АЛГОРИТМ ПОСТРОЕНИЯ ДИАГРАММЫ ДОСТИЖИМОСТИ МОДЕЛИ СОСТОЯНИЯ РАБОТОСПОСОБНОСТИ ИНФОРМАЦИОННОЙ СИСТЕМЫ

Болычев М.В.[1], Мирошниченко Е.Л.[2], Пасечник Р.М.[3]

Аннотация: *в статье рассматривается модель состояния работоспособности информационной системы, применимая для моделирования динамического изменения состояния при проведении компьютерной атаки, а также реагировании при обнаружении и предотвращении деструктивного воздействия. Разработанная модель построена на основе расширенной временной раскрашенной функциональной нечеткой сети Петри. По своему функционалу модель разделена на две сети: статическую и динамическую. Статья включает в себя полное описание частного процесса изменения разметки сети с учетом особенностей используемой сети Петри. Также в статье предлагается алгоритм построения диаграммы достижимости временных разметок сети. Алгоритм основывается на методе поиска в ширину с дополнительными блоками, учитывающими деструктивное воздействие и реагирование на изменение состояния информационной системы. Полученные модель состояния работоспособности информационной системы и диаграмма достижимости временных разметок отображают возможные промежуточные и конечные состояния информационной системы при учете деструктивного воздействия и реагирования на него. Исследования могут быть применены при совершенствовании методов обнаружения и противодействия современным компьютерным атакам, а также при формировании системы поддержки принятия решения.*

Ключевые слова: *защита информации, информационная безопасность, компьютерная атака, сервисы, сети Петри, динамическое моделирование, диаграмма достижимости, деструктивное воздействие, реагирование, методы противодействия, обнаружение.*

DOI: 10.21681/2311-3456-2019-6-79-91

Введение

Анализ периодических отчетов об актуальных угрозах и компьютерных атаках (КА) ведущих организаций, таких как Лаборатория Касперского, Positive Technologies, Symantec Corporation и др., показал рост количества новых способов реализации сложных КА, в том числе использующих «zero day» уязвимости. В связи с чем перед организациями и их центрами мониторинга стоит проблема обнаружения данных КА на их ранних стадиях.

Современная методология противодействия КА основывается на сборе событий информационной безопасности (ИБ) [1-5], выявлении инцидентов и их закрытии с последующим анализом произошедшего и корректировкой системы защиты [6-8]. Плюсами данного подхода является возможность использования готовых «шаблонов» – сигнатур для противодействия известным методам реализации КА. Минусами работы с событиями являются уязвимости к новым видам КА, а также расположение средств защиты, в большинстве случаев они контролируют только периметр защищаемого объекта [9-12].

Методы обнаружения и противодействия КА, основанные на анализе текущего и планируемого состояния защищаемого объекта [13], позволяют оперативно реагировать на изменения в состояниях подсистем объекта, в особенности критических, до момента срабатывания систем мониторинга событий, что позволит локализовать и пресечь новые виды КА.

При оперировании как событиями, так и состояниями информационной безопасности защищаемого объекта рано или поздно встанет вопрос принятия решения в той или иной ситуации, требующий высокой степени автоматизации процесса. Для этого представим модель состояния работоспособности информационной системы и алгоритм построения диаграммы достижимости, которая применима в процессе поддержки принятия решения.

Модель представления состояния работоспособности ИС

Для представления модели состояния работоспособности информационной системы воспользуемся сетью Петри, представляющей собой граф с дополнительными правилами, позволяющими учесть все процессы функционирования и описывать динамические модели [14-21].

1 Болычев Максим Владимирович, сотрудник Восьмого управления Генерального штаба Вооруженных Сил Российской Федерации, г. Москва, Россия. E-mail: stq-mak@yandex.ru

2 Мирошниченко Евгений Леонидович, начальник Научно-исследовательского центра, Краснодарское высшее военное училище, г. Краснодар, Россия. E-mail: mirash_mel@mail.ru

3 Пасечник Родион Маратович, научный сотрудник, Краснодарское высшее военное училище, г. Краснодар, Россия. E-mail: rmpasechnik@mail.ru

Модель представим в виде двух расширенных временных раскрашенных функциональных нечетких сетей Петри, условно названных статической и динамической, и характеризуемых набором:

$$N = \left(\tilde{P}, \tilde{D}, I, O, \tilde{\tilde{\mu}}_0, \tilde{C}, \tilde{V}, \Lambda^*, \Lambda^{**}, \vec{S}, \Delta, R\right), \quad (1)$$

где: $\tilde{P} = \{\tilde{p}_i\}$ – непустое конечное множество нечетких позиций, $\tilde{D} = \{\tilde{d}_j\}$ – непустое конечное множество нечетких переходов, $I: T \times P \to \mathbb{N}_0$ – входная функция переходов, $O: T \times P \to \mathbb{N}_0$ – выходная функция переходов, $\tilde{\tilde{\mu}}_0$ – вектор нечеткой начальной разметки, $\tilde{C}$ – функция цвета маркера, определенная для каждого из маркеров, $\tilde{V} = \{\tilde{v}_a\}$ – множество условий срабатывания переходов в зависимости от цвета маркера и попадания маркеров из переходов в позиции, $\Lambda^* = \{t_j^*\}$ – множество времени минимальной задержки для переходов, $\Lambda^{**} = \{t_j^{**}\}$ – множество времени максимальной задержки для переходов, $\vec{S}$ – вектор времен срабатывания разрешенных переходов, $\Delta = \{\vec{\delta}_n\}$ – множество векторов деструктивного воздействия, $R = \{\vec{r}_m\}$ – множество векторов реагирования.

Статическая сеть

Статическая сеть представляет собой набор $N_s = \left(\tilde{P}, \tilde{D}, I, O, \tilde{\tilde{\mu}}_0, \tilde{C}, \tilde{V}, \Lambda^*, \Lambda^{**}, \vec{S}, \Delta, R\right)$,

где:

$\tilde{P} = \{\tilde{p}_0, \tilde{p}_1, \tilde{p}_2, \dots, \tilde{p}_{12}\}$, $\tilde{D} = \{\tilde{d}_0, \tilde{d}_1, \tilde{d}_2, \dots, \tilde{d}_{15}\}$;

$I(\tilde{p}_0) = \{\tilde{d}_4\}$, $O(\tilde{p}_0) = \{\tilde{d}_0\}$,
$I(\tilde{p}_1) = \{\tilde{d}_0, \tilde{d}_5\}$, $O(\tilde{p}_1) = \{\tilde{d}_1, \tilde{d}_4\}$,
$I(\tilde{p}_2) = \{\tilde{d}_1, \tilde{d}_6\}$, $O(\tilde{p}_2) = \{\tilde{d}_2, \tilde{d}_5\}$,
$I(\tilde{p}_3) = \{\tilde{d}_2, \tilde{d}_7\}$, $O(\tilde{p}_3) = \{\tilde{d}_3, \tilde{d}_6\}$,
$I(\tilde{p}_4) = \{\tilde{d}_3\}$, $O(\tilde{p}_4) = \{\tilde{d}_7\}$,
$I(\tilde{p}_5) = \{\tilde{d}_8\}$, $O(\tilde{p}_5) = \{\tilde{d}_0\}$,
$I(\tilde{p}_6) = \{\tilde{d}_9\}$, $O(\tilde{p}_6) = \{\tilde{d}_1\}$,
$I(\tilde{p}_7) = \{\tilde{d}_{10}\}$, $O(\tilde{p}_7) = \{\tilde{d}_2\}$,
$I(\tilde{p}_8) = \{\tilde{d}_{11}\}$, $O(\tilde{p}_8) = \{\tilde{d}_3\}$;
$I(\tilde{p}_9) = \{\tilde{d}_{12}\}$, $O(\tilde{p}_9) = \{\tilde{d}_4\}$,
$I(\tilde{p}_{10}) = \{\tilde{d}_{13}\}$, $O(\tilde{p}_{10}) = \{\tilde{d}_5\}$,
$I(\tilde{p}_{11}) = \{\tilde{d}_{14}\}$, $O(\tilde{p}_{11}) = \{\tilde{d}_6\}$,
$I(\tilde{p}_{12}) = \{\tilde{d}_{15}\}$; $O(\tilde{p}_{12}) = \{\tilde{d}_7\}$;

