УДК 002.56+621.382

В.Г. СКОБЕЛЕВ

Институт прикладной математики НАН Украины, Донецк, Украина

БЕЗОПАСНОСТЬ IT-СИСТЕМ (ОБЗОР)

В работе дан краткий обзор моделей и методов, предназначенных для обеспечения информационной и функциональной безопасности современных IT-систем с критической областью применения. Рассмотрены основные типы существующих серверов безопасности IT-систем, основные цели политики безопасности IT-систем и методы ее формирования и анализа. Охарактеризованы основные типы непосредственного тестирования IT-систем в процессе их разработки, существующие модели управления доступом к информационным ресурсам и схема контроля доступа к ресурсам IT-системы, основанная на использовании цифровых удостоверений. Рассмотрены методы обеспечения программной безопасности IT-систем при действиях непредумышленных дестабилизирующих факторов.

Ключевые слова: *IT-системы, информационная и функциональная безопасность.*

Введение

В настоящее время существует широкий спектр угроз безопасности IT-систем. Эти угрозы могут быть классифицированы:

1) по природе происхождения (предумышленные и непредумышленные);

2) по направлению осуществления (внешние и внутренние);

3) по объекту воздействия (АРМы пользователей и администраторов, средства документирования и отображения, каналы связи и т.д.);

4) по способу осуществления (информационные, программно-аппаратные, физические, радиоэлектронные, организационно-правовые и т.д.);

5) по жизненному циклу (разработка, ввод в эксплуатацию, эксплуатация, вывод из эксплуатации).

Осознание того, что развитие информационных технологий сопровождается расширением спектра источников угроз безопасности IT-систем, изменением методов и средств инициализации и реализации этих угроз, стимулировало формирование в развитых странах «индустрии безопасности». Действие коллективов, работающих в этой сфере, регламентируется государством. Однако их результатами и продуктами рынка безопасности могут воспользоваться различного рода нарушители, включая преступные группировки, террористические организации и разведки иностранных государств.

Таким образом, на современном этапе развития информационных технологий обеспечение безопасности IT-систем является одной из наиболее актуальных комплексных проблем [1-5], от успешного решения которой зависит не только благополучие отдельных личностей и организаций, но и существование промышленных объектов и стран.

Суть решения этой проблемы состоит в обеспечении конфиденциальности, целостности и доступности информации для тех и только тех объектов и субъектов, для которых эта информация предназначена на всем жизненном цикле IT-системы.

Необходимым условием эффективного использования IT-системы является обеспечение на всех этапах ее жизненного цикла *информационной* и *функциональной безопасности*.

Информационная безопасность IT-системы определяется защитой от негативных воздействий на информационные ресурсы, в том числе от предумышленных воздействий с целью незаконного использования или искажения информации и программ, которые предназначены для применения ограниченным кругом лиц [6].

Функциональная безопасность IT-системы определяется ее способностью противостоять проявлению дефектов программ, данных, аппаратуры и дестабилизирующих воздействий внешней среды, приводящих к потере ее работоспособности, авариям и катастрофам.

Ясно, что задача обеспечения информационной безопасности существенно отличается от задачи обеспечения функциональной безопасности.

Отметим также, что любая IT-система является социо-технической [7], причем отношения между субъектами часто недостаточно формализованы, из-за чего могут возникать конфликты.

Для решения задач обеспечения безопасности IT-систем разработаны формальные модели, определяющие условия, которым должно соответствовать поведение системы. Это дает возможность до-

казать, что при соблюдении установленных правил и ограничений IT-система соответствует принятому критерию безопасности.

Широкий спектр IT-систем, существенно различающихся по своему назначению и по своей архитектуре, определяет многообразие подходов к решению отдельных задач обеспечения безопасности IT-систем.

Цель настоящего обзора – кратко охарактеризовать некоторые из таких подходов.

1. Серверы безопасности

Известно, что одним из основных средств обеспечения безопасности IT-систем являются современные операционные системы (ОС). Они включают в себя комбинацию серверов идентификации, аутентификации, управления доступом, протоколирования, аудита и криптографической защиты. Кроме того, ОС обеспечивают такие базовые инфраструктурные свойства безопасности, как разделение доменов и посредничество при обращениях.

Системы управления базами данных (СУБД), также содержат комбинацию серверов безопасности. Однако, в отличие от ОС, они не являются самодостаточными, так как используют механизмы и функции ОС. Наличие этих двух уровней приводит к появлению специфических угроз и требует соответствующих средств противодействия им.

Для обеспечения безопасности современных IT-систем на стадии их эксплуатации предназначены также специальные аппаратно-программные подсистемы, которые принято называть серверами безопасности. В их функции входят идентификация и аутентификация, управление доступом, контроль целостности, протоколирование и аудит, шифрование, экранирование, туннелирование, контроль защищенности, отказоустойчивость и оперативное восстановление, а также управление. В силу различных причин не для всех них разработаны общие архитектурные и технологические требования. Рассмотрим кратко некоторые из них.

