Модель самовосстанавливающегося вычислительного процесса облачной информационно-вычислительной системы в условиях информационно-технических воздействий

А. А. Балябин
Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В.И. Ульянова (Ленина)
treven.wt@yandex.ru

***Аннотация*. В работе определены типы информационно-технических воздействий, влияющих на нарушение семантической корректности вычислительного процесса облачной информационно-вычислительной системы в соответствии с международной базой знаний о компьютерных атаках. Предложена модель самовосстанавливающегося вычислительного процесса облачной информационно-вычислительной системы в условиях информационно-технических воздействий.**

Ключевые слова: кибербезопасность; киберустойчивость; облачные платформы; самовосстановление облачных вычислений

I. Введение

Современные киберфизические системы развиваются в направлении внедрения облачных, туманных и пограничных вычислений, интернета вещей (IoT) и иных технологий Smart Grid [1]. Это обусловлено постоянным увеличением требований к вычислительным ресурсам, мобильности развертывания и поддержки программного обеспечения и большими объемами обрабатываемых данных. Также растет и сложность программного обеспечения. Программное обеспечение, функционирующее на базе облачных платформ, как правило обладает распределенным, многоуровневым характером и может быть физически разнесено между отдельными серверами (2/3/N-Tier) [2].

Вместе с этим наблюдается постоянный рост количества и сложности кибератак. Так только за 2022 год общее количество кибератак выросло на 38 % [3], а общее количество выявленных за год уязвимостей, занесенных в международную базу данных MITRE CVE, составило 25059 [4], что превысило аналогичные показатели за все предыдущие годы наблюдений. Серьезную угрозу безопасности представляют целенаправленные атаки (Advanced Persistent Threats, APT), характеризующиеся высокой сложностью и интенсивностью, целенаправленным характером и большим количеством задействованных ресурсов. Такие атаки, как правило организуются крупными хакерскими группировками или специальными службами иностранных государств и направлены на предприятия оборонно-промышленного комплекса, государственные учреждения и объекты критической инфраструктуры [5]. Наиболее опасными являются так называемые атаки «нулевого дня» (0-day), использующие новые, ранее не встречавшиеся уязвимости, поскольку они не могут быть своевременно обнаружены и отражены.

Применяемые на сегодняшний день методы и средства защиты несовершенны и не способны в полной мере удовлетворить растущие требования к безопасности информационно-вычислительных систем, а подходы к обеспечению надежности и устойчивости, сводящиеся к реконфигурации, *n*-кратному резервированию или перезапуску системы, не способны предотвратить возможные катастрофические последствия от реализации таких угроз [6].

Очевидно, что появление новых уязвимостей, способов их эксплуатации, тактик и техник кибератак неизбежно ввиду постоянного роста сложности аппаратно-программного обеспечения облачных информационно-вычислительных систем. С другой стороны, очевидна необходимость обеспечения требуемой надежности и устойчивости информационно-вычислительных систем и недостаточность существующих моделей, методов и средств защиты. Наличие такого противоречия делает актуальной задачу разработки новой модели вычислительного процесса облачной информационно-вычислительной системы, обладающего возможностью обнаружения нарушений его корректности и самовосстановления.

В работе предложена модель самовосстанавливающегося вычислительного процесса облачной информационно-вычислительной системы в условиях информационно-технических воздействий.

II. Эталонная модель облачной информационно-вычислительной системы

Облачные вычисления предоставляют собой модель организации удаленного доступа к конфигурируемым разделяемым вычислительным ресурсам, которые могут быть свободно выделены и освобождены с минимальными расходами на управление или взаимодействие с провайдером услуг [7]. Эталонная модель облачной информационно-вычислительной системы, предложенная NIST (National Institute of Standards and Technology) [8], представлена на рис. 1.

