

ClangBuiltLinux



Update on the LLVM port of the Linux Kernel

Behan Webster – 2019.09.10

Acknowledgements

- Nick Desaulniers from Google is now the major contributor now to this effort
- Nick couldn't make Plumbers this year

Why bother with clang?

- Some projects want to use one compiler, which is clang
- Porting Linux to clang improves the code bases of both Linux and LLVM/clang as problems are found and fixed
- Using both gcc and clang helps find Undefined Behaviour as both compilers can interpret the C specification differently
- Helps to improve advanced C support in clang (clang is heavily used as a C++ and Objective-C compiler already)

ASM Goto

- ASM Goto support was added to clang
- It has largely been tested for use with the Linux kernel
- General ASM goto support elsewhere is less well tested
- Enables the use of CONFIG_JUMP_LABEL in a kernel compiled with clang
- CONFIG_JUMP_LABEL was recently made mandatory in X86

Inline Assembly

- Inline assembly has been improved to be more standardized
- This doesn't yet allow for the Integrated Assembler to be used however
- With fewer Gnu Assembler extensions being used, clang can more easily parse inline assembly
- Stephane Agner cleaned up the Unified Syntax in arch/arm
- More encodings and output parameters have also been added to Clang's assembly parser

LLD (LLVM Linker)

- LLVM has equivalent programs to those in binutils which are now being debugged to work with compiling the kernel
- A yet to be released Pixel will be shipped later this year with an LLD linked kernel
- LLD is a heavily multi-threaded linker with similar capabilities to gold
- LLD can provide significant speedups for incremental linkage
- LLD can be used with or without clang

Adoption of clang

- Google is generally moving away from gcc and binutils
- However there is a long tail of fixing features which are extensions in things like binutils and gnu-as

Android

- Android has moved away from gcc
- Any remaining gcc use is on a very old version of gcc
- Android Q requires OEMs to build their kernels with clang
- This provides ABI guarantees for debugging
- Requires OEMs ship a mainline kernel
- Libabigail is being used to make sure that the ABI is stable
- Android now ships cuttlefish (X86_64 dev platform with clang)

ChromeOS

- ChromeOS now uses clang exclusively on X86_64 and aarch64
- Arm32 will soon also be 100% clang

Google cloud services

- Google cloud servers are also moving over to clang
- There had been an issue with kexec which is now resolved
- Further testing is required before completely switching over

LTO

- Pixel 3 and newer will use LTO
- Android R will be shipping an LTO clang kernel
- Once working will be upstreamed to vanilla kernel
- Works with X86_64, arm32, aarch64, powerpc32, powerpc64 and MIPS32 (1 last patch to enable this)
- Arnd Bergman also played with getting it to work on Sparc and SystemZ
- Kernels 4.4, 4.9, 4.19, next and mainline

Testing

- There is now a CI builder on travis-ci, linaro/kernel-ci
- Support for arm and aarch64 in testing, X86_64 still a WIP
- Intel 0-day bot is now using clang, however currently only for Google (as it is still very noisy)
- There are different warnings from gcc and clang
- Testing includes both building and booting

MAINTAINERS

- The clangbuiltlinux mailing list has now been added to the maintainers file
- Clang related problems should end up being CCed to the mailing list now.

Companies now using clang

- The following companies are using clang (at least for some of their kernels)

Arm, Google, Intel, Linaro, Qualcomm, Samsung

Public meeting

- There is a public clangbuiltlinux meeting every 2 weeks
- A google calendar showing the time can be found on <https://clangbuiltlinux.github.io>
- The meeting is on Google Meet
<https://meet.google.com/yjf-jyqk-iaz>

Past talks

- Some of the past talks on using Clang to build Linux
<https://github.com/ClangBuiltLinux/linux/wiki/Talks,-Presentations,-and-Communications>

<https://clangbuiltlinux.github.io>
<https://github.com/ClangBuiltLinux/>
<https://github.com/ClangBuiltLinux/linux/wiki>
<https://github.com/ClangBuiltLinux/linux/issues>
clang-built-linux@googlegroups.com
[#clangbuiltlinux](#) on FreeNode