ClangBuiltLinux Status Update (2022)

Nathan Chancellor, ClangBuiltLinux Meetup 2022

Introduction

- Started hacking on the Linux kernel through Android in 2016
- Started contributing to ClangBuiltLinux in 2018
- Became co-maintainer of ClangBuiltLinux in 2020
- Paid to work on ClangBuiltLinux since 2021 (thanks to the Linux Foundation and Google)

Brief History of ClangBuiltLinux

- 2012-2016: LLVMLinux, skeleton of building kernel with clang merged upstream
- 2017: Google Pixel 2 shipped with a clang built kernel, full support for building with clang merged upstream
- 2018: Chromebooks start shipping clang built kernels
- 2019: Support for `asm goto`, `ld.lld` and other LLVM utilities started seeing use
- 2020: Support for `asm goto` with outputs, minimum version of 10.0.1, `$(LLVM)` and `$(LLVM_IAS)` added
- 2021: LTO and CFI support merged upstream, integrated assembler turned on by default
- Full history on our wiki

Support Matrix

- Linux:
  - Latest LTS releases (4.9, 4.14, 4.19, 5.4, 5.10, and 5.15)
  - Latest stable release (5.19)
  - Mainline (6.0)
  - -next (as much as possible...)
- LLVM:
  - 11.0.0 through 16.0.0 (current tip of tree)
- Architectures:
  - arm, arm64, i386, hexagon, s390, and x86_64: Solid support
  - mips, powerpc, and riscv: Decent support
  - m68k: Nope
- Use newer versions of both kernel and toolchain!

How to Build Entirely with LLVM

With a kernel 5.15 and newer:

```
    $ make ARCH=...  LLVM=1
```

Implicit default of `LLVM_IAS` and need for CROSS_COMPILE in certain cases was changed in 5.15 so with a kernel 4.19 and newer:

```
    $ make ARCH=...  CROSS_COMPILE=...  LLVM=1 LLVM_IAS=1  ...
```

Some architectures do still need GNU binutils, more on that later.

Known Issues

- Many configurations are not `-Werror` clean so `CONFIG_WERROR=y` breaks the build
  - Clean builds are expected as much as possible
  - `ARCH=arm64` and `ARCH=x86_64` allmodconfig are currently `-Werror` clean
- Recent LLVM versions have gotten more aggressive with SCEV, which may result in 64-bit division libcalls getting inserted when there was no division in the original code (upstream issue)

Status of `ARCH=arm`

- No major outstanding issues with recent kernel (4.19+) and LLVM, other than `-Werror` for allmodconfig (older ISA versions might still have issues, v6 and v7 are known to work well)
- Integrated assembler only works with both kernel 5.13 and newer and LLVM 13 and newer
  - Use `LLVM_IAS=0` as necessary when using older LLVM and newer kernel.

Status of `ARCH=arm64`

- No major outstanding issues.

Status of `ARCH=mips`

- Most 32-bit configurations build fine. There are a couple of outstanding issues that show up at runtime, though the first might be resolved (report 1, report 2)
- Cannot build most 64-bit configurations out of the box due to lack of support for `CONFIG_MIPS32_O32`
  - Upstream LLVM issue
  - ClangBuiltLinux issue
- There are some compiler and assembler options that workaround hardware issues, which are not supported in LLVM (issue)

Status of `ARCH=powerpc`

- 32-bit: Integrated assembler may work depending on configuration, certain `-march` flags to assembler cause issues (issue 1, issue 2)
- 64-bit: ELFv1 is unsupported in ld.lld so big endian configurations (like `pseries_defconfig`) will error (issue), may be possible to switch to ELFv2 (series)
- Boot wrapper hardcodes GNU tools, which blocks getting rid of `CROSS_COMPILE` (issue)

Status of `ARCH=riscv`

- No major outstanding issues with recent kernel (5.15+) and LLVM (13.x+)
- Linker relaxation supported in ld.lld 15.0.0 but kernel currently still disables it (patch to enable it)

Status of `ARCH=s390`

- `ld.lld`, `llvm-objcopy`, and `llvm-objdump` do not work with s390; either use `LLVM=1` plus make variables to use GNU binutils or plain `CC=clang`
  - `ld.lld` report: ClangBuiltLinux, upstream
  - `llvm-objcopy` report: ClangBuiltLinux
  - `llvm-objdump` report: ClangBuiltLinux
- s390 requires clang 14.0.0 or newer as of 5.19 to properly support the integrated assembler, as opposed to 11.0.0 like the rest of the kernel

Status of ARCH=x86

- No major outstanding issues for 64-bit
  - x32 has a few issues (1, 2) but those are unlikely to impact most folks, as it requires using an x32 userspace, which is uncommon; workaround has been applied to kernel
  - Some outstanding `objtool` warnings
- Certain 32-bit configurations run into register exhaustion

Continuous integration

- As alluded to earlier, we have quite the matrix to support
- continuous-integration2, powered by TuxSuite, allows us to build hundreds of kernels across all supported trees and LLVM versions (778,000 builds since 2021!)
  - Mainline is built every six hours
  - Certain maintainer trees and Android are built daily with LLVM versions that are still being updated
  - Other tree and LLVM version combinations are built weekly

Tools

- boot-utils: A set of Python script to quickly boot test kernels with a simple buildroot-based rootfs using QEMU or UML
- tc-build: A set of Python scripts to build updated versions of binutils and LLVM

Getting involved

- Follow llvm@lists.linux.dev for patches and problem reports (archive, subscription instructions)
- Follow our issue tracker
  - See a bug that you have input on? Comment!
  - Testing with LLVM and see an error, warning, or other problem? File an issue (if one does not already exist)!
  - Looking for a way to contribute? Check out the good first issue label
- Integrate LLVM/clang into local continuous integration setups
  - Maintainers and developers that do this get access to a second set of warnings, which may reveal issues during development, such as uninitialized variables

Questions?

- Email: nathan@kernel.org
- GitHub: nathanchance
- Twitter: @nathanchance
- Website: https://nathanchance.dev