Серверы идентификации и аутентификации должны обеспечить защиту от активного и пассивного прослушивания сети, поддерживать концепцию единого входа в систему.

Сервверы управления доступом должны обеспечить разграничение доступа пользователей к ресурсам. Они предназначены для защиты от злоумышленных пользователей. На данный момент этот тип серверов безопасности наименее исследован.

Серверы контроля целостности должны обеспечить целостность информации на уровнях порций данных, программных и аппаратных компонент, распределенных конфигураций, а также защитить потоки данных от несанкционированной модификации. На данный момент этот тип серверов безопасности наиболее полно проработан.

Серверы протоколирования и аудита предназначены для выявления нетипичного поведения пользователей, программ, аппаратуры и начала злоумышленных действий против IT-системы. В настоящее время разработке структуры такого типа серверов уделяется значительное внимание, так как они, по своей сути, представляют собой последний рубеж обороны за безопасность IT-системы.

Серверы, осуществляющие шифрование, предназначены для обеспечения конфиденциальности информации. Эти серверы являются одними из наиболее проблемных серверов для современных IT-систем

Серверы, обеспечивающие экранирование, предназначены для разграничения межсетевого доступа посредством фильтрации и преобразования передаваемых данных на основе заданных правил, охватывающих сетевой, транспортный и прикладной уровни.

Серверы, обеспечивающие туннелирование, предназначены для осуществления перехода между сетями с различными протоколами, а также для обеспечения целостности и конфиденциальности передаваемой порции информации, включая служебные поля.

Отметим, что комбинация туннелирования и шифрования на выделенных шлюзах с экранированием на маршрутизаторах поставщиков сетевых услуг дает возможность реализовать виртуальные локальные сети, наложенные на Интернет. Такие сети дешевле и безопаснее, чем собственные сети, построенные на выделенных каналах. Концами туннелей, реализующих виртуальные сети, являются межсетевые экраны, обслуживающие подключение к внешним сетям.

Серверы анализа защищенности должны обнаруживать (раньше, чем злоумышленники) «бреши», появляющиеся в результате ошибок администрирования и в результате обновления версий программного обеспечения, а также накапливать информацию об атаках на IT-систему. В настоящее время существует достаточно много коммерческих и свободно распространяемых продуктов, предназначенных для анализа защищенности. Основная проблема состоит в организации постоянного обновления ее базы данных информацией о выявленных слабостях IT-системы.

Серверы, обеспечивающие отказоустойчивость и оперативное восстановление, предназначены для обеспечения высокой постоянной готовности IT-системы для выполнения своих функций. В настоя-

щее время сформировался широкий спектр средств данного класса, основанный на программном обеспечении промежуточного слоя и кластерных конфигурациях, способных обеспечить также высокую готовность других средств безопасности.

Серверы, предназначенные для управления, должны обеспечить контроль согласованности конфигураций различных компонент, при котором не нарушается защищенность IT-системы. В настоящее время на рынке безопасности только начинают появляться продукты, предназначенные для управления, которые приемлемы по цене, потребляемым ресурсам и обладают достаточной открытостью, расширяемостью и масштабируемостью.

Отметим, что для современных сложных распределенных IT-систем из-за разграничения администрирования может возникать «размывание зон ответственности». В результате могут образоваться «бреши» на стыках различных участков сети, которые являются источниками серьезной уязвимости IT-системы. Именно отсутствие в настоящее время общих архитектурных и технологических требований ко всему комплексу серверов безопасности во многом способствует актуальности проблемы безопасности современных IT-систем.

2. Политика безопасности

Под политикой безопасности IT-системы понимается совокупность норм и правил, регламентирующих процесс обработки информации, выполнение которых должно обеспечить состояние защищенности информации в заданном пространстве угроз.

Основная цель политики безопасности IT-системы состоит в разработке эффективной системы управления ее безопасностью на протяжении всего жизненного цикла. Эффективность такой системы безопасности оценивается на основе системного процесса, который на основе анализа объективных данных о текущем состоянии IT-системы, действий и событий, происходящих в ней, устанавливает уровень их соответствия определенным критериям.

Общие положения политики безопасности IT-системы должны, как минимум, обеспечивать:

1) идентификацию и аутентификацию всех субъектов доступа;

2) доступ к информационным ресурсам;

3) подотчетность пользователей, т.е. какие действия при работе с какими серверами должны быть подотчетны при применении объекта, описанного в профиле;

4) протоколирование и аудит функционирования серверов безопасности;

5) доступность коммуникационных каналов;

6) конфиденциальность и целостность управляющей информации, в том числе и при удаленном администрировании;

7) целостность аппаратно-программной и информационной частей системы;

8) невозможность обхода защитных средств.

Для реализации этих положений необходимо:

1) выявить специфику внешнего информационного обмена с позиции защиты информации;

2) определить взаимосвязи угроз и уязвимости IT-системы, делающие возможной их реализацию;

3) классифицировать угрозы в зависимости от значимости наносимого ими ущерба;

4) формализовать задачи защиты систем внешнего информационного обмена с использованием той или иной модели рисков;

5) осуществить анализ требований, предъявляемых к отдельным средствам защиты IT-системы.

