

ОСОБЕННОСТИ АВТОМАТИЗАЦИИ ТЕСТИРОВАНИЯ ПРОГРАММНО-АППАРАТНЫХ СЗИ

На начальном этапе развития любого продукта в большинстве случаев превалирует использование тестирования вручную, но в дальнейшем, когда его функционал растёт и расширяется, ручное тестирование становится неоптимальным и не даёт ожидаемых результатов. Из-за этого возникает необходимость оптимизации ручной работы и использования средств автоматизированного тестирования в сочетании с тестированием, выполняемым вручную. Однако некоторые компании все же в процессе разработки пытаются использовать автоматизированное тестирование с самого первого этапа развития продукта.

Помимо автоматизированного и ручного видов тестирования существует ещё один вид — полуавтоматизированное тестирование. В этом случае часть проверок выполняется специализированными сторонними программными средствами автоматизации, а часть — тестировщиком вручную. Полуавтоматизированное тестирование используется в тех случаях, когда по каким-то причинам, не связанным с возможностью автоматизации тестирования (например, недостаточно денег на разработку автоматизированных тестов), невозможно использовать автоматизированное тестирование или в силу каких-то особенностей конкретного продукта его использование ограничено.

Вне зависимости от того, какой вид тестирования превалирует в процессе разработки продукта и на каком этапе разработки имеет место внедрение автоматизированного тестирования, необходимо заранее оценить все особенности и нюансы использования автоматизации для тестирования продукта, а также понять, смогут ли существующие программные средства автоматизированного тестирования решить поставленные перед ним задачи, или необходимо разрабатывать собственные.

Не исключением в данном случае является и разработка средств защиты информации (СЗИ). Тестирование программных СЗИ обычно выполняется в соответствии со всеми правилами тестирования программного обеспечения (ПО), которое согласно общепринятой классификации разделяется на несколько основных видов тестирования ПО: функциональное, нефункциональное (в том числе тестирование на отказ, нагрузочное и стрессовое тестирование) и регрессивное (тестирование после исправления ошибок) [1]. При выполнении любого из этих видов тестирования для программных СЗИ можно использовать как ручное, так и автоматизированное или полуавтоматизированное тестирование по аналогии с тестированием ПО.

Помимо программных СЗИ существуют программно-аппаратные, состоящие из программного обеспечения и аппаратной базы, реализующей необходимую аппаратную функциональность СЗИ. Такие СЗИ также должны тестироваться, но уже не только как программное обеспечение (программное СЗИ), а с учетом наличия аппаратной составляющей. Этот факт обуславливает особенности тестирования программно-аппаратных СЗИ, которые наблюдаются в процессе разработки тестов и проведения тестирования.

Целью данной статьи является проведение анализа возможности и необходимости применения автоматизированного тестирования и использования различных (с точки зрения автоматизации) видов тестирования для программно-аппаратных СЗИ с учетом особенностей таких СЗИ, обусловленных наличием аппаратной составляющей.

Статья содержит описание достоинств и недостатков автоматизированного тестирования по сравнению с ручным тестированием, выводы о необходимости использования обоих видов тестирования, классификацию программно-аппаратных СЗИ, описание попыток применения автоматизированного тестирования к СЗИ этих типов, а также описание особенностей автоматизированного тестирования программно-аппаратных СЗИ.

Рассмотрим преимущества и недостатки автоматизированного тестирования по сравнению с ручным по следующим параметрам [3]:

