УДК 621.3

АНАЛІЗ ПРОЦЕСІВ НЕГАТИВНОГО ВПЛИВУ НА ІНФОРМАЦІЙНІ СИСТЕМИ УПРАВЛІННЯ ФРАГМЕНТАМИ ПОЛІГРАФІЧНИХ ТЕХНОЛОГІЙ

Б. В. Дурняк, Т. М. Майба

Українська академія друкарства, вул. Під Голоском, 19, м. Львів

У статті проаналізовані різноманітні фактори негативного впливу на інформаційну систему управління поліграфічним технологічним процесом. На основі проведених досліджень загроз і небезпеки описані типи можливих атак та запропоновані методи протидії зовнішнім негативним впливам на інформаційну систему управління.

Ключові слова: *математичні моделі виявлення загроз та небезпек в системах управління, поліграфічні технології, типи атак, захист і відновлення інформаційних систем управління.*

Постановка проблеми: Процес негативного впливу різних факторів на інформаційну систему управління *(ISU)* є досить складним і досить не визначеним з точки зору дії на неї зовнішніх атак. Окрім того існують внутрішні фактори пов'язані з загрозами і небезпеками, характерними для поліграфічного технологічного процесу. Без детального аналізу можливих загроз і небезпеки неможливо розробити ефективну систему захисту від різноманітних атак на всіх етапах інформаційної системи управління технологічним процесом поліграфічного виробництва. Сучасні інформаційні технології вносять нові вимоги до методів управління поліграфічними технологічними процесами. До таких вимог можна віднести виявлення усіх негативних впливів на процес управління і відповідно підвищення його ефективності.

Аналіз останніх досліджень та публікацій. Проблемами підвищення захисту інформаційних управляючих систем, для складних технологічних процесів, в тому числі, поліграфічної технології займається цілий ряд видатних вчених, до яких можна віднести: Б. В. Дурняка, М. І. Сеньківського, Л. Я. Сікору, М. М. Луцківа, Ю. В. Кравченка, Ю. М. Романишина, О. А. Машкова та ін. Водночас розвиток інформаційних технологій поліграфічного виробництва, зокрема використання зовнішніх і внутрішніх комп'ютерних мереж, вносять певні корективи в існуючі теорії захисту від негативних впливів та зовнішніх атак і потребують подальших досліджень.

Мета статті — дослідити науково-прикладну задачу побудови інформаційної технології виявлення негативних процесів інформаційних систем управління технологічним процесом і запропонувати алгоритми реалізації методів захисту від зовнішніх атак для систем управління друкарськими технологічними процесами.

Виклад основного матеріалу дослідження. Процес негативного впливу різних факторів на інформаційну систему управління *(ISU)* можна формально представити у вигляді наступної послідовності:

$$Np_i = Nb_i \rightarrow Z_i \rightarrow A_i \rightarrow Bz_i, \quad (1)$$

де Nb_i – небезпека, яка активізує і формує негативний процес Np_i;

Z_i– загроза, яка використовується, для реалізації Np_i;

A_i – атака, що реалізує негативні події в середовищі *ISU*;

Bz_i – зміна рівня безпеки, яка виникає в результаті реалізації події $A_i=a_{i1}*...*a_{ik}$.

Відсутність в (1) хоча б однієї складової призводить до неможливості зменшення рівня Bz_i. В рамках розвязування проблем захисту та в рамках їх дослідження, приймаються наступні положення.

Положення 1. В оточуючому середовищі довільної системи типу *ISU*, чи просто інформаційної системи (*IS*), завжди існують процеси, або об'єкти, які можуть бути джерелом небезпек Nb_i, для відповідних систем.

Положення 2. Існування Nb_i, як наслідок існування Np_i, завжди призводить до можливості активізації негативних дій на *IS* у вигляді атак A_i.

Положення 3. Будь-яка *IS* характеризується таким параметром, як загроза Za_i, яка представляє собою такий елемент *IS*, або такий процес, що реалізуються в *IS*, які можуть бути використані Nb_i, для реалізації відповідної атаки A_i.

Положення 4. Будь-який Np_i не мусить мати чітко визначеної цілі активізації негативного процесу у вигляді деякої небезпеки Nb_i.

