МАТЕРИАЛЫ XXI ВСЕРОССИЙСКОЙ
НАУЧНО-ПРАКТИЧЕСКОЙ КОНФЕРЕНЦИИ

БИТ

*И. М. Азымшин, В. О. Чуканов*

# АНАЛИЗ БЕЗОПАСНОСТИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

В статье рассматривается вопрос оценки безопасности работы программного обеспечения (ПО). Безопасность работы ПО обеспечивается надежностью его работы. Оценку надежности ПО можно производить при помощи моделей Джелинского—Моранды и Шика—Волвертона. Но в данных моделях не учитывается возможность возникновения новых ошибок при исправлении ранее обнаруженных. Поэтому предлагается модифицировать выбранные модели, учитывая этот недостаток.

Рассмотрим **модель Джелинского—Моранды**. Это модель с дискретным убыванием интенсивности отказов. В этой модели предполагается, что интенсивность ошибок описывается кусочно-постоянной функцией, пропорциональной числу неустраненных ошибок. То есть предполагается, что интенсивность отказов $\lambda(t)$ постоянна до обнаружения и исправления ошибки, после чего она опять становится постоянной, но с другим, меньшим, значением. При этом предполагается, что между $\lambda(t)$ и числом оставшихся в программе ошибок существует прямая зависимость

$$\lambda(t) = k(M - i) = \lambda_i,$$

где M — неизвестное первоначальное число ошибок; i — число обнаруженных ошибок, зависящих от времени t; k — константа.

Значение неизвестных параметров k и M может быть оценено на основе последовательности наблюдений интервалов между моментами обнаружения ошибок по методу максимального правдоподобия.

Рассмотрим **модель Шика—Волвертона**. Это модификация модели Джелинского—Моранды для случая возникновения на рассматриваемом интервале более одной ошибки. При этом считается, что исправление ошибок производится лишь по истечении интервала времени, на котором они возникли. В основе модели Шика—Волвертона лежит предположение, согласно которому частота ошибок пропорциональна не только количеству ошибок в программах, но и времени тестирования, то есть вероятность обнаружения ошибок со временем возрастает. Интенсивность обнаружения ошибок $\lambda_i$ предполагается постоянной в течение интервала времени $t_i$ и пропорциональна числу ошибок, оставшихся в программе по истечении $(i-1)$-го интервала, но она пропорциональна также и суммарному времени, уже затраченному на тестирование:

$$\lambda_i = C(N - n_{i-1})\left(T_{i-1} + \frac{t_i}{2}\right),$$

где C — коэффициент пропорциональности; N — число ошибок, первоначально присутствующих в программе; $n_{i-1}$ — число ошибок, оставшихся в программе по истечении (i — 1)-го интервала; $T_{i-1}$ — суммарное время, затраченное на тестирование в течение (i — 1) этапов; $t_i$ — среднее время выполнения программы в текущем интервале.

Остальные расчеты аналогичны расчетам модели Джелинского—Моранды.

Представленные модели не учитывают повторяемость этапов разработки программного обеспечения. После этапа разработки и отладки идет этап тестирования, и по его результатам принимается решение о готовности продукта или необходимости его доработки. Но на этапе доработки могут возникнуть изменения, которые могут повлиять на надежность программного обеспечения.

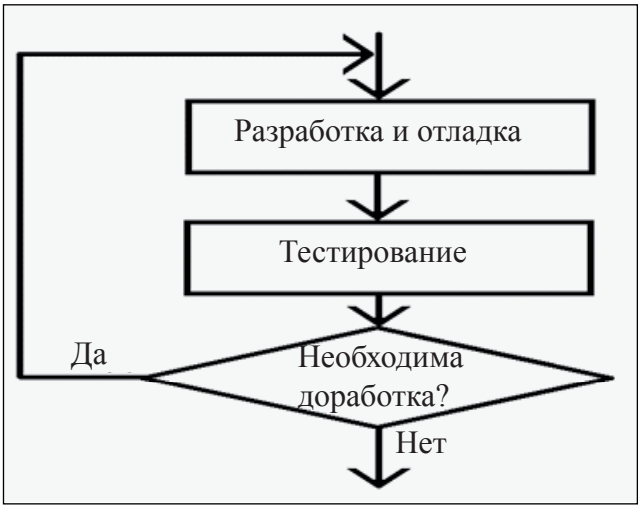



*Рис. 1. Основные этапы разработки программного обеспечения*

**Улучшенная модель Джелинского—Моранды**. Предполагается, что при исправлении ошибок, обнаруженных на предыдущем этапе тестирования, возможно возникновение новых.

$$\lambda(t) = k(M + H - i) = \lambda_i,$$

где $H$ — количество ошибок, появившихся в результате исправления ошибок, обнаруженных на предыдущем этапе тестирования.

$$H = f \cdot g \cdot i \cdot \sum_j P_j,$$

где $i$ — число ошибок на предыдущем этапе; $P_j$ — вероятность появления новой ошибки ($P_1$ — вероятность появления одной ошибки, $P_2$ — вероятность появления второй ошибки и т. д.), определяется при помощи экспертной оценки; $f$ — коэффициент опыта команды; $g$ — коэффициент готовности продукта (функция от времени).

**Улучшенная модель Шика—Волвертона.**

$$\lambda_i = C(N + H - n_{i-1})\left(T_{i-1} + \frac{t_i}{2}\right),$$

где $H$ — количество ошибок, появившихся в результате исправления ошибок, обнаруженных на предыдущем этапе тестирования.

$$H = f \cdot g \cdot i \cdot \sum_j P_j,$$

где $i$ — число ошибок на предыдущем этапе; $P_j$ — вероятность появления новой ошибки ($P_1$ — вероятность появления одной ошибки, $P_2$ — вероятность появления второй ошибки и т. д.),

определяется при помощи экспертной оценки; $f$ — коэффициент опыта команды; $g$ — коэффициент готовности продукта (функция от времени).

В статье рассмотрены модели надежности Джелинского—Моранды и Шика—Волвертона. Описан недостаток данных моделей и предложено решение по их улучшению. Представлены модифицированные модели Джелинского—Моранды и Шика—Волвертона.

## СПИСОК ЛИТЕРАТУРЫ:

1. *Чуканов В. О.* Надежность программного обеспечения и аппаратных средств систем передачи данных атомных электростанций: Учебное пособие. М.: МИФИ, 2008.
2. *Гуров В. В., Чуканов В. О.* Основы теории и организации ЭВМ. М.: БИНОМ. Лаборатория знаний, 2012.
3. *Александрович А. Е., Бородакий Ю. В., Чуканов В. О.* Проектирование высоконадежных информационно-вычислительных систем. М.: Радио и связь, 2004.
4. *Половко А. М., Маликова И. М.* Сборник задач по теории надежности. М.: Советское радио, 1972.

*И. В. Арзамарцев, Г. И. Борзунов*

## ОБОБЩЕННЫЙ АЛГОРИТМ СИМВОЛЬНОГО ИСПОЛНЕНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Многие существующие методы анализа ПО, в том числе направленные на выявление в нем уязвимостей [1, 2] и вредоносной составляющей [3], используют алгоритм символьного исполнения. Символьное исполнение (от англ. symbolic execution) — метод анализа потока данных, основанный на построении алгебраических уравнений, описывающих входные и выходные параметры некоторого функционального блока исследуемого ПО путем подстановки вместо реальных значений входных данных переменных величин и исполнения программы [4]. Как показано в работе [4], широко используемый алгоритм символьного исполнения не позволяет произвести эффективный анализ сложных циклических конструкций с большим либо неопределенным числом итераций из-за экспоненциального роста числа потенциальных ветвей исполнения. В ряде работ авторы предлагали подходы для преодоления данной проблемы, однако предложенные решения носили эвристический характер и не позволяли однозначно определить необходимый набор входных данных для получения искомого набора выходных данных [1, 2].

Ниже предлагается обобщенный алгоритм символьного исполнения произвольного цикла ПО.

Любую итерацию цикла, не имеющего вложенность, над некоторым набором данных $X = (x_1, x_2, \ldots x_n)$ можно представить посредством функции:

$$f(x1, x2, \ldots, xn) = \begin{cases} (\mu 11(x1), \ldots \mu nn(xn)), if\ \alpha 1(x1, \ldots, xn) = true \\ \ldots \\ (\mu 1m(x1), \ldots \mu nm(xn)), if\ \alpha m(x1, \ldots, xn) = true \end{cases}$$

$$\eta(x1, x2, \ldots, xn) = NIC$$

Здесь функция f:X—>X — функция преобразования данных итерации, где $\mu_{ij}$:$X_i$—>$X_i$ — функция преобразования i-го элемента набора данных, α:X—>$V_2$ —функция условия, которому должен удовлетворять входной набор данных, чтобы посредством итерации цикла быть преобразованным в соответствующее результирующее состояние. Представленные функции условия разбивают все пространство данных X на области эквивалентности, в которых изменения данных в итерации цикла не зависят от соотношений между значениями данных. Иными словами, в области $X_k$, в которой $ak(x1,\dots,xn) = true$, однозначно определены всевозможные переходы из этой области посредством многократного применения итерации цикла $(x1,\dots,xn) \to (\mu 1k(x1),\dots \mu nk(xn))$ для данной области. Функция η:X—>$V_2$ — функция условия продолжения цикла (от англ. NIC — next iteration condition), которая в случае отображения в 1 означает выполнение следующей итерации цикла и выхода из цикла при отображении в 0. В этом случае выполнение любой программы можно представить как совокупность переходов в пространстве X между областями, образовавшимися посредством наложения функций условий. При этом отдельно выделяются области начала исполнения, в случае наличия действий до начала цикла (как, например, у цикла for в C/C++), а также области выхода из цикла (в которых NIC = 0), в которых исполнение заканчивается. С учетом описанного способа обобщенного анализа итерации цикла предлагается использовать следующий алгоритм вычисления входного набора данных по заданному выходному:

- выделить всевозможные области, из которых осуществим переход в любую область выхода из цикла, наложенную на область искомых значений выходных данных;
- в случае наличия циклов в переходах на основе работы [5] выделить монотонность изменения данных, что позволит определить возможность выхода из данного цикла в искомую область;
- последовательно осуществить перебор всевозможных маршрутов с учетом предложенных ранее ограничений.

В случае наличия вложенных циклов предлагается вначале осуществить описанным выше методом анализ вложенного цикла с учетом областей эквивалентности внешнего цикла, а затем, представив вложенный цикл как функцию отображения над набором входных данных, использовать предложенный выше метод анализа внешнего цикла.

Описанный в работе алгоритм позволяет в общем виде осуществить анализ сложных циклических конструкций, в том числе вложенных циклов. Однако в случае покрытия маршрута всевозможных областей пространства данных без выделения участков монотонности, как, например, в случае расчета значения хеш-сумм, данный алгоритм не сокращает временную сложность нахождения набора входных данных по заданному выходному, по сравнению с полным перебором.

## СПИСОК ЛИТЕРАТУРЫ:

1. *Cha S. K., Avgerinos T., Rebert A., Brumley D.* Unleashing Mayhem on Binary Code // Proceedings of the 2012 IEEE Symposium on Security and Privacy. P. 380—394.
2. *Babic D., Martignoni L., McCamant S., Song D.* Statically-directed dynamic automated test generation // Proceedings of the 2011 International Symposium on Software Testing and Analysis. P. 12—22.
3. *Moser A., Kruegel C., Kirda E.* Exploring Multiple Execution Paths for Malware Analysis // Research Symposium on Security and Privacy. Oakland. May 2007. P. 231—145.
4. *King J. C.* Symbolic Execution and Program Testing // Communications of the ACM. 1976. V. 19. № 7 P. 385—394.
5. *Арзамарцев И. В., Юров И. А.* Метод анализа зависимостей между входными и выходными данными алгоритмов // Безопасность информационных технологий. 2013. № 2. С. 18—22.

*В. И. Васильев, Е. В. Бурая*

# БИОМЕТРИЧЕСКАЯ КРИПТОСИСТЕМА ИДЕНТИФИКАЦИИ ЛИЧНОСТИ ПО ОТПЕЧАТКАМ ПАЛЬЦЕВ

Сегодня рынок биометрических технологий переживает свой расцвет. Системы биометрической идентификации появляются в аэропортах, правоохранительных учреждениях и на частных предприятиях.

Но использование биометрических систем в открытых и слабо защищенных информационных пространствах становится проблематичным, особенно в случае необходимости приема, обработки и передачи конфиденциальной информации. Наиболее надежны в этом случае криптографические системы. Однако их серьезным недостатком является проблема надежного хранения и правильного использования секретных криптографических ключей.

Выходом из создавшейся ситуации стало появление нового класса биометрических систем идентификации личности с криптографической защитой информации, или биометрических криптосистем [1].

Применение таких систем дает следующие преимущества:

- быстрое и однозначное получение криптографического ключа легальным пользователем по его размытому биометрическому образу;
- существенное снижение вероятности случайного или преднамеренного получения по биометрическим данным криптографического ключа нелегальным пользователем;
- исключение необходимости защищенного хранения ключа.

На сегодняшний день построить биометрическую систему с криптографической защитой можно, используя различные технологии. Рассмотрим одну из них — нечеткий экстрактор. Суть нечеткого экстрактора заключается в том, что он позволяет извлечь случайную равномерно распределенную последовательность символов (секретный ключ) из первоначальных биометрических данных и далее однозначно восстанавливает ее из любых входных данных, достаточно схожих с первоначальными. Для воспроизведения секретного ключа при этом требуются дополнительные открытые данные, соответствующие этому ключу, которые хранятся в памяти [2].

Такая технология (нечеткий экстрактор) позволяет применить к биометрическим данным алгоритмы шифрования, что позволит построить биометрическую систему с криптографической защитой. Предложенный в данной работе нечеткий экстрактор, основой которого являются коды Рида—Соломона [3], работает с такими биометрическими идентификаторами, как отпечаток пальца [4], информационный вектор которого записан в формате согласно ГОСТ Р ИСО/МЭК 19794-2-2005 [5].

Проведенные тестирования позволили выявить зависимость степени схожести биометрических шаблонов от уровня зашумленности биометрических данных. Полученные результаты показали, что использование технологии нечеткого экстрактора позволит улучшить характеристики мэтчера биометрической системы, а именно уменьшить уровень ошибок распознавания.

Для метода шифрования дана сравнительная оценка, которая показывает, что биометрические криптосистемы более эффективны, чем биометрические системы, и не уступают криптосистемам.

Таким образом, использование технологии нечеткого экстрактора в биометрической криптографии не только защищает уязвимые точки биометрических систем, но и обеспечивает устойчивое кодирование биометрических идентификаторов, что, несомненно, положительно влияет на уровень ошибок распознавания системы, а значит, и на уровень ее защищенности.