$\tilde{\tilde{\mu}}_0 = (\tilde{\mu}_{01}, \tilde{\mu}_{02}, \dots, \tilde{\mu}_{0r})$, $\tilde{\mu}_{0i} \in \mathbb{N}_0; \forall \tilde{p}_i \in \tilde{P}$;

$\tilde{C} = \{\tilde{c}_0, \tilde{c}_1, \tilde{c}_2, \dots, \tilde{c}_k\}, k \in \mathbb{R}$;

$\tilde{V} = \{\tilde{v}_0, \tilde{v}_1, \tilde{v}_2, \dots, \tilde{v}_a\}, a \in \mathbb{R}$;

$\Lambda^* = \{t_0^*, t_1^*, t_2^*, \dots, t_{12}^*\}$;

$\Lambda^{**} = \{t_0^{**}, t_1^{**}, t_2^{**}, \dots, t_{12}^{**}\}$;

$\vec{S} = (s_0, s_1, s_2, \dots, s_{15})$;

$\Delta = \{\vec{\delta}_0, \vec{\delta}_1, \vec{\delta}_2, \dots, \vec{\delta}_n\}, n \in \mathbb{R}, \vec{\delta} = (\vec{\mu}_\delta, P_\delta)$;

$R = \{\vec{r}_0, \vec{r}_1, \vec{r}_2, \dots, \vec{r}_m\}, m \in \mathbb{R}, \vec{r} = (\vec{\mu}_{rc}, \vec{\mu}_{re})$.

В статической сети позиции отображают текущее состояние сервиса (рис. 1). Переход сервиса из одного состояния в другое (перемещение маркера между позициями) возможен только при внешнем или внутреннем воздействии, осуществляемом путем срабатывания переходов $\{\tilde{d}_8, \tilde{d}_9, \tilde{d}_{10}, \tilde{d}_{11}\}$ или $\{\tilde{d}_{12}, \tilde{d}_{13}, \tilde{d}_{14}, \tilde{d}_{15}\}$, задающегося векторами $\vec{\delta}_n$ и $\vec{r}_m$. Выбор вектора деструктивного воздействия $\vec{\delta}_n$ происходит стохастически. Выбор вектора реагирования $\vec{r}_m$ зависит от текущего расположения маркеров в позициях $\{\tilde{p}_0, \tilde{p}_1, \tilde{p}_2, \tilde{p}_3, \tilde{p}_4\}$.

Для статической сети модели состояния системы принята следующая интерпретация позиций и переходов:

$\tilde{p}_0$ – исправное состояние сервисов системы, количество которых соответствует наличию маркеров в данной позиции;

$\tilde{p}_1$ – работоспособное состояние сервисов системы, количество которых соответствует наличию маркеров в данной позиции;

$\tilde{p}_2$ – функционирующее работоспособное состояние сервисов системы, количество которых соответствует наличию маркеров в данной позиции;

$\tilde{p}_3$ – нефункционирующее (подлежащее корреляции) состояние сервисов системы, количество которых соответствует наличию маркеров в данной позиции;

$\tilde{p}_4$ – нефункционирующее (не подлежащее корреляции) состояние сервисов системы, количество которых соответствует наличию маркеров в данной позиции;

$\tilde{p}_5$ – деструктивное воздействие, наличие маркера в котором позволяет перевести маркер соответствующего из состояния $\tilde{p}_0$ в состояние $\tilde{p}_1$;

$\tilde{p}_6$ – деструктивное воздействие, наличие маркера в котором позволяет перевести маркер соответствующего сервиса из состояния $\tilde{p}_1$ в состояние $\tilde{p}_2$;

$\tilde{p}_7$ – деструктивное воздействие, наличие маркера в котором позволяет перевести маркер соответствующего сервиса из состояния $\tilde{p}_2$ в состояние $\tilde{p}_3$;

$\tilde{p}_8$ – деструктивное воздействие, наличие маркера в котором позволяет перевести маркер соответствующего сервиса из состояния $\tilde{p}_3$ в состояние $\tilde{p}_4$;

$\tilde{p}_9$ – реагирование, наличие маркера в котором позволяет вернуть маркер соответствующего сервиса из состояния $\tilde{p}_1$ в состояние $\tilde{p}_0$;

$\tilde{p}_{10}$ – реагирование, наличие маркера в котором позволяет вернуть маркер соответствующего сервиса из состояния $\tilde{p}_1$ в состояние $\tilde{p}_2$;

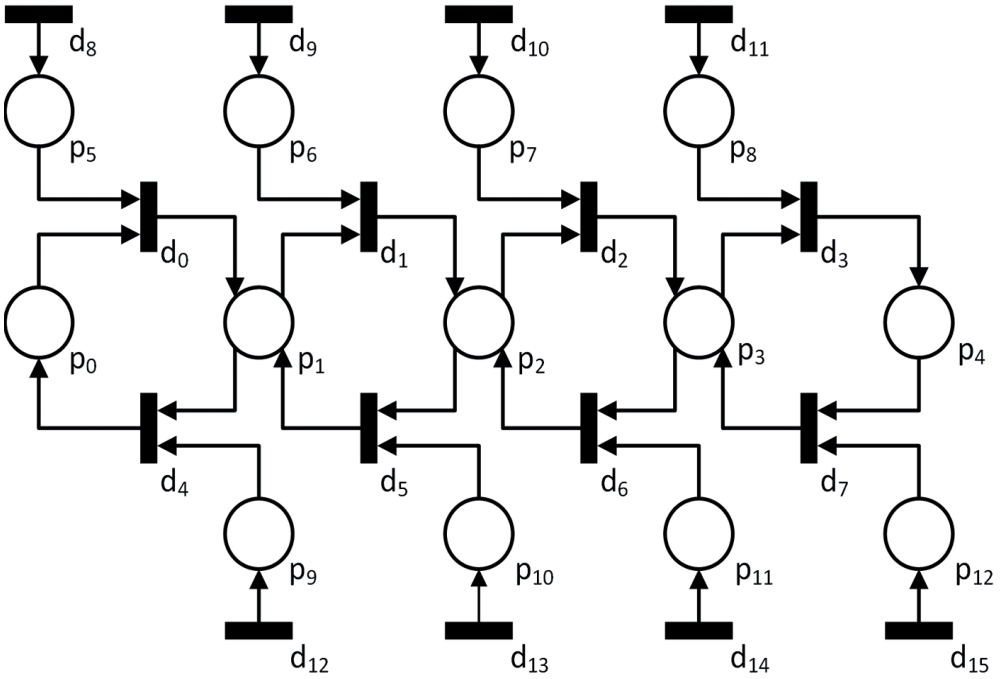

Рис. 1. Графическое изображение статической сети

$\tilde{p}_{11}$ – реагирование, наличие маркера в котором позволяет вернуть маркер соответствующего сервиса из состояния $\tilde{p}_2$ в состояние $\tilde{p}_3$;

$\tilde{p}_{12}$ – реагирование, наличие маркера в котором позволяет вернуть маркер соответствующего сервиса из состояния $\tilde{p}_3$ в состояние $\tilde{p}_4$;

$\tilde{d}_0$ – влияние деструктивного воздействия (состояние $\tilde{p}_5$) на сервис (состояние $\tilde{p}_0$), цвет маркера $\tilde{c}_c$ которого соответствует цвету маркера деструктивного воздействия $\tilde{c}_м$ $(\tilde{c}_c = \tilde{c}_м)$;