Виходячи з наведених положень, можна стверджувати, що ціль, або $c_i(ISU)$, як деяка причина виникнення Nb_i і A_i може бути досить не визначеною, або взагалі не існувати. Такий висновок може виявитися досить незвичним особливо, якщо негативну дію тісно пов'язувати з наслідками цієї дії. Тому приймемо, що формування цілі негативного впливу на об'єкт *ISU* реалізується в процесі функціонування всіх компонент. Повний опис $c_i(ISU)$ формується на останньому етапі реалізації послідовності (1).

Для виникнення Nb_i в середовищі, що оточує *ISU*, не є конче потрібна ціль негативного впливу для фрагментів Nb_i. Такий негативний вплив може зініціюватися процесами, які є внутрішніми відносно зовнішнього середовища. В таких випадках, мова може йти про існування тієї чи іншої цілі негативної дії на *ISU* та про її наслідки і відповідно про зміну значення величини Bz_i. У зв'язку з цим, в склад системи захисту, або системи безпеки *ISU*, яка позначається *SUB*, крім засобів протидії атакам A_i входять наступні компоненти:

- моніторинг (*KM*) різних типів атак;
- розпізнавання типів атак (*RA*);
- цілі атаки ($c_i(ISU)$);
- визначення ризику, для процесу функціонування R(ISU);
- протидія атаці, або засоби захисту від атак (Za).

Формально, таку систему можна описати у вигляді співвідношення:

$$SIB = F[KM, RA, c_i(ISU), R(ISU), Za.$$

Насамперед при побудові математичної моделі необхідно провести якісний аналіз кожної з вищевказаних компонент. Компонента *КМ* виявляє можливі ознаки здійснення атак на *ISU*. Переважно, системи типу *КМ* будуються таким чином, що вони, маючи список ознак появи атак, розпізнають по цих ознаках, фактично, аномалії і порівнюють їх з елементами відповідного списку [1]. Такий підхід до реалізації *КМ* є дещо вузький, оскільки в його рамках існує можливість виявляти лише факт виникнення атаки, який може мати місце тільки в тому випадку, коли активізувалась небезпека Nb_i під дією негативного процесу Np_i у зовнішньому середовищі. Крім того, перш ніж Nb_i активізує атаку, вона повинна виявити в середовищі, або на границях системи загрози, які могли б бути використані для реалізації вибраної атаки A_i. Щоб не обмежувати можливості *КМ* тільки аналізом аномалій, які появляються на етапі A_i, співвідношення (1), в рамках *КМ* необхідно реалізовувати наступні функціональні можливості:

- виявляти загрози в , які можуть існувати з початку функціонування , які можуть виникати також в процесі функціонування ;
- дослідити аномалії, які можуть бути ознакою активізації атаки в середовищі ;
- прогнозувати можливість виникнення небезпек Nb_i в зовнішньому середовищі ISU .

Загрози являють собою певні особливості, або параметри, що характеризують ISU як об'єкт, який може бути атакований зовнішніми чинниками. Внутрішніх факторів, які можуть інтерпретуватися як загрози, розглядати не будемо, оскільки їх можна класифікувати як некоректні реалізації окремих елементів ISU . Приймемо, що Za можуть існувати тільки у певних елементах ISU , до яких відносяться:

- інформаційні входи і виходи між системою та зовнішнім середовищем, що оточує ISU ;
- внутрішні інформаційні процеси, що використовують вхідні дані, які поступають через захищені з'єднання;
- внутрішні дані, які змінюються в процесі функціонування системи.

Це означає, що в KM повинні існувати списки ознак аномалій у відповідних компонентах. У зв'язку з цим, виникає задача створення таких описів можливих аномалій, які дозволили б замінити список всіх можливих аномалій деякою загальною моделлю, що дозволяє формувати опис аномалій у відповідних елементах системи. Для масивів даних, що в процесі роботи ISU можуть мінятися, в якості моделі можна вибрати наступне. Однією з таких моделей може служити модель $M(Da)$, яка будується на основі алгоритмів швидкого пошуку заданих величин, чи фрагментів у невпорядкованій множині даних ($LSPD$). В цьому випадку, завдяки такому алгоритму моніторингу даних у відповідних файлах FD_i , алгоритм може полягати у пошуку граничних, або недопустимих даних. Пошук можна звести до виявлення хоча б одного недопустимого елемента даних у відповідному масиві [2]. При моніторингу процесів, які можуть