Охарактеризуем кратко математические методы, применяемые при анализе и формировании политик безопасности IT-систем.

Для анализа безопасности функционирования IT-системы с критической областью применения необходимо исследовать ее реакции на все возможные входные данные, т.е. осуществить непосредственное тестирование.

Известно, что существует два основных метода сокращения количества исследуемых реакций:

1) доказательство того, что система всегда работает корректно;

2) демонстрация того, что система никогда не выполнит некорректных действий.

В 1-м методе на основе комбинации анализа и эмпирического тестирования определяются реакции, которые могут привести к серьезным сбоям (функционирование при граничных условиях, при условиях, не оговоренных в качестве возможных для компонентов системы и т.д.). Доказательство корректность функционирования сводится к доказательству того, что такие реакции невозможны.

Во 2-м методе предполагается, что система делает что-то некорректное и осуществляется анализ ее реакций с целью выявления состояний, в которых возможно проявление данной некорректности. Доказательство корректности функционирования сводится к доказательству того, что такие состояния недостижимы, т.е. к доказательству от противного.

Оба эти метода основаны на группировании "схожих" реакций (именно по этой причине достаточно рассмотреть только небольшое количество входных воздействий). Такой подход адекватно применим к непрерывным (в некоторой топологии) системам, т.е. к системам, в которых незначитель-

ные изменения входных данных влекут незначительные изменения выходных данных.

К сожалению, IT-системы с критической областью применения содержат, как правило, большое количество дискретных решений, принимаемых при исполнении программного обеспечения, и не являются непрерывными. Поэтому для таких IT-систем рассмотренные методы дают возможность только выявить примитивные ошибки в программном обеспечении, но не могут быть применены для адекватно анализа функционирования, из-за наличия большого количества реакций системы, которые необходимо исследовать.

В отдельных случаях при правильной организации непосредственного тестирования IT-системы с критической областью применения (оно будет рассмотрено в следующем разделе) могут быть получены более или менее приемлемые вероятностные оценки безопасности ее функционирования. Основная сложность при этом состоит в адекватном определении вероятностей исследуемых ситуаций.

Указанные обстоятельства стимулировали разработку формальных методов анализа функционирования IT-систем с критической областью применения, основанных на моделях дискретной математики (множество, граф, частичный порядок, машина конечных состояний, и т.д.). При этом вычисления основываются на методах формальной логики.

Практическое значение применения формальных методов заключается именно в возможности теоретического анализа всех реакций IT-системы. Это обусловлено тем, что при их применении мы имеем дело с внутренней реализацией системы. Поэтому, возникает возможность декомпозиции реакций системы на реакции ее компонент (с помощью анализа формальных спецификаций компонент) с последующей их композицией (что и дает возможность описания всех реакций системы).

Для автоматизации процесса доказательства свойств IT-систем с критической областью применения используются "доказатели теорем", т.е. программы, осуществляющие формальную дедукцию на основе комбинации эвристик и поиска.

Другим видом программного обеспечения, используемого при формальном анализе IT-систем с критической областью применения, являются "контроллеры доказательств", т.е. программы, предоставляющие пользователю возможность проводить последовательность шагов в процессе логических выводов о свойствах системы с проверкой корректности каждого шага.

Ясно, что наиболее эффективны средства, сочетающие в себе свойства двух приведенных выше видов программного обеспечения.

Рассмотренные принципы применения формальных методов анализа IT-систем имеют недостаток, присущий всем методам моделирования: формальные методы имеют дело не с системой, а с ее моделью (а, как известно, модель может не отражать реальность или отражать ее некорректно). Кроме того, излишняя подробность описания IT-системы ведет к резкому росту временных затрат на проведение формального анализа, что может привести к неэффективности его применения. Поэтому задача корректного выбора уровня абстракции (т.е. множества требований к системе, которые должны найти свое отражение в модели) является определяющей при построении модели безопасности IT-системы.

3. Непосредственное тестирование

Каждый вид тестирования IT-системы ориентирован на выявление определенного класса дефектов. Поэтому необходимо упорядоченное проведение различных видов тестирования на основе разработанных методик их выполнения с указанием контролируемых параметров и эталонных результатов. При заключительных испытаниях или сертификации должно также проводиться интегральное тестирование при максимально широком варьировании тестов в условиях, соответствующих нормальной эксплуатации [8-10].

Рассмотрим кратко основные виды тестирования при разработке IT-системы с критической областью применения.

1. Тестирование полноты решения задач при типовых исходных данных. Предназначено для обнаружения ошибок функционирования в типовых условиях, определенных техническим заданием (ТЗ). Эталоном являются цели и задачи создания IT-системы. В соответствие с ними создается формализованное ТЗ и спецификация требований на программное обеспечение. Для систем реального времени в тестах используются, в основном, данные, имитируемые моделями объектов внешней среды. Результаты тестирования обрабатываются и сравниваются с эталонами, как правило, автоматически.

