О подходе к анализу векторов атак злоумышленников на информационные системы с использованием событийно-формальной модели

И.А. Трещев, Я.Ю. Григорьев

Комсомольский-на-Амуре государственный университет

Аннотация: В работе рассматриваются модели поведения злоумышленников при реализации атак на информационные системы. Предлагается использование расширенной событийно-формальной модели размеченных систем переходов с введением дополнительной функции, определяющей время, в течение которого происходит событие, обеспечивающих оценку общего времени реализации сценария атаки с учетом возможных траекторий. Предлагается формальный язык для описания сценариев атак, нотация подобная алгебраической, в которой сложение коммутативно и ассоциативно, умножение не коммутативно и ассоциативно. Формулируются правила построения всевозможных траекторий для определенного вида атак, приводится порядок расчета их количества. Приведена схема и выражение для обобщенных атак с определенными ограничениями. Рассматривается описание атаки с помощью графа, соответствующего системе переходов, и по графу строится выражение на языке сценариев

Ключевые слова: моделирование атак, информационная безопасность, система переходов, временная задержка, формальный язык, сценарий атаки, траектория атаки, вектор атаки, кибербезопасность, анализ уязвимостей, защита информации, поведение злоумышленника.

Введение

Использование традиционных методов защиты, таких как криптография, межсетевые экраны, антивирусные программы, системы обнаружения и предотвращения вторжений, часто оказывается недостаточным, в условиях сложных, целевых и многоэтапных атак, выполняемых высококвалифицированными злоумышленниками [1].

В связи с этим, возникает необходимость в разработке и применении новых методов и средств, позволяющих описывать и прогнозировать поведение злоумышленников, обеспечивающих эффективное противодействие их поведению, формирующих оценку времени необходимого для реализации сценариев атак с учетом неоднозначности в последовательности выполнения этапов таких сценариев.

В настоящее время для моделирования сценариев реализации атак злоумышленников используются деревья и графы атак [2-4], контекстно-свободные грамматики [5], сети Петри и темпоральные модели [6], сценарии использования, вероятностные методы и сети Маркова [7], подходы, основанные на использовании искусственных нейронных сетей [8].

Предполагается, что есть некоторая отправная точка сценария атаки, например, формирование и отправка фишингового письма, рекогносцировка в сети, а в результате реализации сценария злоумышленник получит вполне определенный результат [9], например, эскалация привилегий до учетной записи администратора системы управления базами данных (СУБД), получение прав администратора домена в сети.

Анализ траекторий

Рассмотрим расширение размеченных систем переходов (Labelled Transition System − LTS) [10] введением дополнительно выделенного конечного состояния и функции времени, определяющей задержку при наступлении соответствующего события.

Временная размеченная система переходов (Timed Labelled Transition System − TLTS) определяется как кортеж:

$$TLTS = (S, L, E, Tran, i_0, f, l, T),$$

где S – множество состояний, L – множество меток, E– множество событий, которые могут происходить, $Tran \subseteq S \times E \times S$– отношение перехода между состояниями, i_0 – начальное состояние из S, f– конечное состояние из S, T: $E \to R_{\geq 0}$– тотально определенная на E инъективная функция, задающая время, как задержку при наступлении события (рациональное число, или возможно ноль), l:$E \to L$ – биективная функция, описывающая события при помощи меток.

Для записи перехода будем использовать обозначение:

$$s \overset{a,t}{\Rightarrow} s',$$

которое означает, что система, находясь в состоянии s, при возникновении события с меткой a переходит в состояние s' после задержки t условных единиц времени.

Путем будем называть такую последовательность состояний $(s_1,s_2,\ldots,s_n)$, что одновременно выполняются три условия:

$$1) s_1 = i_0, s_n = f, \forall s_i \in S, i = \overline{1..n}$$

$$2) \exists (a_1, a_2, ..., a_{n-1}) \mid \forall a_i \in E, i = \overline{1..n-1},$$

$$3) \forall j = \overline{1..n-1}, (s_j, a_j, s_{j+1}) \in Tran.$$

Другими словами, имеет место последовательность

$$i_0 \overset{a_1,t_1}{\Rightarrow} s_2 \overset{a_2,t_2}{\Rightarrow} \cdots \overset{a_{n-2},t_{n-2}}{\Rightarrow} s_{n-1} \overset{a_{n-1},t_{n-1}}{\Rightarrow} f$$

Следует отметить, что путь, соединяющий два состояния возможно не единственный, с точки зрения последовательности состояний, входящих в этот путь.

