**Сборка ПО**, также известная как **интеграция** - процесс преобразования файлов с исходным кодом и их компиляция в артефакты, составляющие приложение, такие как бинарные и исполняемые файлы.

Одним из шагов создания сборки является процесс компиляции исходного кода, где файлы преобразуются в промежуточный код. Процесс компиляции в случае, если функция обозначена как внешняя, не проверяет действительно ли такая функция существует.

После компиляции происходит процесс связывания (нахождение реального положения всех функций, обозначенных как внешние). Это выполняется специальной программой - линковщиком. Процесс линковки представляет собой замену относительных адресов функций внешних библиотек на реальные адреса которые будут использоваться программой в процессе её выполнения.

Со сборкой связано понятие версии приложения. Линковщик часто может автоматически устанавливать (увеличивать) номер версии.

Время сборки может сильно варьироваться. Для примера: сборка ядра Linux может занимать от 20 минут до нескольких часов в зависимости от конфигурации ядра и технических параметров компьютера.

Поэтому, если исходный код в том или ином файле не изменялся, то часто код не компилируют заново – это слишком затратно по времени.

Усовершенствованные утилиты сборки также пытаются воздерживаться от повторной компиляции ненужного кода, чтобы сократить время, необходимое для завершения сборки.

**Воспроизводимость сборок**

Часто возникает необходимость узнать, как конструировалась сборка программного продукта и что при этом происходило.

Воспроизводимость – это возможность вернуться назад во времени и воспроизвести данную версию программной системы или среды разработки, существовавшей ранее. Воспроизводимость необходима для решения проблем процесса разработки. Она позволяет узнать, как конструировалась сборка программного продукта, что при этом происходило, а потому эффективно исправлять дефекты в представляемых продуктах, гарантировать, что проект отвечает установленным требованиям и используется корректный код для построения исполняемых программ.

**Повторяемость сборок**

Повторяемость сборки подтверждает, что исполняемый файл собран именно из заявленных исходных текстов и не содержит посторонних закладок, которые, например, могут быть интегрированы в результате использования поражённого из-за атаки компилятора или сборочного инструментария.

Без возможности проверить тождественность бинарной сборки пользователю остаётся лишь довериться выполнившему сборку сервису.

Для обеспечения повторяемости требуется учитывать множество нюансов: при сборке обычными средствами бинарные файлы отличаются и не совпадают с пакетами из репозитория байт в байт из-за использования разных версий компилятора, линковщика, библиотек, различий во времени сборки, разных настроек сборочного окружения и т.п. Для обеспечения повторяемости необходимо полностью воссоздать эталонное сборочное окружение, использовать те же версии сборочного инструментария, идентичный набор опций и настроек по умолчанию.

Один из разработчиков KDE (K Desktop Environment) попытался проанализировать насколько реально воссоздать бинарный файл, идентичный распространяемым готовым сборкам, при наличии полных исходных текстов. Таким образом была оценена возможность проверки собран ли представленный исполняемый файл из указанных исходных текстов и содержит ли то, что заявлено создателями сборки.

В качестве эксперимента было произведено сравнение исполняемых файлов, поставляемых в составе дистрибутивов Debian и Fedora, с исполняемым файлом, собранным из доступных пакетов с исходными текстами. В теории, зная все параметры сборки и версии используемых при сборке компонентов, можно воссоздать исходное сборочное окружение, и если попытаться собрать в нем пакет с исходными текстами, то получаются тождественные исполняемые файлы. На практике всё оказалось не так просто, при каждой сборке получались разные исполняемые файлы. Основной причиной различий является сохранение в исполняемом файле данных, связанных со временем сборки.

При оценке пересборки пакетов Debian результаты полностью оправдали предположения - различие составило 20 байт, и в этих байтах содержалась дата сборки. Для Fedora не удалось достигнуть повторяемости или хотя бы объяснить причину значительных различий. В качестве наиболее вероятной гипотезы отмечаются возможные отличия в сборочной среде - не факт, что аналогичное окружение было использовано разработчиками, а даже при незначительных изменениях используемого инструментария результат меняется кардинально.

Полученные результаты свидетельствуют, что при распространении бинарных сборок недостаточно открывать доступ к коду под свободными лицензиями. Без приведения точной спецификации сборочной среды невозможно проверить на основе представленных исходных текстов, собрано приложение по ним или в нём использованы какие-либо модификации.

Примечательно, что ни одна свободная лицензия не учитывает данное обстоятельство и не требует указания точных параметров сборочного инструментария при публикации бинарных версий. В качестве одного из способов решения проблемы называется включение в состав исполняемого файла информации о компонентах окружения, в котором была произведена сборка.