існувати в ISU і можуть являти собою Z_i, необхідно більш детально визначитися з уявленнями про те, в чому полягає загроза в цьому випадку. Загроза, в цьому випадку, може полягати у тому, що деяка функція $y = f(x_1,\ldots,x_n)$ може допускати відхилення, при своїй програмній реалізації від $y = F(x_1,\ldots,x_n)$. В більшості випадків, програмно реалізовані функції являють собою сукупність окремих перетворень різної природи, наприклад, аналітичних перетворень, дискретних перетворень, логічних перетворень і т.д. Тому, загроза, яку може являти собою функція $y = f(x_1,\ldots,x_n)$, може полягати у тому, що при деяких допустимих, з точки зору $y = F(x_1,\ldots,x_n)$, перетворень активізація відповідних процесів може призводити до виникнення недопустимих ситуацій в системі. Виявлення подібних ознак може полягати у реалізації тестування окремих фрагментів системи програм. Таке тестування реалізується в рамках тестової моделі, що входить в склад діагностичної моделі MDT [2].

Очевидно, що в межах кожного циклу моніторингу проводити описані вище перевірки не доцільно. Тому в рамках SIB формуються образи, або ознаки правильного функціонування окремих компонент системи. В цьому випадку до таких ознак можна віднести дані про виявлення чи не виявлення змін в масивах даних, що підлягають перевірці зі сторони KM. Це означає, що на кожному циклі функціонування окремої компоненти ISU, кожна програма передає результати свого функціонування не тільки до функціонально визначеного споживача цих результатів, а і в систему SUB, в якій формує роботу відповідного вузла. Отже KM в рамках одного циклу вибирає компоненту для перевірки, слід яких відхиляється від вибраних оцінок траєкторії сліду функціонування.

$$M(KM_0) = \Phi\{\left[F_X(x), F_Y(y)\right], L[i,(x_i \vee y_i)]\},$$

де $F_X(x)$ – функція розподілу ймовірності виникнення аномалії в масиві даних,

$\left[F_Y(y)\right]$ – функція розподілу ймовірності виникнення аномалії у функціональних фрагментах,

$L\left[i,(x_i \vee y_i)\right]$ – алгоритм ідентифікації аномалій, ймовірність яких визначається на основі $F_X(x), F_Y(y)$.

Для порівняння поточних величин використовується L_P-метрика:

$$(E\left[\left|x - x_i\right|^P\right])^{1/P} = \{\sum_{x,x_0 \in X \times X} \left|x - x_i\right|^P \cdot P(x, x_i)\}^{1/P},$$

де $x \subset R$, $E\{[\left|z_i\right|^P\} < \infty$, для всіх $z_i \in Z$. В цьому випадку x і x_i означають суміжні значення параметрів, що характеризують виникнення аномалій в файлах даних, чи у функціональних фрагментах $y = \varphi_i(x)$, де φ_i – може бути комбінованою функцією. В якості функцій розподілу вибирається функція Пуассона, яка в найпростішому випадку записується у вигляді співвідношення [3]:

$$P(A_m) = [[\frac{\left(\lambda\left|\tau\right|\right)^n\right]}{n!}] \cdot e^{-\lambda|\tau|},$$

де λ – інтенсивність потоку подій на інтервалі часу τ, яка допускає інтерпретацію ймовірності виникнення події A_m на інтервалі τ. Особливість функціонування SUB полягає у тому, що кожна подія, що використовується в алгоритмах SUB реєструється, що приводить до накопичення випадкових величин, які використовуються у співвідношеннях, по яких реалізуються перевірки критеріїв, для прийняття тих, чи інших рішень.

Завдяки вищенаведеним співвідношенням, в процесі моніторингу стає можливим оптимізувати процес аналізу кожного пункту, де може виникнути аномалія за рахунок недопустимих змін в значеннях параметра, який, для вибраної компоненти, є критичним. Необхідність перевірки окремого елемента в процесі моніторингу визначається на основі обчислення L_p віддалі між вимірюваними величинами.

