УДК 004.78:[002:351.75]-049.5

Е. Б. Дудин, И. А. Жлябинкова, Э. Г. Захарова, Ю. Г. Сметанин

Информационная безопасность в распределенных вычислительных системах. Обзор

Решение информационно-вычислительных задач большой размерности с использованием глобально распределенных вычислительных сред требует особого внимания к проблеме информационной безопасности. Это актуально как для самой информационно-вычислительной системы и ее составляющих подсистем, так и для ее пользователей.

Ключевые слова: *распределенные вычислительные системы: Грид-системы; информационная безопасность; механизмы безопасности; враждебные программы; обнаружение атак; диагностика уязвимости; биометрические технологии идентификации.*

ВВЕДЕНИЕ

Информационная безопасность является одним из основных требований к большим распределенным вычислительным системам, особенно тем, в которых объединены автономные системы, принадлежащие различным коммерческим или научным организациям. Без уверенности в целостности данных и ресурсов, защищенности пользовательской информации и конфиденциальности взаимодействий теряет свою привлекательность возможность совместно использовать процессорные ресурсы и данные.

Сложность обеспечения информационной безопасности в распределенных вычислительных системах связана с тем, что традиционные меры, основанные на изоляции систем и защите ресурсов за счет правил, ограничивающих возможности пользователей, противоречат главной идее Грид-сред (термин грид используется для обозначения технологии интеграции реальных суперкомпьютеров, кластеров, сетей хранения данных и научных приборов с целью образовать сетевой виртуальный суперкомпьютер) — возможности совместного использования ресурсов вне зависимости от границ организаций и даже стран. Поэтому необходимы комплексные мероприятия по обеспечению безопасности, и реализованная в организации стратегия защиты может влиять на всю информационную инфраструктуру организации.

Здесь будут описаны основные задачи, решение которых необходимо для обеспечения информационной безопасности в распределенных вычислительных системах, а также некоторые новые подходы к решению этих задач. Особое внимание уделено требованиям к механизмам защиты, вопросам обнаружения атак и верификации протоколов.

ОБЕСПЕЧЕНИЕ БЕЗОПАСНОСТИ В ВЫЧИСЛИТЕЛЬНОЙ СЕТИ

Вычисления в Грид-средах состоят из множества процессов, число которых может измеряться сотнями. Процессы должны эффективно обмениваться сообщениями и уметь динамически запрашивать ресурсы из различных доменов. Этими особенностями Грид-сред обусловлены повышенные требования к обеспечению безопасности, для решения которых существующие технологии оказываются малопригодными. Размеры и динамическая конфигурация вычислительной сети не позволяют устанавливать доверительные отношения между произвольными узлами, выполняющими приложение. Необходимо при этом организовать безопасное взаимодействие не только между клиентом и сервером, но и между множеством различных процессов, входящих в различные административные домены, и обеспечить безопасность при многодоменной организации вычислительной сети, когда внутри каждого домена используется своя технология безопасности.

Обеспечение безопасности в распределенной вычислительной сети предполагает реализацию таких функций, как аутентификация, авторизация, проверка целостности, обеспечение конфиденциальности и отказоустойчивости.

Далее используются следующие термины:

Субъект — участник взаимодействия (пользователь, процесс, ресурс);

Реквизиты — информация для проверки идентичности субъекта (аутентификации);

Объект — защищенный ресурс;

Авторизация — процесс определения прав субъекта на доступ или использование объекта;

Доверительный домен — логическая административная структура, в рамках которой реализуется единый механизм безопасности;

Атака — действия злоумышленника, предпринятые с целью нарушения информационной безопасности системы; обычные результаты успешной атаки — несанкционированный доступ нарушителя к информации, потеря работоспособности системы или искажение хранящихся данных.

Архитектура обеспечения безопасности включает следующие компоненты: сущности, реквизиты, протоколы. Субъектами являются пользователи и процессы, а объектами — ресурсы. Процесс вычислений во время выполнения может динамически использовать ресурсы с освобождением после завершения их использования. Используя ресурс, процесс вычислений делает это от имени пользователя, однако вариант, когда пользователь взаимодействует с каждым ресурсом напрямую с целью аутентификации, неприемлем, так как количество ресурсов может быть большим или приложение может выполняться в течение длительного времени на динамичном наборе ресурсов.