СПИСОК ЛИТЕРАТУРЫ:

1. *Stan Z. Li*. Encyclopedia of Biometrics. Berlin: Springer Science+Business Media, 2009. — 1500 с.
2. *Васильев В. И*. Интеллектуальные системы защиты информации: учебное пособие. М.: Машиностроение, 2010. — 163 с.
3. The Error Correcting Codes (ECC) Page [Электронный ресурс]. URL: http://www.eccpage.com/ (дата обращения: 25.04.2013).
4. *Кухарев Г. А.* Биометрические системы. Методы и средства идентификации личности человека. СПб.: Политехника, 2001. — 240 с.
5. ГОСТ Р ИСО/МЭК 19794-2-2005 [Электронный ресурс] // Все ГОСТы. URL: http://docs.cntd.ru/document/1200044529 (дата обращения: 03.04.2013).

*В. Э. Вольфенгаген, И. А. Александрова, И. А. Волков, Б. Б. Горелов, Л. Ю. Исмаилова, С. В. Косиков, А. Д. Лаптев, И. А. Парфенова, В. Д. Петров*

## БЕЗОПАСНЫЙ РЕЖИМ КОНЦЕПТУАЛЬНОГО КРАУДСОРСИНГА БОЛЬШИХ ДАННЫХ[1]

Коротко говоря, краудсорсинг может пониматься как управление соотнесениями и свойствами, возникающими при анализе больших данных. Для этого требуется ряд аналитических действий, включая: отбор коллекций для извлечения требуемой информации; извлечение сущностей из неструктурированных или слабоструктурированных источников; оценка близости индивидов в разных коллекциях данных; слияние экземпляров индивидов, в том числе установление и отбрасывание дубликатов; построение схем данных в коллекциях и их отображение в целевую схему; формирование экземпляров действительных индивидов и концептов для данных, соответствующих целевой схеме [1—5].

### Технология краудсорсинга

Область краудсорсинга чрезвычайно бурно развивается, а к настоящему времени общепринятых систем и методов работы с ними все еще не появилось. При краудсорсинге [4] используется ручная работа для обработки, получения или порождения данных по требованию, а также для их классифицирования, ранжирования, выставления пометок или улучшения существующих данных. Эти решаемые вручную задачи часто оказываются сложными для автоматизации, например, при определении рейтинга чего-либо или кого-либо или при определении признаков интереса к какому-либо источнику данных.

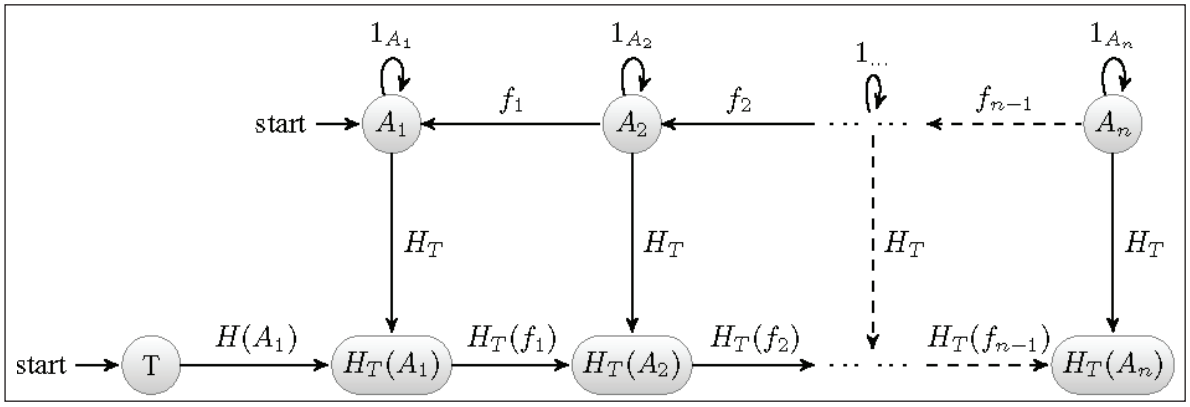
Создаваемые вручную данные можно также рассматривать как равноправный источник данных, поэтому, естественно, хотелось бы интегрировать такой краудсорсинговый источник данных с другими традиционными источниками. Это позволит конечному пользователю вместо работы с разнородными источниками данных взаимодействовать с единой унифицированной базой данных, что является преимуществом.

### Схема распознавания свойств

Группе из $n$ краудсорсеров либо экспертов с «состоянием знаний» $A_1, A_2, \ldots, A_n$ соответственно, ставится задача распознать свойство $T$ посредством распознающей функции $H_T$.

[1] Работа является обобщением результатов, которые связаны с построением концептуальной вычислительной модели и получены в разное время при выполнении проектов, частично поддержанных грантами РФФИ 14-07-00119-а, 12-07-00661-а, 14-07-00072-а, 12-07-00646-а, 13-07-00716-а, 12-07-00554-а, 14-07-00054-а, 14-07-31041-мол_а. Исследование частично поддержано грантом РНФ 14-11-00816.

Пусть они действуют в силу системы отображений $f_1, f_2, \ldots f_{n-1}$. Тогда на качественном уровне карта их последовательных действий имеет вид в соответствии со следующей коммутативной диаграммой:




На этой диаграмме

$H_T(A_j) = \{h' | h' : A_j \to T\}$

представляет собой область: множество индивидов $h'$ «в поле зрения» $A_j$, которые обладают свойством $T$. Если $f : A_{j+1} \to A_j$ для $j \geq 1$ в теории, то пусть $H_T(f_j)$ является отображением, которое переводит индивид $h' \in H_T(A_j)$, распознанный краудсорсером $_j$, в $h' \circ f_j \in H_T(A_{j+1})$, где его может обрабатывать краудсорсер $j + 1$ и т. д.:

$H_T(f_1) : h \in H_T(A_1) \to h \circ f_1 \in H_T(A_2)$

$H_T(f_2) : h \circ f_1 \in H_T(A_2) \to h \circ f_1 \circ f_2 \in H_T(A_3)$

. . . : . . .

$H_T(f_{n-1}) : h \circ f_1 \circ \ldots \circ f_{n-2} \in H_T(A_{n-1}) \to h \circ f_1 \circ \ldots \circ f_{n-1} \in H_T(A_n)$.

В этих условиях нетрудно показать, что

$H_T(f_1 \circ f_2 \circ \ldots \circ f_{n-1}) : h \in H_T(A_1) \to h \circ f_1 \circ f_2 \circ \ldots \circ f_{n-1} \in H_T(A_n)$

и

$H_T(f_{n-1}) \circ \ldots \circ H_T(f_2) \circ H_T(f_1) = H_T(f_1 \circ f_2 \circ \ldots \circ f_{n-1})$.

Возникает вопрос, насколько семантически стабилен проклассифицированный таким способом образ данных, находящийся в информационной системе. Для этого придется сделать определенное заключение о виде распознающей функции.

### Представляющая категория

Соображения о свойствах применяемых распознающих функций сводятся к тому, что они обладают возможностями функторной категории. Для этого рассмотрим декартово замкнутую категорию (д.з.к.) C, из которой берутся все рассмотренные объекты $A_1, A_2, \ldots, A_n$ и отображения $f_1, f_2, \ldots f_{n-1}$. Предложенная конструкция имеет прямую связь с обычной теорией множеств. Пусть $S$ является категорией всех множеств и произвольных функций, которая, как известно, тоже является д.з.к. Возьмем функторную категорию

$C^{op} \to S$

всех контравариантных функторов из $C$ в $S$ с естественными преобразованиями в качестве отображений. Как можно показать, исходная категория $C$ имеет полное и непротиворечивое погружение в $C^{op} \to S$.

Строение контравариантного функтора характеризуется следующим образом. Это отображение

$F: C^{op} \to S$,

которое каждому объекту $A$ из $C$ ставит в соответствие множество $F(A)$ из $S$, а каждому отображению $f : B \to A$ из $C$ – функцию $F(f) : F(A) \to F(B)$, такую, что

$F(1_A) = 1_{F(A)}$,

$F(f \circ g) = F(g) \circ F(f)$

при условии, что функции $f : B \to A$ и $g : C \to B$ берутся из $C$.

Строение естественного преобразования $v : F \to G$ для $F,G : C \to S$ характеризуется следующим образом. Это отображение, которое каждому объекту $A$ из $C$ ставит в соответствие функцию

$v_A : F(A) \to G(A)$

так, что для каждой функции $f : B \to A$ из $C$ выполняется равенство

$v_B \circ F(f) = G(f) \circ v_A$.

Для получения практически значимого для краудсорсинга применения остается конкретизировать вид функций $F$ и $G$. Например, потребовав, чтобы они были концептами. В частности, можно положить $F \equiv H_T$.

**Заключение**

Предложена семантическая характеризация безопасного режима концептуального краудсорсинга больших данных.

1. Построена коммутативная диаграмма, характеризующая работу группы из $n$ краудсорсеров либо экспертов с «состоянием знаний» $A_1, A_2, \ldots, A_n$ соответственно, которым ставится задача распознать свойство $T$ посредством распознающей функции $H_T$.

2. Исследование свойств этой диаграммы показывает, что она представима в функторной категории $C^{op} \to S$, полное и непротиворечивое погружение в которую может служить вычислительным каркасом концептуального краудсорсинга.

3. Введенное представление концептуального краудсорсинга позволяет пошаговое распознавание выделенного свойства с последовательным его уточнением группой краудсорсеров.

## СПИСОК ЛИТЕРАТУРЫ:

1. *Вольфенгаген В. Э., Исмаилова Л. Ю., Косиков С. В.* Структура компьютинга и конструирование вычисления // Наука и образование. МГТУ им. Н. Э. Баумана. Электронный журнал. 2010. № 8. URL: http://technomag.edu.ru/doc/153062.html (дата обращения: 15.12.2012).

2. *Вольфенгаген В. Э., Исмаилова Л. Ю., Косиков С. В.* Модель вычислений, чувствительная к семантической нестабильности // Наука и образование. МГТУ им. Н. Э. Баумана. Электронный журнал. 2010. № 12. URL: http://technomag.edu.ru/doc/163548.html (дата обращения: 15.12.2012).

3. *Исмаилова Л. Ю., Косиков С. В., Вольфенгаген В. Э., Зинченко К. Е.* Средства инструментальной поддержки композиции и специализации предметно-ориентированных механизмов наследования для правовых деловых игр // В мире научных открытий. 2010. № 1–4. С. 32–36. URL: http://nkras.ru/vmno/issues/articles/2010/1-4.pdf (дата обращения: 15.12.2012).

4. *Doan A., Ramakrishnan R., Halevy A.* Crowdsourcing systems on the world-wide web // Communications of the ACM. 2011. Vol. 54, No 4. P. 86–96.

5. *Ismailova L. Y., Kosikov S. V., Zinchenko K. E., Mikhailov A. I., Bourmistrova L. V., Berezovskaya A. V.* Equationally. Expressed Evaluation // 9th International Workshop on Functional and Logic Programming. WFLP 2000. Ed. Maria Alpuente. Benicassim, Spain. September 28–30, 2000. P. 135–143.

*В. Э. Вольфенгаген, А. А. Борзяк, А. Н. Долбин, А. С. Доронин, М. Ю. Ермак, Л. Ю. Исмаилова, С. В. Косиков, М. А. Маслов, В. В. Навроцкий, В. Н. Назаров, М. Л. Файбисович*

# СЕМАНТИЧЕСКИ СТАБИЛЬНОЕ ПРЕОБРАЗОВАНИЕ СПОРНЫХ ДАННЫХ ДЛЯ БЕЗОПАСНОЙ РАБОТЫ ИНФОРМАЦИОННОЙ СИСТЕМЫ[1]

Для больших данных исследован вопрос распознавания группой краудсорсеров заданного свойства посредством выделенной распознающей функции для случая ограниченной трансформации этого свойства. Установлены границы семантически стабильного распознавания заданного свойства в условиях клонирования индивидов области его характеризации. Определены условия семантически стабильной работы информационной системы, которые охарактеризованы коммутативной диаграммой.

## Концептуализация и преобразование спорных данных

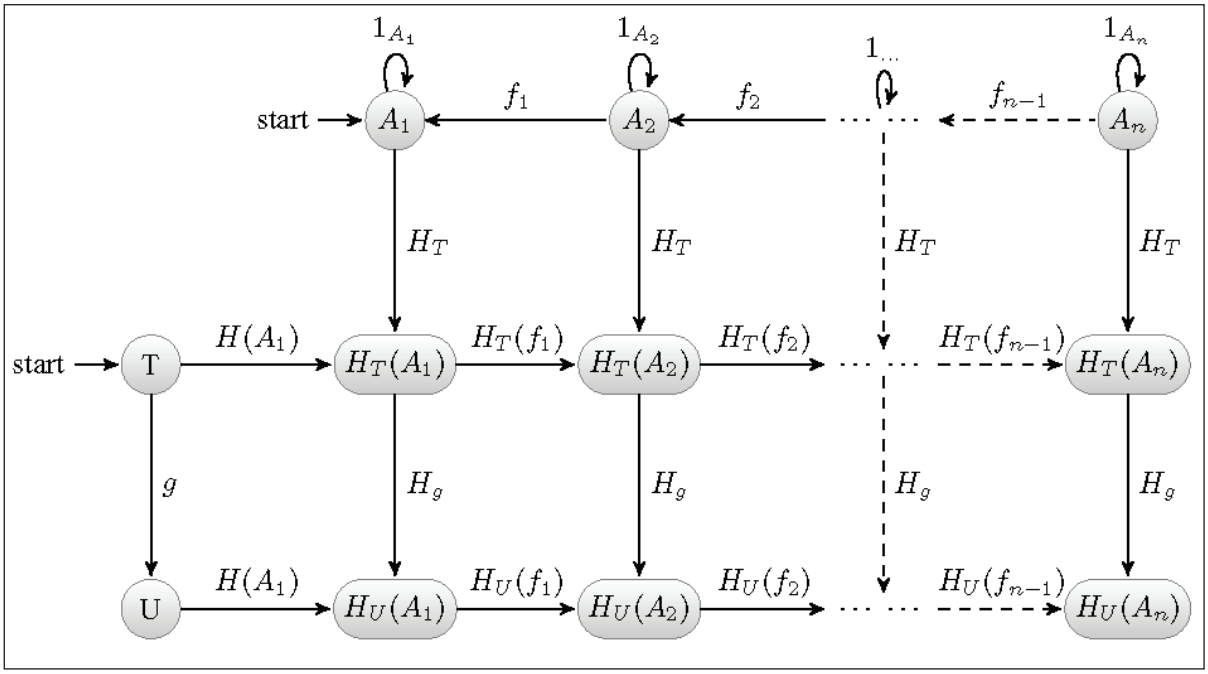
Хотя инструменты анализа данных продолжают улучшаться, аналитики по-прежнему тратят значительное время и усилия, манипулируя данными и оценивая их качество для последующего выполнения их очистки. Представления данных об одном и том же объекте могут различаться как по форме, так и по содержанию, вызывая противоречие, которое называют «спорными данными» (data wrangling). Работа со спорными данными (СД) систематически включает переформатирование значений данных, исправление ошибочных или отсутствующих значений и интегрирование из нескольких источников данных. Отдельные преобразования данных (ПД) часто трудно определимы, их композиции образуют сценарии преобразования, управление которыми является сложной задачей. Повторное использование ПД и/или сценариев ПД на уровне решения задач, командной работы и применения инструментальных средств становится проблематичным или просто невыполнимым. Одной из причин является неоднозначность получения результата ПД. В работе предлагается разработка интерактивной системы для создания преобразований спорных данных и технология преобразований спорных данных с рабочим названием CF (Conceptual Fitter). Система сочетает в себе возможности прямого манипулирования визуализированными данными с автоматизированным выводом соответствующих преобразований, позволяя аналитикам многократно изучать пространство применимых операций и просматривать последствия их выполнения. В качестве контекста проверки и преобразования типов CF на уровне модели вычислений использует систему контекстов-соотнесений, управляемых системой сценариев, в которые группируются разрешенные преобразования и их композиции [1–6]. В частности, соотнесениями могут быть семантические типы данных (например, опубликованный источник, даты, коды классификации и др.). Поддерживается история взаимодействия, где фиксируются отзыв, утончения (выполненная очистка данных) и аннотации сценариев преобразований. Как показывают предварительные эксперименты, применение технологии CF значительно сокращает время очистки данных, и вместо трудоемкого ручного редактирования предоставляется возможность автоматизированного использования надежных, проверяемых сценариев преобразований.