Компонента RA проводить розпізнавання типу атак на основі даних, що отримані, при моніторингу. В результаті моніторингу в SUB реєструється тип виявленої аномалії та місце її знаходження. Для того, щоб можна було виконати функції RA, необхідно визначити наступні фактори, що безпосередньо звязані з RA:

- типи атак, що характеризуються відповідними властивостями, які можуть бути розпізнані в рамках можливостей RA;
- міру адекватності результатів розпізнавання реальній атаці;
- необхідні методи протидії виявленим атакам та необхідну міру елімінації відповідної атаки.
- Розподіл атак на окремі типи може проводитися у відповідності з різними критеріями:
- за характером реалізації процесу атаки;
- за ціллю, з якою атака активізується;
- за змінами, до яких призводить реалізація атаки в середовищі ISU.

В рамках компоненти моніторингу визначення цілі атаки не передбачається. Цю задачу, в певній мірі, передбачається розв'язувати засобами RA. Тому, дані про ціль атаки A_i не можуть бути вхідними даними для RA. Оскільки процес моніторингу реалізується в середовищі ISU, то можна говорити про траєкторію моніторингу. Траєкторія моніторингу $\mathrm{I}(Mo)$ реалізується по окремих елементах системи. Основою побудови такої траєкторії в момент $t_i \in T$, де T – період циклу функціонування ISU, є функціональні зв'язки між відповідними елементами, що активізуються в інтервалі часу $\Delta\mathrm{T}$ функціонування всього TPP. Очевидно, що функціональні зв'язки, можна представити деякою сукупністю траєкторій, які, у відповідності з вимогами TPP, активізуються різними способами. Таким чином, траєкторія моніторингу насамперед аналізує функціонально активні фрагменти об'єкту, який захищається. Формально активні траєкторії можна представити у вигляді деякої графової структури $G\left(e_{i1}^{a} * \ldots * e_{im}^{a}\right)$. Така структура на кожному інтервалі часу δt визначається функцією управління, яку можна описати наступним співвідношенням:

$$G\left(e_i \in E\right) = \forall(t_{ij} \in \Delta_i T) F_1[u_1\left(e_{11}^a * \ldots * e_{1k}^a\right) * \ldots * u_m\left(e_{m1}^a * \ldots * e_{mk}^a\right)],$$

де u_i – фрагмент управляючого процесу, що реалізується в інтервалі;

Ce_{ij}^a – активний елемент середовища ISU .

В рамках таких позначень тип A_i , який визначається траєкторією активності $u_{ij} \in U\left(ISU\right)$. В цьому випадку засоби розпізнавання типу атак, або підсистема RA, буде проводити перевірку відповідного фрагменту з ціллю розпізнавання в ньому атаки шляхом виявлення аномалій an_i в напрямку траєкторії управління $U[\varphi\left(ISU\right)]$, де $\varphi\left(ISU\right)$ – фрагмент траєкторії управління. Наведемо наступне визначення.

Визначення 1. Якщо траєкторія реалізації атаки $\tau(A_i)$ визначається активною траєкторією процесу функціонування $\tau(U\left(ISU\right))$, то тип атаки визначається ціллю, що обумовлює активізацію відповідного процесу управління функціонуванням технологічного процесу.

Формально, це описується наступним співвідношенням:

$$\forall(u_{ij} \in U)\exists(c_j \to u_j)[u_j(e_{j1}^a * \ldots * e_{jm}^a) \to A_i(g_j),$$

де $A_i(g_j)$ – атака i типу g_j ,

g_j – ідентифікатор атаки та має місце співвідношення $g_j = c_j \in C$;

C – ціль функціонування відповідного TPP в період ΔT_i . Величина g_j служить для ідентифікації типу цілі функціонування фрагменту типу A_i , який визначається за типом цілі функціонування фрагмента TPP , який відповідає траєкторії управління $G(u_j) \to \tau(u_j)$.

В літературі, присвяченій проблемам захисту ISU , широко використовується класифікація атак A_i по тих цілях, які атака повинна реалізувати [4]. Можна прийняти, що ціллю реалізації атаки є $A_i(g_j) \to \neg c_i(TPP)$, що означає не допущення реалізації цілі c_i відповідним фрагментом TPP з допомогою атаки A_i . Завдяки такому розпізнаванню A_i є можливим протидіяти відповідним атакам.

