A.P. Durakovskiy, I.A. Korsakov, A.A. Abramov

Automated Testing of Information Security Devices, Which Represented as Removable Media, and its Issues

Keywords: automated testing, information security devices, removable media.
This article describesissues, arising in automatedtesting of information security devices, which represented as removable media. These devices have their own operational system, whichimposes restrictions on its testing process. This paper proposes several new methods for automated testing of such devices.

А.П. Дураковский, И.А. Корсаков, А.А. Абрамов

ОСОБЕННОСТИ ТЕСТИРОВАНИЯ СЗИ, ПРЕДСТАВЛЕННЫХ В ВИДЕ СЪЕМНЫХ НОСИТЕЛЕЙ

В настоящее время сети передачи данных приобрели широкое распространение и часто используются для передачи конфиденциальной информации и в других целях, требующих гарантий безопасности информации. Однако зачастую среда является недоверенной (каналы передачи, оконечные устройства и т.д.), и в связи с этим возникает необходимость организации доверенного сеанса связи для защищенной работы удаленных пользователей с сервисами распределенной информационной системы через сети передачи данных [1].

В настоящее время на рынке присутствуют средства защиты информации, представляющие собой программно-аппаратные комплексы, основной задачей которых является построение доверенного сеанса связи [2].

Чтобы убедиться в корректном и безопасном функционировании данных устройств, необходимо, конечно же, провести их тестирование. Рассматриваемые устройства имеют собственную ОС, поэтому возникает ряд особенностей при их тестировании:

Многие требования, выдвигаемые к операционной системе не могут быть проверены в автоматическом режиме (с одной стороны, в связи со сложностью проверки самого требования, с другой – в связи с невозможностью в некоторых случаях проверить наличие решения под функциональное требование, не имея специализированных доступов, например на уровне ядра ОС или исходным текстам ОС). Кроме того, выбранные в рамках работы решения ОС, используемые в объекте тестирования, зачастую имеют сертификаты ФСТЭК или ФСБ, в связи с этим задача проверки функциональных решений, удовлетворяющих требованиям в рамках ОС решается посредством определения версии операционной системы и сверки полученной версии с эталонной.

Многие функции операционной системы реализуются за счет сторонних компонентов (не включенных в ядро ОС), в связи с этим тесты, направленные на проверку соответствия ОС функциональным требованиям, выполняются формальной проверкой как наличия в системе демонов и пакетов, обеспечивающих то или иное функциональное требование, так и активностью того или иного пакета или демона в момент выполнения теста.

Некоторые типовые решения по проверке ОС требуют некоторых дополнительных прав, таких, например, как создание и удаление пользователей в тестируемой системе, изменение прав доступа этих пользователей к различным объектам. В результате выполнения тестов все учетные записи, использованные исключительно в тестовых це-

лях, должны быть выключены и удалены, с целью предотвращения возникновения изъянов в системе безопасности, за счет известных учетных данных.

Как уже отмечалось, основной особенностью таких устройств является наличие собственной ОС. Для проведения тестирования необходимо загрузить ОС данного устройства с ЭВМ, на которой будет производиться тестирование.

В связи с этим возможны три способа проведения тестирования таких устройств:

1) вызов функций непосредственно из тестируемой ОС;

2) использование средств виртуализации;

3) использование сетевых протоколов удаленного управления.

Каждый из описанных методов имеет ряд преимуществ и недостатков.

