CTF竞赛权威指南

原创

vurtual 于 2021-02-27 17:08:29 发布 860 收藏

分类专栏： CTF竞赛权威指南(PWN篇)学习笔记 文章标签： 安全

本文链接：https://blog.csdn.net/vurtual/article/details/114178276

第二章

二进制文件

编译器的结构分为前端(Front end)和后端(Back end)两部分。

编译过程可大致分为5个步骤：

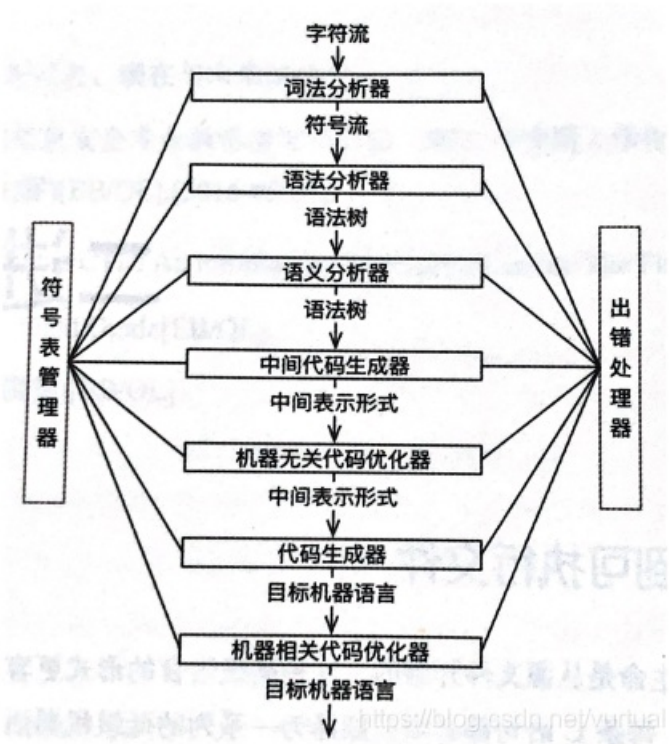
（1）词法分析(Lexical analysis)：读入源程序的字符流，输出为有意义的词素(Lexical)；

（2）语法分析(Syntax analysis)：根据各个词法单元的第一个分量来创建树型的中间表示形式，通常是语法树(Syntax tree)；

（3）语义分析(Semantic analysis)：使用讴歌语法树和符号表中的信息，检测源程序是否满足语言定义的语义约束，同事收集类型信息，用于代码生成、类型检查和类型转换；

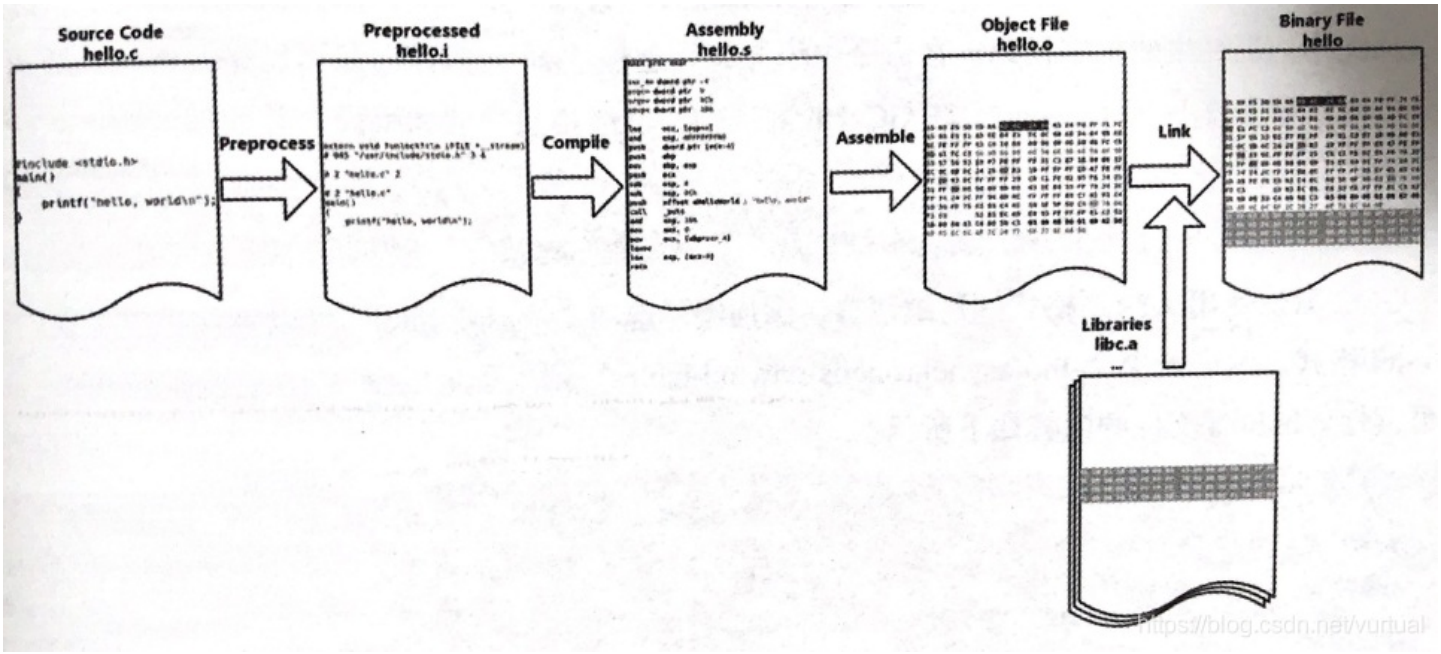
（4）中间代码生成和优化：根据语义分析输出，生成类机器语言的中间表示，如三地址码。然后对生成的中间代码进行分析和优化；

（5）代码生成和优化：把中间表示形式映射到目标机器语言。



GCC编译过程

在编译时添加“-save-temps”和“–verbose”编译选项，前者把编译过程的中间产物保存下来，后者用于查看GCC编译的详细流程。



如图所示，一共使用了ccl、as和collect2三个工具。

ccl是编译器，对应第一和第二阶段，将源文件hello.c编译为hello.s；as是汇编器，对应第三阶段，将hello.s汇编为hello.o目标文件；连接器collect2是对ld命令的封装，用于将C语言运行时中的目标文件(crtl.o、crti.o、crtbegin.o、crtend.o、crtn.o)以及所需的动态链接库(libgcc.so、libgcc_s.so、libc.so)链接到可执行文件hello。