В результате возникла идея *представителя пользователя (user proxy)*, который может осуществлять операции от имени пользователя, не требуя его непосредственного участия. Представитель пользователя — это сеансовый процесс, которому дано право осуществлять операции от имени пользователя в течение ограниченного промежутка времени. Он имеет свои реквизиты, что устраняет необходимость постоянного присутствия пользователя и доступности пользовательских реквизитов.

Аналогично определяется сущность, которая представляет ресурс. *Представитель ресурса (resource proxy)* — это агент, обеспечивающий взаимодействие междоменного и внутридоменного механизмов безопасности.

В дальнейшем будут использоваться следующие обозначения: ПП — представитель пользователя, ПР — представитель ресурса, Сх — реквизиты субъекта X, SIGx{текст} — "текст", подписанный субъектом X.

Наличие взаимодействующих сторон и повышенные требования к защите от злоумышленников заставляют использовать доверенную третью сторону, проверяющую, что взаимодействуют именно те участники, которые и должны взаимодействовать. При этом важнейшим понятием является понятие сертификата, используемого для идентификации пользователей Грид-среды. Сертификат включает:

- имя субъекта,
- открытый ключ субъекта,
- идентификатор органа, предоставившего сертификат и гарантирующий его подлинность,
- цифровую подпись этого органа.

На основании сертификата строятся режимы взаимной идентификации пользователей.

В [1] предложена концепция специальной архитектуры обеспечения безопасности в вычислительных сетях, которая учитывает следующие характерные их черты.

- Сообщество пользователей обширно и динамично, а участниками виртуальных организаций могут быть представители разных организаций, причем состав их может часто меняться.
- Пул ресурсов также обширен и динамичен, и отдельные организации и пользователи сами решают, какие ресурсы они будут предоставлять и каким образом, причем состав и местоположение этих ресурсов могут меняться.
- В процессе вычислений может потребоваться инициировать вспомогательные процессы на удаленных узлах и динамически освобождать их после окончания работы, т. е. процесс вычислений состоит из множества вспомогательных процессов, использующих различные ресурсы и выполняющихся на разных узлах.
- При организации взаимодействия вспомогательных процессов используются разнообразные механизмы, в ходе выполнения которых различные низкоуровневые соединения могут динамически создаваться и уничтожаться.
- Удаленным ресурсам могут требоваться различные механизмы аутентификации и авторизации, корректировка которых пользователем обычно весьма ограничена.
- Индивидуальный пользователь может быть ассоциирован для обеспечения контроля доступа с различными учетными записями на разных узлах, при этом одни пользователи могут иметь постоянную учетную запись, другие могут входить в систему как гости, а для некоторых учетная запись может формироваться динамически.
- Пользователи и используемые ими ресурсы могут находиться в разных странах, что приводит к дополнительным трудностям.

Из приведенных характерных особенностей вычислительных сетей вытекают следующие *требования к механизму безопасности,* учитываемые в предлагаемой концепции:

- однократная аутентификация для обеспечения входа в систему и доступа ко всем ресурсам;
- защищенность всех пользовательских реквизитов;
- интероперабельность с локальными системами безопасности; при этом междоменный механизм доступа обеспечивается основной системой безопасности, а доступ к локальным ресурсам зависит от локального механизма защиты;
- экспортируемость — код должен быть экспортируемым и исполняемым в многонациональных вычислительных сетях, а механизм безопасности не может прямо или косвенно требовать тотального шифрования всех данных;
- унифицированная инфраструктура сертификации;
- поддержка безопасности для взаимодействий в динамических группах;
- поддержка многочисленных реализаций — механизм обеспечения безопасности не должен определять специфику его реализации.

Механизм безопасности представляет собой набор правил, определяющих субъекты и объекты безопасности (пользователи и ресурсы) и отношения между ними. Строить этот механизм предлагается в соответствии со следующими принципами.

1. Вычислительная сеть состоит из многочисленных доверительных доменов. Это значит, что данный механизм должен объединять разнородные наборы локально администрируемых пользователей и ресурсов. Основное внимание должно уделяться междоменным взаимодействиям и их рассмотрению в рамках локальных механизмов безопасности.

2. Операции, связанные с одним доверительным доменом, являются субъектами только для локального механизма безопасности. То есть на локальный механизм безопасности не накладывается никаких дополнительных требований, и его реализация осуществляется независимо от всей среды.

