необходимо студентам, обучающимся по специальностям, связанным с информационной безопасностью. Рассмотрение различных подходов к организации защиты от несанкционированной отправки позволяет обучающимся не только осознать проблему, но и выработать собственное видение эффективной защиты от краж закрытой информации.

СПИСОК ЛИТЕРАТУРЫ:

1. *Корт С. С.* Теоретические основы защиты информации. Гелиос АРВ, 2004.
2. *Cheswick W. R., Bellovin S. M., Rubin A. D.* Firewalls and Internet Security: Repelling the Wily Hacker. Addison-Wesley, 2003.
3. *Yeo L.* Personal Firewalls for Administrators and Remote Users. Prentice Hall PTR, 2003.

А. С. Николаев

ОСОБЕННОСТИ ОБЕСПЕЧЕНИЯ БЕЗОПАСНОСТИ ИНФОРМАЦИИ С ПОМОЩЬЮ МОНИТОРА ВИРТУАЛЬНЫХ МАШИН

Существует целый ряд случаев, когда развитие системного ПО отстает от развития аппаратных компонентов. Одним из таких случаев является разработка защищенного системного программного обеспечения. Связано это с тем, что при производстве защищенных программных систем необходимы дополнительные дорогостоящие и длительные процедуры, связанные с сертификацией. Это приводит к тому, что при выходе данного типа ПО на рынок оно оказывается устаревшим и его использование на базе новейшего аппаратного обеспечения является затруднительным. С течением времени требуется доработка программных компонентов и повторная сертификация, что в некоторых случаях представляется сложным и экономически неоправданным. Данная ситуация известна в литературе как проблема наследуемых операционных систем [1]. Для ее решения предлагается использовать механизмы, предоставляемые технологией виртуализации.

С помощью этой же технологии может быть решена и другая важная проблема — возможность использования недоверенных ОС, т. е. обеспечение инвариантности по отношению к гостевой ОС.

Помещая размещаемую (гостевую) ОС в некоторый контейнер (так называемую «песочницу», sandbox), технология виртуализации решает поставленные задачи следующими методами: предоставляет четко определенный унифицированный интерфейс взаимодействия с размещающей (хостовой) ОС; обеспечивает возможность контроля над всеми обращениями гостевой ОС к внешней среде (аппаратное обеспечение, локальные и сетевые программные ресурсы и т. д.).

Возможность контроля над действиями гостевой ОС основывается на том, что хостовая система предоставляет ограниченный инструментарий взаимодействия гостевой ОС с оборудованием, являясь своего рода промежуточным звеном взаимодействия между ними. С точки зрения хостовой операционной системы, каждая из гостевых ОС является обычным процессом, взаимодействующим при непосредственном участии гипервизора с внешним оборудованием. Поэтому для усиления безопасности обработки информации предлагается использовать механизмы политик безопасности, реализуемых в хостовой ОС и ограничивающих действия самих

контейнеров, в рамках которых работают наследуемые ОС. Данный подход позволяет бороться с нежелательными возможностями самих мониторов виртуальных машин как процессов (передача информации между локальными процессами посредством IPC, буфера обмена и т. д.). Данная концепция может быть реализована на базе платформы GNU/Linux средствами технологии SELinux путем использования политик принудительного управления доступом (MLS-политик), позволяющих контролировать потоки информации между виртуальными машинами в полном соответствии с моделью Белла—Лападулы [2].

Повышение безопасности обработки информации и надежности работы самого монитора виртуальных машин достигается за счет его минимизации [2]. В составе хостовой операционной системы предполагается наличие лишь следующих компонентов: ядро ОС, гипервизор (находящийся в составе ядра), минимальный набор системных библиотек и утилит, минимальная графическая система, позволяющая только отображать информацию от гостевой ОС, политика безопасности и инструментарий, необходимый для обеспечения работы в соответствии с данной политикой.

Следует отметить, что функции безопасности гипервизора не должны влиять на функции безопасности гостевой ОС. Иными словами, безопасность информации в гостевой и хостовой ОС должна быть ортогональной. Достигается это соблюдением следующих требований [3]:

1. изолирование виртуальных машин друг от друга; единственный механизм взаимодействия между ними — внешнее устройство (внешний сетевой интерфейс, USB и т. д.);

2. обычный пользователь должен иметь права только на запуск/останов виртуальной машины, а также подключение к ней через программу просмотра (viewer);

3. программа просмотра не должна иметь никакого доступа к данным и другим приложениям монитора виртуальных машин;

4. программы просмотра должны быть изолированы друг от друга, накопителей, сети и других устройств;

5. привилегированный пользователь должен получать доступ к администрированию лишь после аутентификации.

Применение такого подхода к реализации монитора виртуальных машин позволяет повысить безопасность обработки информации, а также значительно сократить расходы на его сертификацию.

Таким образом, использование минимизированного монитора виртуальных машин, контролируемого политикой безопасности, действующей в рамках хостовой операционной системы и ограничивающей взаимодействие контейнера с наследуемой ОС с внешним аппаратным обеспечением, позволяет: обеспечить безопасность информации, решить проблему наследуемых ОС, повысить безопасность обработки информации недоверенными ОС, сократить расходы на разработку и сертификацию всей программной системы (хостовой и гостевой).

СПИСОК ЛИТЕРАТУРЫ:

1. *Tanenbaum A.* Modern operating systems. 3rd edition. Pearson Education International, 2009. — 1072 с.
2. *Зегжда Д. П., Ивашко А. М.* Основы безопасности информационных систем. М.: Горячая линия — Телеком, 2000. — 452 с.
3. *Федосеев В. Н., Николаев А. С., Шанин О. И., Боршевников А. Н.* Использование и развитие дистрибутива Янукс 3.0 // Тезисы докладов 6-й Всероссийской научно-практической конференции «Методы и средства технической защиты информации». Обнинск, 19—21 мая 2009 г. Обнинск: НОУ «ЦИПК», 2009. С. 20—24.