2. Тестирование функционирования программ в критических ситуациях. Предназначено для анализа исполнения программ в нештатных ситуациях, при которых необходимо обеспечить безопасное функционирование IT-системы. Для разработки таких тестов создаются (как правило, вручную) сценарии критических сочетаний значений исходных данных и условий решения задач. При тестировании применяются имитаторы внешней среды, автоматически подготавливающие исходные данные, а также средства контроля, реагирующие на аномальные резуль-

таты исполнения тестируемых программ, отражающиеся на безопасности IT-системы.

3. Тестирование ресурсов. Предназначено для оценки безопасности исполнения программ при перегрузках памяти и производительности. Осуществляется в реальном времени по сценариям, создающим перегрузки анализируемого ресурса. Проверяется изменение качества, надежности и безопасности функционирования IT-системы из-за пропусков в обработке сообщений, роста промежутка ожидания и увеличения времени решения задач [11]. В результате тестирования определяются реальные характеристики пропускной способности всей системы, а также допустимая интенсивность решения отдельных типов задач и обработки сообщений. При кратковременных перегрузках должна обеспечиваться защита от отказов и восстановление нормального режима функционирования при последующем снижении загрузки.

4. Тестирование параллельного исполнения программ. Предназначено для обнаружения снижения надежности и безопасности из-за несогласованного использования исходных и промежуточных данных, а также устройств при параллельном исполнении программ [9, 12]. Основная задача состоит в том, чтобы обнаружить все тупиковые ситуации. Данные подготавливаются, в основном, автоматически по сценариям наиболее критических их сочетаний.

5. Тестирование эффективности защиты от искажений исходных данных. Предназначено для выявления в программах ошибок, проявляющихся при ложных или искаженных данных. Проводится как при относительно небольших искажениях исходных данных (соответствующих нормированному возрастанию в них ошибок), так и при случайном их искажении. В результате тестирования выявляются ситуации, в которых происходит нарушение работоспособности IT-системы, а также оценивается снижение безопасности ее функционирования в зависимости от интенсивности искажений.

6. Тестирование эффективности защиты от сбоев аппаратуры и не выявленных ошибок программ и данных. Предназначено для проверки качества средств программного контроля и оперативного восстановления (рестарта) при не преднамеренных искажениях функционирования IT-системы. Для разработки таких тестов создаются (как правило, вручную) сценарии имитации соответствующих ситуаций. В результате тестирования для каждого вида сбоя определяются показатели надежности функционирования программ, оценивается вероятность обнаружения отказовой ситуации и средняя продолжительность восстановления.

7. Тестирование надежности и безопасности. Предназначено для определения основных показателей надежности и безопасности при реальном функционировании IT-системы. В процессе тестирования при типовых и критических условиях определяются значения наработки на отказ, оценки длительности восстановления, коэффициента готовности и т.д. Для разработки таких тестов создаются (как правило, вручную) сценарии многочасовых прогонов IT-системы при различных соотношениях (определяемых в соответствии с ТЗ) типовых и критических условий функционирования и исходных данных. Имитация исходных данных и регистрация отказов, как правило, осуществляется автоматически. Особое внимание уделяется регистрации условий нарушения работоспособности IT-системы.

8. Тестирование взаимодействия с пользователем. Предназначено для обнаружения плохо формализуемых ошибок интерфейса, а также для анализа его удобства для пользователя. В результате тестирования выявляются ошибки распределения автоматизируемых функций между пользователем и системой, а также оценивается качество принятия решений пользователем в динамике функционирования IT-системы, особенно в критических ситуациях.

9. Тестирование удобства и качества подготовки пользовательских версий. Предназначено для выявления ошибок методов и средств настройки базовых версий для конкретных условий их применения. В качестве данных используются наиболее типичные (в соответствии с ТЗ) режимы применения IT-системы.

10. Тестирование при различных конфигурациях оборудования. Предназначено для обнаружения ошибок, проявляющихся при изменении состава или характеристик компонент вычислительной системы или внешней среды. В результате тестирования определяются допустимые комплектации оборудования и средства автоматизированной адаптации к ним IT-системы.

Подчеркнем, что технологическая безопасность функционирования IT-систем при непредумышленных угрозах поддерживается многими стандартами, обеспечивающими, в той или иной, мере технологию разработки, жизненный цикл, сопровождение, испытания, сертификацию, а также унификацию интерфейсов с операционной и внешней средой. Ситуация является значительно более сложной с обеспечением безопасности функционирования IT-систем при предумышленных нарушениях.

В заключение подчеркнем, что с позиции поведения IT-системы в условиях критической нагрузки их можно разделить на следующие две группы:

1) IT-системы, у которых наблюдается плавный спад производительности при превышении максимального уровня нагрузки;

2) IT-системы, у которых производительность резко падает при незначительном превышении максимального уровня нагрузки.