3. Имеются глобальные и локальные субъекты, и для каждого доверительного домена существует частичное преобразование глобальных сущностей в локальные сущности. Таким образом, каждый

пользователь будет иметь два имени: глобальное и, возможно отличающееся от него, локальное. Преобразование глобального имени в локальное специфично для каждого узла. Например, узел может преобразовывать глобальные имена пользователей в предопределенное локальное имя, динамически выделять локальное имя или группу имен. Существование глобальных сущностей позволяет обеспечить однократную аутентификацию пользователя.

4. Операции между сущностями, расположенными в разных доверительных доменах, требуют взаимной аутентификации.

5. Глобально идентифицированный субъект, преобразованный в локальный, считается эквивалентом локально идентифицированного субъекта.

6. Все решения, связанные с контролем доступа, принимаются локально на основе локального субъекта, т. е. контроль доступа остается в руках администраторов локальных систем.

7. Процессу разрешается осуществлять действия от имени пользователя и получать права пользователя. Благодаря этому система способна поддерживать долгоживущие процессы и позволяет запускать новые вспомогательные процессы.

8. Процессы, запущенные от имени одного пользователя в рамках одного доверительного домена, могут иметь общее множество реквизитов. Вычисления могут включать сотни процессов, выполняющихся на одном ресурсе. Масштабируемость должна обеспечиваться отказом от присваивания каждому процессу уникальных реквизитов.

Диапазон взаимодействий между сущностями определяется функциональностью вычислительной сети и приложений, однако наиболее вероятны следующие операции:

- выделение ресурса пользователю;
- выделение ресурса процессу;
- взаимодействие процессов, расположенных в разных доверительных доменах.

Важными элементами архитектуры являются протоколы взаимодействия субъектов и объектов. Возможны протоколы:

создания ПП;

выделения ресурса пользователю;

выделения ресурса процессу;

преобразования глобальных субъектов в соответствующие локальные субъекты.

Протокол создания ПП. Пользователь получает доступ к узлу, на котором будет создан ПП, используя локальный механизм идентификации. Далее пользователь формирует реквизиты Спп, используя свои реквизиты для подписи кортежа с необходимой информацией: Спп=SIGп {id пользователя, время начала работы, время окончания работы, идентификационная информация и т. д.}. Далее создается процесс ПП, снабжается созданными реквизитами и начинает свою работу.

Протокол выделения ресурса пользователю. ПП и ПР идентифицируют друг друга, используя Спп и Спр. При этом ПР проверяет, что реквизиты ПП не устарели и не требуют обновления. Далее ПП передает запрос ПР на выделение ресурса в специальной форме SIGпп {спецификация запроса}. Затем ПР на основе локального механизма защиты проверяет права ПП на использование данного ресурса. Если прав достаточно, то создается кортеж с именем пользователя, которому выделен ресурс, именем ресурса и т. п. Далее этот кортеж поступает к ПП, где проверяется и, если проверка прошла успешно, подписывается реквизитами ПП для получения реквизитов Спроц для инициирования процесса на ресурсе. Эти реквизиты поступают к ПР, который производит запуск процесса на ресурсе с данными реквизитами.

Протокол выделения ресурса процессу. Процесс и его ПП идентифицируют друг друга, используя Спроц и Спп. Процесс формирует запрос на выделение (allocation) ресурса в форме SIGпроц {“allocate”, спецификация запроса} и направляет его ПП, который проверяет запрос и направляет его к ПР, используя протокол выделения ресурса пользователю. Идентификатор созданного процесса подписывается ПП и возвращается исходному процессу.

Протокол регистрации преобразований глобальных субъектов в соответствующие локальные субъекты. Происходит взаимная аутентификация ПП и ПР. ПП передает ПР подписанный запрос *MAP-SUBJECT-UP* в специальной форме, указывая в качестве параметров имена глобального субъекта и ресурса. Пользователь подключается к ресурсу, используя механизм аутентификации ресурса, и запускает процесс регистрации преобразования. Процесс регистрации преобразования обращается к ПР с запросом *MAP-SUBJECT-P*, указывая в качестве параметров имена глобального субъекта и ресурса. ПР ждет два запроса с совпадающими параметрами: *MAP-SUBJECT-UP* и *MAP-SUBJECT-P*; убеждается, что процесс регистрации преобразования принадлежит субъекту ресурса, указанному в запросе. Далее ПР устанавливает новое преобразование и отправляет уведомление процессу регистрации преобразования и ПП. Если в течение определенного времени парный запрос не был получен, то ПП очищает пришедший запрос и уведомляет ожидающий субъект. Если уведомление не получено, то запрос считается неудачным.