$\tilde{d}_1$ – влияние деструктивного воздействия (состояние $\tilde{p}_6$) на сервис (состояние $\tilde{p}_1$), цвет маркера $\tilde{c}_c$ которого соответствует цвету маркера деструктивного воздействия $\tilde{c}_м$ $(\tilde{c}_c = \tilde{c}_м)$;

$\tilde{d}_2$ – влияние деструктивного воздействия (состояние $\tilde{p}_7$) на сервис (состояние $\tilde{p}_2$), цвет маркера $\tilde{c}_c$ которого соответствует цвету маркера деструктивного воздействия $\tilde{c}_м$ $(\tilde{c}_c = \tilde{c}_м)$;

$\tilde{d}_3$ – влияние деструктивного воздействия (состояние $\tilde{p}_8$) на сервис (состояние $\tilde{p}_3$), цвет маркера $\tilde{c}_c$ которого соответствует цвету маркера деструктивного воздействия $\tilde{c}_м$ $(\tilde{c}_c = \tilde{c}_м)$;

$\tilde{d}_4$ – реагирование (состояние $\tilde{p}_9$) на изменение состояния сервиса (состояние $\tilde{p}_1$), цвет маркера $\tilde{c}_c$ которого соответствует цвету маркера реагирования $\tilde{c}_р$ $(\tilde{c}_c = \tilde{c}_р)$;

$\tilde{d}_5$ – реагирование (состояние $\tilde{p}_{10}$) на изменение состояния сервиса (состояние $\tilde{p}_2$), цвет маркера $\tilde{c}_c$ которого соответствует цвету маркера реагирования $\tilde{c}_р$ $(\tilde{c}_c = \tilde{c}_р)$;

$\tilde{d}_6$ – реагирование (состояние $\tilde{p}_{11}$) на изменение состояния сервиса (состояние $\tilde{p}_3$), цвет маркера $\tilde{c}_c$ которого соответствует цвету маркера реагирования $\tilde{c}_р$ $(\tilde{c}_c = \tilde{c}_р)$;

$\tilde{d}_7$ – реагирование (состояние $\tilde{p}_{12}$) на изменение состояния сервиса (состояние $\tilde{p}_4$), цвет маркера $\tilde{c}_c$ которого соответствует цвету маркера реагирования $\tilde{c}_р$ $(\tilde{c}_c = \tilde{c}_р)$;

$\tilde{d}_8$, $\tilde{d}_9$, $\tilde{d}_{10}$, $\tilde{d}_{11}$ – формирование сценария деструктивного воздействия на сервисы системы;

$\tilde{d}_{12}$, $\tilde{d}_{13}$, $\tilde{d}_{14}$, $\tilde{d}_{15}$ – формирование сценария реагирования на изменения состояния сервисов системы.

Динамическая сеть

Динамическая сеть представляет собой набор

$$N_d = \left(\tilde{P}, \tilde{D}, I, O, \tilde{\tilde{\mu}}_0, \tilde{C}, \tilde{V}, \Lambda^*, \Lambda^{**}, \vec{S}, \Delta, R\right),$$

где:

$$\tilde{P} = \{\tilde{p}_0, \tilde{p}_1, \tilde{p}_2, \dots, \tilde{p}_8\};$$

$$\tilde{D} = \{\tilde{d}_0, \tilde{d}_1, \tilde{d}_2, \dots, \tilde{d}_{11}\};$$

$$I(\tilde{p}_0) = \{\tilde{d}_4\}, \quad O(\tilde{p}_0) = \{\tilde{d}_0\},$$
$$I(\tilde{p}_1) = \{\tilde{d}_0, \tilde{d}_5\}, \quad O(\tilde{p}_1) = \{\tilde{d}_1, \tilde{d}_4\},$$
$$I(\tilde{p}_2) = \{\tilde{d}_1, \tilde{d}_6\}, \quad O(\tilde{p}_2) = \{\tilde{d}_2, \tilde{d}_5\},$$
$$I(\tilde{p}_3) = \{\tilde{d}_2, \tilde{d}_7\}, \quad O(\tilde{p}_3) = \{\tilde{d}_3, \tilde{d}_6\},$$
$$I(\tilde{p}_4) = \{\tilde{d}_3\}, \quad O(\tilde{p}_4) = \{\tilde{d}_7\},$$
$$I(\tilde{p}_5) = \{\tilde{d}_8\}, \quad O(\tilde{p}_5) = \{\tilde{d}_0\},$$
$$I(\tilde{p}_6) = \{\tilde{d}_9\}, \quad O(\tilde{p}_6) = \{\tilde{d}_1\},$$
$$I(\tilde{p}_7) = \{\tilde{d}_{10}\}, \quad O(\tilde{p}_7) = \{\tilde{d}_2\},$$
$$I(\tilde{p}_8) = \{\tilde{d}_{11}\}, \quad O(\tilde{p}_8) = \{\tilde{d}_3\};$$

$$\tilde{\tilde{\mu}}_0 = (\tilde{\mu}_{01},\ \tilde{\mu}_{02}, \dots, \tilde{\mu}_{0r}), \tilde{\mu}_{0i} \in \mathbb{N}_0; \forall \tilde{p}_i \in \tilde{P};$$
$$\tilde{C} = \{\tilde{c}_0, \tilde{c}_1, \tilde{c}_2, \dots, \tilde{c}_k\}, k \in \mathbb{R};$$
$$\tilde{V} = \{\tilde{v}_0, \tilde{v}_1, \tilde{v}_2, \dots, \tilde{v}_a\}, a \in \mathbb{R};$$
$$\Lambda^* = \{t_0^*, t_1^*, t_2^*, \dots, t_8^*\};$$
$$\Lambda^{**} = \{t_0^{**}, t_1^{**}, t_2^{**}, \dots, t_8^{**}\};$$
$$\vec{S} = (s_0, s_1, s_2, \dots, s_{11});$$
$$\Delta = \{\vec{\delta}_0, \vec{\delta}_1, \vec{\delta}_2, \dots, \vec{\delta}_n\}, n \in \mathbb{R};$$
$$R = \{\vec{r}_0, \vec{r}_1, \vec{r}_2, \dots, \vec{r}_m\}, m \in \mathbb{R}.$$

В динамической сети позиции отображают текущее состояние временного простоя сервиса (Рис. 2). Переход маркеров между позициями происходит по времени или воздействию, осуществляемому путем срабатывания переходов $\{d_8, d_9, d_{10}, d_{11}\}$, задающегося вектором $\vec{r}_m$.

Для динамической сети модели состояния системы принята следующая интерпретация позиций и переходов:

$\tilde{p}_0$ – состояние простоя сервиса (время простоя от 0 до $t_{\text{пр0}} = t_0^* = t_0^{**}$);

$\tilde{p}_1$ – состояние простоя сервиса (время простоя от $t_{\text{пр0}}$ до $t_{\text{пр1}} = t_1^* = t_1^{**}$);

$\tilde{p}_2$ – состояние простоя сервиса (время простоя от $t_{\text{пр1}}$ до $t_{\text{пр2}} = t_2^* = t_2^{**}$);

$\tilde{p}_3$ – состояние простоя сервиса (время простоя от $t_{\text{пр2}}$ до $t_{\text{пр3}} = t_3^* = t_3^{**}$);

$\tilde{p}_4$ – состояние простоя сервиса (время простоя от $t_{\text{пр3}}$ до $t_{\text{пр4}} = t_4^* = t_4^{**}$);

$\tilde{p}_5$ – деструктивное воздействие, наличие маркера в котором позволяет перевести маркер соответствующего из состояния $\tilde{p}_0$ в состояние $\tilde{p}_1$;