Одна из основных задач непосредственного тестирования и состоит в идентификации исследуемой IT-системы в соответствии с этой классификацией.

4. Модели управления доступом

В настоящее время основой защиты информации в IT-системе являются механизмы разграничения доступа к информационным ресурсам. Поэтому в отдельный класс выделены формальные модели управления доступом, предназначенные для определения формальных правил, регламентирующих схему информационных потоков (по времени и по памяти), а также управление ими.

Теоретические основы построения формальных моделей управления доступом были заложены в 60-70 годах XX века [13-16]. Однако они практически не оказали никакого влияния на разрабатываемые в то время ОС, так как проектировщики снабжали разрабатываемые ОС своими собственными средствами обеспечения безопасности, созданными независимо от теоретических исследований. Именно опыт эксплуатации разработанных в то время ОС привел к ясному пониманию того, что встроенные средства обеспечения информационной безопасности образуют «ядро», т.е. абстрактное программное устройство, размещенное на как можно более низком уровне и предназначенное для управления файлами и памятью. Концепция ядра стимулировала разработку международных стандартов, содержащие критерии оценки безопасности информационных технологий.

Однако следует отметить, что до сих пор отсутствует глубокая теоретическая проработка концепции ядра, образованного встроенными средствами обеспечения информационной безопасности. Из-за этого наблюдается тенденция «разбухания ядра» при переходе к новым модификациям ОС.

В современных IT-системах основными являются дискреционные, мандатные и ролевые модели управления доступом [6,17-20]. Общим для этих моделей является то, что сущности разделяются на множества субъектов и объектов. Совокупность множеств субъектов, объектов и отношений между ними и определяет состояние системы. Каждое состояние (в соответствии с используемым критерием) является либо безопасным, либо небезопасным. Подчеркнем, что для каждой конкретной модели осуществляется формальное доказательство ее безопасности.

Кроме того, начиная с 90-х годов XX века, развиваются модели семейства TBAC (Task Based Authorization Control), предназначенные для обеспечения безопасности информационных процессов IT-систем, реализующих workflow. В этих моделях основным понятием является задача, а полномочия изменяются в соответствии с контекстом задачи. Такой подход дает возможность минимизировать привилегии и, тем самым, сократить возможности несанкционированного доступа к данным и неправильного их использования. Для верификации таких моделей управления доступом, в основном, используются сети Петри [21] и средства моделирования сложных эволюционирующих систем нейронными сетями, представленные языком схем радикалов.

5. Контроль доступа к ресурсам

Актуальность обеспечения эффективного контроля доступа к ресурсам IT-систем была осознана еще в 60-70 годах XX века. Эта проблема обусловлена тем, что в процессе проектирования IT-систем основные усилия уделяются технической стороне проекта, а жизненный цикл процессов, лежащих в его основе может быть не полностью понят проектировщиками.

Действенным средством контроля доступа к ресурсам IT-системы с критической областью применения является подсистема управления цифровыми удостоверениями пользователей (identity management). Такая подсистема реализует (в соответствии с требованиями законодательства и политики компании) весь комплекс процессов, предназначенных для обеспечения защищенного доступа конечных пользователей к широкому кругу внутренних и внешних IT-систем, для контроля над цифровыми удостоверениями этих пользователей, а также для управления информацией об этих удостоверениях.

В общем случае цифровые удостоверения представляют собой электронные записи, обозначающие участников сети, в том числе людей, машины, устройства, приложения и службы. Хотя эти удостоверения разбросаны по всей текущей инфраструктуре IT-системы, для обеспечения ее безопасности они должны быть синхронизированы между различными системами и приложениями.

На практике одной из наиболее важных проблем является наличие учетных записей «призраков». Ее причина состоит в разрыве между процессом управления уходом пользователей с должности и процессом управления удостоверениями пользователей. Эффективная система управления цифровыми удостоверениями должна оперативно решать эти задачи, снимая эту нагрузку с системных администраторов. В результате снижаются затраты и повышается безопасность IT-системы.

Жизненный цикл цифрового удостоверения включает в себя инициализацию, сопровождение и отзыв учетной записи.

Инициализация учетной записи представляет собой действие по предоставлению пользователям соответствующего уровня доступа к ресурсам, необходимым для выполнения их должностных обязанностей.

Сопровождение учетной записи состоит в поддержке информации цифрового удостоверения в актуальном состоянии, обеспечении соответствующих уровней доступа пользователя к ресурсам, необходимым для его эффективной работы.

Отзыв учетной записи означает ее своевременное отключение при уходе пользователя с должности.

Особенности процесса проектирования системы эффективного управления жизненным циклом цифровых удостоверений состоят в следующем.

Для обеспечения соответствия изменения удостоверений пользователей в реальной жизни и изменения их цифровых удостоверений должен быть реализован вспомогательный процесс, запускающим событием которого является прием на работу нового сотрудника, уход с должности, расширение либо сужение полномочий, продвижение по службе, увольнение и т.д. Схема такого процесса состоит в следующем.

