

Исполняемый

В <u>информатике</u> **исполняемый код** , **исполняемый файл** или исполняемая **программа** , иногда просто называемая **исполняемым** файлом или **двоичным файлом** , заставляет компьютер «выполнять указанные задачи в соответствии с закодированными <u>инструкциями</u> » $\frac{[\ 2\]}{}$, в отличие от файла данных , который должен быть интерпретирован (анализирован) интерпретатором , чтобы быть функциональным. $\frac{[\ 3\]}{}$

Точная интерпретация зависит от использования. «Инструкции» традиционно означают инструкции машинного кода для физического процессора . [4] В некоторых контекстах файл, содержащий инструкции скрипта (например, <u>байт-код</u>), также может считаться исполняемым.

Шестнадцатеричный дамп исполняемого загрузчика реального режима. Первый столбец состоит из адресов первого байта во втором столбце, который содержит байты данных в шестнадцатеричной нотации (сначала младший байт), а последний столбец состоит из соответствующей формы ASCII. [1]

Генерация исполняемых файлов

Исполняемые файлы могут быть вручную закодированы на машинном языке, хотя гораздо удобнее разрабатывать

программное обеспечение в виде <u>исходного кода</u> на <u>языке высокого уровня,</u> который может быть легко понятен людям. В некоторых случаях исходный код может быть указан на <u>языке ассемблера</u>, который остается понятным человеку, но тесно связан с инструкциями машинного кода.

Язык высокого уровня компилируется либо в исполняемый файл машинного кода, либо в неисполняемый машинный код — объектный файл некоторого вида; эквивалентный процесс в исходном коде на языке ассемблера называется сборкой . Несколько объектных файлов связываются для создания исполняемого файла. Объектные файлы — исполняемые или нет — обычно хранятся в контейнерном формате , таком как Executable and Linkable Format (ELF) или Portable Executable (PE), который является специфичным для операционной системы . [5] Это дает структуру сгенерированному машинному коду, например, разделяя его на разделы, такие как .text (исполняемый код), .data (инициализированные глобальные и статические переменные) и .rodata (данные только для чтения, такие как константы и строки).

Исполняемые файлы обычно также включают в себя <u>систему времени выполнения</u>, которая реализует функции языка времени выполнения (такие как <u>планирование задач</u>, <u>обработка исключений</u>, вызов статических конструкторов и деструкторов и т. д.) и взаимодействие с операционной системой, в частности, передачу аргументов, окружения и возврат <u>статуса выхода</u>, вместе с другими функциями запуска и завершения работы, такими как освобождение ресурсов, таких как <u>дескрипторы файлов</u>. Для С это делается путем присоединения объекта <u>crt0</u>, который содержит фактическую точку входа и выполняет настройку и завершение работы путем вызова <u>библиотеки времени выполнения</u>. [6]

Исполняемые файлы, таким образом, обычно содержат значительный дополнительный машинный код, выходящий за рамки того, что напрямую сгенерирован из конкретного исходного кода. В некоторых случаях желательно опустить это, например, для разработки встраиваемых систем или просто для понимания того, как работают компиляция, компоновка и загрузка. В С это можно сделать, опустив обычную среду выполнения и вместо этого явно указав скрипт компоновщика, который генерирует точку входа и обрабатывает запуск и завершение работы, например, вызывая mainstart и возвращая статус выхода ядру в конце. [7]

Исполнение

Для того, чтобы быть выполненным системой (например, операционной системой, прошивкой или загрузчиком), исполняемый файл должен соответствовать двоичному интерфейсу приложений системы (ABI). В простых интерфейсах файл выполняется путем загрузки его в память и перехода к началу адресного пространства и выполнения оттуда. $\begin{bmatrix} 8 & 1 \\ 1 & 1 \end{bmatrix}$ В более сложных интерфейсах исполняемые файлы имеют дополнительные метаданные, указывающие отдельную точку входа . Например, в ELF точка входа определяется в e_entryполе заголовка, которое указывает (виртуальный) адрес памяти, с которого следует начать выполнение. [9] В коллекции компиляторов GNU это поле устанавливается компоновщиком на основе _startсимвола. [10]

Смотрите также

- Сравнение форматов исполняемых файлов
- Сжатие исполняемого файла
- Исполняемый текст

Ссылки

- 1. Челови, Пол (2002). <u>Встроенная кулинарная книга FreeBSD</u> (https://books.google.com/books?id=E4SFkfsOVaoC&pg=PA188) . Эльзевир . стр. 108, 187–188 . <u>ISBN.</u> 1-5899-5004-6. Получено 2022-03-06 .
- 2. "executable" (http://www.merriam-webster.com/dictionary/executable) . Онлайн-словарь Merriam-Webster . Merriam-Webster . Получено 2008-07-19 . (http://www.merriam-webster.com/dictionary/executable)
- 3. Мюллер, Джон Пол (2007). *Администрирование Windows в командной строке для Windows Vista, Windows 2003, Windows XP и Windows 2000.* (https://books.google.com/books?id=Ua819DVKQPwC&pg=PA24) John Wiley & Sons . стр. 24. ISBN (https://books.google.com/books?id=Ua819DVKQPwC&pg=PA24) 978-0-470-04616-6. Получено 2023-03-06 .
- 4. "Машинные инструкции" (https://www.geeksforgeeks.org/machine-instructions/) . GeeksforGeeks . 2015-11-03 . Получено 2019-09-18 . (https://www.geeksforgeeks.org/machine-instructions/)
- 5. "Глава 4: Объектные файлы" (https://refspecs.linuxbase.org/elf/gabi4+/ch4.intro.html) . refspecs.linuxbase.org . Получено 18.09.2019 . (https://refspecs.linuxbase.org/elf/gabi4+/ch4.intro.html)
- 6. Фишер, Тим. <u>"Список расширений исполняемых файлов" (https://www.lifewire.com/list-of-executable-file-extensions-2626061)</u>. *lifewire.com*. Получено 18.09.2019 . (https://www.lifewire.com/list-of-executable-file-extensions-2626061)
- 7. Маккеллар, Джессика (16.03.2010). «Привет из мира, свободного от libc! (Часть 1)» (http s://blogs.oracle.com/ksplice/enry/hello_from_a_libc_free) . (https://blogs.oracle.com/ksplice/en