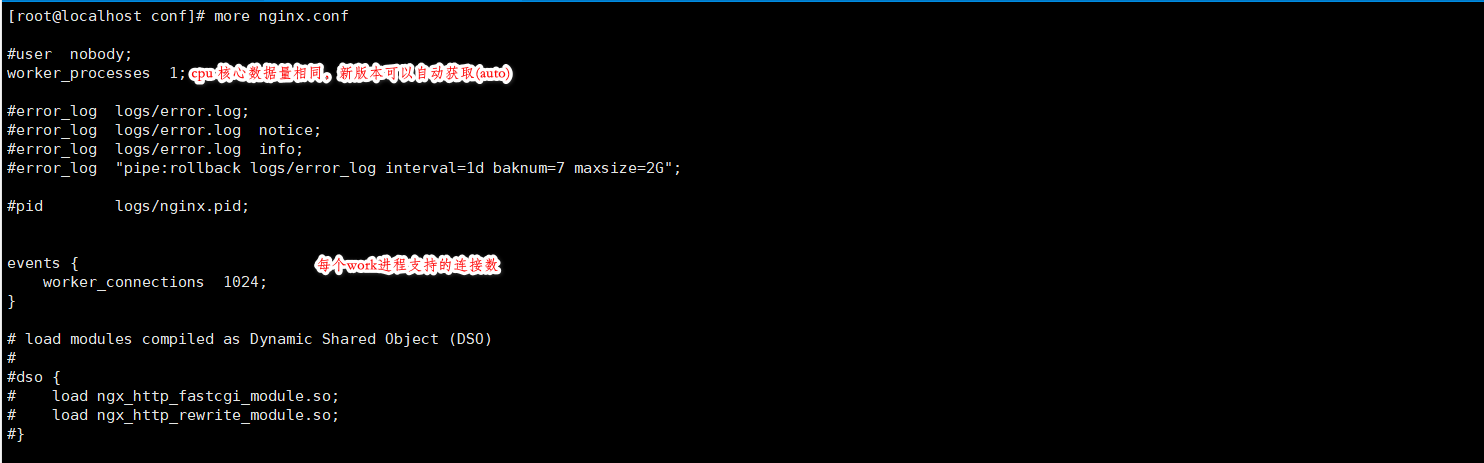
1.运行模式

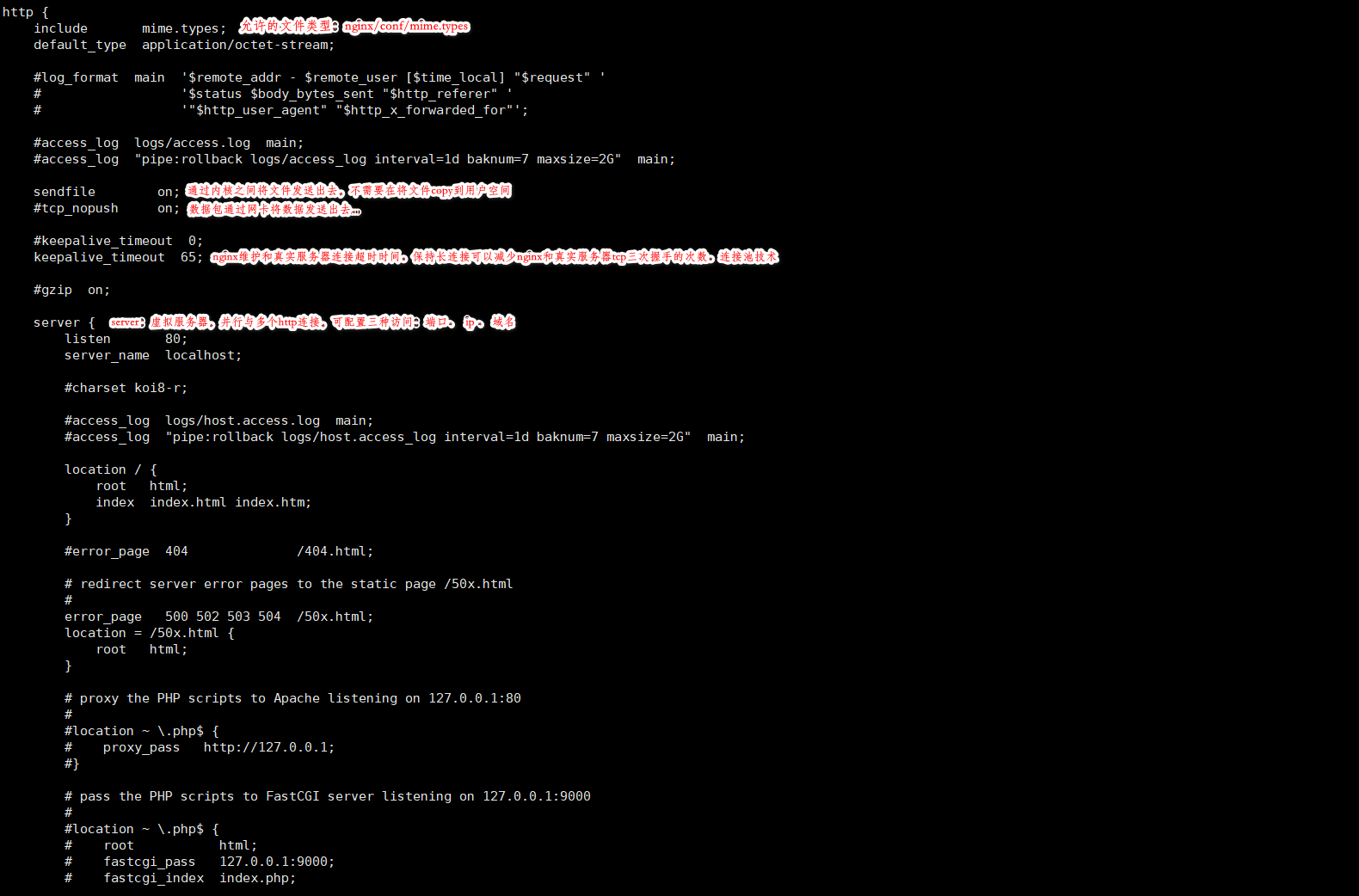
nginx 一个主进程，多个子进程



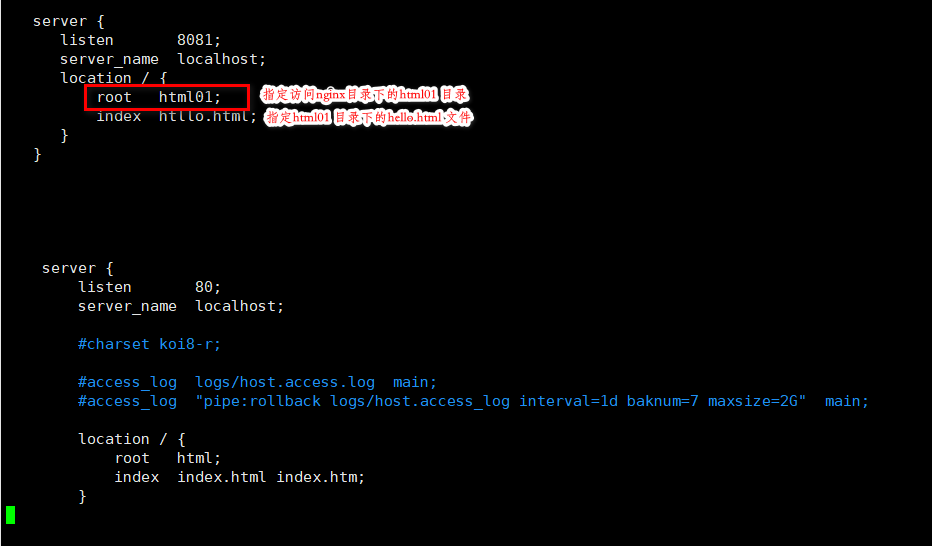
查看内存支持的最大文件描述符数量



2.配置相关