БИОМЕТРИЧЕСКИЕ ТЕХНОЛОГИИ ИДЕНТИФИКАЦИИ ПОЛЬЗОВАТЕЛЕЙ

При решении задач идентификации личности все большее распространение получают биометрические технологии. Сюда входят две большие группы методов, основанные соответственно на статистической идентификации (по папиллярным рисункам, рисункам кровеносных сосудов глазного дна, по особенностям радужной оболочки глаза и т. п.) и на анализе клавиатурного почерка. Преимущества второй группы связаны с тем, что они не требуют дополнительных аппаратных средств, дают возможность оценивать текущее психофизическое и психофизиологическое состояние личности и позволяют реализовать скрытный мониторинг работы операторов компьютерных систем [9].

При статической идентификации обычно используются стандартные методы анализа и распознавания сигналов и изображений. В рассматриваемой предметной области наибольшее распространение получили многомасшабные представления изображений [3], вейвлетное преобразование для выделения информативных признаков для классификации [4], анализ главных компонент для сокращения размерности пространства признаков

[5], линейный дискриминантный анализ [6], скрытые марковские модели [7], нейронные сети [8].

Имеющиеся на рынке системы идентификации пользователя по его клавиатурному почерку осуществляют аутентификацию по фиксированной ключевой фразе (паролю). Компания Net Nanny Software International разработала систему BioPassword — программное средство контроля доступа, которое отслеживает динамику клавиатурного ввода. Точность распознавания в этой системе — 96 — 97%. Продукт предназначен для использования в IBM-совместимых системах. Кроме нее следует отметить продукты Rythym (компания Keystroke Dynamics), Keystroke Biometric System (компания Keystroke International Biometric System) и Biolock (компания Keystroke Phoenix Software Company). Для идентификации пользователя по его клавиатурному почерку компания Electronic Signature Lock Corporation разработала две технологии: идентификации пользователя за локальным или удаленным терминалом Electronic Signature Lock (ESL) и Complex Electronic Signature Lock (CESL), которые позволяют не только идентифицировать оператора по его манере ввода пароля, но и обеспечивают непрерывный контроль адекватности поведения оператора или его замены [9].

СИСТЕМЫ ОБНАРУЖЕНИЯ АТАК НА ИНФОРМАЦИОННЫЕ СИСТЕМЫ

Сложность задачи защиты информационных систем связана с необычайно большим количеством возможных атак, поскольку злоумышленники используют индивидуальные подходы, а также с тем, что сетевые атаки постоянно изменяются в связи с регулярным обновлением программного обеспечения и аппаратных средств.

Распространённые системы обнаружения атак основаны на статистическом подходе [10]. В этом подходе определяются и анализируется отклонения от профиля нормального поведения пользователя, построенного на основе различных параметров (к таким параметрам относятся: загрузка процессов, операции ввода/вывода, системные вызовы и т. п.), или несоответствия между текущим режимом работы и режимом работы, отвечающим штатной модели. Эти профили постоянно изменяются с целью адаптации к поведению конкретного пользователя. Наличие аномалий служит сигналом о возможности атаки. Преимущество методов данного типа заключается в возможности обнаружения новых атак без модификации или обновления параметров модели. Основным недостатком этого метода является возможность пропуска атак и большое число ложных тревог. Кроме того, обычно очень трудно создать точную модель нормального функционирования системы или действий пользователей. Для решения этой задачи требуется строить поведенческие модели, решение которых достаточно сложная задача при имеющихся на практике ограничениях по времени и выделяемых на защиту ресурсов.

Наряду с методами анализа аномалий применяются сигнатурные методы [11]. Системы обнаружения злоупотреблений хранят сигнатуры уже известных атак, на основе которых и выполняется процедура поиска сигнатуры атаки. Среди сигнатурных методов выявления атак наиболее распространен метод контекстного поиска, который заключается в обнаружении определенного множества символов. Часто используются специализированные языки, описывающие сигнатуру атаки. Опыт применения сигнатурных методов показал, что они обеспечивают высокую точность определения факта атаки. Этот подход эффективен при защите от известных атак, но практически непригоден для выявления принципиально новых атак.