## Схема распознавания варьируемых, или переменных, свойств.

Группе из $n$ краудсорсеров либо экспертов с «состоянием знаний» $A_1$, $A_2$, . . . , $A_n$ соответственно, ставится задача распознать свойство $T$ посредством распознающей функции

[1] Работа является обобщением результатов, которые связаны с построением обобщенной вычислительной модели и получены в разное время при выполнении проектов, частично поддержанных грантами РФФИ 14-07-00119-а, 12-07-00786-а, 14-07-00087-а, 12-07-00702-а, 13-07-00716-а, 12-07-00554-а, 14-07-00107-а, 14-07-31041-мол_а, 13-07-00679-а, 13-07-00705-а. Исследование частично поддержано грантом РНФ 14-11-00816.

$H_T$. Пусть они действуют в силу системы отображений $f_1$, $f_2$, . . . $f_{n-1}$. Рассмотрим случай $g$-трансформации свойства $T$, происходящей по каким-либо причинам в соответствии с функцией $g$: $T \rightarrow U$, которая берется из категории $C$. Работа краудсорсеров отражается следующей коммутативной диаграммой:




На этой диаграмме

$H_T(A_j) = \{h' | h' : A_j \rightarrow T\}$

представляет собой область: множество индивидов $h'$ «в поле зрения» $A_j$, которые обладают свойством $T$. Для функций $g : T \rightarrow U$ и $k : U \rightarrow V$, очевидно, получаем

$H_k \circ H_g = H_{k \circ g}$,

что для категории множеств $S$ определяет

$H : C \rightarrow (C^{op} \rightarrow S)$

как ковариантный функтор между категориями. Конечно, $H_T$ единственным образом определяет $T$, а $H_g$ единственным образом определяет $g$.

Таким образом, $H_g$ может рассматриваться как распознающее отображение для $g$-трансформации свойства $T$. Если $H_T$ — распознающая функция для свойства $T$, а $H_U$ — распознающая функция для свойства $U$, $H_g$ — преобразование распознающей функции $H_T$ в распознающую функцию $H_U$. Это подчеркивает регулярность функторов $H_T$, понимаемых как распознающие функции свойств $T$ с естественными преобразованиями между ними вида $H_g$.

Отображения $g$: $T \rightarrow U$, которые берутся из $C$, представляют переходы между «свойствами» $T$ и «последующими» свойствами $U$. Каждый из таких переходов «ограничивает» элементы в $H_U$ на элементы в $H_T$ «вдоль» отображения $g$.

**Схема алгоритма распознавания концептов для переменных областей**

Распознавание концептов для переменных областей базируется на типовой конструкции переменной области $H_T(A)$. Рассмотрим два основных случая трансформации индивидов, которые назовем *переселением* и *клонированием*.

*Распознавание переселенца*. Если $A$-обитатель $h''$ новой области может быть разложен в виде

$h'' = g \circ h$,

где $h$ — $A$-обитатель старой области, то он является $g$-переселенцем $h^g$:

$h'' = g \circ h \equiv h^g$.

В противном случае он «атомарный» $A$-обитатель новой области, то есть он $A$-рожден в новой области $U$ и ее старожил. Сокращенно будем говорить: $A$-старожил в $U$.

*Распознавание клона*. Клоны возникают в связи с отображениями $f : B \to A$, которые берутся из категории $C^{op}$, и представляют переходы между «стадиями» $A$ и «последующими» стадиями $B$. Если $B$-обитатель $h'$ старой области может быть разложен в виде

$h' = h \circ f$,

где $h$ — $A$-обитатель той же, то есть старой, области, то он является $f$-клоном $h_f$ :

$h' = h \circ f \equiv h_f$.

В противном случае он «атомарный» $B$-обитатель старой области, то есть он $B$-рожден в старой области $T$ и ее старожил. Сокращенно, будем говорить: $B$-старожил в $T$.

Таким образом, потенциально можно различать происхождение индивидов в областях и семантически корректно решать задачи управления данными.

**Заключение**

Представлена коммутативная диаграмма семантически стабильного преобразования спорных данных, которая отражает безопасный режим работы информационной системы.

1. Представлена формальная постановка задачи распознавания группой из $n$ краудсорсеров свойства $T$ посредством распознающей функции $H_T$ для общего случая $g$-трансформации свойства $T$.

2. Проанализированы возможности распознавания заданного свойства в условиях его $g$-трансформации. Проанализированы возможности распознавания заданного свойства в условиях $f$-клонирования индивидов области характеризации распознаваемого свойства.

3. Определены условия семантически стабильной работы информационной системы, которые соответствуют предложенной коммутативной диаграмме.

## СПИСОК ЛИТЕРАТУРЫ:

1. *Wolfengagen V. E.* Applicative computing. Its quarks, atoms and molecules / Edited by Dr. L. Yu. Ismailova. Moscow: Center JurInfoR, 2010. – 62 p.

2. *Вольфенгаген В. Э., Исмаилова Л. Ю., Косиков С. В.* Структура компьютинга и конструирование вычисления // Наука и образование. МГТУ им. Н. Э. Баумана. Электронный журнал. 2010. № 8. URL: http://technomag.edu.ru/doc/153062.html (дата обращения: 15.12.2012).

3. *Вольфенгаген В. Э., Исмаилова Л. Ю., Косиков С. В.* Модель вычислений, чувствительная к семантической нестабильности // Наука и образование. МГТУ им. Н. Э. Баумана. Электронный журнал. 2010. № 12. URL: http://technomag.edu.ru/doc/163548.html (дата обращения: 15.12.2012).

4. *Исмаилова Л. Ю., Косиков С. В., Вольфенгаген В. Э., Зинченко К. Е.* Средства инструментальной поддержки композиции и специализации предметно-ориентированных механизмов наследования для правовых деловых игр // В мире научных открытий. 2010. № 1–4. С. 32–36. URL: http://nkras.ru/vmno/issues/articles/2010/1-4.pdf (дата обращения: 15.12.2012).

5. *Вольфенгаген В. Э., Исмаилова Л. Ю., Косиков С. В., Лаптев А. Д., Назаров В. Н., Рословцев В. В., Сафаров И. С., Степанов А. Л.* Аппликативный компьютинг: попытки установить природу вычислений // Вестник Удмуртского университета. Серия 1: Математика. Механика. Компьютерные науки. Электронный журнал. 2009. Вып. 2. С. 110–117. URL: http://vst.ics.org.ru/uploads/vestnik/2_2009/vu09213.pdf (дата обращения: 15.12.2012).

6. *Вольфенгаген В. Э., Исмаилова Л. Ю., Косиков С. В., Лаптев А. Д., Назаров В. Н., Рословцев В. В., Сафаров И. С., Степанов А. Л.* Комбинаторы: объекты, помогающие понять строение компьютинга // Вестник Удмуртского университета. Серия 1: Математика. Механика. Компьютерные науки. Электронный журнал. 2009. Вып. 2. С. 118–131. URL: http://vst.ics.org.ru/uploads/vestnik/2_2009/vu09214.pdf (дата обращения: 15.12.2012).

*В. С. Горбатов, А. А. Мещеряков, В. Р. Петров*

# ФОРМАЛИЗАЦИЯ РАЦИОНАЛЬНОГО ВЫБОРА СРЕДСТВ ОБЕСПЕЧЕНИЯ ДОВЕРЕННОГО СЕАНСА СВЯЗИ В ДИСТАНЦИОННОМ ОБУЧЕНИИ

В настоящее время активно внедряются технологии дистанционного обучения, позволяющие организовать взаимодействие обучающего и обучаемых между собой на расстоянии посредством сети Интернет. В работе [1] была поставлена задача обеспечения доступных и достоверных способов аутентификации обучаемого в ходе контроля знаний. Для решения этой задачи было предложено использование средств обеспечения доверенного сеанса связи. Таким образом, перед разработчиком системы дистанционного обучения стоит задача обоснования выбора того или иного средства обеспечения доверенного сеанса связи.

Для сравнения средств обеспечения доверенного сеанса связи был выбран метод анализа иерархий, позволяющий в интерактивном режиме найти такой вариант, который наилучшим образом согласуется с пониманием сути проблемы лицом, принимающим решение [2]. Целью сравнения является выбор наилучшего средства обеспечения доверенного сеанса. В качестве критериев, по которым проводилась оценка, были выбраны стоимость, простота администрирования, качество реализации функций безопасности и удобство использования.

В качестве альтернатив были выбраны два зарубежных продукта CheckPoint Abra и IronKey Trusted Access и российский продукт СОДС «МАРШ!».

На рис. 1 представлена иерархия выбора средства обеспечения доверенного сеанса. Путем сравнения критериев были вычислены соответствующие им коэффициенты.

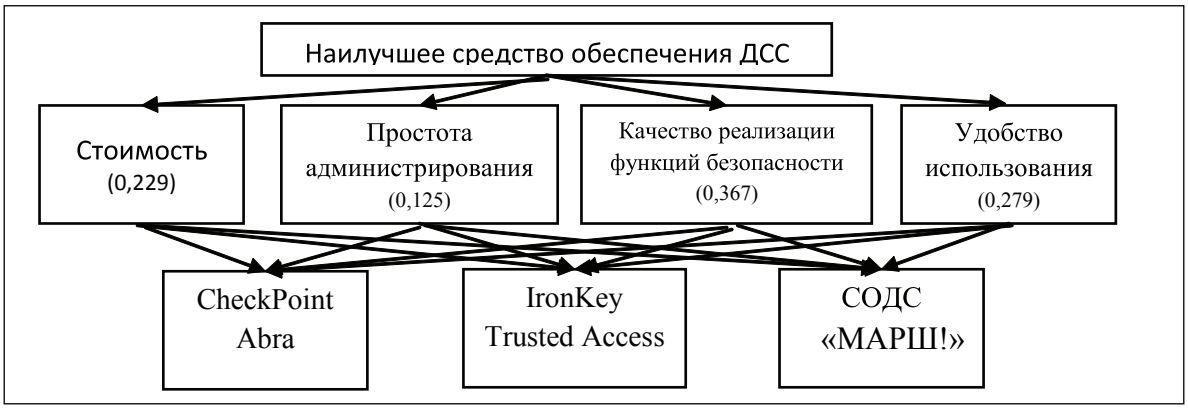



*Рис. 1. Иерархия выбора средств обеспечения доверенного сеанса*

При составлении сравнительной матрицы для всех альтернатив по выбранным критериям использовалась стандартная шкала относительной важности, описанная в [2].

При оценке простоты администрирования учитывалась возможность централизованного администрирования, добавления в доверенную среду дополнительных программ, сложность настройки клиентов и сервера доверенного сеанса связи.

Качество реализации функций безопасности оценивалось по таким критериям, как обеспечение доверенной загрузки, обеспечение выработки и проверки ЭЦП данных, обеспечение идентификации-аутентификации пользователя для доступа к сервисам, обеспечение защищенного соединения с сервером доверенного сеанса связи, наличие сертификата соответствия требований контролирующих государственных органов.

Удобство использования оценивалось с учетом следующих факторов:

- удобство интерфейса, предоставляемого пользователю;
- поддерживаемые виды аутентификации пользователей;

- возможность долговременного хранения файлов;
- совместимость с современным программным и аппаратным обеспечением.

Результатом выполнения метода анализа иерархий является вычисление глобальных приоритетов, которые подсчитываются для каждой из альтернатив. Глобальные приоритеты альтернатив, полученные расчетным путем, приведены в таблице 1.

*Таблица 1. Значение глобальных приоритетов*

| | | Простота | Качество реализации функций безопасности | Удобство | Глобальный приоритет |
|---|---|---|---|---|---|
| Коэффициент/продукт | 0,229 | 0,125 | 0,367 | 0,279 | |
| CheckPoint Abra | 0,249 | 0,432 | 0,238 | 0,413 | 0,313 |
| IronKey Trusted Access | 0,264 | 0,315 | 0,301 | 0,315 | 0,298 |
| СОДС «МАРШ» | 0,487 | 0,253 | 0,461 | 0,272 | 0,388 |

Таким образом, оптимальным средством обеспечения доверенного сеанса связи по нашим критериям является СОДС «МАРШ!», имеющий наибольшее значение глобального приоритета 0,388.

## СПИСОК ЛИТЕРАТУРЫ:

1. *Горбатов В. С., Дворянкин С. В., Дураковский А. П., Петров В. Р.* Постановка задачи по реализации доверенного сеанса связи при дистанционном обучении // Безопасность информационных технологий. 2013. № 3. С. 104–105.
2. *Саати Т.* Принятие решений. Метод анализа иерархий. М.: Радио и связь, 1993. – 279 с.