- повторяемость — все автоматизированные тесты выполняются однообразно, то есть исключен «человеческий фактор». Это является как достоинством, так и недостатком автоматизированного тестирования, так как тестировщик, проводя тест вручную, может обратить внимание на некоторые детали (которые недоступны автоматизированному тесту) и, проведя несколько дополнительных операций, найти ошибку;
- скорость выполнения — в процессе проведения автоматизированных тестов не используются инструкции и документация, это сильно экономит время проверки и является достоинством автоматизированного тестирования. Тестировщик, в свою очередь, должен проверять соответствие выполняемых им действий последовательности, указанной в программе и методике испытаний (ПМИ), и в случае выявления ошибки сравнивать полученные результаты с документацией на предмет корректности их трактовки (возможно, выявленная ошибка — это вовсе не ошибка, а правильное поведение программы);
- затраты времени на анализ результатов тестирования и адаптацию тестов в случае изменения функционала — анализ результатов тестирования автоматических тестов при изменении функционала продукта, их адаптация и повторное проведение тестирования требуют меньше времени, чем аналогичные действия того же объема тестирования вручную, это является достоинством автоматизированного тестирования по сравнению с ручным;
- формирование и рассылка отчетов — автоматизированные тесты могут автоматически рассылать и сохранять отчеты о результатах тестирования. В случае ручного тестирования эти действия должен выполнить тестировщик;
- выполнение без вмешательства человека — процесс выполнения тестов не требует наличия тестировщика, тесты могут выполняться в нерабочее время. Это является достоинством автоматизированного тестирования по сравнению с ручным;
- временные затраты на разработку — разработка автоматизированных тестов — достаточно сложный процесс, так как выполняется разработка программы, которая тестирует другую программу. Это является недостатком автоматизированного тестирования. При ручном тестировании необходимо выполнять разработку ПМИ, что занимает тоже достаточно большое время, но значительно меньшее, чем разработка автоматизированных тестов;
- стоимость инструмента для автоматизации — при использовании для автоматизации тестирования лицензионного ПО необходимо учитывать его стоимость, которая может быть достаточно высока, поэтому можно использовать свободно распространяемые инструменты, но они, как правило, отличаются более простым функционалом и меньшим удобством для работы. Это недостаток автоматизированного тестирования. В случае с ручным тестированием никаких дополнительных инструментов не требуется;
- пропуск мелких ошибок — автоматизированный тест может пропускать мелкие ошибки, на проверку которых он не направлен, такие как неточности в позиционировании окон, ошибки в надписях, которые не проверяются, ошибки форм, с которыми не осуществляется взаимодействие во время выполнения теста, и т. п. Данный недостаток автоматизированного тестирования проявляется гораздо реже при ручном тестировании: тестировщик может обратить (или не обратить) внимание на описанные особенности работы;
- проверка дружелюбности интерфейса — автоматизированный тест не может оценить дружелюбность, цветовую гамму, красоту и стиль интерфейса разрабатываемого продукта, это существенный недостаток автоматизации. С другой стороны, при ручном тестировании мнение тестировщика в данном вопросе может быть субъективным и не отражать реального состояния дел.

Из сравнения приведенных достоинств и недостатков автоматизированного и ручного тестирования можно сделать вывод, что внедрение автоматизации в процесс разработки значительно

сокращает время на проведение проверок и становится возможным проведение тестирования, которое человек сможет выполнить только за достаточно большой период времени, а возможно, и никогда не сможет. Особенно это касается нефункционального стрессового и нагрузочного тестирования, которое в процессе выявления ошибок может потребовать значительного количества однообразных действий от тестировщика, выполнение которых становится невозможным при стремлении этого количества к предельно возможному значению.

Вторым выводом из описанных преимуществ и недостатков автоматизированного и ручного тестирования является то, что в процессе разработки продукта необходимо использовать оба вида тестирования — ручное и автоматизированное, так как использование ручного тестирования позволит нейтрализовать часть недостатков автоматизированного тестирования и наоборот. В случае, когда существуют ограничения на проведение автоматизированных тестов, будет иметь место полуавтоматизированное тестирование.

Как уже было сказано, программно-аппаратные СЗИ состоят из ПО и аппаратного компонента. Рассмотрим возможность использования автоматизации их тестирования. Можно предположить, что по аналогии с автоматизированным тестированием ПО в данном случае можно использовать скрипты (автоматизированные тесты, разработанные при помощи специальных программ) для эмуляции действий пользователя, но существует ряд проблем при их использовании для тестирования программно-аппаратных СЗИ.

Перед рассмотрением этих проблем разделим программно-аппаратные СЗИ на две группы, подход к автоматизации тестирования которых будет различаться в связи с различием в способе их функционирования:

- СЗИ, функционирующие до момента загрузки операционной системы (ОС);
- СЗИ, функционирующие в ОС.

Средства защиты информации, функционирующие до старта ОС, представляют собой аппаратный модуль, подключаемый в средство вычислительной техники (СВТ) и перехватывающий на себя управление до момента загрузки ОС. Примером такого СЗИ является средство защиты информации от несанкционированного доступа (НСД) «Аккорд-АМДЗ».

СЗИ НСД «Аккорд-АМДЗ» — это аппаратный модуль доверенной загрузки (АМДЗ) для IBM-совместимых ПК — серверов и рабочих станций локальной сети, обеспечивающий защиту устройств и информационных ресурсов от несанкционированного доступа.

