Индекс 79187

ISSN 2073-2600

ВОПРОСЫ ЗАЩИТЫ ИНФОРМАЦИИ

2

(141)

Подписывайтесь,

читайте,

пишите в наш журнал

Москва 2023

Все новейшие достижения и современные разработки предприятий оборонного комплекса — в научно-технических журналах ФГУП «НТЦ оборонного комплекса «Компас»

ФГУП «Научно-технический центр оборонного комплекса «Компас» является издателем следующих научных журналов:

Межотраслевой научно-технический журнал

Оборонный комплекс — научно-техническому прогрессу России
(4 выпуска)

Подписной индекс **79379**

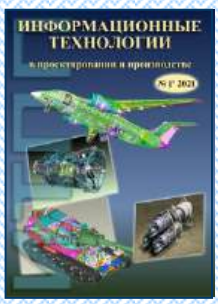
Издается с 1984 года

Межотраслевой научно-технический журнал

Конструкции из композиционных материалов
(4 выпуска)

Подписной индекс **80089**

Издается с 1981 года



Научно-технический журнал

Информационные технологии в проектировании и производстве
(4 выпуска)

Подписной индекс **79378**

Издается с 1976 года

Межотраслевой научно-практический журнал

Экология промышленного производства
(4 выпуска)

Подписной индекс **80090**

Издается с 1993 года

Научно-практический журнал

Вопросы защиты информации
(4 выпуска)

Подписной индекс **79187**

Издается с 1974 года

Все издания ФГУП "Научно-технический центр оборонного комплекса «Компас»:

✓ *включены решением ВАК Министерства науки и высшего образования России в перечень ведущих рецензируемых научных журналов и изданий, в которых должны быть опубликованы основные результаты диссертаций на соискание ученой степени кандидата и доктора наук;*

✓ *метаданные выпусков включены в базу данных Российского индекса научного цитирования (РИНЦ).*

Более подробную информацию об изданиях, подписке, дополнительных услугах можно получить
по тел.: 8 (495) 491-43-17, 8 (495) 491-77-67, 8 (495) 491-77-20 (*подписка*);
факс: 8 (495) 491-44-80.
E-mail: izdanie@ntckompas.ru

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ УНИТАРНОЕ ПРЕДПРИЯТИЕ
«НАУЧНО-ТЕХНИЧЕСКИЙ ЦЕНТР ОБОРОННОГО КОМПЛЕКСА «КОМПАС»
(ФГУП «НТЦ ОБОРОННОГО КОМПЛЕКСА «КОМПАС»)

ВОПРОСЫ ЗАЩИТЫ ИНФОРМАЦИИ

НАУЧНО-ПРАКТИЧЕСКИЙ ЖУРНАЛ

2 (141)

Москва 2023

Основан в 1974 г.

СОДЕРЖАНИЕ

Вопросы защиты информации: Науч.-практ. журн. / ФГУП «НТЦ оборонного комплекса «Компас», 2023.
Вып. 2 (141). С. 1—72.

Компьютерная верстка: *Н. В. Ильина*, *К. В. Трыкина*

Подписано в печать 10.06.2023. Формат 60x84 1/8.
Печать офсетная. Усл. печ. л. 8,4, . Уч.-изд. л. 8,6.
Тираж 400 экз. Заказ 2016. Свободная цена.
Адрес редакции: 125424, Москва,
Волоколамское ш., 77. ФГУП «НТЦ оборонного комплекса «Компас».
http://ntckompas.ru
Отпечатано: 101000, Москва,
Лубянский проезд, д. 15, стр. 4, помещ. IX, ком. 15, 16
ООО «Спиди-Принт.ру»
Индекс 79187.

ИНЖЕНЕРНАЯ КРИПТОГРАФИЯ

УДК 003.26
DOI: 10.52190/2073-2600_2023_2_3
EDN: DPFQXW

Унифицированные способы задания векторных конечных полей как примитивов алгоритмов многомерной криптографии

А. А. Костина
Санкт-Петербургский федеральный исследовательский центр Российской академии наук (СПб ФИЦ РАН), Санкт-Петербург, Россия

Предложены унифицированные способы построения таблиц умножения базисных векторов, с помощью которых задаются конечные коммутативные алгебры над конечным базовым полем F, являющиеся расширениями поля F. Последние позволяют задать нелинейные биективные отображения векторов, задаваемые множеством многочленов над полем F, и могут быть использованы в рамках недавно предложенной новой концепции построения алгоритмов многомерной криптографии, позволяющей существенно сократить размер открытого ключа при заданном уровне стойкости по сравнению с известными постквантовыми алгоритмами цифровой подписи и открытого шифрования, основанными на трудности обращения нелинейных отображений, задаваемых системой многочленов над полем F.

Ключевые слова: компьютерная безопасность, постквантовые шифры, алгоритмы цифровой подписи, многомерная криптография, нелинейные отображения, биективные отображения, конечные алгебры, векторные конечные поля.

Многомерная криптография [1—3] является одной из перспективных направлений постквантовой криптографии, которая основана на вычислительной трудности решения системы из многих степенных (обычно второй и третьей степени) уравнений, заданных над конечным полем достаточно малого порядка, с многими неизвестными. Квантовый вычислитель (компьютер) не является эффективным для решения этой задачи, поэтому алгоритмы многомерной криптографии, обладающие стойкостью в традиционном понимании, являются постквантовыми, т. е. стойкими к атакам с использованием квантовых компьютеров.

Алгоритмы многомерной криптографии обладают достаточно большой производительностью, однако их широкое практическое применение ограничивается тем, что для обеспечения требуемого уровня (например, 128-битной, 192-битной и 256-битной) они требуют использования открытого ключа, имеющего размер в десятки и сотни раз больше по сравнению в другими известными постквантовыми двухключевыми криптоалгоритмами [4].

Для устранения данного недостатка в работе [4] предложен новый подход к построению алгоритмов многомерной криптографии, основанный на задании нелинейных биективных отображений в виде каскада операций возведения во вторую и третью степень, выполняемых в конечных полях. Благодаря использованию векторных конечных полей указанные операции, а значит и указанное отображение, могут быть заданы в виде набора многочленов. Однако известен только один тип таблиц умножения базисных векторов (ТУБВ), с помощью которых могут быть заданы конечные поля в форме конечных алгебр, т. е. векторные конечные поля (см. табл. 1 в [4]). Это вносит определенные ограничения в практической реализации алгоритмов многомерной криптографии в рамках концепции [4].

В целях расширения вариантов задания нелинейных биективных отображений, реализуемых как операция экспоненцирования в векторных конечных полях, и возможностей реализации концепции [4] в данной работе предложены новые способы унифицированного задания ТУБВ, в том числе и для произвольных размерностей векторного пространства, в котором определяется операция умножения векторов.

Костина Анна Александровна, научный сотрудник.
E-mail: to.ann@inbox.ru

Статья поступила в редакцию 31 марта 2023 г.

Предварительные сведения

В многомерной криптографии открытый ключ формируется в виде системы из u степенных многочленов с коэффициентами и переменными, принимающими значения в некотором конечном поле F. Открытый ключ задает нелинейное биективное отображение $\mathcal{P}$ n-мерных векторов (с координатами в поле F) в u-мерные ($u \geq n$). Значение u определяет число уравнений в системе, а n — число неизвестных в упомянутой ранее системе степенных уравнений. Отображение $\mathcal{P}$ является трудно обратимым, но содержит секретную лазейку, известную создателю (владельцу) открытого ключа.

В традиционном подходе многомерной криптографии для формирования открытого ключа разрабатывается нелинейное отображение $\mathcal{N}$, задаваемое набором многочленов над полем $\mathbb{F}$ и позволяющее с достаточной вычислительной эффективностью осуществить обратное отображение $\mathcal{N}^{-1}$, являющееся секретной лазейкой. Последняя маскируется путем вычисления открытого ключа в виде суперпозиции $\mathcal{P} = \mathcal{N} \bullet \mathcal{L}_1$, $\mathcal{P} = \mathcal{L}_2 \bullet \mathcal{N}$ или $\mathcal{P} = \mathcal{L}_2 \bullet \mathcal{N} \bullet \mathcal{L}_1$, где линейные отображения $\mathcal{L}_1$ и $\mathcal{L}_2$ маскируют $\mathcal{N}$, а значит и $\mathcal{N}^{-1}$. Например, $\mathcal{L}_1$ и $\mathcal{L}_2$ можно задать в виде умножения вектора на невырожденную матрицу размера $n \times n$ и $u \times u$ (над полем F) соответственно. Выполнение отображений $\mathcal{L}_1$ и $\mathcal{L}_2$ легко представить как вычисление набора многочленов над F.

В способе [4] открытый ключ $\mathcal{P}$ формируется как суперпозиция, включающая два различных обратимых нелинейных отображения $\mathcal{N}_1$ и $\mathcal{N}_2$, например, в виде $\mathcal{P} = \mathcal{N}_2 \bullet \mathcal{N}_1$ или $\mathcal{P} = \mathcal{N}_2 \bullet \mathcal{L} \bullet \mathcal{N}_1$, где используемое линейное преобразование $\mathcal{L}$ не приводит к увеличению размера открытого ключа, поскольку оно задается как перестановка координат отображаемого вектора. При этом отображения $\mathcal{N}_1$ и $\mathcal{N}_2$ задаются в виде операций экспоненциирования во вторую или третью степень в поле расширения степени k ($1 < k \leq n$) поля F. Для того, чтобы $\mathcal{N}_1$ и $\mathcal{N}_2$ могли быть заданы набором многочленов над F, конечное расширение F задается в виде конечной коммутативной алгебры над F.

Конечная m-мерная алгебра представляет собой m-мерное векторное пространство (например, над полем F), в котором определена замкнутая операция умножения всевозможных пар векторов, обладающая свойствами левой и правой дистрибутивности относительно операции сложения. Вектор $A = (a_1, a_2, \ldots, a_m)$ можно представить в виде суммы его компонент $\mathbf{A} = \sum_{i=1}^{m} a_i \mathbf{e}_i$, где $\mathbf{e}_i$ — базисные векторы.

Операция умножения векторов $\mathbf{A}$ и $\mathbf{B} = \sum_{j=1}^{m} b_j \mathbf{e}_j$ обычно задается по правилу перемножения каждой компоненты первого вектора с каждой компонентой второго вектора, т. е. по формуле:

$$\mathbf{AB} = \sum_{i=1}^{m} \sum_{j=1}^{m} a_i b_j \left(\mathbf{e}_i \mathbf{e}_j \right), \qquad (1)$$

в которой всевозможные произведения $\mathbf{e}_i\mathbf{e}_j$ заменяются на некоторый однокомпонентный вектор $\lambda\, \mathbf{e}_k$ в соответствии с некоторой ТУБВ. Значение $\lambda \neq 1$ называется структурной константой. При этом левый множитель в произведении $\mathbf{e}_i\mathbf{e}_j$ указывает строку, а правый — столбец, пересечение которых выделяет ячейку, содержащую значение $\lambda\mathbf{e}_k$.

Для задания векторного конечного поля следует составить ТУБВ, определяющую свойства коммутативности и ассоциативности операции умножения векторов.

Впервые задание конечных полей расширения в векторной форме рассматривали в работе [5], где был предложен общий вид ТУБВ, включающий три независимые структурные константы и позволяющий задать формирование векторных конечных полей для произвольной размерности $m \geq 2$. В [5] также были сформулированы условия образования конечных алгебр, являющихся полями. Одним из таких условий является делимость порядка мультипликативной группы конечного поля F на значение m.

Для задания векторных конечных полей в [6] были предложены частные случаи ТУБВ другого вида, содержащие $m - 1$ независимых структурных констант. В случае разработки алгоритмов многомерной криптографии по способу [4] представляет интерес использование ТУБВ со сравнительно большим числом независимых структурных констант, поскольку их используют как элементы секретного ключа.

Вопрос унифицированного задания ТУБВ впервые возник в связи с задачей задания конечных некоммутативных ассоциативных алгебр больших размерностей. Для решения этой задачи в работе [7] была предложена компактная формула, по которой для произвольной четной размерности $m \geq 6$ может быть сгенерирована ТУБВ, задающая некоммутативное ассоциативное умножение.

В данной работе предложены формулы различных типов, каждая из которых обеспечивает генерацию ТУБВ произвольных размерностей $m \geq 2$, задающих коммутативное ассоциативное умножение и возможность выбора значений структурных констант, при которых формируется векторное конечное поле.

Первый унифицированный способ

Первый разработанный способ унифицированного задания ТУБВ охватывает типовой вариант [5] как частный случай и описывается следующей математической формулой:

$$\mathbf{e}_i\mathbf{e}_j = \mathbf{e}_{((i+j+d) \bmod m)+1}, \qquad (2)$$

где $i, j, d = 1, 2, \ldots, m$ и d — параметр задающий вид единичного вектора **E** в алгебре, которая задается таблицей, генерируемой по формуле (2). Вектор **E** содержит одну координату с единичным значением и $m - 1$ нулевых координат. Значение $((m - 2 - d) \bmod m) + 1$ задает индекс единичной координаты.

Легко видеть, что формула (2) генерирует ТУБВ, задающую коммутативное умножение векторов. Доказательство ассоциативности умножения выполняется как доказательство выполнимости следующего равенства для произвольной тройки базисных векторов:

$$(\mathbf{e}_i\mathbf{e}_j)\mathbf{e}_k = \mathbf{e}_i(\mathbf{e}_j\mathbf{e}_k). \qquad (3)$$

Левая часть последнего равенства имеет индекс, равный значению

$$\Big[\Big(\big((i+j+d)\bmod m\big)+1+k+d\Big)\bmod m\Big]+1 = \\ =\Big[\big(i+j+k+2d+1\big)\bmod m\Big]+1.$$

Правая часть (3) имеет индекс, равный значению

$$\Big[\Big(\big(i+(j+k+d) \bmod m\big)+1+d\Big)\bmod m\Big]+1= \\ =\Big[\big(i+j+k+2d+1\big)\bmod m\Big]+1.$$

Таким образом, равенство (3) имеет место для всевозможных троек базисных векторов, поэтому из формулы (1) следует, что для произвольных трех векторов **A**, **B** и **C** выполняется равенство (**AB**)**C** = **A**(**BC**), т. е. операция умножения векторов является ассоциативной.

Формула задает распределение базисных векторов в генерируемой ТУБВ. Для того, чтобы последняя могла задать формирование векторных конечных полей, следует в сгенерированную ТУБВ внести структурные константы, распределение каждой из которых можно экспериментально установить по аналогии с распределениями структурных констант в ТУБВ из работ [5, 6]. Табл. 1 и 2 иллюстрируют частные случаи задания четырехмерных векторных полей при $d = 0$ и $d = 3$.

Факт формирования векторных полей проверялся на вычислительном эксперименте, который осуществляли следующим образом. Выбирали случайные значения всех структурных констант и устанавливали наличие вектора, порядок которого равен $p^4 - 1$. Если для данной комбинации случайных значений не удавалось найти вектор указанного порядка, то одно из значений структурных констант модифицировали и поиск продолжали.

Таблица 1

Задание векторного конечного поля $GF(p^4)$ как расширения простого поля $GF(p)$ при $d = 0$ и использовании четырех структурных констант $\tau, \varepsilon, \mu, \lambda \in GF(p)$ (единицей поля $GF(p^4)$ является вектор $\mathbf{E} = (0, 0, \tau^{-1}, 0)$)

×	$\mathbf{e}_1$	$\mathbf{e}_2$	$\mathbf{e}_3$	$\mathbf{e}_4$
$\mathbf{e}_1$	$\tau^{-1}\mu\,\varepsilon\,\mathbf{e}_3$	$\mu\,\mathbf{e}_4$	$\tau\,\mathbf{e}_1$	$\varepsilon\,\mathbf{e}_2$
$\mathbf{e}_2$	$\mu\,\mathbf{e}_4$	$\mu\lambda\,\mathbf{e}_1$	$\tau\,\mathbf{e}_2$	$\tau^{-1}\mu\lambda\,\varepsilon\,\mathbf{e}_3$
$\mathbf{e}_3$	$\tau\,\mathbf{e}_1$	$\tau\,\mathbf{e}_2$	$\tau\,\mathbf{e}_3$	$\tau\,\mathbf{e}_4$
$\mathbf{e}_4$	$\varepsilon\,\mathbf{e}_2$	$\tau^{-1}\mu\lambda\,\varepsilon\,\mathbf{e}_3$	$\tau\,\mathbf{e}_4$	$\lambda\,\varepsilon\,\mathbf{e}_1$

Таблица 2

Задание векторного конечного поля $GF(p^4)$ при $d = 3$ и использовании структурных констант $\tau, \varepsilon, \mu, \lambda \in GF(p)$ (единицей поля является вектор $\mathbf{E} = (0, 0, 0, \tau^{-1})$)

×	$\mathbf{e}_1$	$\mathbf{e}_2$	$\mathbf{e}_3$	$\mathbf{e}_4$
$\mathbf{e}_1$	$\lambda\mu\,\mathbf{e}_2$	$\mu\,\mathbf{e}_3$	$\tau^{-1}\lambda\mu\,\varepsilon\,\mathbf{e}_4$	$\tau\,\mathbf{e}_1$
$\mathbf{e}_2$	$\mu\,\mathbf{e}_3$	$\tau^{-1}\mu\,\varepsilon\,\mathbf{e}_4$	$\varepsilon\,\mathbf{e}_1$	$\tau\,\mathbf{e}_2$
$\mathbf{e}_3$	$\tau^{-1}\lambda\mu\,\varepsilon\,\mathbf{e}_4$	$\varepsilon\,\mathbf{e}_1$	$\lambda\,\varepsilon\,\mathbf{e}_2$	$\tau\,\mathbf{e}_3$
$\mathbf{e}_4$	$\tau\,\mathbf{e}_1$	$\tau\,\mathbf{e}_2$	$\tau\,\mathbf{e}_3$	$\tau\,\mathbf{e}_4$

Если многократное (несколько десятков раз) модифицирование значения некоторой заданной константы не приводило к нахождению вектора порядка $p^4 - 1$, то делался вывод, что значение данной константы не влияет на формирование векторного поля. Такими константами для табл. 1 и 2 являются τ и λ. При этом константа τ определяет значение ненулевой координаты единичного вектора, которая равна τ^{-1}. Таким образом, имеются три типа структурных констант:

- влияющие на формирование векторных полей (ε и μ);
- невлияющие на формирование векторных полей (λ);
- относящиеся ко второму типу и влияющие на значение единичного вектора **E** (τ).

Аналогичная ситуация имеет место и для ТУБВ, генерируемых для других значений размерности (особенностью является отсутствие структурных констант второго типа в случае простого значения размерности m). Табл. 3 показывает пример ТУБВ, генерируемой формулой (1) для случая $m = 5$.

Таблица 3

Задание векторного конечного поля $GF(p^5)$ при $d = 1$ и использовании структурных констант $\tau, \varepsilon, \mu, \eta, \delta \in GF(p)$ (единицей поля является вектор $\mathbf{E} = (0, 0, \tau^{-1}, 0, 0)$)

×	$\mathbf{e}_1$	$\mathbf{e}_2$	$\mathbf{e}_3$	$\mathbf{e}_4$	$\mathbf{e}_5$
$\mathbf{e}_1$	$\delta\varepsilon\,\mathbf{e}_4$	$\delta\varepsilon\,\mathbf{e}_5$	$\tau\,\mathbf{e}_1$	$\delta\mu\,\mathbf{e}_2$	$\tau^{-1}\delta\varepsilon\mu\eta\,\mathbf{e}_3$
$\mathbf{e}_2$	$\delta\varepsilon\,\mathbf{e}_5$	$\varepsilon\eta\,\mathbf{e}_1$	$\tau\,\mathbf{e}_2$	$\tau^{-1}\delta\varepsilon\mu\eta\,\mathbf{e}_3$	$\varepsilon\eta\,\mathbf{e}_4$
$\mathbf{e}_3$	$\tau\,\mathbf{e}_1$	$\tau\,\mathbf{e}_2$	$\tau\,\mathbf{e}_3$	$\tau\,\mathbf{e}_4$	$\tau\,\mathbf{e}_5$
$\mathbf{e}_4$	$\delta\mu\,\mathbf{e}_2$	$\tau^{-1}\delta\varepsilon\mu\eta\,\mathbf{e}_3$	$\tau\,\mathbf{e}_4$	$\delta\mu\,\mathbf{e}_5$	$\mu\eta\,\mathbf{e}_1$
$\mathbf{e}_5$	$\tau^{-1}\delta\varepsilon\mu\eta\,\mathbf{e}_3$	$\varepsilon\eta\,\mathbf{e}_4$	$\tau\,\mathbf{e}_5$	$\mu\eta\,\mathbf{e}_1$	$\mu\eta\,\mathbf{e}_2$

Второй унифицированный способ

Второй разработанный способ унифицированного задания ТУБВ охватывает четные значения размерности, такие, что число $m + 1$ является простым, и описывается следующей математической формулой:

$$\mathbf{e}_i\mathbf{e}_j = \mathbf{e}_{i\cdot j\cdot d \bmod (m+1)}, \qquad (4)$$

где $i, j, d = 1, 2, \ldots, m$ и d — параметр, задающий вид единичного вектора **E** в алгебре, которая задается таблицей, генерируемой по формуле (4). Вектор **E** содержит одну координату с единичным значением и $m - 1$ нулевых координат. Значение $d^{-1} \bmod (m + 1)$ задает индекс единичной координаты.

Легко видеть, что формула (4) задает ТУБВ, определяющую коммутативное умножение векторов. Для доказательства ассоциативности умножения покажем выполнимость равенства (3) для произвольной тройки базисных векторов. Левая часть равенства (3) имеет индекс, равный значению

$$\left[\left((i\cdot j\cdot d)\bmod (m+1)\right)\cdot k\cdot d\right]\bmod(m+1)=$$
$$=\left(i\cdot j\cdot k\cdot d^2\right)\bmod(m+1).$$

Правая часть (3) имеет индекс, равный значению

$$\left[i\left(\left(j\cdot k\cdot d\right)\bmod\left(m+1\right)d\right)\right]\bmod\left(m+1\right)=$$
$$=\left(i\cdot j\cdot k\cdot d^2\right)\bmod\left(m+1\right).$$

Поскольку равенство (3) выполняется для всех троек базисных векторов, то из формулы (1) следует, что для произвольных трех векторов **A**, **B** и **C** выполняется равенство (**AB**)**C** = **A**(**BC**), определяющее свойство ассоциативности операции умножения векторов.

Табл. 4 и 5 представляют две различные ТУБВ, сгенерированные по унифицированной формуле (4) и позволяющие задать конечные поля $GF(p^4)$ с различными видами единичного вектора.

Таблица 4

Задание векторного поля по формуле (4) над простым полем $GF(p)$ при $d = 3$ и использовании четырех структурных констант $\tau, \varepsilon, \mu, \lambda \in GF(p)$

(единицей поля $GF(p^4)$ является вектор $\mathbf{E} = (0, \tau^{-1}, 0, 0)$)

×	$\mathbf{e}_1$	$\mathbf{e}_2$	$\mathbf{e}_3$	$\mathbf{e}_4$
$\mathbf{e}_1$	$\lambda\varepsilon\,\mathbf{e}_3$	$\tau\,\mathbf{e}_1$	$\varepsilon\,\mathbf{e}_4$	$\tau^{-1}\lambda\,\mu\varepsilon\,\mathbf{e}_2$
$\mathbf{e}_2$	$\tau\,\mathbf{e}_1$	$\tau\,\mathbf{e}_2$	$\tau\,\mathbf{e}_3$	$\tau\,\mathbf{e}_4$
$\mathbf{e}_3$	$\varepsilon\,\mathbf{e}_4$	$\tau\,\mathbf{e}_3$	$\tau^{-1}\,\mu\varepsilon\,\mathbf{e}_2$	$\mu\,\mathbf{e}_1$
$\mathbf{e}_4$	$\tau^{-1}\lambda\,\mu\varepsilon\,\mathbf{e}_2$	$\tau\,\mathbf{e}_4$	$\mu\,\mathbf{e}_1$	$\lambda\,\mu\,\mathbf{e}_3$

Таблица 5

Задание векторного поля по формуле (4) над простым полем $GF(p)$ при $d = m = 4$ и использовании четырех структурных констант $\tau, \varepsilon, \mu, \lambda \in GF(p)$

(единицей поля $GF(p^4)$ является вектор $\mathbf{E} = (0, 0, 0, \tau^{-1})$)

×	$\mathbf{e}_1$	$\mathbf{e}_2$	$\mathbf{e}_3$	$\mathbf{e}_4$
$\mathbf{e}_1$	$\tau^{-1}\mu\varepsilon\,\mathbf{e}_4$	$\varepsilon\,\mathbf{e}_3$	$\mu\,\mathbf{e}_2$	$\tau\,\mathbf{e}_1$
$\mathbf{e}_2$	$\varepsilon\,\mathbf{e}_3$	$\lambda\varepsilon\,\mathbf{e}_1$	$\tau^{-1}\lambda\,\mu\varepsilon\,\mathbf{e}_4$	$\tau\,\mathbf{e}_2$
$\mathbf{e}_3$	$\mu\,\mathbf{e}_2$	$\tau^{-1}\lambda\,\mu\varepsilon\,\mathbf{e}_4$	$\lambda\,\mu\,\mathbf{e}_1$	$\tau\,\mathbf{e}_3$
$\mathbf{e}_4$	$\tau\,\mathbf{e}_1$	$\tau\,\mathbf{e}_2$	$\tau\,\mathbf{e}_3$	$\tau\,\mathbf{e}_4$

Второй унифицированный способ применим для задания векторных полей размерности $m = 4$, 6, 10, 12, 16, 18 и т. д. Однако для больших размерностей для нахождения распределения структурных констант в сгенерированных исходных ТУБВ требуется разработать специальные компьютерные программы. При этом для одной из констант имеется стандартное распределение по

ТУБВ, независимо от значения m и способа генерации последней, которое состоит в следующем. Во всех ячейках столбца и строки с номером, равным индексу ненулевой координаты в единичном векторе **E** вносится структурная константа τ и во всех ячейках, содержащих базисный вектор с указанным номером вносится структурная константа τ^{-1} (распределение этих двух разных значений рассматриваются как распределение одной структурной константы τ ввиду явной зависимости этих значений).

Например, для случая генерации ТУБВ по второму способу при $m = 10$ и $d = 3$ имеем единичный вектор $\mathbf{E} = (0, 0, 0, \tau^{-1}, 0, 0, 0, 0, 0, 0)$ и распределение базисных векторов и структурной константы, представленные в табл. 6. Можно ожидать, что существуют различные распределения еще $m - 1$ независимых констант, однако их нахождение представляет задачу, требующую компьютерной поддержки.

Специфика задания нелинейных отображений

Задание нелинейных отображений как операций экспоненцирования во вторую и третью степень в поле $GF(p^m)$, выполняемых как вычисление значений множества многочленов над полем $GF(p)$, автоматически обеспечивает существование обратного отображения и возможность его эффективного выполнения как для малых значений порядка поля $GF(p)$, так и для случая значений p большой разрядности. При этом степень многочленов может быть существенно увеличена (от двух — трех до девяти и более). Поскольку вычислительная трудоемкость решения системы полиномиальных уравнений зависит не только от числа уравнений, но и от значения p [7, 8] и значения степени многочленов, то подход к разработке алгоритмов многомерной криптографии, предложенный в [4], позволяет существенно уменьшить число многочленов, задаваемых как открытый ключ путем задания этих многочленов над полем сравнительно большого порядка.

Заслуживает внимания также и то, что в рамках подхода [4] обратное отображение выполняется как операция возведения в степени $2^{-1} \bmod t$ (в конечных полях характеристики два) и $3^{-1} \bmod t$ (в конечных полях четной и нечетной характеристики), где $t = p^m - 1$ — порядок мультипликативной группы векторного конечного поля. Данные степени представляют собой достаточно большие натуральные числа, поэтому представляется практически невыполнимым представление обратного отображения множеством каких-либо многочленов, например, "пересчитываемых" по многочленам открытого ключа. Это предполагает достаточную секретность потайной лазейки, связанной с открытым ключом, и допускает в качестве основного способа взлома криптоалгоритма решение системы степенных уравнений, задаваемых многочленами открытого ключа.

Предложенные унифицированные способы построения ТУБВ для задания векторных полей различных размерностей существенно увеличивают число возможных ТУБВ для заданного значения размерности m, однако это нецелесообразно использовать для использования секретности ТУБВ,

Таблица 6

Распределение базисных векторов по второму унифицированному способу при $m = 10$ и $d = 3$ и структурной константы τ

×	$\mathbf{e}_1$	$\mathbf{e}_2$	$\mathbf{e}_3$	$\mathbf{e}_4$	$\mathbf{e}_5$	$\mathbf{e}_6$	$\mathbf{e}_7$	$\mathbf{e}_8$	$\mathbf{e}_9$	$\mathbf{e}_{10}$
$\mathbf{e}_1$	$\mathbf{e}_3$	$\mathbf{e}_6$	$\mathbf{e}_9$	$\tau\,\mathbf{e}_1$	$\tau^{-1}\,\mathbf{e}_4$	$\mathbf{e}_7$	$\mathbf{e}_{10}$	$\mathbf{e}_2$	$\mathbf{e}_5$	$\mathbf{e}_8$
$\mathbf{e}_2$	$\mathbf{e}_6$	$\mathbf{e}_1$	$\mathbf{e}_7$	$\tau\,\mathbf{e}_2$	$\mathbf{e}_8$	$\mathbf{e}_3$	$\mathbf{e}_9$	$\tau^{-1}\,\mathbf{e}_4$	$\mathbf{e}_{10}$	$\mathbf{e}_5$
$\mathbf{e}_3$	$\mathbf{e}_9$	$\mathbf{e}_7$	$\mathbf{e}_5$	$\tau\,\mathbf{e}_3$	$\mathbf{e}_1$	$\mathbf{e}_{10}$	$\mathbf{e}_8$	$\mathbf{e}_6$	$\tau^{-1}\,\mathbf{e}_4$	$\mathbf{e}_2$
$\mathbf{e}_4$	$\tau\,\mathbf{e}_1$	$\tau\,\mathbf{e}_2$	$\tau\,\mathbf{e}_3$	$\tau\,\mathbf{e}_4$	$\tau\,\mathbf{e}_5$	$\tau\,\mathbf{e}_6$	$\tau\,\mathbf{e}_7$	$\tau\,\mathbf{e}_8$	$\tau\,\mathbf{e}_9$	$\tau\,\mathbf{e}_{10}$
$\mathbf{e}_5$	$\tau^{-1}\,\mathbf{e}_4$	$\mathbf{e}_8$	$\mathbf{e}_1$	$\tau\,\mathbf{e}_5$	$\mathbf{e}_9$	$\mathbf{e}_2$	$\mathbf{e}_6$	$\mathbf{e}_{10}$	$\mathbf{e}_3$	$\mathbf{e}_7$
$\mathbf{e}_6$	$\mathbf{e}_7$	$\mathbf{e}_3$	$\mathbf{e}_{10}$	$\tau\,\mathbf{e}_6$	$\mathbf{e}_2$	$\mathbf{e}_9$	$\mathbf{e}_5$	$\mathbf{e}_1$	$\mathbf{e}_8$	$\tau^{-1}\,\mathbf{e}_4$
$\mathbf{e}_7$	$\mathbf{e}_{10}$	$\mathbf{e}_9$	$\mathbf{e}_8$	$\tau\,\mathbf{e}_7$	$\mathbf{e}_6$	$\mathbf{e}_5$	$\tau^{-1}\,\mathbf{e}_4$	$\mathbf{e}_3$	$\mathbf{e}_2$	$\mathbf{e}_1$
$\mathbf{e}_8$	$\mathbf{e}_2$	$\tau^{-1}\,\mathbf{e}_4$	$\mathbf{e}_6$	$\tau\,\mathbf{e}_8$	$\mathbf{e}_{10}$	$\mathbf{e}_1$	$\mathbf{e}_3$	$\mathbf{e}_5$	$\mathbf{e}_7$	$\mathbf{e}_9$
$\mathbf{e}_9$	$\mathbf{e}_5$	$\mathbf{e}_{10}$	$\tau^{-1}\,\mathbf{e}_4$	$\tau\,\mathbf{e}_9$	$\mathbf{e}_3$	$\mathbf{e}_8$	$\mathbf{e}_2$	$\mathbf{e}_7$	$\mathbf{e}_1$	$\mathbf{e}_6$
$\mathbf{e}_{10}$	$\mathbf{e}_8$	$\mathbf{e}_5$	$\mathbf{e}_2$	$\tau\,\mathbf{e}_{10}$	$\mathbf{e}_7$	$\tau^{-1}\,\mathbf{e}_4$	$\mathbf{e}_1$	$\mathbf{e}_9$	$\mathbf{e}_6$	$\mathbf{e}_3$

поскольку в этом случае в многочленах открытого ключа потребуется указать не только коэффициенты, но и уникальные наборы произведений переменных, связанные с секретной ТУБВ. Рекомендация, предложенная в [4], по использованию несекретных ТУБВ представляется предпочтительным вариантом. Основное назначение разнообразия ТУБВ, обеспечиваемого предложенными в данной статье унифицированными способами построения ТУБВ, состоит в его использовании как технического приема при разработке алгоритмов многомерной криптографии. В частности, при построении нелинейных отображений, реализуемых как каскады операций экспоненцирования (см. с. 18—19 в [4]) в векторных полях различных типов, задаваемых различными видами ТУБВ для заданного значения m.

Заключение

Предложенные способы построения ТУБВ для задания векторных конечных полей представляют существенный интерес для разработки алгоритмов многомерной криптографии в рамках концепции [4].

Для дальнейшего изучения представляет интерес разработка алгоритмов расстановки структурных констант в ТУБВ, генерируемых по предложенным способам, для случаев больших размерностей. При этом для размерностей $m \geq 10$ актуальным является повышение вычислительной эффективности таких алгоритмов путем уменьшения числа перебираемых вариантов.

Литература

1. *Matsumoto T., Imai H.* Public quadratic polynomial-tuples for efficient signature verification and message-encryption // Advances in Cryptology. Eurocrypt'88 Proceedings. Springer Berlin Heidelberg, 1988. P. 419—453.
2. *Ding J., Petzoldt A.* Current State of Multivariate Cryptography // IEEE Security and Privacy Magazine. 2017. V. 15. № 4. P. 28—36.
3. *Ding J., Petzoldt A., Schmidt D. S.* Multivariate Cryptography. In: Multivariate Public Key Cryptosystems. Advances in Information Security. Springer, New York. 2020. V. 80. P. 7—23. DOI: 10.1007/978-1-0716-0987-3_2.
4. *Молдовян Д. Н.* Альтернативный способ построения алгоритмов многомерной криптографии // Вопросы защиты информации. 2022. № 3. С. 13–21.
DOI: 10.52190/2073-2600_2022_3_13.
5. *Moldovyan N. A., Moldovyanu P. A.* Vector Form of the Finite Fields $GF(p^m)$ // Buletinul Academiei de Stiinte a Republicii Moldova. Matematica. 2009. № 3(61). P. 1—7.
6. *Молдовян Н. А.* Теоретический минимум и алгоритмы цифровой подписи. — СПб: БХВ-Петербург, 2010. — 304 с.
7. *Courtois N., Goubin L., Meier W., Tacier J. D.* Solving Underdefined Systems of Multivariate Quadratic Equations. In: Naccache, D., Paillier, P. (eds) Public Key Cryptography. PKC 2002. Lecture Notes in Computer Science, 2002. V. 2274. Springer, Berlin, Heidelberg. P. 211—227.
DOI: 10.1007/3-540-45664-3_15.
8. *Lokshtano D., Patur R., Tamaki S., Williams R., Yu H.* Beating brute force for systems of polynomial equations over finite fields. In Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA'17, 2190-2202 (Society for Industrial and Applied Mathematics, USA, 2017). DOI: 10.1137/1.9781611974782.143.

Unified methods for defining vector finite fields as primitives of multivariate cryptography algorithms

A. A. Kostina
St. Petersburg Federal Research Center of the RAS (SPC RAS), St. Petersburg, Russia

Unified methods for developing multiplication tables of basis vectors are proposed, with the help of which finite commutative algebras over a ground finite field F, which are extensions of the field F, are defined. The latter allow one to specify nonlinear bijective mappings of vectors defined by a set of polynomials over the field F. The methods can be used in the framework of the recently proposed new concept of constructing multivariate cryptography algorithms, which allows one to significantly reduce the size of the public key at a given level of security compared to the well-known post-quantum digital signature and public encryption algorithms, based on the difficulty of inverting nonlinear mappings given by a system of polynomials over the field F.

Keywords: computer security, post-quantum ciphers, digital signature algorithms, multivariate cryptography, non-linear mappings, bijective mappings, finite algebras, vector finite fields.

Bibliography — 8 references. *Received March 31, 2023*

УДК 004.056.5
DOI: 10.52190/2073-2600_2023_2_9
EDN: DQZYZF

Использование распределения ключей в качестве способа защиты при организации информационного обмена

Е. Е. Филипова, канд. физ.-мат. наук
ФКОУ «Вологодский институт права и экономики ФСИН России», г. Вологда, Россия

Рассмотрены способы повышения защищенности систем передачи данных при использовании способов распределения ключей. Показана необходимость использования распределения ключей при передаче данных, представлено математическое описание способов, дана оценка использования распределения ключей в качестве инструмента защиты информации.

Ключевые слова: информационный обмен, защита информации, протокол, ключ, передача данных, математическое описание, алгоритм, схема Блома.

В современной практике, при работе с конфиденциальной информацией зачастую происходит смена ролей персонала, отвечающего за безопасность информации, хранения ключей шифрования. Возникают ситуации, когда лица, наделенные определенным набором разрешительных функций, при передаче данных вживаются в роль злоумышленника, нанося экономический ущерб организации или учреждению.