*А. П. Дураковский, А. И. Зеленова, В. Р. Петров*

## ОБЕСПЕЧЕНИЕ АУТЕНТИЧНЫХ РЕЗУЛЬТАТОВ КОНТРОЛЬНЫХ МЕРОПРИЯТИЙ ПРИ ДИСТАНЦИОННЫХ ОБРАЗОВАТЕЛЬНЫХ ТЕХНОЛОГИЯХ[1]

При использовании дистанционных методов обучения количество обучаемых увеличивается в разы по сравнению с традиционным очным обучением, учет посещаемости обучаемых оказывается все более сложной задачей. А провести дистанционно достоверную аттестацию –

[1] Данная работа выполнена в НИЯУ МИФИ при финансовой поддержке Министерства образования и науки Российской Федерации в рамках проекта «Создание инженерно-технических решений для высокотехнологичного производства инновационных программно-аппаратных средств защиты информации на базе перспективных высокоскоростных интерфейсов информационного взаимодействия», выполняемого совместно с ООО «ОКБ САПР» по договору № 02.G25.31.0050.

задача архиважная и, как становится понятно, архисложная. У дистанционного образования есть несколько особых требований к психологическим особенностям обучаемых. Например, такой вид образования в основном полагается на самостоятельную работу обучаемых и их ответственный подход к получению, отработке полученной информации, самостоятельной подготовке к дистанционному тестированию и, конечно, самостоятельному прохождению тестов. Однако далеко не все обучаемые выполняют тесты самостоятельно. Высока вероятность подставных лиц. Таким образом, при осуществлении контрольных мероприятий системе необходимо провести распознавание пользователя. Сегодня каждый поступивший на обучение в системах дистанционного обучения (СДО) получает входное имя и пароль, чтобы войти на сервер, где находится учебный материал. Кто с ним зарегистрировался во всех используемых СДО, проверить/разоблачить невозможно. Но решать данную проблему необходимо.

Для решения этой проблемы можно использовать два способа. Первый основывается на применении дополнительного аппаратного обеспечения (камеры), второй — на распознавании пользователей с применением дополнительного программного обеспечения [1].

По нашему мнению, при построении СДО необходимо обеспечить защищенность следующих ресурсов:

1) учебно-методические материалы;

2) электронные средства обучения;

3) информация, содержащая персональные данные обучаемых и преподавателей;

4) служебная информация системы управления, финансовых и аналитических структур;

5) информация, передаваемая посредством компьютерных и телекоммуникационных сетей [2].

И, конечно, необходимо обеспечить в СДО защиту от атак типа «отказ в обслуживании».

Далее необходимо определить действия, которые может совершить злоумышленник. Если пользователь не зарегистрирован в системе, то атаки могут быть вида:

1) получение или подбор имени и пароля одного из пользователей системы;

2) получение доступа к учебному курсу;

3) получение доступа и возможность видоизменения статистики обучаемых.

Если пользователь зарегистрирован, то атаки могут быть следующими:

1) подделка результатов тестирования;

2) обман при прохождении самого теста, например, возвращение на уже заданные вопросы;

3) принуждение web-браузера показывать вопросы, но не учитывать ответы на них;

4) выход из браузера во время теста перед выставлением оценки для того, чтобы данная попытка не была засчитана;

5) попытка получения более высоких прав, чем пользователь имеет на самом деле, для перехода, например, к заключительному тестированию;

6) деструктивные действия, направленные на выведение системы из строя или ее «подвисание» путем, например, очистки базы данных или пересылки непредусмотренного параметра через форму;

7) непреднамеренные действия пользователя, которые могут привести к непредсказуемым последствиям.

На наш взгляд, самым действенным способом реализации защищенной СДО будет использование дополнительного аппаратного обеспечения с применением двухфакторной аутентификации. Выбран способ аутентификации — по паролю и биометрическому признаку с использованием программно-аппаратного комплекса СОДС «МАРШ!» [3].

СПИСОК ЛИТЕРАТУРЫ:

1. *Ложников П. С.* Распознавание пользователей в системах дистанционного образования: обзор // Образовательные технологии и общество (Educational Technology & Society). 2001. № 4 (2). С. 211–216.
2. *Дьяченко Ю. А., Милославская Н. Г., Неменков А. В., Толстой А. И., Трофимов Е. А.* Защищенная система дистанционного обучения и тестирования. // Безопасность информационных технологий. 2002. № 4. С. 45–51.
3. *Мельников Д. А., Петров В. Р., Дириенко Е. В., Абрамов А. А., Корсаков И. А., Кузьминов С. С.* Способ парирования атак на систему сетевой синхронизации, реализованный в СОДС «МАРШ!-3.0» // Безопасность информационных технологий. 2013. № 4. С. 71–76.

*В. Л. Евсеев, В. Г. Иваненко, Н. Р. Леонов*

# ТРЕБОВАНИЯ НОРМАТИВНЫХ ДОКУМЕНТОВ В ОБЛАСТИ ОРГАНИЗАЦИИ ВЫСОКОТЕХНОЛОГИЧНОГО ПРОИЗВОДСТВА СРЕДСТВ ЗАЩИТЫ ИНФОРМАЦИИ

Все более широкое применение информационных технологий во всех сферах деятельности человека приводит к настоятельной необходимости организации современного высокотехнологичного производства средств защиты информации [1]. Под высокотехнологичным производством при этом понимается деятельность организации, направленная на создание инновационной продукции при использовании результатов научно-исследовательских и опытно-конструкторских работ, соответствующих приоритетным направлениям науки и техники РФ. К инновационной продукции, безусловно, относятся и средства защиты информации.

Среди многочисленных разнообразных требований в области организации высокотехнологичного производства важнейшими следует считать требования к надежности продукции (средняя наработка на отказ, коэффициент технического использования и др.) и стойкости к внешним воздействиям, поскольку именно эти показатели определяют эффективность последующего применения изготовленной продукции и ее конкурентоспособности, включая и соответствующие экономические показатели. Для привлекательности со стороны потребителей продукции показатели надежности должны быть максимально высокими, однако с учетом стремительного прогресса в области обработки и защиты информации и относительной краткости жизненного цикла соответствующих технических средств средний срок службы средств защиты информации для большинства их применений вполне можно ограничить пятью-шестью годами. Больший устанавливаемый срок службы, вызывая увеличение расходов на качество изготовления ввиду бурного развития как самих информационных технологий, так и методов и средств их защиты, не даст реального улучшения потребительских свойств рассматриваемой высокотехнологичной продукции.

Из основных требований к порядку разработки и изготовления средств защиты информации и их особенностей следует выделить необходимость разработки технических средств от несанкционированного доступа совместно с программными средствами, обеспечивающими их работоспособность в составе защищенных средств вычислительной техники.

Лицензирование деятельности по разработке и производству средств защиты конфиденциальной информации осуществляет Федеральная служба по техническому и экспертному контролю. Основными лицензионными требованиями при этом являются наличие у соискателя

лицензии не менее двух специалистов, имеющих соответствующее профессиональное образование, наличие соответствующего контрольно-измерительного оборудования, а также необходимой технической, технологической и методической документации.

Из многочисленных рекомендаций нормативных документов по обеспечению безопасности при организации и реализации высокотехнологичного производства уместно обратить внимание на применение безотходных технологий замкнутого цикла производства, а если это возможно, то и на своевременное удаление, обезвреживание и захоронение отходов, являющихся источником вредных производственных факторов, а также на использование оборотного водоснабжения [2].

Должную охрану труда при реализации высокотехнологичного производства обеспечивают не в последнюю очередь требования к профессиональному отбору работающих и проверка их знаний. Основными характеристиками работников при их профессиональном отборе являются показатели состояния здоровья и уровень профессиональной подготовки, соответствующие характеру выполняемых работ. Кроме того, работающие должны систематически подвергаться проверке знаний требований безопасности, а при необходимости с ними обязательно проводится соответствующее обучение.

## СПИСОК ЛИТЕРАТУРЫ:

1. Федеральный закон РФ от 27 июля 2006 г. № 149-ФЗ «Об информации, информационных технологиях и о защите информации».
2. ГОСТ 12.3.002-75 «Процессы производственные. Общие требования безопасности».

*Е. В. Елистратова*

## МЕТОДИЧЕСКИЙ ПОДХОД ДЛЯ ЗАЩИТЫ ДАННЫХ, ХРАНИМЫХ И ОБРАБАТЫВАЕМЫХ НА ОНЛАЙН-СЕРВИСАХ

В настоящее время онлайн-сервисы предоставляют широкий спектр возможностей для обмена информацией (размещения фотографий, видео-/аудио- и текстовых данных, организации тематических сообществ, обмена личными сообщениями и т. п.) и используются пользователями в качестве хранилищ данных и платформ для социальных контактов [1. С. 162–182]. Учитывая сказанное, важной задачей является защита хранимых и обрабатываемых данных.

Основным средством защиты данных, применяемых в настоящее время для решения этой задачи, является использование внутренних механизмов, предоставляемых самим онлайн-сервисом [2]. Однако вследствие массового использования указанные механизмы часто подвергаются атакам и тем самым не являются надежным средством защиты данных.

В работе представлен разработанный автором методический подход для повышения защищенности данных, хранимых и обрабатываемых на онлайн-сервисах. Подход основан на разработке дополнительных компонентов, реализующих шифрование информации на стороне

пользователя и ее передачу (в зашифрованном виде) на сервера для последующего хранения, и включает:

1) исследование архитектуры сервиса, особенностей хранения и передачи данных;

2) выбор алгоритма шифрования данных;

3) разработку компонентов, осуществляющих шифрование данных в соответствии с выбранным алгоритмом;

4) разработку компонентов, обеспечивающих передачу данных на сервера;

5) разработку пользовательского интерфейса.

Практическая апробация разработанного подхода и реализующих его программных средств подтвердила их пригодность для повышения защищенности данных, хранимых и обрабатываемых на онлайн-сервисах. Установлено, что использование шифрования данных на стороне пользователя является надежным средством защиты.

## СПИСОК ЛИТЕРАТУРЫ:

1. *Гусиос Г., Спинеллис Д.* Идеальная архитектура. Ведущие специалисты о красоте программных архитектур. М.: Символ-Плюс, 2010.
2. *Russell J., Cohn R.* Facebook Platform. M.: Книга по требованию, 2012.
3. *Russell J., Cohn R.* Facebook, Inc. M.: Книга по требованию, 2012.

*Д. В. Ефанов, К. Г. Григорьев, П. Г. Рощин*

## НАПРАВЛЕНИЯ ИССЛЕДОВАНИЯ ТЕХНОЛОГИИ SELINUX

На протяжении нескольких десятков лет разработчиками операционных систем (ОС) ведется работа по проектированию и реализации в ОС механизмов мандатного управления доступом. Исторически сложилось так, что под мандатным управлением чаще всего подразумевают программную реализацию модели Белла—Лападулы. Это вызвано тем, что модель Белла—Лападулы была взята за основу при разработке стандарта «Критерии определения безопасности компьютерных систем» Минобороны США в середине 1980-х годов, а затем, в начале 1990-х годов, использовалась в отечественных Руководящих документах Государственной технической комиссии при Президенте РФ.

В соответствии с данными документами были разработаны и сертифицированы десятки ОС для разных аппаратных платформ и областей применения. Наиболее успешной отечественной разработкой является ОС МСВС 3.0, созданная по заказу Минобороны России для работы на ряде отечественных компьютеров и сертифицированная в системе сертификации средств защиты информации Минобороны России в 2001 г.

Однако данные ОС были ориентированы на обработку информации, содержащей государственную тайну, поэтому применение реализованных в них мандатных механизмов в ОС общего назначения не имело смысла.

В 2001 г. для ядра Linux была разработана система мандатного управления доступом SELinux, которая позже официально вошла в исходный код ядра версии 2.6. В SELinux была реализована модель принудительной типизации, которая позволяла существенно приблизиться к реализации принципа «наименьших привилегий» и минимизировать ущерб от атак нулевого дня.

Технология SELinux является примером успешной реализации принципов мандатного управления доступом в операционной системе общего назначения. Основным продуктом, использующим SELinux, является дистрибутив Linux корпоративного уровня компании Red Hat.

Очевидно, что для подготовки бакалавров и магистров по направлениям «Информатика и вычислительная техника», «Прикладная математика и информатика» и другим, связанным с разработкой и эксплуатацией системного программного обеспечения, необходимо изучение современных технологий защиты информации, к которым, безусловно, относится SELinux. В постановке курсов, связанных с изучением SELinux, можно выделить следующие направления: изучение администрирования информационных систем, построенных с применением технологии SELinux, разработка приложений, использующих программный интерфейс SELinux, а также разработка политик безопасности SELinux.

К научной работе можно отнести такие направления, как анализ, верификация и моделирование политики безопасности SELinux, а также повышение производительности проверок доступа.

В настоящий момент рассматриваются два направления в анализе политик безопасности: эвристический анализ непосредственно базы данных политики и построение графа на основе этой политики. Первое направление уже используется в SELinux, есть несколько утилит, но проблема анализа базы данных в том, что она содержит слишком много правил и провести какой-либо результативный анализ не представляется возможным. Второе направление связано с построением ориентированного графа на срезе вектора доступа (например, чтение файла, запись в файл) с последующим проведением его анализа. В данном случае политика должна быть оптимизирована, так как в настоящий момент она содержит очень много похожих правил. В случае графа можно не только анализировать информационные потоки политики, но и проводить своего рода кластеризацию для поиска слабых мест в исходном коде политики.

К первым результатам данного исследования относится обнаружение множества однотипных правил, что в целом подтверждает идею об активном использовании атрибутов в политиках.

Изучение технологии SELinux ведется на 36-й кафедре НИЯУ МИФИ в рамках курса «Операционные системы». Результаты научных исследований используются в ряде отечественных продуктов специального назначения.

*М. Р. Закиров, И. В. Машкина*

## ПРОБЛЕМЫ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ СИСТЕМ ДИСТАНЦИОННОГО БАНКОВСКОГО ОБСЛУЖИВАНИЯ

В последнее время банки все активнее предлагают своим клиентам различные услуги дистанционного банковского обслуживания (ДБО). Под термином ДБО понимают технологии предоставления банковских услуг на основании распоряжений, передаваемых клиентом удаленно (без визита

в банк). Бурное развитие информационных технологий, Интернета, мобильной связи делают перспективы развития услуг ДБО очень высокими. К тому же в условиях развития банковской системы страны, повышения финансовой и технической грамотности населения спрос на услуги ДБО неуклонно растет. По прогнозам крупных игроков рынка, прирост активной аудитории в ДБО может составлять в ближайшем будущем около 40–50 % в год.