Под длиной пути будем понимать количество состояний n.

Временем реализации пути назовем величину

$$d(s_1, s_2, ..., s_n) = \sum_{i=1}^{n-1} t_i,$$

ясно что из того, что один путь длиннее или короче другого в общем случае не следует аналогичное неравенство относительно времени реализации этих путей.

Два пути назовем равными, если равно время их реализации.

Предположим, что злоумышленник не будет возвращаться в то состояние на пути, в котором он уже побывал и будем рассматривать *TLTS* удовлетворяющие следующим условиям:

1) Нет циклов начинающихся и заканчивающихся в одной и той же вершине

$$\forall n \in N, \nexists (a_1, a_2, ..., a_{n-1}), \forall a_i \in E, i = \overline{1..n-1} \mid \exists (s_1, s_2, ..., s_n), \forall s_i \in S, i = \overline{1..n},$$

$$s_1 = s_n \ \& \ \forall j = \overline{1..l}, (s_j, a_j, s_{j+1}) \in Tran$$

2) Отсутствуют петли, частный случай цикла

$$\forall s \in S, \nexists e \in E, (s, e, s) \in Tran$$

3) Для любых двух состояний они могут быть соединены одним и только одним событием

$$\forall s_1, s_2 \in S, e_1 \in E, e_2 \in E, (s_1, e_1, s_2) \in Tran, (s_1, e_2, s_2) \in Tran \Rightarrow e_1 = e_2$$

Обозначим через $Trace(TLTS)$ множество всевозможных путей, соединяющих начальное и конечное состояние, тогда под временем реализации атаки (с использованием некоторой последовательности уязвимостей) будем величину $T(TLTS)$, для которой справедливо

$$\min_{(s_1, s_2, ..., s_n) \in Trace(TLTS)} d(s_1, s_2, ..., s_n) \le T(TLTS) \le \max_{(s_1, s_2, ..., s_n) \in Trace(TLTS)} d(s_1, s_2, ..., s_n)$$

Подобно формальному языку исчисления коммуницирующих систем Милнера (Milner`s calculus of Communicating Systems – Milner`s CCS) [6] мы можем описать поведение TLTS с использованием операторов, при этом процессам поставим в соответствие состояния, а действиям события. Далее будем использовать следующие обозначения:

$(P + Q)$ – означает, что система может перейти или в состояние P или Q, при этом на множестве состояний пара $(S, +)$ образуют коммутативную полугруппу.

- $(P * Q)$ – означает, что система последовательно переходит в состояние P затем в Q. При этом отметим, что операция * некоммутативна (далее будем опускать знак умножения), и $(S, *)$ образуют полугруппу. Операция * дистрибутивна относительно +.

Визуально вектора атак будем представлять в форме графов, где его вершины, это состояния, а ребра соединяют состояния, принадлежащие отношению *Tran*.

Построение траекторий

Пусть сценарий выглядит следующим образом (такого рода сценарий назовем линейным) $V = OA_1A_2\ldots A_nZ$, ясно что в этом случае мы имеем одну траекторию реализации данного сценария. При этом ограничимся рассмотрением систем переходов без времени и без меток. Тогда графовое представление сценария см. рис. 1

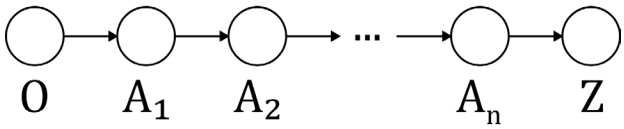

Рис. 1. Линейный сценарий

Если для некоторого i на этапе A_i у злоумышленника появляются две альтернативы (система может перейти в два состояния) A_i^1 или A_i^2, другими словами, тогда имеем две траектории $OA_1A_2A_{i-1}A_i^1A_{i+1}\ldots A_nZ$ или $OA_1A_2A_{i-1}A_i^2A_{i+1}\ldots A_nZ$, или $V = OA_1A_2A_{i-1}(A_i^1 + A_i^2)A_{i+1}\ldots A_nZ$, см. рис. 2.