Принцип работы СЗИ НСД «Аккорд-АМДЗ» следующий: контроллер перехватывает управление сразу после выполнения штатного BIOS ПК и позволяет выполнить дальнейшую загрузку ОС только после прохождения ряда контрольных процедур, а именно:

- идентификация/аутентификация пользователя;
- проверка целостности технических и программных средств ПК (с использованием механизма пошагового контроля целостности).

Также «Аккорд-АМДЗ» позволяет выполнять загрузку различных ОС только с заранее определенных постоянных носителей информации (например, только с жесткого диска). Таким образом, обеспечивается доверенная загрузка ОС.

В случае попыток использования автоматизированных тестов, эмулирующих действия пользователя, для тестирования «Аккорд-АМДЗ» возникает следующая проблема: скрипты могут функционировать только в операционной системе, поэтому их использование для тестирования программно-аппаратных СЗИ, функционирующих до старта ОС, является невозможным. Это одна из главных проблем тестирования СЗИ, функционирующих до старта ОС.

Для выяснения возможности хотя бы частичной автоматизации разделим тестирование рассматриваемых СЗИ на две части: тестирование исполняемого кода СЗИ отдельно от аппаратной

составляющей (например, в среде хостовой ОС) и тестирование конечного функционала СЗИ, доступного пользователю (то есть исполняемого кода вместе с аппаратной составляющей).

Исполняемый код СЗИ может быть запущен на выполнение не в среде ОС аппаратного устройства (в данном случае контроллера «Аккорд-АМДЗ»), а в среде аналогичной ОС. Такая ОС может быть либо установлена на СВТ в качестве реальной ОС, либо развернута в виртуальной машине. Для исполнения кода СЗИ в виртуальной машине должна быть предусмотрена возможность «пробрасывания» аппаратного компонента в конкретной среде виртуализации, что часто бывает невозможно (в нашем случае это невозможно за счет того, что среды виртуализации не позволяют «пробросить» PCI/PCI-Express-устройство). В случае же с реальной ОС таких проблем не возникает. Следовательно, в рамках ОС СВТ можно автоматизировать процесс проверки корректности исполнения различных модулей, провести нагрузочное и стрессовое тестирование, проверить корректную работу интерфейса. Но в данном случае нельзя гарантировать корректность работы финального продукта, так как этот код должен также корректно выполняться и в среде ОС контроллера «Аккорд-АМДЗ». Таким образом, дополнительно необходимо выполнять ручное тестирование по ПМИ, которое также можно автоматизировать, но лишь частично. При этом не всегда получится отслеживать результаты всевозможных проверок. В случае тестирования отдельно исполняемого кода СЗИ в ОС СВТ можно получить определенный результат (код возврата, код ошибки и т. п.), на основе которого можно сказать об успешности проверки. В случае же аппаратного компонента такой результат получить нельзя за счет того, что все выполняется внутри контроллера и не передается наружу. Единственное, что удастся отследить при автоматизации тестирования СЗИ целиком, — это корректность отработки функционала контроллера (полностью, без конкретики того, что могло отработать не так), причем в случае неуспешной отработки все равно требуется ручное вмешательство.

На основании этого можно сделать вывод, что в случае с программно-аппаратными СЗИ, функционирующими до старта ОС, можно использовать только полуавтоматизированное тестирование, когда часть действий по тестированию может быть автоматизирована, а часть должен выполнять тестировщик вручную.

Программно-аппаратные средства защиты информации, функционирующие в ОС, представляют собой подключаемый в СВТ (как правило, в его USB-порт) аппаратный модуль и программное обеспечение, выполняющее взаимодействие пользователя с аппаратной базой из операционной системы.

Подход к автоматизированному тестированию в данном случае несколько другой, нежели для СЗИ, функционирующих до старта ОС.

Рассмотрим способы и особенности автоматизации тестирования программно-аппаратных средств, функционирующих в ОС, на примере персонального средства криптографической защиты информации (ПСКЗИ) ШИПКА.