В рамках такої класифікації атак, їх ідентифікація дозволяє визначити міру адекватності виявленого типу атаки реальній атаці, що активізувалась в ISU . Така задача виникає в наслідок наступних причин:

- атака A_i може реалізовуватися в різних $\tau(u_i)$,
- реалізація атаки A_i може не синхронізуватися тільки з одним з процесів управління u_i , що реалізується в ISU в межах інтервалу ΔT_i ,
- атака може мати ціль $C(A_i)$, яка не пов'язується з ціллю реалізації фрагмента $u_i \to c_i$, а призводить до розв'язку нової для ISU задачі, що формально описується співвідношенням:

$$C\left(A_i\right) \to \{c_i\left(u_i\right) = \left[c_i^*\left(a_j\right) * \ldots * c_i\left(a_{j+m}\right)\right]\}.$$

Очевидно, що атаки такого типу є складними як по своїй реалізації, так і по відношенню до забезпечення досягнення відповідної цілі. Такого типу атаки при-

значені для зміни цілі функціонування TPP, яка була визначена в рамках ISU на період часу ΔT_i. На якісному рівні, реалізацію такої атаки можна розглядати як підміну фрагментів процесу управління таким чином, щоб відповідна підміна була коректною в рамках ISU. Тому детально такого типу атаки розглядати не будемо.

Класифікація атак по змінах в середовищі ISU, до яких вони призводять, тісно пов'язана з класифікацією атаки по типу цілі атаки $C(A_i)$, оскільки, будь-які зміни в ISU, призводять до змін в множині цілей $C = \{c_1, \ldots, c_m\}$. Це співпадає з уже проаналізованим типом атак.

Компонента протидії виявленій атаці $Za(a_i)$, де a_i ідентифікатор типу атаки, є досить складною. Тому проведемо аналіз задач, які повинні розв'язуватися в рамках Za та способи їх реалізації. До таких задач можна віднести:

- визначення адекватності розпізнаного типу атаки $A_i(g_i)$ атаці, яка була активізована в середовищі ISU;
- визначення способу протидії атаці;
- забезпечення неможливості повторення атаки цього типу;
- визначення ефективності реалізованій по відношенню до виявленої атаки, протидії;
- визначення міри часткової протидії активізованій атаці.

Компонента Za функціонує на основі використання даних, які ця компонента отримує від компоненти RA. Очевидно, що RA може не провести повне розпізнавання A_i, що може обумовлюватися необхідністю розширення функціональних можливостей компоненти RA. В багатьох випадках це є не можливим. Тому, Za повинно визначати міру адекватності $A_i(g_i)$. Така міра адекватності визначається на основі наступних операцій, які виконує Za_i:

- обчислюється довжина траєкторії фрагмента $u_i \in U$, де була розпізнана несправність;
- вимірюється довжина траєкторії $\tau(u_i)$ та траєкторії $\tau(a_i)$, що визначена засобом RA.

Якщо $\tau(u_i) - \tau(a_i) > 0{,}5\ \tau(u_i)$, то адекватність результату розпізнавання A_i приписується категорія $\aleph(a_i) = m_i$, де m_i – ідентифікатор адекватності. Якщо $\tau(u_i) - \tau(a_i) < 0{,}5\ \tau(u_i)$, то $\aleph(a_i) = m_k$. Такі вимірювання можуть проводитися з ціллю виявлення різних мір адекватності розпізнавання A_i. Кількість таких мір $\{m_1, \ldots, m_n\}$ визначається можливостями кожної окремої компоненти Za_i по протидії відповідній атаці.

Методи реалізації протидії атакам, що використовуються компонентами Za_i, залежать від типів атак, які визначаються на основі синтезу ознак, одна з яких є траєкторією реалізації атаки. Очевидно, що кількість параметрів, що використовуються для ідентифікації атак, може бути більшою, що залежить від небезпеки, яка формує відповідну атаку. До поширених методів протидії атакам, що реалізуються засобами захисту Za_i, відносяться наступні:

- відновлення змін, які вносяться атаками в середовище ISU;
- блокування модифікованих фрагментів процесу управління, які були змінені атакою;

- імітація дії атаки, яка була виявлена засобами RA;
- модифікація виявленої атаки;
- дублювання функціональних фрагментів процесу функціонування ISU оригінальними фрагментами, які не були модифіковані атакою, оскільки мали статус резерву.