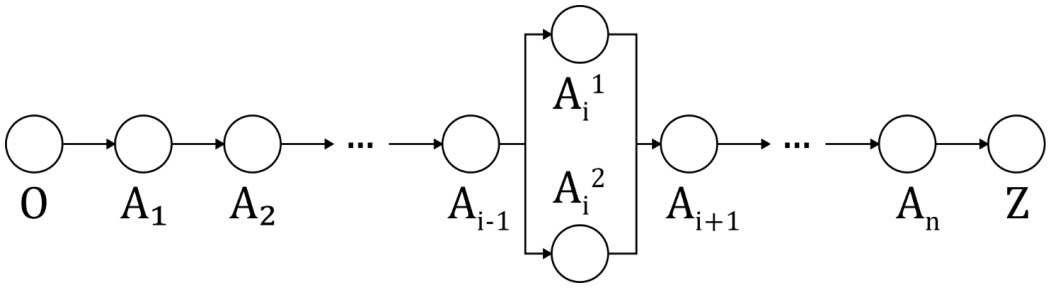

Рис. 2. Альтернативные сценарии

Ясно что если бы альтернатив было m, то и траекторий в этом случае было бы тоже m. Вектор атаки в этом случае описывался задается выражением

$$V = OA_1A_2\ldots A_{i-1}(\sum_{k=1}^{m} A_i^k)A_{i+1}\ldots A_nZ,$$

формирующим m траекторий

$$OA_1A_2A_{i-1}(A_i^1)A_{i+1}\ldots A_nZ, OA_1A_2A_{i-1}(A_i^2)A_{i+1}\ldots A_nZ,\ldots,$$

$$OA_1A_2A_{i-1}(A_i^{m-1})A_{i+1}\ldots A_nZ, OA_1A_2A_{i-1}(A_i^m)A_{i+1}\ldots A_nZ$$

Таким образом сформулируем правило – для каждого слагаемого в скобках существует отдельная траектория, включающая его.

Замечание 1: В том случае если вектор представляет из себя выражение

$$V = OA_1A_2...A_{i-1}(\sum_{k=1}^{m} A_i^k)A_{i+1}...A_{j-1}(\sum_{k=1}^{l} A_j^k)A_{j+1}...A_nZ$$

количество траекторий будет равно произведению ml. И правило формирования траекторий можно обобщить, рассматривая действие состоящие из двух этапов – выбор для траектории слагаемого из первой скобки и из второй. При этом для каждой траектории те состояния, которые наступают последовательно остаются неизменными.

Далее рассмотрим вектора, которые задаются как последовательность сменяющихся этапов двух видов – последовательный переход из одного состояния в другое и выбор из определенного числа альтернатив. Пусть сначала выполняется i_1 этапов атаки один за другим, затем происходит выбор из j_1 альтернатив, после которых вновь последовательно i_2, затем снова выбор из j_2 и так далее до некоторых i_k и j_k. Пусть множество состояний $A = \{O, A_1, A_2,\ldots, A_n, Z\}$ и $i \neq j \Rightarrow A_i \neq A_j$, при этом под числом элементов A будем понимать n. и заданы наборы индексов $I = \{i_1,i_2,..,i_k\}$ и $J=\{j_1,j_2,..,j_k\}$, удовлетворяющих условию

$$\sum_{m=1}^{k}(i_m + j_m) = n$$

и вектор атаки может быть задан в форме:

$$V = OA_1A_2...A_i \times (A_{i_1+1} + A_{i_1+2} + ... + A_{i_1+j_1})A_{i_1+j_1+1} + A_{i_1+j_1+2} \times$$

$$\times(A_{i_1+j_1+i_2+1} + A_{i_1+j_1+i_2+2} + ... + A_{i_1+j_1+i_2+j_2}) \times ... \times$$

$$\times(A_{i_1+j_1+i_2+j_2+...+i_{k-1}+j_{k-1}+i_k+1} + A_{i_1+j_1+i_2+j_2+...+i_{k-1}+j_{k-1}+i_k+2} + ... + A_{i_1+j_1+i_2+j_2+...+i_{k-1}+j_{k-1}+i_k+j_k}) \times Z$$