Вместе с тем наряду с ростом популярности ДБО растет и число хищений, являющихся результатом мошенничества в данных системах. По информации аналитического центра компании «Техносерв» в 2012 г. сумма прямого ущерба клиентам ДБО в РФ составила 95 млн долларов [1]. Возможность реализации угроз информационной безопасности в системах ДБО неразрывно связана с концептуальными проблемами, сопутствующими технологиям удаленного управления банковским счетом. Анализ существующих систем ДБО позволил выявить следующие проблемы:

- несовершенство нормативной базы;
- проблема обнаруженных уязвимостей;
- проблема взаимодействия;
- проблемы клиентской части ДБО;
- проблемы правоохранительной системы.

Несовершенство нормативной базы выражается в недостатках существующего уголовно-процессуального кодекса. Отличительной особенностью преступлений в сфере высоких технологий является их широкая география. Пострадавший может находиться в одном субъекте РФ, получатель мошеннического платежа — в другом, а организатор и вовсе за границей. При этом уголовное дело заводится по месту окончания преступления, что не всегда приемлемо. Также банки не могут в соответствии с Гражданским кодексом «учинять препятствия» при исполнении транзакций, что накладывает ограничения по возможности остановки мошеннического платежного поручения.

Проблема обнаруженных уязвимостей связана с нежеланием разработчиков оперативно реагировать на сообщения об обнаруженных уязвимостях в продуктах ДБО. Также следует отметить, что к разработчикам не предъявляются требования предоставлять результаты анализа кода системы ДБО на наличие недекларированных возможностей. В данном направлении активно работает Банк России, и, вероятнее всего, в ближайшем будущем разработчиков обяжут предоставлять результаты анализа кода.

Проблема взаимодействия связана с препятствиями в организации совместной деятельности банков и правоохранительных органов по борьбе с мошенничеством. Также существует проблема взаимодействия правоохранительных органов разных стран.

Проблемы клиентской части ДБО связаны с возможным низким уровнем компьютерной грамотности клиента. Система ДБО по умолчанию работает в недоверенной среде. Даже такой, казалось бы, обязательный атрибут системы ДБО, как лицензионный антивирус с обновляемыми базами, клиент использует редко.

Существует огромный пласт проблем, связанный с правоохранительной системой. Во-первых, низкий уровень квалификации сотрудников, особенно в регионах. Во-вторых, преступления в сфере информационных технологий не рассматриваются в особом порядке, процесс возбуждения дела может растягиваться во времени до трех дней, что неприемлемо долго.

Привычные для ДБО средства защиты, такие как логин/пароль, одноразовый пароль, электронная подпись, SMS-информирование, носители с неизвлекаемым ключом, в нынешних реалиях не способны эффективно защищать от организованных преступных группировок, в которых давно существует разделение труда и четкое разграничение ролей.

Одним из технических аспектов решения проблемы мошенничества в ДБО является использование специализированных систем борьбы с мошенничествами (ССБМ) в системах ДБО (так называемая антифрод-система от англ. fraud — мошенничество).

Наиболее сложные ССБМ в системах ДБО — комплексные антифрод-системы — разрабатываются, как правило, специализированными компаниями. Анализ имеющихся на рынке предложений комплексных систем показал, что при разработке таких систем учитываются статистические данные, на основе которых могут строиться модели развития бизнеса. Данные системы анализируют множество критериев, связанных между собой весовыми коэффициентами либо логически. Работа подобной системы заключается в анализе логической взаимосвязи критериев.

Однако данные системы имеют и свои недостатки. Для них характерны ошибки первого и второго рода. Кроме того, данные ССБМ в системах ДБО, как правило, очень дороги. Хотя проблемы с большим количеством ложных срабатываний и обходами системы успешно решаются, но в случае превышения количества 10000 транзакций в сутки, эти системы теряют свою эффективность.

На рынке представлены интеллектуальные ССБМ в системах ДБО. Данные системы обладают исключительной сложностью реализации и способны:

- анализировать все основные параметры и поля платежного поручения, будь то предельная сумма транзакции, ИНН получателя платежного поручения, сетевые параметры отправителя (MAC и IP-адреса), назначение платежа, регион банка получателя и т. п.;
- анализировать дополнительные критерии на стороне клиента, например наличие антивируса и «свежих» вирусных баз;
- получать дополнительную информацию из внешних источников — базы данных, Интернета и т. п.;
- выявлять попытки обхода ССБМ, например попытки маскировки получателя;
- самообучаться и пополнять свою базу знаний на основе анализа работы каждого клиента. В результате создается возможность построения моделей ведения бизнеса по каждому клиенту.

Таким образом, в интеллектуальных ССБМ число ложных срабатываний можно снизить в 3–4 раза. Несмотря на высокую эффективность интеллектуальных самообучающихся систем, высокая цена и сложность внедрения, высокая стоимость поддержки, необходимость в штатной единице для сопровождения таких систем могут оттолкнуть потенциального покупателя подобной системы. Кроме того, алгоритмы работы, исходный код неизвестны даже банкам — пользователям данных систем.

Итак, необходимо отметить, что практика свидетельствует о необходимости использования интеллектуальных самообучающихся специализированных систем борьбы с мошенничествами в системах ДБО, разработки методик принятия решения при обнаружении мошеннических транзакций.

## СПИСОК ЛИТЕРАТУРЫ:

1. *Царев Е., Хафизов А.* Ущерб от мошенничества растет с каждым годом. URL: http://www.cnews.ru/reviews/index.shtml?2013/02/05/517975 (дата обращения: 14.01.2014).

*С. В. Зыков, А. В. Кукушкин*

# РАСПРЕДЕЛЕНИЕ ПРАВ ДОСТУПА К ПЛАНОВОЙ ДОКУМЕНТАЦИИ ПРОЕКТА

Политика распределения прав доступа участников крупного проекта к конфиденциальной плановой документации должна соответствовать взаимопротиворечивым целям. С одной стороны, следует обеспечить лиц, принимающих решения, необходимой и достаточной информацией в их области ответственности и, с другой стороны, не допустить утечки данных тем участникам, которые не имеют доступа к ним. Существующие подходы к управлению проектами [1—4] предполагают полную взаимозависимость всех аспектов проекта между собой, что в общем случае является верным, но на практике зачастую приводит к предоставлению избыточного доступа к управленческой документации, в том числе конфиденциальной.

В работе предлагается построение модели данных об управлении проектом на уровне его структурных моделей (структура разбиения работ, организационная структура и т. д.) с использованием ER-нотации, применяемой при проектировании баз данных [5, 6]. Построенная модель дополняется логическими и функциональными связями, соответствующими различным методам управления проектами (метод критического пути, экспертные оценки и т. д. [1, 2]), и может рассматриваться как семантическая сеть, которая описывает отношения между ее элементами [7].

Каждая допустимая комбинация входящих дуг определяет для отдельной вершины сети возможные постановки задачи планирования или актуализации соответствующей структурной модели проекта. Например, исходными данными задачи построения расписания проекта являются сетевой график и продолжительность работ, а для учета ресурсных ограничений также требуются сведения о бюджете, графике наличия других ресурсов и их назначениях. Такая постановка задачи может быть изображена в виде функционального блока бизнес-процесса в нотации IDEF0 [8], где входы определяют потребность исполнителя в информации, а выходы — формируемые им сведения.

На основе исходных данных об ограничениях, приоритетных показателях или изменениях в проекте на семантической сети производится построение бизнес-процессов планирования или управления изменениями. Входные и выходные данные отдельных функциональных блоков полученного бизнес-процесса определяют интерфейсы взаимодействия между участниками и рамки их доступа к необходимой информации, которые могут быть выражены в виде единой матрицы прав доступа.

Для автоматизации процедуры возможно использование метода SLD-резолюции, применяемого в логическом программировании [9], который является алгоритмом обхода дерева решений для построения доказательства целевого утверждения на основе исходных посылок и набора правил. В этом случае исходными данными являются сведения об ограничениях, приоритетах либо изменениях в проекте, а набором правил — множество возможных постановок задач определения или актуализации структурных моделей проекта.

Предложенный подход позволяет повысить уровень информационной безопасности проекта и задействованных в нем организаций за счет направленной и более прозрачной политики предоставления доступа к конфиденциальной плановой документации.

## СПИСОК ЛИТЕРАТУРЫ:

1. Управление проектами: Основы профессиональных знаний, Национальные требования к компетентности специалистов. М.: ЗАО «Проектная ПРАКТИКА», 2010.

2. A Guide to the Project Management Body of Knowledge (PMBOK® Guide). Fifth Edition. USA: Project Management Institute, 2013.
3. Managing successful projects with Prince2. London, UK: TSO, 2005.
4. *Ohara S.* P2M – A Guidebook of Project & Program Management for Enterprise Innovation. Vol. 2. Rev. 3. Project Management Association of Japan. 2005.
5. *Дейт К. Дж.* Введение в системы баз данных. 8-е изд. Пер. с англ. М.: Издательский дом «Вильямс», 2005.
6. *Кукушкин А. В.* Принципы динамического моделирования бизнес-процессов управления проектом (конспект доклада) // Российский экономический конгресс. Сборник докладов участников. М.: ИЭ РАН. 2009. (http://www.econorus.org/consp/files/m8yq.doc, ссылка актуальна на 24.03.14)
7. *Roussopoulos N. D.* A semantic network model of data bases. TR № 104. Department of Computer Science. University of Toronto. 1976.
8. *Ойхман Е. Г., Попов Э. В.* Реинжиниринг бизнеса: реинжиниринг организаций и информационные технологии. М.: Финансы и статистика, 1997.
9. *Сошников Д. В.* Парадигма логического программирования. М.: Вузовская книга, 2006.

*М. А. Куприяшин, Г. И. Борзунов*

# СОКРАЩЕНИЕ ВРЕМЕННОЙ СЛОЖНОСТИ БАЗОВОГО АЛГОРИТМА РЕШЕНИЯ ЗАДАЧИ О РЮКЗАКЕ

Несмотря на то что многие из предложенных алгоритмов решения задачи о рюкзаке оказались нестойкими или не применимыми на практике из-за большого объема необходимых вычислений [1], в настоящее время продолжаются исследования, направленные на усовершенствование этих алгоритмов (см., например: [2]). Как показано в работе [3], в этих исследованиях важную роль играют базовые алгоритмы, что делает актуальным их анализ и совершенствование.

Пусть существует набор вещей, каждая из которых имеет определенный положительный вес. Требуется найти способ укладки части этих вещей в рюкзак таким образом, чтобы вес рюкзака равнялся наперед заданному значению $\omega$. Упорядоченное множество весов вещей называется рюкзачным вектором $\vec{a} = (a_1; a_2; \dots a_n)$. Далее можно задать бинарный вектор $\vec{v}$ длины $n$ таким образом, что при $v_i = 1$ предмет с номером $i$ помещается в рюкзак, а при $v_i = 0$ этот предмет в состав рюкзака не входит. Используя данную систему обозначений, можно сформулировать задачу о рюкзаке: найти все возможные значения вектора $\vec{v}$, при которых выполняется: $\sum_{i=1}^{n} a_i v_i = \omega$.

Задача о рюкзаке (в общем случае) считается NP-полной [2], то есть все известные алгоритмы ее решения имеют экспоненциальную временную сложность. В частности, в случае конструктивного перечисления всех возможных значений вектора $\vec{v}$ с проверкой каждого из них требуется анализ $2^n$ значений этого вектора. Проверка каждого потребует $n$ операций умножения, $n-1$ операций сложения и 1 операцию сравнения (для проверки совпадения полученного веса со значением $\omega$). Таким образом, временная сложность составляет $T = 2^n \cdot 2n$ операций.

Анализ значений вектора $\vec{v}$ на принадлежность множеству возможных решений позволяет сократить сложность решения задачи: если для вектора $\vec{v}_k$ суммарный вес набора предметов равен или превосходит $\omega$, то можно не рассматривать векторы большего веса $\vec{v}_z$, для которых выполнено: $\vec{v}_{k_i} = 1 \Rightarrow \vec{v}_{z_i} = 1; i = \overline{1,n}$. Определим граф $G(\vec{v})$, в качестве вершин которого выступают все возможные значения вектора $\vec{v}$. Пусть каждая вершина размещается на ярусе, номер которого совпадает с числом единичных координат в соответствующем значении вектора $\vec{v}$. Ребра соединяют вершины $\vec{v}_k$ и $\vec{v}_z$ тогда и только тогда, когда вершина $\vec{v}_k$ размещена на некотором

ярусе $i$, а вершина $\vec{v}_z$ — на следующем ярусе $i+1$, и указанные векторы различаются только одной координатой, следующей за последней единичной координатой вектора $\vec{v}_k$: в векторе $\vec{v}_k$ значение этой координаты равно 0, а в векторе $\vec{v}_z$ равно 1. Например, между вершинами 1010 и 1011 имеется ребро, а между 1010 и 1110 — нет. Вершина 1000 связана с 1100, 1010 и 1001. Граф $G(\vec{v})$ при $n=4$ представлен на рис.1.

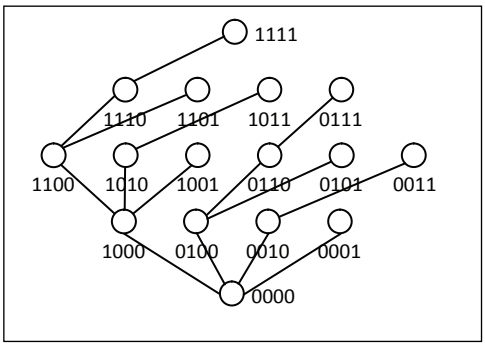



*Рис. 1. Граф $G(\vec{v})$ для вектора $\vec{v}$ размерности 4*

При движении вдоль цепи, начинающейся в нулевой вершине, вес рюкзака может только увеличиваться. Поэтому если для определенной вершины вес рюкзака превысит ω, то можно утверждать, что в продолжении данной цепи решений нет. Это позволяет сократить количество рассматриваемых при переборе решений путем обхода графа $G(\vec{v})$ в глубину из вершины 00…0 и отсеивания всех вершин, соединенных цепью с вершинами меньшего яруса, которым соответствует рюкзак весом более ω. Пример отсечения ветвей перебора при ω=*12*, $\vec{a}=(9;7;5;3)$ представлен на рис. 2.