Как известно, работа с информацией ограниченного доступа требует выполнения определенных принципов и правил. Информация, имеющая статус ограниченного доступа (конфиденциальная информация) или ее часть не может быть передана третьим лицам без разрешительных процедур, предусмотренных политикой безопасности, принятой в организации или учреждении.

При статическом хранении конфиденциальной информации на нескольких файловых серверах и применения методов защиты, таких как резервное копирование, парольная защита, разграничение доступа, в целом удается обеспечить необходимый уровень безопасности. Вместе с тем, рассматривая "динамическое хранение" заведомо конфиденциальной информации можно утверждать, что в процессе передачи по каналам связи вероятность утери информации или ее части существенно возрастает. Таким образом, вопросы безопасности информации при передаче пакетов данных между абонентами не теряют своей актуальности.

Филипова Елена Евгеньевна, доцент кафедры "Информатика и математика" инженерно-экономического факультета.
E-mail: lenphil@mail.ru

Статья поступила в редакцию 26 апреля 2023 г.

На практике широко используют методы шифрования при передаче пакетов, в том числе методы криптографической защиты [1]. Данные методы направлены, в первую очередь, на обеспечение сохранности самих пакетов данных, файлов или сообщений, при использовании достаточно сложных алгоритмов шифрования, устойчивых к взлому. Здесь стоит отметить о необходимости хранения ключей и обеспечения мер безопасности в случае многократного использования данного способа защиты данных. Потенциально опасные ситуации при работе с ключами, такие как ошибки действия персонала (случайные или преднамеренные), могут представлять серьезную угрозу безопасности, ущерб от которой может быть невосполним, несмотря на превентивные меры и использование современных алгоритмов шифрования.

Интересным представляется следующий способ. При его использовании обмен ключами происходит по следующей схеме: отправитель передает получателям лишь фрагменты ключа, при этом каждый получатель знает только известную часть. При этом, недостающую часть ключа получатель вычисляет самостоятельно, используя заранее отработанный алгоритм. Таким способом можно передавать зашифрованные сообщения, сохраняя высокий уровень безопасности и защиты данных. В этом случае сужается круг лиц, имеющих допуск при работе с информацией, а также упрощается процедура поиска и выявления потенциальных нарушителей.

В источниках данный способ называется схемой Блома [2].

Сущность данной схемы (метода) заключается в следующем. Пусть D — некоторое конечное поле. Допустим, существует некая сеть передачи дан-

ных, включающую n абонентов. Для n абонентов необходимо зафиксировать (подсчитать) n попарно различных значений, удовлетворяющих условию:

$$r_1, \ldots, r_n \in D \setminus \{0\}. \quad (1)$$

Т. е. абоненту A_i соответствует элемент r_i, $i = 1, \ldots, n$.

Как было рассмотрено, данные элементы являются только частью передаваемого сообщения и относятся к общедоступной информации. Следовательно, они могут храниться на серверах организации, облачных дисках хранилищах, персональных устройствах хранения информации.

Построим симметричный многочлен степени m, согласно условию $f(x,y) \in D[x,y]$, $1 \ll m \ll n-1$, он запишется в виде:

$$f(x,y) = \sum_{i=0}^{m}\sum_{j=0}^{m} a_{ij} x^i y^j,\ a_{ij} = a_{ji},\ 0 \ll i \ll j \ll m. \quad (2)$$

При этом коэффициенты a_{ij} многочлена $f(x,y)$ должны быть на безопасном хранении в центре распределения ключей, технически организуемом в виде сервера или группы серверов.

В качестве предполагаемого ключа абоненты получают некий набор $(b_{i0}, \ldots, b_{im})$, который состоит из коэффициентов многочлена, представленного в виде:

$$g_i(x) = f(r_i, x) = b_{i0} + b_{i1}x + \ldots + b_{im}x^m. \quad (3)$$

Организовав условную связь между абонентами A_i и A_j, запишем общий ключ в виде выражения

$$k_{ij} = g_i(r_j) = f(r_i, r_j) = f(r_j, r_i) = g_j(r_i) = k_{ji}. \quad (4)$$

При использовании схемы у каждого абонента хранится $(m + 1)$ вместо $(n - 1)$ "секретных" значений.

Можно утверждать, что данная схема будет устойчива к возможному параллельному хранению секретных значений, т. е. при умышленном сговоре группы абонентов (злоумышленников). Утверждение исходит из самого механизма использования данной схемы: если злоумышленнику станут известны ключевые материалы определенного числа абонентов m, он не сможет получить информации о ключах парной связи остального числа абонентов. Этим достигается сохранность ключей в процессе передачи данных.

Заключение

Использование схем распределения ключей позволяет исключить действия злоумышленников при попытке завладеть данными при их передаче по каналам связи, а также свести к минимуму ущерб, наносимый организациям при утере конфиденциальной информации. Включение в информационный обмен новых абонентов невозможно без разрешительной политики безопасности доверенного центра или центра распределения ключей. Этим обеспечивается защищенность информационной системы в целом.

Литература

1. *Аверченков В. И., Рытов М. Ю., Шпичак С. А.* Криптографические методы защиты информации: учеб. пособие. Изд. 2, стер. — М.: ФЛИНТА, 2017. — 215 с. [Электронный ресурс]. URL: https://znanium.com/catalog/product/1090754 (дата обращения: 26.01.2023).
2. *Рацеев С. М.* Математические методы защиты информации: электронное учеб. пособие. — Ульяновск: УлГУ, 2018. 1 CD-R. № гос. регистрации — 0321901084.

Using Key Distribution as a way to protect the organization of information exchange

E. E. Filipova
Vologda Institute of Law and Economics of the Federal Penitentiary Service of Russia, Vologda, Russia

Ways to increase the security of data transmission systems using methods of key distribution are considered. The necessity of using key distribution in data transmission is shown, a mathematical description of the methods is presented, an assessment is made of the use of key distribution as an information protection tool.

Keywords: information exchange, information protection, protocol, key, data transmission, mathematical description, algorithm, Blom's scheme.

Bibliography — 2 references. *Received April 26, 2023*

УДК 004.056.52
DOI: 10.52190/2073-2600_2023_2_11
EDN: EFMXGL

IT-синтез колориметрического и гиперболического кодирования биометрических характеристик

А. М. Трошков, канд. техн. наук; *А. Н. Ермакова*, канд. эконом. наук;
С. В. Богданова, канд. пед. наук; *А. В. Шуваев*, д-р эконом. наук;
А. Н. Хабаров, канд. техн. наук
ФГБОУ ВО «Ставропольский государственный аграрный университет», г. Ставрополь, Россия

Предложено использование синтеза биометрического защищенного кода на основе анатомической геометрии. Доказана применимость диагонали-асимптоты гиперболы для проектирования биометрического кода, основанного на колоритмии. Для повышения защищенности координатной информации разработана методика использования информационной биометрической матрицы через изменяемые координаты биометрических параметров. Дано обоснование использованию координат гиперболы как сегмента криптографической защиты. Разработан алгоритм криптографического колориметрического биометрического шифрования. Предложен механизм повышения стойкости криптографической цифровой последовательности за счёт выпуклости линий параболы и расположения их в пространстве. В результате спроектирован механизм формирования синтезированного цифрового кода, который может быть применен для управления доступом к информационным ресурсам, а также в системном шифровании.

Ключевые слова: анатомическая геометрия, биометрические параметры человека, гипербола, парабола, биометрический мандат, колориметрическая матрица, синтезированный цифровой код.

Область применения анатомической геометрии в биометрическом кодировании достаточно изучена. Существует множество работ и исследований, которые посвящены этому вопросу. Ожидается, что ее использование будет продолжать расти в будущем. Однако исследования этого вопроса остаются актуальным, так как биометрические технологии становятся все более полезными для создания систем биометрической идентификации высокой точности и надежности, доминантно значимых сегодня в сфере защиты личных данных. Особенно важно это в областях, где требуется высокий уровень безопасности, например, при доступе к финансовым и медицинским данным. Дополнительным эффектом может стать развитие новых приложений и сервисов, которые могут быть полезны в повседневной жизни, например, в системах автоматической идентификации, управления доступом и т. д.

Несмотря на преимущества использования анатомической геометрии в биометрическом кодировании, есть ряд проблем, которые требуют дальнейшей работы и исследований для их решения. Это и сложность сбора и анализа большого объема анатомических данных. Для точного определения формы и положения различных анатомических структур требуется большое количество изображений, что может быть дорого и затруднительно в практических приложениях. И её чувствительность к изменениям внешней среды, таких как освещение и угол съемки. Это может привести к трудностям в преобразовании и анализе данных, что в свою очередь может негативно сказаться на точности биометрического кодирования.

Цель исследования — расширение границ имеющихся знаний о применении анатомической геометрии для разработки более эффективных и устойчивых систем идентификации.

Методология

Основными методами проведенного исследования явились сбор данных и выбор подходящего

Трошков Александр Михайлович, доцент, доцент кафедры "Информационные системы".
E-mail: trochkov1954@mail.ru
Ермакова Анна Николаевна, доцент, доцент кафедры "Информационные системы".
E-mail: dannar@list.ru
Богданова Светлана Викторовна, доцент кафедры "Информационные системы".
E-mail: svetvika@mail.ru
Шуваев Александр Васильевич, профессор, профессор кафедры "Информационные системы".
E-mail: a-v-s-s@rambler.ru
Хабаров Алексей Николаевич, доцент кафедры "Информационные системы".
E-mail: habrw@yandex.ru

Статья поступила в редакцию 21 марта 2023 г.

метода анализа. Авторами проведена детальная анатомическая обработка образцов, получены точные измерения и характеристики для каждой точки. Для обработки данных использован метод главных компонент через многомерный анализ данных в целях снижения размерности данных с минимальной потерей информации. Они стали объективной основой для разработки алгоритма рабочей петли проектирования шифр-кода, что в перспективе может стать фундаментом для создания программной платформы распознавания личности, используя различные метрики, такие, как точность, чувствительность, специфичность и т. д.

Результаты

При проведении анализа методологических контуров нашего исследования выбор остановлен на расчете кодированных платформ биометрических параметров человека с применением геометрических свойств различных фигур и их координат.

В частности, для возможности проектирования биометрического защищенного кода исследуем форму гиперболы, определенную уравнением (1) [1].

$$\frac{x^2}{a^2}-\frac{y^2}{b^2}=1. \qquad (1)$$

Из формулы (1), используя математические правила, утверждаем, что целая гипербола имеет две асимптоты:

$$y=\frac{b}{a}x;\quad y=-\frac{b}{a}x. \qquad (2)$$

Применяя формулы (1) и (2), делаем вывод, что гипербола имеет вид, представленный на рис. 1.

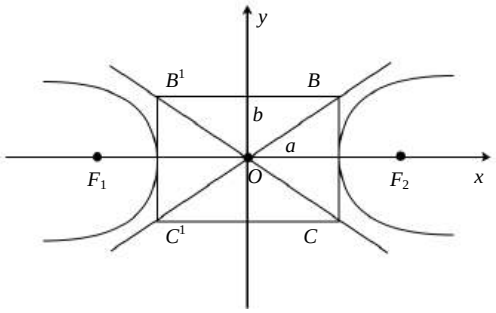


Рис. 1. Двухвершинная гипербола

При построении фокусов гиперболы F_1, F_2 применяем равенство:

$$c^2=a^2+b^2. \qquad (3)$$

Для проектирования биометрического кода, основанного на колоритмии, предлагаем использовать основной прямоугольник гиперболы на рис. 1 BB^1CC^1, сутью которого являются диагонали-асимптоты. Прямоугольник гиперболы, принимаем за колориметрическую матрицу, которая отражает биометрическую характеристику, выбранную за управляемую систему аутентификации/идентификации при доступе/допуске к информационным ресурсам или её сегментам, согласно биометрического мандата, присвоенного пользователю. Рассмотрим форму гиперболы, определённую уравнением

$$-\frac{x^2}{a^2}+\frac{y^2}{b^2}=1. \qquad (4)$$

Используя перестановки x и y, a и b по сравнению с формулой (1), получаем B и B^1, которые лежат на оси Oy на рис. 2.

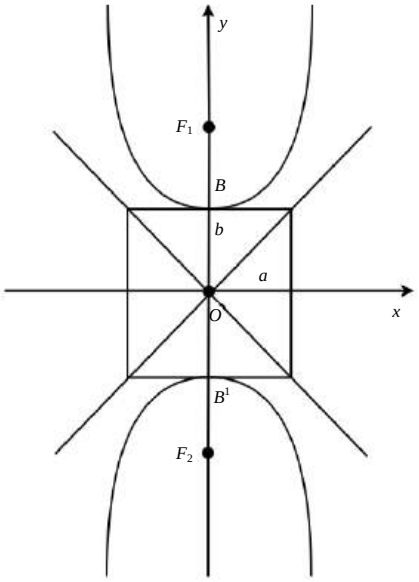


Рис. 2. Каноническая гипербола

Если аналогичным образом прямоугольник использовать как информационную биометрическую матрицу, то изменятся координаты биометрических параметров, что повлияет на повышение защищенности координатной информации.

Кроме того, предлагаем использовать координаты гиперболы как сегмент криптографической защиты.

Для расчёта криптографических элементов выбираем эксцентрису эллипса, которая рассчитываем по формуле

$$e=r/d. \qquad (5)$$

Доказано, что если r — расстояние от произвольной точки гиперболы до некоторого фокуса F_u, d — расстояние от той же точки до фокусной директрисы, то e — const, т. е. вычисление e является несложной операцией. Это вычисление можно

производить, используя различные криптографические методы открытого и закрытого ключа.

Алгоритм криптографического колориметрического биометрического шифрования следующий: F_u — фокус, выбирается цветностью колориметрической матрицы случайным или заданным образом; M — произвольная точка на гиперболе, выбираем цветностью колориметрической матрицы.

Директрису гиперболы предлагается выбирать, исходя из расстояния $\pm\frac{a}{e}$, которое точке задаётся цветностью колорометрической матрицы. Исходя из этого, делаем вывод, что значение F_n, $\pm\frac{a}{e}$, M, r, d — имеют вероятностные значения, которые играют роль в повышении криптографической стойкости. Кроме того, значения x, y, a, b в формулах (1) и (4) также имеют случайный характер, что также будет способствовать повышению стойкости информационной криптографической целостности. Проектируемый алгоритм криптографической последовательности представлен на рис. 3.

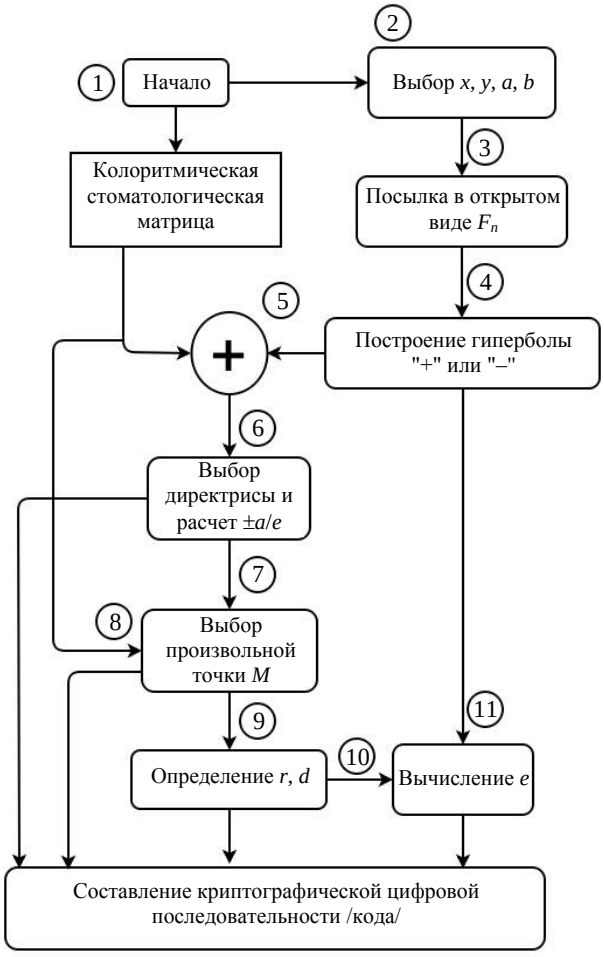


Рис. 3. Алгоритм проектируемой криптографической цифровой последовательности

Стойкость криптографической цифровой последовательности предлагаем повышать за счёт выпуклости линий параболы и расположения их в пространстве, это достигается изменением значений x, y, a, b в уровнях:

$$\frac{x^2}{a^2}-\frac{y^2}{b^2}=1;\quad -\frac{x^2}{a^2}+\frac{y^2}{b^2}=1. \qquad (6)$$

Для повышения стойкости кодируемой цели предлагается ввести геометрическую фигуру эллипс (используется в шифровании на эллиптических кривых) (см. рис. 4).

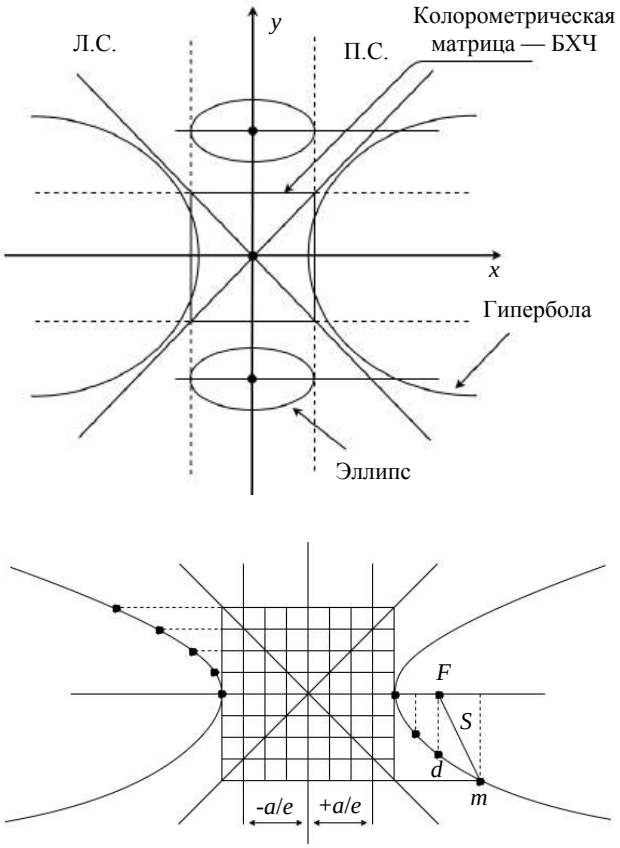


Рис. 4. Геометрическая фигура

Вводим ключ, имеющий три пакета, который предлагаем менять по времени t (рис. 5).

Пакет 1	Пакет 2	Пакет 3
010000	**100000**	**111111**
Проекция влево	Проекция вправо	Проекция эллипс

Рис. 5. Ключевая последовательность

Управление директрис фигуры имеет вид:

$$x = -\frac{a}{e}; \quad x = +\frac{a}{e}. \tag{7}$$

Таким образом, при получении ключа, вычисления производятся на левой или правой стороне гиперболы, а также на эллипсе. Пакеты 1, 2, 3 формулируются по известным криптографическим методикам и могут передаваться как в открытом, так и в закрытом виде.

На основании теоретических исследований предлагаем проект структуры устройства формирования цифрового информационного кода (ЦИК) (рис. 6).

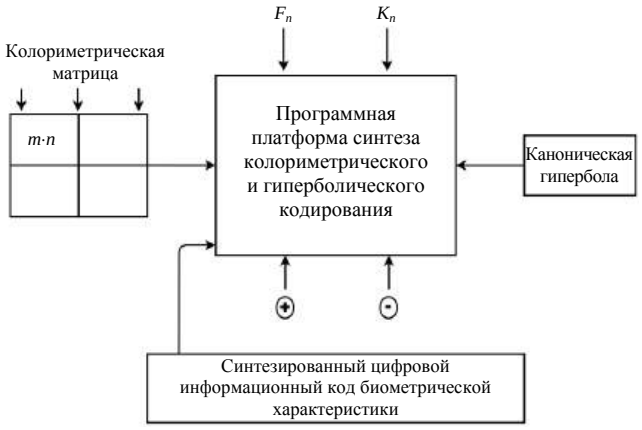


Рис. 6. Проект механизма формирования цифрового информационного кода

Если рассматривать защищенность кода с точки зрения криптографии, то с математической стороны исследуем гиперболу уравнением

$$y = +\frac{b}{a}\sqrt{x^2 - a^2}; \quad x \geq 0. \tag{8}$$

Из формулы (8) видно, что, если $x \to \infty$, то $y \to \infty$. Таким образом точка M (x; y), которая описывает график, движется "вправо" и "вверх", причем удаление точки M по осям Ox, Oy является ∞ (рис. 7).

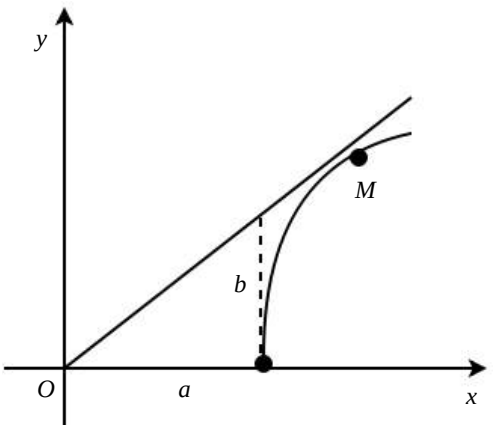


Рис. 7. Графическое движение точки M

Исходя из этого, можно сделать вывод, что точка M имеет вероятностную характеристику, которая сложно определить по координатам, что повышает стойкость кода (ЦИК).

Повысить стойкость криптографического кода предлагаем с применением математических операций и физического решения "задачи Кеплера" [2]. Физический смысл решения этой задачи в следующем: движение точки (•) под действием центральной силы

$$F = \frac{a}{r^3}\, r, \tag{9}$$

где a — const.

Выбор a позволяет судить о кулоновском взаимодействии:

$$\begin{cases} a > 0 & \text{центр } (\bullet) \\ a < 0 & \text{центр } (\bullet) \end{cases} \tag{10}$$

Используя систему (2) регулируем криптографическое начальное состояние. Если траекторию (•) рассматривать, как каноническое сечение в полярных координатах r и P, то систему (10) можно рассматривать, как

$$\begin{cases} r = \dfrac{P}{1 + e\cos f}\,(a > 0) \\ r = \dfrac{P}{-1 + e\cos f}\,(a > 0) \end{cases}, \tag{11}$$

где $P = \dfrac{L^2}{m|a|}$ — параметр нахождения (•);

$e = \sqrt{1 + \dfrac{2W * L^2}{ma^2}}$ — эксцентриситет (•).

При расчете (11) необходимо учитывать W — полную энергию (•), а L — ее момент количества движения относительно выбранного центра. Предлагается в качестве орбиты использовать ранее используемые гиперболу или параболу (рис. 8).

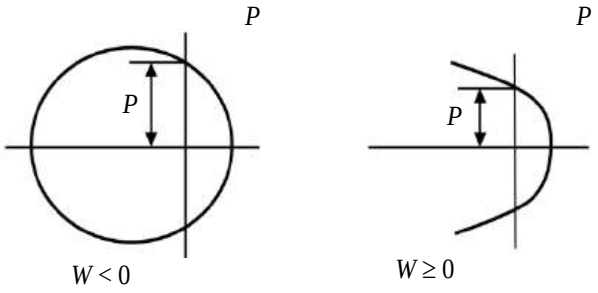


Рис. 8. Эллиптические оси

Если открытым ключом определить гиперболу или параболу, то вычисляем P, а полученный результат позволяет рассчитать a, b:

$$a = \frac{P}{2\left(1-e^2\right)};$$
$$b = \frac{P}{\left(1-e^2\right)}; \quad (12)$$

Значение a и b — это будут начальные криптографические отсчёты. Криптографические отсчеты играют важную роль в стойкости криптографического кода (рис. 9) [3].

При получении значений a и b, информация об их значениях передается на вход в двоичной системе на согласующее устройство, которое переводит их в код суммирования $a + b$, а затем полученная комбинация поступает на вход линейного рекуррентного регистра, который формирует первоначальную криптографическую последовательность на основе заполнения ячеек памяти полученного кода $[a + b]$. После заполнения $[a + b]$ на выходе рекуррентного регистра первоначальная криптографическая последовательность поступает на вход формирователя криптографического кода, который синтезирует шифр-код (ШК), коррекция между основными элементами устройства РР и ФКК осуществляется по цепи обратной связи в обоих направлениях.

Алгоритмическое описание работы устройства имеет вид, представленный на рис. 10.

Разработанный и представленный алгоритм рабочей петли проектирования шифр-кода в перспективе может стать основой для разработки программной платформы обеспечения приведенных выше расчетов. Это позволит осуществлять передачу открытой информации по коммутационным каналам различной физической природы.

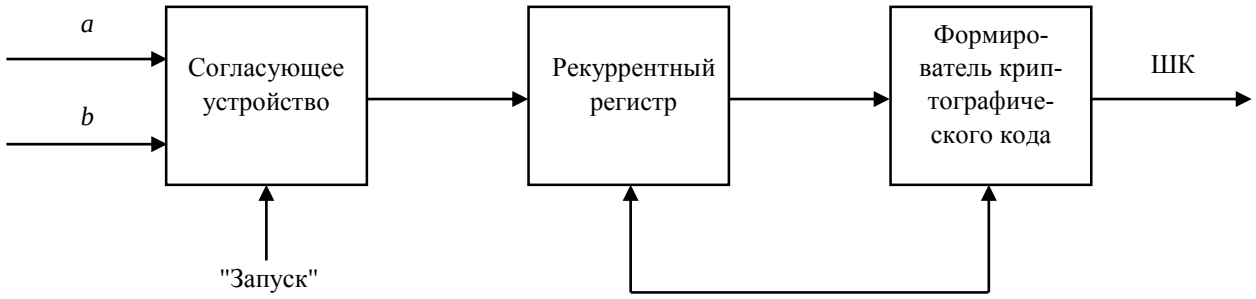


Рис. 9. Устройство, формирующее шифр-код (ШК)

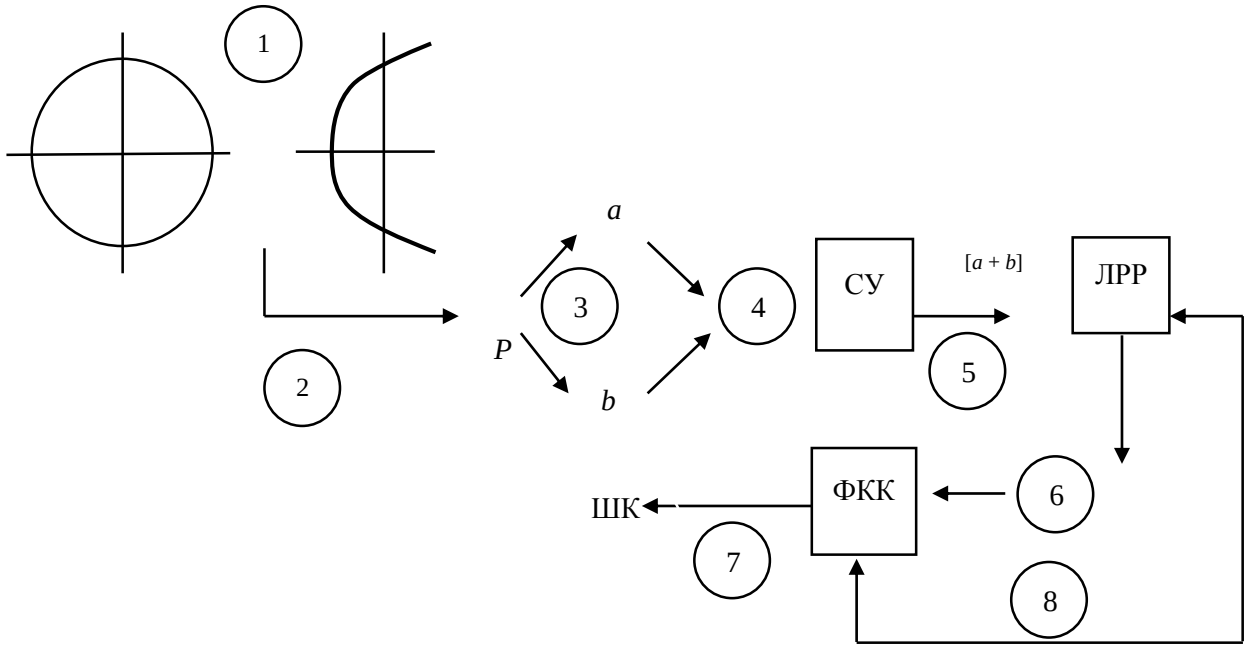


Рис. 10. Алгоритм работы устройства формирования ШК

Заключение

Объективно, что проиллюстрированные подходы, как и любые другие методы, имеют свои ограничения, такие как необходимость точного измерения биометрических параметров и возможность ошибок в расчетах при использовании алгоритмов компьютерной обработки. Однако предложенные в исследовании процедуры позволяют использовать алгоритмы компьютерной обработки отпечатков пальцев, глаз, лица или голоса, которые помогут анализировать уникальные характеристики биометрических данных и создавать уникальный код. Программная платформа с реализованным в ней разработанным в данной работе алгоритмом может быть использована для проверки личности в различных сценариях, например, в аэропорту, в банках или в учреждениях государственной службы и безопасности.

Литература

1. *Ефимов Н. В.* Краткий курс аналитической геометрии. — М.: Изд-во технико-теоретической литературы, 1950.
2. *Яворский Б. М., Детлаф А. А.* Справочник по физике. — М.: Изд-во "Наука". Главная редакция физико-математической литературы, 1974.
3. *Трошков М. А., Трошков А. М., Герасимов В. П., Сапожников В. И.* Метод аутентификации по биометрическим характеристикам и PIN-коду для подтверждения операций мобильного банковского обеспечения // Вестник АПК Ставрополья. 2016. № 1(21). С. 274—279.

IT-synthesis of colorimetric and hyperbolic coding of biometric characteristics

A. M. Troshkov, A. N. Ermakova, S. V. Bogdanova, A. V. Shuvaev, A. N. Habarov
Stavropol State Agrarian University, Stavropol, Russia

A proposal has been made to use the synthesis of a biometric secure code based on anatomical geometry. The applicability of the diagonal asymptote of a hyperbola in designing a biometric code based on colorimetry has been proven. To increase the security of coordinate information, a methodology has been developed for using an information biometric matrix through variable coordinates of biometric parameters. Justification has been given for the use of hyperbola coordinates as a segment of cryptographic protection. An algorithm for cryptographic colorimetric biometric encryption has been developed. A mechanism for increasing the resilience of a cryptographic digital sequence through the convexity of parabolic lines and their location in space has been proposed. As a result, a mechanism for creating a synthesized digital code has been designed that can be used to control access to information resources as well as in system encryption.

Keywords: anatomical geometry, human biometric parameters, hyperbola, parabola, biometric mandate, colorimetric matrix, synthesized digital code.

Bibliography — 3 references. *Received April 6, 2023*

УПРАВЛЕНИЕ ДОСТУПОМ

УДК 004.93'1
DOI: 10.52190/2073-2600_2023_2_17
EDN: ENKYIW

Методы определения живого присутствия пользователя перед видеокамерой в задачах биометрической аутентификации по лицу

И. Е. Панфилова
ФГБОУ ВО "Самарский государственный технический университет", г. Самара, Россия

А. Е. Сулавко, канд. техн. наук
ФГАОУ ВО "Омский государственный технический университет", г. Омск, Россия

Представлен обобщающий обзор методов и технологий, используемых для определения живого присутствия аутентифицируемого субъекта. Среди результатов проведенного анализа можно выделить неуклонную тенденцию смены подходов по определению живого присутствия на основе «ручной» обработки образов на многослойные алгоритмы машинного обучения. Однако анализ подобных алгоритмов показывает, что характерной особенностью их функционирования становится невозможность воспроизведения результатов, полученных при обучении, в реальной практике. Более того, даже незначительные изменения условий процедуры аутентификации для таких алгоритмов становятся критичными с точки зрения робастности всей системы. Возможным решением указанных проблем может стать применение для задач определения живого присутствия методов объяснимого искусственного интеллекта.

Ключевые слова: спуфинг атаки, распознавание лиц, биометрическая аутентификация, глубокое обучение, сверточные нейронные сети, компьютерное зрение.

Биометрия играет ключевую роль в приложениях аутентификации и безопасности. Контроль доступа по лицу, отпечатку пальца или радужной оболочке уже давно существует в повседневной жизни. В последние годы происходит массовое внедрение систем распознавания личности по лицу из-за универсальности и удобства этих систем для пользователей. Однако стоит отметить, что существующие системы распознавания лиц уязвимы с точки зрения так называемых атак на биометрическое предъявление (АБП).

Согласно стандарту ГОСТ Р 58624.3-2019 (ИСО/МЭК 30107-3:2017) [1] любое представление артефакта или биометрического параметра индивида подсистеме сбора биометрических данных в целях нарушения намеченной политики биометрической системы классифицируется как АБП. Чаще для определения подобного типа атак используют термин спуфинг (spoofing attack), пришедший из области сетевой безопасности и понимаемый в контексте биометрии, как попытка обмана системы биометрической идентификации/аутентификации путем предъявления ей поддельного или синтетического биометрического образа.

Особо актуальной становится задача по проектированию и разработке систем, позволяющих обнаруживать спуфинг атаки уже на этапе детекции биометрических образов. В области лицевой биометрии такая постановка задачи сформировала отдельную область исследований — face anti-spoofing (FAS) и liveness detection (обнаружение живости). Второй термин используют чаще, он подразумевает набор методов определения живого присутствия человека в видеопотоке или изображении.

В представленной работе приведен анализ существующих методов обнаружения живого присутствия пользователя, а также обозначены наиболее перспективные из них с точки зрения развития области: глубокое обучение и объяснимый искусственный интеллект.

Панфилова Ирина Евгеньевна, аспирант, инженер.
E-mail: panfilova_2015@bk.ru
Сулавко Алексей Евгеньевич, доцент кафедры "Комплексная защита информации".
E-mail: sulavich@mail.ru

Статья поступила в редакцию 26 апреля 2023 г.

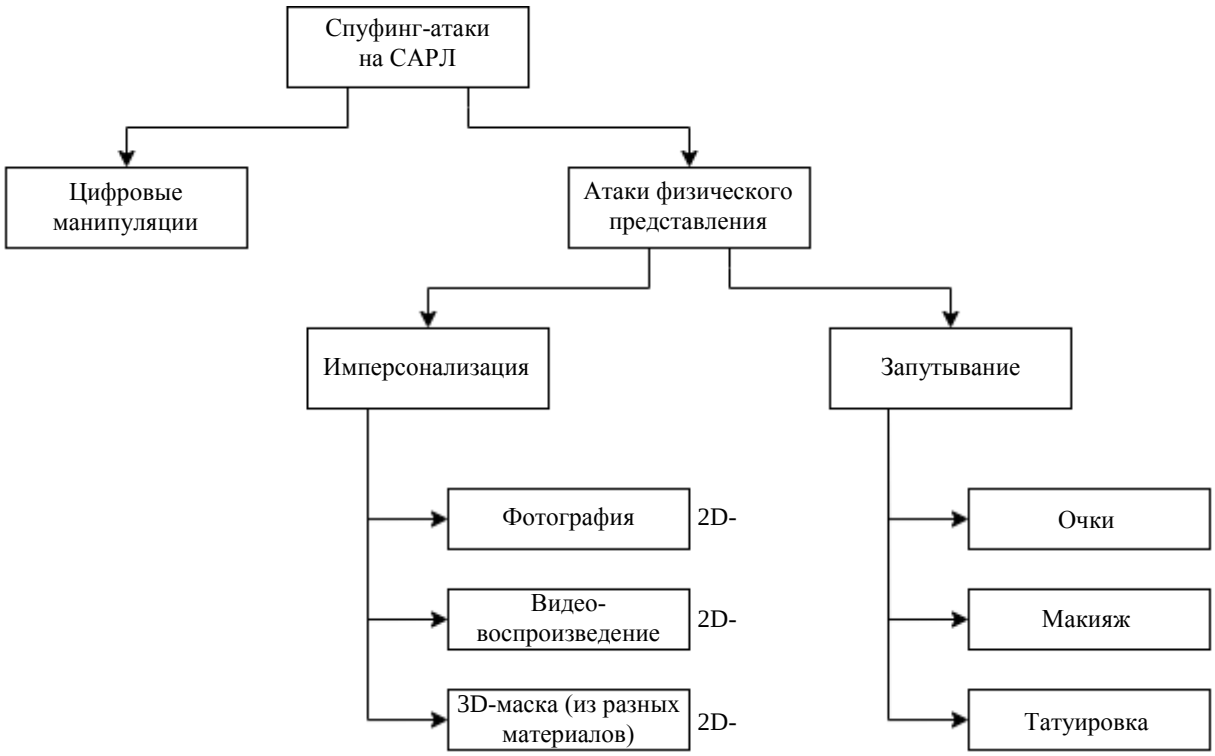
Классификация спуфинг-атак на системы автоматического распознавания лиц (САРЛ)

Спуфинг-атаки представляют собой серьезную угрозу безопасности систем лицевой биометрической аутентификации, поскольку их можно использовать для получения несанкционированного доступа к системе путем выдачи себя за авторизованного пользователя. Для того, чтобы осуществлять защиту САРЛ от подобного рода атак, важно иметь хорошее представление о многообразии подходов к осуществлению спуфинга, а также способах его реализации. В связи с этим в современном исследовательском пространстве сформировалось большое количество терминов и вариантов классификации спуфинг атак на САРЛ. Так, исходя из способа представления поддельных образов в физическом пространстве, выделяют 2D- и 3D-спуфинг атаки [3]. К классу 2D-атак в таком случае относят изображение лица человека, представленное системе с помощью экрана какого-нибудь устройства или фотографии, напечатанной на бумаге или же видеоролика, также воспроизведённого с экрана стороннего устройства. Данный класс атак считается легко воспроизводимым, так как не требует наличия какого-то специального оборудования или обширных знаний о личности жертвы: достаточно получить несколько снимков удовлетворительного качества или непродолжительный видеоролик с участием человека.