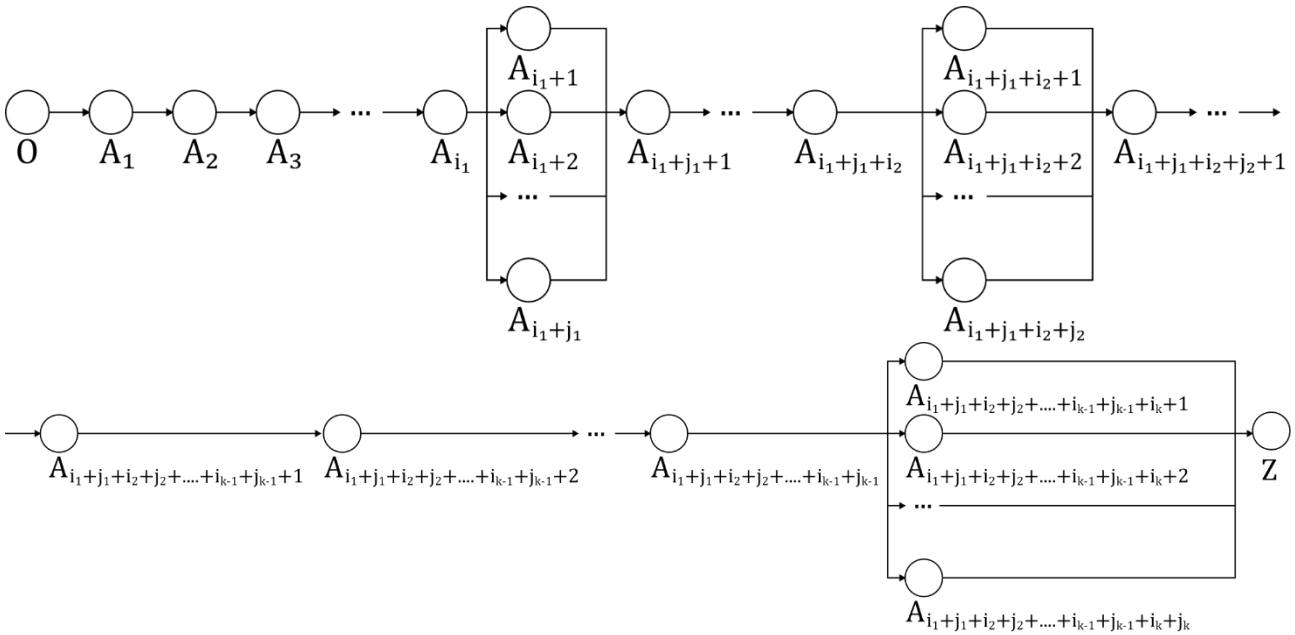

Рис. 3. Графическое представление

Исключая тривиальную линейную траекторию, представимую в виде

$$V = O(\prod_{t=1}^{i_1} A_t)Z$$

и для которой все индексы кроме i_1 равны нулю, вектор атаки может быть представлен в виде:

$$V = O(\prod_{l=1}^{k}(\prod_{t=1}^{i_l} A_{t+\sum_{m=1}^{l-1}(i_m+j_m)})(\sum_{t=1}^{j_l} A_{t+j_l+\sum_{m=1}^{l-1}(i_m+j_m)}))Z$$

Замечание 2: Отметим, что для такого вида вектора атаки справедливо число траекторий будет равно

$$\prod_{l=1}^{k} j_k,$$

что несложно показать по индукции (используя замечание 1).

При этом правило формирования траекторий может быть обобщено на данный случай в форме выполнения действия, состоящего из k этапов – ка каждом этапе выбор слагаемого из соответствующей скобки и сохранение этапов, выполняемых последовательно.

Примеры описания траекторий

Вектор атаки 1: Атака через вредоносное мобильное приложение

Пусть у нас имеется набор состояний, в которых будет пребывать система, причем обозначим начальное состояние i_0=O, конечное состояние $f = Z$.

A – разработка или модификация легитимного мобильного приложения;

B – распространение заражённого приложения среди сотрудников;

C – установка приложения на мобильные устройства сотрудников;

D – запрос расширенных разрешений (камера, микрофон, контакты, геолокация);

E – фоновый сбор данных без уведомления пользователя;

F – кража учётных данных и перехват корпоративного трафика;

G – кейлоггинг для кражи паролей и логинов;

H – перехват кодов двухфакторной аутентификации;

J – удалённое управление устройством.

Данный вектор может быть представлен в графической форме.