Для обеспечения надежной защиты требуется объединить достоинства обоих подходов. Получаемая система должна решать задачи обнаружения модифицированных атак, сигнатуры которых отсутствуют в базе данных, атак с частично известными характеристиками, новых атак, не зарегистрированных ранее. Возможный подход такого комбинирования заключается в создании (нейросетевых) обучающихся систем с использованием (нечетких) представлений, учитывающих неопределенность информации. Обучение должно позволять модифицировать как профили пользователей, так и сигнатуры атак.

Реализация описанной общей схемы позволит проводить диагностику уязвимости программного обеспечения, протоколов и форматов данных — одной из основных причин нарушения безопасности информационных систем.

Принципиальная схема описанного подхода представлена в [12]. Основные этапы её реализации следующие.

1. Построение динамической модели исполнения программы (графа исполнения) на основе комбинации формальных методов и экспериментального анализа.

   1.1. Создание графа исполнения на основе статического анализа исходного кода. Подобный метод применяется в ряде систем исследования программного обеспечения, в частности в дизассемблере IDA фирмы DataRescue.

   1.2. Уточнение графа исполнения на основе трассировки с внесением в программу случайных данных. Подобные методы также применяются в средствах исследования программ, однако шаблоны для внесения данных подготавливаются исследователем, что замедляет процесс. Оригинальными особенностями данного метода являются использование искусственных нейронных сетей для подготовки шаблонов данных в автоматическом режиме, а также наличие обратной связи между процессом выполнения и подготовкой данных для следующего прогона. Комбинация формальных и экспериментальных методов позволяет за малое время получить граф исполнения, близкий к реальному.

2. Исследование графа исполнения на основе искусственных нейронных сетей. Применение искусственных нейронных сетей, в отличие от сигнатурного поиска, используемого в настоящее время, позволит выделить точки нахождения возможных уязвимостей с низкими значениями ошибок ложного опознания и ложного пропуска.

3. Исследование выделенных точек возможных уязвимостей на основе нейронечеткой экспертной системы, оценка наличия и степени опасности уязвимости. Второй этап проверки на основе экспертной системы позволит в значительной мере снизить вероятность ошибки обнаружения уязвимости, ускорить процесс ее обнаружения и уменьшить степень участия эксперта в процессе обнаружения.

4. Формирование рекомендаций по устранению выделенных уязвимостей.

Для реализации предлагаемого подхода разработана архитектура автоматизированного обнаружения уязвимостей. Ее задача — построение за малое время графа исполнения, близкого к реальному. В системе используется объединение статистического и динамического анализа исполняемых модулей программного комплекса. Генератор данных динамически создает входные данные с помощью искусственной нейронной сети таким образом, чтобы обеспечить наибольшую точность модели исполнения программы. Генератор данных, который также использует нейронную сеть, строит входные данные для модуля построения динамической модели программы. Цель генерации входных данных — максимальное покрытие ветвей кода программы при минимальном количестве входных данных. Входные данные строятся на основе заранее заданных шаблонов. Используется многослойный персептрон с тремя скрытыми слоями. Модуль исследования графа выполняет свою работу параллельно с подсистемой подготовки динамической модели исполнения, что позволяет выявлять уязвимости практически в режиме реального времени, а также корректировать работу генератора данных с целью уточнения информации об уязвимости. Для обучения на основе базы признаков для достижения заданной цели используются генетические алгоритмы. Цели внедрения данных — максимальное покрытие ветвей кода и максимальная глубина внедрения данных. Данные модифицируются с использованием методов генетической оптимизации.

Результаты работы модуля исследования графа являются входными данными для экспертной системы, реализованной с помощью нечеткой нейронной сети. Экспертная система определяет узлы графа, которые могут содержать уязвимости. Для этого используются два основных метода:

1) поиск по образцам, в котором искусственная нейронная сеть по элементам кода узла графа определяет возможность наличия в данном сегменте кода уязвимостей. Для этого целесообразно использовать вероятностные нейронные сети;

2) сбор статистических данных о выполнении участков кода и их анализ с использованием нечеткой логики.