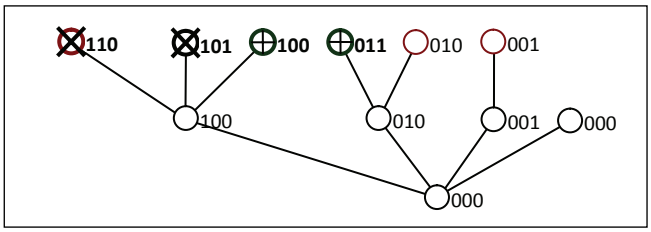



*Рис. 2. Пример отсечения ветвей перебора при ω=12; $\vec{a}=(9;7;5;3)$*

Знаком «X» обозначены вершины, в которых вес рюкзака превысил ω. Знаком «+» обозначены вершины, оказавшиеся решениями. Всего потребовалось обойти 11 вершин из 16. Таким образом, временная сложность решения задачи о рюкзаке понижается на 31 % по сравнению с методом полного перебора, что соответствует коэффициенту ускорения 1,45. Следует отметить, что эта величина ускорения существенно зависит от весов предметов и значения ω. Результаты исследования этой зависимости будут представлены в следующих публикациях.

## СПИСОК ЛИТЕРАТУРЫ:

1. *Шнайер Б.* Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си. М.: Триумф, 2003. – 816 с.
2. *Мурин Д. М.* Компьютерно-аналитическое исследование задач рюкзачного типа как средство анализа и совершенствования систем защиты информации. АКД. Томск: НИТГУ, 2013. – 22 с.
3. *Борзунов Г. И., Войнов А. Е., Сучкова Е. А.* Выбор базового алгоритма для расчета минимального количества процессоров, обеспечивающего достижение заданного значения коэффициента ускорения // Безопасность информационных технологий. 2010. № 1. С. 45–46.

*С. К. Марфенко, В. О. Чуканов*

# ОЦЕНКА УСТОЙЧИВОСТИ И НАДЕЖНОСТИ СИСТЕМ НА БАЗЕ ИНФРАСТРУКТУРЫ JAVA EE

Клиент-серверные приложения, предполагающие одновременную работу большого количества пользователей, как правило, уязвимы к DoS-атакам. Для того чтобы защититься от такого рода угроз, требуется тщательно контролировать тот трафик, который приходит к серверной части приложения, и ограничивать систему от потенциально опасных для нее запросов. В этом случае необходимо получить количественные оценки характеристик надежности и устойчивости программного обеспечения при разной нагрузке со стороны пользователей. Благодаря этой информации можно будет довольно точно подобрать ограничения для входящего трафика определенного приложения, работающего на конкретной аппаратной платформе. Отсюда возникает необходимость в разработке модели надежности, которая может быть использована для оценки соответствующих параметров.

Практически все существующие модели используют данные, полученные в процессе тестирования [1]. Очевидно, что для работы модели, учитывающей показатели нагрузки, потребуется не только функциональное, но и нагрузочное тестирование, которое позволит оценить поведение системы в условиях, приближенных к реальным. Дополнительная трудность в тестировании заключается в том, что архитектура такого рода систем довольно сложна. Например, при развертывании приложения в среде Java EE в обработке запроса могут участвовать веб-сервер, сервер приложений, база данных, система каталогов и т. д. С большой вероятностью эти отдельные приложения будут выполняться на различных физических узлах. Оценивая надежность этого комплекса, мы должны учитывать надежность работы каждого из компонентов, надежность интеграционной логики, связывающей их, надежность сетевой инфраструктуры, которая используется в этом комплексе.

Очевидно, что при разработке сложной системы некоторая часть функционала этого приложения будет представлять собой статическое содержимое, которое можно развернуть на веб-сервере и для получения которого не требуется задействовать сервер приложений и базу данных. Вероятно, что не каждый запрос, приходящий на сервер приложений, требует взаимодействия с базой данных. Нужно определить, для какой части запросов необходимо взаимодействие с сервером приложений и базой данных. Дать соответствующую оценку можно, исходя из предполагаемых паттернов использования приложения. Нужно промоделировать поведение типичного пользователя системы и на основе этого моделирования дать соответствующие оценки:

$$N > M > L,$$

где $N$, $M$, $L$ — доли от общего числа запросов, которые требуют работы веб-сервера, сервера приложений и базы данных соответственно. Исходя из рассматриваемой архитектуры $N = 1$.

С учетом этих оценок вероятность того, что для обработки очередного запроса будет задействован сервер приложений, равна $R_{12} = M \div N = M$. Вероятность того, что для обработки очередного запроса, пришедшего на сервер приложений, потребуется база данных, равна $R_{23} = L \div M$.

Определяя паттерн поведения пользователей системы, следует помнить, что это модель, а в реальной среде использования логика работы пользователей может отличаться от предполагаемой. Поэтому при определении данного паттерна лучше рассматривать некий пессимистический сценарий, с большей нагрузкой на сервер приложений и базу данных. Также следует помнить о том, что потенциально для всех запросов, приходящих в систему, могут потребоваться все ее компоненты. Это может быть связано, во-первых, с соответствующей архитектурой приложения, во-вторых, с возможной DoS-атакой на систему, в которой злоумышленник подберет такой запрос

для приложения, обработка которого потребует максимального количества ресурсов, для выведения системы из строя.

Далее мы должны провести функциональное тестирование системы и оценить надежность каждого из компонентов. Целесообразно использовать модель, которая позволит дать оценку для вероятности безотказной работы системы, например модель Нельсона [2]. По итогам тестирования будут получены значения для надежности каждого из компонентов системы:

- $\rho_{10}$, $\rho_{20}$, $\rho_{30}$ — вероятность безотказной работы веб-сервера, сервера приложений и базы данных соответственно;

- $\rho_{120}$, $\rho_{230}$ — вероятность безотказной работы плагина веб-сервера для взаимодействия с сервером приложений и менеджера соединений для взаимодействия с базой данных соответственно.

Далее проводится нагрузочное тестирование для прогнозируемого количества пользователей, по результатам которого нужно подобрать ключевые настройки для системы.

Когда будут подобраны оптимальные настройки, можно приступать к собственно стрессовому тестированию, в процессе которого будет последовательно увеличиваться нагрузка на систему и будут выясняться точки насыщения для каждого из компонентов системы: $N_1$, $N_2$, $N_3$, $N_{12}$, $N_{23}$.

После оценки значений $N_i$ необходимо проведение повторного стрессового тестирования, в процессе которого мы будем моделировать нагрузку больше найденных величин для того, чтобы оценить поведение системы. Для каждого i-го компонента системы необходимо определить зависимость между количеством пользователей и статистической надежностью. Для любой точки n надежность можно оценить по следующей формуле:

$$p_{i(n)} = p_{i0}\left(1 - \frac{f}{n}\right),$$

где $f$ — количество отказов при нагрузке в $n$ пользователей, $\rho_i$ — надежность компонента, предварительно установленная по результатам функционального тестирования.

Тестирование проводится для нескольких контрольных точек $n$, лежащих за точкой насыщения $N_i$. Каждое стрессовое тестирование проводится несколько раз, и для каждого из прогонов определяется величина $f$.

$$m = \frac{\Sigma_i f_i}{t},$$

где $t$ — количество тестовых прогонов для точки $n$.

Далее можно вынести полученные точки на координатную сетку (количество пользователей — по оси абсцисс, количество отказов — по оси ординат). По определенным точкам можно построить график условной функции. Очевидно, что функция имеет смысл только при целочисленных значениях аргумента. Однако в дальнейшем эта условная кривая может быть использована для оценки вероятности безотказной работы для любого $n$.

По завершении тестирования каждого из компонентов необходимо сделать оценку для вероятности безотказной работы всей системы:

$$P(n) = C_1 p_1(n) + C_2 p_1(n) p_{12}(n) p_2(n) + C_3 p_1(n) p_{12}(n) p_2(n) p_{23}(n) p_3(n).$$

С учетом вышеприведенных формул можно сделать оценку для вероятности безотказной работы с использованием ранее промоделированных зависимостей между количеством тестовых прогонов и количеством отказов, а также предварительных оценок для $R_{12}$ и $R_{23}$.

Новизна предлагаемого подхода заключается в новом дополнительном факторе, влияющем на надежность программного обеспечения, работающего по клиент-серверной архитектуре. Оценка зависимости вероятности безотказной работы системы от количества одновременно работающих пользователей дает возможность применить классический аппарат моделей надежности программного

обеспечения для современных программно-аппаратных комплексов (например, для инфраструктуры на основе Java EE). Это необходимо для того, чтобы оценить надежность и устойчивость программного обеспечения в реальных условиях его эксплуатации. Также с помощью предлагаемой методологии можно давать точные оценки изменения характеристик приложения при потенциальном увеличении количества пользователей. Дальнейшее развитие методологии видится в формализации каждого отдельного шага для реальной платформы Java EE и автоматизации отдельных шагов.

## СПИСОК ЛИТЕРАТУРЫ:

1. *Чуканов В. О.* Надежность программного обеспечения и аппаратных средств систем передачи данных атомных электростанций: Учебное пособие. М.: МИФИ, 2008. – 180 с.
2. *Майерс Г.* Надежность программного обеспечения. М.: Мир, 1980. – 350 с.

*В. Р. Петров, Е. А. Щукин*

## АКТУАЛЬНОСТЬ СОЗДАНИЯ АЛГОРИТМА ПОВСЕМЕСТНОГО КОНТРОЛЯ СЛУШАТЕЛЕЙ ПРИ ДИСТАНЦИОННЫХ ОБРАЗОВАТЕЛЬНЫХ ТЕХНОЛОГИЯХ

В современной России все популярнее становится дистанционное обучение. Это экономит время и дает возможность обучаться где угодно без отрыва от места работы и в удобное для каждого время.

Донести необходимый учебный материал уже не составляет труда. Информационно-телекоммуникационные технологии давно вышли на уровень, позволяющий легко изучать любой, даже специфичный, материал дистанционно. Проверка того, хорошо ли усвоен материал, также не вызывает вопросов. Можно подобрать удобный метод проверки для материала в целом или его отдельных частей. Но территориальная удаленность и, как следствие, отсутствие очного общения накладывают свой отпечаток на весь процесс. В частности, в подавляющем большинстве случаев никак не контролируется, кем именно выполняются задания.

Одним из вариантов организации такого контроля является использование биометрической идентификации пользователей.

Международная группа биометрии (International Biometrics Group – IBG) определяет понятие биометрии как «автоматическое использование физиологических или поведенческих характеристик для установления или подтверждения идентичности» [1]. Не углубляясь в технические детали, можно сказать, что биометрическая идентификация – это метод регистрации человека по его неизменным биологическим параметрам с целью последующего узнавания.

Существует множество методов биометрической идентификации. Для того чтобы выбрать оптимальный, приведем их краткую характеристику и оценим, насколько каждый из них применим в системе дистанционного обучения. Заведомо сложные в воплощении в жизнь варианты опустим.

В первую очередь следует разделить методы на те, которые требуют наличия дополнительного аппаратного обеспечения, и не требующие такового. Первые, соответственно, менее удобны в

системе дистанционного обучения потому, что потенциальному слушателю будут выдвинуты более жесткие требования (например, обязательное наличие веб-камеры). Программные методы, которые не требуют дополнительного аппаратного обеспечения, в этом отношении проще. Все, что необходимо, — это внедрить разработанный алгоритм в уже существующую систему.

Первое, что приходит на ум при упоминании о биометрической идентификации, — это распознавание личности по отпечаткам пальцев. В большинстве ноутбуков уже есть встроенные считыватели. Проблема заключается в том, что их точность невысока. Следовательно, может не пройти идентификацию тот, кто должен. А более точные аппараты приобретаются отдельно и стоят довольно дорого. Учитывая, что возраст слушателей не ограничен, а тонкости работы со считывателем не очевидны, использование такого метода может принести больше проблем, чем пользы.

Еще один вариант — анализ формы лица. Для него необходима веб-камера. Она давно уже не является чем-то новым и часто встроена в ноутбук. Этот метод основан на построении двух- или трехмерной модели лица человека и выделении на нем контрольных точек. Привлекательность данного метода в том, что он наиболее близок к тому, как мы идентифицируем друг друга.

Идентификацию по сетчатке глаза, геометрии руки, венозному рисунку и несколько других будет крайне затруднительно применять в дистанционном обучении из-за того, что для них требуется дорогостоящее оборудование, которое к тому же нельзя купить в соседнем магазине электроники.

Теперь рассмотрим методы, в основу которых положены алгоритмы, не требующие при этом дополнительного аппаратного обеспечения. Это клавиатурный почерк и динамика работы с мышью. Последний основывается на анализе координации движений, времени реакции на событие, скорости и точности манипулирования мышью, особенностей траектории указателя. Сложность внедрения такого типа идентификации в системы дистанционного обучения состоит в том, что для каждой системы, а возможно, и для каждого курса системы придется прорабатывать детали алгоритма, ведь анализ основывается на действиях пользователя в процессе обучения, а каждый курс может быть спроектирован по-своему.

Идентификация по клавиатурному почерку заимствована из области телеграфии, когда при передаче информации кодом Морзе заметили, что каждый оператор имеет свой почерк передачи сигналов. С клавиатурным почерком даже проще. У каждого свой стиль. Он считывается на тестовом тексте и заносится в матрицу. Она и служит впоследствии эталоном для сверки. Вы просто печатаете текст, а вас проверяют. Преимущество данного метода в том, что его внедрение в уже существующие системы не потребует больших трудозатрат. Необходимо лишь немного изменить поля для ввода текстовых ответов.

На основе выявленных свойств хотелось бы выделить в общем перечне два наиболее удобных, на мой взгляд, метода биометрической идентификации для внедрения в системы дистанционного обучения. Это идентификация по геометрии лица и по клавиатурному почерку. Неоспоримый плюс первого в том, что для подтверждения подлинности пользователя ему необходимо просто сидеть перед монитором с веб-камерой. Таким образом, можно не только проконтролировать то, кто выполняет задания, но и не допустить к учебным материалам третьих лиц. Минусом является то, что нужно находиться напротив камеры постоянно ровно, что не всегда возможно.

Клавиатурный почерк удобен тем, что не накладывает на слушателей никаких особых требований, а также тем, что проверку посредством этого метода можно осуществлять незаметно. Недостатком же является то, что не все задания выполняются с использованием клавиатуры, значит, не все они смогут быть достоверно проконтролированы.

Оба выделенных метода могут быть успешно применены в системах ДО. В данной работе было отдано предпочтение идентификации по геометрии лица.

Распознавание лиц является одной из самых изученных задач в таких областях, как цифровая обработка изображений, компьютерное зрение, биометрия, организация видеоконференций, создание интеллектуальных систем безопасности и контроля доступа и т. п. Процесс распознавания лиц обычно состоит из двух этапов: поиск области лица на изображении и сравнение найденного лица с лицами, находящимися в базе данных. Существуют разные математические методы для поиска области лица на изображении, различающиеся скоростями и эффективностью. Эти идеи позволяют построить детектор лица, способный работать в режимах вплоть до режима реального времени. В задаче распознавания лиц такие детекторы успешно применяются для сравнения компонентов, характеризующих цветные изображения, с компонентами, описывающими неизвестные изображения.