В свою очередь, 3D-спуфинг-атаки не имеют четкого разграничения с точки зрения сложности воспроизведения, так как во многом сложность их реализации будет зависеть от самой САРЛ. Для компрометации системы защиты может быть использоваться распечатанная с помощью принтера маска лица жертвы, повторяющая основные его контуры и имеющая трехмерную форму. Данный подход требует высококачественного изображения лица жертвы, но по-прежнему не является трудновоспроизводимым в "домашних условиях". Однако большинство современных САРЛ способны обнаружить атаки этого уровня, поэтому наиболее продвинутыми способами атак становятся 3D-копии лица, воспроизведенные с помощью дополнительного оборудования: отпечатанные 3D-копии лица, театральные маски, силиконовые маски и др. Помимо оборудования, для изготовления такого рода масок требуется подробная информация о пропорциях лица жертвы и особенностях его мимики.

Стоит отметить, что классификация спуфинг-атак на системы распознавания лиц, исходящая из способа представления поддельного образа в физическом пространстве, не является исчерпывающей, так как не учитывает возможности цифровых манипуляций над САРЛ. В работе [2] такой класс угроз выносен в отдельную категорию и подразумевает собой ряд незаметных манипуляций в цифровой структуре видеопотока или изображения САРЛ. Помимо данного класса угроз авторы вводят два дополнительных понятия: имперсонализация (impersonation) и запутывание (obfuscation). В общем случае, имперсонализация включает в себя реализацию атаки путем копирования атрибутов лица подлинного пользователя на специальные носители, такие как фотография, электронный экран и 3D-маска. Таким образом, выделенный класс атак включает в себя как 2D-, так и 3D-реализации, что позволяет обобщить эти подходы в независимости от способа их физического представления. В свою очередь запутывание САРЛ осуществляется путем использования дополнительного реквизита (очков, макияжа, татуировок и т. д.) в целях подстройки внешности злоумышленника под особенности внешности жертвы. Обобщённая структура классификации спуфинг-атак на системы автоматического распознавания лиц представлена на рисунке.

Также, в ряде работ [4, 5] можно встретить понятие атак воспроизведения (replay attack) или атак ретрансляции (rebroadcast attacks). Например, в работе [4] такой тип атак осуществлен с помощью использования уязвимостей сверточных нейронных сетей (СНС), на которых чаще всего и строятся современные САРЛ. Атака реализовывалась исходя из понятия состязательной атаки на СНС, но в сущности являлась классической атакой физического представления. Таким образом, данный тип атак задействует все тот же алгоритм воспроизведения биометрического образа жертвы с помощью дополнительных цифровых устройств, и его можно считать составной частью атак имперсонализации, но, как правило, не включающей в себя 2D-изображения на бумаге (print attack).



Классификация спуфинг-атак на САРЛ

Достигнутые результаты по автоматическому обнаружению живого присутствия пользователя в задачах биометрической аутентификации по лицу

Основываясь на общепринятой классификации, подходы по обнаружению живого присутствия в задачах биометрической аутентификации пользователя по лицу можно разделить на два типа: активное обнаружение и пассивное обнаружение. Активное обнаружение предполагает выполнение пользователем определённых действий, например, отслеживание движущегося объекта на экране, повтора какого-нибудь жеста, произнесения ключевой фразы и т. д. В свою очередь, пассивное обнаружение позволяет детектировать атаку на основе одного кадра с изображением лица. Обычно такой подход позволяет разрабатывать более быстрые и удобные с точки зрения пользователя интерфейсы.

Однако стоит отметить, что в последнее время, как в исследовательской, так и в коммерческой сфере все чаще можно наблюдать выделение отдельного типа технологий — полупассивного обнаружения или гибридных методов реализации liveness detection [6, 45]. Такие способы обнаружения стремятся объединить лучшее из обоих типов детекции живости — высокий уровень безопасности с положительным пользовательским опытом.

Помимо трех типов (пассивное, активное, гибридное определение живости) методов обнаружения живого присутствия можно выделить 5 базовых направлений (в рамках трех перечисленных выше типов).

1. Подходы на основе анализа 2D-изображений.
2. Динамические подходы (на основе анализа последовательности изображений видеопотока).
3. Подходы на основе анализа глубины изображений.
4. Подходы на основе физиологических характеристик человека.
5. Подходы на основе глубокого обучения.

Каждое из перечисленных направлений реализуется для специфического класса угроз и занимает свое место в историческом развитии технологий обнаружения живости.

Поскольку методы защиты систем распознавания лиц от спуфинга обычно базируются на концепции принятия либо отклонения предъявляемого образа лица, то, как и в биометрических системах в целом, для их оценки широко используют два основных показателя: False Acceptance Rate (FAR) и False Rejection Rate (FRR). Альтернативными названиями данных метрик являются ошибка первого рода (вероятность ложного допуска — FAR) и ошибка второго рода (вероятность ложного отказа в допуске — FRR). В общем случае показатели демонстрируют эффективность

и надежность работы биометрической системы и систем типа FAS. Однако чаще всего в литературе можно встретить еще две связанных с ошибками первого и второго рода метрики: Equal Error Rate (EER) и Half Total Error Rate (HTER). Наиболее часто используемым среди них является EER (равная вероятность ошибок), представляющий собой показатель, получаемый при равных значениях ошибок первого и второго рода. Отметим, что EER является частным случаем HTER (половина полной ошибки), который вычисляется как половина от суммы ошибок первого и второго рода.

Анализ двухмерных изображений

Определение поддельного биометрического образа путем применения алгоритмов анализа *текстуры изображения* стало одним из самых первых направлений в этой области [7—9]. Такой подход к реализации liveness detection подразумевает, что поддельные изображения имеют определённые отличия от реального образа с точки зрения текстуры и детализации. Действительно, на распечатанном изображении лица, даже при хорошем разрешении, могут быть обнаружены незаметные глазу изменения текстуры (блики, резкий контраст и прочее). Обнаружить такие отличия, позволяют различные дескрипторы изображений, извлекаемые из полученного кадра. В общем случае дескрипторы используют в задачах компьютерного зрения, она представляют собой описания визуальных особенностей содержимого изображения и его элементарных характеристик (формы, цвета, текстуры и прочего).

В основе подавляющего большинства методов, использующих анализ текстуры изображения, традиционно заложен такой тип дескрипторов изображения, как локальные бинарные паттерны (Local Binary Patterns — LBP). Данный дескриптор используют для описания локальных шаблонов текстуры изображения и вычисляют как двоичное представление разницы интенсивности пикселей в локальной окрестности. Так в работе [10], в которой впервые были рассмотрены возможности применения LBP дескрипторов для задачи определения живого присутствия, авторам удалось достичь достаточно низкого для того времени показателя равного значения ошибок первого и второго рода EER = 2,9 %, а также продемонстрировать преимущества использования данного вида дескрипторов перед его устаревшими аналогами (LPQ-дескрипторами и фильтрами Габора). Подобное исследование проведено в [7].

Использование наряду с LBP-дескрипторами частотного анализа, продемонстрированное в работах [8] и [10], стало следующим этапом развития методов обработки текстуры изображения. В работе [10] изображения прежде чем проходить через алгоритм извлечения LBP-дескрипторов подвергались частотному анализу на основе преобразования Фурье.

Для задач определения живого присутствия в кадре на основе обработки текстуры изображения неоднократно использовались и ряд других дескрипторов, например, SIFT [11], SURF [12], DoG [13] и HOG [14], применяемых для извлечения эффективных паттернов спуфинга из различных цветовых пространств (RGB, HSV и YCbCr). Среди перечисленных подходов отметим работу [14], в которой исследователи не только использовали в качестве дескриптора гистограммы направленных градиентов (HOG — Histogram of Oriented Gradients), но и анализировали *контекст* изображения и его *границы.* Сочетание этих признаков позволило исследователям добиться достаточно низкого показателя EER = 1,1 % на первых двух классических сценариях спуфинг-атак набора данных CASIA Anti-Spoofing.

К рассмотренному классу подходов анализа текстуры изображения можно также отнести исследование [15], посвящённое обнаружению муаров — специфических узоров цифровых изображений, вызванных наложением двух разверсток. Несмотря на ограниченность применения (муары наблюдаются только на изображениях, полученных при демонстрации экрана цифрового устройства), этот метод, основанный на двух дескрипторах (LBP и Dense SIFT), также позволяет получить достаточно высокие показатели HTER = 6% (на наборах данных DIAP, CASIA и RAFS).

Динамические подходы

Наряду с анализом отдельных изображений в области liveness detection достаточно давно сформированы так называемые динамические подходы, основанные на последовательном анализе видеопотока. Среди них классическими являются направления, связанные с обнаружением микродвижений [16] и моргания [17, 18]. Так, в работе [17] для определения подлинности лица применяли метод условных случайных полей, CRF (Conditional Random Fields), являющийся разновидностью метода Марковских случайных полей (Markov random field). В сущности, метод применяют для решения задачи бинарной классификации (глаза открыты и глаза закрыты) на основе предположения о временной зависимости последовательных кадров.

Кроме того, к классу работ, основанных на динамическом подходе, также относятся исследования, посвященные анализу изменений оптического потока [16, 19]. В основном такие работы основаны на предположении о том, что поля оптического потока реального и поддельного образов имеют значимые отличия, которые можно зарегистрировать на "плоских" изображениях.

Анализ глубины изображения

Потенциально более эффективными с точки зрения предотвращения как 2D-, так и 3D-спуфинг атак, являются подходы, основанные на работе с глубиной изображения или видеофрагмента [23, 24]. Например, при оценке времени, необходимого свету для прохождения расстояния от объекта обратно к камере, можно вычислить непосредственное расстояние до объекта и сформировать попиксельную карту глубины изображения. Однако такой подход обладает вполне очевидным недостатком — для его осуществления необходимо наличие дополнительного оборудования, что не всегда является возможным для приложений биометрической аутентификации/идентификации.

Оценка глубины по одному RGB-изображению — фундаментальная проблема компьютерного зрения. В последние годы многие исследователи обучают глубокие нейронные сети на больших наборах данных RGB-D для решения этой задачи. В частности, реконструкция трехмерной модели лица по одному двухмерному изображению [25] или нескольким двухмерным изображениям [26] также может рассматриваться как один из способов оценки глубины.

Отметим также, что в ряде исследований для оценки глубины изображений используются специфические дескрипторы формы лица [28, 29], а иногда производится трехмерная реконструкция лица [27] в целях извлечения признаков, позволяющих отличить реальное лицо и 3D-маску. Эти методы требуют только обычных цветных изображений без необходимости использования специальных датчиков. Тем не менее, их эффективность может снижаться при атаках с использованием высококачественных 3D-масок лица.

Анализ физиологических характеристик человека

Помимо всех перечисленных подходов, в области определения подлинности субъекта можно выделить отдельное направление, основанное на непосредственном определении "живости" субъекта путем измерения показателей, характерных для живого человека, таких как частота сердечных сокращений (ЧСС) [20, 21] или температура лица [22] (последнее подразумевает использование тепловизора или датчика инфракрасного спектра излучений). Так, работы, посвященные измерению ЧСС, основаны на технике фотоплетизмографии и строятся на различных вариациях методов бесконтактного определения импульсов сердцебиения по видеоизображению в видимой части спектра. Например, в сочетании со сверточными нейронными сетями, такие методы способны достичь значения EER, близкие или равные нулю [21], а корреляция с контрольным сигналом может достигать 0,98. При этом стоит отметить, что современная техника определения вариабельности сердечного ритма по видеоизображению не требует никакого дополнительного оборудования (так как для такой задачи вполне подходит обычная RGB камера), в отличие от методов на основе инфракрасного излучения.

Использование методов глубокого обучения многослойных нейронных сетей

В большинстве работ, посвященных определению живого присутствия с использованием методов глубокого обучения, решение данной проблемы сводится к задаче бинарной классификации [30—32]. В таком случае классификация производится по принципу разделения лиц, поступающих на вход алгоритма, на два класса: фальшивое лицо или реальное.

Многие задачи компьютерного зрения (например, классификация пола человека) в значительной степени полагаются на очевидные подсказки, основанные на внешности (например, прическа, одежда, форма лица), тогда как в задачах определения живого присутствия обычно используются нерелевантные (например, не связанные с чертами лица), малозаметные паттерны, которые сложно различить даже человеческому глазу. В связи с этим, сверточные нейронные сети способны не только выделять значимые признаки из наблюдаемого объекта, но и из текстуры и цветового наполнения изображения. СНС широко используют для определения живого присутствия в задачах биометрической аутентификации пользователя по лицу. Эту категорию решений условно можно разделить на 3 группы [2]:

- Гибридные методы: извлечение классических признаков с последующим применением глубокого обучения.
- Традиционные методы обучения с учителем: так называемые end-to-end решения. Определение живого присутствия осуществляется путем приме-

нения исключительно методов глубокого обучения, чаще всего одной глубокой нейронной сети.

- Обобщенные методы машинного обучения: подразумевают обобщение модели глубокого обучения на как можно большее число наборов данных, а также на данные, неиспользовавшиеся в процессе обучения.

Действительно, некоторые подходы к обнаружению живого присутствия сначала извлекают созданные вручную признаки из входных данных лица, а затем используют глубокое обучение для представления семантических функций. Основываясь на богатом низкоуровневом наборе текстур, глубокая модель способна извлекать семантические подсказки с учетом текстуры. С этой целью авторы работы [33] используют LBP в качестве локальных дескрипторов текстуры, а затем используют случайный лес для семантического представления имеющихся признаков. Отметим, что исследователи не применяют CHC и при этом демонстрируют достаточно высокую эффективность на примере эталонного набора данных IDIAP REPLAY-ATTACK.

Авторы [34] предлагают первое полноценное решение систем определения живого присутствия на основе глубокого обучения с использованием 8-слойной неглубокой CNN для представления признаков. Также достаточно часто можно встретить работы [35—37], в которых предварительно обученные модели ImageNet (например, VGG16, ResNet18) настраиваются специально для задач определения живого присутствия.

Все больше исследователей сосредотачиваются на повышении обобщающей способности своих моделей. Показательными с этой точки зрения являются работы [38] и [39]. Исследователи [38] первыми предложили изучить обобщенное пространство признаков, совместно используемое несколькими разными наборами данных, с помощью многосостязательной дискриминационной структуры обобщения предметной области (Multi-adversarial discriminative Deep Domain Generalization). Авторы [39] разработали сеть глубокого дерева (DTN) для изучения семантических атрибутов предопределенных атак и разделения выборок фальсифицированных изображений лиц на семантические подгруппы. Основываясь на сходстве входных признаков, DTN адаптивно направляет известные или неизвестные атаки в соответствующие кластеры.

Известно множество случаев, когда результат, полученный на одном наборе данных, не получается перенести на другой набор данных (тестирование обученной модели в реальных условиях или на другом наборе данных приводит к значительному снижению показателей эффективности). Более того, согласно последним исследованиям [2], значительная часть работ по-прежнему опирается на небольшой пул устаревших наборов данных, которые не могут обеспечить решение задачи определения подлинного присутствия средствами глубокого обучения. Для решения обозначенной задачи применяются различные подходы на основе обучения сразу на нескольких наборах данных (cross-database) и применения специальных протоколов ("intra-dataset intra-type" и "cross-dataset intra-type") реализации атак.

Перспектива применения подходов на основе объяснимости и интерпретируемости искусственного интеллекта

Одним из наиболее перспективных направлений в задачах определения живого присутствия при биометрической аутентификации по лицу сегодня считается применение технологий объяснимости и интерпретируемости искусственного интеллекта (ИИ) [40]. Объяснимый ИИ (Explainable Artificial Intelligence, XAI) позволяет не только получить результат распознавания образа или предсказания, но и дополнительную информацию о том, почему ИИ сделал такое решение, чем оно обусловлено. Эта концепция противоположна концепции "черного ящика", с которым можно сравнить многослойные нейронные сети, когда невозможно объяснить, почему результат работы алгоритма оказался именно таким. Важно отметить, что существует разница между интерпретируемостью и объяснимостью модели ИИ [44]. Интерпретация — это смысл прогнозов, в то время как объяснимость — это то, почему модель предсказывает что-либо и почему кто-то должен доверять модели. Интерпретация позволяет перевести язык объяснимого ИИ на язык, понятный не только инженеру по машинному обучению, но и рядовому пользователю. Так, авторы двух работ [41, 42] по определению живого присутствия в задачах биометрической аутентификации по лицу выбрали именно подход на основе интерпретируемости и применили в своих работах Grad-CAM, технологию, позволяющую получать объяснения для конкретных классов и предоставлять объяснения для каждого слоя сети. Альтернативным интерпретируемости является так называемая оценка по полноте [43], которая сосредоточена на максимально точном описании работы ИИ с математической точки зрения: например, в сети с глубоким обучением раскрытие всех математических операций

с использованием весовых коэффициентов был совершенно полным объяснением.

Методы объяснимости и интерпретируемости в задачах биометрической аутентификации по лицу позволяют исследователям получать представление о значимых признаках именно с точки зрения определения поддельных образов лица, а также визуализировать их с помощью специальных библиотек. Большинство существующих Фреймворков для создания объяснимого ИИ позволяют оценить информативность различных признаков и их влияние на конечный результат предсказаний (например, SHAP, ELI5 и др.). Таким образом, они дают возможность найти информативные признаки с точки зрения распознавания спуфинг-атаки, а не верификации образов конкретного человека.

Результаты исследований перечисленных выше дескрипторов представлены в таблице.

Сводная таблица результатов исследований

Наименование исследования	Год проведенного исследования	Подход (согласно представленной в работе классификации)	Тип спуфинг-атак (2D/3D)	Метод, лежащий в основе подхода	Метрика оценки эффективности проведенного исследования
Face spoofing detection from single images using micro- texture analysis [9]	2011	Анализ текстуры изображения	2D-атаки	LBP дескрипторы	EER = 2,9 %
On the Effectiveness of Local Binary Patterns in Face Anti-spoofing [7]	2012	Анализ текстуры изображения	2D-атаки	LBP дескрипторы	HTER = 15 %
Face liveness detection based on texture and frequency analyses [8]	2012	Анализ текстуры изображения	2D-атаки	LBP дескрипторы	EER = 12,46 %
Face Liveness Detection Based on Frequency and Micro-Texture Analysis [10]	2014	Анализ текстуры изображения	2D-атаки	LBP дескрипторы	EER = 2,7 %
Secure Face Unlock: Spoof Detection on Smartphones [11]	2016	Анализ текстуры изображения	2D-атаки	SIFT дескрипторы	EER = 3,51 %
Context based Face Anti-Spoofing [14]	2013	Анализ текстуры изображения и его контекста	2D-атаки	HOG дескрипторы	EER = 1,1 %
Live Face Video vs. Spoof Face Video: Use of Moire Patterns to Detect Replay Video Attacks [15]	2015	Анализ текстуры изображения (муары)	2D-атаки	LBP дескрипторы + Dense SIFT дескрипторы	HTER = 6 %
Motion-based counter-measures to photo attacks in face recognition [16]	2012	Динамический подход (оптический поток)	2D-атаки и некоторые разновидности 3D-атак	Метод на основе корреляции движений переднего и заднего фонов	HTER = 1,52 %
Generalized face anti-spoofing by detecting pulse from face videos [20]	2016	Подход на основе бесконтактного определения пульса (ЧСС)	3D-атаки	Метод бесконтактного определения (ЧСС)	EER = 1,58 %
Face liveness detection by rPPG features and contextual patch-based CNN [21]	2019	Подход на основе бесконтактного определения пульса (ЧСС)	2D- и 3D-атаки	Сверточная нейронная сеть	EER = 3,4 %
Visible/Infrared face spoofing detection using texture descriptors [22]	2019	Подход на основе инфракрасного излучения	2D-атаки	Сверточная нейронная сеть	EER = 1,01 %
Face Spoofing Detection Based on Depthmap and Gradient Binary Pattern [23]	2015	Подход на основе глубины изображения	2D-атаки	Метод на основе градиентных бинарных паттернов	EER = 31 %
Face Anti-Spoofing Using Patch and Depth-Based CNNs [24]	2017	Подход на основе глубины изображения	2D- и 3D-атаки	Сверточная нейронная сеть	EER = 0,1 %

Окончание табл.

Наименование исследования	Год проведенного исследования	Подход (согласно представленной в работе классификации)	Тип спуфинг-атак (2D/3D)	Метод, лежащий в основе подхода	Метрика оценки эффективности проведенного исследования
Face anti-spoofing to 3d masks by combining texture and geometry features [27]	2018	Подход на основе реконструкции 3D модели лица	3D-атаки	Трехмерная морфологическая модель + сверточная нейронная сеть	EER = 0,9 %
3d facial geometric attributes based anti-spoofing approach against mask attacks [29]	2017	Подход на основе оценки глубины изображения с помощью специальных дескрипторов	3D-атаки	Дескрипторы на основе meshSIFT	EER = 6,72 %
An original face anti-spoofing approach using partial convolutional neural network [30]	2016	Подход на основе машинного обучения	2D-атаки	Частичная сверточная нейронная сеть	EER = 2,9 %
Cross-database face anti-spoofing with robust feature representation [31]	2016	Подход на основе машинного обучения	2D-атаки	Сверточная нейронная сеть в качестве модуля извлечения признаков	HTER = 0,5 %
Deep pixel-wise binary supervision for face presentation attack detection [32]	2019	Подход на основе машинного обучения	2D-атаки	Сверточная нейронная сеть	HTER = 12 %
Learning deep forest with multi-scale local binary pattern features for face anti-spoofing [33]	2019	Подход на основе машинного обучения	2D- и 3D-атаки	Глубокий лес	EER = 1,56 %
Learn convolutional neural network for face anti-spoofing [34]	2014	Подход на основе машинного обучения	2D-атаки	Сверточная нейронная сеть	HTER = 6 %

Исходя из представленной в таблице информации, можно сделать вывод: робастные модели ИИ для извлечения признаков на основе антропометрических знаний о человеке приходят на смену более сложным моделям, которые учатся самостоятельно искать и извлекать признаки, основываясь полностью на анализе наборов данных. Такие модели позволяют автоматизировать процесс извлечения признаков и повысить быстродействие работы алгоритмов. В последние года отмечается рост применения в задачах определения "живого присутствия" сверточных нейронных сетей. Такая популярность СНС во многом обоснована их широкими возможностями работы с изображениями, видео- и их контекстом.

Заключение

Проведенный обзор методов и подходов к реализации технологии liveness detection демонстрирует неуклонную тенденцию смены традиционных алгоритмов "ручного" извлечения признаков (классический анализ 2D-изображений, их текстуры или последовательности), популярных в начале века, на сложные и многослойные алгоритмы машинного обучения. Абсолютным лидером в области таких алгоритмов выступают сверточные нейронные сети, хорошо зарекомендовавшие себя в области компьютерного зрения. Значительная часть работ посвящена различным модификациям архитектур СНС, способам их обучения и поиска оптимальных параметров.

За последние годы удалось повысить количество потенциально возможных спуфинг-атак, которые могут быть детектированы специализированными моделями машинного обучения. Однако основной проблемой остается сложность повторения высоких результатов, полученных на тестовых наборах данных, в реальной практике. Даже незначительные отличия условий (оборудование, угол съемки, техника спуфинга) сильно снижают робастность модели ИИ. Воспроизвести поведение модели ИИ в реальных условиях, показав высокий результат, как на тестовом наборе данных, затруднительно. Возможно, эта проблема связана с недостаточной репрезентативностью наборов данных подделок лиц, так как наивысшие результаты удается получить на основе многослойных нейронных сетей, склонных к переобучению. На самом деле такие модели могут иметь низкую способность к

обобщению, демонстрируя работоспособность только на определенных наборах данных.

Среди возможных подходов к решению данной проблемы можно выделить применение методов объяснимого искусственного интеллекта, позволяющих исследователям выполнять поиск информативных признаков, которые характеризуют легитимность (живое присутствие) пользователя в целом, без привязки к конкретным параметрам определенного человека.

Работа выполнена ОмГТУ в рамках государственного задания Минобрнауки России на 2023—2025 годы (FSGF-2023-0004).

Литература

1. ГОСТ Р 52633.5-2011. Защита информации. Техника защиты информации. Автоматическое обучение нейросетевых преобразователей биометрия-код доступа. — М.: Стандартинформ, 2013. — 16 с.

2. *Zitong Yu, Yunxiao Qin, Xiaobai Li, Chenxu Zhao, Zhen Lei, Guoying Zhao.* Deep Learning for Face Anti-Spoofing: A Survey // IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI). 2022. P. 5609—5631. DOI: 10.1109/TPAMI.2022.3215850.

3. *Galbally J., Satta Ro.* Three-dimensional and two-and-a-half dimensional face recognition spoofing using // IET Biometrics. 2015. V. 5(2). P. 1—9. DOI: 10.1049/iet-bmt.2014.0075.

4. *Zhang B., Tondi B., Barni M.* Adversarial examples for replay attacks against CNN-based face recognition with anti-spoofing capability // Computer Vision and Image Understanding, 2020. P. 197—198. DOI: 10.1016/j.cviu.2020.102988.

5. *Li L., Feng X., Boulkenafet Z., Xia Z., Li M., Hadid A.* An original face anti-spoofing approach using partial convolutional neural network // Sixth International Conference on Image Processing Theory, Tools and Applications (IPTA). 2016. P. 1—6. DOI: 10.1109/IPTA.2016.7821013.

6. *Золотарев В. В., Поважнюк А. О., Маро Е. А.* Методы усиления процедуры идентификации пользователей на основе технологии liveness detection // Научно-технический и прикладной журнал "Известия ЮФУ. Технические науки". 2022. № 2. С. 212—225.

7. *Chingovska I., Anjos A., Marcel S.* On the Effectiveness of Local Binary Patterns in Face Anti-spoofing // IEEE International Conference of the Biometrics Special Interest Group (BIOSIG). 2012. P. 1—7.

8. *Kim G., Eum S., Suhr J. K., Kim D. I., Park K. R., Kim J.* Face liveness detection based on texture and frequency analyses // 5th IAPR International Conference on Biometrics (ICB), New Delhi, India. 2012. P. 67—72. DOI: 10.1109/ICB.2012.6199760.

9. *Maatta J., Hadid A., Pietikainen M.* Face spoofing detection from single images using microtexture analysis // Proc. International Joint Conference on Biometrics (UCB). 2011. P. 1—7. DOI: 10.1109/IJCB.2011.6117510.

10. *Das D., Chakraborty S.* Face liveness detection based on frequency and micro-texture analysis // International Conference on Advances in Engineering & Technology Research (ICAETR). 2014. P. 1—4. DOI: 10.1109/ICAETR.2014.7012923.

11. *Patel K., Han H., Jain A. K.* Secure Face Unlock: Spoof Detection on Smartphones // IEEE Transactions on Information Forensics and Security. 2016. V. 11. № 10. P. 2268—2283. DOI: 10.1109/TIFS.2016.2578288.

12. *Boulkenafet Z., Komulainen J., Hadid A.* Face Anti-Spoofing using Speeded-Up Robust Features and Fisher Vector Encoding // IEEE Signal Processing Letters. 2016. P. 141—145. DOI: 10.1109/LSP.2016.2630740.

13. *Tan X., Li Y., Liu J., Jiang L.* Face Liveness Detection from a Single Image with Sparse Low Rank Bilinear Discriminative Model // Lecture Notes in Computer Science. 2010. P. 504—517. DOI: 10.1007/978-3-642-15567-3_37.

14. *Komulainen J., Hadid A., Pietikainen M.* Context based face anti-spoofing // IEEE Sixth International Conference on Biometrics: Theory, Applications and Systems (BTAS). 2013. P. 1—8. DOI: 10.1109/BTAS.2013.6712690.

15. *Patel K., Han H., Jain A. K., Ott G.* Live face video vs. spoof face video: Use of moiré patterns to detect replay video attacks // International Conference on Biometrics. 2015. P. 98—105. DOI: 10.1109/ICB.2015.7139082.

16. *Anjos A., Chakka M. M., Marcel S.* Motion-based countermeasures to photo attacks in face recognition // IET Biometrics. 2014. V. 3(3). P. 147—158. DOI:10.1049/iet-bmt.2012.0071.

17. *Pan G., Sun L., Wu Z., Lao S.* Eyeblink-based anti-spoofing in face recognition from a generic webcamera // ICCV. 2007. P. 1—8. DOI: 10.1109/ICCV.2007.4409068.

18. *Li J.-W.* Eye blink detection based on multiple gabor response waves // IEEE ICMLC. 2008. V. 5. P. 2852—2856. DOI: 10.1109/ICMLC.2008.4620894.

19. *Wei Bao, Hong Li, Nan Li, Wei Jiang.* A liveness detection method for face recognition based on optical flow field // In Image Analysis and Signal Processing (IASP). 2009. P. 233—236. DOI: 10.1109/IASP.2009.5054589.

20. *Li X., Komulainen J., Zhao G., Yuen P.-C., Pietikainen M.* Generalized face anti-spoofing by detecting pulse from face videos // ICPR. 2016. P. 4244—4249. DOI: 10.1109/ICPR.2016.7900300.

21. *Lin B., Li X., Yu Z., Zhao G.* Face liveness detection by rPPG features and contextual patch-based CNN // ICBEA. ACM. 2019. DOI: 10.1145/3345336.3345345.

22. *Mohamed S., Ghoneim A., Youssif A.* Visible/Infrared face spoofing detection using texture descriptors // MATEC Web of Conferences. 2019. — 5 p. DOI: 10.1051/matecconf/201929204006.

23. *Roomi M., Beham M. P., Dharmalakshmi D.* Face spoofing detection based on depthmap and gradient binary pattern // International Journal of Applied Engineering Research. 2015. V. 9(21). P. 4990—4996.

24. *Atoum Y., Liu Y., Jourabloo A., Liu X.* Face anti-spoofing using patch and depth-based CNNs // IEEE International Joint Conference on Biometrics (IJCB). 2017. P. 319—328. DOI: 10.1109/BTAS.2017.8272713.

25. *Jourabloo A., Liu X.* Large-pose face alignment via CNN-based dense 3D model fitting: proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016. P. 4188—4196. DOI: 10.1109/CVPR.2016.454.

26. *Roth J., Tong Y., Liu X.* Adaptive 3D face reconstruction from unconstrained photo collections: proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016. P. 4197—4206. DOI: 10.1109/CVPR.2016.455.

27. *Wang Y., Chen S., Li W., Huang D., Wang Y.* Face anti-spoofing to 3D masks by combining texture and geometry features // Chinese Conference on Biometric Recognition. Springer. 2018. P. 399—408.

28. *Kose N., Dugelay J.-L.* On the vulnerability of face recognition systems to spoofing mask attacks // IEEE International Conference Acoustics, Speech and Signal Processing (ICASSP). 2013. P. 2357—2361. DOI: 10.1109/ICASSP.2013.6638076.

29. *Tang Y., Chen L.* 3d facial geometric attributes based anti-spoofing approach against mask attacks // 12th IEEE International Conference Automatic Face & Gesture Recognition (FG 2017). 2017. P. 589—595. DOI: 10.1109/FG.2017.74.

30. *Li L., Feng X., Boulkenafet Z., Xia Z., Li M., Hadid A.* An original face anti-spoofing approach using partial convolutional neural network // IPTA. 2016. P. 1—6. DOI: 10.1109/IPTA.2016.7821013.

31. *Patel K., Han H., Jain A. K.* Cross-database face anti-spoofing with robust feature representation // CCBR. 2016. P. 611—619.

32. *George A., Marcel S.* Deep pixel-wise binary supervision for face presentation attack detection // ICB. CONF. 2019. P. 1—8. DOI: 10.1109/ICB45273.2019.8987370.

33. *Cai R., Chen C.* Learning deep forest with multi-scale local binary pattern features for face anti-spoofing // arXiv preprint. 2019. P. 1—12.

34. *Yang J., Lei Z., Li S. Z.* Learn convolutional neural network for face anti-spoofing // arXiv preprint. 2014. — 8 p. https://doi.org/10.48550/arXiv.1408.5601.

35. *Lucena O., Junior A., Moia V., Souza R., Valle E., Lotufo R.* Transfer learning using convolutional neural networks for face anti-spoofing // ICIAR. 2017. P. 27—34. DOI: 10.1007/978-3-319-59876-5.

36. *Chen H., Hu G., Lei Z., Chen Y., Robertson N. M., Li S. Z.* Attention-based two-stream convolutional networks for face spoofing detection // TIFS. 2019. V. 15. P. 578—593. DOI: 10.1109/TIFS.2019.2922241.

37. *George A., Marcel S.* On the effectiveness of vision transformers for zero-shot face anti-spoofing // arXiv preprint. 2020. — 8 p.

38. *Shao R., Lan X., Li J., Yuen P. C.* Multi-adversarial discriminative deep domain generalization for face presentation attack detection // CVPR. 2019. P. 10015—10023. DOI: 10.1109/CVPR.2019.01026.

39. *Liu Y., Stehouwer J., Jourabloo A., Liu X.* Deep tree learning for zero-shot face anti-spoofing // CVPR. 2019. P. 4675—4684. DOI: 10.1109/CVPR.2019.00481.

40. *Khairnar S., Gite Sh., Kotecha K., Thepade S.* Face liveness detection using artificial intelligence techniques: A systematic literature review and future directions // Big Data Cogn. Comput. 2023. — 37 p. DOI: 10.3390/bdcc7010037.

41. *Selvaraju R. R., Cogswell M., Das A., Vedantam R., Parikh D., Batra D.* Grad-cam: Visual explanations from deep networks via gradient-based localization // ICCV. 2017. P. 618—626. DOI: 10.1109/ICCV.2017.74.

42. *Wilson Silva, Tiago Filipe Sousa Gonçalves, Ana Sequeira, João Ribeiro Pinto.* Explainable Artificial Intelligence for Face Presentation Attack Detection // 26th Portuguese Conference in Pattern Recognition (RECPAD). 2020.

43. *Murilo Leite Nóbrega.* Explainable and Interpretable Face Presentation Attack Detection Methods // Mestrado Integrado em Engenharia Eletrotécnica e de Computadores. 2021.

44. *Мишра П.* Объяснимые модели искусственного интеллекта на Python. Модель искусственного интеллекта. Объяснения с использованием библиотек, расширений и фреймворков на основе языка Python / пер. с англ. Минца С. В. — М.: ДМК Пресс, 2022. — 298 с.

45. *Gao C., Li X., Zhou F., Mu S.* Face Liveness Detection Based on the Improved CNN with Context and Texture Information // Chinese Journal of Electronics. 2019. V. 28(6). P. 1092—1098. DOI: 10.1049/cje.2019.07.012

Methods for determining the live presence of a user in front of a video camera in the task of biometric face authentication

I. E. Panfilova
Samara State Technical University, Samara, Russia

A. E. Sulavko
Omsk State Technical University, Omsk, Russia

This paper provides a general overview of the methods and technologies of liveness detection. Among the results of the analysis, we can single out a steady trend of changing approaches to determining live presence based on "manual" processing of images to multilayer machine learning algorithms. However, the analysis of such algorithms shows that a characteristic feature of their functioning is the impossibility of reproducing the results obtained during training in real practice. Moreover, even minor changes in the conditions of the authentication procedure for such algorithms become critical in terms of the robustness of the entire system. A possible solution to these problems can be the use of explainable artificial intelligence methods for the tasks of liveness detection.

Keywords: spoofing attacks, face recognition, biometric authentication, deep learning, convolutional neural network, computer vision.

Bibliography — 45 references. *Received April 26, 2023*

УДК 004.056.52
DOI: 10.52190/2073-2600_2023_2_27
EDN: EYGDDI

Разработка алгоритма колориметрического шифрования в системах биометрии

А. М. Трошков, канд. техн. наук; *М. А. Трошков*, канд. техн. наук;
А. Н. Ермакова, канд. эконом. наук; *С. В. Богданова*, канд. пед. наук;
А. В. Шуваев, д-р эконом. наук
ФГБОУ ВО «Ставропольский государственный аграрный университет», г. Ставрополь, Россия

Доказана пригодность применения спектра видимого цветового излучения для синтезирования кода (шифра) в системах биометрии и колориметрического шифра. Предложено использование цветовой генерации искусственного происхождения для формирования кода и светового идентификатора пользователя информации. Дано обоснование аккомодации квантовой модели излучения света в условиях большого количества цветовых оттенков и сложного состава монохроматических излучений разных частот, что гарантирует высокую стойкость колориметрического шифрования. Для генерации колориметрического кода разработана методика преобразования колориметрических триад геометрическими операциями, разработан алгоритм функционирования расчетной системы различных цветностей. Результаты данного исследования рекомендуется включать в функционал программных сервисов расчета и оценки качества полученного колориметрического кода.

Ключевые слова: биометрия, колориметрическое кодирование, световое генерирование, квантовая модель излучения света, цветность светового спектра излучения, фотометрия.

Колориметрическое кодирование (шифрование) еще недостаточно изученная область в системе информационной безопасности, исследование принципов и различных способов (методов) светового генерирования и их пространственного распределения в оптическом диапазоне представляет определенную научную ценность. Его применение позволяет создавать безопасные биометрические системы с высокой стойкостью их защиты на основе применения оптических принципов. Эти подходы дают возможность хранить и анализировать цветовые характеристики биологических объектов, таких как лица, глаза и пальцы, гарантируя высокую точность и надежность их распознавания, тем самым, развивая системы идентификации и аутентификации возможных потребителей информации. Особое значение в исследуемом процессе имеет волновая теория света, которая через измерение светового потока и яркости света открывает возможности изучения структуры биологических объектов, таких как клетки и ткани, наделяя их уникальными цветовыми характеристиками.

