GDB调试工具

## 一、GDB简介

UNIX及UNIX-like下的调试工具。GUN开源组织发布的一个强大的程序调试工具。

### 1.1功能

1、启动你的程序，可以按照你的自定义的要求随心所欲的运行程序。

2、可让被调试的程序在你所指定的调置的断点处停住。（断点可以是条件表达式）

3、当程序被停住时，可以检查此时你的程序中所发生的事。

4、你可以改变你的程序，将一个BUG产生的影响修正从而测试其他BUG。

### 1.2 调试

gcc -g simple\_accumulation.c -o simple\_accumulation

首先在编译时，我们必须要把调试信息加到可执行文件中。使用编译器（cc/gcc/g++）的 -g 参数可以做到这一点。

如果没有-g，你将看不见程序的函数名、变量名，所代替的全是运行时的内存地址。当你用-g把调试信息加入之后，并成功编译目标代码以后，让我们来看看如何用gdb来调试他。

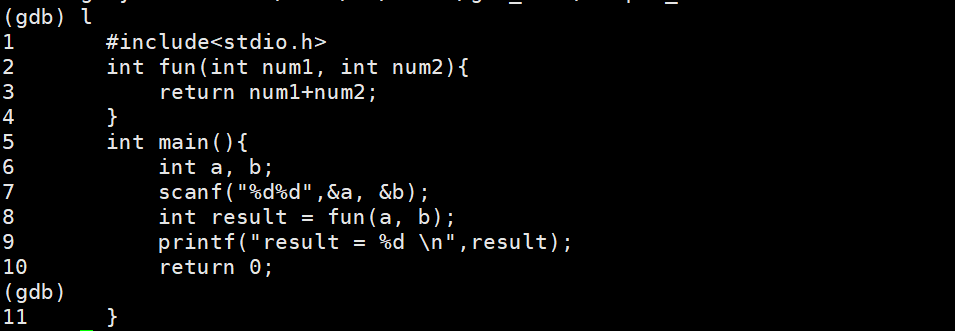
编译后生成的文件名

* **启动**

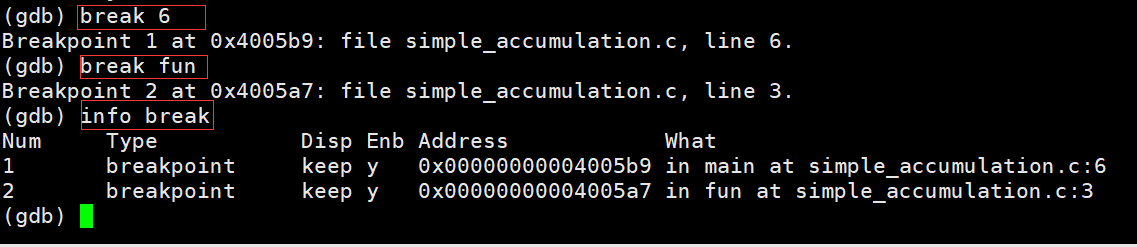
gdb) simple\_accumulation 启动gdb.