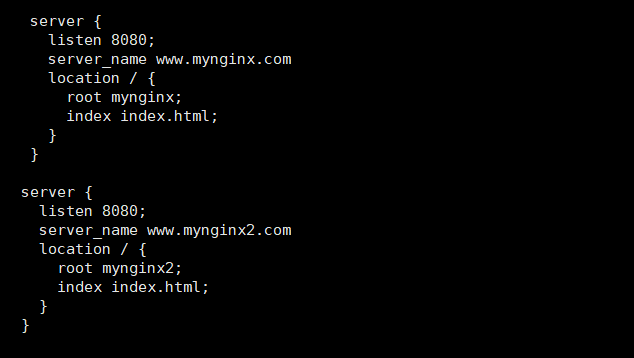


3.虚拟server 基于端口



4.虚拟server基于IP

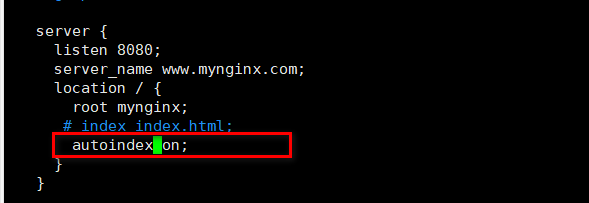
6.虚拟sever基于域名



需要window配置 hosts



location 开启自动索引



会把mynginx 目录下的文件以列表的形式展现出来。