Цель исследования — повышение значимости использования спектральных характеристик цвета через создание индивидуальных неповторимых цветовых пространств с последующим шифрованием для точной идентификации легального пользователя в информационных системах.

Методология

Излучение световых потоков является перспективным научным направлением, в рамках которого использован ряд методов измерения и описания цвета биологических объектов: колориметрия для измерения яркости, насыщенности и тона цвета, фотометрия для измерения светового потока и анализа отражения и поглощения света, спектрофотометрия для измерения поглощения, отражения и преломления света различными материалами. Трехмерная колориметрия для определения цветового пространства позволила описать и из-

Трошков Александр Михайлович, доцент, доцент кафедры "Информационные системы".
E-mail: trochkov1954@mail.ru
Трошков Михаил Александрович, доцент, научный сотрудник кафедры "Информационные системы".
E-mail: m_troshkov@mail.ru
Ермакова Анна Николаевна, доцент, доцент кафедры "Информационные системы".
E-mail: dannar@list.ru
Богданова Светлана Викторовна, доцент кафедры "Информационные системы".
E-mail: svetvika@mail.ru
Шуваев Александр Васильевич, профессор, профессор кафедры "Информационные системы".
E-mail: a-v-s-s@rambler.ru

Статья поступила в редакцию 21 марта 2023 г.

мерить цвет точнее, чем другие методы. Вывод однозначен — каждый метод имеет свои преимущества и ограничения в зависимости от конкретной задачи и условий исследования. Авторами изучены открытые источники информации о современном состоянии развития колориметрического шифрования, проведен всесторонний анализ перспектив биометрических систем контроля доступа к информационным ресурсам.

Результаты

Опираясь на утверждение о том, что различные тела могут испускать световую энергию естественного и искусственного происхождения [1], авторы для синтезирования кода или шифрования информации, а также формирования светового идентификатора предлагают использовать цветовую генерацию искусственного происхождения, например, лампы, лазеры, генераторы цвета. Применяя волновую теорию света Х. Гюйгенса, по которой излучение света представляет собой волновое движение, и электромагнитную теорию света Д. Максвелла, по которой свет — это электромагнитная волна, квантовую теорию излучения света, по которой свет состоит из отдельных энергетических пакетов, рассмотрено распространение электромагнитных волн в виде гармонического колебательного движения, которое описывается волновым уравнением:

$$y(x,t) = A\left[2\pi\sin\left(\frac{t}{T} - \frac{x}{\lambda}\right) + \varphi_0\right] \qquad (1)$$

где $y(x,t)$ — определяющее отклонение от положения баланса (равновесия);
A — амплитуда $\to \max y(x,t)$;
λ — длина волны S;
T — период полного колебания;
φ_0 — начальная фаза, при $t = 0$.

Приведенная формула (1) отражает все основные характеристики волнового процесса. Для удобства колориметрического кодирования (шифрования), исходя из формулы (1), вводится такое понятие, как спектр оптических излучений (таблица).

Спектр оптических излучений

№	Длина волны, нм	Вид	Октава
1	1—780	Инфракрасное	10 октав
2	780—380	Видимое	1 октава
3	380—10	Ультра-фиолетовое	5 октав

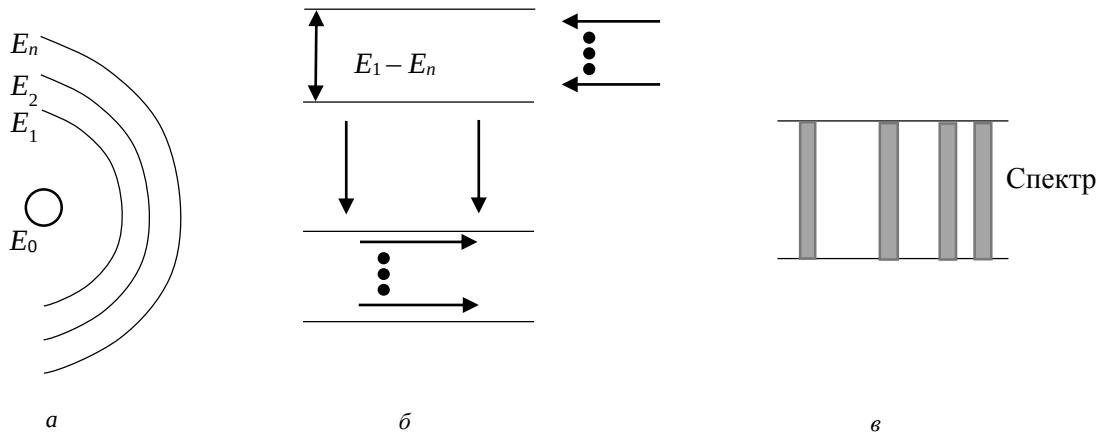
На ее основании делается вывод, что видимое цветовое излучение пригодно для синтезирования кода (шифра) в системах биометрии и колориметрического шифра.

Анализ исследовательских работ показывает возможность колориметрического шифрования с использованием следующих моделей: волновая модель (отражение, дифракция, интерференция), квантовомеханическая модель (поглощение, генерация), комбинированная модель. Для удобства представления колориметрии рекомендуется применить квантовую или комбинированную модели, используя технический термин — колориметрические системы. В основу систем заложено понятие спектрообразования (рис. 1).

На рис. 1 *а* и *б* — квантовые переходы в min частице, а *в* — спектр (его вид). Причем энергию кванта выражают в электрон-вольтах (эВ) в системе разности потенциалов $\mho$, т. е.

$$h\nu = \frac{hc}{\lambda} = e\nu \qquad (2)$$

где ν — частота в Гц;
λ — длина волны.



а *б* *в*

Рис. 1. Принцип спектрообразования

Поскольку для стойкости колориметрического шифрования предлагаем применить достаточно большое количество цветовых оттенков, то излучение будет сложное, состоящее из монохроматических излучений разных частот спектральных составляющих. Спектральный набор предлагаем проектировать на базовой основе дисперсионных процессов принципиальной схемы (рис. 2).

Поскольку для колориметрического кодирования предлагается применить различные оттенки цветов, необходимо учитывать световой поток Ф и отражающий материал тела, учитывая, что световой поток разделяется на три составляющие, то Ф рассчитывается по формуле:

$$\Phi = \Phi_p + \Phi_a + \Phi_t \qquad (3)$$

где Φ_p — световой поток, отраженный от тела;
Φ_a — поток, поглощённый телом;
Φ_t — поток, проходящий через тело.

Тогда при проектировании колориметрического кода (3), необходимо учитывать Ф и его составляющие (Φ_p, Φ_a, Φ_t).

Учитывая предложение о применении множества оттенков цветовых решений, прежде чем приступить к синтезу цветовых комбинаций, определимся с фотометрией. Если проверка кода будет соответствовать образцу, то можно предложить визуальный (зрительный) метод светового измерения. Если же будет применен автоматизированный метод идентификации, то тогда применим физический метод светового измерения с использованием устройств приема излучения, например, фотоэлементов, фоторaспознователей, фотографических индикаторов и т. д.

Для формирования колориметрического кода предлагаем способ на основе местных полей (рис. 3), когда используются локальные характеристики цвета изображения, такие как цветовое распределение и текстура, для формирования кода. Для этого изображение разделяют на блоки, над каждым из которых производят вычисления с использованием методов анализа цветового распределения и текстурных характеристик. Эти значения затем используют для формирования кода, который можно внедрить в изображение.

Для таких полей лучше всего использовать призму с высоким коэффициентом отражения т. е. равно ярким излучателем. Синтез полей сравнения (рис. 3) предлагаем применять для идентификации колориметрического кода при предъявлении пользователем.

В целях формирования различных оттенков цветовой гаммы предлагаем использовать способ преобразования колориметрических триад геометрическими операциями [2]. Систему триад необходимо рассматривать в колориметрической системе трехмерного геометрического пространства *x, y, z*, алгоритм функционирования расчетной системы представлен на рис. 4.

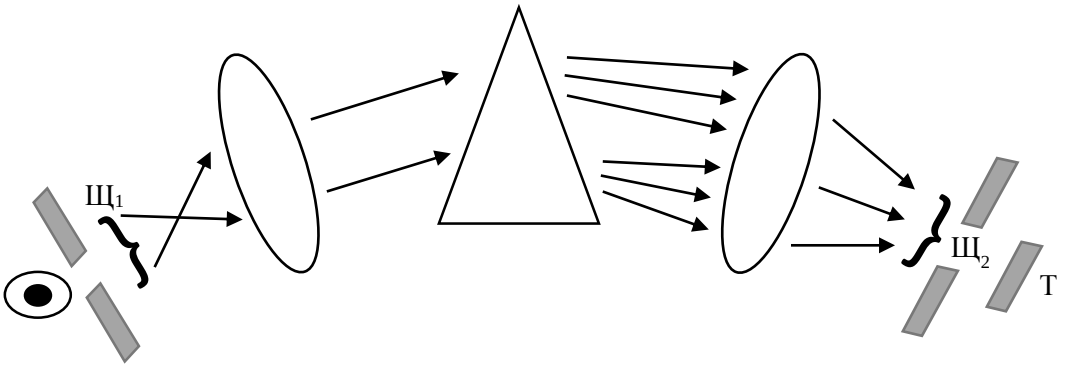


Рис. 2. Схема спектрообразования

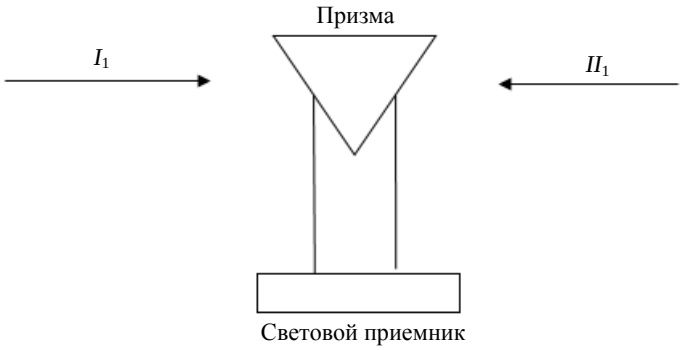


Рис. 3. Синтез полей сравнения

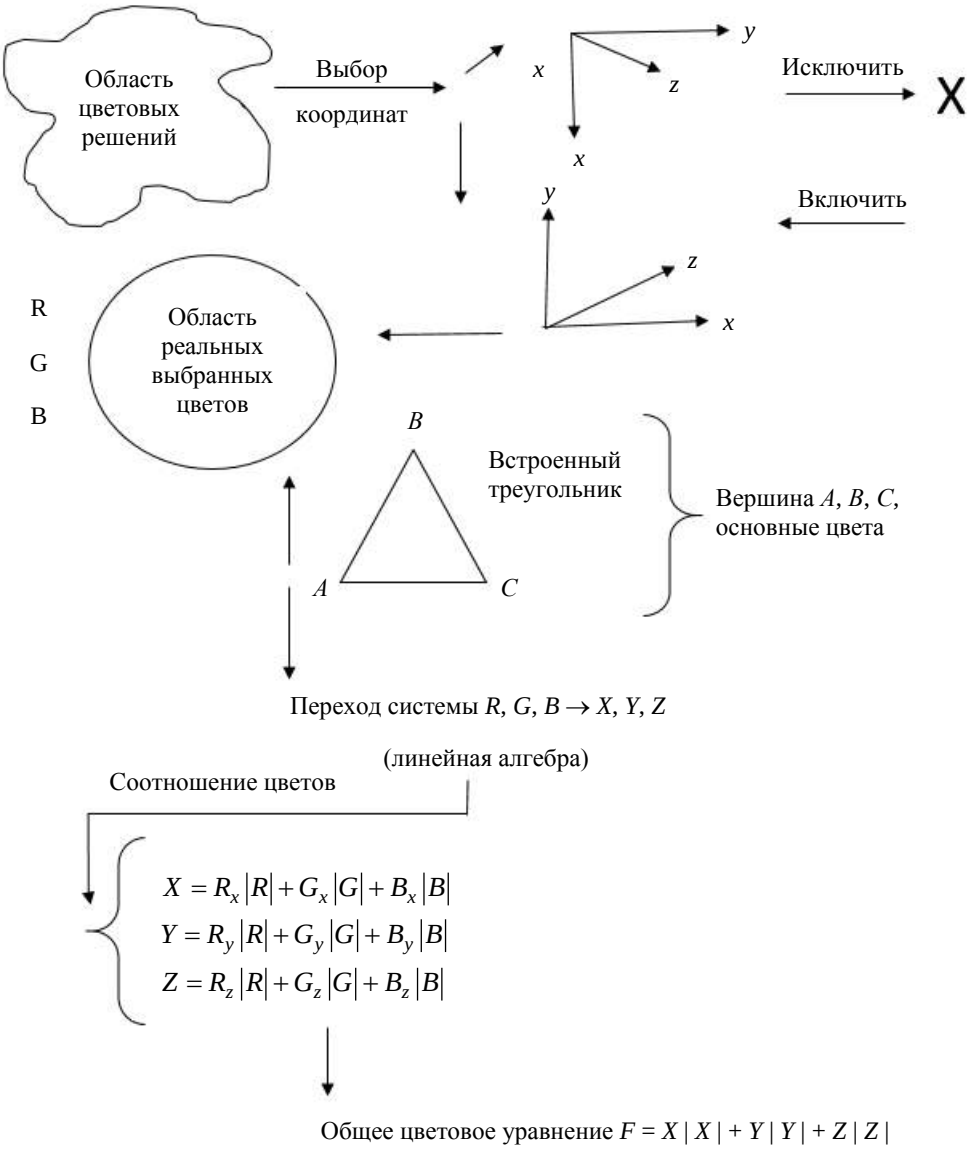


Рис. 4. Алгоритм функционирования расчетной системы цветности

На рис. 4 (x, y, z) — координаты цвета, тогда координаты оттенков цветности следующие:

$$x = \frac{x}{\delta};\ y = \frac{y}{\delta};\ z = \frac{z}{\delta}. \qquad (4)$$

где $\delta = x + y + z$ — цветовой модуль.

Таким образом, из представленных геометрических операций логически следует вывод о возможности достижения различных цветностей для дальнейшего измерение в колориметрическом кодировании (шифровании).

Заключение

На основании проведенных исследований сделан вывод о том, что основными характеристиками для оценки колориметрического кодирования являются: значение энергии кванта (эВ) $h\nu$ (2), цветовой поток Ф (3), общее цветовое уравнение (4). Для функционирования расчетной системы в геометрическом пространстве (x, y, z) предлагаем использовать алгоритм соотношения цветов.

Рассмотренные подходы имеют место в процессе синтеза колориметрического кода и могут быть использованы при разработке программного сервиса расчета и оценки качества полученного колориметрического кода. Развитие предложенной идеи авторы видят в использовании метода на основе моделей цветового пространства и методов, использующих машинное обучение.

Литература

1. *Гуторов М. М.* Основы светотехники и источники света. Изд. 2. — М: Энергия, 1983. — 384 с.

2. *WysZecki G., Fielder G.* Color-difference matches // Jourh. Opt. Soc. Am. 1971. V. 61. P. 1503—1513.

Development of a colorimetric encryption algorithm for biometric systems

A. M. Troshkov, M. A. Troshkov, A. N. Ermakova, S. V. Bogdanova, A. V. Shuvaev
Stavropol State Agrarian University, Stavropol, Russia

The applicability of the visible color spectrum for code synthesis (cipher) in biometric and colorimetric cipher systems has been proved. The use of artificial color generation to form a code and light identifier of the user of information is proposed. A justification is given for the accommodation of the quantum model of light emission under conditions of a large number of color shades and a complex composition of monochromatic emissions of different frequencies, which guarantees a high stability of colorimetric encryption. To generate a colorimetric code, a method of transforming colorimetric triads by geometric operations was developed, and an algorithm for the functioning of the calculation system of different chromaticities was developed. The results of this study are recommended to be included in the functionality of the software services for calculating and evaluating the quality of the colorimetric code obtained.

Keywords: biometrics, colorimetric coding, light generation, quantum model of light emission, chromaticity of the light emission spectrum, photometry.

Bibliography — 2 references. *Received March 21, 2023*

ДОВЕРЕННАЯ СРЕДА

УДК 004
DOI: 10.52190/2073-2600_2023_2_32
EDN: EZKHMY

Порядок и результаты экспериментальных исследований влияния обфускации на качество выявления угроз информационной безопасности, реализуемых посредством эксплоитов, в файлах неисполняемых форматов

А. Н. Архипов
Министерство обороны Российской Федерации, Москва, Россия

В. А. Пиков, В. В. Кабаков
Московский авиационный институт (национальный исследовательский университет), Москва, Россия

Представлены порядок и результаты экспериментальных исследований влияния обфускации на качество выявления угроз информационной безопасности, реализуемых посредством эксплоитов в файлах неисполняемых форматов, существующими методами. Результаты анализа показали существенное снижение качества выявления эксплоитов после проведения их обфускации, что подтверждает актуальность разработки новых, более эффективных методов выявления обфусцированных эксплоитов в файлах неисполняемых форматов.

Ключевые слова: компьютерные атаки, система защиты информации, эксплоит, выявление угроз информационной безопасности, обфускация.

Внедрение информационных технологий достигло уровня, при котором реализация угроз нарушения информационной безопасности может привести к существенным негативным последствиям в области обеспечения обороны страны, безопасности государства, правопорядка, а также в социальной, экономической, политической и экологической сферах [1].

Одной из основных угроз нарушения информационной безопасности является применение вредоносного кода [2], которое может быть выражено в сборе информации о системах или пользователях, уничтожении системных данных, создании закладки для дальнейшего несанкционированного проникновения в систему, фальсификации системных данных и отчетов, внесении путаницы в системные процессы и создании сложностей обслуживающему персоналу [3].

Архипов Александр Николаевич, сотрудник.
E-mail: diskpart111@mail.ru
Пиков Виталий Александрович, старший преподаватель.
E-mail: pikov@ya.ru
Кабаков Виталий Валериевич, старший преподаватель.
E-mail: ser-kvv73@mail.ru

Статья поступила в редакцию 21 марта 2023 г.

Следствием широкого распространения вредоносного кода и средств его разработки является постоянное усложнение методов проведения компьютерных атак на объекты информационной инфраструктуры (70—85 % от общего числа выявляемых случаев компьютерных атак) [4, 5]. Одним из ключевых факторов неконтролируемого распространения вредоносного кода является широкая доступность и анонимность глобального информационного пространства.

При этом вредоносные коды, используемые при компьютерных атаках, могут быть представлены в виде исполняемых и неисполняемых форматов файлов.

Учитывая, что для обеспечения защиты от последствий применения вредоносного кода в качестве контрмер в защищаемой информационной инфраструктуре существует прямой запрет на прием и передачу файлов исполняемых форматов, который реализуется контентной фильтрацией посредством использования соответствующих средств защиты информации, в настоящее время становится все более актуальным распространение вредоносного кода через неисполняемые форматы файлов.

Как следствие, компании, осуществляющие свою деятельность в области обеспечения инфор-

мационной безопасности, в своих докладах и отчетах [6, 7] отмечают тенденцию роста количества компьютерных атак на объекты информационной инфраструктуры с использованием вредоносного кода, распространяемого через неисполняемые форматы файлов в форме автоматических эксплоитов [3].

Вредоносной код, распространяемый в форме автоматических эксплоитов в файлах неисполняемых форматов, будем называть эксплоитом.

Эксплоит маскируется под легитимные неисполняемые файлы, что значительно усложняет решение задачи его своевременного выявления и в сочетании с методами социальной инженерии превращает его в мощный инструмент проведения компьютерных атак.

Повсеместное использование эксплоитов в ходе компьютерных атак объясняется их техническими преимуществами над другими формами вредоносного кода, которые, в свою очередь, позволяют применять множество техник для обхода системы защиты информации и закрепления в целевой информационной системе жертвы [8].

Наиболее известными примерами вредоносных программ, распространяющихся через эксплоиты, являются: "StuxNet", "WannaCry", "NoPetya", "RemSec", "DUQU" и др.

Выявление угроз нарушения информационной безопасности, реализуемых посредством эксплоитов, — задача технически сложная и именно поэтому многие разработчики решений в области информационной безопасности сходятся в едином мнении, что эксплоит — одна из самых серьезных угроз нарушения информационной безопасности [7].

Своевременное выявление эксплоитов является одним из важнейших направлений исследований в области информационной безопасности. В силу очевидных причин, в решении данной задачи анализ файловых объектов на наличие эксплоитов имеет определяющее значение. Наряду с традиционными сигнатурными методами выявления эксплоитов активно используют эвристические (поведенческие) подходы.

Данная группа подходов опирается на формально неопределенные правила (эвристики), чье практическое использование показало свою применимость для выявления эксплоитов, имеющих явное сходство с ранее известными образцами.

Сигнатурные и эвристические (поведенческие) методы имеют хорошие показатели эффективности при выявлении известных образцов эксплоитов, для которых не применялись технологии обфускации (запутывания) программного кода.

При этом, обе группы подходов выявления эксплоитов по сути базируются на поиске устойчивых паттернов в теле эксплоита (их совокупности и взаимосвязи), ранее обнаруженных в исследованных образцах, поэтому злоумышленники создают и постоянно совершенствуют методы специального преобразования эксплоитов, обеспечивающие необходимую изменчивость элементов эксплоита для обхода указанных методов анализа.

Так, в последние годы существенно увеличилось количество компьютерных атак, успешность реализации которых достигалась за счет применения эксплоитов, созданных с использованием методов (технологий) обфускации (запутывания) программного кода [6, 7].

Применение указанных методов не позволяет эффективно бороться с такими угрозами нарушения информационной безопасности, как эксплоиты.

В явном виде понятие обфускации программ было введено в 1997 г. в работе Коллберга, Томборсона и Лоу [8].

Обфускацией программы называется всякое ее преобразование, которое сохраняет вычисляемую программой функцию (эквивалентное преобразование), но при этом придает программе такую форму, что извлечение из текста программы (программного кода) ключевой информации об алгоритмах и структурах данных, реализованных в этой программе, становится трудоемкой задачей. Обфускация программ в противоположность реорганизации (рефакторингу) преследует цель затруднить понимание программ и воспрепятствовать целенаправленной их модификации [9].

Таким образом, справедлива гипотеза о существовании влияния обфускации на качество выявления угроз информационной безопасности, реализуемых посредством эксплоитов в файлах неисполняемых форматов.

Для подтверждения или опровержения данной гипотезы, проведем соответствующие экспериментальные исследования.

Описание проведения экспериментальных исследований

Экспериментальные исследования проводили в следующем порядке:

- определение минимального объема экспериментов;
- подготовка двух тестовых выборок с файлами неисполняемых форматов и внедренными эксплоитами. В первую выборку включали набор файлов с эксплоитами в первозданном виде (выборка № 1), а вторая выборка содержала те же самые файлы с эксплоитами, но измененные с применением технологий обфускации (запутывания) программного кода (выборка № 2);

- обе тестовые выборки анализировали с помощью методов, реализованных в существующих коммерческих средствах антивирусной защиты информации, а также средствах реализованных в рамках научных трудов по данной тематике;
- проведен анализ результатов экспериментального исследования.

Коллекция файлов неисполняемых форматов, содержащих эксплоиты, была сформирована из образцов, размещенных в открытом доступе на ресурсе "MalwareBazaar Database" [10].

Определение минимального объема экспериментов

Первоначально целесообразно решить задачу определения минимального количества экспериментов (исследуемых файлов) для получения информации о влиянии обфускации на качество выявления угроз информационной безопасности, реализуемых посредством эксплоитов в файлах неисполняемых форматов.

Для этого будем исходить из того, что исследуемый процесс, как и большинство процессов, существующих в природе, изменяется по закону нормального распределения изучаемой случайной величины.

Согласно классической теории статистики, в данном случае должна решаться задача определения доверительных интервалов, покрывающих математическое ожидание нормального распределения α, с надежностью γ и точностью δ.

Пусть параметры распределения таковы:

$$M\left(\bar{X}\right)=\alpha,\ \ \sigma\left(\bar{X}\right)=\frac{\sigma}{\sqrt{n}}.$$

Зададим требование, предусматривающее выполнение выражения:

$$P\left(\left|\bar{X}-\alpha\right|<\delta\right)=\gamma,$$

где γ — заданная надежность, получим:

$$P\left(\left|X-\alpha\right|<\delta\right)=2\Phi\left(\frac{\delta}{\sigma}\right).$$

Заменив X на $\bar{X}$ и σ на $\sigma\left(\bar{X}\right)=\frac{\delta}{\sigma}$, получим:

$$P\left(\left|\bar{X}-\alpha\right|<\delta\right)=2\Phi\left(\frac{\delta\sqrt{n}}{\sigma}\right)=2\Phi\left(t\right),\qquad(1)$$

где

$$t=\frac{\delta\sqrt{n}}{\sigma}.\qquad(2)$$

Найдя из последнего тождества $\delta=\frac{t\sigma}{\sqrt{n}}$, справедливо будет выражение:

$$P\left(\left|\bar{X}-\alpha\right|<\frac{t\sigma}{\sqrt{n}}\right)=2\Phi\left(t\right).$$

Приняв во внимание, что вероятность P задана и равна γ, окончательно имеем (чтобы получить рабочую формулу, выборочную среднюю обозначим за $\bar{x}$):

$$P\left(\bar{x}-\frac{t\sigma}{\sqrt{n}}<\alpha<\bar{x}+\frac{t\sigma}{\sqrt{n}}\right)=2\Phi\left(t\right)=\gamma.\qquad(3)$$

Смысл полученного отношения заключается в том, что с надежностью γ можно утверждать, что доверительный интервал $\left(\bar{x}-\frac{t\sigma}{\sqrt{n}},\ \bar{x}+\frac{t\sigma}{\sqrt{n}}\right)$ покрывает неизвестный параметр α с точностью оценки $\delta=\frac{t\sigma}{\sqrt{n}}$. Число t определяем из равенства $2\Phi\left(t\right)=\gamma$ или $\Phi\left(t\right)=\frac{\gamma}{2}$. По таблице функции Лапласа находим аргумент t, которому соответствует значение функции Лапласа, равное $\frac{\gamma}{2}$ [11].

Если известно математическое ожидание с заданной точностью δ и надежностью γ, то минимальный объем выборки, который обеспечит эту точность, находят по формуле (как следствие из $\delta=\frac{t\sigma}{\sqrt{n}}$):

$$n=\frac{t^2\sigma^2}{\delta^2}.\qquad(4)$$

Учитывая, что характеристиками стандартного нормального распределения являются $\alpha = 0$ и $\sigma = 1$, то формула (1) примет вид (5):

$$P\left(\left|\bar{X}\right|<\delta\right)=2\Phi\left(\delta\sqrt{n}\right)=2\Phi\left(t\right),\qquad(5)$$

где $t=\delta\sqrt{n}$.

Из (5) следует, что расчет минимального объема выборки производится по формуле:

$$n=\frac{t^2}{\delta^2}.\qquad(6)$$

Как показывает полученная формула (6), минимальное число опытов прямо пропорционально квадрату значения t, которое находится по табличным значениям функции Лапласа:

$$\Phi\left(t\right)=\frac{\gamma}{2}.\qquad(7)$$

То есть с увеличением надежности минимальное число элементов увеличивается в параболической зависимости. С другой стороны, минимальное число опытов обратно пропорционально точности, число элементов увеличивается, а с уменьшением δ, т. е. с увеличением точности, число элементов наоборот увеличивается.

Примем показатель надежности γ равным 0,99, а показатель точности 0,05. Используя полученную формулу (6), произведем соответствующие вычисления:

- по таблице функции Лапласа определим значение показателя *t*, используя формулу (7). С учетом принятого значения надежности, равного 0,99, значение *t* составляет 0,66;
- с учетом принятого значения точности 0,05 произведем расчет минимального количества экспериментов по формуле (6):

$$n = \frac{0,66^2}{0,05^2} = 174,24.$$

Минимальное количество экспериментов при заданных значениях надежности и точности составляет 174 вредоносных файла.

Таким образом, выборка № 1 была сформирована из 174 образцов вредоносных файлов, содержащих эсплоиты.

Выборка № 2 получена путем обфускации образцов вредоносных файлов из выборки № 1.

В качестве обфускаторов были выбраны свободно доступные на хостинге IT-проектов Github [12] реализации, подходящие для автоматизированного запутывания большого количества исходных кодов.

Для оценки влияния технологий обфускации на качество выявления угроз информационной безопасности, реализуемых посредством эксплоитов, экспериментальному исследованию подверглись существующие методы выявления угроз информационной безопасности, реализованные в современных отечественных коммерческих программных средствах антивирусной защиты.

В качестве указанных средств применяли Kaspersky Endpoint Security для Windows [13], Dr. Web Curelt [14].

Результаты экспериментальных исследований

Для оценки качества выявления угроз информационной безопасности, реализуемых посредством эксплоитов в файлах неисполняемых форматов, использовали такой критерий, как точность корректной классификации исследуемых образцов на чистые и вредоносные.

Введем понятия корректной и некорректной классификации:

- если файл, содержащий эксплоит, классифицирован как чистый, это — некорректная классификация;
- если файл, содержащий эксплоит, классифицирован как вредоносный, это — корректная классификация.

Понятие корректной классификации не нуждается в подобном разбиении на более мелкие подклассы. События корректной и некорректной классификации составляют полное множество элементарных событий. Таким образом, можно считать, что проводимые эксперименты удовлетворяют условию экспериментов Бернулли.

Точность исследуемых методов в данном случае определяется как вероятность корректной классификации и вычисляется следующим образом:

$$P = \frac{S}{N}$$

где S — количество событий корректной классификации;

N — количество попыток классификации.

Таким образом, были получены следующие результаты экспериментального исследования методов выявления угроз нарушения информационной безопасности, реализуемых посредством эксплоитов (таблица).

Результаты экспериментального исследования методов

Выборка	Выявлено, %			
	Сигнатурный анализ		Эвристический анализ	
	KES	Dr. Web Curelt	KES	Dr. Web Curelt
№ 1	99,2	98,8	98,3	97,9
№ 2	73,4	65,7	71,2	64,3

Результат экспериментального исследования с усредненными значениями точности выявления представлен на рис. 1.

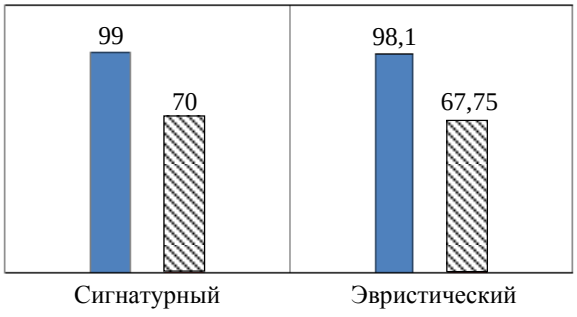


Рис. 1. Результат экспериментального исследования методов:
— точность без обфускации; — точность с обфускацией

Интерпретация результатов

В системе защиты информации информационной инфраструктуры, имеющей подсистему противодействия компьютерным атакам, связь понятия защищенности информации с эффективностью методов выявления угрозы нарушения информационной безопасности, реализуемой посредством эксплоитов, используемых в указанной подсистеме, осуществляется через понятие риска.

Для угроз нарушения информационной безопасности риск оценивается следующим образом [15]:

$$R = \left(P_p - \sum P_n\right) U$$

где R — риск;
P_p — вероятность реализации угроз;
U — ущерб от реализации угрозы;
P_n — вероятность противодействия с использованием конкретного метода (средства) средства защиты информации.

Нарушения информационной безопасности, реализуемые посредством эксплоитов, являются для информационной инфраструктуры одним из средств защиты информации. Вероятность корректного выявления эксплоитов определяет точность исследуемых методов. Поскольку точность методов напрямую влияет на вероятность выявления угрозы нарушения информационной безопасности, точность является основополагающей характеристикой при выборе метода выявления.

Таким образом, повышение точности ведет к снижению вероятности реализации угрозы нарушения информационной безопасности, реализуемой посредством эксплоитов, и, в конечном итоге, к снижению риска этой угрозы. Снижение риска, в свою очередь, повышает защищенность информации, обрабатываемой в информационной инфраструктуре [16].

С учетом изложенного, именно точность методов выявления угроз нарушения информационной безопасности, реализуемых посредством эксплоитов, напрямую влияя на защищенность информации, является основополагающей характеристикой указанных методов.

Результаты вычисления риска нарушения информационной безопасности информационной инфраструктуры при проведении компьютерных атак с использованием обфускации и без использованияпредставлены на рис. 2.

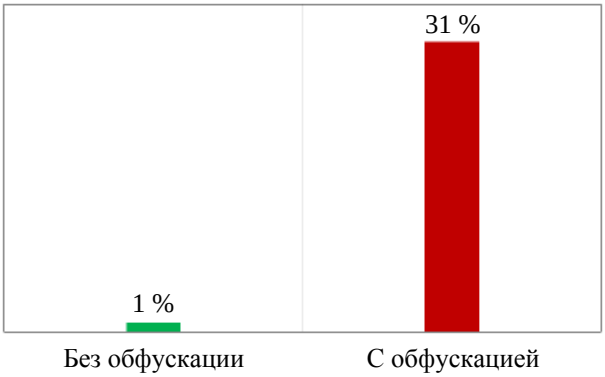


Рис. 2. Показатели риска при реализации угроз нарушения информационной безопасности

Заключение

Произведена оценка влияния обфускации на качество выявления угроз информационной безопасности, реализуемых посредством эксплоитов в файлах неисполняемых форматов. Проведены экспериментальные исследования с файлами, содержащими эксплоиты в их первозданном виде и после их обфускации.

Результаты проведенного исследований на практике подтвердили существенное влияние обфускации на качество выявления угроз нарушения информационной безопасности, реализуемых посредством эксплоитов.

Из результатов, полученных в ходе проведения экспериментальных исследований, следует, что обфускация значительно снижает вероятность корректной классификации угроз нарушения информационной безопасности, реализуемых посредством эксплоитов. Данный вывод подтверждает выдвинутую гипотезу о существовании влияния обфускации на качество выявления угроз информационной безопасности, реализуемых посредством эксплоитов в файлах неисполняемых форматов. В результате выявленного влияния справедливы последствия в виде увеличения риска нарушения информационной безопасности информационной инфраструктуры при проведении компьютерных атак в среднем на 31 %.

Литература

1. Российская газета "Войны виртуальные и реальные" Федеральный выпуск № 180(7938) [Электронный ресурс]. https://rg.ru/: [сайт] [2019] URL: https://rg.ru/2019/08/14/chislo-opasnyh-kiberatak-na-obekty-v-rf-vyroslo-v-11-raz-za-tri-goda.html/ (дата обращения: 22.10.2022).

2. Федеральное агентство по техническому регулированию и метрологии. Национальный стандарт Российской Федерации ГОСТ Р 59506-2021 "Безопасность машин. Вопросы защиты информации в системах управления, связанных с обеспечением функциональной безопасности". — М.: Стандартинформ, 2021. С. 12.

3. Федеральное агентство по техническому регулированию и метрологии. Национальный стандарт Российской Федерации ГОСТ Р от 10 ноября 2014 г. № 56205-2014 "Сети коммуникационные промышленные. Защищенность (кибербезопасность) сети и системы. Часть 1-1. Терминология, концептуальные положения и модели". С. 33.

4. Актуальные киберугрозы: итоги 2020 года АО "Позитив Текнолоджиз" [Электронный ресурс]. https://www.ptsecurity.com: [сайт] [2020]. Режим доступа, URL: https://www.ptsecurity.com/ru-ru/research/analytics/cybersecurity-threatscape-2020/#id3/ (дата обращения: 22.10.2022).

5. Актуальные киберугрозы: II квартал 2021 года АО "Позитив Текнолоджиз" [Электронный ресурс]. https://www.ptsecurity.com: [сайт] [2021]. Режим доступа: URL: https://www.ptsecurity.com/ru-ru/research/analytics/cybersecurity-threatscape-2021-q1/ (дата обращения: 22.10.2022).

6. "THE DUQU 2.0 Technical Details" Version: 2.1 (11 June 2015)" Отчет АО "Лаборатория Касперского" [Электронный ресурс]. https://media.kasperskycontenthub.com/: [сайт] [2015]. Режим доступа: URL: https://media.kasperskycontenthub.com/wp-content/uploads/sites/43/2018/03/07205202/The_Mystery_of_Duqu_2_0_a_sophisticated_cyberespionage_actor_returns.pdf (дата обращения: 22.10.2022).

7. Информационная безопасность в цифрах [Электронный ресурс]. https://www.anti-malware.ru: [сайт] [2021]. Режим доступа: URL: https://www.anti-malware.ru/analytics/Threats_Analysis/2021-cybersecurity-statistics/ (дата обращения: 22.10.2022).

8. *Collberg C., Thomborson C., Low D.* A Taxonomy of Obfuscating Transformations // Technical Report. № 148. Univ. of Auckland, 1997.

9. *Варновский Н. П., Захаров В. А., Кузюрин Н. Н., Шокуров А. В.* Современное состояние исследований в области обфускации программ: определения стойкости обфускации // Труды ИСП РАН. 2014. Т. 26. Вып. 3. С. 167—198.

10. [Электронный ресурс]: // https://bazaar.abuse.ch/: Режим доступа: URL: https://bazaar.abuse.ch/browse/ (дата обращения: 22.10.2022).

11. *Гмурман В. Е.* Теория вероятностей и математическая статистика: учеб. пособие для студентов вузов. Изд. 8, стер. — М.: Высш. шк., 2002. — 479 с.

12. [Электронный ресурс]. https://github.com/: [сайт] [2022]. Режим доступа: URL: https://github.com/ (дата обращения: 22.10.2022).

13. [Электронный ресурс]. https://www.kaspersky.ru/: [сайт] [2022]. Режим доступа: URL: https://www.kaspersky.ru/small-to-medium-business-security/downloads/endpoint (дата обращения: 22.10.2022).