ВЕРИФИКАЦИЯ ФУНКЦИЙ БЕЗОПАСНОСТИ ПРОТОКОЛА IPsec v2

Для решения вопросов защиты передачи данных в Интернете Комитет стандартизации Интернет (Internet Engineering Task Force, IETF) разработал и опубликовал в 1995–1998 гг. ряд стандартов, регулирующих архитектуру и протоколы защиты сообщений сетевого уровня. Разработанный набор стандартизованных технологий, протоколов и сервисов защиты IPsec широко использовался для защиты сетевого уровня; он оказался эффективным при построении закрытых виртуальных сетей (Virtual Private Network, VPN). В 2005–2006 гг. разработана вторая версия IPsec — IPsec v2, в которой была значительно переработана общая архитектура защиты, при этом был разработан новый протокол обмена ключами IKE v2, не совместимый с предыдущей версией. В настоящее время уже появились первые реализации IPsec второй версии, и нет сомнений в том, что поддержка IPsec будет встраиваться во всё большее число реализаций стека протоколов Интернета.

При разработке набора сервисов и протоколов IPsec необходимо уделить особое внимание обеспечению совместимости: две корректные реализации IPsec должны успешно взаимодействовать при передаче защищённых данных. Для обеспечения совместимости различных реализаций IPsec v2 в Интернете необходимо проверить, что каждая из них удовлетворяет стандартам IPsec v2. Основным средством верификации является тестирование проверки на соответствие стандартам.

Работы по созданию тестовых наборов для IPsec v2 ведутся во многих организациях, однако готовых тестовых наборов для верификации реализаций IPsec v2 в настоящее время пока нет. Обычно тесты разрабатываются вручную, без использования автоматизации, в результате возникают проблемы полноты тестового набора и оценки корректности тестового набора. Подробный обзор возникающих задач и подходов к их решению представлен в [13].

Некоторый прогресс в решении указанных проблем достигнут в исследованиях в области тестирования с использованием моделей. В этом подходе для автоматической генерации тестовых последовательностей используются формальные модели системы и тестов. Модели полезны для автоматической проверки корректности реализации и оценки полноты проведенного тестирования. Наибольшее распространение в телекоммуникационных системах получили подходы к формальному моделированию протоколов с помощью автоматных моделей, алгебр взаимодействующих процессов, систем размеченных переходов и темпоральных логик. При попытке их прямого применения к протоколам защиты данных оказывается, что получающиеся модели либо дают слишком упрощенное представление о поведении протокола, либо сталкиваются с проблемой так называемого комбинаторного взрыва.

Программа работ по созданию унифицированной методики верификации выглядит следующим образом [13].

1. Систематизация различных аспектов функций безопасности семейства IPsec v2, включая обеспечение конфиденциальности, целостности, доступности и композиционности, а также исследование методов их спецификации и верификации.

2. Разработка тестового набора для верификации реализаций IPsec v2.

2.1. Извлечение функциональных требований из стандартов IPsec v2.

2.2. Разработка формальной спецификации стандартов IPsec v2.

2.3. Разработка набора тестов, проверяющих соответствие формальной спецификации IPsec v2.

3. Разработка пакета прикладных программ для управления разработанным тестовым набором и анализа результатов тестирования.

4. Апробация разработанного тестового набора на распространённых реализациях IPsec v2

СПЕЦИФИКА ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ В УСЛОВИЯХ ЧАСТЫХ ИЗМЕНЕНИЙ И РЕСТРУКТУРИРОВАНИЯ ДАННЫХ В РАСПРЕДЕЛЕННЫХ СРЕДАХ С ИНТЕНСИВНЫМ ОБМЕНОМ ДАННЫМИ

Безопасное управление информацией в условиях ее частых изменений и изменения структуры, а также интенсивным обменом данными, представляет собой комплексную проблему; ее актуальность практически общепризнанна [14], однако общего решения пока не имеется. Известные технологии носят преимущественно локальный характер [15]. Наиболее существенное препятствие на пути решения возникающих задач — отсутствие адекватных общих моделей.

Перспективным подходом к устранению принципиальных затруднений, возникающих при попытке найти такое решение [16], является новый оригинальный подход на основе реляционных модулей, интегрированных с вычислительной средой аппликативного типа, в которых персонификация информации и обеспечение ее безопасности будет вестись с организацией среды полной частично упорядоченной системы объектов метаданных [17].

Еще один интересный подход — попытка создания концепции "естественного компьютинга" [18] в противовес нынешнему "искусственному компьютингу", однако эта концепция, в общем, пока еще находится на уровне деклараций.