## СПИСОК ЛИТЕРАТУРЫ:

1. Школа технологий рынка безопасности. Бурный старт биометрии. Часть 1 // БДИ. Безопасность. Достоверность. Информация. 2004. № 4 (55). С. 38–42. [Электронный ресурс]. URL: http://mx1.algoritm.org/arch/55/55_9.pdf (дата обращения: 13.12.2013).

*А. Ю. Сенцова, И. В. Машкина*

## АНАЛИЗ ПРОБЛЕМЫ ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ В ОБЛАЧНЫХ СРЕДАХ[1]

Современные тенденции развития ИТ-индустрии позволяют переходить от традиционных методов обработки информации к более прогрессивным. Одним из таких методов является перенос вычислений организации-заказчика в облачные структуры провайдера облачных вычислений — вендора. Вычисления в облаке — это базирующиеся на совокупности разных технологий способы предоставления клиенту через Интернет масштабируемых ресурсов как услуг, при которых средства поддержки этих услуг скрыты от него, а сами ресурсы оплачиваются клиентом по мере их использования [1]. Облачные вычисления позволяют не усложнять информационную инфраструктуру клиента благодаря использованию объединенных в виртуальную инфраструктуру ресурсов вендора.

Облачная модель состоит из сервисов клиентов, управляемого централизованного контента и виртуальной инфраструктуры. Кроме таких компонентов традиционной инфраструктуры информационной системы, как сеть, компьютеры, серверы, в архитектуру облачной системы входят *специализированное* программное обеспечение, blade-сервер и облачные приложения [2]. На виртуальном сервере под управлением различных операционных систем может одновременно функционировать *множество* приложений различных пользователей. Виртуальное разделение ресурсов позволяет создавать *сетевые домены*, предоставляющие разным клиентам услуги по обработке конфиденциальной информации.

[1] Работа выполнена при поддержке гранта РФФИ № 14-07-00928-а.

Облако представляет собой набор сервисов разного уровня [3], характеризующихся моделями развертывания. Самый нижний уровень и наиболее простая модель развертывания облачных сервисов позволяют заказчику взять в аренду у вендора только поддерживающую инфраструктуру, которая называется IaaS (Infrastructure-as-a-Service — инфраструктура как услуга).

Более сложная модель развертывания облачных вычислений охватывает уровень платформы информационной системы и получила название PaaS (Platform-as-a-Servise — платформа как услуга). Этот уровень включает в себя не только инфраструктуру, но и некоторые сервисные службы, например операционные системы и их обслуживание.

Модель развертывания облачных сервисов, характеризующаяся наиболее полным предоставлением услуг заказчику, предполагает использование приложений из облака для работы на локальном компьютере заказчика и называется SaaS (Software-as-a-Service — программное обеспечение как услуга). В этом случае вендор разрабатывает веб-приложение и предоставляет заказчику доступ к программному обеспечению (ПО) облака через Интернет.

Таким образом, для заказчика облачного вендора открываются широкие возможности, позволяющие снизить требования к вычислительной мощности собственной информационной системы, а любому, сколь угодно слабому вычислительному устройству получить потенциал самого современного и дорогостоящего оборудования.

Сегодня в ИТ-индустрии можно наблюдать стремительные темпы развития облачных вычислений, однако при этом недостаточно широко освещается проблема использования облачных сервисов с точки зрения информационной безопасности. Концепция вычислительного облака является отражением всеобщей тенденции глобализации информационных систем, которая, однако, сопровождается расширением перечня информационных угроз, появлением новых, ранее неизвестных уязвимостей, совершенствованием способов реализации информационных атак.

Считается, что клиент имеет тот уровень защищенности в облачной среде, который обеспечивается вендором, поэтому существующая система обеспечения информационной безопасности облака должна периодически подвергаться независимому экспертному аудиту, который в соответствии с требованиями международных стандартов является одним из обязательных этапов жизненного цикла любой информационной системы [4].

Что касается создания защищенной среды облачных сервисов, то данная проблема обостряется отсутствием не только в России, но и за рубежом общепринятых стандартов обеспечения информационной безопасности для облачных вычислений [5]. В США ассоциация Cloud Security Alliance выпустила Cloud Controls Matrix. Этот документ представляет собой перечень существующих технологий информационной безопасности, которые могут быть использованы в облачных сервисах [6]. Хотя некоторые специалисты считают, что для управления ИБ при построении облака SaaS могут быть использованы стандарты ISO 27001 и ISO 27002 [7], все же необходима разработка специальных стандартов для облачных вычислений.

Для облачных сред угрозы удаленного взлома и заражения вредоносным кодом весьма значимы из-за параллельного существования множества виртуальных машин. Отличительной *особенностью* виртуальной машины, которую нужно учитывать, является возможность ее заражения в *выключенном* состоянии [1, 8], если есть доступ к хранилищу образов виртуальных машин через Сеть. Поэтому особое внимание должно быть уделено *политикам безопасности* в виртуальных инфраструктурах.

Кроме того, проблема обеспечения безопасности в облачной среде связана с *постоянной изменчивостью* виртуальной машины (перемещение между физическими серверами). Определенную сложность создает процедура взаимной аутентификации заказчиков и вендора, так как серверов может быть несколько, они передают данные с одного узла на другой.

Некоторые проблемы возникают и в процессе интеграции антивирусного программного обеспечения в облачные виртуальные среды. Для обеспечения работы традиционного антивирусного решения необходимы большие вычислительные ресурсы. Установка антивируса на каждую виртуальную машину потребует затрат большого количества оперативной памяти и ресурсов процессоров.

Также стоит отметить, что специализированные средства обеспечения информационной безопасности, которые применяются в современных информационных системах, снижают эффективность и скорость обработки информации клиента в облаке, а облачные продукты безопасности, помогающие преодолеть эту проблему, в настоящее время не сертифицированы ФСТЭК и ФСБ.

Информационная безопасность должна обеспечиваться на всей цепочке, включая поставщика облачного решения, клиента и связывающих их коммуникаций. Клиент вендора обязан вводить в своей системе соответствующую политику безопасности, исключающую передачу прав доступа к информации, предоставленной вендором, третьим лицам. Облака не отменяют необходимости разработки и внедрения политики безопасности в сегменте клиента и использования сервисов безопасности, призванных гарантировать защиту пользовательских рабочих мест на стороне клиента.

Уровень же применяемых подходов и средств защиты как со стороны вендора, так и со стороны клиента должен определяться исходя из критичности облачных приложений для обеспечения бизнес-процессов заказчика облачных услуг.

## СПИСОК ЛИТЕРАТУРЫ:

1. *Шаньгин В. Ф.* Защита информации в компьютерных системах и сетях. М.: ДМК Пресс, 2012. — 592 с.
2. *Lee G., Antonopoulos N.* Cloud Computing: Principles, Systems and Applications. L.: Springer, 2010. — 379 p. (Computer Communications and Networks).
3. *Елманова Н.* Коротко о вычислениях в облаке // КомпьютерПресс. 2010. № 3. С 35—38.
4. ГОСТ Р ИСО/МЭК 12207-99 «Процессы жизненного цикла программных средств».
5. *Демурчев Н. Г., Ищенко С. О.* Проблемы обеспечения информационной безопасности при переходе на облачные вычисления // Материалы XI Международной научно-практической конференции «Информационная безопасность». Ч. 1. Таганрог: Изд-во ТТИ ЮФУ, 2010. — 256 с.
6. Официальный сайт Cloud Security Alliance (CSA) [Электронный ресурс]. URL: https://cloudsecurityalliance.org/ (дата обращения: 18.12.2013).
7. Безопасность как головная боль облачных вычислений [Электронный ресурс]. URL: http://eopu.tu-bryansk.ru/index.php/news/46-bezopasnost-kak-golovnaya-bol-oblachnyx-vychislenij.html (дата обращения: 20.11.2013).
8. Официальный сайт журнала «Компьютер-Пресс» [Электронный ресурс]. URL: http://compress.ru/article.aspx?id=21238&iid=967 (дата обращения: 23.11.2013).

*А. В. Сорокин*

## ОПРЕДЕЛЕНИЕ ПРИНАДЛЕЖНОСТИ КЛАСТЕРА ЦИФРОВОГО НОСИТЕЛЯ ФАЙЛУ ФОРМАТА JPEG

Задача восстановления данных чаще всего решается либо частными пользователями ПЭВМ в результате компьютерных инцидентов, либо в ходе компьютерно-технической экспертизы

носителей данных. Отличительными особенностями второго варианта является возникающее, как правило, требование восстановления максимально возможного объема информации в условиях неограниченных временных и вычислительных ресурсов. Широко распространенные программные средства для восстановления данных в большинстве случаев успешно решают поставленную задачу, однако могут не давать удовлетворительного результата в сложных случаях.

Наиболее распространенным методом восстановления файлов в рамках экспертного исследования носителей данных является метод копирования (carving). Этот метод описан, например, в [1]. Данный метод не справляется с восстановлением фрагментированных или частично перезаписанных файлов. Для случая фрагментированных файлов можно использовать различные алгоритмы, позволяющие определять корректность выделяемых фрагментов восстанавливаемых файлов того или иного типа. Форматы файлов, не использующие кодирование данных, во многих случаях позволяют осуществлять такую проверку. Особенности формата JPEG создают ряд затруднений для проведения такой проверки. Для автоматизированного выявления точки начала фрагментации файла в работах зарубежных авторов был предложен соответствующий алгоритм, подробно описанный в [2]. На его основе авторами работы [2] была предложена модификация исходного метода, способная восстанавливать файлы данного формата, состоящие из небольшого числа фрагментов. В последующих публикациях [3] те же авторы приходят к заключению, что для дальнейшего повышения быстродействия полученного алгоритма требуется возможность определения принадлежности произвольного отдельного кластера цифрового носителя файлу данного формата.

Во многих случаях при восстановлении графической информации необходимо решать задачу точного восстановления потерянного файла. В ходе решения данной задачи был проведен ряд исследований, часть которых описана в работе [4]. В этой работе показано, что особенности формата JPEG препятствуют однозначному сопоставлению пикселя или группы пикселей видимого пользователем изображения с кластером соответствующего файла. Следствием этого эффекта является невозможность визуализации отдельного кластера файла. Автором сформулирована и доказана теорема, указывающая список всех двоичных последовательностей длиной до 48 знаков, для которых построено представление соответствующей корректной интерпретации. Из этого делается вывод, что поиск запретных последовательностей с целью исключения кластеров, не принадлежащих файлам формата JPEG, не представляется эффективным.

Далее автором в [4] был предложен способ построения сигнатур и интерпретации результатов поиска этих сигнатур в произвольном кластере цифрового носителя размером 4 кб. Этот метод, также описанный в [4], позволяет добиться приемлемых значений ошибок первого и второго рода для критерия об отнесении произвольного фрагмента дисковой памяти к восстанавливаемому файлу. Таким образом, поставленная задача была решена.

Указанный результат, помимо использования в решении задач по дальнейшей оптимизации существующих методов, может быть использован и для доработки результатов применения исходного метода копирования данных к фрагментированным файлам формата JPEG. Для иллюстрации использования приведенных выше результатов был создан макет программного средства, осуществляющего поиск и восстановление файлов формата JPEG. Указанный макет позволяет пользователю визуализировать вариант восстанавливаемого файла, полученного с использованием другого средства восстановления данных. Помимо этих кластеров, в восстанавливаемый файл предлагается добавить кластеры, выделенные указанным программным средством на основании наличия в них признаков сигнатур фрагментов файла формата JPEG с параметрами кодов из заголовков восстанавливаемого файла. Используя предлагаемый список кластеров, пользователь может вручную исключить из файла кластеры, соответствующие посторонним фрагментам. Таким образом, на основании визуализируемого на каждом шаге изображения можно минимизировать потери информации и восстановить исходный файл с максимально возможной достоверностью.

СПИСОК ЛИТЕРАТУРЫ:

1. *Кэрриэ Б.* Криминалистический анализ файловых систем. СПб.: Питер, 2007.
2. *Pal A., Sencar T., Memon N.* Detecting File Fragmentation Point Using Sequential Hypothesis Testing // Digital Investigations. 2008. № 5. P. S2–S13.
3. *Pal A., Memon N.* The Evolution of File Carving // IEEE Signal Processing Magazine. Vol. 26. Issue 2. March 2009. P. 59–71.
4. *Сорокин А. В.* Об определении принадлежности кластеров диска файлам формата JPEG // Проблемы информационной безопасности. Компьютерные системы. 2012. № 4. С. 61–67.

*А. И. Тупицын*

## АВТОМАТИЗИРОВАННАЯ СИСТЕМА МОНИТОРИНГА ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ КОМПЬЮТЕРНЫХ СЕТЕЙ

В настоящее время разработано достаточно много средств мониторинга информационной безопасности компьютерных систем. В большинстве из них данные, собираемые средством мониторинга, имеют достаточно большой объем и описывают операции низкого уровня — сообщения мониторинга относятся к операциям, выполняемым операционной системой, а не к действиям пользователей и инициированных ими процессов. Поиск администратором безопасности запрещенных действий в таком наборе сообщений мониторинга является трудоемкой задачей.

Разрешенные и запрещенные действия пользователей и инициированных ими процессов в современных компьютерных системах задаются на достаточно высоком уровне абстракции — в виде политик безопасности. Политика безопасности компьютерной системы — это одно или несколько правил, процедур, практических приемов или руководящих принципов в области безопасности, которыми руководствуется организация в своей деятельности [1]. Поэтому администраторы безопасности вынуждены вручную преобразовывать высокоуровневое описание политик безопасности в описание низкоуровневых действий операционной системы и выделять те из них, которые являются нарушениями. Однако такое преобразование связано с возможными неточностями и ошибками.

С целью устранения указанных выше проблем в работе администраторов безопасности в настоящей работе представлен разработанный автором программный комплекс «Простор», автоматизирующий мониторинг информационной безопасности компьютерных систем на основе анализа выполнения политик безопасности. Данный программный комплекс позволяет автоматизировать описанные выше действия администраторов безопасности по защите компьютерных систем.