14. [Электронный ресурс]. https://free.drweb.ru/: [сайт] [2022]. Режим доступа: URL: https://free.drweb.ru/cureit/ (дата обращения: 22.10.2022).

15. *Сивачев А. В. и др.* Эффективность стеганодетектирования на основе методов машинного обучения // Вопросы кибербезопасности. 2017. № 2(20). С. 512.

16. *Степанов Е. А., Корнеев И. К.* Информационная безопасность и защита информации. — М.: ИНФРА-М, 2001. С. 54.

The order and results of experimental studies of the influence of obfuscation on the quality of detecting information security threats implemented through exploits in files of non-executable formats

A. N. Arkhipov
Minister of Defense Russian Federation, Moscow, Russia

V. A. Pikov, V. V. Kabakov
Moscow Aviation Institute (National Research University), Moscow, Russia

In this article, an experimental study of the effect of obfuscation on the quality of identifying information security threats implemented through exploits in files of non-executable formats by existing methods is carried out. The results of the analysis showed a significant decrease in the quality of detecting exploits after their obfuscation. The presented results confirm the relevance of developing new more effective methods for detecting obfuscated exploits in files of non-executable formats.

Keywords: computer attacks, information security system, exploit, identification of threats to information security, obfuscation.

Bibliography — 16 references. *Received March 21, 2023*

УДК 004.4’22
DOI: 10.52190/2073-2600_2023_2_38
EDN: FBPMFC

Подход к автоматизации процессов фаззинг-тестирования в цикле непрерывной разработки ПО

К. В. Фурман; Е. П. Сураев; В. В. Егорова; А. С. Панов
ООО «РусБИТех-Астра», Москва, Россия

К. В. Пителинский, канд. техн. наук
Московский политехнический университет, Москва, Россия

Разработан собственный подход к автоматизации фаззинг-тестирования. Проведен анализ существующих подходов к автоматизации; изложены необходимые требования к автоматизации. Предложен подход, удовлетворяющий описанным требованиям, апробация которого подтвердила его эффективность и удобство применения как со стороны разработчиков, так и со стороны инженеров безопасности. проведено сравнение проекта OSS-Fuzz и предложенного подхода.

Ключевые слова: динамическое тестирование, DAST, автоматизация, CI/CD, фаззинг, DevSecOps, OSS-Fuzz.

По данным АНБ США от ноября 2022 г. [1], порядка 70 % уязвимостей программного кода в продуктах от Google и Microsoft связаны с использованием небезопасных в работе с памятью языков программирования, таких как C и C++, предоставляющих большую свободу и гибкость для программиста, но требующих пристального внимания при работе с памятью. Ошибки, допущенные программистом случайно, по незнанию или злонамеренно, могут привести к уязвимостям различного характера, например: переполнение буфера (Stack Overflow, Buffer Overflow, Out-of-bounds, Write/Read и др.), висячий указатель (Dangling pointer, Use-After-Free), утечка памяти (Memory Leak), обращение к нулевому указателю (NULL pointer dereference), целочисленное переполнение (Integer Overflow) и др. Эти и другие ошибки во многих случаях могут быть проэкспуатированы, т. е. привести к реализации атаки со стороны злоумышленника, которая может быть направлена на критическую информационную инфраструктуру.

Фурман Кирилл Владимирович, научный сотрудник.
E-mail: kfurman@astralinux.ru
Сураев Егор Петрович, научный сотрудник.
E-mail: esuraev@astralinux.ru
Егорова Виктория Вячеславовна, руководитель направления динамического анализа.
E-mail: vegorova@astralinux.ru
Панов Алексей Сергеевич, старший научный сотрудник.
E-mail: apanov@astralinux.ru
Пителинский Кирилл Владимирович, MBA, доцент, доцент кафедры "Информационная безопасность".
E-mail: yekadath@gmail.com

Статья поступила в редакцию 10 мая 2023 г.

Разработку и поддержку (тестирование, анализ безопасности и эффективности, исправление ошибок и уязвимостей) программного обеспечения (ПО) принято описывать, как жизненный цикл разработки, наиболее популярным подходом к формированию которого является DevSecOps, подразумевающий обеспечение безопасности на всех этапах разработки программных продуктов.

По результатам исследования GitLab Global DevSecOps Survey за 2022 г. [2], модель DevOps/DevSecOps занимает лидирующее место среди других методик по разработке ПО в компаниях. Среди опрошенных в рамках этого исследования 47 % заявили о том, что их ПО тестируется полностью в автоматическом режиме, а 53 % разработчиков проводят тестирование самостоятельно в процессе написания кода. Опрос инженеров безопасности показал, что только 24 % ошибок и уязвимостей обнаруживают разработчики на этапе разработки (Development — этап "Dev" в рамках цикла DevOps\DevSecOps). Данная статистика говорит о том, что автоматизированное тестирование, проводимое в рамках DevSecOps, зачастую носит характер автоматизированного тестирования в рамках Continuous Integration (CI), в котором тестируется успешность сборки и корректность работы программы (автоматическое тестирование, Unit-тестирование, модульное тестирование, функциональное тестирование и др.). На долю инженеров безопасности на этапе эксплуатации (Operations — этап "Ops" в DevOps\ DevSecOps) остается поиск и устранение более 75 % ошибок и уязвимостей в коде.

Практики использования средств анализа ПО позволяют существенно повысить безопасность

кода путем обнаружения ошибок, ненайденных в процессе запуска автоматического тестирования и сборки, делая небезопасный (с точки зрения работы с памятью) код, более безопасным [1]. Решение проблем, найденных анализаторами, может потребовать больших усилий от разработчиков, но в результате это позволит повысить стабильность и безопасность разрабатываемого продукта.

В соответствии с ГОСТ Р 56939-2016 [3] для обеспечения соответствия необходимому уровню доверия, начиная с минимального шестого уровня, разработчик должен проводить динамический анализ программного кода в целях выявления уязвимостей ПО, в том числе проводить фаззинг-тестирование — вид работ по исследованию программы, направленный на оценку ее свойств и основанный на передаче программе случайных или специально сформированных входных данных, отличных от данных, предусмотренных алгоритмом работы программы. Данный подход проведения исследования безопасности благодаря своей эффективности стал одной из ведущих технологий в обеспечении безопасности приложений, начиная с 2000-х гг. [4]. Это повлекло за собой разработку разного рода инструментальных средств и подходов по проведению фаззинга.

Для обеспечения применимости инструментов фаззинг-тестирования в рамках цикла непрерывной разработки (Continious development, далее — CD) требуется автоматизация всех процессов: от сборки стендов тестирования до сбора результирующего покрытия, что в результате позволит обеспечить высокий уровень доверия к разрабатываемым продуктам на протяжении всего жизненного цикла, устраняя большинство программных ошибок на ранних этапах, тем самым осуществляя практику SLS (Shift Left Security) [5] — обеспечение безопасности на ранних этапах разработки ПО, позволяющую сократить расходы на устранение уязвимостей.

Обзор существующих решений

Сегодня существует не так много решений по автоматизации процессов фаззинг-тестирования и внедрения их в цикл непрерывной разработки ПО. Одним из решений является OSS-Fuzz от Google [6], использующий CIFuzz [7]. Данное решение интегрируется в систему контроля версий (далее — СКВ) и позволяет запускать фаззинг-тестирование по результатам фиксации изменений (коммитов). Так как оно предоставляется в виде проекта с открытым исходным кодом, многие команды разработчиков используют его и дорабатывают под собственные нужды. Одним из известных деривативов является OSS-Sydr-Fuzz [8], доработанный Институтом системного программирования им. В. П. Иванникова Российской академии наук (ИСП РАН), дополняющий фаззинг при помощи AFL++ [9] и libFuzzer [10] собственным инструментом динамического анализа Sydr, обеспечивающим символьное исполнение — технику анализа ПО, позволяющую найти все наборы входных данных, способствующие выполнению каждого из возможных путей.

Схема функционирования OSS-Fuzz отражена на рис. 1.

С помощью CIFuzz после внесения изменений в код разрабатываемого проекта и загрузки их в СКВ производится сборка и запуск стендов фаззинг-тестирования с использованием существующих оберток (специальных программ, созданных для вызова определенного функционала в тестируемом ПО), подготовленных для этого проекта. Если в процессе фаззинга будет обнаружено аварийное завершение программы, CIFuzz сохранит трассировку ошибки, входные данные для ее воспроизведения и завершит тестирование в CI, пометив его красным крестом. Если за отведенное время (10 минут по умолчанию) аварийных завершений программы не будет, CI тестирование закончится с положительным результатом и будет помечено зеленой галочкой.

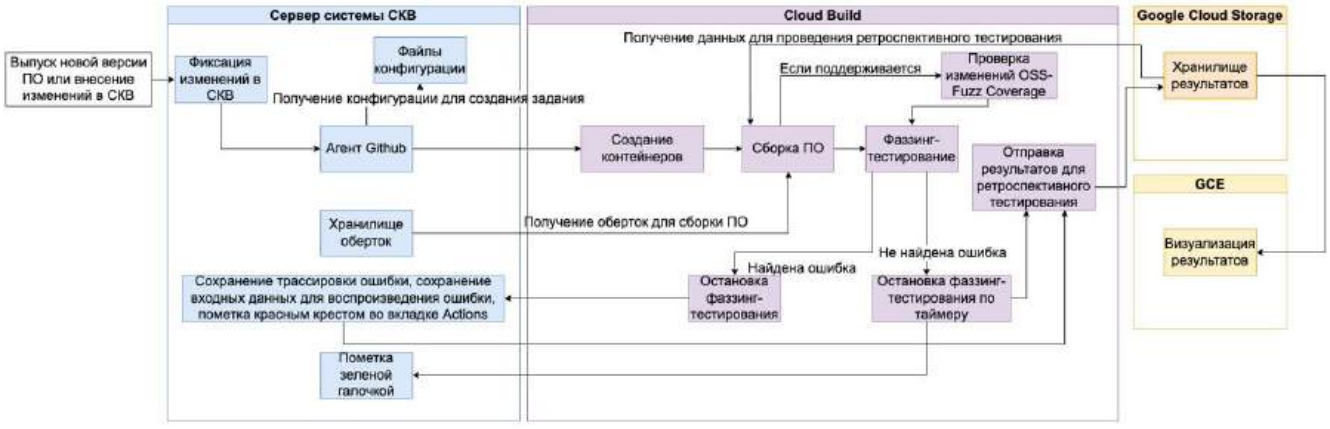


Рис. 1. Схема работы инструмента OSS-Fuzz

При наличии поддержки покрытия кода посредством OSS-Fuzz в тестируемом проекте CIFuzz будет использовать для фаззинга только те обертки, которые непосредственно взаимодействуют с измененными участками кода. В противном случае отведенное на фаззинг время будет разделено на все обертки в проекте. Регрессионное тестирование, как отдельная сущность, в данном инструменте отсутствует. Однако в качестве замены отдельного модуля используется обработка ранее полученных входных данных перед началом фаззинга, которую производит инструмент (AFL++, libFuzzer) по умолчанию. Результаты, полученные во время предыдущих циклов тестирования (ошибки и зависания, корпуса, входные тестовые примеры) сохраняются в течение 30 дней и подаются на вход перед следующими запусками.

За время своего существования (с 2016 г. по февраль 2023 г.), OSS-Fuzz помог обнаружить более 8900 уязвимостей и внести более 28000 исправлений ошибок в код 850 проектов [6].

В OSS-Fuzz не предусмотрено продолжительное фаззинг-тестирование — время фаззинга задается в конфигурационном файле и составляет максимум 6 часов, что является максимально возможным временем выполнения рабочего процесса (job execution time) в GitHub Actions [11]. Опыт разработки показывает, что для продуктов с большой кодовой базой и системного ПО проведение фаззинг-тестирования на протяжении непродолжительного времени бывает недостаточно — фаззер может не успеть достичь необходимого процента покрытия кода и подобрать необходимые входные данные для выявления ошибок и уязвимостей. Также отсутствует тестирование по планировщику задач (запуск тестирования в определенное время) на заданный временной промежуток. Данная функция может быть полезна для инженеров безопасности в случае, когда необходимо провести тестирование всего программного продукта, например, при проведении сертификационных испытаний.

Просмотр результатов доступен на вкладке Actions (действия) в СКВ GitHub. Предусматривается функционал просмотра результатов проведенного тестирования со сводной информацией о запуске, откуда можно выгрузить отчет об ошибке в случае её обнаружения.

Такой подход направлен, прежде всего, на программистов, ведущих разработку проекта. Спустя непродолжительное время после отправки изменений в СКВ, разработчик может убедиться, что внесенные им изменения не повлекли за собой появление новых ошибок. Однако работа инженеров по безопасности будет затруднена. Тестирование проходит автоматически в каждом проекте в СКВ и носит направленный характер, отчего для проведения полноценного всеобъемлющего тестирования ПО инженеру безопасности приходится в ручном режиме посещать страницу проекта, клонировать его к себе на рабочую станцию и проводить динамический анализ каждой утилиты в течение необходимого для полного тестирования времени (критерии завершения тестирования регламентируются ответственными инженерами по безопасности в соответствии с требованиями, установленными в организации, и зависят от уровня критичности тестируемого ПО).

Из-за того, что фаззинг-тестирование проводится по умолчанию на вычислительных мощностях компании Google, проводить тестирование ПО с закрытой кодовой базой не представляется возможным — исходный код потенциально может быть доступен для третьих лиц, что несет экономические риски и риски безопасности.

Существует платформа GitLab DAST [12], доступная в Ultimate версии GitLab. Она позволяет обнаруживать уязвимости в веб-приложениях посредством следующих динамических анализаторов: DAST proxy-based analyzer (для приложений, использующих HTML), DAST browser-based analyzer (для приложений, использующих JavaScript), DAST API analyzer (для сканирования web-API, таких, как REST, GraphQL, SOAP). Инструмент позволяет запускать и офлайн-динамическое тестирование, схожее с тем, что предоставляет OSS-Fuzz. GitLab DAST интегрирован в GitLab CI и запускается в рамках процесса CI/CD.

Несмотря на то, что динамический анализ получил широкое распространение, в открытых источниках нет информации об автоматизированных системах фаззинга, обеспечивающих требования для сертификации ПО. В связи с этим возникает необходимость в разработке централизованной системы по проведению автоматизированного фаззинг-тестирования, удобной как для разработчиков, так и для инженеров безопасности. При этом необходимо, чтобы разрабатываемая система имела возможность не только запускать фаззинг после изменений в коде в СКВ, но и имела отдельный интерфейс для запуска фаззинга на определенное время для определенной утилиты или библиотеки и позволяла визуализировать процесс фаззинга с указанием обнаруженных ошибок и процента достигаемого покрытия в режиме реального времени.

Требования, предъявляемые к подсистеме автоматизации процессов фаззинг-тестирования в CI/CD

Из-за специфики фаззинг-тестирования полностью автоматизировать его процессы на текущий момент не представляется возможным. Разработчикам и инженерам безопасности необходимо создавать обертки (инструмент автоматической генерации оберток для фаззинга Futag [13] направленно вызывает конкретные функции и предназначен для написания оберток к библиотекам, на данный момент еще развивается и его использование так или иначе подразумевает участие инженеров безопасности), а также анализировать результаты и составлять результирующие отчеты. При этом для достижения наибольшей эффективности фаззинга важно отдавать предпочтение наиболее передовым и технологически развитым средствам анализа, учитывать условия функционирования и специфику каждого из тестируемых компонентов. Помимо этого, необходимо корректно определять компоненты, составляющие поверхность атаки объекта оценки (инструмент для определения поверхности атаки Natch [14] требует участие инженеров безопасности для его запуска, настройки и отладки), в целях наиболее быстрого достижения критичных участков кода без снижения качественных показателей. Автоматизация фаззинг-тестирования в рамках цикла CI/CD может облегчить работу программистов и инженеров безопасности.

С учетом изложенного, возникает необходимость в разработке подхода к автоматизации, способного соответствовать всем предложенным требованиям безопасности, но в то же время быть удобным для инженеров безопасности и разработчиков ПО.

Разрабатываемая система автоматизации должна предусматривать модуль регрессионного тестирования, использовать средства контейнеризации для обеспечения изолированности окружения, аккумулировать результаты фаззинг-тестирования в удобной для работы аналитика единой базе данных, иметь графический интерфейс с возможностью просмотра результатов и запуска тестирования. Система должна интегрироваться в СКВ, при этом ее автоматический запуск должен настраиваться по заранее определенным критериям. Сначала должно запускаться непродолжительное тестирование, необходимое разработчику для понимания корректности вносимых им изменений, т. е. повлекли ли данные изменения за собой ошибки в коде программы. После проведения непродолжительного фаззинг-тестирования должен проводиться продолжительный запуск. При этом более длительный запуск после первичного подтверждения не лишает разработчиков и инженеров по безопасности возможности в дальнейшем обнаружить более серьезные ошибки, которые не были найдены во время непродолжительного прогона фаззера.

Предлагаем реализовать новый подход к автоматизации фаззинг-тестирования, который будет удовлетворять всем изложенным спецификациям и обеспечит соответствие требованиям по фаззинг-тестированию, необходимым для сертификации программных продуктов по различным уровням доверия, отраженным в ГОСТ Р 56939-2016 [3], для объекта оценки.

Подход к автоматизации процессов фаззинг-тестирования в CI/CD

Ключевая идея предлагаемого подхода к автоматизации процессов фаззинг-тестирования — унификация и объединение циклов тестирования всех разрабатываемых продуктов в рамках компании в единую систему тестирования и предоставление результатов в удобном виде как для разработчиков, так и для инженеров безопасности.

Подход к автоматизации заключается в объединении нескольких инструментов для увеличения скорости проведения фаззинг-тестирования и повышения его эффективности. Необходимо использовать различные инструменты динамического анализа, учитывающие специфику разрабатываемого продукта и его функциональные особенности. Применение нескольких инструментов фаззинга для анализа одного продукта позволяет находить новые ошибки, специфичные для конкретного инструмента. Также не исключено и применение других методов динамического анализа в сочетании с фаззингом, например, символьное исполнение.

Предлагаемая платформа автоматизации интегрируется в СКВ, что поможет разработчикам после внесения изменений быстро проверить, не вызывают ли эти изменения новых ошибок или не воспроизводят ранее обнаруженные (это обычно происходит при возврате к предыдущей версии ПО, если в новой версии не работает определенный функционал или она не прошла все этапы тестирования). Интеграция системы в СКВ будет незаметной и удобной для разработчиков ПО — им не придется погружаться в процессы динамического анализа (запуск непродолжительного фаззинг-тестирования будет происходить автоматически).

Автоматический запуск непродолжительного тестирования можно настраивать по следующим

критериям: изменения в определенной ветке, значимые изменения, затрагивающие кодовую базу определенной утилиты. Значимость внесенных изменений определяют разработчики или инженеры безопасности по соответствующим событиям — триггерам. Для автоматического запуска выставляется ограничение по времени (от 10 до 60 мин.). Если в течение выделенного времени фаззером не будет обнаружено ошибок, система укажет разработчику, что предварительное тестирование проведено успешно, а далее запустит более продолжительный фаззинг. Так разработчик в короткое время сможет получить краткий отчет об ошибках при внесении изменений (если они были обнаружены в ходе фаззинг-тестирования), содержащий в себе входные данные для воспроизведения ошибки, трассировку ошибки и тип ошибки. При необходимости получить более подробную информацию о проведенном тестировании разработчики могут воспользоваться графическим интерфейсом.

Важной составляющей представленной автоматизированной системы является модуль регрессионного тестирования, обеспечивающий проверку обнаруженных ранее ошибок на новой версии ПО (с доработанным кодом) в целях подтверждения их успешного устранения. При этом данные регрессионного тестирования необходимо сохранять в течение настраиваемого периода времени, регулируемого вручную (зависит от конкретного ПО, специфики его работы и подходов к разработке, принятых в компании). Как показала практика работы в ООО "РусБИТех-Астра" [15], сохранение входных данных, вызывающих ошибки в ПО, должно происходить в течение всего жизненного цикла разработки во избежание повторного возникновения уже решенных ошибок, т. к. из-за большого количества продуктов, выпускаемых под разные аппаратные платформы и эксплуатирующихся в различных сценариях, возникает риск возникновения ошибки, ранее решенной на одной из платформ, на другой платформе. Тестирование утилит производится параллельно, а сами утилиты собраны с различными датчиками срабатывания ошибок для нахождения разных типов ошибок (утечка памяти, переполнение буфера и т. д).

Суть предлагаемого подхода заключается в следующем (рис. 2).

- Разработчик отправляет изменения кода на СКВ.
- По заданным требованиям, производится проверка на необходимость запуска агента СКВ в результате произошедших в коде изменений.
- Если изменения приняты системой, как критичные или соответствуют требованиям, запускается процесс фаззинга.
- Клиент собирает соответствующие обертки из СКВ, загружает из хранилища минимизированные входные корпуса и ошибки, полученные в результате предыдущих итераций фаззинг-тестирования рассматриваемого проекта.
- В течение заданного минимального времени проводится фаззинг-тестирование.
- Если по его результатам не было обнаружено ошибок или аварийных завершений программы, агент завершает и посылает разработчику уведомление об успешно проведенном тестировании, после чего посылает новое задание клиенту, тем самым начиная этап продолжительного фаззинг-тестирования проекта.
- В БД создается задача на фаззинг со статусом "в работе".
- Клиент проверяет открытые задачи в БД, из СКВ собираются обертки, из хранилища загружаются корпуса и входные данные, вызывающие ошибки.
- Клиент создает контейнер, внутри которого происходит сборка программы с различными подходами по инструментации, а также с различными детекторами ошибок (по умолчанию ASAN, LSAN и с другими при необходимости), а также компиляция программы под сбор покрытия.
- Проводится регрессионное тестирование, после чего запускается процесс фаззинг-тестирования.
- В процессе тестирования клиент опрашивает контейнер, автоматически собирает необходимую статистику текущей задачи и отправляет её в БД для обеспечения функционала по просмотру результатов работы фаззера в режиме реального времени.
- После завершения фаззинга происходит сбор покрытия, генерация отчетов, дедупликация и кластеризация результатов.
- Клиент загружает получившиеся артефакты в хранилище и обновляет статус задачи в БД на "завершено".

Для просмотра результатов, отслеживания текущей статистики и ручного запуска фаззинга предусмотрен графический интерфейс, который дает возможность просмотра: состояния процессов фаззинга в режиме реального времени; покрытия по коду каждой тестируемой утилиты; статистики выполненных задач; входных корпусов данных, приводящих к ошибкам в ходе исполнения для конкретной утилиты.

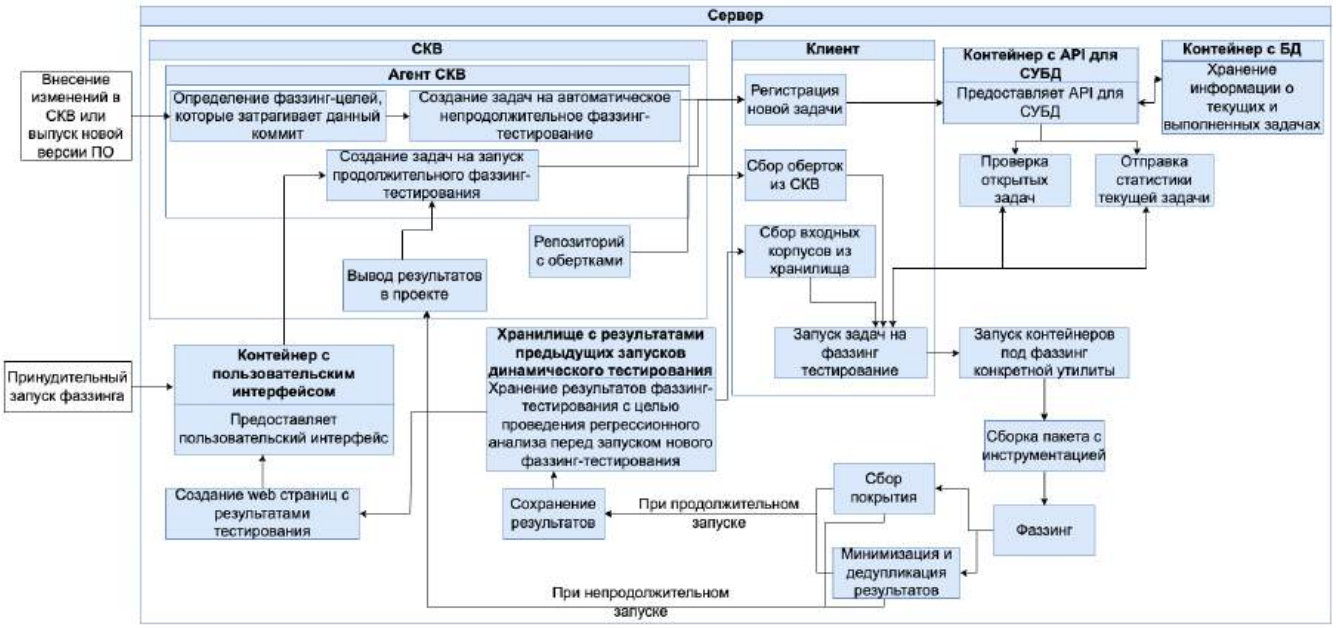


Рис. 2. Предложенный подход

Посредством графического интерфейса пользователь может выбрать для фаззинга конкретную утилиту со следующими параметрами: обертка, время тестирования с условием (достижение определенного времени тестирования, достижение определенного количества запусков, достижение определенного времени с момента обнаружения последнего "интересного" для фаззера тестового примера и др.), выбор датчиков срабатывания ошибок (ASAN, LSAN и др.), выбор ветви с исходным кодом, выбор базового образа для контейнера. После определения этих параметров пользователь запускает процесс фаззинга, который проходит по описанным в алгоритме этапам (за исключением стадии запуска непродолжительного фаззинг-тестирования, инициируемого при автоматическом запуске агентом СКВ). В целях балансировки нагрузки, выбор сервера, на котором будет запущен фаззинг, осуществляется автоматически — клиент получает информацию о степени загрузки серверов, выбирает из них наименее загруженный и посылает на него задание.

Сравнение проекта OSS-Fuzz и предложенного подхода

Для оценки применимости подходов к тестированию в компаниях с закрытой кодовой базой необходимо сравнить проект OSS-Fuzz [6] (как самый известный проект по автоматизации фаззинг-тестирования) с предложенным подходом. Сравнение будет происходить по критериям оценки эффективности фаззинг-тестирования исходя из опыта его проведения применительно к системному и прикладному ПО и СЗИ ОС Astra Linux в ООО "РусБИТех-Астра". Для сравнения были выделены основные требования, предъявляемые к фаззинг-тестированию.

Модульное и регрессионное тестирование должны проводиться для модулей, составляющих поверхность атаки ПО. Фаззинг-тестирование должно выполняться в течение необходимого времени — 8 часов по умолчанию или не менее двух после нахождения последнего пути. В рамках OSS-Fuzz сложно ограничить тестирование только модулями, составляющими поверхность атаки, а само фаззинг-тестирование происходит по результатам изменений, внесенных разработчиком в СКВ (и, как следствие, зависит от количества и регулярности изменений), причем максимальное время запуска фаззинг-тестирования — 6 часов. В рамках предложенного подхода определение поверхности атаки и составление для этого программ-оберток остается за инженерами по безопасности и разработчиками, а тестирование можно проводить по результатам внесенных в СКВ изменений вручную, посредством графического интерфейса, и настраиваться планировщиком (максимальное время запуска фаззинг-тестирования не ограничено и выставляется в файле конфигурации). Фаззинг-тестирование может быть остановлено при наступлении событий (достиже-

ние заранее определенного количества запусков и др.), настраиваемых при запуске.

Сборка ПО при тестировании и регрессионном анализе должна осуществляться со встроенными датчиками срабатывания ошибок (санитайзерами и отладочными аллокаторами), а также со сбором покрытия по строкам исходного кода, функциям и ветвям. В проекте OSS-Fuzz сборка ПО происходит внутри контейнера, а использование датчиков срабатывания ошибок при сборке контролируется разработчиком, сборка покрытия по умолчанию осуществляется посредством OSS-Fuzz, либо также контролируется разработчиком, а вывод отчетов по собранному покрытию производится на публичном web интерфейсе, из-за чего возможно тестирование исключительно проектов с открытым исходным кодом, что накладывает существенные ограничения для разработчиков коммерческого ПО с закрытым исходным кодом. В предлагаемом подходе сборка ПО происходит также внутри контейнера, а программы собираются со встроенными датчиками срабатывания ошибок (при этом каждая программа собирается сразу в нескольких экземплярах для тестирования с различными датчиками и отладочными аллокаторами и с различными способами инструментации), а также производится сборка программы под сбор покрытия, причем результаты покрытия (процент покрытия по строкам и ветвям) доступны в графическом интерфейсе, располагающемся на внутренних серверах компании, что позволяет использовать данную систему для тестирования продуктов как с открытой, так и с закрытой кодовой базой.

Процент покрытия по строкам кода, достигаемый на этапе квалификационного тестирования, должен составлять не менее 80 %, а общий достигаемый в результате фаззинг-тестирования процент покрытия должен быть выше, чем в результате подачи программе стартовых образцов входных данных. Для рассмотренных подходов, как и для любого процесса фаззинг-тестирования, процент полученного в результате покрытия напрямую зависит от написанной разработчиком обертки, количества и качества входных тестовых примеров и словарей и времени фаззинг-тестирования. Ввиду того, что в рамках OSS-Fuzz время ограничено максимум 6 часами, фаззер может не успеть найти максимальное количество путей исполнения программы. Критерии завершения фаззинга — время и обнаружение ошибки (фаззинг завершается, если будет найдена воспроизводимая ошибка). При поддержке проектом сбора покрытия посредством OSS-Fuzz, будет проводиться направленный фаззинг с использованием только тех программ-оберток, которые так или иначе затрагивают внесенные разработчиком изменения, из-за чего полученное в результате покрытие может быть несущественным. Такой подход хорошо подходит для быстрого анализа внесенных изменений, однако для длительного анализа при прохождении сертификации его применимость сомнительна. Предложенный в статье подход не имеет ограничений по времени фаззинга (если они не заданы вручную), а, следовательно, фаззер может находить новые пути исполнения даже спустя продолжительное время, тем самым увеличивая покрытие кода. Несмотря на это, с помощью одной обертки сложно достигнуть высокого уровня покрытия кода, поэтому каждый проект тестируется с использованием нескольких оберток, после чего происходит объединение полученных результатов и консолидация покрытия.

Сформированные в ходе фаззинг-тестирования тесты должны сохраняться и использоваться при последующих запусках фаззера с целью проведения регрессионного анализа. В проекте OSS-Fuzz результаты для проведения ретроспективного тестирования хранятся в течение 30 дней. Предложенный подход не ограничивает пользователей во времени хранения результатов, а потому информация о предыдущих итерациях тестирования может быть сохранена в течение необходимого количества времени. При этом резервное копирование всех артефактов фаззинг-тестирования позволяет сохранить результаты даже в случае непредвиденных системных сбоев и нарушения работы серверов.

Для повышения производительности фаззинг-тестирования и обеспечения изолированности окружения тестирование должно проводиться с использованием средств контейнеризации. В обоих рассматриваемых подходах фаззинг-тестирование проводится в средах контейнеризации, что позволяет обеспечить минимальное воздействие внешних факторов на процесс фаззинг-тестирования объекта оценки.

В качестве оценок соответствия будет применяться следующая шкала:

- 0 — подход не соответствует требованию;
- 0,5 — подход частично соответствует требованию;
- 1 — подход соответствует требованию в полной мере.

Краткие результаты анализа приведены в таблице.

Сравнение подходов

Предъявляемые системе требования	Подход OSS-Fuzz	Предложенный подход
Модульное и регрессионное тестирование должны проводиться для модулей, составляющих поверхность атаки ПО	0,5	1
Сборка ПО при тестировании должна осуществляться со встроенными датчиками срабатывания ошибок	0,5	1
Сборка ПО при тестировании и проведении регрессионного анализа должна осуществляться со сбором покрытия по строкам кода, функциям и ветвям	0,5	1
Фаззинг-тестирование должно выполняться в течение необходимого времени — 8 часов по умолчанию или не менее двух после нахождения последнего пути	0,5	1
Процент покрытия по строкам кода, достигаемый на этапе квалификационного тестирования, должен составлять не менее 80%, а общий достигаемый в результате фаззинг-тестирования процент покрытия должен быть выше, чем в результате подачи программе стартовых образцов входных данных	0,5 (Поведение непредсказуемо и зависит от написанной обертки)	1
Сформированные фаззинг-тестированием тесты должны сохраняться и использоваться при последующих запусках фаззера	0,5	1
Тестирование должно производится с использованием средств контейнеризации	1	1

Результаты сравнения

Из сравнения подходов следует, что предложенный подход по автоматизации соответствует всем поставленным требованиям и подходит для прохождения сертификации ПО любого уровня доверия по требованиям, относящимся к фаззинг-тестированию. При этом были учтены слабые места подхода OSS-Fuzz, которые значительно улучшены при проектировании и разработке собственной платформы.

Заключение

Для повышения эффективности процессов фаззинг-тестирования сформирован подход по его автоматизации, направленный на повышение безопасности и отказоустойчивости программных продуктов и решающий проблему автоматизации процессов безопасности в модели DevSecOps. Подход учитывает интересы и удобен не только для разработчиков ПО, но и для инженеров безопасности, работающих над повышением безопасности конечного продукта. Сводная информация по тестированию программных компонент доступна в общем виде как для программистов, так и для специалистов безопасности, тем самым открывает возможность решать возникающие проблемы (ошибки кода, безопасности) напрямую, без лишнего взаимодействия между подразделениями, повышает скорость устранения ошибок и уязвимостей. Также предусмотрена балансировка нагрузки на сервера, что позволяет применять такую платформу как в больших компаниях с обширной ресурсной инфраструктурой, так и в небольших организациях в условиях ограниченности программных и аппаратных ресурсов, что позволяет повысить отказоустойчивость системы.

Фаззинг-тестирование в рамках предложенной автоматизированной системы может проводиться довольно продолжительное время — время фаззинга настраивается и зависит исключительно от потребностей специалистов. Это позволяет максимизировать эффективность фаззинг-тестирования в условиях объемной кодовой базы, автоматизировать рутинные процессы тестирования и упростить обнаружение программных ошибок на ранних этапах цикла разработки. При этом предложенная система расширяема и масштабируема, что в перспективе позволяет легко добавлять в план тестирования не только новые проекты, но и дополнительные инструменты динамического анализа.

Литература

1. Software Memory Safety, Cybersecurity Information Sheet // U.S. Department of Defense. Режим доступа: URL: https://media.defense.gov/2022/Nov/10/2003112742/-1/-1/0/CSI_SOFTWARE_MEMORY_SAFETY.PDF (дата обращения: 18.03.2023).

2. Thriving in an insecure world, The GitLab 2022 Global DevSecOps Survey // GitLab. Режим доступа: URL: https://cdn.pathfactory.com/assets/10519/contents/432983/c6140cad-446b-4a6c-96b6-8524fac60f7d.pdf (дата обращения: 19.03.2023).

3. ГОСТ Р 56939-2016 "Защита информации. Разработка безопасного программного обеспечения. Общие требования".

4. *Godefroid P.* Fuzzing: Hack, Art, and Science // Rightslink the Acm. V. 64. № 2 Режим доступа: URL: https://patricegodefroid.github.io/public_psfiles/Fuzzing-101-CACM2020.pdf (дата обращения: 20.03.2023).

5. What is Shift Left Security? // Fortinet. Режим доступа: URL: https://www.fortinet.com/ru/resources/cyberglossary/shift-left-security (дата обращения: 23.03.2023).

6. Проект OSS-Fuzz [Электронный ресурс]. Режим доступа: https://github.com/google/oss-fuzz/ (дата обращения: 20.03.2023).

7. Инструментальное средство фаззинг-тестирования CIFuzz [Электронный ресурс]. Режим доступа: https://github.com/CodeIntelligenceTesting/cifuzz (дата обращения 25.03.2023).

8. Проект OSS-Sydr-Fuzz [Электронный ресурс]. Режим доступа:https://github.com/ispras/oss-sydr-fuzz (дата обращения: 21.03.2023).

9. Инструментальное средство фаззинг-тестирования AFLplusplus [Электронный ресурс]. Режим доступа:https://github.com/AFLplusplus/AFLplusplus (дата обращения: 21.03.2023).

10. Инструментальное средство фаззинг-тестирования libFuzzer [Электронный ресурс]. Режим доступа: https://llvm.org/docs/LibFuzzer.html (дата обращения: 25.03.2023).

11. Документация GitHub Actions [Электронный ресурс]. Режим доступа: https://docs.github.com/en/actions/learn-github-actions/usage-limits-billing-and-administration (дата обращения: 22.03.2023).

12. GitLab, Dynamic Application Security Testing (DAST), URL: https://docs.gitlab.com/ee/user/application_security/dast/ (дата обращения: 22.03.2023).

13. Проект Futag [Электронный ресурс]. Режим доступа: https://github.com/ispras/Futag (дата обращения: 23.03.2023).

14. Инструментальное средство определения поверхности атаки Natch [Электронный ресурс]. Режим доступа: https://github.com/ispras/natch (дата обращения 25.03.2023).

15. ООО "РусБИТех-Астра". Режим доступа: URL: https://astralinux.ru/ (дата обращения: 24.03.2023).

Approach to automation of fuzzing-testing processes in the cycle of continuous software development

K. V. Furman, E. P. Suraev, V. V. Egorova, A. S. Panov
"RusBITech-Astra" LLC, Moscow, Russia

K. V. Pitelinskiy
Moscow Polytechnical University, Moscow, Russia

The paper analyzes existing approaches to automation; outlines the necessary requirements to automation; proposes an approach that meets the described requirements and is convenient both for developers and security engineers; compares the OSS-Fuzz approach and the proposed approach.

