УДК 004.94

ОБ ИНФОРМАЦИОННЫХ ПОТОКАХ ПО ВРЕМЕНИ В КОМПЬЮТЕРНЫХ СИСТЕМАХ[1]

М. А. Качанов

Случаи возникновения информационных потоков по времени, описанные в работе [1], охватывают множество возможностей их реализации в реальных компьютерных системах (КС), но не всегда полно отражают действительность. Ниже приводятся примеры новых видов информационных потоков по времени в таких КС, как операционная система (ОС) $GNU/Linux$ и система управления базами данных (СУБД) $MySQL$.

Для реализации информационного потока по времени в ОС $GNU/Linux$ используется виртуальная файловая система *proc*. Особенностью *proc* является то, что информация о действиях одного процесса может отображаться в файлах, доступных для чтения процессам, запущенным от имени других пользователей. Например, пусть имеются два процесса (P_1 и P_2) с идентификаторами $pid1$ и $pid2$ соответственно и пусть процесс P_2 имеет право чтения файла $/proc/pid1/status$. В этом файле, в частности, отображается количество нитей ($threads$), которыми оперирует процесс P_1. При создании либо удалении процессом P_1 нити информация об этом будет заноситься ядром ОС $GNU/Linux$ в файл $/proc/pid1/status$. Читая данный файл, процесс P_2 может получить данные от процесса P_1. На первый взгляд может показаться, что данный способ реализации информационного потока по времени подпадает под уже описанные в рамках ДП-моделей случаи, но это не так. Существенной особенностью приведенного выше примера является то, что при создании нити процесс P_1 не осуществляет никаких обращений к файловой системе, а данные в файл $/proc/pid1/status$ записывает ядро ОС. Таким образом, P_1 может вообще не иметь никаких прав доступа в файловой системе, но тем не менее информационный поток по времени может быть реализован. Стоит уточнить, что реализация *proc* в ОС $GNU/Linux$ такова, что пользователь, от имени которого запущен P_1, хоть и является владельцем файла $/proc/pid1/status$, но тем не менее не может менять права доступа к нему и не может открыть этот файл на запись. Для предотвращения возможности реализации подобного информационного потока по времени может быть использовано средство $SELinux$, позволяющее наложить дополнительные ограничения на стандартную политику безопасности $GNU/Linux$ и запретить чтение файла $/proc/pid1/status$ всем процессам, кроме P_1.

В случае СУБД $MySQL$ аналогичная ситуация возникает, когда некоторый пользователь осуществляет запросы к базе данных (БД). Ядро СУБД ведет статистику о количестве и типах запросов, об объеме принятых и переданных данных и некоторых других параметрах. Например, при всяком запросе пользователя `show databases;` ядро СУБД будет увеличивать текущее значение счетчика подобных запросов на единицу. Стоит отметить, что даже пользователь с минимальными правами в БД может тем или иным образом влиять на параметры, статистику о которых ведет ядро СУБД. С помощью запроса `show status;` пользователи системы могут получить полный отчет о накопленной статистике и увидеть текущие значения параметров, в том числе количество определенных запросов всех пользователей системы. Таким образом, один пользователь БД может передать данные другому пользователю, лишь совершая запросы к БД, разрешенные ему политикой безопасности, причем второй пользователь

[1]Работа выполнена в рамках реализации ФЦП «Научные и научно-педагогические кадры инновационной России» на 2009–2013 гг. (гос. контракт № П1010).

может не иметь никаких прав доступа к таблицам БД, с которыми работает первый пользователь.

Оба приведенных примера объединяет то, что информационные потоки по времени, возникающие в результате осуществления описанных действий, реализуются за счет отображения ядром системы информации о ее функционировании в сущностях, к которым субъекты системы непосредственно не получали доступ. Ядро системы само заносит данные о действиях субъектов в доступные на чтение другим субъектам сущности, причем первые могут и не иметь никаких прав доступа к данным сущностям.

В связи с обнаружением информационных потоков по времени нового типа возникает необходимость учета данных потоков при анализе защищенности КС. В рамках семейства ДП-моделей возможно введение нового вида ассоциированных сущностей, указывающих на возможность реализации к ним информационных потоков по времени в зависимости от выполняемых субъектом действий. Кроме того, возможно введение новых правил преобразований, а также формулировка и обоснование необходимых и достаточных условий возможности реализации информационных потоков по времени между сущностями КС.

ЛИТЕРАТУРА

1. *Девянин П. Н.* Анализ безопасности управления доступом и информационными потоками в компьютерных системах. М.: Радио и связь, 2006.

УДК 681.322

ОБУЧЕНИЕ НА ПЛАТФОРМЕ CISCO ОСНОВАМ ПОСТРОЕНИЯ ЗАЩИЩЕННЫХ ВЫЧИСЛИТЕЛЬНЫХ СЕТЕЙ[1]

Д. Н. Колегов

Основным известным подходом к созданию многоуровневой защиты сетевых компьютерных систем является архитектура Cisco SAFE [1]. В ней рассматриваются принципы и механизмы повышения безопасности сетевой инфраструктуры, предлагаются типовые схемы сетей, маршрутизации и коммутации в них, а также приводятся рекомендации по проектированию и настройке сетевых технологий защиты информации. Методы и подходы, изложенные в руководстве Cisco SAFE, в принципе, не зависят от производителя конкретных средств защиты и могут быть применены в сетях, построенных на основе технологий различных производителей, таких, как Cisco Systems, Check Point, Juniper, IBM ISS, Microsoft, H3C, HP, D-Link и др.

Одним из требований ФГОС ВПО третьего поколения в области информационной безопасности является наличие дисциплины «Основы построения защищенных вычислительных сетей». На кафедре защиты информации и криптографии Томского государственного университета данный курс читается автором на протяжении двух семестров и состоит из двух частей — теоретической (лекционной) и практической (лабораторной). В теоретической части, излагаемой на основе [1, 2], рассматриваются основные принципы и методы проектирования защищенных сетей, а в практической — изученные принципы и методы применяются студентами в лабораторных работах, проводимых в среде эмуляции Cisco Packet Tracer [3].

В теоретической части курса рассматриваются следующие основные вопросы:

[1]Работа выполнена в рамках реализации ФЦП «Научные и научно-педагогические кадры инновационной России» на 2009–2013 гг. (гос. контракт № П1010).