Рассмотрим способ автоматизированного тестирования, заключающийся в вызове функций непосредственно из тестируемой ОС. Такой подход требует достаточного большого объема работ по организации тестирования непосредственно в процессе разработки СЗИ, так как на СЗИ должен присутствовать некий интерфейс, посредством которого и будет осуществляться вызов тестируемых функций. Безусловно, с точки зрения реализации данный метод является одним из самых надежных в связи с тем, что присутствует минимальное число ограничений для его использования. На этапе выполнения тестов нет необходимости каким-либо образом изменять тестируемую систему, так как она уже содержит в себе среду для тестирования. Однако отсюда вытекают и существенные недостатки этого подхода: система должна содержать в себе среду для тестирования, которая, так или иначе, потребляет ресурсы ЭВМ, на которой проводится запуск СЗИ (помимо самой среды, ОС должна содержать все необходимые компоненты для ее функционирования). Кроме того, тестовая среда имеет доступ ко всем функциям тестируемой ОС, при этом некоторые из этих функций обычно доступны только администратору системы, наличие среды для тестирования создает дополнительную угрозу, поскольку если рядовой пользователь или злоумышленник получит доступ к интерфейсу для тестирования, то он может полностью контролировать систему. Поэтому, используя такой подход при организации тестирования, требуется постоянно помнить о наличии угрозы использования злоумышленником/пользователем интерфейса для проведения атаки и, фактически, к списку функции устройства необходимо добавить: «невозможность доступа неуполномоченного пользователя к интерфейсу для тестирования системы». Естественно, это приводит к дополнительным сложностям при создании СВТ. Использовать данный подход разумно, если тестируемая система достаточно сложна и команда тестирования может быть привлечена к процессу разработки системы. Тестируемая система рассматривается как «белый ящик». Кроме того, она может быть разбита на отдельные части, которые могут быть протестированы как независимо друг от друга, так и во взаимодействии между собой, что существенно облегчает локализацию ошибок, однако, как уже говорилось, объем работ по организации тестированияпо сравнению с другими методами существенно больше.

Теперь рассмотрим подход с использованием средств виртуализации. Он может успешно применяться во всех видах тестирования, где система может представлять из себя «черный» или «серый» ящик. Во многом данный подход позволяет предоставить наиболее корректные результаты теста, в связи с тем, что, по сути, будут выполняться действия реальных пользователей. Суть подхода заключается в том, что объект тестирования (в данном случае – ОС) помещается в некоторую виртуальную среду, которая содержит необходимые инструменты для тестирования. Эти инструменты позволяют переводить систему в желаемые (тестируемые) состояния путем эмуляции действий конечного пользователя. В ходе тестирования имеется доступ ко всем переменным, описывающим состояние системы в виртуальной среде, при этом не вмешиваясь в ра-

боту самой системы (состояние регистров, состояние оперативной памяти, взаимодействие с сетевым адаптером, взаимодействие с жестким диском, взаимодействие с прочими внешними устройствами), причем будут получены реальные результаты этого взаимодействия, а не то, как это представлено в системе. К недостаткам данного метода можно отнести сложность детального тестирования сложных систем, так как физически невозможно протестировать все состояния сложной системы за приемлемое время воздействуя на нее как пользователь. Этот недостаток свойствен любому тестированию, рассматривающему систему как «черный ящик». Кроме того, существуют сложности получения некоторых специфичных для тестирования артефактов (при тестировании оконных компонентов системы могут быть сложности с получением дескрипторов окон и элементов окон, в связи с этим могут возникать сложности с выполнением и записью тестов). Таким образом, данный подход, несмотря на существующие преимущества, является достаточно зависимым от окружающей среды и сложен для применения в потоковом тестировании готового продукта или решения). Кроме того, средства виртуализации, в данном контексте, больше подходят для тестирования графических интерфейсов при необходимости неизменения состава пакетов объекта тестирования. Как было показано выше, проводить функциональное тестирование с использованием графических элементов, в общем случае, не имеет смысла, так как большинство функций успешно реализуется через вызов консольных команд. При этом графическая оболочка является надстройкой над самой операционной системой и реализует всего лишь отображение результатов работы тех или иных функций системы.

Наконец, рассмотрим подход с использованием протоколов удаленного управления.

