NAME

execstack − tool to set, clear, or query executable stack flag of ELF binaries and shared libraries

SYNOPSIS

execstack [OPTION...] [FILES]

DESCRIPTION

execstack is a program which sets, clears, or queries executable stack flag of ELF binaries and shared libraries. Linux has in the past allowed execution of instructions on the stack and there are lots of binaries and shared libraries assuming this behaviour. Furthermore, GCC trampoline code for e.g. nested functions requires executable stack on many architectures. To avoid breaking binaries and shared libraries which need executable stack, ELF binaries and shared libraries now can be marked as requiring executable stack or not requiring it. This marking is done through the p_flags field in the PT_GNU_STACK program header entry. If the marking is missing, kernel or dynamic linker need to assume it might need executable stack. The marking is done automatically by recent GCC versions (objects using trampolines on the stack are marked as requiring executable stack, all other newly built objects are marked as not requiring it) and linker collects these markings into marking of the whole binary or shared library. The user can override this at assembly time (through **−−execstack** or **−−noexecstack** assembler options), at link time (through **−z execstack** or **−z noexecstack** linker options) and using the **execstack** tool also on an already linker binary or shared library. This tool is especially useful for third party shared libraries where it is known that they don't need executable stack or testing proves it.

OPTIONS

−s −−set−execstack

Mark binary or shared library as requiring executable stack.

−c −−clear−execstack

Mark binary or shared library as not requiring executable stack.

−q −−query

Query executable stack marking of binaries and shared libraries. For each file it prints either **−** when executable stack is not required, **X** when executable stack is required or **?** when it is unknown whether the object requires or doesn't require executable stack (the marking is missing).

−V   Print **execstack** version and exit.

−? −−help

Print help message.

−−usage

Print a short usage message.

ARGUMENTS

Command line arguments should be names of ELF binaries and shared libraries which should be modified or queried.

EXAMPLES

execstack -s ˜/lib/libfoo.so.1

will mark ˜/lib/libfoo.so.1 as requiring executable stack.

execstack -c ˜/bin/bar

will mark ˜/bin/bar as not requiring executable stack.

execstack -q ˜/lib/libfoo.so.1 ˜/bin/bar

will query executable stack marking of the given files.

SEE ALSO

ld.so(8).

BUGS

execstack doesn't support yet marking of executables if they do not have PT_GNU_STACK program header entry nor they have room for program segment header table growth.

AUTHORS
Jakub Jelinek <jakub@redhat.com>.