Ответственность за выполнение запроса на изменение удостоверения должны нести сотрудники отдела кадров или соответствующие менеджеры. Для авторизации эти изменения должны быть утверждены вышестоящим начальством, владельцами приложения или руководством отдела кадров. После этого группа поддержки вносит в соответствии с запросом изменения в IT-систему. Таким образом, любые изменения удостоверений будут своевременно введены и отражены в цифровых удостоверениях, хранящихся в IT-системе.

В заключение отметим, что для разработки эффективной автоматизированной системы управления удостоверениями для конкретной IT-системы существенную помощь могут оказать предлагаемые на рынке решения по таким вспомогательным процессами, как инициализация, отзыв, авторизация, проверка подлинности учетных записей пользователей, управление паролями, аудит, самообслуживание пользователей, централизованное и делегированное администрирование и т.д. Основная задача проектировщиков автоматизированной системы управления удостоверениями для IT-системы состоит именно в анализе процессов управления удостоверениями и выборе наиболее подходящего решения, обеспечивающего эффективную интеграцию этих процессов.

6. Программная безопасность IT-систем

На реальные IT-системы кроме предумышленных дестабилизирующих факторов действуют также и непредумышленные дестабилизирующие факторы. Причем последние способны вызвать аномалии функционирования и катастрофические последствия значительно более тяжелые, чем злоумышленные действия.

Непредумышленные дестабилизирующие факторы имеют свою природу, особенности и требуют самостоятельного анализа и разработки адекватных методов и средств защиты.

Рассмотрим проблему обеспечения безопасности программных систем (ПС) и баз данных (БД) при действиях непредумышленных дестабилизирующих факторов.

Известно, что в сложных ПС и БД невозможно обеспечить полное отсутствие дефектов проектирования и реализации [22]. Поэтому часто исходят из того, что основная непредумышленная угроза обусловлена наличием именно этих внутренних дестабилизирующих факторов и внешних дестабилизирующих факторов, обусловленных средой функционирования IT-системы.

Наиболее полно безопасность IT-системы характеризует величина ущерба при проявлении дестабилизирующих факторов и среднее время между проявлениями угроз, нарушающих ее безопасность. Однако формально описать и измерить возможный ущерб при нарушении безопасности IT-системы с критической областью применения практически невозможно. Поэтому реализации угроз характеризуют интервалами времени между их проявлениями, отражающиеся на безопасности.

Эти показатели аналогичны показателям надежности IT-системы. Различие состоит в том, что в показателях надежности учитываются все реализации отказов, а в характеристике безопасности учитываются только те отказы, которые отражаются на ней [23-25].

В предположении о безотказности аппаратуры первопричиной нарушения работоспособности программ является конфликт между реальными исходными данными и программой, осуществляющей их обработку. Работоспособность программы можно гарантировать только при тех исходных данных, которые использовались при отладке и испытаниях. Реальные данные могут принимать значения, отличающиеся от них и от значений, заданных техническим заданием.

Из-за этого функционирование программ трудно предсказать заранее, и возможны аномалии, завершающиеся отказами.

При отсутствии физического разрушения аппаратуры для классификации сбоев и отказов программ используется показатель длительности восстановления после искажения программ, данных или вычислительного процесса [10]. Если эта длительность не превосходит заданный порог, то аномалию функционирования ПС относят к сбоям, в противном случае – к отказам. Некоторые отказы не влияют на безопасность применения ПС и устраняют достаточно быстро. Проявление других видов отказов и длительность восстановления после них могут быть катастрофическими. Такие отказы и квалифицируются как нарушение безопасности функционирования ПС. Именно на их основе при испытании и на завершающих фазах комплексной отладки ПС рассчитываются такие интегральные показатели качества программ, как коэффициенты устойчивости, восстанавливаемости, готовности и т.д. Подчеркнем, что эти показатели мало применимы в процессе реальной эксплуатации IT-системы.

Для обеспечения гибкости модификации и безопасности ПС и БД при развитии IT-системы выработаны общие принципы и правила, направленные на унификацию структуры и взаимодействия компонент в пределах проблемной области. Хотя они могут иметь свои особенности для различных проблемно-ориентированных областей, их выполнение существенно влияет на снижение трудоемкости, длительности, стоимости разработки ПС, БД и их версий, а также на повышение их безопасности.

Уровень и влияние дестабилизирующих факторов на безопасность применения IT-системы определяется в значительной степени качеством технологии проектирования, разработки, сопровождения и документирования ПС и БД. Из-за ограниченных ресурсов для достижения заданных требований по безопасности возникает необходимость обеспечения управление качеством в течение всего цикла создания ПС и БД. Такое управление поддерживается методиками, типовыми документами и средствами автоматизации разработки. Отметим, что при создании современных IT-систем с критической областью применения управление качеством ПС и БД в значительной мере обеспечивается применением CASE-технологий, языков 4-го поколения и стандартов открытых систем. Однако при этом следует учитывать, что массовый перенос программ и данных на различные аппаратные и операционные платформы способствует распространению дефектов и невыявленных ошибок, остающихся в переносимых компонентах.