* **显示源代码**

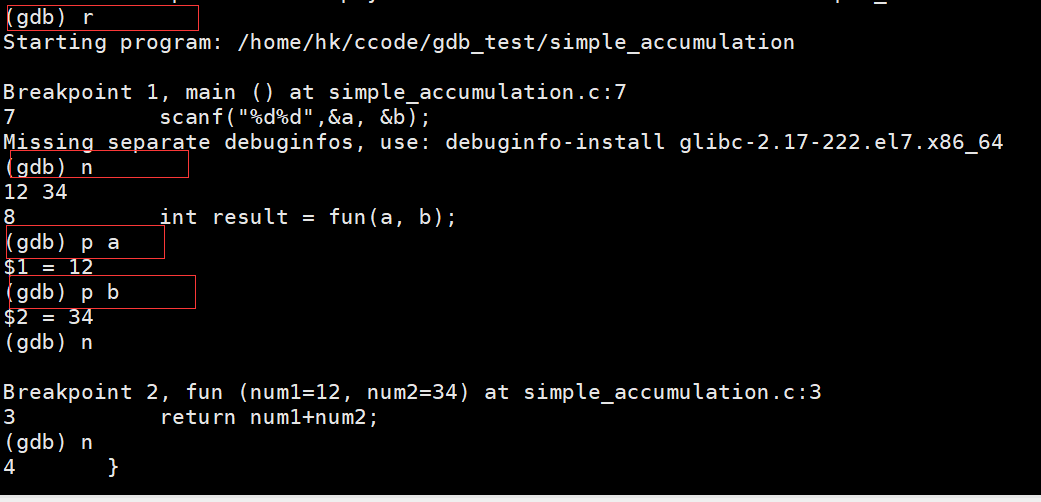
gdb –l



* **设置断点**



* **运行程序**



**R 是run的意思，开始运行程序。**

**N是next的意思，继续执行下一条语句。**

**P 是输出的意思**

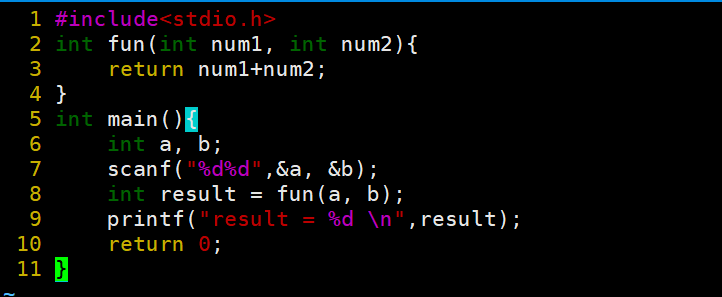
* **退出函数**

**Gdb）finish**

* **退出gdb**

**Gdb) q**

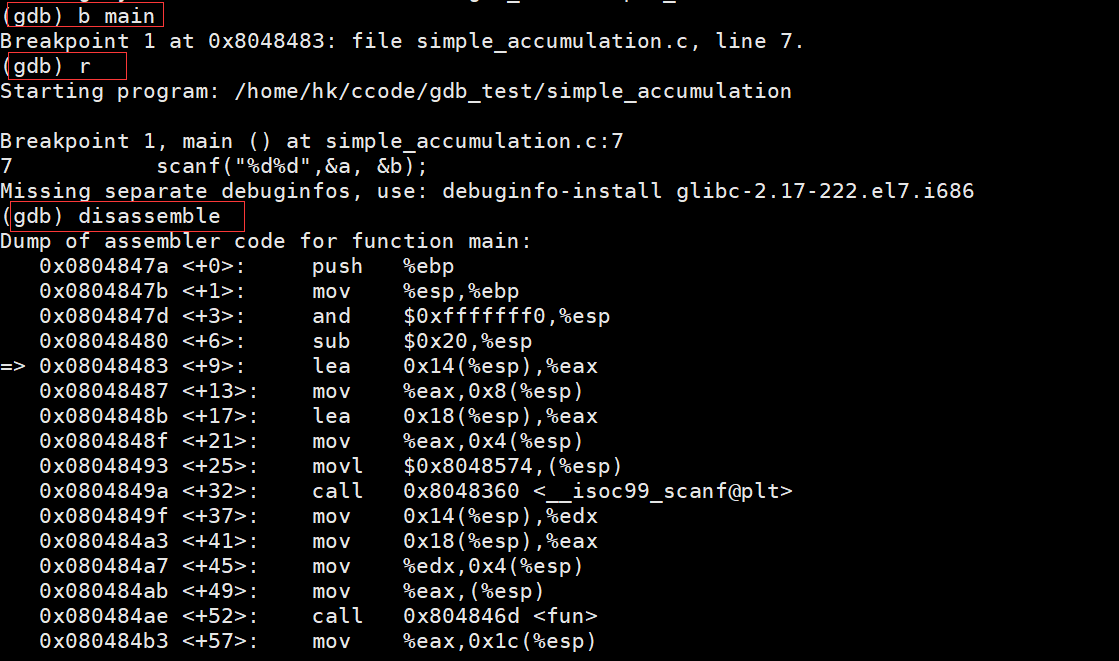
## 二、GDB调试汇编堆栈过程分析



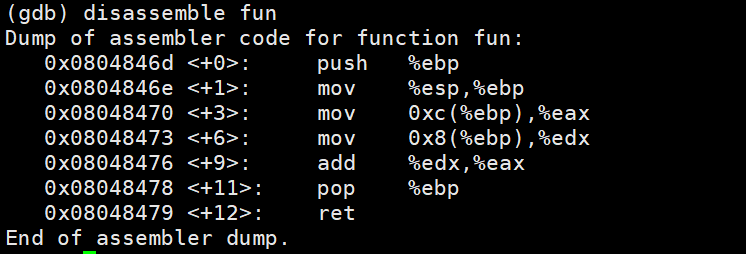
### 2.1过程分析

1. 通过gcc - g gdb.c -o gdb -m32指令在64位的机器上产生32位汇编。（向下兼容）
2. 进入gdb调试，先在main函数处设置一个断点，再run一下，使用**disassemble**指令获取汇编代码

Main函数汇编代码



Fun 函数的汇编代码



通过info registers列出使用的寄存器.