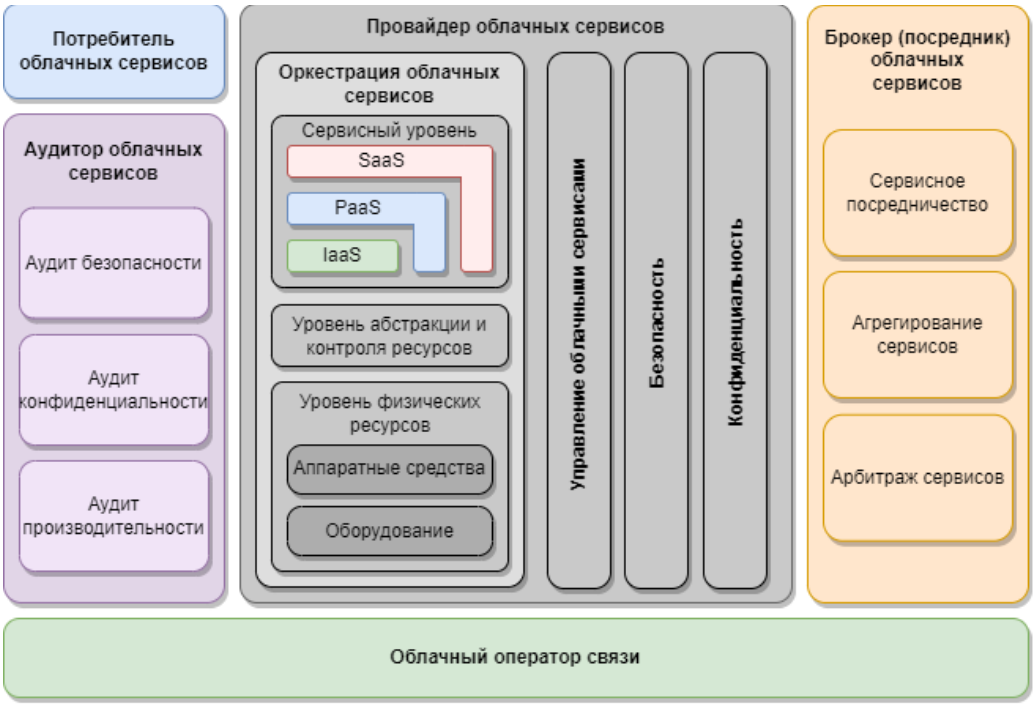

Рис. 1. Эталонная модель облачной информационно-вычислительной системы

В соответствии с моделью выделяются следующие модели предоставления облачных сервисов [7]:

- Программное обеспечение как сервис (*Software as a Service, SaaS*) – прикладное программное обеспечение функционирует в облачной инфраструктуре, доступно с различных клиентских устройств для использования, поддерживается и администрируется провайдером. При этом пользователю не доступна иная виртуальная и физическая инфраструктура;
- Платформа как сервис (*Platform as a Service, PaaS*) – пользователю предоставляется набор базового программного обеспечения для разработки, тестирования и развертывания своего программного обеспечения, однако доступа к элементам инфраструктуры, таким как операционная система, сетевые порты и др., у пользователя нет. Провайдер поддерживает функционирование платформы и ее компонентов, таких как среда разработки (IDE), набор инструментов разработки (SDK), инструменты развертывания и др.;
- Инфраструктура как сервис (*Infrastructure as a Service, IaaS*) – пользователю предоставляется возможность самостоятельного управления облачной инфраструктурой, включая операционные системы, системное и прикладное программное обеспечение, вычислительные ресурсы.

Кроме этого, выделяют следующие модели развертывания [7]:

- Приватное облако – облачная инфраструктура выделена и используется только одной компанией, при этом она может размещаться как у облачного провайдера, так и в центре обработки данных самой компании;
- Публичное облако – облачная инфраструктура предоставляется одновременно множеству компаний, у которых нет физического доступа к оборудованию, для совместного пользования;
- Гибридное облако – является комбинацией публичного и приватного облаков.

Данную классификацию необходимо учитывать при построении модели самовосстанавливающего вычислительного процесса, поскольку в зависимости от наличия или отсутствия доступа к физическому оборудованию, вычислительным и сетевым ресурсам, средствам разработки, системному и прикладному программному обеспечению возможны различные уровни контроля вычислительных процессов.

III. Определение типов информационно-технических воздействий на облачные информационно-вычислительные системы

Следует сделать небольшое отступление и ввести некоторые термины.

Под *программой* понимается совокупность конструктивных объектов, представляющих собой слова (тексты), графы, некоторые конечные множества и системы множеств. Под *вычислительным процессом* понимается процесс последовательного преобразования состояний системы, осуществляемый во время выполнения программы путем применения к ней некоторого универсального алгоритма [10]. Иными словами, вычислительный процесс представляет собой выполняющуюся программу в совокупности ее элементов: команд и данных. Программа описывается элементами некоторого языка, корректный способ конструирования которых определяется его *синтаксисом*. В свою очередь под *семантикой* как

правило понимают формализованное смысловое значение синтаксически корректных конструкций языка описания программы – действия абстрактного исполнителя.