Первый способ — это классический способ, уже рассмотренный выше, — использование автоматизированных скриптов, разработанных при помощи специальных программ автоматизации. Скрипты позволяют проводить функциональное и нефункциональное тестирование ПСКЗИ ШИПКА. Так как ПСКЗИ ШИПКА состоит из программного обеспечения и аппаратного компонента, то обязательно нужно учитывать при организации тестирования следующее: в ходе функционального тестирования ПСКЗИ ШИПКА аппаратную часть продукта необходимо переподключать к USB-порту ПК, что нарушает идеологию проведения автоматизированных тестов, которые должны проводиться без участия тестировщика. Поэтому полностью автоматизировать такое функциональное тестирование затруднительно, так как в данном случае необходимо участие тестировщика для выполнения указанных действий. Аналогичная проблема возникает при нефункциональном тестировании, так как в случае возникновения ошибок в ходе выполнения скрипта устройство необходимо физически переподключить к ПК.

Второй способ автоматизации тестирования ПСКЗИ ШИПКА представляет собой использование комплекта разработчика (SDK – Software Development Kit) для стандартных интерфейсов, который реализован для ПСКЗИ ШИПКА в целях возможности использования криптографической функциональности устройства сторонними разработчиками в своих продуктах. Для ПСКЗИ ШИПКА реализовано три стандартных интерфейса: Crypto API, PKCS#11, APDU, а SDK для них включает набор примеров по их использованию. Эти примеры можно использовать для автоматизации функционального тестирования ПСКЗИ ШИПКА, так как их успешное прохождение гарантирует, не полностью, но частично, что утилиты, которые работают на базе этих интерфейсов, будут работать с ПСКЗИ ШИПКА корректно. При этом остается необходимость функционального и нефункционального тестирования самих утилит продукта.

Итак, можно сделать вывод, что ручное и автоматизированное тестирование — две неотъемлемые части при проверке качества любого выпускаемого продукта, в том числе и программно-аппаратного СЗИ. Невозможно полностью автоматизировать все тесты, так как при работе, например, с программно-аппаратным СЗИ, функционирующим в ОС, — ПСКЗИ ШИПКА, может потребоваться переподключение устройства в USB-порт ПК. А при тестировании программно-аппаратных СЗИ, функционирующих до старта ОС, таких как СЗИ НСД «Аккорд-АМДЗ», существует ряд проблем, которые нельзя встретить при проверке программных СЗИ, а также программно-аппаратных СЗИ, функционирующих в ОС. Это обусловлено тем, что тестирование аппаратного компонента без возможности доступа к его функциональности из ОС достаточно затруднительно. Также при использовании автоматизированного тестирования не получится проверить дружелюбность интерфейса и т. п.

Однако внедрение автоматизированного тестирования значительно сокращает время на проведение проверок, и с его помощью становится возможным провести тестирование, которое тестировщик сможет выполнить только за достаточно большой период времени, а возможно, и никогда не сможет, поскольку, сложно представить, как тестировщик вручную тестирует предельные значения, например, в случае с ПСКЗИ ШИПКА, проверяет корректность функционирования устройства с максимально возможным количеством криптографических ключей, которое может быть больше 200. Для этого тестировщику нужно вручную сгенерировать все эти ключи и только потом проверить корректность функционирования устройства. Если же в данном процессе возникнут ошибки, то нужно будет поэтапно уменьшать количество ключей и повторять тестирование до определения критического значения. Также достаточно трудной задачей для тестировщика является ручное выполнение стрессового тестирования, при котором необходимо генерировать различные случайные значения и подавать их в качестве входных параметров в устройство для проверки реакции ПСКЗИ ШИПКА или «Аккорд-АМДЗ» на каждое из них. В случае с ПСКЗИ ШИПКА, например, необходимо задавать различные случайные значения длины PIN-кода и проверять, как устройство будет реагировать на каждое из введенных значений. В случае с «Аккорд-АМДЗ» — вводить всевозможные случайные значения пароля пользователя и т. п. Эти факты подтверждают правильность подхода, предполагающего использование автоматизированного тестирования наряду с ручным. В любом случае, при попытках автоматизации тестирования программно-аппаратных СЗИ необходимо учитывать рассмотренные в данной статье его особенности.

СПИСОК ЛИТЕРАТУРЫ:

1. Синицын С. В., Налютин Н. Ю. Верификация программного обеспечения. М.: Бином, 2008. — 157 с.
2. Степанченко И. В. Методы тестирования программного обеспечения. Волгоград: РПК «Политехник», 2006. — 74 с.
3. Антонов В. Автоматизированное тестирование программного обеспечения. 2013. URL: <http://www.protesting.ru/automation/> (дата обращения: 09.05.2013).