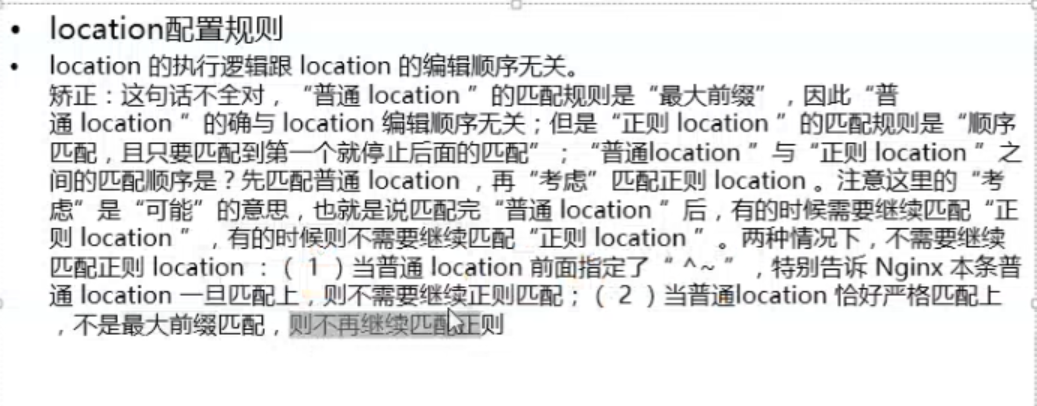


四、配置解析

Index\_2

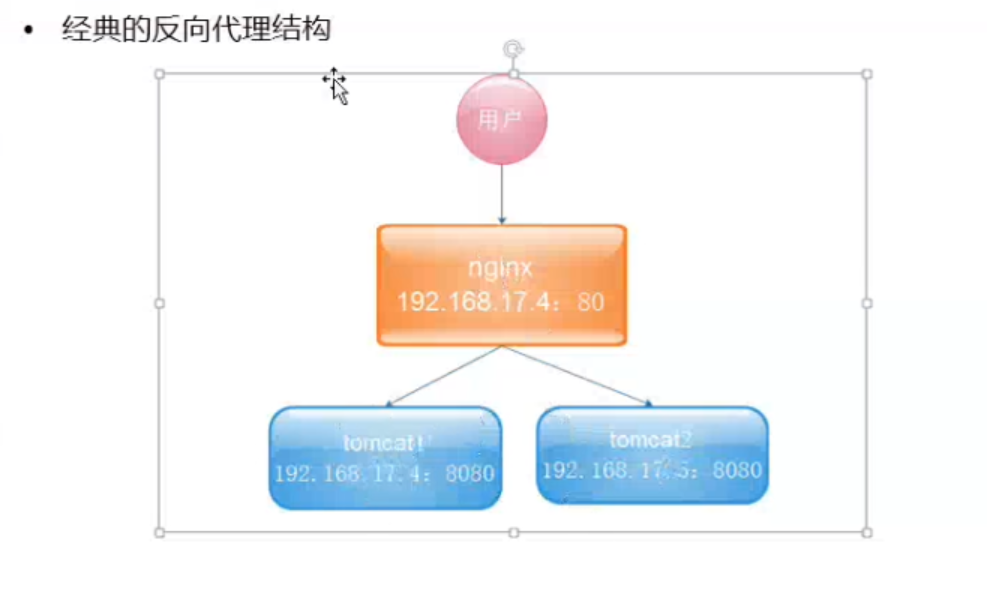


Index\_14



location 下 root 后跟目录

反向代理: proxy\_pass



反向代理多台服务器