Keywords: dynamic analyze, DAST, automatization, CI/CD, fuzzing, DevSecOps, OSS-Fuzz.

Bibliography — 15 references. *Received May 10, 2023*

УДК 003.26
DOI: 10.52190/2073-2600_2023_2_47
EDN: FHAHJC

Формирование универсального алгоритма определения угроз информационной безопасности для информационных систем персональных данных

В. В. Кабаков; Н. И. Фокин
Московский авиационный институт (национальный исследовательский университет), Москва, Россия

Одной из наиболее актуальных атак на информационные системы является злоумышленное получение персональных данных. В связи с этим, актуализируется вопрос, связанный с обеспечением защиты и разработкой алгоритма определения угроз информационной безопасности. Предложен универсальный алгоритм, работа которого направлена на определение угроз информационной безопасности информационных систем персональных данных.

Ключевые слова: защита информации, информационная безопасность, угроза, персональные данные, информационная система.

Одним из главных трендов развития научно-технического прогресса является разработка и интеграция различных цифровых и информационных решений. Информационные технологии (ИТ) используют повсеместно при решении различных задач, связанных с хранением, обработкой и передачей информации. Однако наряду с множеством преимуществ наблюдается активное распространение угроз информационной безопасности (ИБ). Исходя из этого, актуализируется вопрос, связанный с обеспечением информационной безопасности современных информационных систем (ИС). В частности, наиболее актуально использование специальных методов защиты именно для информационных систем персональных данных (ИСПДн).

Обеспечение ИБ является комплексной задачей, которая включает в себя не только устранение, но также и нахождение угроз информационной безопасности. Однако на современном этапе развития отсутствует универсальная схема, позволяющая выполнить комплексный анализ для определения актуальных угроз информационной безопасности для той или иной ИСПДн.

Исходя из этого, задача, связанная с разработкой универсального алгоритма определения угроз информационной безопасности в вопросе несанкционированного доступа к персональным данным, становится все более актуальной [1].

Наряду с определением угроз ИБ остро стоит вопрос и осуществления мер по защите информации, выбор которых зависит относительно выявленных угроз. Именно по этой причине были использованы результаты научных исследований, отражающих эффективные инструменты обеспечения информационной безопасности. Используемые работы посвящены интеллектуальным, а также ряду иных автоматизированных средствам оценки и обеспечения информационной безопасности.

Таким образом, определена высокая значимость проблемы, связанной с отсутствием универсального подхода к определению угроз информационной безопасности ИСПДн. Исходя из этого, поставлена задача, связанная с анализом текущего уровня развития методов определения угроз ИБ, что необходимо для последующей проработки вопроса по созданию универсального алгоритма (методики) определения угроз ИБ информационных систем персональных данных.

Материалы и методы

При выполнении работы применяли такие методы научного исследования, как анализ и синтез. Помимо этого были использованы результаты научных исследований зарубежного и отечественного авторства. Именно на основе существующих результатов исследования методов определения

Кабаков Виталий Валериевич, старший преподаватель.
E-mail: ser-kvv73@mail.ru
Фокин Николай Иванович, старший преподаватель.
E-mail: nik.fokin.63@bk.ru

Статья поступила в редакцию 21 марта 2023 г.

угроз информационной безопасности выполнен анализ и синтез информации для последующей систематизации материалов.

Так, в целях получения более подробной информации и актуальных данных в работе использованы научные работы отечественного и зарубежного авторства. В результате работы автором используются научные материалы таких авторов, как: Р. В. Жук, А. А. Миняев, П. В. Мельников, И. П. Волошин и др. [1—3]. В каждой из данных работ затрагиваются фундаментальные вопросы, необходимые с целью воспроизведения общего анализа, касающегося информационной безопасности в сетях сотовой подвижной связи.

Анализ и методы определения угроз информационной безопасности

Оценка рисков информационной безопасности представляет собой процесс выявления и оценки рисков для активов, которые могут быть затронуты информационными атаками. Фактически, определяются как внутренние, так и внешние угрозы. При определении угроз ИБ предполагается оценка их потенциального влияния на такие вещи, как доступность, конфиденциальность и целостность информации. Немаловажным фактором является и оценка затрат, связанных с инцидентом ИБ. С помощью этой информации руководство может адаптировать средства управления информационной безопасностью и защитой данных в соответствии с фактическим уровнем устойчивости организации к рискам.

Для того, чтобы приступить к оценке рисков ИТ-безопасности, необходимо рассмотреть на три важных вопроса: каковы критически важные активы информационных технологий организации, то есть данные, потеря или раскрытие которых окажет серьезное влияние на бизнес-операции; какие ключевые бизнес-процессы используют или требуют эту информацию; какие угрозы могут повлиять на способность этих бизнес-функций работать.

Как только будет определено, что нужно защищать, станет возможным эффективно направить ресурсы в разработку стратегий защиты. Однако прежде чем тратить бюджет или другие ресурсы на реализацию решения по снижению риска, необходимо обязательно проанализировать, с каким риском работает та или иная информационная система, насколько высок его приоритет [2].

Регулярное проведение тщательной оценки ИТ-безопасности помогает организациям создавать прочную основу для обеспечения успеха в бизнесе. В частности, это позволяет: выявить и устранить пробелы в ИТ-безопасности; предотвратить утечку данных; выбрать соответствующие протоколы и элементы управления для снижения рисков; отдать приоритет защите актива с наибольшей ценностью и наибольшим риском; устранить ненужные или устаревшие меры контроля; оценить потенциальных партнеров по безопасности; точно прогнозировать будущие потребности и риски.

Оценка ИТ-рисков должна включать четыре ключевых компонента:

- *Угроза*. Любое событие, которое может нанести вред людям или активам организации. Примеры включают стихийные бедствия, сбои веб-сайтов и корпоративный шпионаж;
- *Уязвимость*. Любое потенциально слабое место, которое может позволить угрозе нанести ущерб. Например, устаревшее антивирусное программное обеспечение представляет собой уязвимость, которая может привести к успешной атаке вредоносных программ. Наличие серверной комнаты в подвале является уязвимостью, которая увеличивает вероятность того, что ураган или наводнение разрушит оборудование и приведет к простою. Другие примеры уязвимостей включают недовольных сотрудников и устаревшее оборудование. Национальная база данных уязвимостей NIST содержит список конкретных уязвимостей [3].
- *Воздействие*. Общий ущерб, который может понести организация, если уязвимость будет использована угрозой. Например, успешная атака программы-вымогателя может привести не только к потере производительности и расходам на восстановление данных, но и к раскрытию данных клиентов или коммерческой тайны, что приведет к потере бизнеса, судебным издержкам и штрафам за несоблюдение нормативных требований.
- *Вероятность*. Риск возникновения угрозы. Обычно это не конкретное число, а диапазон.

Таким образом, анализ угроз ИБ является главным аспектом, необходимым для получения всей необходимой информации относительно информационных угроз. Именно на основе результатов данного анализа определяют потенциальный ущерб материальных и нематериальных последствий, а также вырабатываются наиболее адекватные и эффективные меры противодействия.

На рис. 1 представлены основные методы определения угроз информационной безопасности, используемые при защите информационных систем от несанкционированного доступа к персональным данным.

Рис. 1. Методы определения угроз информационной безопасности

Прямая экспертная оценка основывается на том, что параметры оценки угроз ИБ задаются экспертами. Именно ими определяются основные перечни параметров, которые характеризуют угрозы информационной безопасности и предоставляют субъективные коэффициенты важности каждого из данных параметров.

Статистический анализ представляет собой исследование информационных угроз, основанное на накопленных данных относительно произошедших ранее инцидентов информационной безопасности. Так, например, данный анализ может включать сведения о частоте возникновения угроз определенного типа, источниках и итогах противодействия. При этом имеющиеся данные о частоте проявления угроз ИБ представляют возможность определить вероятность ее последующего возникновения в определенном промежутке времени. Однако для возможности использования данного метода необходим набор инструментов для работы с технологией Big Data. Вместе с этим активно используются интеллектуальные методы, позволяющие производить быстрый анализ большого количества информации и выявление взаимосвязи между инцидентами [4].

Факторный анализ основывается на выявлении и определении факторов, которые с той или иной долей вероятности способны привести к негативным последствиям, связанным с ИБ. Подобными факторами могут быть наличие информации ограниченного доступа, информационные активы, уязвимости самой ИСПДн, уровень вирусной активности и др. Данный анализ особенно актуален при определении угроз информационной безопасности современных ИСПДн ввиду наличия влияния множества факторов.

Именно данные методы являются основными инструментами определения угроз ИБ относительно несанкционированного доступа к персональным данным. Необходимо отметить, что наиболее эффективным решением станет использование данных методов в комплексе. Это способно значительно повысить точность оценки угроз, а также снизить риск упущения определенной угрозы информационной безопасности.

Моделирование угроз информационной безопасности персональных данных

Моделирование угроз безопасности является основным инструментом проведения упреждающей оценки, анализа и определения ключевых приоритетов в работе по устранению угроз ИБ. Главной особенностью проведения данных мероприятий является определение того, где необходимо прилагать наибольшие усилия, необходимые для обеспечения информационной безопасности ИСПДн. При этом принципы моделирования непрерывно меняются в зависимости от добавления, удаления или изменений информационных систем и пользовательских требований [5].

Любая схема обмена информацией может быть представлена как совокупность элементарных информационных потоков. Элементарный информационный поток состоит из трех элементов: передатчик, канал передачи информации, приемник. Необходимо ввести следующие обозначения: V — набор носителей информации (набор вершин графа), E — набор каналов передачи информации (набор ребер графа). Сравнивая любые два элемента из V и один из E, получается элементарный информационный поток в виде неориентированного графа с двумя вершинами (рис. 2).

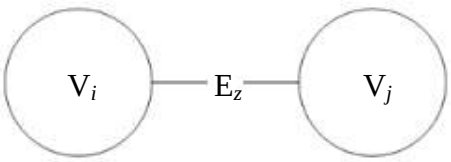


Рис. 2. Элементарный информационный поток

Модель угрозы, в основе которой выступает несанкционированный доступ, влечет за собой возникновение в системе дополнительного элемента, осуществляющего данный доступ (рис. 3). Модель угрозы на основе несанкционированных потоков приводит к угрозе нарушения целостности и доступности, а при взаимодействии с информацией в этом потоке наблюдается угроза нарушения конфиденциальности.

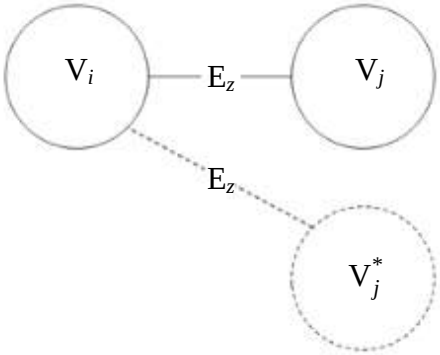


Рис. 3. Возникновение несанкционированного элемента

Также важно отметить другую основную модель — модель угрозы конфиденциальности информации. Принцип построения модели угроз основан на разработанной модели информационных потоков, а именно на концепции элементарного информационного потока. Очевидно, что канал передачи информации — это не какой-то абстрактный объект, а вполне реальный элемент системы, обладающий некоторыми физическими и/или виртуальными свойствами. Это означает, что к нему можно получить доступ таким же образом, как и к двум другим элементам потока. Само определение угрозы конфиденциальности информации подразумевает появление в системе нового элемента, который будет обеспечивать это нарушение (рис. 4).

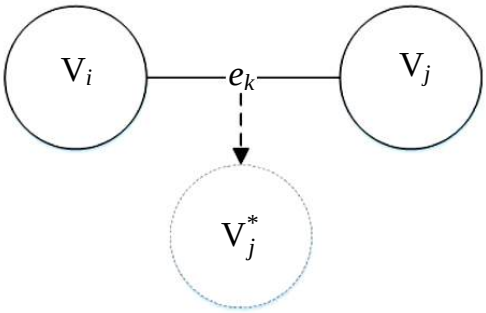


Рис. 4. Возникновение нового элемента, который получает информацию от канала передачи данных

Итак, если говорить исключительно о конфиденциальности информации, то по определению ее нарушение не подразумевает нарушения целостности или доступности, хотя и может привести к этому. Становится очевидным, что конфиденциальность может быть нарушена при замене любого из ее элементов, то есть возможны следующие случаи — замена любой из двух вершин и замена канала [6].

Исходя из представленной информации следует отметить, что модель угроз может выступать как физическое, математическое и описательное представление свойств и характеристик угроз безопасности информации. В свою очередь, модели угроз нарушения безопасности информационных потоков представляют собой совокупность условий и факторов, при которых создается потенциальная или реальная угроза нарушения безопасности использования информации.

Моделирование угроз в данном случае представляет собой итеративный процесс, состоящий из определения активов предприятия, а также определения того, что каждая из информационных систем с ними делает. Помимо этого, необходимо включать создание профиля безопасности для каждой отдельной ИС, определять потенциальные угрозы ИБ и устанавливать приоритеты по их устранению. Моделирование позволяет обеспечить соответствие защиты интенсивно меняющимся условиям. При отсутствии реализации моделирования угроз ИБ можно наблюдаться возникновение новых угроз, оставляя системы и данные наиболее уязвимыми к новым инцидентам информационной безопасности.

Универсальный метод определения угроз информационной безопасности

В качестве универсального метода определения угроз ИБ ИСПДн необходимо использовать математический аппарат. При этом подразумевается разработка имитационной или же математической модели. Наличие и использование данного рода модели предоставит возможность наглядного изучения воздействия угроз на систему, функционирующую с персональными данными. Для построения имитационной модели необходимо интерпретировать систему в качестве линий связи, на основе которых передаются запросы и узлы воздействия угроз. На рис. 5 представлена блок-схема функционирования возможного варианта исполнения такой имитационной модели. Использование данного алгоритма позволит выявить уровень угрозы, а также оценить потенциальную возможность ее устранения без использования дополнительных мер защиты [7].

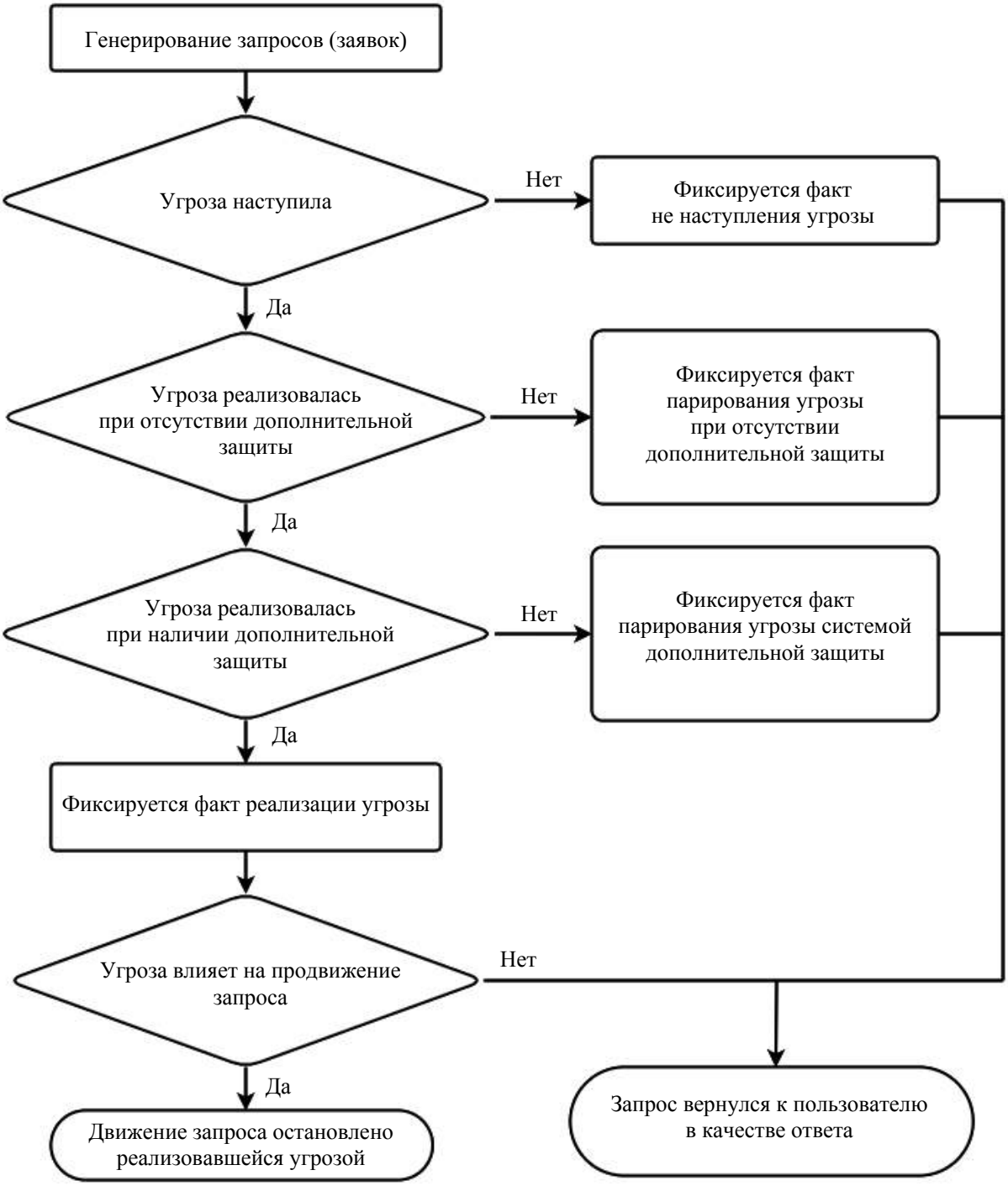


Рис. 5. Алгоритмическая интерпретация универсальной имитационной модели определения угрозы ИБ в ИСПДн

Вместе с этим, использование имитационной модели предоставит возможность расчета необходимости использования дополнительных средств защиты на основе анализа следующих показателей:

- вероятность наступления или появления в информационной системе угрозы ИБ. Данный показатель отражает потенциальное число угроз определенного типа, которые могут взаимодействовать на систему в течение условного промежутка времени;
- вероятность реализации угрозы с учетом исходного уровня защищенности ИСПДн. Данное значение определяет потенциальное количество угроз из числа наступивших, которые не были обнаружены существующей системой защиты;
- вероятность реализации угрозы с учетом дополнительных средств защиты. В данном случае понимается вероятность наступления угрозы при наличии в системе дополнительных уровней защиты [8].

Таким образом, для успешной реализации угрозы необходимо ее наступление с определенной долей вероятности. Помимо этого, необходимо с определенной долей вероятности обойти базовую защиту ИСПДн и с определенной долей вероятности обойти дополнительные модули защиты.

При этом определение первоначальных данных, которые требуются для анализа поведения ИСПДн и выявления актуальных угроз безопасности, необходимо выполнять на основе двухуровневой системы опроса экспертов.

На первом уровне участвует головная группа, которая разрабатывает список потенциальных угроз, возможных к взаимодействию с ИСПДн.

На втором уровне необходимо использовать специальные группы экспертов, которые должны рассматривать угрозы, разделенные относительно типа возникновения и воздействия на систему. Именно данной группой формируется множество вероятностей возникновения и реализации потенциальных угроз, определенных головной группой.

На рис. 6 представлена блок-схема взаимодействия групп экспертов, работающих над определением угроз информационной безопасности.

Рис. 6. Блок-схема взаимодействия экспертов первой и второй группы

Исходя из этого, основной задачей специальной группы экспертов должен стать анализ перечня актуальных угроз безопасности, а также способов их взаимодействия применительно к каждой информационной системе при наличии базовых и дополнительных систем защиты. Помимо этого, специальные группы определяют вероятность появления и реализации угроз при базовых и дополнительных средствах защиты [9].

В результате этого, полученные сведения передаются головной группе экспертов, которым необходимо произвести дальнейшую обработку. Данная группа экспертов выполняет анализ предложенных значений с последующим проведением статистических расчетов. На основе полученной информации утверждается множество значений вероятностей, которые используются для определения наиболее актуальных угроз информационной безопасности относительно несанкционированного доступа к персональным данным.

Заключение

Таким образом, основной целью представленной статьи являлось выполнение анализа относительно вопроса по определению актуальных угроз информационной безопасности информационных систем персональных данных. В результате работы выполнено комплексное исследование относительно текущего состояния вопроса и актуальности обеспечения информационной безопасности ИСПДн. Автором рассмотрены ключевые аспекты формирования универсальной системы определения угроз ИБ, а также предложен уникальный алгоритм для решения данной задачи, основанный на имитационном моделировании. В основе данного алгоритма заложена оценка уязвимости информационной системы той или иной угрозе с параллельной оценкой потенциальной опасности при базовом и дополнительном уровне защиты.

Проработка и создание моделей угроз безопасности требуется в целях выявления слабых мест в существующих системах защиты и разработки соответствующих инструментов по реализации защиты. Активное развитие информационных технологий и использования киберпространства предъявляет повышенные требования к вопросу обеспечения безопасности использования информационных потоков. Исходя из этого, представленные материалы имеют научную значимость, заключающуюся в систематизации сведений относительно текущего уровня развития вопроса и формирования научно-теоретической базы для разработки новых решений.

При это практическая значимость представленного решения подтверждается тем, что именно использование математического аппарат предоставляет возможность получения наиболее точного результата и прогнозирования вероятности реализации угроз на основе данных о имеющейся системе защиты ИС. Наряду с этим, привлечение специалистов, выступающих в роли головной и специальной группы экспертов, также повышает качество и эффективность определения угроз ИБ. Использование универсальной имитационной модели определения угрозы ИБ в ИСПДн является актуальным направлением развития системы защиты информации современных информационных систем, функционирующих с информацией ограниченного доступа. При этом возможность использования представленного алгоритма относит-

ся не только к ИСПДн, но и к иным видам информационных систем.

Литература

1. *Мельников П. В., Ешенко Р. А.* Интеллектуальная система оценки угроз информационной безопасности // Вестник науки. 2020. Т. 1. № 6(27). С. 179—184.
2. *Миняев А. А.* Моделирование угроз безопасности информации в территориально-распределенных информационных системах // Наукоемкие технологии в космических исследованиях Земли. 2021. Т. 13. № 2. С. 52—65.
3. *Жук Р. В., Дзьобан П. И., Власенко А. В.* Определение актуальности угроз информационной безопасности в информационных системах обработки персональных данных с использованием математического аппарата нейронных сетей // Прикаспийский журнал: управление и высокие технологии. 2020. № 1(49). С. 169—178.
4. *Селифанов В. В., Звягинцева П. А., Юракова Я. В., Слонкина И. С.* Применение методов автоматизации при определении актуальных угроз безопасности информации в информационных система с применением банка данных угроз ФСТЭК России // Интерэкспо Гео-Сибирь. 2017. Т. 8. С. 202—209.
5. *Жук Р. В., Дзьобан П. И., Власенко А. В.* Построение взаимосвязи между нарушителем информационной безопасности и уязвимостями информационных активов в информационных системах обработки персональных данных // Прикаспийский журнал: управление и высокие технологии. 2020. № 1(49). С. 162—169.
6. *Ibragimova A. N.* The concept of personal data: information security of the right to privacy according to the analysis of article 8 of the European convention on human rights // North Caucasian Legal Vestnik. 2021. № 4. P. 92—103.
7. *Мельников А. В., Чирков В. Е., Пузарин А. В.* Метод оценки и уменьшения опасности угроз информационной безопасности на основе анализа последовательностей эксплуатации уязвимостей // Вестник ВИ МВД России. 2020. № 1. С. 39—47.
8. *Пиджикян Д. С., Хашева И. А.* Безопасность личных данных во всемирной информационной сети // Student. 2021. Т. 4. № 6. С. 1526.
9. *Селифанов В. В., Юракова Я. В., Карманов И. Н.* Методика автоматизированного выявления взаимосвязей уязвимостей и угроз безопасности информации в информационных системах // Интерэкспо Гео-Сибирь. 2018. № 7. С. 271—276.

Formation of a universal algorithm for determining information security threats for personal data information systems

V. V. Kabakov, N. I. Fokin
Moscow Aviation Institute (National Research University), Moscow, Russia

One of the most relevant attacks on modern information systems is the malicious receipt of personal data. In this regard, the issue related to the provision of protection and the development of an algorithm for determining threats to information security is being update. The purpose of the current article is to form a universal algorithm, the work of which will be direct towards identifying threats to the information security of personal data information systems.

Keywords: information protection, information security, threat, personal data, information system.

Bibliography — 9 references. *Received March 21, 2023*

ЭЛЕКТРОННАЯ ПОДПИСЬ В ИНФОРМАЦИОННЫХ СИСТЕМАХ

УДК 003.26
DOI: 10.52190/2073-2600_2023_2_54
EDN: FMWOTT

Постквантовый алгоритм цифровой подписи с удвоенным проверочным уравнением

А. А. Молдовян, д-р техн. наук
Санкт-Петербургский федеральный исследовательский центр Российской академии наук (СПб ФИЦ РАН), Санкт-Петербург, Россия

Представлен разработанный алгоритм цифровой подписи со скрытой группой и удвоенным проверочным уравнением, основанный на вычислительной трудности решения систем многих квадратных уравнений с многими неизвестными. В качестве носителя алгоритма используются конечные некоммутативные ассоциативные алгебры. Рассмотрены два типа атак на предложенный алгоритм, с учетом которых даны оценки стойкости различных значений размерности алгебраического носителя. Благодаря сравнительно малым размерам открытого ключа и подписи рассмотренный алгоритм представляет интерес как практичная постквантовая криптосхема.

Ключевые слова: информационная безопасность, постквантовая криптография, цифровая подпись, конечная ассоциативная алгебра, некоммутативная алгебра, скрытая группа.

Одним из подходов к разработке постквантовых криптосхем с открытым ключом, включая алгоритмы электронной цифровой подписи (ЭЦП), является использование вычислительной трудности решения системы многих квадратных уравнений с многими неизвестными [1—3]. Данный подход наиболее широко представлен и исследован в рамках многомерной криптографии [4, 5], начиная с 1988 г. [6]. Основным недостатком известных алгоритмов многомерной криптографии является чрезвычайно большой размер открытого ключа. Предложенная недавно новая концепция разработки алгоритмов многомерной криптографии [7], видимо позволит разработать алгоритмы с открытым ключом существенно меньшего размера (в 10 и более раз), однако при уровне стойкости 2^{100} (2^{192}) ожидаемый размер открытого ключа остается сравнительно большим и составляет примерно 6 (36) Кбайт [6].

Алгебраические алгоритмы со скрытой группой, основанные на трудности решения систем квадратных уравнений, представляют возможность существенно уменьшить размер открытого ключа [8, 9], однако обоснование конкретных значений параметров этих алгоритмов, определяющих размер открытого ключа, требует выполнения их криптоанализа, т. е. рассмотрения возможных атак и оценки трудоемкости последних. Пока получены только достаточно предварительные оценки стойкости алгебраических алгоритмов данного типа.

Автор предлагает проанализировать алгебраический алгоритм ЭЦП, основанный на трудности решения систем многих квадратных уравнений с многими неизвестными, для которого предложены два типа атак, на основе которых получены оценки стойкости и размера открытого ключа в зависимости от размерности алгебраического носителя, в качестве которого используются конечные некоммутативные ассоциативные алгебры (КНАА).

Используемые конечные алгебры

Конечные алгебры задаются как векторные пространства с дополнительно определенной операцией умножения векторов, обладающей свойствами замкнутости и дистрибутивности слева и справа. Если операция умножения также обладает свойством ассоциативности, то имеем КНАА. Обычно операция умножения задается с помощью так называемых таблиц умножения базисных векторов (ТУБВ) [10—12].

Учитывая рассмотрение реализаций разработанного алгебраического алгоритма ЭЦП на КНАА различных размерностей m, будем рассматривать в качестве алгебраического носителя алгебры, в которых операция умножения, обладающая свойством ассоциативности, определяется по

Молдовян Александр Андреевич, профессор.
E-mail: maa1305@yandex.ru

Статья поступила в редакцию 20 февраля 2023 г.

ТУБВ, заданным по следующей унифицированной формуле, предложенной в [13]:

$$\mathbf{e}_i\mathbf{e}_j = \begin{cases} \mathbf{e}_{i+j \bmod m}, \text{ if } i \bmod 2 = 0; \\ \mathbf{e}_{i-j \bmod m}, \text{ if } i \bmod 2 = 1,\ j \bmod 2 = 0; \\ \lambda\mathbf{e}_{i-j \bmod m}, \text{ if } i \bmod 2 = 1,\ j \bmod 2 = 1. \end{cases} \quad (1)$$

где $\mathbf{e}_i$, $\mathbf{e}_j$ — базисные векторы, $i, j \in \{0, 1, \ldots, m-1\}$.

Данная формула позволяет вычислить ТУБВ с глобальной двухсторонней единицей, задающие ассоциативное умножение векторов для произвольных четных размерностей. При этом для размерностей $m \geq 6$ задаваемая операция умножения является ассоциативной. Таблицы 1 и 2 иллюстрируют случаи $m = 6$ и $m = 8$. В рамках данной статьи рассматривается случай задания КНАА над простым конечным полем $GF(p)$ с нечетным значением характеристики p, однако формула (1) может быть использована и для задания алгебр над конечными полями характеристики два.

Отметим, что применение в разработанном алгоритме ЭЦП алгебр с множеством глобальных односторонних единиц является нецелесообразным из-за существования разбиения таких алгебр на множество изоморфных подалгебр [14], которое может быть использовано для существенного снижения стойкости.

Таблица 1

Задание шестимерной КНАА с глобальной двухсторонней единицей E = (1, 0, 0, 0, 0, 0) ($\lambda \neq 0$)

•	$\mathbf{e}_0$	$\mathbf{e}_1$	$\mathbf{e}_2$	$\mathbf{e}_3$	$\mathbf{e}_4$	$\mathbf{e}_5$
$\mathbf{e}_0$	$\mathbf{e}_0$	$\mathbf{e}_1$	$\mathbf{e}_2$	$\mathbf{e}_3$	$\mathbf{e}_4$	$\mathbf{e}_5$
$\mathbf{e}_1$	$\mathbf{e}_1$	$\lambda\,\mathbf{e}_0$	$\mathbf{e}_5$	$\lambda\,\mathbf{e}_4$	$\mathbf{e}_3$	$\lambda\,\mathbf{e}_2$
$\mathbf{e}_2$	$\mathbf{e}_2$	$\mathbf{e}_3$	$\mathbf{e}_4$	$\mathbf{e}_5$	$\mathbf{e}_0$	$\mathbf{e}_1$
$\mathbf{e}_3$	$\mathbf{e}_3$	$\lambda\,\mathbf{e}_2$	$\mathbf{e}_1$	$\lambda\,\mathbf{e}_0$	$\mathbf{e}_5$	$\lambda\,\mathbf{e}_4$
$\mathbf{e}_4$	$\mathbf{e}_4$	$\mathbf{e}_5$	$\mathbf{e}_0$	$\mathbf{e}_1$	$\mathbf{e}_2$	$\mathbf{e}_3$
$\mathbf{e}_5$	$\mathbf{e}_5$	$\lambda\,\mathbf{e}_4$	$\mathbf{e}_3$	$\lambda\,\mathbf{e}_2$	$\mathbf{e}_1$	$\lambda\,\mathbf{e}_0$

Для шестимерной КНАА заданной по табл. 1 условием обратимости вектора $\mathbf{A} = (a_0, a_1, a_2, a_3, a_4, a_5)$ является следующее:

$$\frac{1}{4}\left(\left(a_0 + a_2 + a_4\right)^2 - \lambda\left(a_1 + a_3 + a_5\right)^2\right) \times$$

$$\times \begin{pmatrix} \left(a_0 - a_2\right)^2 + \left(a_0 - a_4\right)^2 + \\ +\left(a_2 - a_4\right)^2 - \lambda\left(a_1 - a_3\right)^2 - \\ -\lambda\left(a_1 - a_5\right)^2 - \lambda\left(a_3 - a_5\right)^2 \end{pmatrix}^2 \neq 0. \quad (2)$$

На основе формулы (1) для случая $m = 6$ в работе [15] предложена параметризуемая формула, по которой могут быть заданы КНАА с шестью различными видами глобальной двухсторонней единицей. Можно ожидать, что для таких КНАА условие обратимости векторов будет иметь другой конкретный вид, сохраняя определенное сходство с формулой (2).

Нахождение обратимого шестимерного вектора может быть выполнено путем генерации случайных шести его координат и проверкой выполнимости неравенства (2). При задании КНАА над простым конечным полем $GF(p)$ для случаев размерности $m \geq 8$, для которых не получены формулы задающие условие обратимости, генерация обратимых векторов может быть выполнена выбором случайного вектора **B** и возведением его в степени $p^2 - 1$ и $(p-1)^2$ пока не будет получен единичный вектор **E**. Поскольку процедура быстрого экспоненцирования всегда реализуется в рамках программной реализации алгоритмов со скрытой группой, второй способ нахождения обратимых векторов также является достаточно удобным.

В случае использования четырехмерной КНАА в качестве алгебраического носителя, последняя задается по ТУБВ, представленной в виде табл. 3.

Таблица 2

Задание восьмимерной КНАА с глобальной двухсторонней единицей E = (1, 0, 0, 0, 0, 0, 0, 0) ($\lambda \neq 0$)

•	$\mathbf{e}_0$	$\mathbf{e}_1$	$\mathbf{e}_2$	$\mathbf{e}_3$	$\mathbf{e}_4$	$\mathbf{e}_5$	$\mathbf{e}_6$	$\mathbf{e}_7$
$\mathbf{e}_0$	$\mathbf{e}_0$	$\mathbf{e}_1$	$\mathbf{e}_2$	$\mathbf{e}_3$	$\mathbf{e}_4$	$\mathbf{e}_5$	$\mathbf{e}_6$	$\mathbf{e}_7$
$\mathbf{e}_1$	$\mathbf{e}_1$	$\lambda\,\mathbf{e}_0$	$\mathbf{e}_7$	$\lambda\,\mathbf{e}_6$	$\mathbf{e}_5$	$\lambda\,\mathbf{e}_4$	$\mathbf{e}_3$	$\lambda\,\mathbf{e}_2$
$\mathbf{e}_2$	$\mathbf{e}_2$	$\mathbf{e}_3$	$\mathbf{e}_4$	$\mathbf{e}_5$	$\mathbf{e}_6$	$\mathbf{e}_7$	$\mathbf{e}_0$	$\mathbf{e}_1$
$\mathbf{e}_3$	$\mathbf{e}_3$	$\lambda\,\mathbf{e}_2$	$\mathbf{e}_1$	$\lambda\,\mathbf{e}_0$	$\mathbf{e}_7$	$\lambda\,\mathbf{e}_6$	$\mathbf{e}_5$	$\lambda\,\mathbf{e}_4$
$\mathbf{e}_4$	$\mathbf{e}_4$	$\mathbf{e}_5$	$\mathbf{e}_6$	$\mathbf{e}_7$	$\mathbf{e}_0$	$\mathbf{e}_1$	$\mathbf{e}_2$	$\mathbf{e}_3$
$\mathbf{e}_5$	$\mathbf{e}_5$	$\lambda\,\mathbf{e}_4$	$\mathbf{e}_3$	$\lambda\,\mathbf{e}_2$	$\mathbf{e}_1$	$\lambda\,\mathbf{e}_0$	$\mathbf{e}_7$	$\lambda\,\mathbf{e}_6$
$\mathbf{e}_6$	$\mathbf{e}_6$	$\mathbf{e}_7$	$\mathbf{e}_0$	$\mathbf{e}_1$	$\mathbf{e}_2$	$\mathbf{e}_3$	$\mathbf{e}_4$	$\mathbf{e}_5$
$\mathbf{e}_7$	$\mathbf{e}_7$	$\lambda\,\mathbf{e}_6$	$\mathbf{e}_5$	$\lambda\,\mathbf{e}_4$	$\mathbf{e}_3$	$\lambda\,\mathbf{e}_2$	$\mathbf{e}_1$	$\lambda\,\mathbf{e}_0$

Таблица 3

Задание четырехмерной КНАА ($\lambda \neq 0$) с глобальной двухсторонней единицей вида (1, 1, 0, 0) [16]

•	$\mathbf{e}_0$	$\mathbf{e}_1$	$\mathbf{e}_2$	$\mathbf{e}_3$
$\mathbf{e}_0$	$\mathbf{e}_0$	0	0	$\mathbf{e}_3$
$\mathbf{e}_1$	0	$\mathbf{e}_1$	$\mathbf{e}_2$	0
$\mathbf{e}_2$	$\mathbf{e}_2$	0	0	$\lambda\,\mathbf{e}_1$
$\mathbf{e}_3$	0	$\mathbf{e}_3$	$\lambda\,\mathbf{e}_0$	0

Выбор такой алгебры в качестве алгебраического носителя связан со следующими двумя моментами: наличие восьми ячеек с нулевым значением структурной константы (прореженность ТУБВ) существенно уменьшает вычислительную сложность операций умножения и экспоненцирования; строение этой КНАА детально изучено с точки зрения декомпозиции на множество коммутативных подалгебр [16].

Способ задания скрытой группы и формирование открытого ключа

Впервые метод удвоения проверочного уравнения был предложен для реализации так называемого общего критерия постквантовой стойкости при разработке алгебраических алгоритмов ЭЦП, основанных на вычислительной трудности скрытой задачи дискретного логарифмирования и использующих в качестве своего алгебраического носителя КНАА [17, 18]. Дальнейшее развитие этого метода связано с его применением для задания скрытой группы в алгоритмах ЭЦП на коммутативных ассоциативных алгебрах [19]. Стойкость алгоритмов последнего типа основана на своеобразной форме скрытой задачи дискретного логарифмирования.