В данной работе будем рассматривать информационно-технические воздействия на облачные информационно-вычислительные системы, приводящие к нарушению семантики вычислений. Под нарушением семантики вычислений будем понимать отклонение потока управления, сопровождающееся выполнением синтаксически корректных конструкций, не предусмотренных семантикой исходной программы.

В работе [9] подробно рассмотрена модель угроз облачной информационно-вычислительной системы, основные атаки и методы обнаружения вторжений. Схема таксономии атак, предложенная авторами, приведена на рис. 2.

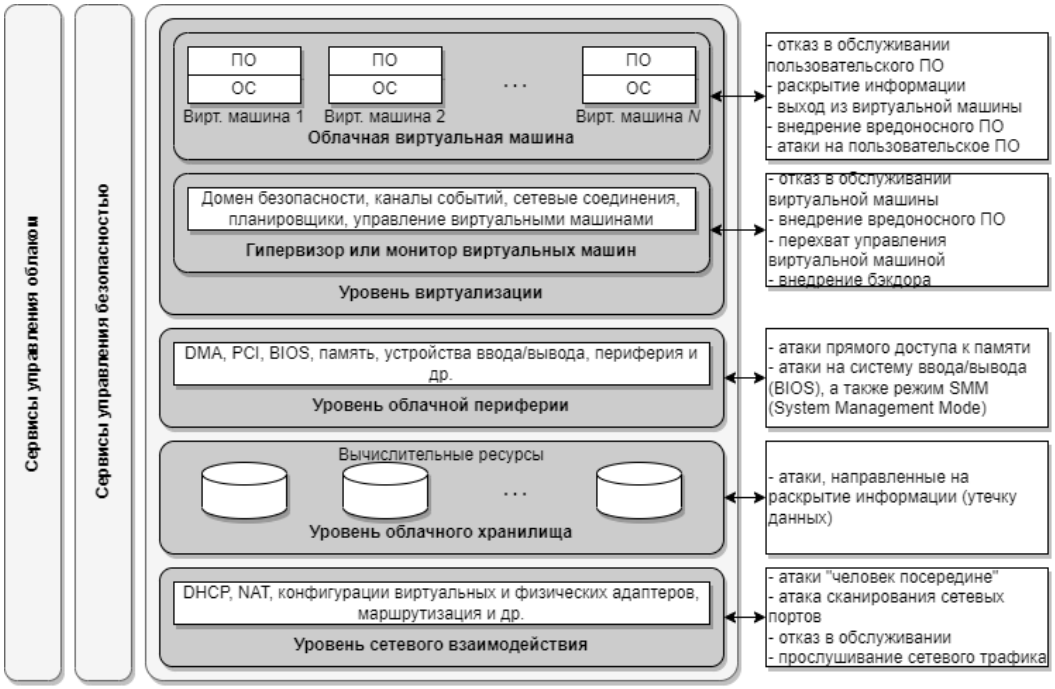

Рис. 2. Таксономия атак в облачной среде

Определим подмножество техник атак, влияющих на семантику вычислительного процесса, на основе международной базы знаний MITRE ATT&CK [11]. Отметим, что данное подмножество не претендует на полноту, поскольку тактики и техники компьютерных атак непрерывно совершенствуются. Для более полного рассмотрения формальных аспектов нарушения семантики вычислений при реализации приведенных атак необходимо проведение отдельного исследования.

A. Внедрение кода в процесс (T1055)

Под внедрением кода в процесс понимается выполнение произвольного кода в адресном пространстве некоторого уже выполняющегося процесса, которое может предоставить доступ к памяти, системным или сетевым ресурсам, повышенным привилегиям и др. Примерами данного типа атак являются:

- внедрение динамической библиотеки, сопровождающееся выполнением произвольного кода;
- внедрение исполняемого файла (PE);
- прямая запись в память процесса.

B. Загрузка разделяемых модулей (T1129)

Злоумышленник может выполнить произвольный вредоносный код путем загрузки разделяемого модуля (динамической библиотеки). Например, в ОС «Windows» загрузка динамической библиотеки выполняется посредством вызова команд WinAPI, таких как «CreateProcess», «LoadLibrary» и др., при этом после загрузки модуля в память осуществляется его инициализация путем выполнения функции «DllMain» или аналогичной.