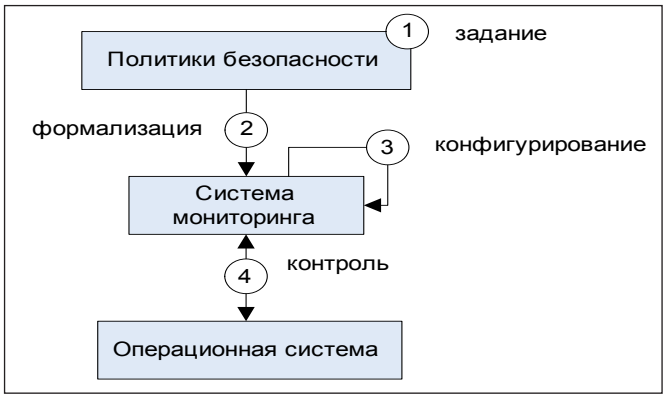



*Рис. 1. Автоматизированный технологический процесс эксплуатации системы мониторинга*

Программный комплекс «Простор» функционирует совместно с системой мониторинга (см. рис. 1). При эксплуатации программного комплекса «Простор» политики безопасности сначала задаются на естественном языке и согласовываются как с эксплуатирующим подразделением, так и с подразделением, отвечающим за безопасность. Далее с использованием средств программного комплекса «Простор» осуществляется формализация заданных политик безопасности. После этого осуществляется конфигурирование системы мониторинга для контроля выполнения формализованных политик безопасности. После этого осуществляется анализ сообщений мониторинга, формируемых системой мониторинга, и контроль соответствия сообщений мониторинга заданным политикам безопасности.

В сообщениях мониторинга присутствует код события, задающий семантику действия, а также дополнительные параметры, описывающие, кто, когда, как, над чем осуществил это действие. Языком спецификации политик безопасности, наиболее близким к данной постановке задачи, является XACML [2]. Данный язык стандартизирован организацией OASIS и допускает автоматизированное формирование политик безопасности.

При контроле выполнения заданных политик безопасности требуется осуществлять корреляцию сообщений, связанных с использованием общесистемных ресурсов вычислительных средств и действий пользователей компьютерной системы. В качестве основы для разработки метода корреляции сообщений использовался язык описания составных событий STATL [3], расширенный для применения в рассматриваемой предметной области. На основании языка STATL был разработан собственный метод корреляции сообщений.

В рамках настоящего исследования разработан базовый набор политик информационной безопасности для типовой компьютерной системы, основанный на рекомендациях международных стандартов.

Разработанный в рамках настоящего исследования программный комплекс «Простор» апробирован на тестовой компьютерной сети.

На основании полученных результатов можно сделать вывод, что предлагаемая автоматизация мониторинга на основе политик безопасности повысит эффективность работы администраторов безопасности компьютерных систем.

## СПИСОК ЛИТЕРАТУРЫ:

1. ГОСТ Р ИСО/МЭК 15408-1-2008 «Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий». Часть 1. Введение и общая модель.

2. *Bhatti R., Joshi J. B. D., Bertino E., Ghafoor A.* XML-Based Specification for Web Services Document Security // IEEE Computer. 2004. Vol. 37. № 4. P. 41–49.

3. *Eckmann S. T., Vigna G., Kemmerer R. A.* STATL: An Attack Language for State-based Intrusion Detection // Journal of Computer Security. 2002. Vol. 10. № 1–2. P. 71–103.

*А. В. Чупис*

# РАЗРАБОТКА АППАРАТНО-ПРОГРАММНОГО СТЕНДА ДЛЯ МОДЕЛИРОВАНИЯ СЕТИ ПУНКТОВ МОНИТОРИНГА В КОМПЛЕКСЕ ПЕРЕВОЗКИ СПЕЦИАЛЬНЫХ ГРУЗОВ

В последнее время отрасль различных специализированных транспортировок активно развивается. Перевозка специальных материалов и грузов по территории Российской Федерации осуществляется при условии обеспечения их физической защиты и с помощью специализированных транспортных средств, оборудованных системой безопасности. В ходе перевозки решаются три основные задачи: предупреждения, своевременного обнаружения и пресечения несанкционированных действий в отношении грузов.

Основной системой, контролирующей сохранность и защиту специальных грузов в ходе перевозки, является система безопасности контроля перевозок. Такими комплексами безопасности оборудуются транспортные средства, а также пункты мониторинга, входящие в систему мониторинга.

Пункты мониторинга (ПМ) являются важнейшей частью сети автоматизированных пунктов мониторинга системы контроля перевозок.

Целью функционирования пунктов мониторинга является обеспечение мониторинга транспортирования грузов, информационная поддержка перевозок, оповещение пользователей в случае возможных аварий, аварийных ситуаций и несанкционированных действий в отношении груза, возникающих во время транспортирования.

Проблема подготовки и обучения персонала, работающего на объектах, входящих в сеть ПМ, является очень важной задачей, стоящей перед системой безопасности перевозок.

Актуальность проблемы подготовки персонала заключается в том, что сеть существующих на данный момент ПМ не приспособлена под обучение. В связи с этим встал вопрос о создании полноценного учебного центра, в состав которого будет входить учебный стенд ПМ.

Учебный стенд (УС) используется в качестве базового элемента для обучения персонала автоматизированных ПМ предприятий, входящих в сеть ПМ. На учебном стенде моделируются возможные сценарии работы комплекса мониторинга в целом, что позволяет операторам перевозок получить навыки по реагированию на множество вероятных сценариев и возможные нештатные ситуации, которые могут произойти при транспортировке.

Так как штатный ПМ — это достаточно сложный и дорогой объект, а при его обслуживании занято множество специалистов, то было решено не переоборудовать уже существующий ПМ под учебные требования, а создать виртуализованный учебный пункт мониторинга, который позволит не только удешевить разработку УС, но и дополнить его новыми функциями, предназначенными специально для обучения персонала, которые отсутствовали в эксплуатируемых пунктах мониторинга. При создании виртуализованного учебного стенда применяются технологии виртуализации. Такие технологии позволят унифицировать аппаратную платформу ПМ и сократят затраты на оборудование и обслуживание учебного стенда.

Виртуализация — это некая технология, которая позволяет запустить и организовать одновременную работу на одном компьютере нескольких виртуальных машин — гостевых операционных систем (ОС). Виртуальная машина (ВМ) представляет собой некую программу, которая запускается с базовой ОС, и в дальнейшем уже в ВМ можно произвести установку гостевой ОС. Одной из основных частей виртуализованного учебного стенда является виртуальная машина KVM [1].

В данной статье рассматривается подход, при котором виртуализация применяется для создания новых учебных систем и технологий, и впоследствии решения, полученные с помощью виртуализации, могут использоваться при проектировании новых или модернизации уже имеющихся комплексов безопасности перевозок.

Серьёзным преимуществом виртуализации является то, что при переходе на такую технологию не требуется вносить никаких изменений в структуру программного обеспечения — существующее ПО является универсальным для различных платформ, чего не наблюдалось ранее [2].

В настоящее время при опытной эксплуатации комплексов мониторинга наблюдаются определенные проблемы: низкая эффективность использования мощностей серверов с установленным на них специальным программным обеспечением, высокая стоимость оборудования для организации серверных стоек, разработка и тестирование новых решений для системы безопасности перевозок, невозможность быстрого восстановления системы при сбоях в ее работе. Создание некоторой виртуальной платформы является одним из возможных путей решения обозначенных проблем, которая решит проблему переносимости и совместимости различных программных и аппаратных платформ. Виртуализация предоставляет новые возможности при создании новой системы обучения персонала и при адаптации существующего аппаратно-программного комплекса системы безопасности контроля перевозок под необходимые учебные требования. При помощи виртуализации расширится функционал учебного тренажерного стенда пункта мониторинга.

Результатом внедрения технологий виртуализации станет опытный образец виртуализованного учебного пункта мониторинга, который будет являться достаточно простым и дешевым устройством, предназначенным для обучения специалистов за контролем перевозок. По своим выполняемым функциям такой учебный ПМ будет соответствовать оригинальному пункту мониторинга. На виртуализованном учебном стенде можно эмулировать любую ситуацию и любой процесс, а обслуживать виртуальный стенд может один оператор. Использование технологий виртуализации позволит проводить параллельное обучение специалистов на одном и том же сервере. Важным достоинством внедрения таких технологий является то, что на виртуализованном учебном стенде можно пошагово отслеживать и контролировать весь процесс обучения, а следовательно, появляется возможность защитить систему от неправильных действий пользователя.

## СПИСОК ЛИТЕРАТУРЫ:

1. *Tholeti B. P.* Hypervisors, virtualization, and the cloud: Dive into the KVM hypervisor // IBM developerWorks. 2011. P. 1–3.
2. *Янюшкин В. В.* Использование технологий виртуализации вычислительных и графических серверов при проектировании тренажеров, тренажерно-моделирующих комплексов // Программные продукты и системы. 2013. № 3. С. 21–27.

*Э. Э. Яндыбаева, И. В. Машкина*

# МОДЕЛЬ НАРУШИТЕЛЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ В ИНФОРМАЦИОННОЙ СИСТЕМЕ ЭЛЕКТРОННОЙ ТОРГОВОЙ ПЛОЩАДКИ

Целью работы является разработка модели нарушителя информационной безопасности (ИБ) в информационной системе (ИС) электронной торговой площадки (ЭТП). Модель нарушителя включает описание и классификацию лиц, потенциально опасных с точки зрения обеспечения ИБ. При этом учитываются опыт и знания таких лиц, доступные им ресурсы, необходимые для реализации угрозы ИБ, а также возможная мотивация их действий [1].

В настоящей работе под нарушителем понимается лицо, которое действует в пределах контролируемой зоны информационной системы и совершает заранее обдуманное действие с осознанием его опасных последствий. Под злоумышленником понимается лицо, осуществляющее атаки из-за пределов контролируемой зоны ИС. В качестве объекта защиты рассматривается ИС электронной торговой площадки, где оператор ЭТП – владелец ИС ЭТП, клиент ЭТП – зарегистрированное на сайте ЭТП физическое или юридическое лицо.

К злоумышленникам в ИС ЭТП могут относиться конкуренты оператора ЭТП и конкуренты клиентов ЭТП. Злоумышленник может иметь сообщников среди работников оператора ЭТП.

У злоумышленника могут быть следующие мотивы:

- нанесение ущерба оператору ЭТП;
- нанесение ущерба клиенту ЭТП;
- победа в электронном аукционе нечестным путем.

В зависимости от сведений, которыми они обладают, злоумышленники разделены на две категории. Категория 1 – незарегистрированные в ИС ЭТП физические и юридические лица. Данные лица обладают информацией об ИС ЭТП только из общедоступных источников. Категория 2 – зарегистрированные клиенты ЭТП. Лицо данной категории обладает логином, паролем и ключом электронной подписи, обеспечивающими доступ в ИС ЭТП, а также конфиденциальными данными, доступными в личном кабинете.

Каналами атак злоумышленников являются каналы связи ИС ЭТП, выходящие за пределы контролируемой зоны, сеть Интернет и элементы информационной инфраструктуры, которые в процессе своего жизненного цикла (модернизации, сопровождения, ремонта, утилизации) оказываются за пределами контролируемой зоны.

Нарушителями являются сотрудники ЭТП. Они находятся в пределах контролируемой зоны информационной системы. Кроме того, к нарушителям относятся сотрудники внешних по отношению к оператору ЭТП компаний, которые занимаются обслуживанием ИС ЭТП. Возможности нарушителя существенным образом зависят от действующих в пределах контролируемой зоны режимных, организационных и технических мер защиты, в том числе по допуску физических лиц к защищаемой информации и контролю порядка проведения работ. Содержанием деятельности нарушителя является прямое нецелевое использование предоставленного ему в порядке выполнения служебных обязанностей контроля над защищаемыми ресурсами либо нерегламентированная деятельность для получения контроля над такими ресурсами. При этом он будет стремиться к сокрытию следов своей деятельности [2].

Мотивы нарушителя могут быть следующими:

- нанесение ущерба оператору электронной площадки,
- сговор со злоумышленником.

Нарушители подразделяются на семь категорий в зависимости от способа доступа и полномочий доступа к защищаемой информации.

Категория 1 — лица, имеющие санкционированный доступ к локальной вычислительной сети, в которой функционирует ИС ЭТП, но не имеющие доступа к самой защищаемой информации. Лица данной категории обладают фрагментами информации о топологии ИС ЭТП, используемых протоколах и их сервисах.

Категория 2 — пользователи ИС ЭТП, осуществляющие ограниченный доступ к защищаемой информации с рабочего места (администраторы ЭТП). Администратор ЭТП обладает логином, паролем и ключом электронной подписи, обеспечивающими доступ в ИС ЭТП, а также конфиденциальными данными, к которым имеет доступ. К таким данным в том числе относится вся информация о проводимых аукционах и вся информация, размещенная в личных кабинетах клиентов ЭТП.

Категория 3 — системные администраторы ИС ЭТП. Лица данной категории владеют информацией о топологии информационной системы и локальной вычислительной сети, о системном и прикладном программном обеспечении информационной системы, о технических средствах и конфигурации информационной системы.

Категория 4 — администраторы безопасности ИС ЭТП. Данные лица обладают полной информацией об информационной системе.

Категория 5 — программисты-разработчики (поставщики) прикладного программного обеспечения ИС ЭТП и лица, обеспечивающие его сопровождение. Лица данной категории обладают сведениями об алгоритмах и программах обработки информации в ИС ЭТП.

Категория 6 — разработчики и лица, обеспечивающие поставку, сопровождение и ремонт технических средств ИС ЭТП. Лица данной категории обладают фрагментами информации о топологии и технических средствах обработки и защиты информации в ИС ЭТП.

Категория 7 — сотрудники удостоверяющего центра, генерирующие ключи электронной подписи для клиентов и администраторов ЭТП. Лица данной категории обладают полными сведениями о процедуре выдачи ключа электронной подписи клиентам ЭТП и администраторам ЭТП, сведениями об алгоритмах и программах генерации ключа электронной подписи, фрагментами информации о топологии и технических средствах обработки и защиты информации в системе генерации ключей электронной подписи.

Таким образом, проведена классификация лиц, потенциально опасных с точки зрения обеспечения ИБ ИС ЭТП. Выделены и описаны две категории злоумышленников и семь категорий нарушителей. Разработанная модель нарушителя ИБ ИС ЭТП позволяет перейти к моделированию преднамеренных угроз ИБ ИС ЭТП.

## СПИСОК ЛИТЕРАТУРЫ:

1. Методические рекомендации по обеспечению с помощью криптосредств безопасности персональных данных при их обработке в информационных системах персональных данных с использованием средств автоматизации, утвержденные руководством 8 Центра ФСБ России 21 февраля 2008 г.

2. Базовая модель угроз безопасности персональных данных при их обработке в информационных системах персональных данных, утвержденная заместителем директора ФСТЭК России 15 февраля 2008 г.