$\tilde{p}_6$ – деструктивное воздействие, наличие маркера в котором позволяет перевести маркер соответствующего сервиса из состояния $\tilde{p}_1$ в состояние $\tilde{p}_2$;

$\tilde{p}_7$ – деструктивное воздействие, наличие маркера в котором позволяет перевести маркер соответствующего сервиса из состояния $\tilde{p}_2$ в состояние $\tilde{p}_3$;

$\tilde{p}_8$ – деструктивное воздействие, наличие маркера в котором позволяет перевести маркер соответствующего сервиса из состояния $\tilde{p}_3$ в состояние $\tilde{p}_4$;

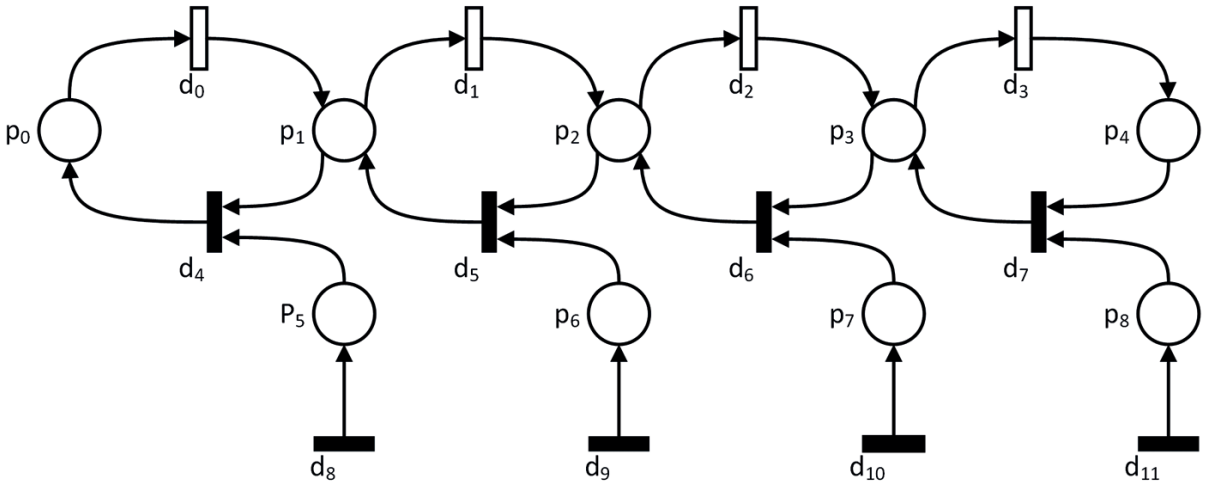

Рис. 2. Схема динамической сети

$\tilde{d}_0$, $\tilde{d}_1$, $\tilde{d}_2$, $\tilde{d}_3$ – переход маркера сервиса из одного состояния простоя в следующее;

$\tilde{d}_4$ – реагирование (состояние $\tilde{p}_5$) на изменение состояния сервиса (состояние $\tilde{p}_1$), цвет маркера $\tilde{c}_c$ которого соответствует цвету маркера реагирования $\tilde{c}_p$ ($\tilde{c}_c = \tilde{c}_p$);

$\tilde{d}_5$ – реагирование (состояние $\tilde{p}_6$) на изменение состояния сервиса (состояние $\tilde{p}_2$), цвет маркера $\tilde{c}_c$ которого соответствует цвету маркера реагирования $\tilde{c}_p$ ($\tilde{c}_c = \tilde{c}_p$);

$\tilde{d}_6$ – реагирование (состояние $\tilde{p}_7$) на изменение состояния сервиса (состояние $\tilde{p}_3$), цвет маркера $\tilde{c}_c$ которого соответствует цвету маркера реагирования $\tilde{c}_p$ ($\tilde{c}_c = \tilde{c}_p$);

$\tilde{d}_7$ – реагирование (состояние $\tilde{p}_8$) на изменение состояния сервиса (состояние $\tilde{p}_4$), цвет маркера $\tilde{c}_c$ которого соответствует цвету маркера реагирования $\tilde{c}_p$ ($\tilde{c}_c = \tilde{c}_p$);

$\tilde{d}_8$, $\tilde{d}_9$, $\tilde{d}_{10}$, $\tilde{d}_{11}$ – формирование сценария реагирования на изменения состояния сервисов системы.

Правила изменения разметки сети

Динамика изменения начальной и последующих разметок рассматриваемой сети N, описывающая процесс ее функционирования после момента запуска, подчиняется, как и любая сеть Петри, следующим правилам:

- правило определения текущей разметки;
- правило доступности маркеров в позиции;
- правило разрешения переходов;
- правило срабатывания переходов;
- правило начальной разметки;
- правило аддитивности.

Рассмотрим данные правила применительно к предлагаемой сети N и особенностям.

Текущая временная разметка сети N определяется двумя векторами:

$\vec{\mu}_д = (\mu_{д0}, \mu_{д1}, \dots, \mu_{дk})$ – количество доступных маркеров в позиции $\tilde{p}_i \in \tilde{P}$;

$\vec{\mu}_н = (\mu_{н0}, \mu_{н1}, \dots, \mu_{нk})$ – количество недоступных маркеров в позиции $\tilde{p}_i \in \tilde{P}$.

Исходя из этого вектор текущей разметки представим в виде:

$$\vec{\mu} = \langle \vec{\mu}_д, \vec{\mu}_н \rangle \quad (2)$$

Маркер, появившейся в позиции $\tilde{p}_i \in \tilde{P}$ в некоторый момент времени τ_i, будет недоступен в интервале времени с τ_i, по $\tau_i + t_i$,, где $t_i^* < t_i < t_i^{**}$,, начиная с момента его появления в позиции $\tilde{p}_i$ ($t_i^* \in \Lambda^*$ – минимальное время задержки маркера в позиции, $t_i^{**} \in \Lambda^{**}$ – максимальное время задержки маркера в позиции). В момент времени $\tau_i + t_i$ маркер становится доступным.

Переход $\tilde{d}_i \in \tilde{D}$ является разрешенным при текущей разметке $\vec{\mu} = \langle \vec{\mu}_д, \vec{\mu}_н \rangle$, когда во всех его входных позициях имеется такое количество доступных маркеров цвета c_j, которое больше или равно количеству дуг, соединяющих данные позиции с рассматриваемыми переходами:

$$\vec{\mu}_д \geq \left(I(\tilde{d}_i, \tilde{p}_1), I(\tilde{d}_i, \tilde{p}_2), \dots, I(\tilde{d}_i, \tilde{p}_{12}) \right), \quad (3)$$

Срабатывание перехода $\tilde{d}_i \in \tilde{D}$ при разметке $\vec{\mu} = \langle \vec{\mu}_д, \vec{\mu}_н \rangle$ в момент времени τ_i, начинается при его разрешении, с продолжительностью s_i, до момента времени $\tau_i + s_i$ – завершения срабатывания.

Срабатывания перехода приводит к новой разметке $\vec{\mu}' = \langle \vec{\mu}'_д, \vec{\mu}'_н \rangle$, компоненты которой определяются следующими формулами:

$$\mu'_{дj} = \mu_{дj} - I(\tilde{d}_i, \tilde{p}_i); \mu'_{нj} = \mu_{нj} (\forall p_j \in P) \quad (4)$$

В момент времени τ_i, из всех входных позиций удаляется такое количество доступных маркеров цвета c_j, сколько дуг ведет из данной позиции в рассматриваемый переход. Переход $\tilde{d}_i \in \tilde{D}$ в момент времени τ_i, остается активным в течение времени s_i.

Условия изменения разметки μ':

а. В момент времени $\tau_k \in [\tau_i, \tau_i + s_i)$ происходит начало срабатывания некоторого другого разрешенно-

го перехода $\tilde{d}_k \in \tilde{D}(\tilde{d}_k \neq \tilde{d}_i)$ сети N. Возникает новая разметка, компоненты которой определяются по формуле 4.

