Command	Description
`addr2line`	Converts program addresses into filenames and numbers by reading the debug symbol tables in an executable file. It is very useful when decoding addresses printed out in a system crash report.
`ar`	The archive utility is used to create static libraries.
`as`	This is the GNU assembler.
`c++filt`	This is used to demangle C++ and Java symbols.
`cpp`	This is the C preprocessor, and is used to expand `#define`, `#include`, and other similar directives. You seldom need to use this by itself.
`elfedit`	This is used to update the ELF header of ELF files.
`g++`	This is the GNU C++ front-end, which assumes source files contain C++ code.
`gcc`	This is the GNU C front-end, which assumes source files contain C code.

Command	Description
`gcov`	This is a code coverage tool.
`gdb`	This is the GNU debugger.
`gprof`	This is a program profiling tool.
`ld`	This is the GNU linker.
`nm`	This lists symbols from object files.
`objcopy`	This is used to copy and translate object files.
`objdump`	This is used to display information from object files.
`ranlib`	This creates or modifies an index in a static library, making the linking stage faster.
`readelf`	This displays information about files in ELF object format.
`size`	This lists section sizes and the total size.
`strings`	This display strings of printable characters in files.
`strip`	This is used to strip an object file of debug symbol tables, thus making it smaller. Typically, you would strip all the executable code that is put onto the target.