В данном исследовании метод удвоения проверочного уравнения впервые используется для задания скрытой группы в алгоритмах ЭЦП, использующих в качестве своего алгебраического носителя КНАА и основанных на вычислительной трудности решения систем многих квадратных уравнений с многими неизвестными. При этом образуется схема построения, для которой является более понятным построение потенциальных атак на разрабатываемые алгоритмы и имеется возможность оценки их сложности, которые могут быть использованы для обоснования выбираемых параметров алгоритма с учетом требуемого уровня стойкости.

Фиксирование скрытой группы реализуется за счет того, что одно и то же значение подписи удовлетворяет двум сходным проверочным уравнениям, параметрами которых являются различные элементы открытого ключа, вычисленные в зависимости от векторов, принадлежащим некоторой секретной коммутативной группе (называемой скрытой группой). Для реализации возможности выбора скрытой группы, обладающей двухмерной цикличностью (в понимании статьи [20]) и включающей только векторы простого порядка q, КНАА, используемые в качестве алгебраического носителя, будем задавать над полем $GF(p)$ с простой характеристикой $p = 2q - 1$, где q — 128-битное простое число.

Скалярные векторы в КНАА с глобальной двухсторонней единицей $\mathbf{E}$ имеют вид $\mathbf{L} = \alpha\mathbf{E}$, где $\alpha = 1, 2, \ldots, p - 1$. Алгебра, заданная по формуле (1), при различных значениях размерности m содержит большое множество различных коммутативных групп порядков $p^2 - 1$ (циклические группы) и $(p-1)^2$ (группы с двухмерной цикличностью, т. е. порождаемые минимальной системой образующих (базисом) из двух векторов порядка $p - 1$). В качестве секретной (скрытой) группы в предлагаемом алгоритме ЭЦП используются коммутативные группы второго типа, задаваемые некоторым случайно генерируемы базисом $<\mathbf{G}, \mathbf{J}>$, в котором каждый из независимых векторов $\mathbf{G}$ и $\mathbf{J}$ имеет простой порядок, равный числу q. Алгоритм генерации базиса $<\mathbf{G}, \mathbf{J}>$ включает следующие шаги:

1. Сгенерировать случайный обратимый вектор $\mathbf{V}$ порядка $p - 1$, не входящий в множество скалярных векторов.

2. Вычислить вектор $\mathbf{G} = \mathbf{V}^2$, имеющий порядок q.

3. Сгенерировать случайное натуральное число k $(0 < k < p - 1)$ и случайный первообразный корень β по модулю p и вычислить число $\psi = \beta^2 \bmod p$.

4. Выполняя скалярное умножение на ψ, вычислить вектор $\mathbf{J} = \psi\,\mathbf{G}^k$, имеющий порядок q.

Возможность вычисления значения подписи, удовлетворяющего двум разным уравнениям, обеспечивается знанием (и использованием) представления элементов открытого ключа в виде замаскированных (за счет левых и правых умножения на маскирующие множители) элементов скрытой группы. Например, это реализуется следующим способом формирования открытого ключа в виде восьми векторов $\mathbf{Y}_1$, $\mathbf{Z}_1$, $\mathbf{U}_1$, $\mathbf{W}_1$, $\mathbf{T}_1$, $\mathbf{Y}_2$, $\mathbf{Z}_2$, $\mathbf{U}_2$, $\mathbf{W}_2$, $\mathbf{T}_2$, описываемым такими вычислительными шагами:

1. Сгенерировать базис первой скрытой группы, обладающей двухмерной цикличностью и включающей два случайных вектора $\mathbf{G}$ и $\mathbf{J}$ порядка q.

2. Сгенерировать случайный обратимый вектор $\mathbf{A}$, удовлетворяющий неравенству $\mathbf{AG} \neq \mathbf{GA}$, и вычислить вектор $\mathbf{Y}_1 = \mathbf{AGA}^{-1}$.

3. Сгенерировать случайное натуральное число $b < q$ и случайный обратимый вектор $\mathbf{D}$, удовлетворяющий неравенствам $\mathbf{DG} \neq \mathbf{GD}$ и $\mathbf{DA} \neq \mathbf{AD}$, и вычислить векторы $\mathbf{T}_1 = \mathbf{D}^{-1}\mathbf{JD}$ и $\mathbf{Z}_1 = \mathbf{AG}^b\mathbf{JD}$.

4. Сгенерировать случайное натуральное число $d < q$ и случайный обратимый вектор $\mathbf{B}$, удовлетворяющий неравенству $\mathbf{BG} \neq \mathbf{GB}$, и вычислить векторы $\mathbf{Y}_2 = \mathbf{BGB}^{-1}$ и $\mathbf{Z}_2 = \mathbf{BG}^d\mathbf{D}$.

5. Сгенерировать базис второй скрытой группы, обладающей двухмерной цикличностью и включающей два случайных вектора $\mathbf{F}$ и $\mathbf{H}$ порядка q.

6. Сгенерировать случайный обратимый вектор $\mathbf{L}$, удовлетворяющий неравенству $\mathbf{LF} \neq \mathbf{FL}$, и вычислить вектор $\mathbf{U}_1 = \mathbf{LFL}^{-1}$.

7. Сгенерировать случайное натуральное число $u < q$ и случайный обратимый вектор $\mathbf{N}$, удовлетворяющий неравенствам $\mathbf{NF} \neq \mathbf{FN}$ и $\mathbf{NL} \neq \mathbf{LN}$, и вычислить вектор $\mathbf{W}_1 = \mathbf{NF}^u\mathbf{HL}^{-1}$.

8. Сгенерировать случайное натуральное число $w < q$ и случайный обратимый вектор $\mathbf{Q}$, удовлетворяющий неравенству $\mathbf{QF} \neq \mathbf{FQ}$, и вычислить векторы $\mathbf{U}_2 = \mathbf{QFQ}^{-1}$, $\mathbf{T}_2 = \mathbf{NHN}^{-1}$ и $\mathbf{W}_2 = \mathbf{NF}^w\mathbf{Q}^{-1}$.

Процедуры генерации и верификации ЭЦП

Процедуру формирования цифровой подписи к некоторому документу M зададим в следующем виде:

1. Используя некоторую согласованную коллизионно стойкую хэш-функцию $f_h(\bullet)$, разрядность которой равна удвоенной разрядности простого числа q, вычислить ее значение h, представленное в виде конкатенации двух чисел h_1 и h_2 одинаковой разрядности, от подписываемого документа

$$h = h_1 \| h_2 = f_h(M).$$

2. Сгенерировать случайный обратимый вектор $\mathbf{R}$ и случайные натуральные числа $k < q$ и $t < q$ и вычислить векторы $\mathbf{R}_1$ и $\mathbf{R}_2$:

$$\mathbf{R}_1 = \mathbf{AG}^k\mathbf{J}^{h_1+1}\mathbf{RHF}^t\mathbf{L}^{-1};$$
$$\mathbf{R}_2 = \mathbf{BG}^k\mathbf{G}^{d-b}\mathbf{RH}^{h_2}\mathbf{F}^{w-u}\mathbf{F}^t\mathbf{Q}^{-1}.$$

3. Вычислить значение хэш-функции $f_h(\bullet)$, представленное в виде конкатенации двух чисел e_1 и e_2 разрядности $|q|$ ($|q|$ — битовая длина числа q), от подписываемого документа с присоединенными к нему векторами $\mathbf{R}_1$ и $\mathbf{R}_2$:

$$e = e_1 \| e_2 = f_h(M, \mathbf{R}_1, \mathbf{R}_2).$$

4. Вычислить значения s_1 и s_2:

$$s_1 = k - e_1 - b \bmod q;$$
$$s_2 = t - e_2 - u \bmod q.$$

5. Вычислить вектор $\mathbf{S} = \mathbf{D}^{-1}\mathbf{G}^{s_1}\mathbf{RF}^{s_2}\mathbf{N}^{-1}$.

Подписью является пара значений $(e, \mathbf{S})$. Секретным ключом, используемым для вычисления ЭЦП является множество, включающее два числа b и u и следующие десять векторов $\mathbf{A}$, $\mathbf{B}$, $\mathbf{G}$, $\mathbf{J}$, $\mathbf{H}$, $\mathbf{F}$, $\mathbf{L}^{-1}$, $\mathbf{Q}^{-1}$, $\mathbf{G}^{d-b}$, $\mathbf{F}^{w-u}$ (последние четыре вычисляются заранее, чтобы понизить вычислительную сложность процедуры генерации ЭЦП).

Процедура проверки подлинности ЭЦП к документу M включает следующие шаги:

1. Вычислить векторы $\mathbf{R}'_1$ и $\mathbf{R}'_2$:

$$\mathbf{R}'_1 = \mathbf{Y}_1^{e_1}\mathbf{Z}_1\mathbf{T}_1^{h_1}\mathbf{SW}_1\mathbf{U}_1^{e_2};$$
$$\mathbf{R}'_2 = \mathbf{Y}_2^{e_1}\mathbf{Z}_2\mathbf{ST}_2^{h_2}\mathbf{W}_2\mathbf{U}_2^{e_2}.$$

2. Вычислить значение хэш-функции e' от документа, к которому присоединены векторы $\mathbf{R}'_1$ и $\mathbf{R}'_2$: $e' = f_h(M, \mathbf{R}'_1, \mathbf{R}'_2)$.

3. Если имеет место равенство $e' = e = e_1 \| e_2$, то ЭЦП признается подлинной, иначе — ложной.

Доказательство корректности представленной схемы ЭЦП выполняется в виде демонстрации того, что правильно вычисленная ЭЦП проходит проверочную процедуру как подлинная подпись:

$$\mathbf{R}'_1 = \mathbf{Y}_1^{e_1}\mathbf{Z}_1\mathbf{T}_1^{h_1}\mathbf{SW}_1\mathbf{U}_1^{e_2} =$$
$$= \mathbf{AG}^{e_1}\mathbf{A}^{-1}\mathbf{AG}^b\mathbf{JDD}^{-1}\mathbf{J}^{h_1}\mathbf{D} \times$$
$$\times \mathbf{D}^{-1}\mathbf{G}^{k-e_1-b}\mathbf{RF}^{t-e_2-u}\mathbf{N}^{-1}\mathbf{NF}^u\mathbf{HL}^{-1}\mathbf{LF}^{e_2}\mathbf{L}^{-1} =$$
$$= \mathbf{AG}^k\mathbf{J}^{h_1+1}\mathbf{RF}^t\mathbf{HL}^{-1};$$

$$\mathbf{R}'_2 = \mathbf{Y}_2^{e_1}\mathbf{Z}_2\mathbf{ST}_2^{h_2}\mathbf{W}_2\mathbf{U}_2^{e_2} =$$
$$= \mathbf{BG}^{e_1}\mathbf{B}^{-1}\mathbf{BG}^d\mathbf{DD}^{-1}\mathbf{G}^{k-e_1-b}\mathbf{RF}^{t-e_2-u}\mathbf{N}^{-1} \times$$
$$\times \mathbf{NH}^{h_2}\mathbf{N}^{-1}\mathbf{NF}^w\mathbf{Q}^{-1}\mathbf{QF}^{e_2}\mathbf{Q}^{-1} =$$
$$= \mathbf{BG}^k\mathbf{G}^{d-b}\mathbf{RH}^{h_2}\mathbf{F}^t\mathbf{F}^{w-u}\mathbf{Q}^{-1}.$$

$$\{\mathbf{R}'_1 = \mathbf{R}_1;\ \mathbf{R}'_2 = \mathbf{R}_2\} \Rightarrow e' = e.$$

Последнее равенство означает, что проверяемая подпись прошла процедуру верификации как подлинная ЭЦП, что означает корректность предложенного алгоритма.

Потенциальные атаки и оценка стойкости

Рассмотрим атаку типа попытки поделки подписи, т. е. вычисления ЭЦП к документу M без использования секретного ключа. Пусть имеется некоторая подлинная подпись $(e^*, \mathbf{S}^*)$, где $e^* = e_1^* \| e_2^*$, к документу M^*, для которой вычисляем значение хэш-функции $h^* = h_1^* \| h_2^* = f_h(M)$ и векторы

$$\mathbf{R}_1 = \mathbf{R}_1'^* = \mathbf{Y}_1^{e_1^*} \mathbf{Z}_1 \mathbf{T}_1^{h_1^*} \mathbf{S}^* \mathbf{W}_1 \mathbf{U}_1^{e_2^*} \text{ и} \qquad (3)$$

$$\mathbf{R}_2 = \mathbf{R}_2'^* = \mathbf{Y}_2^{e_1^*} \mathbf{Z}_2 \mathbf{S}^* \mathbf{T}_2^{h_2^*} \mathbf{W}_2 \mathbf{U}_2^{e_2^*}. \qquad (4)$$

После этого вычисляем хэш-значение $e = e_1 \| e_2 = f_h(M, \mathbf{R}_1, \mathbf{R}_2)$ и попытаемся вычислить вектор $\mathbf{S}$, такой, что пара $(e, \mathbf{S})$ будет удовлетворять равенствам

$$\mathbf{R}_1' = \mathbf{Y}_1^{e_1} \mathbf{Z}_1 \mathbf{T}_1^{h_1} \mathbf{S} \mathbf{W}_1 \mathbf{U}_1^{e_2} = \mathbf{R}_1 \text{ и} \qquad (5)$$

$$\mathbf{R}_2' = \mathbf{Y}_2^{e_1} \mathbf{Z}_2 \mathbf{S} \mathbf{T}_2^{h_2} \mathbf{W}_2 \mathbf{U}_2^{e_2} = \mathbf{R}_2. \qquad (6)$$

Выражая элементы открытого ключа через элементы секретного ключа, из выражений (3) и (5) легко получить единственное значение $\mathbf{S} = \mathbf{S}'$, при котором справедливо равенство (5):

$$\mathbf{S}' = \mathbf{D}^{-1} \mathbf{G}^{e_1^* - e_1} \mathbf{J}^{h_1^* - h_1} \mathbf{D} \mathbf{S}^* \mathbf{N} \mathbf{F}^{e_2^* - e_2} \mathbf{N}^{-1}.$$

Аналогичным способом из (4) и (6) легко получить единственное значение $\mathbf{S} = \mathbf{S}''$, при котором справедливо равенство (6):

$$\mathbf{S}'' = \mathbf{D}^{-1} \mathbf{G}^{e_1^* - e_1} \mathbf{D} \mathbf{S}^* \mathbf{N} \mathbf{F}^{e_2^* - e_2} \mathbf{H}^{h_2^* - h_2} \mathbf{N}^{-1}.$$

Для коллизионно-стойкой хэш-функции вероятность выполнения равенства $\mathbf{S}' = \mathbf{S}''$ пренебрежимо мала, поэтому рассмотренный способ подделки практически нереализуем. Вычислительная сложность данной атаки может быть оценена, как вычислительная сложность нахождения коллизии, используемой хэш-функции, равная $\approx 2^{|q|}$ операций вычисления хэш-функции.

Рассмотрим атаку другого типа, а именно, направленную на вычисление секретного ключа по открытому ключу путем решения следующих двух систем векторных квадратных уравнений (см. процедуру формирования открытого ключа):

$$\begin{cases} \mathbf{A}^{-1}\mathbf{Y}_1 = \mathbf{G}\mathbf{A}^{-1}; \\ \mathbf{D}\mathbf{T}_1 = \mathbf{J}\mathbf{D}; \\ \mathbf{A}^{-1}\mathbf{Z}_1 = \left(\mathbf{G}^b\mathbf{J}\right)\mathbf{D}; \\ \mathbf{B}^{-1}\mathbf{Y}_2 = \mathbf{G}\mathbf{B}^{-1}; \\ \mathbf{B}^{-1}\mathbf{Z}_2 = \mathbf{G}^d\mathbf{D}; \end{cases}$$

$$\begin{cases} \mathbf{U}_1\mathbf{L} = \mathbf{L}\mathbf{F}; \\ \mathbf{T}_2\mathbf{N} = \mathbf{N}\mathbf{H}; \\ \mathbf{W}_1\mathbf{L} = \mathbf{N}\left(\mathbf{F}^u\mathbf{H}\right); \\ \mathbf{U}_2\mathbf{Q} = \mathbf{Q}\mathbf{F}; \\ \mathbf{W}_2\mathbf{Q} = \mathbf{N}\mathbf{F}^w. \end{cases}$$

В первой системе имеем пять уравнений со следующими семью неизвестными векторами: $\mathbf{A}^{-1}$, $\mathbf{D}$, $\mathbf{B}^{-1}$, $\mathbf{G}$, $\mathbf{J}$, $\mathbf{G}^d$ и $\mathbf{G}^b\mathbf{J}$, из которых последние четыре связаны условием выбора из одной и той же коммутативной группы.

Во второй системе также имеем пять уравнений с другими семью неизвестными векторами: $\mathbf{L}$, $\mathbf{F}$, $\mathbf{N}$, $\mathbf{F}$, $\mathbf{H}$, $\mathbf{F}^w$ и $\mathbf{F}^u\mathbf{H}$, из которых последние четыре связаны условием выбора из одной и той же коммутативной группы. Чтобы устранить необходимость рассмотрения экспоненциальных уравнений с неизвестными натуральными значениями b, d, u и w, мы рассматриваем векторы $\mathbf{G}$, $\mathbf{J}$, $\mathbf{G}^d$ и $\mathbf{G}^b\mathbf{J}$ в первой системе и $\mathbf{F}$, $\mathbf{H}$, $\mathbf{F}^w$ и $\mathbf{F}^u\mathbf{H}$ во второй, как независимые неизвестные выбранные из одной и той же коммутативной группы. Принадлежность одной коммутативной группе может быть задана дополнительными уравнениями $\mathbf{GJ} = \mathbf{JG}$, $(\mathbf{G}^d)\mathbf{J} = \mathbf{J}(\mathbf{G}^d)$ и $(\mathbf{G}^b\mathbf{J})\mathbf{J} = \mathbf{J}(\mathbf{G}^b\mathbf{J})$ для первой системы и $\mathbf{FH} = \mathbf{HF}$, $(\mathbf{F}^w)\mathbf{H} = \mathbf{H}(\mathbf{F}^w)$ и $(\mathbf{F}^u)\mathbf{H} = \mathbf{H}(\mathbf{F}^u)$ для второй.

С учетом дополнительных векторных квадратных уравнений имеем две системы из восьми квадратных уравнений с семью неизвестными. Прямолинейное представление векторных квадратных уравнений в скалярные квадратные уравнения по ТУБВ, задающей используемую m-мерную КНАА в качестве алгебраического носителя, приводит к рассмотрению двух независимых систем из $8m$ скалярных квадратных уравнений с числом скалярных неизвестных, равным $7m$.

Однако число скалярных уравнений и скалярных неизвестных может быть понижено, если использовать результаты изучения разбиения используемой КНАА на коммутативные подалгебры. Для случая $m = 4$ была показана общность строения четырехмерных КНАА, заданных над полем $GF(p)$, различных типов, и выведены формулы, описывающие множество элементов коммутативной подалгебры по координатам вектора, содержащегося в данной подалгебре, причем порядок таких подалгебр равен p^2, а формулы, описывающие подалгебру, включают две целочисленные переменные, принимающие значения от 0 до $p - 1$ [16, 21].

При переходе от системы векторных уравнений к системе скалярных уравнений задание значения из скрытой группы, например, вектора $\mathbf{G}$ в первой

системе (и **F** во второй системе), как совокупности четырех скалярных неизвестных (т. е. его координат), позволяет выразить каждый из неизвестных векторов **J**, $\mathbf{G}^d$ и $\mathbf{G}^b\mathbf{J}$ (**H**, $\mathbf{F}^w$ и $\mathbf{F}^u\mathbf{H}$) через два неизвестных скалярных значения.

С учетом описанного, для случая четырехмерных КНАА можно легко показать, что каждая из двух исходных систем из пяти векторных квадратных уравнений преобразуется в систему из $5m$ скалярных степенных уравнений в поле $GF(p)$, включающую $4m + 3(m/2)$ скалярных неизвестных. При этом часть уравнений будет иметь третью степень, однако при использовании лучших известных алгоритмов решения систем многих степенных уравнений с многими неизвестными это несущественно влияет на вычислительную сложность нахождения решения по сравнению со случаем квадратных уравнений (при заданном числе уравнений и неизвестных). Видимо, аналогичная ситуация с уменьшение числа скалярных уравнений имеет место и для КНАА других размерностей, однако для случаев $m \geq 6$ в литературе отсутствуют данные по декомпозиции КНАА на множество коммутативных подалгебр. Будем полагать, что указанный результат по снижению числа скалярных уравнений и неизвестных для четырехмерных КНАА можно распространить и на случай размерностей $m \geq 6$. Данное предположение приемлемо, поскольку оно не повышает, а занижает оцениваемый уровень стойкости разработанного алгоритма ЭЦП.

Рассмотренные две системы векторных уравнений не являются полностью независимыми, поскольку они связаны условием выполнимости двух проверочных уравнений для одного и того же значения подписи. Это условие определяет следующее векторное уравнение (которое может быть получено приравниванием значений элемента подписи **S**, выраженных из первого и второго проверочных уравнений):

$$\mathbf{G}^{d-b}\mathbf{J}^{-h_1-1}\mathbf{A}^{-1}\mathbf{R}_1'\mathbf{L} = \mathbf{B}^{-1}\mathbf{R}_2'\mathbf{Q}\mathbf{F}^{u-w}\mathbf{H}^{-h_2+1}.$$

С учетом последнего уравнения указанные две системы векторных уравнений задают единую систему скалярных уравнений, число которых равно $11m$ при числе скалярных неизвестных $2(4m + 3(m/2)) = 11m$. Принимая во внимание лучшие известные алгоритмы решения систем степенных (включая квадратные) уравнений, использующие так называемые алгоритмы F4 [22] и F5 [23], в работе [24] получены оценки для минимального числа уравнений (см. табл. 4), которое необходимо для достижения заданного уровня стойкости при различных значениях порядка n поля $GF(n)$, в котором заданы уравнения, при числе уравнений, равном числу неизвестных.

Таблица 4

Минимальное число уравнений в поле $GF(n)$, обеспечивающее заданный уровень стойкости W [24]

W	2^{80}	2^{100}	2^{128}	2^{192}	2^{256}
$n = 16$	30	39	51	80	110
$n = 31$	28	36	48	75	103
$n = 256$	26	33	43	68	93

С учетом данных табл. 4 видно, что при реализации на m-мерной КНАА предложенный алгоритм обеспечивает стойкость 2^{128}, 2^{192} и 2^{256} при $m = 4$, 6 и 8 соответственно. Предполагая, что найдутся способы раздельного нахождения значений половины неизвестных, и с учетом того, что для предложенного алгоритма уравнения задаются в поле порядка $n = 2^{128}$, для случая такой потенциальной атаки были получены оценки, приведенные в табл. 5. Как и следовало ожидать, для рассмотренных вариантов атаки второго типа повышение стойкости алгоритма достигается за счет увеличения разрядности КНАА, используемой в качестве алгебраического носителя.

Таблица 5

Предполагаемый уровень стойкости W при потенциальной возможности раздельного вычисления $11m$ скалярных неизвестных

m	4	6	8	10	12
Число неизвестных	22	33	44	55	66
W	$>2^{80}$	$>2^{100}$	$>2^{128}$	$>2^{160}$	$>2^{192}$
Размер подписи, байт	96	128	160	192	224
Производительность алгоритма, отн. ед.	125	28	16	10	7

В отличие от этого случая, для атаки первого типа, основанной на поиске коллизии используемой хэш-функции, ожидаемый уровень стойкости не зависит от размерности КНАА и составляет 2^{128}. Стойкость к атаке первого типа может быть повышена до значения 2^{256}, если задать использование двух независимых коллизионно-стойких хэш-функций для вычисления значений $h = h_1 \| h_2$ и $e = e_1 \| e_2$, используемых в предложенном алгоритме. При этом размер ЭЦП не увеличивается для заданного значения размерности.

Заключение

Благодаря сравнительно малым размерам открытого ключа и подписи предложенный алгоритм представляет интерес, как практичная постквантовая схема ЭЦП. Представляет интерес реализация предложенного алгоритма на КНАА, заданных над полями $GF(2^z)$ характеристики два при различных значениях степени расширения z, что представляет интерес для снижения стоимости

аппаратной реализации и повышения производительности. Эта возможность связана с тем, что разложимость порядка скрытой группы на простые множители малого размера не снижает стойкости алгоритма.

Литература

1. *Ding J., Petzoldt A., Schmidt D. S.* Solving Polynomial Systems. In: Multivariate Public Key Cryptosystems // Advances in Information Security. 2020. V. 80. P. 185—248. DOI: 10.1007/978-1-0716-0987-3_8
2. *Moldovyan N. A.* Algebraic signature algorithms with a hidden group, based on hardness of solving systems of quadratic equations // Quasigroups and Related Systems. 2022. V. 30. № 2. P. 287—298.
3. *Shuaiting Q., Wenbao H., Yifa L., Luyao J.* Construction of Extended Multivariate Public Key Cryptosystems // International J. Network Security. 2016. V. 18. № 1. P. 60—67.
4. *Ding J., Petzoldt A., Schmidt D. S.* Multivariate Cryptography: Multivariate Public Key Cryptosystems. Advances in Information Security. 2020. V. 80. P. 7—23. DOI: 10.1007/978-1-0716-0987-3_2
5. *Hashimoto Y.* Recent Developments in Multivariate Public Key Cryptosystems // In: Takagi, T., Wakayama, M., Tanaka, K., Kunihiro, N., Kimoto, K., Ikematsu, Y. (eds) International Symposium on Mathematics, Quantum Theory, and Cryptography. Mathematics for Industry. Springer, Singapore. 2021. V. 33. P. 209—229. DOI: 10.1007/978-981-15-5191-8_16
6. *Matsumoto T., Imai H.* Public quadratic polynomial-tuples for efficient signature verification and message-encryption // Advances in Cryptology. Eurocrypt'88 Proceedings. Springer Berlin Heidelberg, 1988. P. 419—453.
7. *Молдовян Д. Н.* Альтернативный способ построения алгоритмов многомерной криптографии // Вопросы защиты информации. 2022. № 3. С. 13—21. DOI: 10.52190/2073-2600_2022_3_13
8. *Moldovyan A. A., Moldovyan D. N., Moldovyan N. A.* A Novel Method for Developing Post-quantum Digital Signature Algorithms on Non-commutative Associative Algebras // Информационно-управляющие системы. 2022. № 1. С. 44—53. DOI: 10.31799/1684-8853-2022-1-44-53
9. *Молдовян Д. Н.* Типовые проверочные уравнения в алгебраических алгоритмах ЭЦП со скрытой группой // Вопросы защиты информации. 2022. № 1. С. 31—37. DOI: 10.52190/2073-2600_2022_1_31
10. *Молдовян Д. Н.* Протокол бесключевого шифрования на основе скрытой задачи дискретного логарифмирования // Вопросы защиты информации. 2019. № 3. С. 26—32.
11. *Молдовян Н. А., Костина А. А., Курышева А. А.* Протоколы коллективной и слепой подписи на конечных группах с многомерной цикличностью // Вопросы защиты информации. 2021. № 2. С. 22—29. DOI: 10.52190/2073-2600_2021_2_22
12. *Молдовян Н. А., Костина А. А.* Альтернативный способ построения схем цифровой подписи, удовлетворяющих критерию постквантовой стойкости // Вопросы защиты информации. 2020. № 3. С. 16—21.
13. *Moldovyan N. A.* Unified Method for Defining Finite Associative Algebras of Arbitrary Even Dimensions // Quasigroups and Related Systems. 2018. V. 26. № 2. P. 263—270.
14. *Левина А. Б., Молдовян А. А.* О выборе алгебраического носителя схем цифровой подписи на некоммутативных алгебрах // Вопросы защиты информации. 2021. № 4. С. 39—44. DOI: 10.52190/2073-2600_2021_4_39.
15. *Молдовян Д. Н.* Задание шестимерных алгебр как носителей криптосхем, основанных на скрытой задаче дискретного логарифмирования // Вопросы защиты информации. 2021. № 1. С. 26—32. DOI: 10.52190/2073-2600_2021_1_26.
16. *Moldovyan D. N.* A practical digital signature scheme based on the hidden logarithm problem // Computer Science J. Moldova. 2021. V. 29. № 2(86). P. 206—226.
17. *Moldovyan D. N., Moldovyan A. A., Moldovyan N. A.* Digital signature scheme with doubled verification equation // Computer Science J. Moldova. 2020. V. 28. № 1(82). P. 80—103.
18. *Молдовян А. А., Молдовян Н. А., Молдовян Д. Н., Фархутдинов Р. Ш.* Схемы цифровой подписи с удвоенным проверочным уравнением // Вопросы защиты информации. 2021. № 2. С. 30—36. DOI: 10.52190/2073-2600_2021_2_30
19. *Moldovyan N. A., Moldovyan D. N.* A novel method for developing post-quantum cryptoschemes and a practical signature algorithm // Applied Computing and Informatics. 2021. DOI: 10.1108/ACI-02-2021-0036
20. *Moldovyan N. A., Moldovyanu P. A.* New primitives for digital signature algorithms // Quasigroups and Related Systems. 2009. V. 17. № 2. P. 271—282.
21. *Молдовян А. А., Молдовян Д. Н., Молдовян Н. А., Костина А. А.* Конечные кватернионоподобные алгебры как носители постквантовых алгоритмов ЭЦП // Вопросы защиты информации. 2022. № 2. С. 21—29. DOI: 10.52190/2073-2600_2022_2_21
22. *Faugére J.-C.* A new efficient algorithm for computing Gröbner basis (F4) // J. Pure Appl. Algebra. 1999. V. 139. № 1–3. P. 61—88.
23. *Faugére J.-C.* A new efficient algorithm for computing Gröbner basis without reduction to zero (F5): Proceedings of the International Symposium on Symbolic and Algebraic Computation. 2002. P. 75—83. DOI: 10.1145/780506.780516
24. *Ding J., Petzoldt A.* Current State of Multivariate Cryptography // IEEE Security and Privacy Magazine. 2017. V. 15. № 4. P. 28—36.

Post-quantum digital signature algorithm with doubled verification equation

A. A. Moldovyan

St. Petersburg Federal Research Center of the RAS (SPC RAS), St. Petersburg, Russia

The developed digital signature algorithm with a hidden group and a doubled verification equation, based on the computational difficulty of solving systems of many quadratic equations with many unknowns, is presented. Finite non-commutative associative algebras are used as the carrier of the algorithm. Two types of attacks on the proposed algorithm are considered and taken into account for security evaluation of the algorithm for various values of the dimension of the algebraic carrier. Due to the relatively small sizes of the public key and signature, the considered algorithm is of interest as a practical post-quantum crypto scheme.

Keywords: information security, post-quantum cryptography, digital signature, finite associative algebra, non-commutative algebra, hidden group.

Bibliography — 24 references. *Received February 20, 2023*

ОБЩИЕ ВОПРОСЫ БЕЗОПАСНОСТИ ИНФОРМАЦИИ И ОБЪЕКТОВ

УДК 004.056.5+004.08
DOI: 10.52190/2073-2600_2023_2_61
EDN: DVESQT

Способ реализации требования контроля перемещения носителей данных за пределы контролируемой зоны

С. П. Панасенко, канд. техн. наук
АО «Актив-софт», Москва, Россия

Рассмотрена проблема контроля перемещения носителей данных за пределы контролируемой зоны. Проанализирован классический подход к реализации такого контроля и предложен альтернативный вариант контроля на основе постоянного определения местоположения контролируемых носителей данных и доказательства их наличия в определенных местах в пределах контролируемой зоны. Предложенный подход может быть расширен на контроль местонахождения любых носимых предметов, представляющих ценность либо важных с точки зрения обеспечения информационной безопасности.

Ключевые слова: меры защиты информации, машинные носители информации, контролируемая зона, отслеживание местоположения, RFID.

Требования по контролю перемещения носителей данных за пределы контролируемой зоны

В числе прочих нормативных документов требования по защите информации (ЗИ) в информационных системах (ИС) различного назначения устанавливаются приказами Федеральной службы по техническому и экспортному контролю (ФСТЭК) России [1—4], в которых также перечислен ряд мер по защите информации, применяемых для обеспечения соответствия таким требованиям. Данные приказы перечислены в табл. 1.

Таблица 1

Приказы, устанавливающие требования по защите данных

№ приказа	Дата приказа	К чему предъявляет требования по ЗИ
17	11.02.2013	Государственные информационные системы
21	18.02.2013	Информационные системы персональных данных
31	14.03.2014	Автоматизированные системы управления производственными и технологическими процессами
239	25.12.2017	Объекты критической информационной инфраструктуры

Панасенко Сергей Петрович, директор по научной работе.
E-mail: panasenko@guardant.ru

Статья поступила в редакцию 10 мая 2023 г.

Отметим, что меры ЗИ группируют в базовые наборы мер защиты для обеспечения надлежащего уровня защиты для требующихся классов (уровней) защищенности ИС или категорий значимости объектов критической информационной инфраструктуры.

Подмножество мер по ЗИ, перечисленных в приказах [1—4], посвящено управлению машинными носителями информации (МНИ) и защиты как самих МНИ, так и информационных систем от различных угроз, которые могут быть реализованы с помощью МНИ. Одна из таких мер ЗИ (обозначена в приказах [1—4] как "ЗНИ.3") требует проведения контроля перемещения МНИ за пределы контролируемой зоны (КЗ).

Мера ЗНИ.3 не входит в базовые наборы мер защиты ни для одного класса и типа ИС. Тем не менее, данная мера может быть использована для обеспечения соответствия требованиям по ЗИ следующим образом:

- *в качестве дополнительной меры* — в дополнение к мерам из требуемого базового набора мер защиты;
- *в качестве компенсирующей меры* — используемой вместо какой-либо меры из базового набора, которую по каким-либо причинам невозможно реализовать.

Рассмотрим далее вопросы реализации меры ЗНИ.3.

Классический подход к реализации меры ЗНИ.3

Классическим подходом к реализации данной меры является организация досмотра сотрудников по периметру КЗ в целях подтверждения отсутствия выносимых ими МНИ. Подобный досмотр становится более сложно осуществлять с течением времени по причине миниатюризации переносных МНИ. В частности, USB-флеш-носители данных общего назначения становятся всё меньше с точки зрения физических размеров, что продиктовано удобством использования носителей небольшого размера.

Противодействием подобной миниатюризации могут быть следующие методы:

- запрет на использование в ИС носителей, физический размер которых меньше определенного (здесь и далее подразумеваем, что в ИС, для которых требуется соответствие требованиям приказов [1—4], обеспечивается невозможность бесконтрольного подключения произвольных МНИ к техническим средствам);
- принудительное увеличение размера (за счет, например, помещения в некий короб) или детектируемости (за счет, например, оснащения несъемными радиочастотными метками — RFID) используемых МНИ.

Перечисленные методы не являются панацеей и (теоретически) могут быть обойдены за счет, в частности, следующих действий:

- создания миниатюрного МНИ, эмулирующего разрешенный к применению;
- экранирования RFID-метки для выноса оснащенного меткой МНИ за пределы КЗ.

Рассмотрим альтернативный вариант реализации указанной меры ЗИ.

Предлагаемый подход к реализации меры ЗНИ.3

Предлагаемый альтернативный подход к организации контроля перемещения МНИ за пределы КЗ основан не на организации досмотра/контроля по периметру КЗ, а на доказательстве факта присутствия каждого контролируемого МНИ в определенном месте в пределах КЗ.

Для этого необходимо постоянно определять местонахождение МНИ в пределах КЗ, что можно осуществить следующим образом:

- каждый контролируемый МНИ всегда должен находиться (определенным образом детектироваться — см. далее) в одном из разрешенных для него местоположений в пределах КЗ;
- отсутствие МНИ в разрешенных местоположениях допускается (для переноса из одного из разрешенных местоположений в другое), но только в пределах определенного для каждого МНИ тайм-аута;
- отсутствие МНИ в разрешенных местоположениях свыше тайм-аута трактуется, как событие информационной безопасности (ИБ); данное событие инициирует соответствующие меры реагирования.

Примерами таких разрешенных местоположений могут быть технические средства (ТС) в пределах КЗ, в частности, компьютеры пользователей, автоматизированные рабочие места (АРМ) и т. п. Частным случаем разрешенных местоположений может быть док-станция (или склад), куда подключаются (или сдаются) носители после завершения их использования.

На рис. 1 приведена схема распределенной клиент-серверной системы определения местоположения МНИ в режиме реального времени.

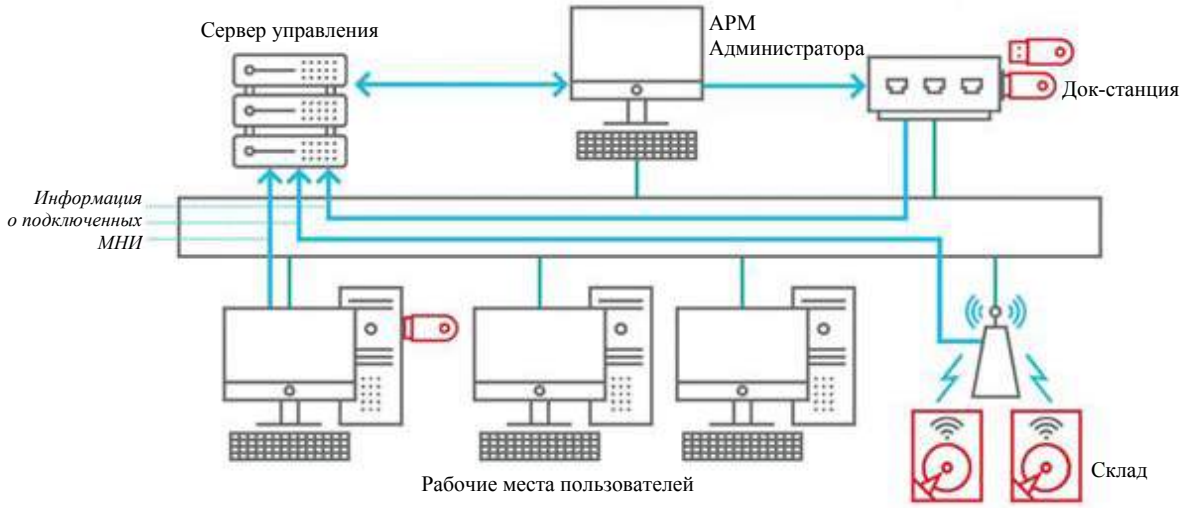


Рис. 1. Схема системы определения местоположения МНИ

Данная система состоит из следующих основных компонентов:

- клиентское программное обеспечение (ПО) устанавливается на каждое ТС, представляющее собой одно из разрешенных местоположений, и отслеживает подключение МНИ к ТС; информация о событиях подключения и отключения МНИ передается серверному ПО;
- клиентское ПО (возможно, его специфические варианты) также устанавливается на докстанцию и склад МНИ; в последнем случае оператор ПО склада может вручную ставить в ПО отметку о сдаче на склад определенного МНИ;
- серверное ПО обеспечивает сбор, обработку и хранение всей информации о МНИ, поступающей от клиентского ПО; обработка подразумевает, как минимум, определение местоположения всех контролируемых МНИ, отслеживание периодов их отсутствия и генерацию событий ИБ;
- АРМ Администратора предоставляет администратору возможности по управлению системой, которые включают в себя настройку перечня контролируемых МНИ, их разрешенных местоположений и тайм-аутов отсутствия, а также возможных реакций на события ИБ.