НЕВИДИМОСТЬ ВРАЖДЕБНЫХ ПРОГРАММ И БЕЗОПАСНЫЕ ИНТЕРФЕЙСЫ

Перспективное направление исследований — решение проблемы "невидимости враждебных программ", когда подсистема защиты информации "не видит" в памяти и в процессе функционирования враждебную программу и поэтому не может обеспечить защиту [19]. Для решения этой задачи в [20] предложена концепция разработки безопасных интерфейсов для распределенных информационно-вычислительных систем [20], которую можно кратко описать следующим образом. За счет снижения пропускной способности скрытых каналов, в которых возможно вторжение интеллектуальных противников вне защищаемой автоматизированной системы, можно внести усложнения, ставящие перед противником сложные и неожиданные задачи. Поскольку интеллектуальный потенциал программно и аппаратно реализованных интеллектуальных агентов нарушителя безопасности компьютерной среды, используемых для нанесения ущерба, ограничен, усложнения могут привести к тому, что нарушитель не сможет реализовать эффективное решение поставленных перед ним задач. Эти задачи связаны с распознаванием искомых данных и их структур, распознаванием программ, реализующих определенные алгоритмы, отделением истинных исходных данных от ложных данных и истинных результатов решенных задач от ложных решений. При правильном выборе усложнений агент не сможет увидеть скрываемые процессы и данные. Принцип защиты заключается в разработке требуемых усложнений.

Разработка безопасных интерфейсов, в частности, должна решать задачу защиты скрытых каналов от взаимодействия враждебных программно-аппаратных агентов в распределенной информационной системе. Эта задача тесно связана с разработкой моделей взаимодействия бизнес-компонент в реальных бизнес-процессах. Исследование этих вопросов ведется в рамках единого моделирования информационной безопасности и бизнес-процессов на языке UML. На этом пути получены достаточно интересные теоретические и некоторые практические результаты [20].

ЗАКЛЮЧЕНИЕ

Решение информационно-вычислительных задач большой размерности предполагает либо создание суперкомпьютера неограниченной мощности, либо использование глобально распределенной вычислительной среды, допускающей включение неограниченного множества участников, способной к самоорганизации, автоматической реструктуризации и обеспечению информационной безопасности как для самой системы, так и для ее потребителей. Последнее направление, называемое некоторыми авторами *универсальным метакомпьютингом* [21], и развивается в Грид-технологиях и системах, информационной безопасности которых посвящен настоящий обзор.

СПИСОК ЛИТЕРАТУРЫ

1. Foster I., Kesselman C., Tsudik G., Tuecke S. A security architecture for computational grids // ACM Conf. Comput. and Security.— 1998.— P. 83–91.

2. Брюхомицкий Ю. А., Казарин М. Н. Метод биометрической идентификации пользователя по клавиатурному почерку на основе разложения Хаара и меры близости Хэмминга // Известия ТРТУ.— Таганрог: Изд-во ТРТУ, 2003.— Т. 33, № 4: Материалы 5-й Междунар. науч.-практ. конф. "Информационная безопасность".— С. 141–149.

3. Jacobs C. E., Finkelstein A., Salesin D. H. Fast multiresolution image querying // Proc. of SIGGRAPH'95.— New York, 1995.— Vol. 286.— P. 277–286.

4. Столинц Э., Де Роуз Т., Салезин Д. Вейвлеты в компьютерной графике / пер. с англ.— Ижевск: НИЦ РХД, 2002.— 272 с.

5. Martinez A. M., Kak A. C. PCA versus LDA // IEEE Trans. Pattern Anal. and Mach. Intell.— 2001.— Vol. 23, № 2.— P. 228–233.

6. Lu J., Plataniotis K. N., Venetsanopoulos A. N. Face recognition using LDA-based algorithms // IEEE Trans. Neural Networks.— 2003.— Vol. 14, № 1.— P. 195–200.

7. Baumes J., Goldberg M., Magdon-Ismail M., Wallace W. On hidden groups in communication networks // Technical report.— 2005.— 15 May.— P. 1–25.

8. Ryan J.; Lin M., Mikkulainen R. Intrusion detection with neural networks // AAAI workshop: Approaches to fraud detection and risk management (Providence, RI).— 1997.— P. 72–79.

9. Брюхомицкий Ю. А. О способах представления клавиатурных параметров личности // Информационная безопасность: Материалы 10-й Междунар. науч.-практ. конф.— Таганрог, 2008.— С. 221–224.