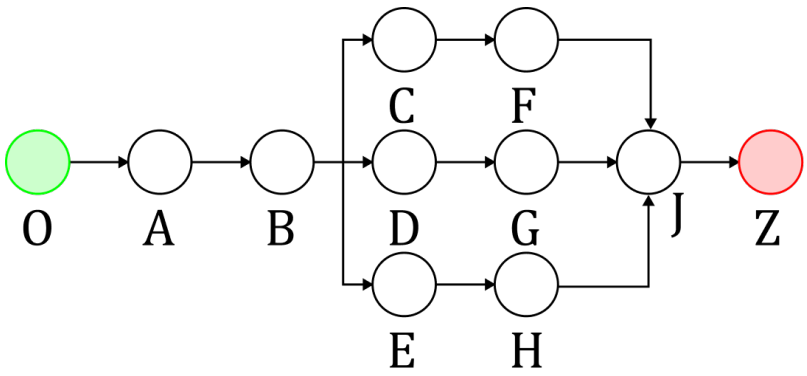

Рис. 4. – Графическое представление вектора 1

или во введенной нотации

$$V_1(O, A, B, C, F, D, G, E, H, Z) = OAB(CF + DG + EH)JZ$$

Раскрыв скобки, мы получим всевозможные траектории реализации атаки O*ABCFJZ*, O*ABDGJZ*, O*ABEHJZ*.

Вектор атаки 2: Компрометация системы видеонаблюдения и физических устройств

Аналогично первому этапу введем *O*, *Z*. Пусть атака включает следующие этапы:

A – сканирование сети на наличие уязвимых оконечных устройств интернета вещей (Internet of Things – IoT);

B – анализ настроек видеонаблюдения и поиск открытых видеопотоков;

C – эксплуатация известных уязвимостей прошивки (настройки по умолчанию, заводские настройки парольной защиты);

D – эксплуатация слабых паролей и уязвимостей и внедрение бэкдора;

E – отключение безопасности или изменение конфигураций IoT-устройств;

F – перехват и подмена видеопотоков;

G – управление IoT-устройствами для физического проникновения;

H – подготовка к использованию IoT-устройств для последующих атак;

I – проведение атак типа отказ в обслуживании (Denial of Service – DoS) на внутреннюю сеть;

J – использование IoT-устройств для ботнет сети при реализации других атак.

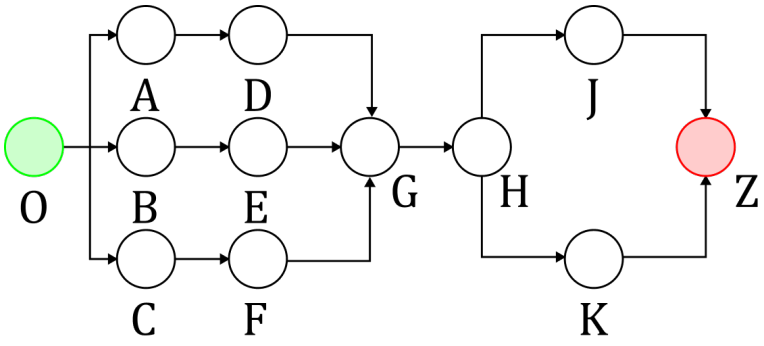

Рис. 5. – Графическое представление модели вектора 2

Во введенной нотации

$V_2=O(AD+CF+BE)GH(J+K)Z$

Раскрыв скобки, мы получим всевозможные траектории реализации атаки OADGHJZ, OADGHKZ, OCFGHJZ, OCFGHKZ, OBEGHJZ, OBEGHKZ.

Заключение

Предложенная в работе нотация позволяет моделировать действия злоумышленников при помощи расширения размеченных систем переходов, что позволяет в некоторых случаях ввести априорные оценки времени реализации векторов атаки. Подход для построения возможных траекторий реализации сценариев нарушения состояния защищенности информации может быть использован при создании систем обеспечения конфиденциальности, целостности и доступности. Использование нотации подобной описанной в работе, позволяет формализовать процесс реализации всей поверхности атаки для соответствующего вектора. Дальнейшие исследования необходимо проводить для более сложных структур траекторий с учетом противодействия злоумышленникам со стороны службы защиты информации.

Литература

1. Котенко Д.И., Котенко И.В., Саенко И.Б. Методы и средства моделирования атак в больших компьютерных сетях: состояние проблемы // Труды СПИИРАН. 2012. № 3(22). С. 5-30. EDN PCCYJH.

2. Чечулин А.А., Котенко И.В. Построение графов атак для анализа событий безопасности // Безопасность информационных технологий. 2014. Т. 21, № 3. С. 135-141. EDN RSUFZL.