Відновлення змін внесених у фрагменти $u_i \in U$ є можливим в тому випадку, коли на основі аналізу оточення відповідного фрагменту можна відновити модифіковані частини. Оскільки модифікації атакою підлягають у більшості випадків логічні елементи програмних фрагментів, або дані, що отримані в результаті функціонування окремих фрагментів, то в рамках засобів Za_i реалізуються наступні методи відновлення: метод відновлення логіки функціонування фрагменту програмних засобів; метод відновлення змінених даних.

Метод відновлення логіки функціонування ґрунтується на використанні системи логічного виводу, що адаптований до формул описів логічних функцій, що використовуються в мові, на якій спроектовані програмні засоби. Такі методи широко використовуються в різних задачах, що потребують формування логічних структур [5].

Система відновлення модифікованого фрагменту ґрунтується на використанні адаптованих правил виводу логічних схем програмних засобів, що складають безпосереднє оточення відповідного фрагменту програми та ґрунтується на використанні логічного образу фрагменту цілі $c_i \in C$, який безпосередньо залежить від відновлюваного фрагменту $u_i \in U$. Розглянемо формальні аспекти реалізації цього методу. Оточуючі фрагменти реалізації програмних засобів можна представити у вигляді системи таких співвідношень:

$$\left.\begin{array}{l} u_{i-1}(e_{(i-1),1} * \ldots * e_{(i-1),k}) \\ u_{i+1}(e_{(i+1),1} * \ldots * e_{(i+1),m}) \\ u_i^*(e_{i1} * \ldots * e_{is}) \\ c_i(e_{i1}^c * \ldots * e_{ir}^c) \end{array}\right\} \qquad (2)$$

Правила реалізації виводу $u_i(e_{i1} * \ldots * e_{is})$ з системи (2) можна записати у вигляді наступного співвідношення:

$$\Omega = L\{\omega_1, \ldots, \omega_n\}, \qquad (3)$$

де ω_i – правила виводу адаптовані до способу реалізації логічних операторів, що використовуються в мові програмування, на якій сформовано u_{i-1}, u_i^* u_{i+1}.

Розглянемо наступне твердження.

Твердження 1. Модифікувати логіку фрагменту u_i^* з ціллю її відновлення є можливим, якщо найближче логічне оточення u_{i-1} та u_{i+1} орієнтоване на реалізацію c_i, яка є спільною для фрагментів u_{i-1}, u_i^* u_{i+1} та існує не суперечлива система виводу Ω.

Приймемо, що структура $\varphi(u) = u_{i-1}$, $u_i^*\ u_{i+1}$ є лінійною. Дійсно, якщо зв'язок між u_{i-1} * u_i^* не існує, або $\neg(u_{i-1}$ * $u_i^*)$, то існує в u деякий фрагмент u_k, який є лінійно з'єднаний з u_i^*, або u_k * u_i^*, якщо має місце $\neg(u_k$ * $u_i^*)$, то u_i^* є початковим фрагментом в U. Тоді, згідно з твердженням, перейдемо до $\ddot{O} = \varphi(u) = u_{i-1}^* * u_i^{**} * u_{i+1}$, що відповідає заміні $u_i^* \to u_i^{**}$. Отже існує такий фрагмент $\varphi^*(u)$, для якого має місце $\varphi^*(u) = u_{i-1}^{'} * u_i^{**} * u_{i+1}^{'}$. Згідно з умовою твердження, має місце $u_{i-1}^{'} \to c_i^1$, $u_i^* \to c_i^2$, $u_{i+1}^{'} \to c_i^3$, де $\tilde{n}_i = c_i^1 \,\&\, c_i^2 \,\&\, c_i^3$, оскільки $\tilde{n}_i$ є однією ціллю, для $u_{i-1}^{'}, u_i^{**}, u_{i+1}^{'}$. Система виводу $\Omega = F_s[\Gamma]$, де Γ – система виводу Генцена, яка є не суперечна. Функція F_S є функцією структурною, яка реалізує логічних формул A на B, а $A \Rightarrow B$ є підстановка B замість A. Ці операції виводяться з Γ і тому розширення ними Γ не призводить до виникнення суперечностей в Γ. Приймемо, що існує вивід $[(u_{i-1}^{'}, u_i^{**}, u_{i+1}^{'})$ $\&\Gamma] \to C$, або можна записати, що $\Gamma[(u_{i-1}^{'}, u_i^{**}, u_{i+1}^{'}) \to c_i$. Оскільки $u_{i-1}^{'} * u_i^{**}$ використовують операції з системи $L = \{\&, \vee, \to, \neg\}$, яка є повною, то можна записати співвідношення:

$$\Gamma[u_{i-1}^{'}, u_i^{**}, u_{i+1}^{'}] \to \Gamma[(u_{i-1}^{'}, u_i^{**}, u_{i+1}^{'}) \to c_i].$$