10. Winkeler J. R. A UNIX prototype for intrusion and anomaly detection in secure networks // The Thirteenth national computer security conference (Washington, DC).— 1990.— P. 115–124.

11. Lindqvist U., Porras P. A. Detecting computer and network misuse with the production-based expert system toolset // IEEE Symp. Security and Privacy.— Los Alamitos: IEEE CS Press, 1999.

12. Бабенко Л. К., Захаревич В. Г., Макаревич О. Б. Современные проблемы информационной безопасности и их реализация в научной деятельности Южного Федерального Университета // Известия ЮФУ. Техн. науки.— 2007.— № 1.— С. 6–19.

13. Бурдонов Н. Б., Косачев А. С., Пономаренко В. Н., Шнитман В. З. Обзор подходов к верификации распределенных систем / Препринт.— М.: ИСП РАН, 2006.— 61 с.

14. Carpenter B. The Internet engineering task force. Overview, Activities, Priorities // IETF Report to ISOC BoT.— 2006.— Oct.— P. 2–10.

15. Zaniolo C., Ceri S., Faloutsos C., Snodgrass R. T., Subrahmanian V. S., Zicari R. Advanced Database Systems.— Morgan Kaufmann, 1997.— 574 p.

16. Zimmermann O., Tomlinson M., Peuser S. Perspectives on Web services.— New York: Springer Verlag, 2003.— 648 p.

17. Вольфенгаген В. Э. Методы и средства вычислений с объектами. Аппликативные вычислительные системы.— М.: Центр ЮрИнфоР, 2004.— 789 с.

18. Denning P. J. Computing is a natural science // Commun. ACM.— 2007.— Vol. 50, № 7.— P. 13–18.

19. Grusho A., Grebnev N., Timonina E. Covert channel invisibility theorem // Proceedings of Fourth International Conference on Mathematical Methods, Models, and Architectures for Computer Network Security (MMM-ACNS 2007).— 2007.— P. 1867–1896.

20. Грушо А. А., Грушо Н. А., Тимонина Н. А. О безопасности и надежности подсистем защиты в распределенных информационных системах // Системы и средства информатики / Ин-т проблем информатики РАН.— 2007.— № 17.— С. 79–85.

21. Золотницкий И. Ю., Шеин А. В. Проблемы защиты информации в глобально-распределенных информационно-вычислительных системах [Электрон. ресурс].— URL: http://rcdl2002.jinr.ru/Reports/Vol_2/vol2_272-280.pdf

Материал поступил в редакцию 27.03.09.

УДК [002::004]:002

М. С. Трахтенгерц

ABCD — автоматизированная библиотечная система на базе WinISIS

Описываются структура и возможности создаваемой системы автоматизации он-лайновых электронных и традиционных библиотек, использующей СУБД обработки текстовой информации WinISIS.

Ключевые слова: *автоматизация, он-лайновые электронные библиотеки, обработка текстовой информации*

На всемирном конгрессе пользователей мультибазовой системы WinISIS [1, 2] было объявлено о планах создания ряда прикладных комплексов, использующих её для автоматизации некоторых информационных процессов. В их числе была система автоматизации библиотечной деятельности, названная "ABCD". В настоящее время, т. е. к середине 2009 г., практически полностью закончена её отладка и создана функциональная версия системы. Она, как и другие продукты семейства ISIS, распространяется бесплатно.

ABCD является акронимом для полного наименования "Программное средство для автоматизации библиотек и центров документации", по-французски — Automation des Bibliotheques et Centres de Documentacion.

Основной чертой разрабатываемой системы является её взаимодействие с пользователями через Интернет. При этом решаются следующие задачи:

— Библиотеки, выполнявшие до сих пор классические функции традиционного обслуживания читателей книгами и другими печатными текстами, могут выполнять действия, присущие центрам документации.

— Подразумевается существенное расширение видов "текстовых продуктов", вошедших в читательский и научный обиход в последнее время, в том числе существующих только в форме электронных документов.

— Описываются и включаются в активное использование документы со структурой, отличающейся от традиционной (книги, статьи), например, таблицы данных, графическая информация, многотомные продолжающиеся издания, когда пользователю нужны не полные фолианты, а лишь определенные разделы из них и т. д.

— Применение различных программных модулей, разработанных в процессе автоматизации реально существующих библиотек, прежде всего по достаточно сложной медицинской тематике, позволит не только облегчить управление библиотекой, но и создавать новые услуги для потребителей.