б. В момент времени $\tau_k \in [\tau_i; \tau_i + s_i)$ происходит окончание срабатывания некоторого другого активного перехода $\tilde{d}_l \in \tilde{D}(\tilde{d}_l \neq \tilde{d}_i)$ сети N.

в. В момент времени $\tau_k \in [\tau_i; \tau_i + s_i)$ истекает временная задержка недоступных маркеров в некоторой позиции $\tilde{p}_j \in \tilde{P}$. Новая разметка $\vec{\mu}'_l$ отличается от $\vec{\mu}'$ только двумя компонентами j и $j+n$: $\mu'_{\text{д}j} = \mu_{\text{д}j} + b_j; \mu'_{\text{н}j} = \mu_{\text{н}j} - b_j$, где b_j – количество недоступных маркеров в позиции p_j, для которых в момент времени τ_k истекла временная задержка в недоступном состоянии в данной позиции.

На момент завершения срабатывания перехода $\tilde{d}_i \in \tilde{D}$ разметка $\vec{\mu}'$ изменяется на $\vec{\mu}'' = \langle \vec{\mu}''_{\text{д}}, \vec{\mu}''_{\text{н}} \rangle$, компоненты которой определяются следующими формулами:

$$\begin{aligned} &\mu''_{\text{д}j} = \mu'_{\text{д}j} + O(\tilde{d}_i, \tilde{p}_j), \mu''_{\text{н}j} = \\ &= \mu'_{\text{н}j}(\forall \tilde{p}_j \in \tilde{P}, t_j = 0); \\ &\mu''_{\text{н}j} = \mu'_{\text{н}j} + O(\tilde{d}_i, \tilde{p}_j), \mu''_{\text{д}j} = \\ &= \mu'_{\text{д}j}(\forall \tilde{p}_j \in \tilde{P}, t_j \neq 0, t_j^* < t_j < t_j^{**}). \end{aligned} \quad (5)$$

В связи с тем, что динамика изменения разметки сети N зависит от времени, то за исходное состояния сети N в момент ее запуска τ_0 принимается разметка $\vec{\mu}'_0 = \langle \vec{\mu}'_{0\text{д}}, \vec{\mu}'_{0\text{н}} \rangle$, компоненты которой определяются следующими формулами:

$$\begin{aligned} &\vec{\mu}'_{0\text{д}j} = \mu_{0j}, \mu'_{0\text{н}j} = \\ &= 0\ (\forall \tilde{p}_j \in \tilde{P}, t_j = 0); \\ &\vec{\mu}'_{0\text{д}j} = 0,\ \mu'_{0\text{н}j} = \\ &= \mu_{0j}(\forall \tilde{p}_j \in \tilde{P}, t_j \neq 0, t_j^* < t_j < t_j^{**}). \end{aligned} \quad (6)$$

В результате чего получаем, что в момент τ_0 на маркеры начальной разметки $\vec{\mu}_0$ начинают действовать временные задержки, определяемые значением $t, t^* < t < t^{**}$.

В каждый момент времени τ изменение разметки $\vec{\mu} = \langle \vec{\mu}_{\text{д}}, \vec{\mu}_{\text{н}} \rangle$ происходит при следующих независимых условиях:

а. В момент τ истекают временные задержки для b_j недоступных маркеров в одной или нескольких позициях $\tilde{p}_j \in \tilde{P}$ сети N, что приведет к увеличению соответствующих компонент вектора $\vec{\mu}_{\text{д}}$ на значение b_j и одновременному уменьшению соответствующих компонент вектор $\vec{\mu}_{\text{н}}$ на значение b_j. Разметка $\vec{\mu}'$ в момент времени τ аддитивно определяется как общий результат окончания временных задержек для всех $\tilde{p}_j \in \tilde{P}$ сети N.

б. В момент τ истекают времена срабатывания активных переходов $\tilde{d}_i \in \tilde{D}$. Разметка $\vec{\mu}'$ в момент времени τ аддитивно определяется как общий результат окончания активных переходов сети N согласно формулам 5.

в. В момент τ становятся разрешенными неконфликтующие между собой неактивные переходы $\tilde{d}_i \in \tilde{D}$, что приводит к изменению разметки согласно формуле 4. Разметка $\vec{\mu}'$ в момент времени τ аддитивно определяется как общий результат начала срабатывания всех переходов $\{\tilde{d}_{i0}, \tilde{d}_{i1}, \dots, \tilde{d}_{ik}\}$ сети N согласно формуле 4.

В случае, когда одновременно выполняются два или три представленных выше условия, разметка $\vec{\mu}'$ в момент времени τ последовательно определяется как общий результат первоначального аддитивного окончания срабатывания соответствующих активных переходов сети N и последующего срабатывания всех ставших разрешенными в момент времени τ попарно неконфликтных переходов.

Входными данными для сети являются:

- начальная разметка, представленная в виде раскрашенных маркеров, для которых каждый отдельный цвет обозначает сервис или группу сервисов, расположенных в своих позициях, обозначающих их текущее состояние;
- время минимальной и максимальной задержек, а также время срабатывания каждого перехода;
- вектор деструктивного воздействия, являющегося последовательностью маркеров, цвет которых указывает сервис, на который будет проводится воздействие (может задаваться как стохастически, так и определенной последовательностью)
- вектор реагирования – последовательность маркеров, цвет которых переводит состояние сервиса в более высокое. Задается только заранее определенной последовательностью.

На выходе рассматриваемой модели получаем диаграмму достижимости временных разметок, узлами которой являются возможные состояния сервисов при том или ином деструктивном воздействии и реагировании, с учетом времени их достижения.

Алгоритм построения диаграммы достижимости

Установления и анализ свойств сети N возможны после построения и анализа множества достижимых

временных разметок и соответствующего процесса их изменения. Учитывая, что процесс изменения временных разметок может быть недетерминированным, в виду наличия в отдельные моменты времени конфликтных неустойчивых переходов, то наиболее общим представлением временных разметок является диаграмма достижимости временных разметок.

Для всех сетей Петри существует два основных способа построения диаграмм достижимости, основанных на исчерпывающем поиске в глубину и исчерпывающем поиске в ширину.

Ввиду того, что количество маркеров в сети постоянно изменяется из-за влияния векторов деструктивного воздействия и реагирования $\vec{\delta}_n$ и $\vec{r}_m$, для построения диаграммы достижимости разметки целесообразно использовать дополненный алгоритм исчерпывающего поиска в ширину.

Особенностью классического алгоритма является последовательный просмотр и анализ достижимости временных разметок с целью определения для каждой анализируемой разметки множества непосредственно достижимых из нее временных разметок без возвращения к анализу уже просмотренных разметок. Диаграмма достижимости сети N строится последовательно с возрастающими моментами времени изменения разметок, начиная с начальной разметки $\vec{\mu}_0$, при этом переход к анализу достижимых разметок нижнего уровня осуществляется только после завершения анализа всех разметок верхнего уровня.

Для описания алгоритма воспользуемся следующими обозначениями:

$Q(\vec{\mu}_0)$ – множество достижимых разметок сети N;

$Q(\vec{\mu}_0, l)$ – множество достижимых разметок на -й итерации.

При этом множество достижимых разметок состоит из двух подмножеств

$$Q(\vec{\mu}_0, l) = \bar{Q}(\vec{\mu}_0, l) \cup \tilde{Q}(\vec{\mu}_0, l), \quad \bar{Q}(\vec{\mu}_0, l) \cap \tilde{Q}(\vec{\mu}_0, l) \neq \emptyset, \tag{7}$$

где: $\bar{Q}(\vec{\mu}_0, l)$ – множество просмотренных достижимых разметок на предыдущей итерации $l-1$, $\tilde{Q}(\vec{\mu}_0, l)$, – множество непросмотренных достижимых разметок на предыдущей итерации $l-1$.