Данный подход не требует проведения существенных изменений в составе пакетов тестируемой ОС и создания виртуальной среды. Протоколы удаленного управления позволяют воздействовать на тестируемую систему, как с использованием терминала (SSH), так и с помощью графического интерфейса (RDP). В данном подходе достаточно просто отправлять (получать) ответы системы и анализировать их. Однако требуется, чтобы в тестируемой системе присутствовала поддержка управления этой системой удаленно, с использованием выбранного протокола. По сравнению с хранением целой тестовой среды, это условие более щадящее к ресурсам тестируемой системы, но всё же требует некоторых изменений в ней. По аналогии с недостатками подхода непосредственного вызова функции, в подходе с использованием протоколов удаленного управления возникает дополнительный интерфейс взаимодействия с системой, в данном случает воздействие вообще приходит извне, что создает еще более сложную проблему для безопасности. Чтобы не позволить злоумышленнику воспользоваться этим интерфейсом, требуется расширить перечень функций устройства и добавить: «невозможность доступа неуполномоченного пользователя к интерфейсу удаленного управления системой». Более надежным решением будет отключение данного интерфейса сразу после завершения процедуры тестирования, но даже в этом случае нет полной защищенности от использования злоумышленником этого пути.

Резюмируя вышесказанное, можно отметить, что ни один из подходов не имеет абсолютного превосходства над другими.

Вызов функций непосредственно из тестируемой ОС требует больших временны́х и трудовых затрат на создание тестовой среды; требуется наибольшее количество дополнительных ресурсов от тестируемой ЭВМ, на которой развернут процесс выполнения тестов; просто происходит процесс выполнения тестов и анализа результатов; легко локализуются ошибки.

Использование средств виртуализации позволяет получить результаты работы системы в реальных условиях; несколько менее сложное создание тестовой среды; тестируемая система не подвергается изменениям, необходимым для воспроизведения тестов; сложность тестирования определённых состояний системы, имея в распоряжении только интерфейс пользователя; сложность анализа взаимодействия с графическим интерфейсом; контролируются все переменные, описывающие состояние системы и все взаимодействия системы со средой.

Использованием протоколов удаленного управления позволяет получить результаты работы системы в реальных условиях; еще менее сложное создание тестовой среды; требует небольших изменений в тестируемой системе; простота управления тестируемой системой; сложность тестирования определённых состояний системы; сложность анализа взаимодействия с графическим интерфейсом; возможность взаимодействия через терминал.

Резюмируя всё вышесказанное, можно отметить, что все способы проведения автоматизированного тестирования, описанные в статье, имеют свои преимущества и недостатки. В настоящее время не существует такого подхода, который можно было бы назвать оптимальным, поэтому при проведении работ по разработке автоматизированного тестирования, необходимо учитывать конкретные условия, сроки и нюансы разработки СЗИ.

СПИСОК ЛИТЕРАТУРЫ:

1. Конявская С.В., Счастный Д.Ю., Лыдин С.С., Маренникова Е.А., Грунтович Д.В., Чепанова Е.Г. Технологии защищенного применения USB-носителей: методическое пособие. (Серия «Учебная книга факультета «Кибернетика и информационная безопасность» НИЯУ МИФИ»). М.: НИЯУ МИФИ, 2014.
2. Конявская С.В., Счастный Д.Ю.,Кубеев Е.О., Ясиновская Е.Д. Технология доверенногосеанса связи (ДСС) и средства обеспечения доверенного сеанса связи (СОДС) «МАРШ!»: методическое пособие. (Серия «Учебная книга факультета «Кибернетика и информационная безопасность» НИЯУ МИФИ»). М.: НИЯУ МИФИ, 2015.

REFERENCES:

1. KonyavskayaS.V., Schastnyj D.YU., Lydin S.S., Marennikova E.A., Gruntovich D.V., Chepanova E.G. Tekhnologii zashchishchennogo primeneniya USB-nositelej: metodicheskoe posobie.: (Seriya «Uchebnayaknigafakul'teta «Kibernetikaiinformacionnayabezopasnost'» NIYAUMIFI»). M.: NRNU MEPHI, 2014.
2. Konyavskaya S.V., Schastnyj D.YU., KubeevE.O., Yasinovskaya E.D. Tekhnologiya doverennogo seansa svyazi (DSS) I sredstva obespecheniya doverennogo seansa svyazi (SODS) «MARSH!»: metodicheskoe posobie. Avt.: (Seriya «Uchebnaya kniga fakul'teta «Kibernetika i informacionnaya bezopasnost'» NIYAUMIFI»). M.: NRNU MEPHI, 2015.