Разработка гипервизора Jinet

Захаров Илья Александрович
Школа №1533 ”ЛИТ”

Научный руководитель: Байков Борис Камалевич,
ведущий программист, руководитель группы, ОАО «Т-Платформы»

Цель работы – это создание минимального монитора виртуальных машин (гипервизора) с использованием механизмов аппаратной виртуализации архитектуры x86-64 (AMD64).

С каждым годом технологии виртуализации всё глубже и глубже входят в мир информационных технологий, находя применения в самых разных областях IT:

1. изоляция серверных систем для обеспечения их безопасности
2. эффективное сегментирование ресурсов компьютера
3. одновременное использование разных ОС на настольном компьютере
4. отладка гостевых систем

Большинство из ныне существующих гипервизоров массивны и поддерживают множество функций. Реализованный в рамках этой работы гипервизор может послужить базой для гипервизоров, заточенных под конкретные задачи. Мотивацией для работы стали интерес и актуальность технологий виртуализации и прямого программирования оборудования. Исследованы механизмы виртуализации, получен опыт разработки кода управления ими.

В ходе работы над проектом было изучено большое количество документации по процессорам Intel ([1]) и AMD ([2]), документация `gnu make` ([3]), `gnu ld` ([4]), `gcc` ([5]), `fasm` ([6]), формата ELF ([7]), таблиц ACPI ([8]).

Гипервизор Jinet был написан на языках программирования ассемблер (диалекты `fasm` и `as`) и C (компилятор `gcc`). Сборка проекта осуществляется с помощью сборщика `gnu ld` и утилиты `gnu make`. В качестве системы контроля версий используется `git`. Гипервизор позволяет запускать код в изолируемом окружении виртуальной машины. Код, инициализирующий подсистемы компьютера и режим VMX, можно использовать для демонстрации возможностей виртуализации, а также в учебных целях.

Исходный код проекта распространяется под лицензией MIT. `git`-репозиторий гипервизора располагается по адресу `https://github.com/jinet-vm/vmm`.

Литература

[1] Intel. Intel® 64 and IA-32 Architectures Software Developer Manual. 2017.

[2] AMD. AMD64 Architecture Programmer's Manual, Volume 2: System Programming. 2017.

[3] GNU Project and Free Software Foundation. GNU Make Manual.

[4] GNU Project and Free Software Foundation. GNU Linker Manual.

[5] GNU Project and Free Software Foundation. GCC Manual.

[6] Grysztar Tomasz. flat assembler 1.71.

[7] Portable Formats Specification Version 1.1. Executable and Linkable Format (ELF).

[8] Toshiba HP Intel Microsoft Phoenix. Advanced Configuration and Power Interface Specification, Revision 5.0a.

[9] Popek G. J., Goldberg R. P. Formal requirements for virtualizable third generation architectures // Communications of ACM. 1974.

[10] @Atakua. Аппаратная виртуализация. Теория, реальность и поддержка в архитектурах процессоров. `https://habrahabr.ru/company/intel/blog/196444/`. 2013.

[11] М. Чурдакис. The Infamous Trilogy: CPU internals, Virtualization, Raw multicore programming. `https://www.codeproject.com/articles/45788/the-real-protected-long-mode-assembly-tutorial-for`. 2015.

[12] OSDev Wiki. `http://wiki.osdev.org/Main_Page`.

[13] В. Садовников. Начала программирования в защищённом режиме. `http://e-zine.excode.ru/online/1/Introduction.html`. 2006.