

Arquitectura de Computadoras

Referencia GDB

| Comando | | Descripción |
|--------------------------------------|----------------------------------|---|
| Varios | | |
| nasm -g -F DWARF -f elf32 prog.asm | ld -g -m elf_i386 prog.o -o prog | nasm + ld |
| gcc -g -m32 prog.c -o prog | | gcc |
| tgdb | | abrir debugger |
| Propios de GDB | | |
| file <programa> | | abrir programa a debuguear |
| set args <arg1> <argn> | | correr el programa con argumentos |
| show args | | revisar argumentos al prog. |
| start | | comienza. Similar a "run" pero se detiene en main (run no) |
| b <addr> | | insertar breakpoint (ejemplos): <addr> = *0xFFFFABCD <addr> = _start |
| delete <n> | | delete breakpoint. <n> = segun lista de "dashboard breakpoints" |
| s | | step linea de código |
| si | | step instrucción assembly |
| n | | next (no meterse en la función) |
| finish | | correr hasta terminar función actual |
| continue | | correr hasta siguiente breakpoint |
| print <algo> | | mostrar valor de (ejemplos): <algo> = \$eax <algo> = <symbol> // nombre de var, nombre de argumento, etc <algo> = *0xFFFF0000 <algo> = *(char*)\$eax // casteo eax, y miro el valor del byte <algo> = {char}\$eax // idem anterior (notación alternativa) <algo> = sizeof(long) // reviso tamaño de datos usados al castear |
| quit | | finaliza gdb |
| Pertenecientes a Dashboard | | |
| dashboard | | refrescar la pantalla |
| dashboard breakpoints | | toggle on/off listado de breakpoints |
| dashboard memory watch 0xFFFF1234 20 | | visualizar 20 bytes desde el addr. |
| dashboard memory unwatch 0xFFFF1234 | | ocultar el rango de mem. mostrado |