



Contribution ID: 363

Type: **not specified**

## BTF linking and deduplication in the Linux kernel using the toolchain

*Wednesday, 18 September 2024 15:50 (40 minutes)*

We are extending the CTF type format (in the GNU toolchain for several years now) by making its next file format version a superset of BTF, with the option to emit straight BTF as well; this means that the existing toolchain machinery (generation with GCC, deduplication and linking with GNU ld, reading and writing with libctf) will start working with BTF as well. (It will remain capable of generating CTF for any target, not only BPF.)

This means that we could integrate this into the kernel build process and both free pahole from the need to deduplicate BTF and also remove the need to generate DWARF at build time, speeding up builds.

libctf has some extra features that may be useful for the kernel's BTF generation process: it can deduplicate all in-tree modules and core kernel BTF against each other and store them in a fast-to-access, compact archive (entire enterprise kernels including all modules fit in 16MiB); it can identify and represent types with conflicting definitions; it can reliably detect incompatible parent containers so as to refuse to import them; and it can represent the various static-scope and global-scope variables found in kallsyms.

This activity introduces this machinery and asks for opinions on how (and if) to improve it with a view to integrating it into the kernel build process.

**Primary author:** ALCOCK, Nick (Oracle Corporation)

**Presenter:** ALCOCK, Nick (Oracle Corporation)

**Session Classification:** Toolchains Track

**Track Classification:** Toolchains Track