C. Эксплуатация уязвимостей в публично доступном программном обеспечении (T1190)

Злоумышленник может воспользоваться недостатками программного обеспечения, взаимодействующего с пользователями через открытые сетевые соединения. В качестве недостатков могут выступать бинарные, логические, архитектурные и иные недостатки, при эксплуатации которых нарушается семантика вычислений. Программным обеспечением, осуществляющим сетевое взаимодействие могут являться базы данных (SQL и др.), драйверы (SMB, SSH, HTTP и др.) и иные сетевые службы.

D. Эксплуатация уязвимостей в пользовательском программном обеспечении (T1203)

Злоумышленник может воспользоваться недостатками, содержащимися в пользовательском программном обеспечении, развернутом в облачной инфраструктуре. Недостатки могут быть вызваны ошибками кодирования, проектирования, развертывания и др. В этом случае злоумышленник может добиться выполнения произвольного кода с правами пользователя, запустившего процесс.

E. Эксплуатация уязвимостей для обхода механизмов защиты (T1211)

Злоумышленник может воспользоваться недостатками в прикладном и системном программном обеспечении или ядре операционной системы для нарушения работы механизмов защиты. В некоторых случаях недостатки могут содержаться в самих средствах защиты.

F. Нарушение межпроцессного взаимодействия (T1559)

Злоумышленник может использовать недостатки механизма межпроцессного взаимодействия (Inter-Process Communication, IPC) для выполнения вредоносного кода. Механизм IPC обычно используется процессами для обмена данными, синхронизации, избежания ситуаций взаимной блокировки и др.

G. Перехват потока управления (T1574)

Злоумышленник может выполнить вредоносный код путем перехвата выполнения операционной системой кода программы. Такой перехват может быть выполнен, например, путем подмены динамической библиотеки на вредоносную с последующей ее загрузкой или модификации переменных окружения, считываемых операционной системой во время запуска программного обеспечения, таких как «LD_PRELOAD».

IV. Модель самовосстанавливающегося вычислительного процесса облачной информационно-вычислительной системы в условиях информационно-технических воздействий

Рассмотрим два вычислительных процесса облачной информационно-вычислительной системы *p1* и *p2*, представимых уравнениями, имеющими вид [12]:

$$\sum_{i=1}^{q} \varphi_{ui} = 0, u = 1,2,...,r$$

$$\sum_{i=1}^{q} \phi_{ui} = 0, u = 1,2,...,r$$

где $\varphi_u = \prod_{j=1}^{n} x_j^{\alpha_{ul}}$ и $\phi_u = \prod_{j=1}^{n} X_j^{\alpha_{ul}}$ – однородные функции входных параметров вычислительных процессов.

Прямая теорема подобия гласит, что если два процесса однородно подобны, то справедлива система соотношений:

$$\frac{\varphi_{ui}}{\varphi_{uq}} = \frac{\phi_{ui}}{\phi_{uq}}, u = 1,2,...,r; s = 1,2,...,(q-1),$$

в которой выражения вида $\pi_{us} = \frac{\varphi_{ui}}{\varphi_{uq}}$ называются инвариантами подобия. Такие отношения численно равны для взаимно подобных вычислительных процессов. Равенство инвариантов подобия вычислительных процессов является необходимым, но не достаточным условием отнесения вычислительных процессов к подклассу взаимно подобных. Для определения достаточного условия необходимо рассмотреть обратную теорему подобия, которая гласит, что два процесса однородно подобны, если их полные уравнения возможно привести к виду, в котором инварианты подобия численно равны [6].

Обозначим *P* вычислительный процесс в облачной информационно-вычислительной системе. Пусть *T* – множество дискретных моментов времени *t*, в которые выполняется наблюдение вычислительного процесса. *X* и *Y* – множества входных и выходных параметров вычислительного процесса соответственно. Поскольку вычислительный процесс характеризуется последовательной сменой состояний, обозначим *Z* их множество, характеризующееся в каждый дискретный момент времени $t \in T$ последовательностью выполняемых в контрольной точке арифметических операций. Обозначим *F* и *Ф* – множества операторов перехода f_i и операторов выхода ϕ_i, отвечающих за смену состояний и формирование результатов вычислений соответственно. Тогда вычислительный процесс *P* облачной информационно-вычислительной системы представим в виде:

$$P = < T, X, Y, Z, F, \Phi > .$$

Для определения отношений между заданными множествами необходимо ввести отображения:

$\lambda : T \times X \to Z'$ – отображение, определяющее информационно-технические воздействия на вычислительный процесс;

$\psi : Z' \to \Pi'$ – отображение, определяющее формирование инвариантов подобия в условиях воздействий, где П’ – множество инвариантов подобия;

$\mu : \Pi' \to \Pi$ – отображение, определяющее сравнение инвариантов подобия между собой;

$\upsilon : \Pi \to E$ – отображение, определяющее сигнал о нарушении целостности вычислений, где Е – множество сигналов об ошибках;

$\xi : \Pi \to Z$ – отображение, определяющее восстановление корректных вычислений;

$\chi : Z \to Y$ – отображение, определяющее вычисление корректного результата.

