binutils Binary inspection tools

CONCEPTS

> Most tools read a file and never run it, so they are safe on untrusted binaries. ldd is the exception: it may execute the file.

Source cc -g -c
↓ compiles to
Object .o nm, objdump
↓ ar / ld
Archive .a / ELF readelf, size
↓ loaded as
Process ldd, addr2line

> Add -C to demangle C++ names; readelf, objdump and nm all accept it (readelf -C only for symbols).

> Stripped binaries have no .symtab: use nm -D or readelf --dyn-syms for the dynamic symbols that remain.

SYMBOLS (NM)

nm -C *<file>*
List symbols, demangled

nm -D *<file>*
Dynamic symbols of a shared lib

nm -u *<file>*
Undefined symbols it needs

nm -S --size-sort -C *<file>*
Biggest symbols last, with size

ELF STRUCTURE (READELF)

readelf -h *<elf>*
ELF header: type, arch, entry

readelf -S *<elf>*
Section headers

readelf -d *<elf>*
Dynamic section: NEEDED, RPATH

DISASSEMBLY (OBJDUMP)

objdump -d *<file>*
Disassemble code sections

DISASSEMBLY (OBJDUMP) (cont.)

objdump -d -C *<file>*
Disassemble, demangled names

objdump -d -M intel *<file>*
Intel syntax instead of AT&T

objdump -S *<file>*
Interleave source (needs -g)

objdump -d --disassemble=*<sym>* *<file>*
Only one function

STRINGS AND SIZES

strings *<file>*
Printable strings, 4+ chars

strings -n *<len>* *<file>*
Set minimum string length

size *<file>*
Text, data, bss totals

c++filt *<name>*
Demangle a C++ name

ADDRESSES AND LIBRARIES

addr2line -e *<elf>* *<addr>*
Address to file:line

addr2line -e *<elf>* **-f -C -i** *<addr>*
Add function, demangle, inlines

ldd *<bin>*
Shared libs it needs

readelf -d *<bin>* **| grep NEEDED**
Direct dependencies, no execution

file *<bin>*
Type, arch, stripped or not

STRIP, COPY, ARCHIVES

strip *<file>*
Remove all symbols and debug info

ar t *<lib.a>*
List archive members

ar rcs *<lib.a>* *<obj.o>*...
Create or update static lib