3. Крюков Д.М. Графоаналитическая модель процесса ликвидации последствий компьютерных атак и реагирования на компьютерные инциденты // Инженерный вестник Дона. 2022. № 4. URL: http://www.ivdon.ru/ru/magazine/archive/n4y2022/7632

4. Будко Н.П., Васильев Н.В. Обзор графо-аналитических подходов к мониторингу информационно-телекоммуникационных сетей и их применение для выявления аномальных состояний // Системы управления, связи и безопасности. 2021. № 6. С. 53-75. DOI: 10.24412/2410-9916-2021-6-53-75.

5. Lavore D. E., Gianola A., Román M. et al. Span(Graph): a canonical feedback algebra of open transition systems // Softw Syst Model. 2023. Vol. 22. pp. 495-520. DOI: 10.1007/s10270-023-01092-7

6. Makwana M.D., Thakkar V., Das D., Kumar R. Simulating Cyber-Attack Scenarios by Discovering Petri-Nets from Large-Scale Event Logs // 2024 16th International Conference on COMmunication Systems & NETworkS (COMSNETS). 2024. pp. 49-54. DOI: 10.1109/COMSNETS59351.2024.10427052.

7. Канаев А.К., Опарин Е.В., Опарина Е.В. Моделирование действий злоумышленника при ведении сетевой разведки с использованием инфраструктуры комплексной системы синхронизации и доставки шкалы времени // Известия Петербургского университета путей сообщения. 2025. Т. 22, № 1. С. 263-273. DOI 10.20295/1815-588X-2025-1-263-273. EDN KCYOOS.

8. Метельков А.Н. Моделирование сценариев кибератак в киберполигонах // Научно-аналитический журнал "Вестник Санкт-Петербургского университета Государственной противопожарной службы МЧС России". 2023. № 2. С. 161-176. EDN WFLOEL.

9. Георгица И.В., Гончаров С.А., Мохов В.А. Мультиагентное моделирование сетевой атаки типа распределенный отказ в обслуживании (Distributed Denial of Service – DDoS) // Инженерный вестник Дона. 2013. № 3. URL: ivdon.ru/magazine/archive/n3y2013/1852.

10. Gorrieri R. Labeled Transition Systems // Process Algebras for Petri Nets. Monographs in Theoretical Computer Science. An EATCS Series. Springer, Cham, 2017. URL: doi.org/10.1007/978-3-319-55559-1_2

References

1. Kotenko D.I., Kotenko I.V., Saenko I.B. Trudy SPIIRAN. 2012, no. 3(22), pp. 5-30. EDN PCCYJH.

2. Chechulin A.A., Kotenko I.V. Bezopasnost' informatsionnykh tekhnologii. 2014, vol. 21, no. 3, pp. 135-141. EDN RSUFZL.

3. Kryukov D.M. Inzhenernyi vestnik Dona. 2022, № 4. URL: ivdon.ru/ru/magazine/archive/n4y2022/7632

4. Budko N.P., Vasil'ev N.V. Sistemy upravleniya, svyazi i bezopasnosti. 2021, no. 6, pp. 53-75. DOI: 10.24412/2410-9916-2021-6-53-75.

5. Lavore D. E., Gianola A., Román M. et al. Softw Syst Model. 2023, vol. 22, pp. 495-520. DOI: 10.1007/s10270-023-01092-7

6. Makwana M.D., Thakkar V., Das D., Kumar R. 2024 16th International Conference on COMmunication Systems & NETworkS (COMSNETS). 2024, pp. 49-54. DOI: 10.1109/COMSNETS59351.2024.10427052.

7. Kanaev A.K., Oparin E.V., Oparina E.V. Izvestiya Peterburgskogo universiteta putei soobshcheniya. 2025, vol. 22, no. 1, pp. 263-273. DOI 10.20295/1815-588X-2025-1-263-273. EDN KCYOOS.

8. Metel'kov A.N. Nauchno-analiticheskii zhurnal "Vestnik Sankt-Peterburgskogo universiteta Gosudarstvennoi protivopozharnoi sluzhby MChS Rossii". 2023, no. 2, pp. 161-176. EDN WFLOEL.

9. Georgitsa I.V., Goncharov S.A., Mokhov V.A. Inzhenernyi vestnik Dona. 2013, no. 3. URL: ivdon.ru/magazine/archive/n3y2013/1852.

10. Gorrieri R. Process Algebras for Petri Nets. Monographs in Theoretical Computer Science. An EATCS Series. Springer, Cham, 2017. URL: doi.org/10.1007/978-3-319-55559-1_2.

Дата поступления: 11.11.2025 **Дата публикации: 25.12.2025**