Для повышения безопасности функционирования ПС и БД на этапе эксплуатации применяются методы оперативного обнаружения дефектов при исполнении программ и искажений данных, основанные на использовании временной, информационной и программной избыточности. Такие методы предполагают наличие вычислительных ресурсов, предназначенных для быстрого обнаружения проявления дефектов, классификации типа уже имеющихся и возможных последствий искажений, а также для автоматизированных мероприятий, обеспечивающих быстрое восстановление нормального функционирования IT-системы.

Наряду с оперативной реакцией необходимо осуществлять автоматизированное накопление и статистическую обработку информации о проявлениях дефектов для того, чтобы использовать эти данные для локализации первичного источника ошибок и исправления соответствующих программ, данных или компонент аппаратуры.

При проектировании IT-систем с критической областью применения особое внимание требует использование зарубежных ПС и БД [26]. Хотя «вирусы» и «закладки» маловероятны в серийных широко тиражируемых ПС и БД, необходимо применять специальные методы и средства их целенаправленного обнаружения и устранения. Кроме того, в них возможны те же самые дефекты проектирования и реализации, как и рассмотренные выше.

В заключение отметим, что обеспечения безопасности IT-систем на практике должны в равной степени учитываться как непредумышленные, так и предумышленные дестабилизирующие факторы.

Заключение

В работе кратко рассмотрены подходы, предназначенные для решения некоторых задач, связанных со сложной многогранной проблемой обеспечения безопасности IT-систем. За рамками осталось много актуальных задач, которые в настоящее время находятся только на стадии исследования, и анализ ситуации с каждой из которых является темой для соответствующего аналитического обзора.

К ним, в частности, относятся следующие задачи:

1) анализ существующих стандартов обеспечения безопасности IT-систем;

2) выбор надежной аппаратной архитектуры IT-системы, а также ее сетевой структуры;

3) анализ возможностей применения достижений технологии программирования (включая объектно-ориентированный подход) при разработке программных средств защиты IT-систем;

4) анализ принципов криптографической защиты информации;

5) анализ надежности технологий межсетевого обмена данными;

6) анализ надежности технологий обнаружения вторжений;

7) выбор структуры управления средствами сетевой безопасности;

8) анализ кибер-атак на IT-системы с критической областью применения;

9) анализ кибер-преступлений против безопасности персональных данных.

Рассмотренные в работе и перечисленные выше задачи показывают, что основным направлением обеспечения безопасности современных IT-систем является исследование подходов и разработка методов организации четкого эффективного управления и четкого эффективного контроля использования ресурсов.

В заключение подчеркнем, что применение отдельных локальных систем обеспечения безопасности IT-систем является недостаточно эффективным из-за возможных противоречий между их решениями. Координация действий таких локальных систем безопасности и устранение противоречий в их решениях является одной из основных функций специализированной службы обеспечения комплексной безопасности организации.

В настоящее время проблеме разработки эффективной и надежной структуры таких специализированных служб только начинает уделяться должное внимание.

Литература

1. Столлингс, В. Криптография и защита сетей: принципы и практика [Текст] / В. Столлингс. – М.: Вильямс, 2001. – 672 с.

2. Казарин, О.В. Безопасность программного обеспечения компьютерных систем [Текст] / О.В. Казарин. – М: МГУЛ, 2003. – 212 с.

3. Гайдамакин, Н.А. Теоретические основы компьютерной безопасности [Текст] / Н.А. Гайдамакин. – Екатеринбург: Изд-во Уральского университета, 2008. – 212 с.

4. Шаньгин, В.Ф. Информационная безопасность компьютерных систем и сетей [Текст] / В.Ф. Шаньгин. – М.: ИД «ФОРУМ», 2011. – 416 с.

5. Федотов, А.М. Информационная безопасность в корпоративной сети [Текст] / А.М. Федотов // Проблемы безопасности и чрезвычайных ситуаций. – М.: ВИНИТИ, 2008. – № 2. – С. 88-101.

6. Гайдамакин, Н.А. Разграничение доступа к информации в компьютерных системах [Текст] / Н.А. Гайдамакин. – Екатеринбург: Изд-во Уральского университета, 2003. – 328 с.

7. Волобуев, С.В. Философия безопасности социотехнических систем: информационные аспекты [Текст] / С.В. Волобуев. – М.: Вузовская книга, 2004. – 360 с.

8. Musa, J.D. Software Reliability: Measurement, Prediction, Application [Текст] / J.D. Musa, A. Iannino, K. Okumoto.– N.Y.: McGraw Hill, 1987. – 621 p.

9. Howden, W.E. Functional program testing and analysis [Текст] / W.E. Howden, – N.Y.: McGraw Hill, 1987. – 175 p.

10. Липаев, В.В. Отладка сложных программ [Текст] / В.В. Липаев. – М.: Энергоатомиздат, 1993. – 384 с.