С учетом особенности разметки сети N схема алгоритма построения диаграммы достижимости временных разметок методом поиска в ширину будет иметь вид, представленный на (Рис. 4).

Различие между классическим алгоритмом заключается в добавлении дополнительных блоков добавления маркеров воздействия и реагирования, а также проверки наличия во множествах векторов деструктивного воздействия Δ и реагирования R оставшихся, неиспользованных элементов.

Предлагаемый алгоритм построения диаграммы достижимости представляет собой итеративное повторение следующей последовательности шагов:

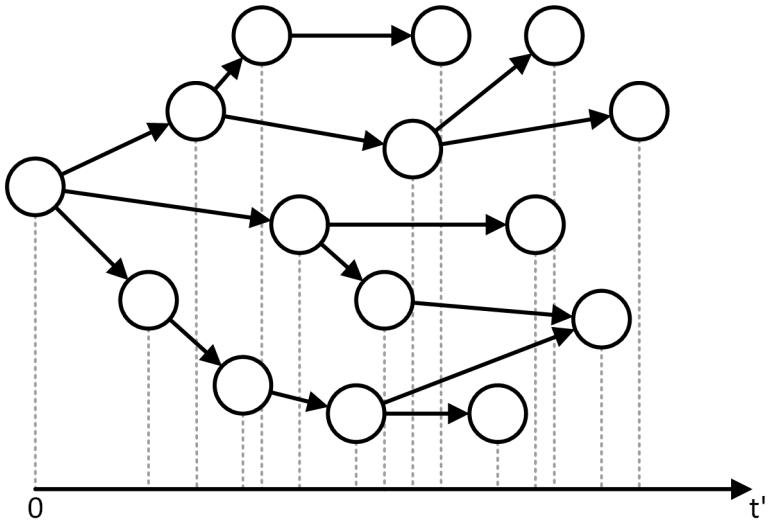

Рис. 3. Схема диаграммы достижимости временных разметок

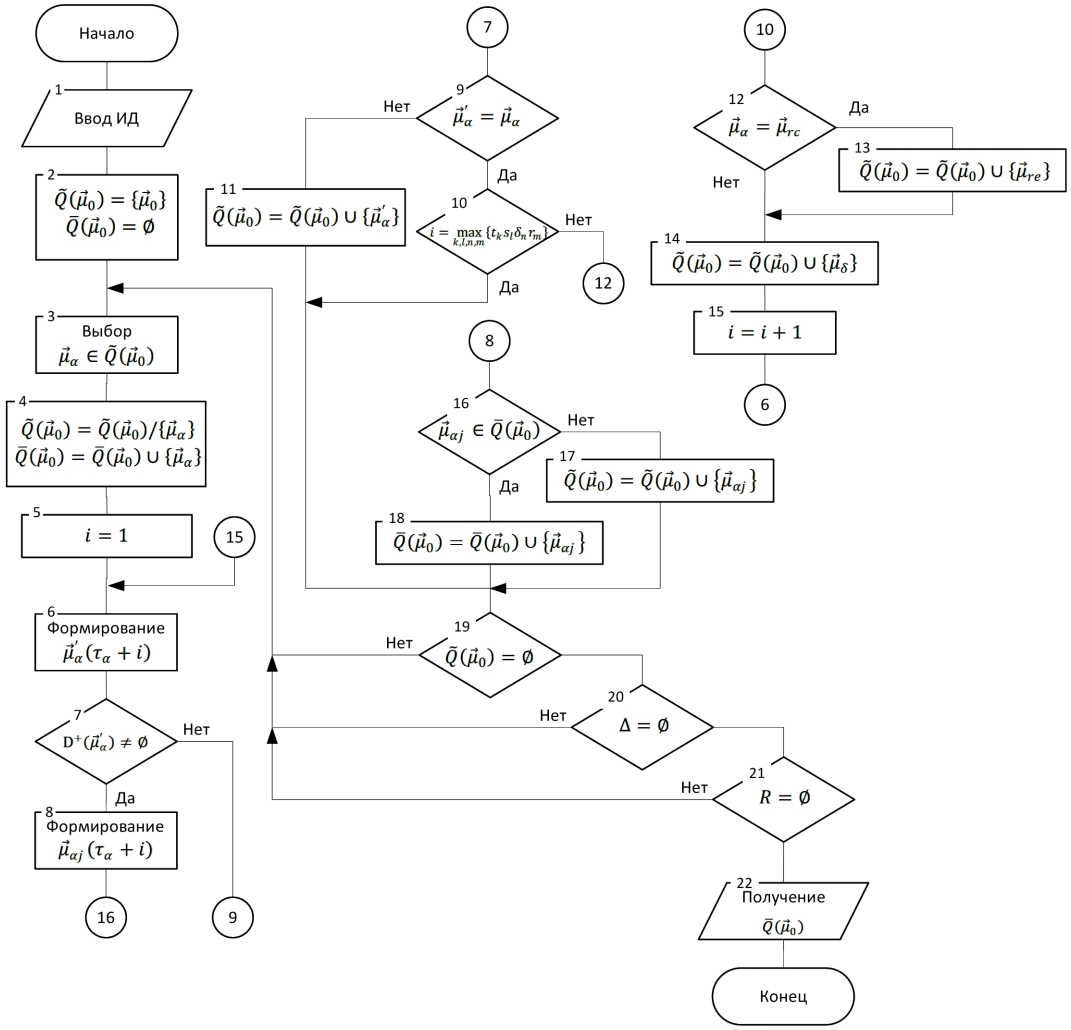

Рис. 4. Схема дополненного алгоритма построения диаграммы достижимости временных разметок методом поиска в ширину

Шаг 1.

В блоке 2 задается начальное множество достижимых разметок сети N: $Q(\vec{\mu}_0, l) = \{\vec{\mu}_0\}; \bar{Q}(\vec{\mu}_0, l) = \emptyset; \tilde{Q}(\vec{\mu}_0, l) = \{\vec{\mu}_0\}, l = 1.$

Переход к следующему шагу 2 к блоку 3.

Шаг 2.

В блоке 3 для анализа выбирается очередная непросмотренная разметка $\vec{\mu}_\alpha \in$. Для нее определяется момент времени ее достижения τ_α. Далее в блоке 4 происходит перенос $\vec{\mu}_\alpha$ из множества $\tilde{Q}(\vec{\mu}_0, l)$ во множество $\bar{Q}(\vec{\mu}_0, l)$: $\tilde{Q}(\vec{\mu}_0, l+1) = \tilde{Q}(\vec{\mu}_0, l) \backslash \{\vec{\mu}_\alpha\}, \bar{Q}(\vec{\mu}_0, l+1) = \bar{Q}(\vec{\mu}_0, l) \cup \{\vec{\mu}_\alpha\}$. В блоке 5 определяется начальное значение временного смещения момента времени $\tau_\alpha + i, i = 1.$

Переход к следующему шагу 3 к блоку 6.