Якщо прийняти $[(u_{i-1}^{'} \to u_{i-1}) \,\&\, (u_i^{**} \to u_i) \,\&\, (u_{i+1}^{'} \to u_{i+1})]$ і $\Gamma \subset \Omega$, то можна записати, що $[\Omega[u_{i-1} * u_i * u_{i+1}] \to c_i] \to (u_i^* \to u_i)$, що доводить твердження.

Метод відновлення змін, що пов'язані з даними, можна реалізовувати різними способами, до яких можна віднести:

- використання тестових даних;
- вибір даних, які узгоджуються з даними, що використовувались на попередніх циклах;
- формування даних, що найбільше відповідають відповідній підцілі;
- блокування використання змінених даних.

Висновки. Тестові дані, якими проектується заміняти змінені в результаті дії атаки дані, являють собою не ті дані, що використовуються для перевірки окремих програм, а ті, які використовуються для тестування цілого технологічного процесу. Це означає, що така заміна не призведе до недопустимих змін у функціонуванні фрагменту технологічного процесу.

Оскільки, окремі фрагменти повторюють своє функціонування від циклу до циклу, то вибір даних з попереднього циклу, який функціонально є тим самим, не призведе до недопустимих відхилень в досягненні цілі. Узгодження нових даних з ціллю очевидно, призведе до недопустимого способу функціонування системи.

Список використаних джерел

1. Ирвин Дж. Передача данных в сетях: инженерный подход / Дж. Ирвин, Д. Харль. – СПб. : БХВ-Петербург, 2003.
2. Макконел Дж. Анализ алгоритмов. Активный обучающий подход / Дж. Макконел. – М. : Техносфера, 2009. – 416 с.
3. Непомнящий В. А. Прикладные методы верификации программ / В. А. Непомнящий, О. М. Рякин. – М. : Радио и связь, 1988.

4. Лукацкий А. Обнаружение атак / А. Лукацкий. – СПб. : БХИ-Петербург, 2001. – 624 с.
5. Карри Х. Б. Основания математической логики / Х. Б. Карри. – М. : Мир, 1969.

References

1. Irvin Dzh. (2003), Peredacha dannyh v setjah: inzhenernyj podhod / Dzh. Irvin, D. Harl'. – SPb. : BHV-Peterburg. (in Russian)
2. Makkonel Dzh. (2009), Analiz algoritmov. Aktivnyj obuchajushhij podhod / Dzh. Makkonel. – M. : Tehnosfera. – 416 s. (in Russian)
3. Nepomnjashhij V. A. (1988), Prikladnye metody verifikacii programm / V. A. Nepomnjashhij, O. M. Rjakin. – M. : Radio i svjaz'. (in Russian)
4. Lukackij A. (2001), Obnaruzhenie atak / A. Lukackij. – SPb. : BHI-Peterburg. – 624 s. (in Russian)
5. Karri H. B. (1969), Osnovanija matematicheskoj logiki / H. B. Karri. – M. : Mir. (in Russian)

ANALYSIS OF PROCESSES OF NEGATIVE IMPACT ON INFORMATION MANAGEMENT SYSTEMS BY FRAGMENTS OF PRINTING TECHNOLOGIES

B.V. Durniak, T. M. Maiba

Ukrainian Academy of Printing 19, Pid Holoskom St., Lviv

The article analyzes various factors of negative impact on the information management system of the printing technological process. It describes types of potential attacks and offers methods of combating external negative impacts on information management system basing on the researches of threats and dangers

Key words: *mathematical models of threats and dangers detection in management systems, printing technologies, types of attacks, protection and restoration of information systems.*

Стаття надійшла до редакції 22.01.2015
Received 22.01.2015