11. Липаев, В.В. Распределение ресурсов в вычислительных системах [Текст] / В.В. Липаев. – М.: Статистика, 1979. – 247 с.

12. Майерс, Г. Искусство тестирования программ / Г. Майерс. – М.: Финансы и статистика, 1982. – 176 с.

13. Harrison, M. Protection in operating systems [Текст] / M. Harrison, W. Ruzzo, J. Ullman // Communication of ACM. 1976. – N 8. – P. 461-471.

14. Bell, D.E. Secure computer systems: unified expositional multics interpretation [Текст] / D.E. Bell, L.J. LaPadula // MTR-2797 Rev 1. – Bedford, Mass.: MITRE Corp., 1976.

15. Lanawehrn, E. A security model for military message system [Текст] / E.A. Lanawehrn // ACM Transactions on Computer Systems. – 1984. – N 3. – P. 198-222.

16. Bishop, M. Computer security: art and science [Текст] / M. Bishop. – NY: Addison Wesley Publishing Company, 2002. – 1136 p.

17. Зегжда, Д.П. Основы безопасности информационных систем [Текст] / Д.П. Зегжда, А.М. Ивашко. – М.: Горячая линия-Телеком, 2000. – 452 с.

18. Девянин, П.Н. Модели безопасности компьютерных систем [Текст] / П.Н. Девянин. – М.: Издательский центр «Академия», 2005. – 144 с.

19. Девянин, П.Н. Обзорные лекции по моделям безопасности компьютерных систем [Текст] / П.Н. Девянин // Прикладная дискретная математика. – 2009. – № 2. – С. 151-190.

20. Щербаков, А.Ю. Современная компьютерная безопасность. Теоретические основы. Практические аспекты [Текст] / А.Ю. Щербаков. – М.: Книжный мир, 2009. – 352 с.

21. Knorr, K. Dynamic access control through Petri net workflows[Текст] / K. Knorr // Proceedings of the 16th Annual Computer Security Applications Conference, New Orleans, LA, December 2000. – P. 159-167.

22. Castano, S. Database security [Текст] / S. Castano, M.G. Fugini, G. Martello, et al. – N.Y.: Addison Wesley Publishing Company, 1995. – 456 p.

23. Боэм, Б. Характеристики качества программного обеспечения [Текст] / Б. Боэм, Х. Каспар. – М.: Мир, 1981. – 208 с.

24. Боэм, Б. Инженерное проектирование программного обеспечения [Текст] / Б. Боэм. – М.: Радио и связь, 1985. – 512 с.

25. Waters, A. Software quality assurance. Vol. II [Текст] / A. Waters, J. Vincent, J. Sinclair. – N. Y.: Prentice-Hall, 1988. – 167 с.

26. Юсупов, Р.М. Безопасность компьютерной инфосферы систем критических приложений [Текст] / Р.М. Юсупов, Б.П. Пальчун // Вооружение. Политика. Конверсия. – 1993. – № 2. – С. 52 – 56.

Поступила в редакцию 14.02.2013, рассмотрена на редколлегии 13.03.2013

Рецензент: д-р техн. наук, проф., зав. каф. компьютерных систем и сетей В.С. Харченко, Национальный аэрокосмический университет им. Н.Е. Жуковского «ХАИ», Харьков, Украина.

БЕЗПЕКА IT-СИСТЕМ (ОГЛЯД)

В.Г. Скобелєв

В роботі дано короткий огляд моделей та методів, які призначено для забезпечення інформаційної та функціональної безпеки сучасних IT-систем з критичною областю застосування. Розглянуто основні типи існуючих серверів безпеки IT-систем, основні цілі політики безпеки IT-систем і методи її формування та аналізу. Охарактеризовано основні типи безпосереднього тестування IT-систем у процесі їх розробки, існуючі моделі керування доступу до інформаційних ресурсів та схема контролю доступу до ресурсів IT-системи, яка заснована на використанні цифрових посвідчень. Розглянуто методи забезпечення програмної безпеки IT-систем при дії ненавмисних дестабілізуючих факторів.

Ключові слова: IT-системи, інформаційна і функціональна безпека.

SECURITY OF IT-SYSTEMS (A SURVEY)

V.G. Skobelev

In the given paper it is presented short survey of models and methods intended to provide informational safety and functional of modern IT-systems with critical scope of applications. It is presented basic types of existing security servers used in IT-systems, main aims of security policy in IT-systems and methods intended for its organization and analysis. It is characterized methods intended for direct testing of IT-systems in the system elaboration process, existing models of information resources' control and some scheme of resources' access based on digital certificates. Methods intended to provide security of software for IT-systems under actions of unpremeditated destabilizing factors are considered.

Key words: IT-systems, informational and functional security.

Скобелев Владимир Геннадиевич – д-р физ.-мат. наук, д-р техн. наук, профессор, ведущий научный сотрудник Института прикладной математики и механики НАН Украины, Донецк, Украина, e-mail: skbv@iamm.ac.donetsk.ua