Шаг 3.	В блоке 6 формируется разметка $\vec{\mu}'_\alpha(\tau_\alpha + i)$ путем проверки следующих условий: а. наличие для разметки $\vec{\mu}_\alpha$ активных переходов, время срабатывания которых заканчивается в момент времени $\tau + i$: $D^-_{\tau_\alpha+i}(\vec{\mu}_\alpha) \neq \emptyset$ б. наличие в сети N позиций, для которых в момент времени $\tau + i$ истекают временные задержки находящихся в них недоступных маркеров: $P_{\tau_\alpha+i}(\vec{\mu}_\alpha) \neq \emptyset$. При выполнении одного из условий образуется промежуточная разметка $\vec{\mu}'_\alpha(\tau_\alpha + i)$, в противном случае $\vec{\mu}'_\alpha(\tau_\alpha + i) = \vec{\mu}_\alpha$. Переход к следующему шагу 4 к блоку 7.
Шаг 4.	Для разметки $\vec{\mu}'_\alpha(\tau_\alpha + i)$ проверяется выполнение условия $D^+_{\tau_\alpha+i}(\vec{\mu}'_\alpha) \neq \emptyset$. При выполнении условия в блоке 8 образуется подмножество $D^+_{\tau_\alpha+i,1}(\vec{\mu}'_\alpha), D^+_{\tau_\alpha+i,2}(\vec{\mu}'_\alpha), \dots, D^+_{\tau_\alpha+i,c}(\vec{\mu}'_\alpha)$, сформировываются новые непосредственно достижимые разметки $\vec{\mu}'_{\alpha,1}(\tau_\alpha + i), \vec{\mu}'_{\alpha,2}(\tau_\alpha + i), \dots, \vec{\mu}'_{\alpha,c}(\tau_\alpha + i)$ и осуществляется переход к шагу 6 к блоку 16. В противном случае осуществляется переход к блоку 9 и проверка условия $\vec{\mu}_\alpha \neq \vec{\mu}'_\alpha$. При его выполнении в блоке 11 определяется $\vec{\mu}'_{\alpha,1}(\tau_\alpha + i) = \vec{\mu}'_\alpha$ и осуществляется переход к шагу 7 к блоку 19. В обратном случае осуществляется переход к шагу 5 к блоку 10.
Шаг 5.	В блоке 10 проверяется условие $i = \max\{t_k s_l \delta_n r_m\}$. При выполнении осуществляется переход к шагу 7 к блоку 19, в противном случае в блоке 12 осуществляется проверка текущей разметки на возможности срабатывания реагирования. В случае успеха в блоке 13 множество $\tilde{Q}(\vec{\mu}_0)$ дополняется разметкой реагирования. В блоке 14 с вероятностью P_δ возникает деструктивное воздействие на сеть N, дополняя множество $\tilde{Q}(\vec{\mu}_0)$. Независимо от результата работ блоков 12-14 в блоке 15 увеличивается значение смещения момента времени $i = i + 1$ и осуществляется возврат к шагу 3 к блоку 6.
Шаг 6.	В блоке 16 для каждой новой разметки $\vec{\mu}'_{\alpha,j}(\tau_\alpha + i)(\forall j \in \{1, \dots, c\})$ проверяется условие $\exists \vec{\mu}_\beta \in \bar{Q}(\vec{\mu}_0, l)$, такое, что $\vec{\mu}'_{\alpha,j} = \vec{\mu}_\beta$ и множества не закончивших срабатывание переходов для них совпадают. При выполнении условия в блоке 17 изменяется $\bar{Q}(\vec{\mu}_0, l + 1) = \bar{Q}(\vec{\mu}_0, l) \cup \{\vec{\mu}'_{\alpha,j}(\tau_\alpha + i)\}$, в противном случае в блоке 18 $\tilde{Q}(\vec{\mu}_0, l + 1) = \tilde{Q}(\vec{\mu}_0, l) \cup \{\vec{\mu}'_{\alpha,j}(\tau_\alpha + i)\}$. Переход к следующему шагу 7 к блоку 19.
Шаг 7.	В блоке 19, 20 и 21 проверяются условия $\tilde{Q}(\vec{\mu}_0, l + 1) \neq \emptyset$, $\Delta \neq \emptyset$, и $R \neq \emptyset$ соответственно. При выполнении любого условия $l = l + 1$ и осуществляется возврат к шагу 2 к блоку 3. В обратном случае осуществляется переход к блоку 22 и получение $\bar{Q}(\vec{\mu}_0)$. Выполнение алгоритма заканчивается.

Заключение

Представленная модель состояния работоспособности ИС и ИТКС позволяет моделировать изменение состояния сервисов и системы в целом, при случайном или заданном векторе деструктивного воздействия. Алгоритм позволяет построить диаграмму достижимости временных разметок, являющихся возможными состояниями защищаемого объекта. Алгоритм учитывает динамику возникновения деструктивного воздействия и реагирования на него имеющимися механизмами. Полученный результат возможно применить в системах поддержки принятия решения при обнаружении и предотвращении компьютерных атак, в том числе использующих новые уязвимости, для автоматизации процессов и оптимизации времени принятия решения.

Литература:

1. Бирюков Р.Е., Заминалов А.М., Заводцев И.В. Принятие решения о выборе эффективных стратегий для противодействия компьютерным атакам / Специальная связь и безопасность информации (ССБИ – 2016). 2016. С. 32-37.
2. Гаврилов Е.А. Исследование методов обнаружения сетевых атак / Научные записки молодых исследований. 2017. С. 55-58.
3. Кузнецов А.В. Взаимосвязь процесса управления событиями с другими процессами управления предприятия // Вопросы кибербезопасности. 2017. № 5 (24). С. 17-22. DOI: 10.21681/2311-3456-2017-5-17-22.
4. Ниссембаум О.В., Пономарев К.Ю. Методы и модели выявления событий нарушения информационной безопасности компьютерной сети / Безопасность информационного пространства. 2016. С. 169-174.
5. Петренко С.А., Цирлов В.Л. Импортозамещение решений IDS и SIEM // Защита информации. Инсайд. 2017. № 5 (77). С. 46-51.
6. Карташевский В.Г., Крыжановский А.В. Анализ методов и средств выявления инцидентов информационной безопасности / Вестник УРФО. Безопасность в информационной сфере. 2018. С. 50-54.
7. Васильев В.И., Шамсутдинов Р.Р. Интеллектуальная система анализа инцидентов информационной безопасности (на основе методологии SIEM-систем с применением механизмов иммунокомпьютинга) / Моделирование, оптимизация и информационные технологии. 2019. С. 536-547.
8. Баранова Е.К., Завадская Е.Д. Организация центра управления событиями информационной безопасности // Системы высокой доступности. 2018. С. 8-14.
9. Юнусова Д.С., Константинов Е.В., Рахматуллина А.Р. Современные системы обнаружения и предотвращения компьютерных атак // Актуальные проблемы социального, экономического и информационного развития современного общества. 2017. С. 327-332.
10. Бабошин В.А., Васильев В.А., Голубев В.Е. Обзор зарубежных и отечественных систем обнаружения компьютерных атак // Информация и космос. 2015. С. 36-41.
11. Мустафаев А.Г. Нейросетевая система обнаружения компьютерных атак на основе анализа сетевого трафика // Вопросы безопасности. 2016. С. 1-7.
12. Корчагина Е.В., Зыбин Д.Г., Бутова Л.В. // Разработка решений по обнаружению, блокированию (нейтрализации) компьютерных атак в информационных системах учреждений ФСИН России. 2017. С 88-94.
13. Косенкова Ю.И., Яковлев А.В. Разработка информационной модели системы обнаружения инцидентов информационной безопасности на основе анализа состояний системы // Информатика: проблемы, методология, технологии. 2017. С. 109-113.
14. Талыбов Н.Г., Мустафаев В.А., Гусейнов А.Г., Салманова М.Н. // Моделирование динамических взаимодействующих процессов с применением нечетких временных сетей Петри // Электротехнические и информационные комплексы и системы. 2017. №2. С.48-54.
15. Скородумов П.В. Анализ перспективных расширений сетей Петри // Наука и мир. 2014. № 10(14). С. 66-68.
16. Маршаков Д.В. Моделирование нечётких продукционных правил цветными сетями Петри // Математические методы в технике и технологиях – ММТТ. 2016. №1(83). С. 113-116.
17. Nesterov R.A., Lomazova I.A. Interface patterns for compositional discovery of distributed system models // Труды ИСП РАН. 2017. Том. 29, N. 4. С. 21-38. DOI: 10.15514/ispras-2017-29(4)-2.
18. Романов Д.О. О преобразовании сети Петри в нейронную сеть // Сборник научных трудов новосибирского государственного технического университета. 2016. С. 98-103.
19. Агроник А.Ю., Кочина Л.В., Хачумова М.Я. Имитационное моделирование и анализ технологических процессов сетями Петри // Приборы и системы. управление, контроль, диагностика. 2017. С. 19-28.
20. Кудряшова Е.С. Применение временных сетей Петри для разработки систем реального времени мониторинга состояния объектов // Ученые записки Комсомольского-на-Амуре государственного технического университета. 2014. № 1(17). С. 40-46.
21. Мустафаев В.А., Салманова М.Н. // Разработка модели функционирования обрабатывающего центра с применением нечетких временных сетей Петри. 2018. №3. С. 13-19.