Тогда вычислительный процесс облачной информационно-вычислительной системы, с учетом информационно-технических воздействий и самовосстановления в терминах отображений, возможно представить как показано на рис. 3.

Рис. 3. Диаграмма отображений самовосстанавливающегося вычислительного процесса облачной информационно-вычислительной системы в условиях информационно-технических воздействий

Формализованное представление о семантике вычислительного процесса возможно получить путем построения его формальной модели в виде графа потока управления:

$$\Gamma(B,D),$$

где $B=\{B_i\}$ – множество линейных участков программы, представленных последовательностями вычислительных операторов $B_i=(b_{i1},b_{i2},...,b_{il})$;

$D=\{B\times B\}$ – множество связей по управлению между линейными участками графа потока управления.

Каждый элементарный путь в графе (путь без циклов) является реализацией программы:

$$B^k=\left(B_1^k,B_2^k,...,B_t^k\right),$$

где $B^k\subseteq B$ и $B_i^k=\left(b_{i1}^k,b_{i2}^k,...,b_{il}^k\right),\forall i=\overline{1,p}$ – последовательность выполняемых арифметических операторов.

Исследования, проведенные в работе [6] показали, что наиболее эффективным способом контроля вычислительных процессов является проверка соотношений, опирающихся на свойства вычислений. Такие соотношения задают семантические связи между объектами и неизменны относительно различных реализаций программы.

Пусть $f_i^k\left(x_1,x_2,...,x_N\right)$ – первичное соотношение, задающее последовательность выполнения арифметических операций на линейном участке управляющего графа программы. Тогда для *k*-ой реализации программы возможно записать некоторую последовательность первичных соотношений:

$$\begin{cases} y_1=f_1^k\left(x_1,x_2,...,x_N\right),\\ y_2=f_2^k\left(x_1,x_2,...,x_N,y_1\right),\\ ...\\ y_M=f_M^k\left(x_1,x_2,...,x_N,y_1,y_2,...,y_{M-1}\right). \end{cases}$$

Выполнив суперпозицию, получим:

$$\begin{cases} y_1=z_1^k\left(x_1,x_2,...,x_N\right),\\ y_2=z_2^k\left(x_1,x_2,...,x_N\right),\\ ...\\ y_m=z_m^k\left(x_1,x_2,...,x_N\right). \end{cases} \quad (1)$$

Каждое *i*-ое соотношение представимо в виде суммы степенных одночленов:

$$y_i=\sum_{i=1}^{p_i}z_{ij}\left(x_1,x_2,...,x_N\right),$$

где $z_{ij}\left(x_1,x_2,...,x_N\right)$ – степенной одночлен.

Слагаемые суммы (1) имеют одинаковые размерности, то есть выполняется равенство:

$$\left[z_{ij}\left(x_1,x_2,...,x_N\right)\right]=\left[z_{il}\left(x_1,x_2,...,x_N\right)\right],\ j,l=\overline{1,p_i}\ .$$

Зададим некоторую функцию $\rho=X\to[X]$, которая каждому $x_j\in X$ ставит в соответствие его абстрактную размерность $\left[x_j\right]\in[X]$.

Получим выражение для абстрактных размерностей:

$$\left[z_{ij}\left(x_1,x_2,...,x_N\right)\right]=\prod_{n=1}^{N}\left[x_n\right]^{\lambda_{jn}},\ j=\overline{1,p_i}\ .$$

Воспользовавшись равенством размерностей слагаемых суммы степенных одночленов, получим систему соотношений:

$$\prod_{n=1}^{N}\left[x_n\right]^{\lambda_{jn}}=\prod_{n=1}^{N}\left[x_n\right]^{\lambda_{ln}},\ j,l=\overline{1,p_i}$$

или

$$\prod_{n=1}^{N}\left[x_n\right]^{\lambda_{jn}-\lambda_{ln}}=1,\ j,l=\overline{1,p_i} \quad (2)$$

