ENABLE INTEL LAM IN LINUX

H.J. Lu

Intel

August 2021

Intel LAM

Intel LAM (Linear Address Masking) Extension allows software to locate metadata in data pointers and dereference them without needing to mask the metadata bits. It supports:

- LAM_U48
  - Activate LAM for user data pointers and use of bits 62:48 as masked metadata.
- LAM_U57
  - Activate LAM for user data pointers and use of bits 62:57 as masked metadata.

Enable LAM in Linux

- Enable LAM on Linux is equivalent to porting Linux to a new architecture.
  - Only LAM enabled Linux on LAM processors can provide LAM features
  - Every piece of OS must be LAM enabled, starting from kernel, toolchain, libraries, …
    - A binary is LAM enabled only if all its components are LAM enabled.
- LAM enabled OS is backward compatible.
  - The same LAM-enabled OS binary can run on LAM and legacy processors.
    - Provide LAM features only on LAM processors.
    - Minimum performance loss on legacy processors.

Enable LAM in GCC

- GCC:
  - Enable memory tagging with LAM in x86 codegen
  - Enable LAM in HWASAN run-time:
    - Upstream is in LLVM repo.

Enable LAM in Binutils

- Linker: Properly mark programs as LAM enabled when all its components are marked as LAM enabled.

  -z lam-u48 Generate GNU_PROPERTY_X86_FEATURE_1_LAM_U48

  -z lam-u48-report=[none|warning|error] (default: none)

  Report missing LAM_U48 property

  -z lam-u57 Generate GNU_PROPERTY_X86_FEATURE_1_LAM_U57

  -z lam-u57-report=[none|warning|error] (default: none)

  Report missing LAM_U57 property

  -z lam-report=[none|warning|error] (default: none)

  Report missing LAM_U48 and LAM_U57 properties

Enable LAM in Glibc (C Run-time)

- Proposed <sys/tagged-address.h>: An API for tagged address for LAM and TBI:
  - https://sourceware.org/pipermail/libc-alpha/2021-August/129856.html
  - All bits between 0 and N - 1, where N is the number of tagged address bits, are used in address translation.
  - All pointers participating in a pointer arithmetic operation should have the same tag if they point to the same memory object so that pointer equality operation can be performed on tagged pointers.
- Avoid pointer operations incompatible with LAM.
  - memmove:
    - Mask out memory tags before pointer comparison.

LAM Kernel Interface

- LAM kernel API is an extension of CET kernel API.
- Extend arch_prctl () for LAM. X86 features:
  - IBT: GNU_PROPERTY_X86_FEATURE_1_IBT
  - SHSTK: GNU_PROPERTY_X86_FEATURE_1_SHSTK
  - LAM_U48: GNU_PROPERTY_X86_FEATURE_1_LAM_U48
  - LAM_U57: GNU_PROPERTY_X86_FEATURE_1_LAM_U57
- Before passing control to user space:
  - If the binary is marked with LAM_U48 enabled, kernel may enable LAM_U48.
  - Else if the binary is marked with LAN_U57 enabled, kernel may enable LAM_U57.

LAM Kernel Interface (cont. 1)

- Rename ARCH_X86_CET_STATUS to ARCH_X86_FEATURE_1_STATUS
- Rename ARCH_X86_CET_DISABLE to ARCH_X86_FEATURE_1_DISABLE
- Add ARCH_X86_FEATURE_1_ENABLE

/* Enable FEATURE_1 features in unsigned int features. Only LAM_U48 and LAM_U57 are allowed. */

#define ARCH_X86_FEATURE_1_ENABLE 0x3004

Linux LAM Run-time

At run-time, kernel starts loader of a dynamic application with LAM is enabled. Loader disables LAM if any loaded shared objects aren't LAM enabled. Loader issues an error when dlopening a legacy shared object from a LAM-enabled process.

- Configure option, --enable- lam=permissive:
  - Disable LAM when dlopening a legacy shared object from a LAM-enabled process.

Glibc Tunables for LAM

- Run-time control (Not applicable on SUID binaries):
  - Select LAM_U48 vs LAM_U47
    - LAM_U48: GLIBC_TUNABLES=glibc.cpu.x86_lam_size=48
    - LAM_U57: GLIBC_TUNABLES=glibc.cpu.x86_lam_size=57
  - Permissive
    - Disable LAM when dlopening a legacy shared object
    - GLIBC_TUNABLES=glibc.cpu.x86_lam=permissive
  - Always off
    - Always disable LAM.
    - GLIBC_TUNABLES=glibc.cpu.x86_lam=off
  - Always on
    - Never disable LAM.
    - GLIBC_TUNABLES=glibc.cpu.x86_lam=on

Enable LAM in Applications

•95% Linux applications are LAM compatible and can be marked as LAM enabled.

•Applications which reuse the upper bits in addresses must be updated for LAM:

- Database.
- OpenJDK.
- JS VMs.
- HTTP servers.

Software Status

- LAM has been enabled in binutils 2.36.
- Targeting GCC 12 for LAM support.
- Targeting glibc 2.34 for LAM support:
- Pending
  - <sys/tagged-address.h>: API for LAM and TBI.
  - LAM kernel API.

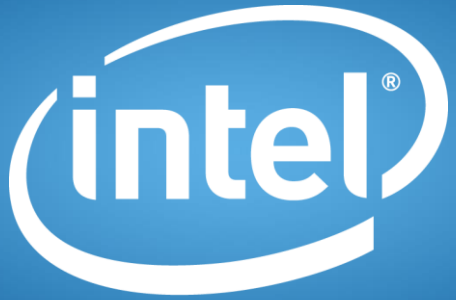
Linux Kernel Status

- LAM kernel API is an extension of CET kernel API.
- CET kernel patches are under review.
- LAM kernel patches have been submitted.

Call To Action

Enable LAM in the rest 5% of Linux OS.

- Other high level languages:
  - Rust
  - Go
  - …
- Browsers: Chrome, Firefox
  - Javascript
  - Sandbox.
- Java: OpenJDK.



Software