Стоит отметить, что и в случае применения подобной системы проблема создания эмулирующих МНИ остается актуальной. Альтернативным вариантом отслеживания местоположения МНИ, позволяющим противодействовать применению эмуляторов разрешенных к использованию МНИ, является применение несъемных RFID-меток, с помощью которых определяется местоположение МНИ (вместо отслеживания подключений МНИ к ТС).

В этом случае в разрешенных местоположениях устанавливаются RFID-считыватели, которые подключаются к ТС, оснащенным клиентским ПО, а дальнейшая работа системы эквивалентна описанной выше. Такой вариант показан на рис. 1 для склада.

Применение RFID-меток позволяет также распространить задачу отслеживания местонахождения МНИ в пределах КЗ на отслеживание местонахождения любых мобильных предметов в пределах некой территории, которые могут быть оснащены несъемными RFID-метками и отслеживание которых может по каким-либо причинам представлять интерес для организации (например, ценное малогабаритное оборудование). Такой принцип применения RFID-меток не позволит использовать их экранирование для обхода системы защиты, поскольку любой вариант невидимости RFID-метки (в т. ч. достигнутой путем ее экранирования) для системы трактуется системой как отсутствие контролируемого предмета в разрешенных местоположениях с генерацией события ИБ по истечении таймаута.

Усиление возможностей системы определения местоположения МНИ

Путем введения дополнительных разрешительных списков описанная выше система отслеживания местоположения МНИ может быть дополнена полезными дополнительными функциями, в частности:

- контроля подключения МНИ к ТС — путем оснащения клиентского ПО функцией контроля подключаемых МНИ (и прочих внешних устройств) в соответствии со списком разрешенных;
- контроля ТС, к которым может подключаться конкретный МНИ, — в случае применения специальных носителей и наличия у них списка ТС, подключение к которым разрешено; при этом использование специальных МНИ (например, оснащенных механизмами строгой аутентификации) может позволить решить описанную выше проблему защиты от эмуляции разрешенных МНИ.

Схема системы с дополнительными функциями контроля представлена на рис. 2.

По сравнению с описанной выше системой отслеживания местоположения МНИ компоненты системы с расширенными функциями предназначены для выполнения также следующих дополнительных действий:

- клиентское ПО, помимо отслеживания подключения и отключения МНИ, контролирует, разрешено ли подключение конкретного МНИ к конкретному ТС;
- серверное ПО также обеспечивает хранение списков разрешенных подключений для ТС и МНИ и загрузку списков разрешений на ТС;
- АРМ Администратора также должно давать возможность настройки дополнительных списков разрешений и загрузки списков разрешений на МНИ.

Система с подобными расширенными функциями контроля позволяет не только реализовать меру ЗНИ.3, но и полностью или частично обеспечить выполнение еще ряда мер ЗИ, предусмотренных приказами [1—4], часть которых входит в базовые наборы мер ЗИ. Такие меры ЗИ перечислены в табл. 2.

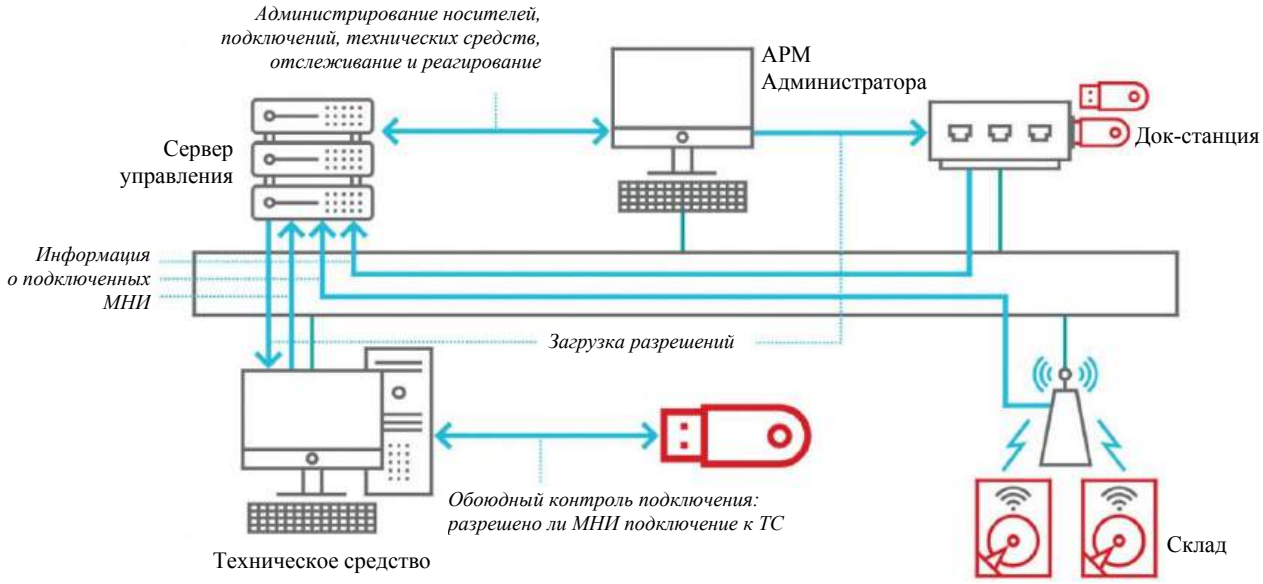


Рис. 2. Схема системы с расширенными функциями контроля

Таблица 2

Реализуемые (полностью или частично) меры ЗИ

Обозначение	Мера ЗИ	Вхождение в базовые наборы
ЗНИ.0	Разработка политики защиты МНИ	Входит частично
ЗНИ.1	Учет МНИ	Входит частично
ЗНИ.2	Управление доступом к МНИ	Входит частично
ЗНИ.3	Контроль перемещения МНИ за пределы КЗ	Не входит
ЗНИ.4	Исключение возможности несанкционированного ознакомления с содержанием информации, хранящейся на МНИ, и использования носителей информации в иных ИС	Не входит
ЗНИ.7	Контроль подключения МНИ	Входит частично
ИАФ.4	Управление средствами аутентификации, в т. ч. хранение, выдача, инициализация, блокирование средств аутентификации и принятие мер в случае утраты и (или) компрометации средств аутентификации	Входит
УПД.15	Регламентация и контроль использования в ИС мобильных ТС	Входит частично

Заключение

Реализация контроля перемещения МНИ за пределы КЗ путем организации досмотровых мероприятий по периметру КЗ затруднена. Более эффективным выглядит постоянное отслеживание местонахождения МНИ внутри КЗ.

Такое отслеживание может являться одной из функций системы регистрации и контроля МНИ, направленной на реализацию комплекса мер ЗИ в соответствии с приказами [1—4], при этом действие такой системы может быть расширено на отслеживание местонахождения любых предметов (в случае целесообразности такового), которые можно контролировать через их физические подключения к ТС или с помощью RFID-меток.

Литература

1. Приказ ФСТЭК России от 11 февраля 2013 г. № 17 "Об утверждении требований и защите информации, не составляющей государственную тайну, содержащейся в государственных информационных системах".

2. Приказ ФСТЭК России от 18 февраля 2013 г. № 21 "Об утверждении состава и содержания организационных и технических мер по обеспечению безопасности персональных данных при их обработке в информационных системах персональных данных".

3. Приказ ФСТЭК России от 14 марта 2014 г. № 31 "Об утверждении требований к обеспечению защиты информации в автоматизированных системах управления производственными и технологическими процессами на критически важных объектах, потенциально опасных объектах, а также объектах, представляющих повышенную опасность для жизни и здоровья людей и для окружающей природной среды".

4. Приказ ФСТЭК России от 25 декабря 2017 г. № 239 "Об утверждении требований по обеспечению безопасности значимых объектов критической информационной инфраструктуры Российской Федерации".

A method to implement the requirement to control the movement of data carriers outside the controlled zone

S. P. Panasenko
JSC "Active-soft", Moscow, Russia

The article considers the problem of controlling the movement of data carriers out the controlled zone. The classic approach to the implementation of such control is analyzed and an alternative method is proposed; the method is based on the continuous detection of the location of controlled data carriers and on the proof of their presence in definite places within the controlled zone. The proposed approach can be extended to control the location of any kind of wearable items, which are valuable or important from the point of view of information security.

Keywords: information security measures, data carriers, controlled zone, location tracking, RFID.

Bibliography — 4 references. *Received May 10, 2023*

УДК 004; 62-5
DOI: 10.52190/2073-2600_2023_2_66
EDN: EBCUXL

Направления повышения кибербезопасности систем управления мобильной техники

А. В. Пузанов, канд. техн. наук
Ковровская государственная технологическая академия им. В. А. Дегтярева,
г. Ковров, Владимирская обл., Россия

К. А. Пузанова
Московский авиационный институт (национальный исследовательский университет),
Москва, Россия

Работа посвящена вопросам повышения информационной защищенности систем управления мобильной техники. На основе приведенной схемы информационного обмена интеллектуальных систем управления мобильной техники предложены направления реализации противодействий киберугрозам применительно к существующим и вновь разрабатываемым изделиям.

Ключевые слова: интеллектуальные системы управления, мобильная техника, защита информации, кибербезопасность.

Четвертая промышленная революция, и в русле ее векторов Стратегия научно-технического развития РФ определяют в числе приоритетов — переход к цифровым, интеллектуальным производственным технологиям, роботизированным системам и искусственному интеллекту [1]. В отражении этих тенденций развитие мобильной техники (МТ) реализуется интеллектуализацией алгоритмов управления, внедрением и совершенствованием встроенной электроники, программного обеспечения, датчиков и других технологий взаимодействия с внешним миром и между собственными компонентами [2, 3].

Одновременно с интеллектуализацией оборудования фиксируется рост количества инцидентов (преступлений) в сфере информационных технологий, и в частности, применительно к техническим объектам с цифровыми системами управления. Подобные устройства могут содержать уязвимости, которыми могут воспользоваться киберпреступники, что влечет возникновение угроз пользователя или технической системы [4]. Таким образом, обеспечение безопасности является одной из основных проблем, связанных с интеллектуализацией приборов и систем [5].

Пузанов Андрей Викторович, доцент кафедры приборостроения.
E-mail: puzanov@dksta.ru
Пузанова Ксения Андреевна, студентка.
E-mail: puzanova_2017_ksu@mail.ru

Статья поступила в редакцию 4 апреля 2023 г.

Анализ регулярных отчетов компании McAfee об угрозах в сфере информационной безопасности позволяет сделать вывод о том, что ежеминутно появляется до 180 новых киберугроз. Стоит предположить, что объем атак с использованием различных типов уязвимостей безопасности будет увеличиваться и далее. Простота базовых веб-интерфейсов, использующихся в системах управления техническими объектами, делает их уязвимыми перед удаленными атаками. Даже при наличии дополнительных методов повышения кибербезопасности устройств, большинство из них трудно реализуемо ввиду недостаточности вычислительной мощности. К тому же большинство информационно-технических решений использует ПО, имеющее собственный набор проблем и уязвимостей. Таким образом, задача повышения устойчивости к кибератакам интеллектуальных систем управления — актуальная научно-техническая задача. По данным отчета Gartner, рост затрат на кибербезопасность в 2023 г. увеличится на 11,3 % [6].

Причинами развития и актуализации киберугроз в мобильной технике является тот факт, что базовые технологии реализации встраиваемых интеллектуальных систем разрабатывали и эволюционировали без учета требований безопасности, поскольку основной задачей производителей было минимизировать себестоимость и сроки разработки, сократить излишние издержки производства. В результате подобной политики узловые микрочипы работают на предельных режимах. Из-за недостатка вычислительных ресурсов большинство

инструментов безопасности не могут быть интегрированы в наследуемые устройства, что делает их уязвимыми для киберпреступлений [7].

Системы управления мобильной техникой (СУ МТ), применительно к встраиваемым исполнениям мобильной техники, классифицируются по характеру использования информации на автономные и контролируемые системы. Автономные системы оперируют информацией внутри контролируемой зоны, получая данные из окружающего мира собственными датчиками и формируя сигналы встроенным приводам. Искажению подвержена либо внешняя информация (визуальное искажение или воздействие на чувствительный элемент датчика), либо информация, транслируемая МТ на средства отображения или фиксации данных. Контролируемые (извне) системы, кроме перечисленного получают управляющие, программные или корректирующие сигналы из внешней (неконтролируемой) среды. Тем самым данные каналы обмена информацией могут подвергаться стороннему вмешательству — блокировке или искажению данных.

Элементы системы управления МТ (управление движением, включая основное, вспомогательное, регулирование скорости старт и торможение, управление технологическим оборудованием, связь, управление датчиками и обработки данных) реализуются посредством программируемой электроники (центральная система управления, внешние пульты дистанционного управления или контроля, системы связи, системы управления движением, системы управления оборудованием, внешним или внутренним, системы очувствления или видеонаблюдения) (см. рисунок).

Устройства систем управления техническими объектами, потенциально уязвимые для кибератак (рисунок):

- компоненты систем управления (оснащенные встроенными технологиями сбора, обработки, хранения, передачи информации, интеллектуального принятия решений);
- вычислительные системы;
- компоненты связи (между устройствами — беспроводные или кабель-канальные сети);
- компоненты обработки информации (разнородного содержания: видео или аудио, данных сгенерированных в реальном времени интеллектуальными датчиками, устройствами и т. п.);
- системы искусственного интеллекта, предиктивной аналитики;
- системы мониторинга (компоненты сбора, накопления, хранения и обработки данных).

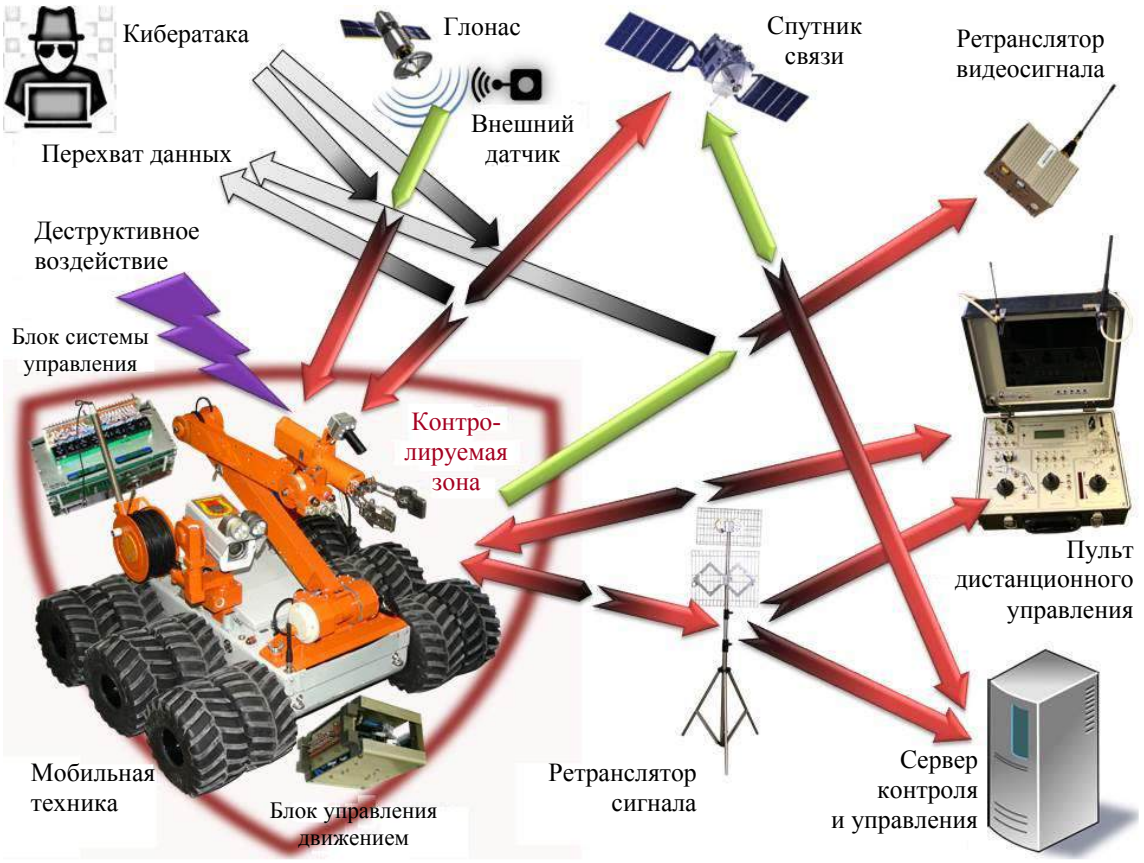


Схема информационного обмена системы управления мобильной техники

Основной целью кибератак является искажение обрабатываемых данных и результатов работы устройств при неверных данных, нарушение текущих процессов (отказ в обслуживании), раскрытие любой конфиденциальной информации, хранящейся на устройстве (ключи и пароли). Основные направления киберугроз применительно к мобильной технике с цифровой системой управления [8]:

- умышленные действия: вредоносное ПО; эксплойт; целевая атака; DDoS-атака; скомпрометированное устройство; утрата конфиденциальности; модификация информации;
- перехват информации: подключение к активной сессии, перехват соединения или информации;
- технический сбой: уязвимости на программном уровне, сторонние ошибки;
- физическое воздействие: нарушение функциональности устройства; уничтожение устройства, отключение питания;
- нарушение систем виртуализации (полной или дополненной) реальности.

По степени тяжести последствий для внешней среды и внутренних компонентов, а также в соответствии с применяемой моделью атаки формулируются категории угроз, алгоритмы их нейтрализации.

Информационная защита интеллектуальной системы управления техническими объектами (ее кибербезопасность) призвана обеспечить:

- безопасность функционирования (по отношению к людям, внутреннему оборудованию и внешним объектам);
- непрерывность, стабильность и управляемость функционирования [9];
- эффективность выполнения функций в соответствии с заданными целями [10];
- устойчивость сохранения параметров системы управления [11];
- возможность адаптации системы управления к нештатным ситуациям;
- требуемый уровень доверия к информационным каналам [12].

ГОСТами и РД рекомендуются и нормируются орг-тех мероприятия по разграничению доступа и формирующие эталонные модели доступа к информационным каналам [13—22]. В данной работе мы это не рассматриваем. В данной работе ограничимся рассмотрением технических противодействия киберугрозам встраиваемой электроники систем управления мобильной техники средств.

Таким образом формируются контролируемые зоны и коридоры потоков информации в шифрованном и открытом виде. Формулируются модели угроз, атак, а также проблемы безопасности, применительно к устройствам и системам управления техническими объектами [5]:

- уязвимость устройств и систем;
- конвергенция информационных и операционных технологий;
- устаревшие промышленные системы управления;
- небезопасные протоколы;
- человеческий фактор;
- неиспользуемые функции;
- обеспечение безопасности продукта после его реализации.

Кибербезопасность интеллектуальных систем управления техническими объектами может быть реализована по направлениям: аппаратная, программная или комбинированная защита [13].

Рассмотрим достоинства и недостатки программных и аппаратных средств защиты интеллектуальных устройств систем управления технических объектов к воздействиям кибератак:

- чисто программный механизм защиты, используемый в устройстве, сам по себе может быть уязвим для удаленных атак.
- программные средства требуют регулярных обновлений и модификаций, поскольку вредоносное ПО перманентно эволюционирует.
- модификация ПО большинства удаленных устройств не всегда физически реализуема.
- аппаратные системы безопасности используют аппаратные модули для сбора информации о микроархитектуре для анализа преобладающих угроз и уязвимостей на программном уровне.

Аппаратная защита может быть как встроенной в процессор или выполненной отдельным устройством. Защита интеллектуальных систем управления технических объектов призвана обеспечить целостность системного и прикладного ПО, защиту данных (шифрование сбора, передачи, хранения), а также защиту линий связи (шифрование, контроль целостности).

Учитывая ограниченность вычислительных ресурсов, повышение устойчивости к кибератакам реализуется либо модернизацией (включающей средства защиты), либо встраиванием дополнительного модуля, либо алгоритмическим выбором времени для проведения проверки (в периоды снижения основных вычислений) или в случае отклонений в поведении от эталонной модели или повреждении данных (переключение приоритета) [16].

Для "легких", энергодефицитных систем криптографические аппаратные решения не подходят. В этом случае применяется аппаратный мониторинг событий микроархитектуры (Security

information and event management). При этом накапливается информация выполнения отдельных процессов и посредством фильтров выявляются отклонения, связанные с кибератаками. На внешнем мониторинге работы системы и связанного хоста основана разработка Адаптивного обнаружения вторжений в сетях (BehavioR based Adaptive Intrusion detection in Networks). При этом используются методы машинного обучения для моделирования поведения приложений и сетевой статистики: при обнаружении кибератаки IP-адрес заносится в черный список (и удаляется при необходимости). Для минимизации ошибочного определения или пропуска кибератак применяются различные методы машинного обучения, позволяющие изучать и различать такие события, а также идентифицировать любой вид аномалии.

Другим, относительно "дешевым" вариантом контроля исполняемого кода является его периодическое дизассемблирование и сравнение с эталонным кодом, контроль целостности базового кода, его защита и блокировка.

Модернизация существующих изделий мобильной техники реализуется введением дополнительного шлюза проверки корректности и целостности данных системой управления. Для новой техники — встраиванием алгоритмов контроля и проверки (антивирус) с обновлением баз.

Кроме этого, на базовой станции управления возможно внедрение системы оценки модели поведения управляемого объекта с точки зрения различных моделей воздействия, аудит актуальных уязвимостей, дистанционное тестирование, в том числе на возможность или успешность кибератак [14, 15].

Заключение

С развитием информационных технологий и соответствующей элементной базы возникает отдельное направление противоправных действий, ориентированных на интеллектуальные системы управления техническими объектами. В зависимости от используемых технических и программных решений необходимо применять программно-аппаратные средства для минимизации вероятности повреждения систем, а также для их скорейшего восстановления.

Кибербезопасность реализуется как "на борту", внедрением дополнительных программно-аппаратных модулей защиты, так и анализом поведения мобильной техники на внешнем ресурсе при анализе данных с датчиков с использованием эталонной модели цифрового двойника.

Литература

1. Национальная технологическая инициатива (НТИ). — [Электронный ресурс]. Режим доступа: URL: https://fea.ru/compound/national-technology-initiative (дата обращения: 03.08.2021).
2. Пузанов А. В. Мультидисциплинарный анализ систем управления мобильной техники // Автоматизация. Современные технологии. 2016, № 10. С. 13—17.
3. Пузанов А. В. Transdisciplinary models of hydraulic drives of mobile machinery // Системный анализ и прикладная информатика. 2018, № 4. С. 51—55.
4. Internet of things — from research and innovation to market deployment / O. Vermesan, P. Friess (eds.). Aalborg, Denmark : River Publishers, 2014. 373 p.
5. Верещагина Е. А., Капецкий И. О., Ярмонов А. С. Проблемы безопасности Интернета вещей: учебн. пос., — Москва: Мир науки, 2021. — Сетевое издание. Режим доступа: https://izd-mn.com/PDF/20MNNPU21.pdf.
6. Gartner Identifies Three Factors Influencing Growth in Security Spending. — URL: https://www.gartner.com/en/ newsroom/press-releases/2022-10-13-gartner-identifies-three-factors-influencing-growth-i.
7. Erguler I. A potential weakness in RFID-based Internet-of-things systems // Pervasive and Mobile Computing. 2015. Vol. 20. P. 115–126. https://doi.org/10.1016/j.pmcj. 2014.11.001.
8. Baseline Security Recommendations for IoT in the context of Critical Information Infrastructures / ENISA. Hague : European Union Agency For Network And Information Security, 2017. 103 p. DOI: 10.2824/03228.
9. ГОСТ Р 27.102-2021. Надежность в технике. Надежность объекта. Термины и определения: национальный стандарт Российской Федерации: издание официальное: утв. и введ. Приказом Федерального агентства по техническому регулированию и метрологии от 8 октября 2021 г. N 1104-ст.: Введ. 2022-01-01 / Разработан ЗАО "Научно-исследовательский центр контроля и диагностики технических систем" — Москва : Стандартинформ, 2022. —46 с.
10. Энциклопедия кибернетики" под ред. В. М. Глушкова, Т. 1., Киев, — Мир. 1974. — 606 с.
11. Горский Ю. М., Астафьев В. И., Казначеев В. П. и др. Гомеостатика живых, технических, социальных и экологических систем : По материалам семинаров. Отв. ред. Ю. М. Горский. — Новосибирск : Наука : Сиб. отд., 1990. — 346 с.
12. ГОСТ Р ИСО/МЭК 15408-3-2002. Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий = Ч. 3. Требования доверия к безопасности : Information technology. Security techniques. Evaluation criteria for IT security. Part 3. Security assurance requirements : принят и введ. в действие Постановлением Госстандарта России 04.04.02 № 133-ст : Введ. 2004-01-01 / Госстандарт РФ. — Москва : Изд-во стандартов, 2002. — 107 с.
13. Информационная безопасность устройств IoT с использованием аппаратной поддержки. — URL: https://habr.com/ru/post/534300/.
14. Минзов А. С., Невский А. Ю., Баронов О. Р. Управление рисками информационной безопасности: монография / под ред. А. С. Минзова. — Москва : ВНИИгеосистем, 2019. — 110 с.
15. Минзов А. С., Черемисина Е. Н., Токарева Н. А., Бобылева С. В. Моделирование рисков информационной безопасности в цифровой экономике: монография / под ред. А. С. Минзова. — Москва : КУРС, 2021. — 112 с.
16. ГОСТ Р 57700.37-2021 Компьютерные модели и моделирование. Цифровые двойники изделий. Общие положения: национальный стандарт РФ: издание официальное: утв. и

введ. в действие Приказом Федерального агентства по техническому регулированию и метрологии от 16 сентября 2021 г. № 979-ст. Введ. 2022-01-01 / Разработан ФГУП "РФЯЦ-ВНИИЭФ" совместно с Санкт-Петербургским политехническим университетом Петра Великого. — Москва: ФГБУ "РСТ", 2021, — 15 с.

17. Методические рекомендации по обеспечению с помощью криптосредств безопасности персональных данных при их обработке в информационных системах персональных данных с использованием средств автоматизации (утв. ФСБ РФ 21.02.2008 N 149/54-144) // Федеральная служба по надзору в сфере связи, информационных технологий и массовых коммуникаций. [Электронный ресурс]. Режим доступа: URL: https://26.rkn.gov.ru/law/p7096/p14816/.

18. Концепция защиты средств вычислительной техники и автоматизированных систем от несанкционированного доступа к информации // Федеральная служба по техническому и экспортному контролю. РД. Утв. решением ГТК при Президенте РФ от 30 марта 1992 г. [Электронный ресурс]. Режим доступа: URL: https://fstec.ru/tekhnicheskaya-zashchita-informatsii/dokumenty/ 114-spetsialnye-normativnye-dokumenty/ 384-rukovodyashchij-.

19. Приказ об утверждении требований о защите информации, не составляющей государственную тайну, содержащейся в государственных информационных системах от 11 февраля 2013 г. № 17 // Федеральная служба по техническому и экспортному контролю. [Электронный ресурс]. Режим доступа: URL: https://fstec.ru/normotvorcheskaya/akty/53-prikazy/702.

20. Методический документ методика оценки угроз безопасности информации. Утв. ФСТЭК России 5 февраля 2021 г. // Федеральная служба по техническому и экспортному контролю. — Москва. 2021, —83 с. [Электронный ресурс]. Режим доступа: URL: https://fstec.ru/tekhnicheskaya-zashchita-informatsii/dokumenty/114-spetsialnye-normativnye-dokumenty/ 2170-metodicheskij-dokument-utverzhden-fstek-rossii-5-fevralya-2021.

21. ГОСТ Р 51275-2006. Защита информации. Объект информатизации. Факторы, воздействующие на информацию. Общие положения = Protection of information. Object of informatisation. Factors influencing the information. General principles : национальный стандарт Российской Федерации: взамен ГОСТ Р 51275-99 : введ. 2008-02-01 / Федеральное агентство по техническому регулированию и метрологии. — Москва : Стандартинформ, 2007. — 7 с.

22. ГОСТ Р ИСО/МЭК 15408-1-2012. Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий = Ч. 1. Введение и общая модель : Information technology. Security techniques. Evaluation criteria for IT security. Part 1. Introduction and general model : национальный стандарт РФ : взамен ГОСТ Р ИСО/МЭК 15408-1-2008 : введ. 2013-12-01 / Федеральное агентство по техническому регулированию и метрологии. — Москва : Стандартинформ, 2009. — 35 с.

Directions for improving cybersecurity of mobile technology control systems

A. V. Puzanov
Kovrov State Technological Academy named after V. A. Degtyareva, Kovrov, Vadimir region, Russia

K. A. Puzanova
Moscow Aviation Institute (National Research University), Moscow, Russia

The work is devoted to improving the information security of mobile technology control systems. Based on the above information exchange scheme of intelligent control systems of mobile technology, directions for implementing counteraction to cyber threats in relation to existing and newly developed products are proposed.

Keywords: intelligent control systems, mobile technology, information protection, cybersecurity.

Bibliography — 22 references. *Received April 4, 2023*

Правила для авторов по оформлению рукописей статей, предлагаемых к публикации в журналах

Направляя рукопись статьи в редакцию журнала, авторы передают редколлегии и издателю журнала безвозмездное неисключительное право опубликовать ее на русском языке в качестве статьи в печатной и электронной версиях журнала в сети Интернет. При этом за авторами сохраняются их интеллектуальные права на рукопись статьи (в том числе "авторское право"). В связи с этим и с учетом Четвертой части (Раздел VII) Гражданского кодекса РФ авторами должно быть представлено в редакцию письмо в следующей форме:

Лицензионный договор о передаче права на публикацию
(издательский лицензионный договор)

Мы, нижеподписавшиеся, авторы рукописи ……..……., предоставляем редколлегии журнала и журнала и издателю ФГУП «НТЦ оборонного комплекса «Компас» ……..……
(название журнала)
безвозмездную простую (неисключительную) лицензию на публикацию рукописи статьи как в печатной, так и в электронной версиях журнала.

Мы подтверждаем, что данная публикация не нарушает интеллектуальных прав других лиц или организаций.

Подписи авторов: …… (ф.и.о., ученая степень, дата)
……..…….
……...…….

Статья должна быть подписана всеми авторами. В случае нескольких авторов указывается фамилия автора, ответственного за переписку с редакцией. Рукопись статьи направляется на адрес редакции журнала: 125424, Москва, Волоколамское ш., д. 77, ФГУП «НТЦ оборонного комплекса «Компас», Редакция журнала. Дополнительная информация может быть получена в редакции при обращении по телефону: 8 (495) 491-43-17 или по E-mail: izdanie@ntckompas.ru.

Каждая статья рецензируется в соответствии с решением редколлегии журнала. Рецензенты выбираются из числа ученых и специалистов, компетентных в вопросах, рассматриваемых в статье, и имеющих собственные публикации в данном направлении. Решение о порядке публикации статьи принимается редколлегией с учетом мнения рецензентов. В случае разногласий среди членов редколлегии окончательное решение принимает главный редактор. При полном отклонении рукописи статьи от публикации редакция журнала направляет авторам мотивированный отказ. По этому факту редакция не вступает в полемику с авторами.

Комплектование статьи (обзора)

Статья должна быть представлена в редакцию в следующем комплекте:

- сопроводительное письмо, подписанное руководством организации;
- экспертное заключение о возможности опубликования в открытой печати и распространения в Российской Федерации и других государствах. При отсутствии в организации экспертной комиссии вместо экспертного заключения может быть представлено соответствующее заявление на имя главного редактора журнала от имени руководства организации, где работают авторы. В экспертном заключении или заявлении в обязательном порядке должна быть отражена возможность открытой публикации и передачи публикуемых материалов за границу.
- лицензионный договор по приведенному образцу;
- рукопись статьи — объем статьи (без рисунков) не должен превышать 10 страниц формата А4 при 1,5 межстрочных интервалах, а объем статьи обзорного характера — 20 страниц. Рекомендуемая гарнитура — New Times Roman. Размер шрифта — 12;
- Материал статьи представляется в редакцию в печатном виде (на бумажном носителе) и в электронном варианте с текстом в формате Word на электронную почту.

Оформление статьи:

- статья начинается с указания УДК;
- название статьи набирается строчными буквами (кроме начальной прописной) полужирным шрифтом, размер шрифта 14, для остального текста используется простой шрифт размером 12, причем рекомендуемая гарнитура шрифта — Times New Roman;
- после названия — список авторов, инициалы авторов предшествуют их фамилиям;

- далее представляется аннотация статьи (10—15 строк с раскрытием цели работы и её основных результатов);
- далее приводится список ключевых слов для данной статьи (не более десяти);
- страницы текста нумеруются без пропусков и добавлений литерных обозначений (типа 1*а*, 2*б* и т. п.), причем в сквозную нумерацию должны быть включены все элементы статьи;
- внизу первой страницы текста помещается отдельный абзац (с полужирным шрифтом), содержащий контактную информацию об авторе (или авторах) в следующем виде: фамилия, имя, отчество, должность, ученая степень, почтовый адрес предприятия, телефон, E-mail;
- основной текст статьи должен начинаться с четкой постановкой цели и задач работы, сопровождаемой аргументами в пользу ее выполнения на фоне существующего состояния затронутой в статье проблемы. Дальнейший текст статьи также должен иметь смысловые рубрикаторы (разделы и подразделы) без их нумерации. Заканчиваться статья должна отдельным разделом «Заключение» с перечислением основных результатов, следующих из них выводов и, по возможности, предложений по развитию исследований и использованию их результатов.
- после основного текста — список использованных источников "Литература" (не менее 5 источников); Список использованной литературы должен соответствовать всем ссылкам на внешние источники в тексте статьи. Ссылки оформляются в квадратных скобках, например, [1—6], [7, 8]. Внутренние ссылки, т. е. ссылки на формулы, рисунки и таблицы статьи оформляются с использованием круглых скобок, например, формула (1), уравнение (4), (рис. 3), (табл. 2). Любые ссылки в подписях к рисункам и в самих рисунках не рекомендуются;
- далее размещается подробная англоязычная информация о статье: название статьи, фамилия и инициалы авторов (английская транслитерация), предприятие, аннотация, ключевые слова (Keywords).

Оформление рисунков:

- рисунки и графики вставляются непосредственно в нужном месте в статье и в желаемом масштабе.
- рядом с осями графиков указываются отображаемые физические величины только в символьной (буквенной) форме, а через запятую — размерность величины по-русски (прямым шрифтом). Различные кривые на графиках рекомендуется нумеровать, даже если они характеризуются отдельным цветом или типом линии. Графики представляются только на белом фоне. Вспомогательные сетки на площади графика не допускаются;
- подписи под соответствующими рисунками (полужирный курсивный) представляются в нужных местах текста. Каждая подпись должна быть по возможности лаконичной, но емкой по содержанию.

Оформление формул:

- простые формулы вводить в текст в формате используемого текстового редактора, более сложные формулы — с использованием редактора формул MathType;
- стандартные математические обозначения (например, max, log, sin, exp и т. д.) должны быть набраны прямо. То же относится к цифрам и числам;
- для символьного обозначения невекторных физических (технических) величин использовать только латинский и греческий алфавиты, при этом в тексте для греческих букв использовать прямой шрифт, для латинских букв — наклонный шрифт (курсив);
- векторы и матрицы обозначать полужирным прямым шрифтом;
- для нижних и верхних индексов применять арабские цифры, латинские или греческие буквы. Если индекс представляет собой сокращенную форму русского слова — характеристики, то допустимо использовать в его обозначении русские буквы (прямой шрифт), например $U_{вх}$, $I_{вых}$, $\nu_{гр}$ и т. п.
- размерность физических величин обозначается всегда только по-русски прямым шрифтом.

Оформление таблицы:

- содержание таблицы не должно дублировать данные, приводимые на графиках или в тексте;
- графы должны иметь название без сокращения отдельных слов.

Рисунки, формулы и таблицы должны иметь свою отдельную сквозную нумерацию. Если на конкретную формулу нет дополнительных (возвратных) ссылок в тексте или она в единственном числе, то нумерация ее не нужна. Единственные таблица и/или рисунок также не нумеруются.

При публикации в журнале каждая статья (в контактной информации) сопровождается сноской со знаком охраны авторского права ©, поставленным перед фамилией автора (фамилиями авторов) и годом издания.

Авторы (или автор) каждой статьи после выхода журнала в свет имеют право на получение от редакции электронной версии статьи в PDF-формате (редактор Adobe Acrobat).