Путем взятия логарифма в (2) получим систему однородных линейных уравнений:

$$\sum_{n=1}^{N}\left(\lambda_{jn}-\lambda_{ln}\right)\ln\left[x_n\right]=0,\ j,l=\overline{1,p_i} \quad (3)$$

При выполнении подобного (3) преобразования для $\forall B_i^k\in B^k$, получим систему уравнений k-ой реализации вычислительного процесса:

$$A^k\omega=0\,.$$

Следует учитывать, что каждая отдельная реализация $B_i^k\in B^k$ вычислительного процесса облачной информационно-вычислительной системы является частным решением для набора входных данных *X*. Так как последовательности выполняемых арифметических операций для различных реализаций вычислительного процесса могут частично совпадать ($B^k\cap B^l\neq\varnothing,\forall B^k,B^l\in B$), то математические зависимости между группами арифметических операторов при переходе между этими реализациями также должны сохраняться, что позволяет говорить об общности критериев подобия [6].

Восстановление искаженных вычислений возможно при обнаружении нарушения семантической корректности в одной из контрольных точек, расположенных на линейном участке графа потока управления, путем применения обратного преобразования матрицы инвариантов, генерации плана восстановления и выполнения точечных модификаций в памяти.

V. Заключение

В работе рассмотрена эталонная модель, а также модель угроз безопасности облачной информационно-вычислительной системы. Проведен анализ соответствующих типов информационно-технических воздействий. По результатам анализа выделены типы атак, влияющих на семантическую корректность вычислительных процессов облачных информационно-вычислительных систем. Предложена модель самовосстанавливающегося вычислительного процесса

облачной информационно-вычислительной системы в условиях информационно-технических воздействий.

Список литературы

[1] X. Yu and Y. Xue, "Smart Grids: A Cyber–Physical Systems Perspective," in *Proceedings of the IEEE*, vol. 104, no. 5, pp. 1058-1070, May 2016, doi: 10.1109/JPROC.2015.2503119.

[2] K. Cao, Y. Liu, G. Meng and Q. Sun, "An Overview on Edge Computing Research," in *IEEE Access*, vol. 8, pp. 85714-85728, 2020, doi: 10.1109/ACCESS.2020.2991734.

[3] Check Point Research Reports a 38% Increase in 2022 Global Cyberattacks (2023). URL: https://blog.checkpoint.com/2023/01/05/38-increase-in-2022-global-cyberattacks/ (дата обращения: 19.03.2023).

[4] Common Vulnerability Database: Published CVE Records. URL: https://www.cve.org/About/Metrics (дата обращения: 19.03.2023).

[5] Киберугрозы для АСУ и промышленных предприятий в 2023 году. URL: https://ics-cert.kaspersky.ru/publications/reports/2022/11/22/ics-cyberthreats-in-2023-what-to-expect/ (дата обращения: 19.03.2023).

[6] Петренко С.А. Кибериммунология: научная монография. СПб: «Издательский Дом «Афина». 2021. 240 с.

[7] Hogan M., Liu F., Sokol A., Tong J. NIST Cloud Computing Standards Roadmap. Computer Security Division Information Technology Laboratory National Institute of Standards and Technology Gaithersburg, MD 20899-8930 July 2011. URL: http://www.nist.gov/customcf/get_pdf.cfm?pub_id=909024 (дата обращения: 19.03.2023).

[8] Liu F., Tong J., Mao J., Bohn R., Messina J., Badger L., Leaf D. NIST Cloud Computing Reference Architecture. Recommendations of the National Institute of Standards and Technology. Cloud Computing Program Information Technology Laboratory National Institute of Standards and Technology Gaithersburg, MD 20899-8930 September 2011. URL: https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=909505 (дата обращения: 19.03.2023).

[9] Mishra, P., Pilli, E.S., & Joshi, R.C. (2021). Cloud Security: Attacks, Techniques, Tools, and Challenges (1st ed.). Chapman and Hall/CRC. https://doi.org/10.1201/9781003004486

[10] Ершов А.П. Теория программирования и вычислительные системы. М.: Издательство «Знание». 1972. 64 с.

[11] Globally accessible knowledge base of adversary tactics and techniques MITRE ATT&CK. URL: https://attack.mitre.org/ (дата обращения: 19.03.2023).

[12] Харжевская А.В., Ломако А.Г., Петренко С.А. Представление программ инвариантами подобия для контроля искажения вычислений // Вопросы кибербезопасности. 2017. № 2(20). С. 9-20. DOI: 10.21581/2311-3456-2017-2-9-20.