**nginx目录结构**

这里去除了我们不研究的目录

./

├── auto # 包含编译相关的脚本

├── objs # 执行 configure 后产生的目录

└── src # 源码目录

**Src目录**

去除了我们不研究的目录

src

├── core # 基本数据结构和核心代码

├── http # http 模块代码

└── os # 操作系统相关的代码

**在nginx中使用c++**

nginx默认不包含c++的编译选项，要使用c++需修改auto/make文件

在25行附件，新增c++编译选项

CC =  $CC

CFLAGS = $CFLAGS

CPP = $CPP

LINK =  $LINK

# c++编译选项

CXX = g++

CXXFLAGS = -std=c++0x -Wall

LINK = g++

在340行附件修改

if test -n "$NGX\_ADDON\_SRCS"; then

ngx\_cc="\$(CC) $ngx\_compile\_opt \$(CFLAGS) $ngx\_use\_pch \$(ALL\_INCS)"

    # 新增 c++ 编译命令

ngx\_cxx="\$(CXX) $ngx\_compile\_opt \$(CXXFLAGS) $ngx\_use\_pch \$(ALL\_INCS)"

    for ngx\_src in $NGX\_ADDON\_SRCS

    do

        ...

        ngx\_src=`echo $ngx\_src | sed -e "s/\//$ngx\_regex\_dirsep/g"`

        # 根据后缀判断使用 gcc 还是 g++

        ext=`echo ${ngx\_src} | cut -d . -f 2`

        ngx\_gcc=$ngx\_cc

        if [ $ext = "cpp" ]; then

            ngx\_gcc=$ngx\_cxx

        Fi

        cat << END                                            >> $NGX\_MAKEFILE

# 将 $ngx\_cc$ngx\_tab$ngx\_objout$ngx\_obj$ngx\_tab$ngx\_src$NGX\_AUX

# 改为 $ngx\_gcc$ngx\_tab$ngx\_objout$ngx\_obj$ngx\_tab$ngx\_src$NGX\_AUX

# 注意 $ngx\_gcc 前面是 tab 符，不是空格

$ngx\_obj: \$(ADDON\_DEPS)$ngx\_cont$ngx\_src

$ngx\_gcc$ngx\_tab$ngx\_objout$ngx\_obj$ngx\_tab$ngx\_src$NGX\_AUX

**头文件**

本教程使用的nginx头文件为 ngx\_http.h ，在 c++ 中引入为

extern "C" {

#include <ngx\_http.h>

}