УДК 004.056 DOI:10.21681/2311-3456-2019-6-79-91

ALGORITHM FOR CREATING THE MODEL OF ATTAINABILITY OF INFORMATION SYSTEM OPERABILITY STATUS MODEL

Bolychev M.V.[4], Miroshnichenko E.L.[5], Pasechnik R.M.[6]

The article presents the model of an information system operability state applicable to modelling of the state dynamic change under a computer attack and when responding to identification and prevention of destructive influence. The model has been developed based on the extended time colored functional fuzzy Petri net. According to its functionality, the model is divided into two nets – static and dynamic. The article includes the comprehensive description of a particular process of changing the net marking that considers peculiarities of the used Petri net. Moreover, the article suggests an algorithm to be used to create the attainability diagram for the net time tokens. The algorithm is based on the breadth-first search technique including additional blocks that take into account the destructive influence and response to changes in the information system state. The elaborated model of an information system operability state and diagram of time tokens attainability represent possible intermediate and final states of the information system state while considering the destructive influence and corresponding response. The research can be used to improve the methods of detection and response to modern computer attacks, as well as to develop a decision support system.

Keywords: *information security, computer attack, services, Petri nets, dynamic modeling, reachability diagram, destructive impact, response, countermeasures, detection.*

References

1. Biryukov R.E., Zaminalov A.M., Zavodcev I.V. Prinyatie resheniya o vybore effektivnyh strategij dlya protivodejstviya komp'yuternym atakam. Special'naya svyaz' i bezopasnost' informacii (SSBI – 2016). 2016. S. 32-37.
2. Gavrilov E.A. Issledovanie metodov obnaruzheniya setevyh atak. Nauchnye zapiski molodyh issledovanij. 2017. S. 55-58.
3. Kuznecov A.V. Vzaimosvyaz' processa upravleniya sobytiyami s drugimi processami upravleniya predpriyatiya. Voprosy kiberbezopasnosti [Cybersecurity issues]. 2017. № 5 (24). C. 17-22. DOI: 10.21681/2311-3456-2017-5-17-22.
4. Nissembaum O.V., Ponomarev K.YU. Metody i modeli vyyavleniya sobytij narusheniya informacionnoj bezopasnosti komp'yuternoj seti. Bezopasnost' informacionnogo prostranstva. 2016. S. 169-174.
5. Petrenko S.A., Cirlov V.L. Importozameshchenie reshenij IDS i SIEM. Zashchita informacii. Insajd. 2017. № 5 (77). S. 46-51.
6. Kartashevskij V.G., Kryzhanovskij A.V. Analiz metodov i sredstv vyyavleniya incidentov informacionnoj bezopasnosti. Vestnik URFO. Bezopasnost' v informacionnoj sfere. 2018. S. 50-54.
7. Vasil'ev V.I., SHamsutdinov R.R. Intellektual'naya sistema analiza incidentov informacionnoj bezopasnosti (na osnove metodologii SIEM-sistem s primeneniem mekhanizmov immunokomp'yutinga). Modelirovanie, optimizaciya i informacionnye tekhnologii. 2019. S. 536-547.
8. Baranova E.K., Zavadskaya E.D. Organizaciya centra upravleniya sobytiyami informacionnoj bezopasnosti. Sistemy vysokoj dostupnosti. 2018. S. 8-14.
9. YUnusova D.S., Konstantinov E.V., Rahmatullina A.R. Sovremennye sistemy obnaruzheniya i predotvrashcheniya komp'yuternyh atak. Aktual'nye problemy social'nogo, ekonomicheskogo i informacionnogo razvitiya sovremennogo obshchestva. 2017. S. 327-332.
10. Baboshin V.A., Vasil'ev V.A., Golubev V.E. Obzor zarubezhnyh i otechestvennyh sistem obnaruzheniya komp'yuternyh atak. Informaciya i kosmos. 2015. S. 36-41.
11. Mustafaev A.G. Nejrosetevaya sistema obnaruzheniya komp'yuternyh atak na osnove analiza setevogo trafika. Voprosy bezopasnosti. 2016. S. 1-7.
12. Korchagina E.V., Zybin D.G., Butova L.V.. Razrabotka reshenij po obnaruzheniyu, blokirovaniyu (nejtralizacii) komp'yuternyh atak v informacionnyh sistemah uchrezhdenij FSIN Rossii. 2017. S 88-94.
13. Kosenkova YU.I., YAkovlev A.V. Razrabotka informacionnoj modeli sistemy obnaruzheniya incidentov informacionnoj bezopasnosti na osnove analiza sostoyanij sistemy. Informatika: problemy, metodologiya, tekhnologii. 2017. S. 109-113.
14. Talybov N.G., Mustafaev V.A., Gusejnov A.G., Salmanova M.N.. Modelirovanie dinamicheskih vzaimodejstvuyushchih processov s primeneniem nechetkih vremennyh setej Petri. Elektrotekhnicheskie i informacionnye kompleksy i sistemy. 2017. №2. S.48-54.
15. Skorodumov P.V. Analiz perspektivnyh rasshirenij setej Petri. Nauka i mir. 2014. № 10(14). S. 66-68.
16. Marshakov D.V. Modelirovanie nechyotkih produkcionnyh pravil cvetnymi setyami Petri. Matematicheskie metody v tekhnike i tekhnologiyah – MMTT. 2016. №1(83). S. 113-116.

4 Maxim Bolychev, General Staff of the Armed Forces of the Russian Federation, Moscow, Russia. E-mail: stq-mak@yandex.ru

5 Evgeniy Miroshnichenko, Krasnodar Higher Military School, Krasnodar, Russia. E-mail: mirash_mel@mail.ru

6 Rodion Pasechnik, Krasnodar Higher Military School, Krasnodar, Russia. E-mail: rmpasechnik@mail.ru

17. Nesterov R.A., Lomazova I.A. Interface patterns for compositional discovery of distributed system models. Trudy ISP RAN. 2017. Tom. 29, N. 4. C. 21-38. DOI: 10.15514/ispras-2017-29(4)-2.
18. Romanov D.O. O preobrazovanii seti Petri v nejronnuyu set'. Sbornik nauchnyh trudov novosibirskogo gosudarstvennogo tekhnicheskogo universiteta. 2016. S. 98-103.
19. Agronik A.YU., Kochina L.V., Hachumova M.YA. Imitacionnoe modelirovanie i analiz tekhnologicheskih processov setyami Petri. Pribory i sistemy. upravlenie, kontrol', diagnostika. 2017. S. 19-28.
20. Kudryashova E.S. Primenenie vremennyh setej Petri dlya razrabotki sistem real'nogo vremeni monitoringa sostoyaniya ob"ektov. Uchenye zapiski komsomol'skogo-na-amure gosudarstvennogo tekhnicheskogo universiteta. 2014. № 1(17). S. 40-46.
21. Mustafaev V.A., Salmanova M.N.. Razrabotka modeli funkcionirovaniya obrabatyvayushchego centra s primeneniem nechetkih vremennyh setej Petri. 2018. №3. S. 13-19.